



Zawiera CD

 WILEY

Wyczerpujące źródło wiedzy
o fenomenalnej dystrybucji Linuksa

William von Hagen

Ubuntu[®] Linux[®]

Zainstaluj i skonfiguruj

Ubuntu oraz środowisko
graficzne GNOME

Wykorzystaj

w pracy oprogramowanie
dołączone do Ubuntu

Napisz

własne oprogramowanie dla Ubuntu

Naucz się

administrować systemem Ubuntu



Biblia

Wiedza obiecana

Tytuł oryginału: Ubuntu Linux. Biblia
Tłumaczenie: Adam Bąk (wstęp, rozdz. 1–18)
Marek Pętflicki (rozdz. 19–32, dodatek)

ISBN: 978-83-246-8452-6

All rights reserved. This translation published under license with the original publisher John Wiley & Sons, Inc. Copyright © 2007 by Wiley Publishing, Inc., Indianapolis, Indiana

The Bible Brand trade dress is a trademark of John Wiley & Sons, Inc. in the United States and/or other countries. Used by permission.

Wiley and the Wiley logo, and related trade dress are trademarks or registered trademarks of John Wiley and Sons, Inc. and/or its affiliates in the United States and other countries, and may not be used without written permission. Ubuntu is a trademark of Canonical Limited. Linux is registered trademark of Linus Torvalds. All other trademarks are the property of their respective owners. Wiley Publishing Inc. is not associated with any product or vendor mentioned in this book.

Translation copyright © 2008 by Wydawnictwo Helion.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 032 231 22 19, 032 230 98 63
e-mail: helion@helion.pl
WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Dodatkowe materiały do książki można znaleźć pod adresem:
ftp://ftp.helion.pl/przyklady/ubunbi_ebook.zip Rozmiar pliku: 681 MB

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

http://helion.pl/user/opinie?ubunbi_ebook

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

- [Poleć książkę na Facebook.com](#)
- [Kup w wersji papierowej](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Dorothy — więcej niż mogą słowa...
Quidquid latine dictum sit, altum sonatur

Rzut oka na książkę

O autorze	19
Podziękowania	21
Wprowadzenie	23
Część I Rozpoczęcie pracy z Ubuntu	29
Rozdział 1. Projekt Ubuntu	31
Rozdział 2. Instalowanie Ubuntu	53
Rozdział 3. Instalowanie Ubuntu jako systemu specjalnego zastosowania	81
Część II Korzystanie z Ubuntu	119
Rozdział 4. Podstawowe założenia systemu Linux	121
Rozdział 5. Środowisko GNOME	143
Rozdział 6. Korzystanie z linii poleceń	185
Rozdział 7. Praca z plikami tekstowymi w Ubuntu	219
Rozdział 8. Obsługa poczty elektronicznej za pomocą programu Evolution	251
Rozdział 9. Surfowanie z Firefoksem	281
Rozdział 10. Tworzenie i publikowanie dokumentów	305
Rozdział 11. Pozostałe składniki pakietu biurowego: arkusz kalkulacyjny Calc i program do tworzenia prezentacji	343
Rozdział 12. Praca z grafiką	383
Rozdział 13. Multimedia	407
Rozdział 14. Zagrasz?	449

Rozdział 15. Łączenie się z innymi systemami	473
Rozdział 16. Przesyłanie i udostępnianie plików w Ubuntu	491
Rozdział 17. Urządzenia elektroniczne w Ubuntu	513
Rozdział 18. Tworzenie oprogramowania w Ubuntu	545
Część III Ubuntu dla administratorów systemu	581
Rozdział 19. Proces rozruchu i zamykania systemu Ubuntu	583
Rozdział 20. Dodawanie, usuwanie i aktualizacja oprogramowania	601
Rozdział 21. Zarządzanie użytkownikami, grupami i zaawansowane uprawnienia	649
Rozdział 22. Wykonywanie i przywracanie kopii zapasowych	681
Rozdział 23. Dodawanie sprzętu i przyłączanie urządzeń peryferyjnych ...	717
Rozdział 24. Konfiguracja sieci i bezpieczeństwo	749
Rozdział 25. Sieć bez kabli	773
Część IV Konfiguracja serwerów w Ubuntu	789
Rozdział 26. Konfiguracja serwera WWW	791
Rozdział 27. Konfiguracja serwera poczty	807
Rozdział 28. Konfiguracja serwera DHCP	833
Rozdział 29. Konfiguracja serwera DNS	847
Rozdział 30. Konfiguracja serwera wydruków	867
Rozdział 31. Konfiguracja serwera NFS	879
Rozdział 32. Konfiguracja serwera Samba	895
Dodatki	911
Zawartość dysku CD	913
Skorowidz	919

Spis treści

O autorze	19
Podziękowania	21
Wprowadzenie	23
Część I Rozpoczęcie pracy z Ubuntu	29
Rozdział 1. Projekt Ubuntu	31
Wprowadzenie	32
Dlaczego warto korzystać z Linuksa?	32
Co to jest dystrybucja?	34
Wprowadzenie do Ubuntu	35
Manifest Ubuntu	36
Kalendarz wydań Ubuntu	37
Uaktualnienia i wsparcie systemu w Ubuntu	38
Ubuntu a Debian	39
Dlaczego warto wybrać Ubuntu?	40
Wymagania instalacyjne	41
Obsługiwane architektury	42
Wymagania sprzętowe	42
Czas instalacji	43
Płyty instalacyjne Ubuntu	43
Wsparcie techniczne dla Ubuntu	44
Społeczność Ubuntu — wsparcie i informacje	45
Dokumentacja	48
Komercyjne wsparcie techniczne dla Ubuntu	49
Uzyskiwanie dodatkowych informacji o Ubuntu	51
Podsumowanie	52
Rozdział 2. Instalowanie Ubuntu	53
Uzyskiwanie płyt Desktop CD dla komputerów 64-bitowych i PPC	54
Uruchamianie systemu z płyty Desktop CD	55
Instalacja Ubuntu z płyty Desktop CD	56
Uruchamianie Ubuntu	65
Uruchamianie Ubuntu w trybie podwójnego rozruchu	65
Pierwsze uruchomienie Ubuntu	66
Testowanie Ubuntu	66
Zawartość folderu Examples	67
Uzyskiwanie dostępu do dysku twardego z systemu Desktop CD	69

Tryb persistence	73
Kopiowanie plików do innego komputera dostępnego w sieci	75
Instalacja programów dla Windows umieszczonych na płycie CD	76
Podsumowanie	78

Rozdział 3. Instalowanie Ubuntu

jako systemu specjalnego zastosowania 81

Tryb podwójnego rozruchu — informacje	82
Proces uruchamiania komputera	82
Przygotowywanie systemu do uruchomienia trybu podwójnego rozruchu	83
Repartycjonowanie zamontowanego dysku	84
Pozyskiwanie innych płyt instalacyjnych	93
Płyty w wersji Server i Alternate Install CD — uruchamianie systemu	93
Menu instalacyjne płyty Server Install CD	94
Instalacja serwera Ubuntu	96
Określanie układu partycji	104
Instalowanie serwera LAMP	113
Pierwsze uruchomienie systemu	113
Opcje instalacyjne dla płyty Alternate Install CD	114
Instalowanie systemu Ubuntu Desktop w trybie tekstowym	115
Instalowanie Ubuntu w trybie OEM	116
Instalowanie serwera z płyty Alternate Install CD	117
Podsumowanie	117

Część II Korzystanie z Ubuntu 119

Rozdział 4. Podstawowe założenia systemu Linux 121

Praca z plikami i katalogami	122
Katalogi systemu Linux	123
Inne katalogi systemu Linux	124
Wprowadzenie do systemu plików w Linuksie	124
Dyski, partycje i punkty montowania	125
Lokalne systemy plików: standardowy i z kroniką	126
Sieciowe systemy plików	128
Praca z partycjami i systemami plików	129
Montowanie systemu plików	129
Automatyczne montowanie systemów plików podczas uruchamiania komputera	132
Automatyczne montowanie systemów plików znajdujących się na przenośnych nośnikach danych	135
Podstawowe informacje o uprawnieniach w systemach Linux	136
Podstawowe koncepcje: użytkownicy i grupy	136
Uprawnienia do plików i katalogów w Linuksie	138
Nadawanie uprawnień domyślnych podczas tworzenia plików i katalogów	139
Wykonywanie czynności wymagających specjalnych uprawnień	139
Podsumowanie	141

Rozdział 5. Środowisko GNOME 143

Co to jest pulpit? Środowiska graficzne dla Linuksa	143
Korzystanie z myszy	145
Wprowadzenie do GNOME	147
Okno aplikacji GNOME	149

Menu w GNOME	150
Menu na panelach	150
Menu kontekstowe	153
Dostosowywanie menu	153
Dostosowywanie pulpitu	159
Dostosowywanie działania myszy	159
Konfigurowanie rozdzielczości wyświetlania	161
Dostosowywanie paneli	162
Konfigurowanie wygaszacza ekranu	167
Zmiana tła pulpitu	169
Przełączanie motywów	170
Technologie wspierające w GNOME	171
Skróty klawiaturowe w GNOME	173
Menedżer plików Nautilus — wprowadzenie	174
Podstawowe operacje w Nautilusie	174
Przykłady użycia Nautilusa	175
Dodatkowe informacje o Nautilusie	178
Korzystanie z menedżera okien	178
Podsumowanie	183
Rozdział 6. Korzystanie z linii poleceń	185
Po co używać linii poleceń?	186
Wykonywanie komend w linii poleceń	187
Co to jest powłoka?	191
W głąb powłoki	195
Korzystanie z terminala w GNOME	195
Korzystanie z terminala w systemie X Window System	196
Często używane komendy linii poleceń	199
Poruszanie się po systemie plików Linuksa	200
Kopiowanie, przenoszenie, zmiana nazwy i usuwanie plików i katalogów	200
Zmiana uprawnień do plików i katalogów	204
Wyszukiwanie odpowiednich poleceń	206
Korzystanie z powłoki bash	207
Korzystanie z historii poleceń	208
Uzupełnianie nazw plików	209
Wyrażenie wieloznaczne	210
Potoki i przekierowanie danych	211
Wprowadzenie do kontroli zadań	213
Eksploracja pliku konfiguracyjnego powłoki	214
Zmienne środowiskowe	215
Definiowanie i używanie aliasów	217
Podsumowanie	217
Rozdział 7. Praca z plikami tekstowymi w Ubuntu	219
Wprowadzenie do edytorów tekstowych w Linuksie	220
Korzystanie z edytora vi	221
Uruchamianie i wyłączanie edytora vi	223
Wstawianie tekstu w edytorze vi	224
Przemieszczanie kursora w edytorze vi	225
Usuwanie i zmienianie tekstu w edytorze vi	226
Wycinanie, kopiowanie i wstawianie tekstu w edytorze vi	227

Wyszukiwanie i zastępowanie tekstu w edytorze vi	228
Cofanie zmian w edytorze vi	229
Korzystanie z wielu okien w programie vi	229
Dostosowywanie edytora vim	231
Graficzne wersje edytora vi	232
Dodatkowe informacje na temat edytorów vi i vim	233
Korzystanie z programu emacs	233
Kilka słów o poleceniach w edytorze emacs	236
Terminologia	238
Uruchamianie i wyłączanie edytora emacs	238
Poruszanie się w edytorze emacs	240
Wycinanie, kopiowanie i wklejanie w edytorze emacs	242
Wyszukiwanie i zastępowanie tekstu w edytorze emacs	242
Praca z wieloma oknami i buforami w edytorze emacs	242
Dostosowywanie edytora emacs	243
Zdobywanie dodatkowych informacji o edytorze emacs	246
Używanie programu gedit	246
Inne edytory tekstowe w Ubuntu	247
Podsumowanie	248
Rozdział 8. Obsługa poczty elektronicznej	
za pomocą programu Evolution	251
Uruchamianie programu Evolution	252
Kreator ustawień Evolution	253
Wysyłanie i odbieranie poczty elektronicznej	263
Wysyłanie wiadomości w programie Evolution	266
Konfigurowanie ustawień dla wysyłanych i odbieranych wiadomości	267
Odzyskiwanie usuniętych wiadomości	270
Tworzenie i korzystanie z folderów	270
Korzystanie z folderów wyszukiwania	272
Filtrowanie wiadomości przychodzących	274
Stosowanie filtrów dla poczty przychodzącej	275
Automatyczne sprawdzanie niechcianej poczty	278
Dodatkowe źródła informacji o Evolution	279
Podsumowanie	280
Rozdział 9. Surfowanie z Firefoksem	281
Krótką historia Firefoksa	281
Uruchamianie programu Firefox	283
Interfejs programu Firefox	283
Standardowe elementy okna Firefoksa	283
Używanie myszy w programie Firefox	285
Specjalne i niespecjalne funkcje programu Firefox	286
Konfigurowanie Firefoksa	289
Ustawianie strony domowej	289
Wyskakujące okienka, JavaScript i inne	290
Konfigurowanie ustawień z zakładki Prywatność	292
Zakładki	294
Tworzenie zakładek w programie Firefox	295
Zarządzanie zakładkami	296
Tworzenie dynamicznych zakładek	297

Ulepszanie Firefoksa	299
Instalowanie rozszerzeń do programu Firefox	299
Dodawanie nowych motywów do programu Firefox	302
Podsumowanie	303
Rozdział 10. Tworzenie i publikowanie dokumentów	305
Języki znaczników w Ubuntu	306
Instalowanie systemu TeX w Ubuntu	307
Korzystanie z programu TeX	307
Wprowadzenie do znaczników LaTeX	309
Formatowanie i wyświetlanie plików DVI	311
Korzystanie z narzędzi graficznych do pracy z systemami TeX i LaTeX	311
Dodatkowe źródła informacji o systemach TeX i LaTeX	314
Edycja tekstu w programie OpenOffice.org Writer	314
Instalowanie plików dla programu OpenOffice.org Writer	315
Krótki przewodnik po programie Writer	317
Wpisywanie danych osobistych	320
Tworzenie dokumentów za pomocą kreatorów	320
Modyfikowanie stylów i wyglądu dokumentów	324
Tworzenie i używanie szablonów	326
Importowanie plików utworzonych w innych procesorach tekstu	329
Dodatkowe informacje o programie Writer	331
Komputerowy skład tekstu za pomocą programu Scribus	332
Instalowanie Scribusa w Ubuntu	332
Krótki przewodnik po programie Scribus	333
Dodatkowe informacje o programie Scribus	340
Inne procesory tekstu i pakiety biurowe dla Linuksa	340
Podsumowanie	341
Rozdział 11. Pozostałe składniki pakietu biurowego:	
arkusz kalkulacyjny Calc	
i program do tworzenia prezentacji	343
Wprowadzenie do arkuszy kalkulacyjnych — przewodnik	344
Korzystanie z Gnumeric	353
Rozpoczęcie pracy z programem Gnumeric	354
Krótki przewodnik użytkownika programu Gnumeric	354
Funkcje w programie Gnumeric	355
Określanie rodzaju danych wprowadzanych do programu Gnumeric	357
Importowanie istniejących arkuszy	359
Korzystanie z programu OpenOffice.org Calc	361
Instalowanie plików dla programu OpenOffice.org Calc	361
Uruchamianie programu Calc	363
Krótki podręcznik użytkownika programu Calc	363
Funkcje w programie Calc	364
Określanie rodzaju danych wprowadzanych do programu Calc	366
Importowanie istniejących arkuszy	368
Korzystanie z programu OpenOffice.org Impress	371
Instalowanie plików dla programu Impress	372
Uruchamianie programu Impress	373
Krótki przewodnik po programie Impress	375
Tworzenie prezentacji	377
Importowanie istniejących prezentacji	380
Podsumowanie	381

Rozdział 12. Praca z grafiką	383
Terminologia	384
Używanie programu GIMP	387
Uruchamianie programu GIMP	388
Krótki przewodnik po programie GIMP	389
Podstawowe operacje wykonywane w programie GIMP	391
Dodatkowe informacje o GIMP-ie	396
Używanie programu OpenOffice.org Draw	397
Instalowanie programu OpenOffice.org Draw	397
Uruchamianie programu Draw	398
Przewodnik po programie Draw	399
Dodatkowe informacje o programie OpenOffice.org Draw	401
Inkscape — program do edycji grafiki wektorowej	401
Instalowanie programu Inkscape	402
Uruchamianie programu Inkscape	402
Krótki przewodnik po programie Inkscape	403
Dodatkowe informacje o programie Inkscape	405
Podsumowanie	405
Rozdział 13. Multimedia	407
Przegląd terminologii audio i wideo	408
Konfigurowanie urządzeń dźwiękowych, poziomu dźwięku i dźwięków systemowych	414
Testowanie i dostosowywanie systemu dźwiękowego	414
Ustawianie poziomu głośności	415
Uzyskiwanie dodatkowych informacji o karcie dźwiękowej	418
Konfigurowanie napędów CD i DVD	420
Instalowanie GStreamer'a i wtyczek	422
Instalowanie w systemie kodeków Microsoft Windows	424
Instalowanie w systemie oprogramowania umożliwiającego odtwarzanie płyt DVD	425
Używanie płyt CD	425
Odtwarzanie płyt CD za pomocą programu Sound Juicer	426
Zgrywanie płyt CD za pomocą programu Sound Juicer	427
Nagrywanie płyt z użyciem programu Serpentine	429
Inne źródła dźwięku	430
Pliki dźwiękowe i internetowe radio w Rhythmboksie	431
Konwertowanie plików audio	436
Płyty DVD	440
Odtwarzanie płyt DVD	440
Zgrywanie płyt DVD	442
Używanie programu K3b do kopiowania i nagrywania płyt DVD	442
Podsumowanie	448
Rozdział 14. Zagrasz?	449
Domyślne gry dla GNOME	450
Gry karciane	450
Gry planszowe	453
Gry wideo	457
Inne popularne gry znajdujące się w repozytoriach Ubuntu	462
Miliony wersji szachów	462
Go	465
Klony gry Monopol	467
Klony gry Scrabble	470
Podsumowanie	471

Rozdział 15. Łączenie się z innymi systemami	473
Ustanawianie bezpiecznego połączenia z innymi systemami	473
Logowanie się na zdalnych systemach za pomocą ssh	474
Uruchamianie demona SSH w systemie	476
Łączenie się z innymi systemami za pomocą VNC	477
Pobieranie programu VNC w wersji serwerowej oraz klienckiej dla odpowiedniej platformy	478
Korzystanie z programu vncviewer	481
Łączenie ze zdalnym serwerem usług terminalowych Windows	482
Korzystanie z programu rdesktop	483
Korzystanie z programu tsclient	485
Korzystanie z innych narzędzi do zdalnego łączenia się z komputerem	488
Podsumowanie	489
Rozdział 16. Przesyłanie i udostępnianie plików w Ubuntu	491
Przesyłanie plików do innych systemów	492
Korzystanie z FTP	493
Dostęp do zasobów systemów Windows	499
Dostęp do katalogów NFS z systemów Linux	502
Udostępnianie plików w sieciach P2P	503
Instalowanie i korzystanie z programu gtk-gnutella	505
Korzystanie z BitTorrenta w Ubuntu	509
Podsumowanie	512
Rozdział 17. Urządzenia elektroniczne w Ubuntu	513
Konfigurowanie systemu do obsługi urządzeń elektronicznych	514
Konfigurowanie obsługi aparatów cyfrowych i cyfrowych kamer wideo	514
Konfigurowanie obsługi iPodów i innych odtwarzaczy muzycznych	515
Konfigurowanie obsługi palmtopów i smartfonów	516
Konfigurowanie obsługi kart pamięci i innych urządzeń elektronicznych	517
Aparaty cyfrowe i Ubuntu	517
Palmtopy, smartfony i Ubuntu	520
Konfigurowanie i synchronizowanie urządzeń Palm OS	521
Konfigurowanie i synchronizowanie urządzeń z systemem Microsoft Windows	525
iPody, inne odtwarzacze audio i Ubuntu	530
Korzystanie z iPod'a w Ubuntu	531
Korzystanie z innych cyfrowych odtwarzaczy audio	541
Obsługa kart CompactFlash i SD	542
Podsumowanie	544
Rozdział 18. Tworzenie oprogramowania w Ubuntu	545
Instalowanie i używanie narzędzi programistycznych w Ubuntu	546
Instalowanie GCC, make i innych narzędzi	546
Kompilator GCC — podstawowe informacje	548
Używanie kompilatora GCC C	548
Tworzenie prostych plików makefile	550
Korzystanie ze zintegrowanych środowisk programistycznych	551
Popularne IDE dla Linuksa	552
Instalowanie środowiska Eclipse	554
Używanie Eclipse do tworzenia aplikacji w języku C	555
Dodatkowe informacje o Eclipse	570

System kontroli wersji	570
CVS — Concurrent Versioning System	571
Instalowanie i korzystanie z Subversion	571
Podsumowanie	580

Część III Ubuntu dla administratorów systemu 581

Rozdział 19. Proces rozruchu i zamykania systemu Ubuntu 583

Proces rozruchowy dystrybucji Ubuntu Linux	584
Boot Monitor lub BIOS	584
Program rozruchowy	584
Ładowanie jądra	587
Ładowanie i wykorzystanie inicjującego dysku RAM lub systemu plików RAM	588
Proces init	589
Analiza procesu startowego z użyciem programu Boot Chart	592
Optymalizacja procesu rozruchowego dystrybucji Ubuntu	594
Proces zamykania systemu Ubuntu Linux	598
Podsumowanie	599

Rozdział 20. Dodawanie, usuwanie i aktualizacja oprogramowania 601

Przegląd oprogramowania do zarządzania pakietami w Ubuntu	603
Repozytoria i ich komponenty w systemie Ubuntu	604
Uaktywnianie poszczególnych elementów repozytorium	606
Uaktywnianie dodatkowych repozytoriów z użyciem edytora tekstu	608
Uaktywnianie dodatkowych repozytoriów z wykorzystaniem narzędzia Ustawienia oprogramowania	609
Problemy z dodawaniem i dostępem do niestandardowych repozytoriów	612
Mieszanie repozytoriów Ubuntu i Debiana	613
Eksploracja systemu z użyciem programu dpkg i pokrewnych	616
Sporządzanie listy pakietów zainstalowanych w systemie	616
Sporządzanie listy pakietów dostępnych do instalacji	618
Odczyt informacji o pakiecie	621
Odczyt informacji o zawartości pakietu	621
Określanie pakietu, do którego należy istniejący plik	622
Określenie pakietu, w którym znajduje się brakujący plik	623
Usuwanie oprogramowania z użyciem programu apt-get	624
Aktualizacja systemu z użyciem programu apt-get	625
Inteligentna aktualizacja systemu z użyciem programu apt-get	625
Pobieranie kodów źródłowych pakietów z użyciem programu apt-get	626
Obsługa zależności kompilacyjnych przy użyciu programu apt-get	627
Instalacja i usuwanie oprogramowania z użyciem programu aptitude	628
Wskazówki dotyczące wykorzystania interfejsu użytkownika programu aptitude	629
Instalacja zalecanych pakietów z użyciem aptitude	630
Zalety użycia programu aptitude do instalacji i usuwania oprogramowania	632
Dodawanie i usuwanie pakietów z użyciem programu Synaptic	633
Konfiguracja ustawień programu Synaptic	635
Wyszukiwanie oprogramowania w programie Synaptic	636
Instalacja pakietów za pomocą programu Synaptic	638
Usuwanie pakietów za pomocą programu Synaptic	641
Wykorzystanie menedżera aktualizacji	643

Konwersja pakietów z innych formatów	644
Utrzymanie porządku w systemie	647
Podsumowanie	647
Rozdział 21. Zarządzanie użytkownikami, grupami i zaawansowane uprawnienia	649
Tworzenie i zarządzanie grupami i użytkownikami	650
Tworzenie nowych użytkowników	651
Zarządzanie istniejącymi użytkownikami	654
Tworzenie nowych grup	656
Zarządzanie istniejącymi grupami	658
Podsystem PAM i proces uwierzytelniania użytkowników w Linuksie	659
Pliki konfiguracji mechanizmów PAM dla aplikacji i usług	661
Przykład: konfiguracja PAM dla procesu logowania w systemie	662
Konfiguracja plików różnych modułów PAM	666
Co zrobić, gdy brakuje plików konfiguracyjnych?	666
Konfiguracja polecenia sudo w systemie Ubuntu	667
Wykorzystanie ACL-i do zaawansowanego zarządzania dostępem do plików	670
Przegląd ACL-i w Linuksie	671
Instalacja w Ubuntu poleceń do obsługi ACL-i	672
Uaktywnienie obsługi ACL-i w systemie plików	673
Praca z ACL-ami z wiersza poleceń	674
Praca z ACL-ami przy użyciu graficznego narzędzia	676
Podsumowanie	679
Rozdział 22. Wykonywanie i przywracanie kopii zapasowych	681
Podstawowe zagadnienia dotyczące kopii zapasowych	682
Po co wykonywać kopie zapasowe?	682
Różne typy kopii zapasowych	684
Weryfikacja i testowanie kopii zapasowych	688
Decyzja o tym, co kopiować	688
Oprogramowanie dla Linuksa do wykonywania kopii zapasowych	689
Oprogramowanie dla Linuksa do lokalnego wykonywania kopii zapasowych i ich odtwarzania	690
Oprogramowanie dla systemu Linux do wykonywania sieciowych kopii zapasowych	693
Wykonywanie kopii zapasowych na lokalnych nośnikach wymiennych	695
Archiwizacja i przywracanie danych z użyciem programu tar	696
Wykonywanie aktualnych kopii katalogu lokalnego z użyciem polecenia cp	697
Wykonywanie aktualnych kopii katalogu lokalnego z użyciem polecenia rsync	698
Instalacja i wykorzystanie programu BackupPC	700
Instalacja pakietu backuppc	702
Konfiguracja programu BackupPC	703
Identyfikacja hostów do wykonywania kopii zapasowych	704
Definiowanie kopii zapasowej z użyciem protokołu rsyncd	705
Definiowanie kopii zapasowych z użyciem protokołu SMB	708
Uruchamianie kopii zapasowych w programie BackupPC	709
Odtwarzanie danych z kopii zapasowej w programie BackupPC	712
Podsumowanie	715

Rozdział 23. Dodawanie sprzętu i przyłączanie urządzeń peryferyjnych ... 717

Dodawanie drukarek	718
Wykorzystanie skanera	722
Wykorzystanie zewnętrznych dysków i napędów CD/DVD	727
Konfiguracja automatycznego wykrywania urządzeń	727
Rozwiązywanie problemów z automatycznym wykrywaniem urządzeń	728
Dodawanie wewnętrznych dysków i napędów CD/DVD	729
Dodawanie do systemu napędów EIDE/ATA	730
Dodawanie napędów SATA	731
Dodawanie napędów SCSI	732
Rozwiązywanie problemów z uruchamianiem systemu po dodaniu nowych urządzeń	733
Lokalizowanie, partycjonowanie i formatowanie dysków twardych	736
Wykorzystanie kart PCMCIA	744
Przyłączanie kart PCI	745
Rozwiązywanie problemów z urządzeniami za pomocą narzędzi graficznych	746
Podsumowanie	748

Rozdział 24. Konfiguracja sieci i bezpieczeństwo 749

Podstawowe zagadnienia dotyczące sieci	751
Ręczna konfiguracja sprzętu sieciowego	754
Ręczna konfiguracja połączeń modemowych	757
Definiowanie i wykorzystanie wielu konfiguracji sieciowych	760
Testowanie sieci z użyciem narzędzi sieciowych dla środowiska GNOME	761
Wskazówki dotyczące zabezpieczenia systemu	763
Instalacja zapory sieciowej	765
Podstawy zapór sieciowych i filtrowania pakietów w Linuksie	766
Instalacja i konfiguracja zapory sieciowej z użyciem programu Lokkit	767
Podsumowanie	772

Rozdział 25. Sieć bez kabli 773

Ogólne informacje o sieciach bezprzewodowych	774
Konfiguracja interfejsów sieci bezprzewodowej	777
Narzędzia do konfiguracji sieci bezprzewodowych obsługiwane z wiersza poleceń	780
Instalacja i wykorzystanie sterowników dla systemu Windows	781
Instalacja pakietu NDIS Wrapper i narzędzi pomocniczych	783
Instalacja sterowników systemu Microsoft Windows	784
Sztuczki i kruczki związane z mechanizmem NDIS Wrapper	786
Podsumowanie	787

Część IV Konfiguracja serwerów w Ubuntu 789**Rozdział 26. Konfiguracja serwera WWW 791**

Podstawy sieci WWW	792
URL-e a URI	793
Wprowadzenie do serwerów WWW i Apache	794
Instalacja serwera Apache	795
Instalacja serwera Apache z wiersza poleceń	796
Instalacja Apache za pomocą programu Synaptic	797
Pliki serwera Apache 2	798
Konfiguracja serwera Apache	799

Usuwanie problemów	803
Więcej informacji	804
Podsumowanie	805
Rozdział 27. Konfiguracja serwera poczty	807
Serwerowe technologie poczty elektronicznej	808
Popularne linuxowe aplikacje MTA	809
Po co instalować własny serwer poczty?	811
Instalacja Postfixa i pakietów pomocniczych	812
Wbudowane modele konfiguracji Postfixa	813
Instalacja Postfixa z wiersza poleceń	814
Instalacja Postfixa i pakietów pomocniczych z użyciem menedżera pakietów Synaptic	817
Konfiguracja serwera Postfix	819
Pliki konfiguracyjne Postfixa	819
Identyfikacja zaufanych hostów i domen	822
Przepisywanie adresów w poczcie wychodzącej	822
Przyjmowanie poczty dla całej domeny	823
Uaktywnienie obsługi POP/POP3 za pomocą pakietu Qpopper	823
Konfiguracja mechanizmów antyspamowych i antywirusowych	825
Obsługa mechanizmu greylisting za pomocą pakietu postgrey	826
Parametry serwera Postfix służące do odrzucania niechcianej poczty	826
Integracja programów MailScanner, SpamAssassin i ClamAV z serwerem Postfix	829
Więcej informacji	831
Podsumowanie	832
Rozdział 28. Konfiguracja serwera DHCP	833
Ogólne informacje o DHCP	834
Instalacja serwera DHCP	836
Zarządzanie serwerem DHCP z wiersza poleceń	837
Tworzenie plików konfiguracyjnych serwera DHCP za pomocą edytora tekstu	838
Dodatkowe ustawienia w pliku konfiguracyjnym serwera DHCP	840
Zarządzanie serwerem DHCP za pomocą interfejsu graficznego	841
Rozwiązywanie problemów z DHCP	844
Podsumowanie	845
Rozdział 29. Konfiguracja serwera DNS	847
Podstawowe informacje o DNS i serwerze BIND	849
Instalacja serwera DNS za pomocą menedżera pakietów Synaptic	851
Przegląd plików konfiguracyjnych serwera BIND	852
Tworzenie definicji strefy DNS i pliku danych dla zapytań odwrotnych	854
Wykorzystanie wspólnych wpisów dla stref oraz plików zapytań odwrotnych: SOA i \$TTL	854
Tworzenie plików stref	857
Tworzenie pliku odwzorowań odwrotnych DNS	859
Instalacja w serwerze BIND pliku strefy i pliku odwzorowania odwrotnego	860
Ponowne uruchamianie i testowanie serwera DNS	861
Rozwiązywanie problemów z serwerami DNS	863
Więcej informacji na temat DNS i serwera BIND	866
Podsumowanie	866

Rozdział 30. Konfiguracja serwera wydruków	867
Historia drukowania w systemach Linux i Unix	868
Konfiguracja zdalnych hostów do wykorzystania serwera wydruków CUPS	869
Uaktywnianie opcji administracyjnych w interfejsie WWW podsystemu CUPS	870
Integracja podsystemu wydruków Windows z serwerem wydruków Ubuntu	871
Integracja podsystemu wydruków Mac OS X z serwerem wydruków Ubuntu	872
Usuwanie problemów z udostępnianymi drukarkami	874
Sprawdzenie dzienników podsystemu CUPS	875
Odblokowanie dostępu do wybranych funkcji interfejsu administracyjnego WWW podsystemu CUPS	876
Obsługa wstępnie sformatowanych wydruków	877
Gdzie można znaleźć dalsze informacje o podsystemie CUPS?	877
Podsumowanie	878
Rozdział 31. Konfiguracja serwera NFS	879
Podstawy technologii NFS	880
Jak działa NFS?	881
Porównanie różnych wersji NFS	883
Instalacja serwera NFS i pakietów pomocniczych	884
Wykorzystanie narzędzia Foldery współdzielone do udostępniania zasobów	886
Weryfikacja działania mechanizmu NFS	889
Ręczne definiowanie eksportów w pliku /etc/exports	891
Gdzie znaleźć więcej informacji na temat mechanizmu NFS?	892
Podsumowanie	893
Rozdział 32. Konfiguracja serwera Samba	895
Przegląd technologii udostępniania zasobów w sieci Microsoft Windows	896
Wprowadzenie do Samby	897
Instalacja serwera Samba i pakietów pomocniczych	898
Podstawy konfiguracji serwera Samba	899
Identyfikacja grupy roboczej lub domeny	900
Konfiguracja uwierzytelniania Samby	901
Udostępnianie drukarek i katalogów domowych z użyciem Samby	902
Weryfikacja pliku konfiguracyjnego Samby	903
Sprawdzanie dostępności i usług oferowanych przez Sambę	904
Wykorzystanie folderów udostępnionych	906
Dalsze informacje o Sambie	909
Podsumowanie	910
Dodatki	911
Zawartość dysku CD	913
skorowidz	919

O autorze

William von Hagen (Bill) jest administratorem systemów uniksowych od ponad dwudziestu lat, a miłośnikiem Linuksa od początku lat 90. Dla Linuksa pracował jako kierownik produktu, programista systemowy, administrator systemu, autor, projektant aplikacji, bębniarz i kierownik ds. zarządzania treścią. Bill jest autorem i współautorem książek poświęconych zagadnieniom, takim jak hacking serwerów linuksowych, systemy plików w Linuksie, SUSE Linux, Red Hat Linux, GCC, SGML, Mac OS X i hacking TiVo. Napisał też wiele artykułów o Linuksie, systemach wbudowanych, Mac OS X, Uniksie i otwartym oprogramowaniu. Kolekcjonuje komputery, a specjalizuje się w stacjach roboczych, posiada ich w swoich zbiorach ponad 200. Można się z nim skontaktować pod adresem vonhagen@vonhagen.org.

Podziękowania

Wszystko, co do tej pory napisałem, było możliwe dzięki miłości, wsparciu i nieskończonej cierpliwości Dorothy Fisher, mojej żony i najlepszego przyjaciela. Dorothy uwierzyła we mnie od pierwszego dnia, usunęła z moich tekstów kilka tysięcy przecinków i zaakceptowała moje wymówki — „Nie mogę tego zrobić teraz, wróć za piętnaście minut albo sześć godzin” — więcej razy, niż można by oczekiwać tego od kogokolwiek. Mam też ogromne szczęście mieć takich przyjaciół jak Jeff „Sunshine” Kaminski, dr Joe O’Lear, Jim Morgan i Kim Walter. Słyszałem kiedyś, że przyjaciel wyciągnie cię z więzienia, ale prawdziwy przyjaciel będzie siedział z tobą i jeszcze powie: „Stary, ale to była zabawa!”. Na szczęście, nie musiałem jeszcze testować słuszności tych słów, ale myślę, że wszystkim nam byłoby do twarzy w kolorze pomarańczowym.

Chciałbym też podziękować Carol Long, Tomowi Dinse’owi, Kit Kemper i innym osobom z wydawnictwa Wiley, które umożliwiły mi napisanie tej książki i pomagały podczas jej tworzenia. Wszystkie błędy są wyłącznie moją winą, ale bez waszego wkładu byłyby dużo słabsza. Mówiąc szczerze, ta książka nie powstałaby bez was.

I na koniec — oczywiście nie powstałaby bez ludzi, takich jak Linus Torvalds, Mark Shuttleworth, członkowie społeczności Ubuntu, projektu Debian, Richard Stallman, FSF i miliony osób czerpiących z rogu obfitości GNU/Linux.

Wprowadzenie

Dystrybucja Linuksa to właściwie suma wszystkich rzeczy, które są potrzebne do uruchomienia Linuksa na komputerze. Istnieje wiele różnych dystrybucji, każda adresowana do określonej grupy odbiorców, wyposażona w zestaw funkcji, narzędzi administracyjnych i fanclub, znany też pod nazwą społeczności danej dystrybucji. Większość członków społeczności, pomijając fanatyków, to osoby, które z tych czy innych powodów zaczęły używać danej dystrybucji. Przyczyny mogą być różne: od polecenia przez przyjaciela, poprzez chęć użycia płyty CD/DVD dołączonej do czasopisma, aż do książki, popularyzującej daną dystrybucję.

Ubuntu to najbardziej ekscytująca dystrybucja Linuksa z tych, które pojawiły się w ciągu kilku ostatnich lat. Przewrotnie to, co w Ubuntu można uznać za nowe, w Linuksie dostępne jest od wielu lat, a dzieje się tak dlatego, że korzenie tej dystrybucji sięgają jednej z najstarszych i najpopularniejszych dystrybucji, znanej jako Debian GNU/Linux. Osoby, które powołały i wspierają Ubuntu oraz firmę Canonical, w prostej linii pochodzą od fanów Debiana, chcących zaoferować bardziej dynamiczną i aktualniejszą dystrybucję. Dlatego w duchu Linuksa i otwartego oprogramowania utworzyli według własnego przepisu Ubuntu, czyli dystrybucję, która ma najlepsze cechy Debiana oraz innych dystrybucji i programów.

Ubuntu oznacza „humanitaryzm dla innych”. Dla twórców tej dystrybucji nie jest to tylko przyjemnie brzmiący i chwytliwy slogan. Wyjątkowość Ubuntu polega na udanym połączeniu aspektów biznesowych i socjalnych dla wszystkich użytkowników. Ubuntu jest wydawane regularnie co sześć miesięcy, a wsparcie i aktualizacje dla danego wydania są dostępne co najmniej przez osiemnaście miesięcy. Więcej informacji na ten temat zamieszczam w 1. rozdziale, gdzie można też znaleźć wykładnię filozofii, opis społeczności oraz wyjaśnienie, dlaczego suma wszystkich składników sprawia, że Ubuntu jest dystrybucją wyjątkową.

Mówiąc w skrócie: Ubuntu to dystrybucja dla ludzi. W trakcie lektury niniejszej książki czytelnik przekona się, że jest wiele doskonałych technicznych powodów, aby zacząć używać Ubuntu nawet wtedy, jeżeli użytkownik jest bardzo wymagający w tym zakresie. Ale nie o to chodzi w Ubuntu — to dystrybucja dla ludzi, którzy chcą korzystać ze swojego komputera i potrzebują solidnego oprogramowania, niezależnie od tego, czy piszą kody programów, czy zwiedzają internet, wysyłają i odbierają pocztę elektroniczną, pracują ze zdjęciami cyfrowymi, oglądają DVD, słuchają muzyki itd. Ubuntu oferuje oprogramowanie, za którego pomocą można zrobić dokładnie to, na co ma się ochotę.

Podobnie jak inne dystrybucje Linuksa, również Ubuntu można bezpłatnie pobrać z sieci. Ale jest jeszcze lepsze rozwiązanie. Książka zawiera płytę CD z wydaniem Ubuntu o długoterminowym wsparciu (Ubuntu 6.06 LTS), choć w sieci dostępne są już nowsze wersje. Jeżeli użytkownik nie dysponuje nagrywarką CD, potrzebuje wersji dla architektur innych niż x86 lub po prostu nie ma czasu, ludzie z Canonical prześlą odpowiednią wersję, którą będzie można przetestować na komputerze lub od razu zainstalować. To więcej niż darmowe pobieranie — to rewolucja! Bezpłatne kopie dystrybucji do pobrania z sieci to żadna nowość, ale wysyłanie użytkownikom nagranych płyt, jeśli tylko tego potrzebują, to coś więcej niż tylko kolejna dystrybucja Linuksa. Ludzie Ubuntu są oddani swojej misji, a my wszyscy jesteśmy szczęśliwymi zwycięzcami.

Dla kogo przeznaczona jest ta książka?

Jeżeli czytelnik przegląda tę książkę w księgarni i nie jest pewien, z którą dystrybucją Linuksa rozpocząć przygodę lub czy w ogóle pracować z tym programem, ta książka jest właśnie dla niego. Ubuntu to kompletna, przyjazna wizualnie i ukierunkowana na społeczność dystrybucja, która bardzo ułatwia rozpoczęcie współpracy z Linuksem. Ubuntu została zaprojektowana jako dystrybucja dla przeciętnych użytkowników, ale — jak będzie się można przekonać w trakcie lektury niniejszej książki — dzięki niesamowitej mocy Linuksa może służyć zarówno jako oprogramowanie pojedynczego komputera, jak i serwera działającego w skomplikowanych sieciach firmowych. Ubuntu oferowane jest z ogromnym wyborem aktualnego i różnorodnego oprogramowania, a kolejne aplikacje dostępne są za pomocą narzędzi administracyjnych. Jest też często uaktualniane i nie ma dla niego wirusów czy „przypadkowo zainstalowanych programów”. Linux jest bezpieczny.

Kolejnym bogactwem jest ogromna społeczność Ubuntu. Są miejsca do zadawania pytań i uzyskiwania odpowiedzi. A ludzie chcą sobie wzajemnie pomagać. Fora Ubuntu oraz listy dyskusyjne to żywe dowody słuszności filozoficznych i społecznych założeń Ubuntu (więcej informacji na ten temat zamieszczam w rozdziale 1.). Mówiąc w skrócie: każda dystrybucja Linuksa ma swoich fanów, zwolenników, fora i listy dyskusyjne, ale właśnie te z Ubuntu są najciekawszymi i najszyteczniejszymi z tych, które odwiedzałem podczas swojej kilkunastoletniej przygody z Linuksem. Wręcz można wyczuć ekscytację. Ubuntu spełnia obietnicę internacjonalizacji — istnieją wersje w ponad stu językach (wraz z odpowiednim układem klawiatury), a kolejne są opracowywane.

Niniejsza książka może być również przydatna czytelnikom już korzystających z Ubuntu, ponieważ objaśniam w niej wiele aspektów korzystania ze standardowych aplikacji obecnych w tej dystrybucji, konfigurowania systemu, wykonywania czynności administracyjnych itd. Jeżeli czytelnicy są już zaawansowanymi użytkownikami desktopowych wersji Ubuntu, książka pozwoli im wykorzystać potencjał tej dystrybucji w roli serwera plików, pocztowego, sieciowego itp.

Jak zorganizowana jest ta książka?

Książka jest podzielona na cztery główne części.

Część I — Rozpoczęcie pracy z Ubuntu

W pierwszej części niniejszej książki przedstawiam podstawowe informacje dotyczące Ubuntu, włącznie z instrukcjami instalowania systemu na komputerze czy też po prostu testowania go za pomocą płyty live CD. W rozdziale 1. podaję podstawowe dane o dystrybucji, omawiam jej założenia filozoficzne, skupioną wokół niej społeczność oraz strony internetowe, na których można znaleźć informacje i pomoc w zakresie konfigurowania systemu (przydatne, kiedy pomoc potrzebna jest natychmiast i nie ma czasu na czytanie całej książki). W rozdziale 1. wyjaśniam też, dlaczego właśnie Ubuntu jest właściwą dystrybucją i porównuję jej możliwości z kilkoma innymi popularnymi dystrybucjami. W rozdziale 2. opisuję, jak zainstalować Ubuntu jako jedyny system operacyjny, jako jeden z dwóch na komputerze, na którym działa już inny system, a użytkownik nie może z niego zrezygnować. W rozdziale 3. podaję, jak wypróbować tę dystrybucję w sytuacji, kiedy użytkownik nie posiada żadnego systemu komputerowego czy wolnego miejsca na dysku. Jest to możliwe dzięki płytom live CD z Ubuntu, które umożliwiają uruchomienie dystrybucji na komputerze bez konieczności dokonywania jakichkolwiek zmian w systemie.

Część II — Korzystanie z Ubuntu

W części drugiej omawiam obszerny zestaw programów, które są dostępne dla Ubuntu; opisuję, jak ich używać, aby wykonać czynności, które zwykle wykonują dzisiejsi użytkownicy komputerów, czyli odbieranie i wysyłanie poczty elektronicznej, surfowanie w internecie, tworzenie dokumentów i arkuszy kalkulacyjnych, granie w gry komputerowe, odtwarzanie kolekcji muzycznej i zarządzanie nią, odtwarzanie płyt DVD i muzyki z internetu. W rozdziałach 4., 5. i 6. omawiam podstawy Linuksa i Uniksa, czytelnicy zainteresowani linią poleceń, interfejsem graficznym systemu i ogólnie organizacją Ubuntu powinni tam zajrzeć. Rozdziały od 7. do 14. poświęcam opisowi powszechnych czynności wykonywanych za pomocą komputerów, takich jak edytowanie plików, czytanie i wysyłanie poczty elektronicznej, surfowanie w internecie, drukowanie dokumentów, korzystanie z multimediów, arkuszy kalkulacyjnych, prezentacji itd. W rozdziałach 15. i 16. piszę, na czym polega udostępnianie i przesyłanie plików, zarówno w sieci lokalnej, jak i przez internet. W rozdziale 17. umieszczam opis podłączania do Ubuntu palmtopów i synchronizowania danych. Ta część kończy się rozdziałem 18., w którym omawiam tworzenie programów w Ubuntu, ze szczególnym uwzględnieniem standardowych narzędzi GNU/Linux, takich jak kompilatory, narzędzia do automatycznego tworzenia oprogramowania, graficzne zintegrowane środowiska programistyczne i systemy kontroli wersji kodu, które — oczywiście — wszystkie są bezpłatne.

Część III — Ubuntu dla administratorów systemu

Każdy, kto konfigurował swój komputer, tak aby miał dostęp do sieci, umożliwiał logowanie się innym użytkownikom itp., ma — przynajmniej ja tak uważam — kwalifikacje młodego specjalisty ds. administrowania systemem. W tej części opowiadam, jak wykonać

wszystkie te czynności administracyjne, które mogą być interesujące dla użytkowników domowych, ale poruszam też zagadnienia, którymi mogą być zainteresowani administratorzy zarządzający sieciami firmowymi. W części trzeciej wyjaśniam, co dzieje się podczas uruchamiania systemu, tam też zamieszczam wprowadzenie do zagadnień ochrony plików i katalogów w Linuksie oraz do wykonywania kopii zapasowych. W rozdziale 19. szczegółowo omawiam proces uruchamiania systemu, którego poznanie pozwoli użytkownikom dodawać nowe usługi i łatwiej diagnozować potencjalne problemy. W rozdziale 20. opisuję metody aktualizowania systemu oraz miejsca, gdzie należy szukać nowego oprogramowania i jak je instalować. W rozdziale 21. przybliżam standardowe (dla grup i użytkowników) i zaawansowane metody zabezpieczania systemu, zabezpieczanie plików przed nieautoryzowanym dostępem i ich współdzielenie. W rozdziale 22. omawiam metody wykonywania kopii zapasowych, dzięki czemu można zabezpieczyć swoje najważniejsze pliki i w razie awarii łatwo i szybko je przywrócić. W rozdziale 23. opisuję instalowanie nowego sprzętu w komputerach z Ubuntu: jeżeli brakuje miejsca na dysku, potrzebna jest lepsza karta graficzna itd., warto tu zajrzeć. W rozdziale 24. poruszam zagadnienia bezpieczeństwa w sieci — co zrobić, aby zabezpieczyć się przed włamywaczami oraz innymi wandalami sieciowymi. Ta część książki kończy się rozdziałem 25., w którym przedstawiam zagadnienia dotyczące korzystania z Ubuntu na komputerach przenośnych i używanie bezprzewodowych klawiatur, myszy komputerowych, a nawet słuchawek.

Część IV — Konfiguracja serwerów w Ubuntu

W części czwartej opisuję konfigurowanie serwerów dostarczających usług sieciowych i internetowych. W osobnych rozdziałach opowiadam, jak skonfigurować serwer sieciowy, pocztowy, scentralizowany serwer wydruku, serwer udostępniający pliki i usługi drukowania dla systemów Microsoft Windows działających w tej samej sieci, a także serwery DNS, DHCP i NFS.

Konwencje typograficzne zastosowane w książce

Wszystkie elementy typograficzne w tej książce zostały zaprojektowane tak, aby jak najbardziej ułatwić czytelnikom zdobywanie informacji.

Za każdym razem, kiedy chcę zwrócić uwagę czytelników na ważne aspekty danego zagadnienia, wówczas pojawiają się *Uwaga*, *Wskazówka* i *Ostrzeżenie*.



Akapit zawierający dodatkowe i pomocnicze informacje, które mogą się przydać, ale są tylko uzupełnieniem prezentowanego zagadnienia.



Informacje, które mają ułatwić użytkownikom pracę; zwykle są to metody działania pozwalające osiągnąć dany cel szybciej i łatwiej.



Zamieszczone w tym akapicie informacje są ważne, dlatego je wyodrębniono i oznaczono specjalną ikoną. Znajdują się tu ważne informacje, na których należy się skupić, ze względu na potencjalne niedogodności lub zagrożenia danych czy systemu. Użytkownik powinien zwrócić uwagę, że wykonanie opisanych procedur niesie poważne konsekwencje dla systemu, danych, sprzętu czy oprogramowania.

Dołączona płyta CD

Większość książek zawiera we wstępie część „Dołączona płyta CD”, a my (ja i wydawnictwo) nie chcielibyśmy zawieść czytelnika. Płyta dołączona do książki zawiera wersję Ubuntu 6.06 LTS (z długoterminowym wsparciem). Pozwoli ona użytkownikom wypróbować Ubuntu bez konieczności dokonywania jakichkolwiek zmian w systemie lub też skorzystać z graficznego programu instalacyjnego i umieścić tę dystrybucję na stałe na dysku komputera klasy Pentium. Testowanie i instalowanie Ubuntu dokładniej omawiam w rozdziale 2. Dostępne są również płyty z Ubuntu dla innych architektur (PowerPC, 64 bitowych i SPARC), a także dla specjalnych zastosowań (takich jak instalowanie wersji serwerowej) i w różnych trybach *alternate* czy OEM (ang. *Original Equipment Manufacturer*). Informacje na ten temat można znaleźć w rozdziałach 2. i 3.

Zgodnie z informacjami zamieszczonymi w rozdziale 1., nowe wydanie Ubuntu jest publikowane co 6 miesięcy, dlatego nowsze wydania od opisanego tutaj można pobrać ze stron Ubuntu lub też zamówić w ramach projektu ShipIt na stronie <https://shipit.ubuntu.com>. Wystarczy tylko utworzyć konto, podać swój adres i listę życzeń. W ten sposób do czytelników dotrą najnowsze wydania, a książka — nadal użyteczna — będzie spoczywać na półce.

Jeżeli czytelnik posiada nagrywarkę CD i nie chce czekać, zawsze może pobrać plik obrazu najnowszego wydania Ubuntu Desktop, Server lub Alternate ze strony <http://www.ubuntu.com/getubuntu/download>. Jeżeli użytkownik chciałby wypróbować Ubuntu ze środowiskiem KDE, wówczas powinien pobrać Kubuntu (www.kubuntu.org). Jeśli z kolei użytkownik chciałby zobaczyć, jak działa środowisko graficzne o mniejszych wymaganiach, może skorzystać Xfce i Xubuntu (www.xubuntu.org). Tę wersję projektu Ubuntu omawiam w rozdziale 1.

Część I

Rozpoczęcie pracy z Ubuntu

W tej części:

Rozdział 1. Projekt Ubuntu

Rozdział 2. Instalowanie Ubuntu

Rozdział 3. Instalowanie Ubuntu jako systemu specjalnego zastosowania

Rozdział 1.

Projekt Ubuntu

W tym rozdziale:

- ♦ Wprowadzenie do Ubuntu
- ♦ Powody wyboru Ubuntu
- ♦ Wymagania instalacyjne
- ♦ Płyty instalacyjne Ubuntu
- ♦ Wsparcie dla Ubuntu
- ♦ Uzyskiwanie dodatkowych informacji o Ubuntu

Komputery osobiste i systemy operacyjne przebyły długą drogę od lat 70., kiedy to po raz pierwszy pojawiły się na rynku. W tamtych czasach można było przełączać programy tylko za pomocą przycisków znajdujących się na obudowie, a same urządzenia mogły uruchamiać tylko jeden program, przynajmniej dopóty, dopóki użytkownik nie umieścił w komputerze innego, co z kolei wiązało się z usunięciem pierwszego. Dzisiejsze komputery oferują zaawansowaną grafikę i rozbudowane interfejsy, dzięki czemu można łatwo uruchamiać wiele programów jednocześnie.

Pierwsi użytkownicy komputerów domowych byli społecznością osób, które chciały coś zrobić ze swoimi maszynami. Tworzyli kluby i publikowali biuletyny, aby dzielić się swoimi zainteresowaniami i wiedzą, a często też programami, które pisali i użytkowali na swoich komputerach. Wiele firm, dostrzegając możliwości i rozwijający się rynek, zaczęło pisać i sprzedawać programy, takie jak edytory tekstu, arkusze kalkulacyjne, gry czy systemy operacyjne ułatwiające zarządzanie różnymi programami, ładowanie ich i uruchamianie.

Chociaż moc i możliwości dzisiejszych komputerów wyprzedziły o całe lata świetlne pierwsze maszyny, to jednak idea pisania oprogramowania i bezpłatnego udostępniania innym nie przeminęła. Chociaż nigdy nie było o niej specjalnie głośno (ponieważ nikt nie zarabiał na tym pieniędzy), wolne oprogramowanie (a często i jego kod źródłowy) wciąż jest dostępne za pośrednictwem społeczności użytkowników komputerów, biuletynów informacyjnych czy sieci komputerowych, takich jak internet. Ruch wolnego oprogramowania rozpowszechnił się dzięki trzem wydarzeniom:

- ♦ powołaniu przez Richarda Stalmanna w roku 1983 GNU (www.gnu.org), czyli projektu tworzenia oprogramowania, którego kod źródłowy zawsze będzie dostępny;
- ♦ ogłoszeniu powstania fundacji wolnego oprogramowania (*Free Software Foundation*, www.fsf.org), mającej wspomagać projekt GNU;
- ♦ wprowadzeniu w roku 1991 przez fińskiego studenta informatyki Linusa Torvaldsa wolnego systemu operacyjnego znanego jako Linux.

Niniejsza książka nie mogłaby powstać bez tych trzech zdarzeń, były one zaskakujące w swojej niezależności i mocy, spowodowały też (choć nie tylko one) kontynuację rozwoju ducha wspólnoty w środowisku. Czasem, przy odrobinie szczęścia, im więcej rzeczy się zmienia, tym więcej pozostaje takich samych.

System operacyjny i programy omówione w niniejszej książce są bezpłatne, a ich kod źródłowy wolno dostępny. Każdy, kto tego zapagnie, może je kompilować, instalować i uruchamiać. Wokół nich rozwinęła się ogromna społeczność użytkowników łącznie z wyspecjalizowanymi grupami zajmującymi się tworzeniem łatwych w instalacji zestawów oprogramowania, zwanych dystrybucjami Linuksa. W niniejszym rozdziale omówiono filozofię, społeczność oraz historię jednej z najnowszych i chyba — dzięki swojej dostępności, łatwości instalacji i użytkowania wolnego oprogramowania — najlepszej dystrybucji Ubuntu.

Wprowadzenie

Pojawienie się Linuksa, wolno dostępnego systemu operacyjnego, to punkt zwrotny w dziejach współczesnych komputerów osobistych. Linux jest obecnie prawdopodobnie najpopularniejszym środowiskiem serwerowym i szybko zdobywa coraz większą liczbę użytkowników systemów przeznaczonych do użytku domowego. Następne dwa podrozdziały są wprowadzeniem dla tych, którzy właśnie postanowili stosować ten system lub nie są zaznajomieni z pewnymi podstawowymi założeniami i pojęciami, takimi jak np. dystrybucja. Czytelnicy swobodnie poruszający się w tych zagadnieniach, którzy postanowili rozpocząć korzystanie z Ubuntu, mogą pominąć poniższe punkty i przejść wprost do „Wprowadzenia do Ubuntu”.

Dlaczego warto korzystać z Linuksa?

Można założyć, że czytelnik kupił niniejszą książkę, ponieważ chce się czegoś dowiedzieć o Linuksie. Jeżeli jednak chciałby poznać powody, dla których warto zacząć z niego korzystać, oto kilka z nich.

- ♦ **Potężny i nowoczesny projekt.** Linux został zaprojektowany od podstaw tak, aby umożliwić jednoczesne działanie wielu programów oraz dostarczyć usługi, z których będzie mogło korzystać wielu użytkowników. Większość komputerowych systemów operacyjnych, takich jak np. Microsoft Windows, na początku swego rozwoju mogła uruchamiać tylko jeden program na raz, dlatego od zawsze próbowały nadrobić zaległości.

- ♦ **Wolno dostępny kod źródłowy oznacza brak blokady producenta¹.** Bez względu na używany system operacyjny, zawsze mogą pojawić się niezliczone problemy dotyczące aplikacji czy ogólnego działania. Podczas korzystania z gotowego, dostępnego w handlu detalicznym systemu operacyjnego, takich firm jak Microsoft czy Apple, poprawki i uaktualnienia można otrzymać tylko od producenta. Linux jest systemem otwartym, zatem kiedy użytkownikowi nie będzie odpowiadał sposób działania Red Hata, jego koszt czy typ oferowanego wsparcia technicznego, zawsze może przejść na SUSE Novella, Mandrivę czy — co preferuję sam — Ubuntu.
- ♦ **Tysiące wolnych, potężnych aplikacji.** Potrzebny jest edytor tekstu? Wystarczy pobrać i zainstalować Writera z pakietu OpenOffice.org, AbiWorda, Kwrite'a czy dziesiątki innych. Potrzebna jest baza danych? Wystarczy pobrać i zainstalować MySQL, PostgreSQL czy wiele innych. Potrzebny jest program do tworzenia grafiki i obróbki zdjęć cyfrowych? Nie ma lepszego od GIMP-a (czyli *GNU Image Manipulation Program*). Jeżeli jest jakiś problem, to tylko ze zbyt wieloma możliwościami wyboru, ale żadna z tych opcji nic nie kosztuje.
- ♦ **Wsparcie w standardzie.** Linux i przeznaczone dla niego programy są zaprojektowane tak, aby wspierać standardy, ponieważ standardy są językiem wolnego intelektualnego handlu. Programy dla Linuksa współdziałają ze współczesnymi aplikacjami i obsługują formaty multimedialne, dokumenty elektroniczne, arkusze kalkulacyjne i wiele więcej. Ponieważ Linux jest otwarty i darmowy, nie może przydarzyć się coś takiego jak własnościowy Linux czy własnościowy format danych. To przyczynia się zarówno do rozwoju wymiany danych pomiędzy różnymi linuksowymi programami, jak i stanowi gwarancję, że użytkownik zawsze będzie mógł uzyskać dostęp do swoich danych.
- ♦ **Niższe koszty.** Niezależnie od tego, czy Linux ma być używany na pojedynczym komputerze, czy też w firmie, dostawa jest bezpłatna, a w dodatku są całe zastępy linuksowych specjalistów, którzy pomogą zrobić wszystko, co jest potrzebne. Nie ma opłat licencyjnych, a jeżeli trzeba za coś zapłacić, to za uaktualnienia i wsparcie techniczne ze strony dostawcy dystrybucji.
- ♦ **Stabilny, wydajny i wolny od wirusów.** Linux jest dojrzałym, wielodostępowym, niezawodnym, stabilnym, wyposażonym w zabezpieczenia i odpornym na wirusy systemem — wrażliwy jest jedynie na błędy administratora.

Kiedyś do używania Linuksa niezbędna była specjalna wiedza, ale obecnie nie ma już takich wymogów. Dystrybucje, takie jak Ubuntu, sprawiły, że Linux jest prosty — właściwie przezroczysty. Czytelnik niniejszej książki będzie mógł się przekonać, że Ubuntu to łatwy w użytkowaniu system operacyjny wyposażony we wszelkie programy, które mogą być potrzebne użytkownikowi. Celem „Ubuntu. Biblia” jest eksploracja tej dystrybucji, prezentacja, jak można czerpać przyjemność z wykonywania poszczególnych zadań, oraz dostarczenie szczegółowych informacji, interesujących spostrzeżeń i niezbędnej wiedzy. Sza! Twoja babcia wcale nie musi wiedzieć, że używa Linuksa.

¹ Blokada producenta, czyli *vendor lock* to sytuacja, w której użytkownik nie może łatwo zmienić np. oprogramowania na inne z powodu braku kompatybilności. Dobrym przykładem jest tu program Płatnik służący do prowadzenia rozliczeń z ZUS-em przez przedsiębiorstwo; jego zamknięty kod uniemożliwia legalne utworzenie wersji dla systemu Linux, co wymusza używanie systemu Windows — *przyp. tłum.*

Co to jest dystrybucja?

Osoby wcześniej interesujące się Linuksem zauważyły zapewne wprawiający w zakłopotanie fakt dostępności wielu różnorodnych rodzajów systemu. W czasopismach komputerowych i internetowych serwisach poświęconych Linuksowi znaleźć można sporo informacji o dystrybucjach, takich jak Red Hat, SUSE Linux, Novell Linux Desktop, Fedora Core Linux i wielu innych, których nazwy kończą się na „Linux”, a każda z nich produkowana jest przez inną firmę czy organizację. Zrozumienie, co dokładnie kryje się za słowem „Linux”, jest kluczem do uświadomienia sobie, jak wiele różnych wersji tej samej rzeczy może być dostępnych. Ale to wymaga bliższego przyjrzenia się, w jaki sposób działają współczesne komputery z punktu widzenia programów.

Podczas uruchamiania na komputerze systemu operacyjnego, takiego jak Linux, Microsoft Windows czy Apple Mac OS X, instalowane jest oprogramowanie, którego działanie pozostaje niewidoczne dla zwykłego użytkownika, ponieważ funkcjonuje niejako poza sceną. Programy te obsługują planowanie zadań, uruchamianie i wyłączanie różnych programów, komunikację z podzespołami komputera i urządzeniami peryferyjnymi (np. drukarkami). Ogólnie można je określić nazwą oprogramowania systemowego, ponieważ użytkownik zwykle nie korzysta z niego wprost, ale jego działanie jest niezbędne dla poprawnego funkcjonowania systemu. Podstawowy zestaw oprogramowania systemowego jest zwykle nazywany **jądrem** (ang. *kernel*), ponieważ to centralny fragment oprogramowania, a wszystko inne oparte jest na usługach zapewnianych właśnie przez jądro.

Jądro jako takie nie jest specjalnie interesujące; ludzie chcą uruchamiać programy, a nie jądro systemu. Jednak te programy są zależne od usług dostarczanych przez jądro i inne aplikacje systemowe. Kiedy np. użytkownik chce wydrukować plik, potrzebne jest oprogramowanie, które przekształci tenże plik w postać czytelną dla drukarki, a następnie umieści w kolejce wydruku. Jeszcze inny program odpowiedzialny jest za wysłanie pliku do drukarki, sprawdzenie, czy plik jest drukowany poprawnie itd.

W popularnym użyciu „Linux” to zbiorcza nazwa dla jądra systemu operacyjnego oraz powiązanych aplikacji. W rzeczywistości jednak „Linux” to techniczna nazwa jądra systemowego — większość używanych w Linuksie programów wywodzi się z innych projektów wolnego oprogramowania. **Dystrybucja Linuksa** to z kolei właściwa nazwa dla zestawu oprogramowania działającego „na górze” i narzędzi do instalacji oraz konfiguracji systemu. Każda organizacja czy firma zajmująca się przygotowywaniem dystrybucji korzysta z zalet wolnego oprogramowania, jakim jest jądro oraz programy działające z nim. Firmy łączą „właściwą” wersję jądra systemu oraz zestaw oprogramowania, który uznają za „właściwy”, i udostępniają tak, by każdy mógł z nich korzystać.



Ponieważ wiele kluczowych dla działania systemu aplikacji ma swoje korzenie w projekcie GNU, dlatego poprawna nazwa powinna wyglądać następująco: GNU/*nazwa dystrybucji*. Jednakże pamiętając o dużej liczbie różnorodnych projektów, które wniosły swój wkład w dzisiejsze dystrybucje, w niniejszej książce używać będę terminu „dystrybucja” zamiast np. „GNU/GNOME/KDE/TeX/*twój-ulubiony-projekt*/dystrybucja Linuksa”. To w żaden sposób nie pomniejsza fundamentalnego wkładu, jaki w rozwój dzisiejszej informatyki wniosły projekty GNU i Free Software Foundation (FSF). Autor niniejszej książki jest członkiem FSF i zaleca, aby czytelnicy zostali nimi również. Na stronie www.fsf.org/associate można znaleźć więcej szczegółów. Zawsze dobrym pomysłem jest wspieranie spraw, w które się wierzy (i od których się w pewnym sensie zależy).

Oczywiście, zdobycie płyty CD czy DVD, które zawierają pakiet oprogramowania, będzie niemal bezcelowe, jeżeli nie będzie dostępny prosty sposób jego instalacji i konfiguracji tak, aby mogło współdziałać z innymi częściami systemu (czyli identyfikować urządzenia peryferyjne, komunikować się poprzez sieć, tworzyć konta użytkowników itd.). Dlatego każdy, kto przygotowuje dystrybucje Linuksa, dostarcza też narzędzia do instalacji i konfiguracji systemu, do których dostęp można uzyskać podczas uruchamiania systemu z płyty CD czy DVD. Narzędzia do instalacji i konfiguracji systemu należą zwykle do systemu zarządzania pakietami, dzięki któremu można łatwo dodawać zestawy powiązanych aplikacji i identyfikować zależności pomiędzy różnymi komponentami oprogramowania, co z kolei daje pewność, że aplikacje będą działać poprawnie.

Dystrybucje są kluczem do zrozumienia, jak Linux może być zarazem wolny i sprzedawany. Kod źródłowy jądra systemu oraz aplikacje o otwartym kodzie są rzeczywiście wolno dostępne na tysiącach stron internetowych. Każdy, kto chce, może je pobrać, ale połączenie ich w jedną całość, którą w dodatku da się łatwo zainstalować, to już zupełnie inna rzecz. Osoby sprzedające Linuksa zwykle pobierają opłaty za nośnik, czas i pracę poświęconą na złożenie w całość poszczególnych elementów oraz — czasami — za wsparcie techniczne udzielane podczas instalacji czy konfiguracji systemu.

Projektowanie dystrybucji i czynienie ich szeroko dostępnymi było kluczowe dla zaadoptionowania Linuksa jako systemu operacyjnego, ponieważ to właśnie dystrybucje sprawiły, że instalowanie i użytkowanie narzędzi GNU stało się dostępne dla zwykłych użytkowników.

Wprowadzenie do Ubuntu

Ubuntu to dystrybucja utworzona w roku 2004 i od początku ukierunkowana na potrzeby końcowego użytkownika. Ta dystrybucja to efekt projektu sponsorowanego przez Canonical Ltd. (www.canonical.com), firmę założoną przez Marka Shuttlewortha, działającego z sukcesami południowoafrykańskiego przedsiębiorcę, programistę Debiana z długim stażem i rzecznika ruchu oprogramowania o otwartym kodzie. Ubuntu to dystrybucja bazująca na Debianie — innej dystrybucji (więcej na ten temat nieco dalej) wykorzystującej środowisko GNOME (które szczegółowo omówiono w rozdziale 5., „Środowisko GNOME”). Siostrzane projekty to Kubuntu, czyli wersja Ubuntu wykorzystująca środowisko KDE zamiast GNOME; Xubuntu, czyli wersja, w której używane jest środowisko Xfce, oraz Edubuntu — wersja Ubuntu ukierunkowana na aplikacje edukacyjne, której zadaniem jest popularyzacja Linuksa w szkołach.

Wszystko musi mieć jakąś nazwę, ale co właściwie oznacza „ubuntu”? Jak można się spodziewać, najlepiej wyjaśniono to na stronie projektu:

Ubuntu to starodawne afrykańskie słowo oznaczające „humanitaryzm (człowieczeństwo) ukierunkowany na innych”. Oznacza ono też „jestem, kim jestem, dzięki temu, kim wszyscy jesteście”. Ta dystrybucja wnosi ducha Ubuntu do świata oprogramowania.

Być może dla niektórych to nieco zbyt emocjonalne, ale trudno polemizować z sukcesem i zaangażowaniem. W roku 2005, czyli pierwszym, kiedy Ubuntu było dostępne, dystrybucja otrzymała następujące nagrody: Linux Journal’s Reader’s Choice, Tux Magazine’s

Reader's Choice 2005 dla ulubionej dystrybucji, Ars Technica Best Distribution, UK Linux & Open Source Industry's Best Distribution i Linux World Expo dla najlepszej dystrybucji wywodzącej się z Debiana. Nie najgorzej, jak na zupełnie nowy produkt.

Oprócz swojej technicznej doskonałości i użyteczności (oraz funduszu założycielskiego Marka Shuttlewortha), swój sukces Ubuntu zawdzięcza również temu, że jego twórcy nie są tradycyjnymi linuksowymi fanatykami, chcą tworzyć oraz promować użyteczną i łatwą w obsłudze dystrybucję dla użytkowników na całym świecie.

Manifest Ubuntu

Manifest Ubuntu to oficjalne założenia programowe sformułowane w klasycznej formie manifestu, tak ulubionej przez artystów i polityków. Manifest Ubuntu dostępny jest na stronach internetowych projektu (<http://ubuntu.pl/manifest.html#FilozofiaUbuntu>). Oto jego podstawowe idee.

- ♦ Każdy użytkownik komputera powinien w sposób całkowicie wolny uruchamiać, rozpowszechniać, analizować, zmieniać i ulepszać oprogramowanie, którym się posługuje bez konieczności ponoszenia jakichkolwiek opłat licencyjnych.
- ♦ Każdy użytkownik komputera powinien mieć możliwość używania oprogramowania we właściwej mu wersji językowej.
- ♦ Każdy użytkownik komputera powinien używać dowolnego oprogramowania bez względu na swoją ewentualną niepełnosprawność oraz bez względu na charakter tej niepełnosprawności.

Pierwszy punkt jest wyraźnym nawiązaniem do ogólnych zasad propagowanych przez twórców wolnego oprogramowania, ale to drugi i trzeci przyczyniły się do sukcesu Ubuntu.

Internacjonalizacja to termin określający pisanie programów, które wyświetlają wszystkie komunikaty, okna dialogowe, wiadomości systemowe itd. w języku użytkownika wraz ze wszystkimi znakami charakterystycznymi dla danego alfabetu. Ponieważ słowo to jest dość długie, dlatego często używa się skrótu *i18n* (po literze *i* następuje 18 kolejnych, a cały wyraz — w wersji angielskiej *internationalization* — kończy się literą *n*). Dwa kolejne aspekty internacjonalizacji to **tłumaczenie**, dzięki któremu system operacyjny i okna dialogowe są dostępne w danym języku, oraz **lokalizacja**, dzięki której wiadomości i tekst mogą być wyświetlane z użyciem znaków charakterystycznych dla danego języka. Dla lokalizacji stosuje się odpowiednio skrót *l10n*.

Twórcy Linuksa oraz używanych w nim programów zajmują się *i18n* od lat, prym wiodą organizacje Li18nux (www.li18nux.net/) i Free Standards Group Open Internationalization Initiative (www.openi18n.org/). Organizacje te starają się, aby do programów o otwartym kodzie włączano *i18n* już na etapie projektowania, zarządzania oraz poprawiania kodu. Wiele ostatnich zmian w środowiskach graficznych, takich jak GNOME i KDE, miało na celu upewnienie się, czy aplikacje są w stanie obsługiwać różne języki i czcionki.

Klucze do sukcesu internacjonalizacji są dwa — nie tylko programy i środowiska graficzne muszą obsługiwać wiele języków i czcionek, ale również musi być dostępne tłumaczenie komunikatów systemowych, okien dialogowych itp. Ponieważ Ubuntu od początku

skupione było na tworzeniu dystrybucji użytecznej dla międzynarodowej społeczności, spowodowało to, że projekt stał się swego rodzaju centrum skupiającym prace translatorские i lokalizacyjne dla Linuksa (<http://www.ubuntu.com/community/participate#10n>) i GNOME (który jest podstawowym system graficznym — ale o tym później) za pośrednictwem sieciowego systemu ROSETA (<https://launchpad.net/rosetta>), wspierającego pracę tłumaczy, listy dyskusyjne i inne źródła.



Dodatkowe informacje na temat internacjonalizacji Linuksa można znaleźć w sieci, np. na stronach organizacji i18nGurus (<http://www.i18ngurus.com/docs/984813514.html>) oraz w podręczniku dostępnym pod adresem: <http://home.no.net/david/i18n.php>.

Chociaż wiele zmian w ostatnich wydaniach środowisk graficznych, takich jak GNOME czy KDE, było związanych z internacjonalizacją, ogromny postęp dokonał się również w zakresie ułatwień w użytkowaniu dla osób niepełnosprawnych. Wyposażanie programów w skróty klawiaturowe i obsługę gestów myszy dla wszystkich poleceń menu i obsługi okien dialogowych to coraz ważniejsza część projektowania aplikacji i środowisk graficznych. Poprawianie użyteczności przez Ubuntu to dobrodziejstwo, z którego korzystają wszyscy użytkownicy, niezależnie od tego, czy są niepełnosprawni, czy też nie.



Dodatkowe informacje na temat projektów skupiających się na uświadamianiu idei użyteczności można znaleźć w sieci pod następującymi adresami: Linux Accessibility HOWTO (www.tldp.org/HOWTO/Accessibility-HOWTO/) i Linux Developers Accessibility HOWTO (http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/html_single/Accessibility-HOWTO.html) (<http://larswiki.atrc.utoronto.ca/wiki/>).

Kalendarz wydań Ubuntu

Nadanie odpowiedniego tempa projektowaniu oprogramowania o otwartym kodzie jest istotne dla wszystkich, którym zależy na tym, aby dystrybucje Linuksa mogły korzystać z najnowszych wersji jądra i pakietów programów. Jądro i powiązane sterowniki urządzeń dostarczają poprawek dotyczących bezpieczeństwa, ułatwiają używanie najnowszych podzespołów i często ulepszają wydajność istniejących urządzeń i protokołów. Najnowsze wersje pakietów dostarczają zwykle poprawek bezpieczeństwa i ulepszeń wydajności. Ponieważ społeczność zajmująca się oprogramowaniem o otwartym kodzie gwarantuje, że jednocześnie pracuje się nad tysiącami poprawek, dlatego też dostarczanie, integrowanie i testowanie najnowszych wersji jądra systemu oraz oprogramowania to złożone zadanie, jednak jest ono niezmiernie istotne dla rozpowszechniania dystrybucji Linuksa i odniesienia przez nie sukcesu.

Ludzie z Ubuntu dostarczają nowe wydanie Ubuntu co sześć miesięcy. Te regularne wydania wyposażone są w aktualne i przetestowane jądro systemu, zintegrowany zestaw oprogramowania obejmujący system graficzny X Window publikowany przez x.org, ostatnie stabilne wydanie GNOME oraz kluczowe programy dla Linuksa i GNOME wraz ze specyficznymi dla Ubuntu aplikacjami i usprawnieniami.

Stały kalendarz wydań jest czymś niezwykle w przestrzeni linuksowej, jego konsekwencją jest też unikalna numeracja kolejnych wersji Ubuntu. Tradycyjny sposób numerowania wydań opierał się na wydaniach głównych (zawierających większe zmiany) i pobocznych (często ograniczających się do wprowadzania drobnych poprawek). I tak np. wydanie nr 4.2 oznacza poboczne wydanie głównego oznaczonego cyfrą 4; wydanie 4.2 było z kolei

poprzedzone wydaniem 4.1. W Ubuntu pierwsza część numeru wydania oznacza rok edycji, a druga miesiąc, i tak wydanie 6.06 zostało opublikowane w czerwcu roku 2006. W typowy dla Linuksa, beztroski sposób każdemu wydaniu nadano przezwisko. W trakcie pisania niniejszej książki kolejne wydania Ubuntu nosiły następujące nazwy:

- ♦ 4.10: Warty Warthog, czyli Piegowaty Guziec, (październik 2004),
- ♦ 5.04: Hoary Hedgehog, czyli Sędziwy Jeż, (kwiecień 2005),
- ♦ 5.10: Breezy Badger, czyli Serdeczny Borsuk, (październik 2005),
- ♦ 6.06: LTS (Long Term Support): Dapper Drake, czyli Elegancki Kaczor, wersja z długoterminowym wsparciem, (lipiec 2006),
- ♦ 6.10: Edgy Eft, czyli Zakręcona Traszka (październik 2006),
- ♦ w przygotowaniu jest 7.04 Feisty Fawn, czyli Dzielny Jelonek, (kwiecień 2007).

To dość odświeżające spojrzenie na kwestię numerowania kolejnych wydań, szczególnie w świecie oprogramowania o otwartym kodzie, które znajduje się w nieustannym rozwoju. Dzięki temu nie tylko można łatwo zaplanować aktualizację systemu do nowszej wersji, ale również łatwo zidentyfikować rocznik zainstalowanego systemu bez konieczności szukania dodatkowych informacji np. w internecie.

Uaktualnienia i wsparcie systemu w Ubuntu

Częstotliwość, z jaką publikowane są kolejne wydania, jest istotna dla każdego użytkownika, dzięki temu bowiem zyskuje on dostęp do najnowszych wersji i do najlepszego oprogramowania. Jednakże firmy zainteresowane są przede wszystkim wydajnością, obniżaniem kosztów oraz elastycznością Linuksa, ale chyba jeszcze ważniejszy jest czas, przez jaki dane wydanie będzie uaktualniane i utrzymywane. W wielu firmach działają setki czy nawet tysiące komputerów. W przedsiębiorstwach z rozwiniętą infrastrukturą wiele czasu poświęca się na testowanie poprawek i aktualizacji, zanim zostaną one zainstalowane w poszczególnych systemach komputerowych. Kiedy już proces testowania uaktualnień zostanie zakończony, wówczas następuje etap ich wdrażania, który w warunkach działów zarządzania informacjami i informatycznych bezpośrednio przekłada się na pieniądze.

Domyślnie poprawki bezpieczeństwa są publikowane przez kolejnych 18 miesięcy od daty wydania kolejnej wersji. Po tym okresie istniejące poprawki są nadal dostępne, ale nie ma gwarancji, że pojawią się kolejne aktualizacje. Aby wyjść naprzeciw komercyjnym użytkownikom Ubuntu oczekującym długoterminowego wsparcia, Mark Shuttleworth oraz Canonical Ltd. powołali i ufundowali w połowie 2005 roku fundację Ubuntu. Fundacja zatrudnia niektórych kluczowych członków społeczności Ubuntu, aby zagwarantować ciągłość procesu rozwoju i wydawania dystrybucji. Jednym z aspektów pracy fundacji jest zapewnienie wsparcia i aktualizacji. Wydanie 6.06 było pierwszym, które ma zapewnione wsparcie przez 3 lata dla systemów biurkowych i 5 lat dla serwerów, co jest dwukrotnym przedłużeniem standardowego 18-miesięcznego wsparcia dla pozostałych edycji Ubuntu. Aby podkreślić ten fakt, do nazwy dodano skrót LTS oznaczający właśnie długoterminowe wsparcie (ang. *Long Term Support*).

Regularne wydania Ubuntu pozwalają dotrzymywać kroku rozwojowi sprzętu oraz oprogramowania o otwartych źródłach, a także upewniają użytkowników, że zawsze mają stabilne, bezpieczne i aktualne wersje oprogramowania, z którego korzystają. Dostarczają też uaktualnień i zapewniają wsparcie, a są to elementy niezbędne do odniesienia przez Ubuntu sukcesu w komercyjnych zastosowaniach.

Ubuntu a Debian

Debian to jedna z najdłużej istniejących i wciąż dostępnych dystrybucji Linuksa. Jest to też dystrybucja, z której wywodzi się Ubuntu. Nazwa Debian pochodzi od imion założycieli projektu, Debry i Iana Murdocków.

Projekt Debian (www.debian.org) został powołany w roku 1993 i od tego czasu oferuje jakość GNU/Linux. Debian jest dobrze znany ze stabilnych wydań opartych na ogromnych zbiorach gruntownie przetestowanego i zupełnie zintegrowanego oprogramowania. Niestety, konieczność dogłębnego testowania i integrowania oprogramowania prowadzi do boleśnie małej liczby wydań. Pomiedzy oficjalnymi wydaniem Debiana dosłownie upływają lata. Jednak uczciwość wymaga, by dodać, że zawsze są dostępne trzy wersje aktualnego wydania: stabilna (właściwe wydanie), testowa (kandydat do następnego wydania) i niestabilna (wersja dla programistów). Jednak wielu użytkowników zarówno domowych, jak i „firmowych” nie lubi dyskomfortu wynikającego z używania czegoś **testowego** czy **niestabilnego**. Szybszy proces publikowania nowych wersji, skupienie się na podstawowych technologiach, takich jak np. interfejs użytkownika GNOME, i opracowanie lepszego mechanizmu dostarczania uaktualnień oraz powiadamiania o nich to kluczowe przyczyny powstania projektu Ubuntu.

Oto kilka podstawowych przykładów współzależności Ubuntu i Debiana.

- ♦ W Ubuntu wykorzystuje się format pakietowania używany w Debianie, podobnie jak doskonałą i imponującą technologię do identyfikowania i rozwiązywania zależności oraz relacji pomiędzy różnymi pakietami oprogramowania o otwartych źródłach, która pochodzi również z Debiana.
- ♦ Programiści Ubuntu wprowadzają zmiany i usprawnienia do pakietów oprogramowania o otwartym kodzie, które przekazują społeczności, ale dostarczają je także wprost do programistów Debiana odpowiedzialnych za pakiety, podobnie jak wszelkie informacje umieszczają w systemie śledzenia błędów Debiana. Poprawki oraz powiązane usprawnienia przygotowywane przez programistów Ubuntu są dostarczane w takim stanie, w jakim zostały przygotowane podczas procesu tworzenia i testowania wydania bez specjalnego rozgłosu. Ta metoda jest lepsza dla wszystkich.
- ♦ Debian i Ubuntu nieco się różnią w doborze oprogramowania, ale korzystają z tej samej metody organizowania pakietów w osobne zestawy, co omówiono w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Dystrybucje Debian i Ubuntu są blisko powiązane, wzajemnie się uzupełniają, choć ich twórcom przyświecają różne cele. Ubuntu nie mogłoby istnieć bez pionierskich wysiłków i wkładu programistów Debiana, ale dostarcza bardziej przewidywalną dystrybucję z lepszymi kanałami wsparcia dla wielu użytkowników indywidualnych, a także dla firm z rozbudowanymi działami informatycznymi.

Dlaczego warto wybrać Ubuntu?

Jak wspomniałem wcześniej, jeśli spojrzeć na półki z czasopismami komputerowymi w lokalnej księgarni, można na nich znaleźć dziesiątki magazynów z dystrybucjami Linuksa. W końcu wszystkie są darmowe, prawda? Większość użytkowników Linuksa (poza specjalistami) to osoby, które usłyszały, że powinny używać określonej dystrybucji, a zatem kupiły czasopismo z nośnikiem, na którym znajdowała się jakaś dystrybucja bądź też książkę poświęconą konkretnej dystrybucji.

Ubuntu oznacza „humanitaryzm ukierunkowany na innych”, ale nagłówek na stronie internetowej projektu głosi „Linux dla ludzi” (ang. *Linux for people*) i to właśnie oddaje istotę rzeczy: to dystrybucja dla tych, którzy chcą wykonać swoją pracę bez niepotrzebnego zamieszania. I nic nie szkodzi, że jest to również technicznie skomplikowany Linux z aktualnym oprogramowaniem. Czy można z jego pomocą wykonać swoją pracę?

Odpowiedź jest bezdyskusyjna: „Tak!”. Jeżeli jednak czytelnik jest nieprzekonany lub planuje wziąć udział w przyjęciu, na którym omawiany będzie Linux, i potrzebuje więcej danych empirycznych, oto kilka atrybutów Ubuntu, które czynią z niego dystrybucję atrakcyjną dla niemal każdego.

- ♦ **Regularne i aktualne wydania.** Jądro systemu oraz tysiące pakietów oprogramowania, które składają się na środowisko użytkownika i administratora, są stale aktualizowane. Jak wspomniałem wcześniej, dostarczanie najnowszego i najlepszego jądra oraz aplikacji w regularnych wydaniach to jedna z fundamentalnych zasad Ubuntu.
- ♦ **Ukierunkowanie na jakość.** Jakość dystrybucji linuksowych zależy od dwóch rzeczy: po pierwsze, jak dobra jest ona sama w sobie, a po drugie, jak szybko dostawca jest w stanie usuwać pojawiające się problemy. W obu przypadkach Ubuntu zdecydowanie się wyróżnia. Każde wydanie przechodzi szczegółowe wewnętrzne testy wykonywane przez członków zespołu Ubuntu oraz publiczne testy prowadzone przez społeczność, kiedy już osiągnie status kandydata do wydania (ang. *release candidate*). Po opublikowaniu wydania uaktualnienia wypuszczane co najmniej przez 18 miesięcy (oczywiście wtedy, gdy są potrzebne).
- ♦ **Wsparcie społeczności i komercyjne.** Spora część wsparcia dystrybucji pochodzi od społeczności, trudno przebić pasję i zaangażowanie członków społeczności Ubuntu. Tak jak żadna firma nie może sobie pozwolić na korzystanie z systemu operacyjnego bez rozsądnych gwarancji usunięcia jego usterek, tak też żadna firma nie może być uzależniona od systemu operacyjnego, który pozbawiony jest gwarantowanej obsługi technicznej. W dalszej części rozdziału omówiono pełne możliwości wsparcia zarówno komercyjnego, jak i za pośrednictwem społeczności.
- ♦ **Łatwe uzyskiwanie programów i uaktualnień.** W poprzednich punktach położyłem nacisk na możliwość instalacji aktualnego systemu operacyjnego. Ubuntu wyposażone jest w znakomite narzędzie powiadamiające użytkownika o dostępności uaktualnień i ułatwiające ich instalację. Graficzny menedżer aktualizacji oraz menedżer pakietów Synaptic (omówione szerzej w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”) są najlepszymi narzędziami tego typu, jakich kiedykolwiek używał autor niniejszej książki.

- ♦ **Ukierunkowanie na użyteczność.** Ubuntu definiowane jest jako „Linux dla ludzi” i wyposażono je w grafikę, dekoracje okien, tematy kolorystyczne, które mają uatrakcyjnić i uczynić użytecznym środowisko dla zwykłych ludzi, którzy muszą po prostu wykonać swoją pracę. Podobnie jak w każdej innej dystrybucji, można swobodnie dostosowywać system do własnych potrzeb, korzystać z jednego z wielu dostępnych menedżerów okien, które łatwo pozyskiwać i instalować za pomocą Synaptica. Domyślnie w Ubuntu wykorzystuje się środowisko GNOME, które jest znane z dobrego wsparcia oraz ukierunkowania na jak największą przystępność, co objawia się m.in. w możliwości wykorzystywania klawiatury do obsługi menu.
- ♦ **Ukierunkowanie na internacjonalizację.** Dla niektórych jest niespodzianką, że na tej planecie żyją ludzie, którzy nie mówią po angielsku, podobnie jak to, że w innych językach nie korzysta się z alfabetu angielskiego i takiego zestawu czcionek. Niestety, to nie żart. Ubuntu jest nadzwyczajnie ukierunkowane na wspieranie tłumaczenia i dostarczanie swojej dystrybucji tak, aby ludzie mogli korzystać z niej w swoim własnym języku i z własnym zestawem czcionek.
- ♦ **Aktywna i zaangażowana społeczność.** Trudno znaleźć bardziej aktywną, dynamiczną i zaangażowaną społeczność od tej, która otacza Ubuntu (szerzej omówiono to w następnym rozdziale). Bezpośrednio przekłada się to na większą ilość miejsc, w których można zadawać pytania, większą szansę uzyskania odpowiedzi i przyjaźniejsze doświadczenia w trakcie poszukiwań.

Jak wynika z powyższej listy, Ubuntu skupia się na rozwiązywaniu kwestii, które są problemami w innych dystrybucjach lub które utrudniają nowym użytkownikom zaadaptowanie Linuksa jako własnego systemu operacyjnego. W większości dystrybucji Linuksa problemy są podobne, ale Ubuntu jest wyjątkiem, gdy weźmiemy pod uwagę sposób ich rozwiązywania.

Wymagania instalacyjne

Od czasu, kiedy w dystrybucjach Linuksa zaczęto używać jąder z serii 2.6, a sam Linux zaczął być coraz bardziej popularny, szansa, że w komputerze użytkownika znajduje się sprzęt, który nie będzie obsługiwany, jest coraz mniejsza. Poniżej opisałem rodzaje architektur obsługiwanych przez Ubuntu oraz ogólne wymagania sprzętowe tego systemu.



Jednym z najlepszych aspektów Linuksa jest fakt, że wspiera on szeroką gamę architektur. Większość dystrybucji — włączając Ubuntu — można uruchomić na starszym sprzęcie, który w przeciwnym razie służyłby zapewne tylko jako dodatkowa półka. Jednakże oprogramowanie działa wolniej na wolniejszym sprzęcie (czytelnicy mogą cytować tę opinię). Korzystając z takiego komputera, użytkownik zapewne będzie chciał używać trybu tekstowego lub środowiska graficznego o mniejszych wymaganiach sprzętowych niż domyślne GNOME (czy KDE w Kubuntu). Więcej informacji na ten temat można znaleźć w rozdziale 5., w podrozdziale „Co to jest pulpit? Środowiska graficzne dla Linuksa”.

Obsługiwane architektury

Ubuntu można uruchomić na każdej z wymienionych poniżej architektur systemowych:

- ♦ i386 lub kompatybilne procesory firm Intel, AMD, Cyrix itd.,
- ♦ G3 lub nowsze systemy PowerPC (PPC) firmy Apple Macintosh,
- ♦ 64-bitowe procesory AMD lub EM64T (co obejmuje następujące typy: Athlon64, Opteron i EM64T Xeon),
- ♦ systemy Sun UltraSPARC.



W czasie pisania niniejszej książki wydanie Ubuntu 6.06 dla architektury UltraSPARC wciąż miało status nieoficjalnego, choć obie firmy (Ubuntu i Sun) ogłosiły, że ta dystrybucja będzie obsługiwać procesory UltraSPARC, szczególnie model T1 (o nazwie kodowej „Niagara”) używany w systemach Sun Fire T1000 i serwerach T2000.

Wymagania sprzętowe

Jak w każdym programie komputerowym, lepsze wyniki osiągniemy, instalując Ubuntu na najmocniejszym dostępnym komputerze, chociaż z technicznego punktu widzenia system będzie poprawnie — choć wolno — działał również na komputerach typu i386 25 MHz. W tym przypadku jednak doskonale sprawdza się amerykański slogan „Więcej to lepiej”. Porzucając na chwilę misję nawracania świata na Linux, można powiedzieć, że nie warto zwracać sobie głowy instalowaniem Ubuntu na systemach z procesorem wolniejszym niż 166 MHz i ilością pamięci mniejszą niż 96 MB. Autor używa właśnie komputera o takich parametrach do prowadzenia testów (to stary ThinkPad 380XD firmy IBM) i trzeba przyznać, że GNOME w takiej konfiguracji działa nieznośnie wolno. Jeżeli użytkownik chce koniecznie uruchomić Ubuntu na podobnym sprzęcie, warto sprawdzić wcześniej podaną informację o korzystaniu z alternatywnych środowisk graficznych przeznaczonych dla mniej wydajnego sprzętu.

Minimalne wymagania sprzętowe dla zainstalowania i w miarę przyzwoitego korzystania z Ubuntu są następujące:

- ♦ procesor o częstotliwości 700 MHz lub lepszy,
- ♦ 256 MB pamięci,
- ♦ napęd CD-ROM,
- ♦ karta sieciowa,
- ♦ karta graficzna typu VGA,
- ♦ 3 GB dostępnej przestrzeni dyskowej.

Jeżeli komputer czytelnika spełnia te wymogi, wszystko zmierza w dobrą stronę. Można, oczywiście, zainstalować Ubuntu na wolniejszym komputerze wyposażonym w mniejszą ilość pamięci, ale będzie to jak zakładanie sportowego ogumienia na „malucha” — najlepsze doznania i tak pozostaną niedostępne.

Czas instalacji

Czas potrzebny na instalację Ubuntu zależy od wydajności komputera, sposobu konfiguracji systemu oraz typu instalowanej dystrybucji. Instalacja systemu na komputerze przenośnym, na którym już działa Microsoft Windows czy Mac OS X, w taki sposób, aby dostępna była opcja podwójnego rozruchu, może zająć ponad godzinę. Instalacja na nowym komputerze bez systemu powinna zamknąć się w około pół godziny. Podsumowując, na instalację należy zaplanować ok. 2 godziny, jeżeli okaże się, że potrzeba było mniej czasu, czytelnik z pewnością nie będzie narzekał.

Płyty instalacyjne Ubuntu

Płyta CD dołączona do niniejszej książki, *Ubuntu Desktop CD*, umożliwia przetestowanie Ubuntu na komputerze z zainstalowanym już innym systemem bez konieczności zmieniania czegokolwiek na dysku komputera. Na płycie znajduje się też prosty w obsłudze program instalacyjny, dzięki któremu można umieścić system na dysku komputera na stałe. Aktualnie dostępne są trzy różne płyty CD dla różnych architektur. Oto krótki opis ich cech.

- ♦ **Desktop CD** (do instalacji na komputerach biurkowych). Wersja jest dołączona do niniejszej książki; jest to rozruchowa wersja Ubuntu, której uruchomienie nie powoduje żadnych zmian na dysku komputera. Wersja ta, znana również jako *live CD*, zawiera też prosty w obsłudze program instalacyjny umożliwiający umieszczenie Ubuntu na stałe na dysku komputera. I wreszcie, na płycie można znaleźć kilka popularnych programów o otwartym kodzie w wersji dla Windows, takich jak np. OpenOffice. Więcej informacji o korzystaniu z tej płyty czytelnik znajdzie w rozdziale 2., „Instalowanie Ubuntu”.



Przy korzystaniu z systemu *live CD* każda wykonana praca, każdy utworzony plik itd. zostaną utracone podczas ponownego uruchomienia komputera, chyba że wcześniej zapiszemy je na innym komputerze dostępnym w sieci lub na przenośnym nośniku pamięci, takim jak dysk USB, przenośny dysk twardy itp. W rozdziale 2. znajduje się punkt („Tryb Persistence”) opisujący, w jaki sposób można wykorzystać pamięć flash lub przenośny dysk twardy do automatycznego zapisywania i przywracania zmian wprowadzonych podczas pracy z płytą *live CD*. W rozdziale 2. można też znaleźć informacje, jak ręcznie zapisywać prace wykonane w systemie uruchamianym z płyty CD („Uzyskiwanie dostępu do dysku twardego z systemu Desktop CD” i „Kopiowanie plików do innego komputera dostępnego w sieci”).

- ♦ **Server CD**. Umożliwia instalację Ubuntu w wersji przeznaczonej dla komputerów używanych jako serwery. Można zainstalować podstawowy serwer, a następnie dodać oprogramowanie serwerowe zgodnie z potrzebami, ale można też zainstalować serwer LAMP (Linux, Apache, MySQL, Perl) z pakietami potrzebnymi do działania takiego serwera internetowego. Żadna z tych wersji nie zawiera instalatora w trybie graficznym, chociaż użytkownik może go dodać samodzielnie. Więcej informacji na temat płyty CD i instalowania z niej systemu można znaleźć w rozdziale 3., w podrozdziale zatytułowanym „Menu instalacyjne płyty Server Install CD”.
- ♦ **Alternate CD**. Umożliwia instalację Ubuntu na systemach o specjalnych charakterystykach sprzętowych lub specyficznych konfiguracjach. Obejmuje

komputery wstępnie przygotowane do redystrybucji w ramach programu OEM (ang. *Original Equipment Manufacturers*), uaktualnienie istniejących systemów bez dostępu do sieci i skonfigurowanie automatycznej instalacji Ubuntu na wielu systemach. Zaawansowani użytkownicy mogą zainstalować system, wykorzystując LVM (ang. *Logical Volume Management*), RAID (ang. *Redundant Arrays of Inexpensive Disks*), mogą też zdecydować, czy GRUB ma zostać zainstalowany gdzie indziej, a nie głównym sektorze rozruchowym dysku (MBR, czyli ang. *Master Boot Record*), czy zainstalować system na komputerach wyposażonych w mniej niż 192 MB pamięci RAM. Więcej informacji na temat tej wersji płyty CD i instalowania z niej systemu znajduje się w rozdziale 3., w podrozdziale „Menu instalacyjne płyty Server Install CD”.

Płyta *desktop CD* dołączona do niniejszej książki jest wersją najczęściej wykorzystywaną przez osoby instalujące Ubuntu. Jednak w zależności od systemu, jaki użytkownik będzie chciał utworzyć, do jego dyspozycji pozostawiono kopie innych wersji instalacyjnych. Strona internetowa Ubuntu oferuje bezpłatnie obrazy ISO wszystkich dostępnych wersji Ubuntu na wszystkie platformy (<http://us.releases.ubuntu.com/releases>). Pliki ISO to obrazy dysków CD zapisane zgodnie ze standardem International Standards Organization CD, które można pobrać z internetu, a następnie samodzielnie nagrać na nośniki. Wystarczy odnaleźć łącze do ostatniego wydania, wybrać odpowiedni plik, pobrać go na dysk komputera, a następnie nagrać na płytę CD. Można też poprosić ludzi z Ubuntu o przesłanie płyty instalacyjnej z dystrybucją — temu służy program *ShipIt*. Aby to zrobić, należy przejść na stronę <https://shipit.ubuntu.com>, następnie utworzyć nowe konto, podając swój adres poczty elektronicznej oraz hasło, i poprosić o przesłanie płyty z odpowiednim wydaniem Ubuntu. Można jednocześnie zamówić kilka płyt przeznaczonych dla różnych systemów. Program *ShipIt* to doskonałe rozwiązanie, jeżeli czytelnik chciałby zdobyć płyty z systemem dla swoich przyjaciół, ale jeżeli czytelnikowi zależy na jak najszybszej dostawie, warto pamiętać, że przesyłka może wędrować nawet kilka tygodni.

Wsparcie techniczne dla Ubuntu

Ze swojej natury oprogramowanie komputerowe wymaga, aby użytkownik — od czasu do czasu — zadał pytanie na temat, jak go używać, czy poprosił kogoś o pomoc w rozwiązaniu jakiegoś problemu. Szczególnie sprawdza się to dla oprogramowania linuksowego, gdzie oprócz instalowania setek programów należy zainstalować również system operacyjny, od którego te programy zależą. Główną przewagą systemów operacyjnych sprzedawanych w pudełkach, takich jak Microsoft Windows czy Apple Mac OS X, jest fakt, że w przypadku kłopotów z zainstalowaniem, skonfigurowaniem czy używaniem można — prawdopodobnie — skontaktować się ze sprzedawcą. W rzeczywistości kontakty te pozostawiają wiele do życzenia (proszę spróbować skontaktować się z pomocą techniczną Microsoftu i poinformować autora o wynikach). Książki, takie jak ta, dostarczają informacji na temat instalowania, konfigurowania i w ogóle porad typu „jak zrobić...”, ale zawsze pojawi się jakieś pytanie, którego nie można było przewidzieć. Dlatego tak ważne jest, aby wiedzieć, jak uzyskiwać odpowiedzi na swoje pytania oraz gdzie szukać pomocy, kiedy będzie potrzebna.

Ubuntu oferuje imponujący wachlarz możliwości uzyskania wsparcia, poczynając od pomocy oferowanej przez społeczność aż po płatne wsparcie sponsora Ubuntu, firmę Canonical, Ltd., a także szereg innych firm na całym świecie. W dalszej części rozdziału omówiono wiele miejsc, w których można zadawać pytania, uzyskiwać odpowiedzi, zamawiać płatne wsparcie, a nawet wynająć eksperta, który pomoże rozwiązać potencjalne problemy.

Spółeczność Ubuntu — wsparcie i informacje

Ponieważ oprogramowanie w Linuksie zależy od modelu programowania opartego na społeczności, bezpłatna pomoc często oferowana jest na tych samych zasadach. To jedna z tych dziedzin, w których Ubuntu naprawdę się wyróżnia: listy dyskusyjne, blogi i fora są doskonałymi źródłami zdobywania aktualnych informacji. Umożliwiają one zadawanie doświadczonym użytkownikom określonych pytań i uzyskiwanie odpowiedzi od tych, którzy rozwiązali dany problem. Te strony są również pożywką dla twórców dystrybucji, ponieważ dzięki nim dowiadują się, z jakimi problemami najczęściej mierzą się użytkownicy i które z nich należy rozwiązać w następnym wydaniu.

Blogi

Blogi (nazwa pochodzi od zbitki *Web log*) to popularne określenie, ostatnio pasujące do wszystkich, którzy posiadają klawiaturę i dostęp do stron internetowych. Blog Ubuntu, znany jako Planet Ubuntu (<http://planet.ubuntulinux.org/>) jest trochę inny od tradycyjnych blogów, ponieważ nie ma w nim wypowiedzi tylko jednej osoby. Jest to miejsce, w którym programiści oraz członkowie społeczności mogą dzielić się różnorodnymi pomysłami, spostrzeżeniami, porażkami i sukcesami.

Fani Ubuntu, którzy chcą być na bieżąco z informacjami tam zamieszczanymi, a niekoniecznie za każdym razem odwiedzać stronę internetową, mogą skorzystać z udogodnień technicznych i dokonać subskrypcji wiadomości, ponieważ są one dostępne w następujących formatach (dzięki technologii RDF, czyli *Resource Description Framework*):

- ♦ FOAF (ang. *Friend of a Friend*), dostępne pod adresem <http://planet.ubuntulinux.org/foafroll.xml>,
- ♦ OPML: (ang. *Outline Processor Markup Language*), dostępne pod adresem <http://planet.ubuntulinux.org/opml.xml>,
- ♦ RSS (ang. *Really Simple Syndication* lub *Rich Site Summary* — zależnie od tego, kogo zapytać); Planet Ubuntu dostępne jest w formacie RSS 1.0 (<http://planet.ubuntulinux.org/rss10.xml>) i RSS 2.0 (<http://planet.ubuntulinux.org/rss20.xml>); należy zwrócić uwagę na wybór odpowiedniej wersji czytnika dla danego formatu,
- ♦ Planet-pl, czyli stronę utrzymywaną przez polskich członków społeczności (i co ważne w języku polskim) można znaleźć pod adresem <http://planet.ubuntu-pl.org/rss20.xml>.

Fora internetowe

Fora internetowe to najnowsza generacja czegoś, co kiedyś było znane pod nazwą BBS-ów; jest to atrakcyjna alternatywa dla list dyskusyjnych — wymaga jedynie czasu na odwiedzanie stron internetowych, na których znajdują się fora. Forum Ubuntu mieści się na www.ubuntu.com.

ubuntuforums.org/. Na stronie tej znaleźć można ogromny wybór dobrze zorganizowanych forów tematycznych, dzięki czemu dużo łatwiej wyszukiwać potrzebne informacje, umieszczać własne pytania, rozmawiać lub czytać wypowiedzi innych korzystających z forów. I wcale nie musisz być zaawansowanym członkiem społeczności. Aby przeglądać forum, nie trzeba być zarejestrowanym użytkownikiem, jest to jednak konieczne, kiedy chce się umieszczać własne wpisy. Proces rejestracji jest łatwy, wystarczy przejść na stronę [www.ubuntuforums.org/register.php](http://ubuntuforums.org/register.php) i przeczytać oraz zaakceptować zasady tego forum. Następnie należy podać nazwę użytkownika, pod którą chce się być widocznym na forum, wpisać hasło oraz podać adres poczty elektronicznej i kilka podstawowych informacji o sobie.

Na stronie indeksu forum (www.ubuntuforums.org/index.php) można znaleźć kategorie, według których zorganizowano witrynę, oto one.

- ♦ **Beginner Community** (czyli społeczność początkujących). Sekcja forum poświęcona dla *absolutnie początkujących użytkowników*, gdzie można znaleźć odpowiedzi na ogólne pytania dotyczące komputerów, Linuksa, Uniksa i Ubuntu. Jeżeli czytelnik obawia się zadawać pytania, ponieważ sądzi, że są zbyt podstawowe, to jest forum przeznaczone dla niego!
- ♦ **Ubuntu Release Assistance**. Składają się na nie trzy kolejne fora:
 - ♦ **News and Announcements**: ważne doniesienia związane z Ubuntu, kolejnymi wydaniem, informacjami o kwestiach bezpieczeństwa i o samym forum Ubuntu,
 - ♦ **Main Support Categories**: rozbudowana część podzielona na wiele dodatkowych kategorii (m.in.: dla użytkowników komputerów Apple z procesorami Intel, o środowisku graficznym, multimedia i wideo, instalowanie i aktualizacje itd.); można szukać tu odpowiedzi na pytania dotyczące wszystkich wydań i wersji Ubuntu (również Kubuntu, Xubuntu i Edubuntu),
 - ♦ **Other Support Categories**: tutaj zamieszczono dyskusje, które nie zmieściły się w działach poprzedniej sekcji, czyli m.in. takie jak projektowanie nowych wersji, edukacja i nauka, repozytoria, przewodniki i porady, serwery i kwestie bezpieczeństwa itd.
- ♦ **Third-Party Ubuntu Projects**. Tutaj toczono są dyskusje na temat projektów związanych ze społecznością Ubuntu, ale niesponsorowanych i niewspieranych przez Canonical, Ubuntu czy Ubuntu Forum. Należy tu umieszczać wpisy związane tylko z tymi projektami, a są to m. in.: Easy Ubuntu, Ubuntu Woman, Ubuntu Christian Edition czy testowanie płyt CD i DVD.
- ♦ **Community Discussions**. Część forum poświęcona ogólnym dyskusjom i doniesieniom, podzielona jest na dwie sekcje:
 - ♦ **Forum Community**: tutaj można znaleźć informacje o bieżącej działalności społeczności Ubuntu i tego, co dzieje się w poszczególnych działach, można sprawdzić nowiny z kawiarni (Ubuntu Café) oraz z lodówki (więcej szczegółów nieco dalej),
 - ♦ **Ubuntu LoCo Teams**: na ogólny sukces Ubuntu składa się praca dziesiątek różnorodnych zespołów porzucanych po całym świecie, tutaj można znaleźć ich listę oraz adresy.

Chociaż inne dystrybucje mają podobne fora internetowe (np. Fedora Core na stronie www.fedoraforum.org), jednak to jest o tyle wyjątkowe, że odzwierciedla zupełnie wyjątkowe podejście do użytkowników zarówno aktualnych wydań, jak i tych wcześniejszych, które nadal posiadają oficjalne wsparcie.

Dodatkiem do omówionego forum prowadzonego w języku angielskim są fora prowadzone w innych językach, co nadaje Ubuntu prawdziwie międzynarodowy charakter. Nie jest to po prostu przetłumaczone forum anglojęzyczne, w wielu przypadkach fora te różnią się układem i organizacją. Odnośniki do tych stron można znaleźć na www.ubuntulinux.org/community/forums. W czasie pisania niniejszej książki dostępne były fora w językach chińskim, niderlandzkim, fińskim, francuskim, niemieckim, włoskim, polskim i portugalskim. Ubuntu to prawdziwie międzynarodowy wysiłek!

Na podobnych zasadach działa polskie forum użytkowników Ubuntu, które znaleźć można na stronie <http://forum.ubuntu.pl/>; aby być jego aktywnym członkiem, również należy przejść procedurę rejestracji. Forum to jest — oczywiście — mniejsze od anglojęzycznego, ale ilość wpisów jest i tak imponująca, a użytkownicy, którzy zastosują się do zasad i zwyczajów tam panujących, mogą liczyć na znalezienie pomocy.

IRC

Internet Relay Chat, czyli popularny IRC, to mechanizm umożliwiający prowadzenie w internecie interaktywnych rozmów o czymkolwiek. Anglojęzyczny kanał Ubuntu IRC nosi nazwę `#ubuntu` i jest dostępny za pośrednictwem sieci irc.freenode.net. Oczywiście, dostępne są również kanały w innych językach: polskim (`#ubuntu.pl`), chińskim (`#ubuntu-zh`), niderlandzkim (`#ubuntu-nl`), niemieckim (`#ubuntu-de`), hebrajskim i arabskim (`#ubuntu-il`), włoskim (`#ubuntu-it`), portugalskim (`#ubuntu-pt`), rosyjskim (`#ubuntu-ru`) i hiszpańskim (`#ubuntu-es`) (stan w czasie pisania książki).

Kanały IRC to znakomite narzędzie działające w czasie rzeczywistym, które może służyć do rozwiązywania problemów i uzyskiwania pomocy *online* (oczywiście, przy założeniu, że dane problemy nie dotyczą łączenia się z internetem). Program Xchat do korzystania z IRC-a instalowany jest domyślnie jako część Ubuntu, dzięki czemu nawiązywanie połączeń oraz korzystanie z zalet wsparcia i zasobów społeczności online jest zupełnie proste.

Listy dyskusyjne

Listy dyskusyjne to świetne rozwiązanie, dzięki któremu pytania i odpowiedzi dostarczane są bezpośrednio do odpowiedniej osoby, czyli inaczej niż na forach dyskusyjnych, gdzie najpierw należy połączyć się z odpowiednią stroną, a następnie wyszukać potrzebne informacje samodzielnie.

Istnieje ogromna liczba list dyskusyjnych, wiele z nich wyspecjalizowało się w określonych zagadnieniach, dlatego wymienianie tu ich wszystkich byłoby po prostu marnowaniem papieru, ponieważ i tak należy zapisać się na nie w sieci. Podstawowy zestaw list dyskusyjnych dotyczących Ubuntu można znaleźć na stronie www.ubuntulinux.org/community/lists, obejmuje on m.in.:

- ♦ **Ubuntu Announcement** (*ubuntu-announce*) — pojawia się na niej bardzo niewiele nowych wiadomości, mniej niż jedna na miesiąc, służy do informowania członków o nowych wydaniach Ubuntu oraz znaczących nowych projektach,
- ♦ **Ubuntu Development** (*ubuntu-devel*) — przeznaczona jest do prowadzenia dyskusji o zaawansowanych technicznych i programistycznych aspektach aktualnego wydania,
- ♦ **Ubuntu Security Announcement** (*ubuntu-security-announce*) — na tę listę nie można wysyłać wiadomości, natomiast uzyskiwane z niej informacje dotyczą publikowanych poprawek bezpieczeństwa; jest to wyjątkowo przydatne narzędzie dla administratorów systemów, którzy tu mogą uzyskiwać najnowsze informacje dotyczące aktualizacji związanych z bezpieczeństwem systemu,
- ♦ **Ubuntu User** (*ubuntu-users*) — to niesłychanie aktywna lista przeznaczona na zagadnienia wsparcia technicznego; użytkownicy Ubuntu mogą wysyłać na nią swoje zapytania,
- ♦ **Ubuntu Women** (*ubuntu-women*) — w zamierzeniu jest to lista dla wszystkich użytkowniczek i użytkowników, ochotniczek i ochotników, programistek i programistów chcących zachęcić więcej kobiet do czynnego udziału w życiu społeczności Ubuntu.

Oczywiście, dostępnych jest o wiele więcej list dyskusyjnych poświęconych Ubuntu, wymienione powyżej to tylko mała część. Kompletną listę można znaleźć na stronie <http://lists.ubuntu.com>, skąd użytkownik zostanie przeniesiony na stronę <http://lists.ubuntu.com/mailman/listinfo>. Do zarządzania listami używane jest popularne oprogramowanie MailMan.

Na stronie można też znaleźć listę dyskusyjną prowadzoną przez Polaków, jej adres to: <https://lists.ubuntu.com/mailman/listinfo/ubuntu-pl>. Aby zostać jej członkiem, należy wypełnić krótki formularz rejestracyjny i już po chwili można brać aktywny udział w społeczności Ubuntu.



Dla fanów Usenet listy dyskusyjne Ubuntu są również dostępne jako grupy w Usenecie, a wszystko dzięki ludziom z Gmane (www.gmane.org). Powiązane z Ubuntu grupy można znaleźć na stronie: <http://news.gmane.org/index.php?prefix=gmane.linux.ubuntu>.

Co jest w lodówce?

Serwis Fridge (czyli lodówka), który można znaleźć na stronie <http://fridge.ubuntu.com>, służy do udostępniania najświeższych informacji dotyczących wszelkich nowin ze świata Ubuntu. Można tu znaleźć wieści o zbliżających się spotkaniach zespołów, konferencjach, najnowsze wiadomości z sieci poświęcone Ubuntu, informacje zespołów projektowych oraz biuletyny informacyjne, takie jak Ubuntu Desktop News i Ubuntu Documentation News.

Dokumentacja

W przypadku tradycyjnego oprogramowania dokumentacja, która ma służyć użytkownikowi pomocą, zwykle przyjmuje postać papierowego podręcznika i jest dostarczana razem z programem. Nikomu chyba nie trzeba wyjaśniać, dlaczego i jak ważna jest dobra dokumentacja. Nie ma znaczenia, jak dobry jest program lub system operacyjny, jeżeli użyt-

kownik nie może z niego korzystać. Dokumentacja Linuksa to interesujący przypadek, ponieważ w większości przypadków można ją bezpłatnie pobrać z internetu. Istnieje kilka dystrybucji Linuksa, które można kupić w sklepach komputerowych w pudełkach, ale nawet w ich przypadku papierowe instrukcje są zwykle bardzo skromnych rozmiarów. Ze względu na tempo rozwoju Linuksa oraz ogromną ilość sprzętu, który można zainstalować w Linuksie, drukowanie kompletnej dokumentacji jest trudne do wykonania, nie wspominając już o samym zarządzaniu takim przedsięwzięciem. Warto jednak wspomnieć o firmie Novell, producencie SUSE Linux, która słynie z wysokiej jakości przygotowywanej przez siebie dokumentacji produktu dołączanej do pudełkowych wersji programów.

W Ubuntu działają duże i dobrze zorganizowane zespoły zajmujące się tworzeniem wysokiej jakości dokumentacji ukierunkowanej na użytkownika, której zadaniem jest ułatwienie pracy z systemem operacyjnym. Dokumentacja Ubuntu dostępna jest w następujących lokacjach:

- ♦ <http://help.ubuntu.com>: źródło całej dokumentacji tworzonej przez zespoły Ubuntu,
- ♦ <https://wiki.ubuntu.com/UserDocumentation>: zhierarchizowany zbiór zasobów w formacie Wiki, dzięki czemu łatwiej odnaleźć poszukiwane zagadnienia.

Dokumentacja Ubuntu jest dobrze zorganizowana, tworzące ją zespoły stosują się do ustalonych zasad, które mają na celu uzyskanie tego rodzaju spójności, jakiego można oczekiwać od nakładów pracy wniesionych w jej napisanie. Strona domowa całego projektu to <http://doc.ubuntu.com/>, można na niej przeglądać już opracowane dokumenty, jak również te, które znajdują się w trakcie przygotowywania. Cała komunikacja pomiędzy członkami zespołu odbywa się za pomocą IRC-a, pocztą elektroniczną i list dyskusyjnych opisanych na stronie <https://wiki.ubuntu.com/DocteamCommunications>. Wszyscy zainteresowani udziałem w projekcie tłumaczeń mogą sprawdzić listę obecnie prowadzonych prac na stronie <https://wiki.ubuntu.com/DocteamProjects>.

Polscy użytkownicy systemu również mogą korzystać z dokumentacji, wyraźnym wkładem polskiej społeczności są tłumaczenia przewodników, jeden z nich można znaleźć na stronie http://ubuntuguide.org/wiki/Ubuntu_dapper_pl.

Komercyjne wsparcie techniczne dla Ubuntu

Jak wspomniano wcześniej, dla firm planujących wdrożenie rozwiązań linuksowych szczególnie ważna jest dostępność gwarantowanej obsługi technicznej. Choć w większości przypadków odpowiedzi na wiele wątpliwości oraz rozwiązania problemów można znaleźć w sieci, to jednak kierownicy działów IT niechętnie akceptują odpowiedź typu „Można to wygooglać” zwłaszcza wtedy, gdy firma pozostaje bez dostępu do sieci lub część pracowników nie może wykonywać swoich obowiązków z powodu problemów z programami czy systemem operacyjnym. Właśnie dlatego dla firm istotne jest, aby mieć określone zasoby, które pomogą rozwiązać potencjalne problemy w jak najkrótszym czasie.

Choć duża część tego rozdziału poświęcona była zaletom i użyteczności rozbudowanej społeczności Ubuntu, to jednak nie można zapominać, że dostępne jest również komercyjne wsparcie, którego szczegóły opisano poniżej.

Płatne wsparcie oferowane przez firmę Canonical Ltd.

Canonical Ltd., sponsor Ubuntu, oferuje dwa różne poziomy płatnego wsparcia dla swojej dystrybucji, są to odpowiednio Standard i Premium. Warunki techniczne oraz finansowe można znaleźć na stronie www.ubuntu.com/support/supportoptions/paidsupport. Oto podstawowe różnice pomiędzy tymi dwoma pakietami.

- ♦ Wersja Standard: pomoc jest udzielana za pośrednictwem połączeń internetowych, a gwarantowany czas reakcji to dwa dni robocze; pakiet obejmuje 10 interwencji rocznie.
- ♦ Wersja Premium: pomoc jest udzielana za pośrednictwem połączeń internetowych i telefonicznych, gwarantowany czas reakcji to jeden dzień roboczy; pakiet obejmuje 25 interwencji rocznie; reakcja telefoniczna gwarantowana jest w czasie 4 godzin.

Canonical oferuje również wsparcie w razie kłopotów z oprogramowaniem, zależy ono jednak od rodzaju używanych programów, czy raczej ich położenia w strukturze. Zgodnie z informacjami zawartymi w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, aplikacje znajdujące się w repozytoriach Ubuntu są pogrupowane w kilka różnych klas, jest to głównie uzależnione od rodzaju licencji, na jakiej są udostępniane. Pełne wsparcie oferowane jest dla klasy **main**, częściowe dla klasy **restricted**, natomiast nie oferuje się wsparcia dla klas **universe** i **multiverse**. W rozdziale 20. można znaleźć więcej szczegółów na temat organizacji repozytoriów Ubuntu i różnic pomiędzy klasami pakietów.

Canonical oferuje również specjalne płatne certyfikaty dla firm chcących świadczyć usługi serwisowe użytkownikom domowym.

The Ubuntu Marketplace

Ubuntu Marketplace to część witryny Ubuntu, na której można znaleźć informacje na temat firm oferujących pomoc techniczną dla komputerów biurkowych oraz serwerów działających pod kontrolą Ubuntu. Stronę można znaleźć pod adresem www.ubuntu.com/support/supportoptions/marketplace, znajduje się tam baza firm z całego świata, w których można zamówić pomoc techniczną. Na stronie znajdują się odnośniki do pięciu regionów świata: Afryki, Azji, Europy, Ameryki Łacińskiej, Ameryki Północnej i Oceanii. Powiązane łącze można znaleźć na stronie forum (www.ubuntuforums.org) wspomnianej wcześniej, na której aktualnie znajduje się więcej informacji, aniżeli obecnie na stronie Ubuntu Marketplace.

Taka scentralizowana baza zawierająca informacje na temat firm świadczących usługi dla przedsiębiorstw, które zamierzają migrować na Linuksa, jest bezcenna, zwłaszcza że można znaleźć firmy znajdujące się najbliżej danej lokalizacji.

Podczas pisania niniejszej książki nie było określonych certyfikatów, które byłyby wymagane, aby dana firma mogła znaleźć się na stronie Ubuntu Marketplace — wystarczy na adres marketplace@ubuntu.com wysłać wiadomość zawierającą informacje wskazane na stronie www.ubuntu.com/support/supportoptions/marketplace/join.

Uzyskiwanie dodatkowych informacji o Ubuntu

Popularność Ubuntu zwiększa się w tempie nienotowanym dotąd dla żadnej dystrybucji Linuksa. Techniczna doskonałość z jednej strony, świetna organizacja i rozmach witryny internetowej z drugiej, przyczyniają się do coraz większej popularności. Strona Ubuntu przeznaczona jest raczej dla fanów i użytkowników tej dystrybucji, jednak ważne jest, co piszą inni. Czy zatem w internecie są inne miejsca, w których można szukać pomocy i informacji o Ubuntu?

Na to pytanie zdecydowanie należy odpowiedzieć twierdząco. W sieci można znaleźć dużą ilość stron zawierających informacje o Ubuntu i to bardziej szczegółowe niż tylko ogólne wzmianki. Oto uzupełnienie informacji o wsparciu technicznym i i innych tematach omówionych w tym rozdziale; poniżej znajduje się lista związanych z Ubuntu i ulubionych stron autora.

- ♦ **DistroWatch** (<http://distrowatch.com/table.php?distribution=ubuntu>) zawiera informacje o większości dystrybucji Linuksa, również o Ubuntu. Znaleźć tam można szczegółowe informacje na temat zawartości poszczególnych wydań, ale co ważniejsze, znajdują się tam też łącza do innych stron zawierających informacje o Ubuntu, recenzje, nowości i wiele więcej. Internauci nieznający jeszcze serwisu DistroWatch koniecznie powinni odwiedzić tę kopalnię informacji o dystrybucjach Linuksa.
- ♦ **Ubuntu Blog** (<http://ubuntu.wordpress.com/>) to blog poświęcony Ubuntu, w którym zamieszczany jest doskonały wybór informacji o Ubuntu posegregowanych w następujących kategoriach: administracja, biuro, serwery itd. Znajdują się tam również łącza do innych stron zawierających różnego rodzaju poradniki, np. jak uruchomić MythTV w Ubuntu.
- ♦ **UbuntuGuide** (<http://ubuntuguide.org/>) to miejsce, w którym znajdują się znakomite przewodniki dla początkujących, choć może nie zawsze nadążają one za aktualnymi wydaniem. Warto tam zajrzeć, ponieważ można znaleźć solidną porcję informacji na temat wykonywania różnych zadań w Ubuntu.
- ♦ **Ubuntu Women** (www.ubuntu-women.org/) to witryna, na której można znaleźć FAQ, wiki, blog i wiele innych materiałów mających zachęcić kobiety do szerszego udziału w projekcie Ubuntu oraz FLOSS (*Free/Libre/Open-Source Software*). To wspaniała strona, z ważnym przesłaniem, dzięki któremu wszyscy możemy wygrać.
- ♦ **Ubuntux** (www.ubuntux.org/) to społeczność użytkowników Ubuntu oferująca różnorodne fora internetowe, blogi, łącza do wielu artykułów o Debianie i Ubuntu oraz szereg różnych materiałów, których przeznaczeniem jest niesienie pomocy wszystkim korzystającym z Ubuntu. Szczególnie przyjemne są fora internetowe, ponieważ omawiane tematy obejmują szerokie spektrum zagadnień: od samych początków z Ubuntu aż po zaawansowane dostosowywanie i optymalizowanie systemu. Na Ubuntux porusza się również zagadnienia dotyczące projektów towarzyszących, takich jak Kubuntu, Edubuntu i Ubuntu Lite, a także wydań przeznaczonych na serwery. Dostępne są także kanały RSS, dzięki którym można być na bieżąco z najnowszymi nowinami ze świata Ubuntu.

W sieci znaleźć można wiele interesujących stron powiązanych z Ubuntu, a każdego dnia pojawiają się kolejne. Na DistroWatch umieszczono listę ciekawych witryn, ale jak to zwykle bywa w sieci, najlepszym przyjacielem internauty powinna być ulubiona wyszukiwarka, za jej pomocą szybko będzie można sprawdzić setki innych stron, na których łatwo odszukać odpowiedzi na postawione pytania czy po prostu dowiedzieć się, co myślą o Ubuntu inni.

Podsumowanie

Ubuntu jest najszybciej rozwijającą się dystrybucją Linuksa i jednym z najlepszych przykładów potęgi oprogramowania o otwartych źródłach oraz siły społeczności, jaką do tej pory można było zaobserwować. Po zaprezentowaniu ogólnych wiadomości o Linuksie omówiłem w tym rozdziale filozofię stojącą za Ubuntu oraz założenia przyjęte dla tej dystrybucji. Duża część sukcesu ma swoje źródło w technicznej doskonałości oraz łatwości użytkowania, jak również w szybko rozwijającej się społeczności i dobrze zorganizowanej witrynie internetowej, udostępniającej w przyjazny sposób różnego rodzaju informacje i zasoby dotyczące Ubuntu.

Rozdział 2.

Instalowanie Ubuntu

W tym rozdziale:

- ◆ Uruchamianie systemu z płyty Desktop CD
- ◆ Instalacja Ubuntu
- ◆ Testowanie Ubuntu
- ◆ Uzyskiwanie dostępu do istniejących partycji z poziomu live CD
- ◆ Zapisywanie prac wykonanych w systemie live CD
- ◆ Instalacja programów dla Windows dostępnych z płyty CD

Płyta CD dołączona do niniejszej książki zawiera Ubuntu 6.06 LTS. Desktop CD to przykład systemu *live CD* umożliwiającego eksperymentowanie ze wstępnie skonfigurowanym oprogramowaniem bez dokonywania jakichkolwiek modyfikacji na dysku komputera. Płytę *live CD* warto potraktować jak przenośny system, za którego pomocą można czasowo przekształcić w stację roboczą niemal każdy komputer, niezależnie od tego, czy zainstalowany jest na nim Microsoft Windows, Mac OS X czy inna dystrybucja Linuksa. Ubuntu w wersji *live CD* oferuje również imponującą funkcję, która umożliwia zapisanie efektów pracy oraz ustawień systemu na nośnikach wymiennych, dzięki czemu można je przywrócić podczas następnego uruchomienia systemu z płyty. W większości przypadków po uruchomieniu systemu z płyty CD można nawet uzyskać dostęp do zamontowanych partycji, co pozwala na zapisanie efektów pracy wprost na dysku komputera i skorzystanie z nich za pośrednictwem innego systemu operacyjnego.

W Ubuntu Desktop CD rozwinięto koncepcję płyt *live CD* umożliwiającą szybką instalację systemu za pomocą zintegrowanego i łatwego w użyciu instalatora, który dostępny jest wprost z systemowego biurka. W niniejszym rozdziale skupiłem się na zastosowaniu wersji Desktop CD do zainstalowania Ubuntu na komputerach z rodziny x86 oraz 64-bitowych, a także na systemach typu PowerPC (Apple G3, G4 i G5) bez zachowywania istniejących na nich danych. W systemach x86 i 64-bitowych instalator Ubuntu potrafi automatycznie zmniejszyć rozmiar istniejących partycji systemu Microsoft Windows, a zwolnioną przestrzeń dyskową można wykorzystać na instalację Ubuntu (oczywiście przy założeniu, że istniejące partycje Windows nie są całkowicie wykorzystane).



Instalacja Ubuntu na komputerach PowerPC (PPC) z zachowaniem istniejącego systemu i opcją podwójnego rozruchu została omówiona w rozdziale 3., w podrozdziale zatytułowanym „Tryb podwójnego rozruchu — informacje”. Jeżeli użytkownik chce zachować dane istniejące na dysku komputera PPC, a nie dysponuje co najmniej 3 GB wolnej przestrzeni dyskowej, wówczas musi dokładnie wykonać instrukcje repartycjonowania dysku przed instalacją na nim Ubuntu. W rozdziale 3. omówiłem również, jak przygotować dysk komputera, gdy znajduje się na nim kilka partycji Windows lub jeżeli użytkownik będzie chciał dokonać tego samodzielnie.

Jeżeli nawet użytkownikowi komfortowo pracuje się z istniejącym systemem Microsoft Windows czy Mac OS X i chce, aby pozostały one na dysku, nadal może zauważyć wiele korzyści płynących z używania Ubuntu Desktop CD. Można wykorzystać tę płytę do eksperymentowania z Ubuntu oraz oprogramowaniem o otwartych źródłach dostępnych wraz z tą dystrybucją. Płyta Ubuntu Desktop CD zawiera również wersje wielu popularnych programów o otwartych źródłach przeznaczonych dla Windows, zatem użytkownicy mogą korzystać z ulubionych programów, instalując je w systemie Microsoftu wprost z płyty CD.

Płyta Ubuntu Desktop CD dołączona do niniejszej książki jest przeznaczona dla użytkowników systemów 32-bitowych rodziny x86. W tym rozdziale napisałem też, w jaki sposób uzyskać Ubuntu *live CD* w wersji odpowiedniej do używanego komputera, a także jak uzyskać dostęp do istniejących partycji i plików wprost z *live CD*, co umożliwi pracę z nimi; wyjaśniłem również, w jaki sposób zachować efekty pracy wykonanej za pomocą systemu uruchamianego z płyty CD. Na końcu rozdziału omówiłem aplikacje o otwartym kodzie, zamieszczone na płycie dołączanej do książki, które można zainstalować na komputerach typu x86 oraz 64-bitowych z uruchomionym systemem Microsoft Windows.

Uzyskiwanie płyt Desktop CD dla komputerów 64-bitowych i PPC

Płyta Ubuntu Desktop CD dołączona do niniejszej książki przeznaczona jest dla komputerów PC x86. Użytkownicy komputerów z 64-bitowymi procesorami lub PPC muszą uzyskać wersję systemu skompilowaną i skonfigurowaną do działania właśnie z tymi typami komputerów. Jeżeli użytkownik posiada łącznie szerokopasmowe (lub mnóstwo wolnego czasu) oraz nagrywkę CD, najprostszym sposobem na zdobycie Ubuntu będzie pobranie z internetu pliku ISO. Obrazy ISO to pliki zawierające „zdjęcie” zawartości płyt CD lub DVD, jest to standard zatwierdzony przez International Standards Organization pod numerem 9660. Aby pobrać plik ISO, należy przejść na stronę www.ubuntu.com/download, następnie wybrać serwer lustrzany znajdujący się najbliżej i określić odpowiedni dla posiadanego komputera rodzaj płyty CD. Następnie trzeba zapisać plik na dysku, nagrać go na płytę CD i gotowe!

Wolne łącznie internetowe lub brak nagrywarki CD nie są powodami do paniki. Jak wspomniałem wcześniej, ludzie z Ubuntu dostarczą odpowiednią płytę CD, choć może potrwać to kilka tygodni, ponieważ będzie wysyłana z Holandii. Nie jest to najlepsze rozwiązanie dla tych, którzy potrzebują natychmiastowego rozwiązania, ale może okazać się jedyne. Aby złożyć zamówienie na płytę z Ubuntu, należy przejść na stronę <https://shipit.ubuntu.com>,

utworzyć konto, podając swój adres oraz hasło, i zamówić jeden lub więcej nośników z odpowiednim wydaniem. Można też jednocześnie zamówić darmowe płyty dla różnych rodzajów komputerów.

Uruchamianie systemu z płyty Desktop CD

Uruchamianie systemu z płyty Desktop CD jest łatwe: wystarczy umieścić odpowiednią płytę CD w napędzie, uruchomić ponownie komputer i upewnić się, że system jest uruchamiany z płyty, a nie z partycji dysku twardego. W komputerach PC o kolejności wczytywania systemu operacyjnego decyduje BIOS, podczas gdy w komputerach PPC ustawienie te można zmienić, przytrzymując wciśnięty klawisz *C* podczas uruchamiania komputera.



Jeżeli komputer uruchomił system operacyjny z partycji dysku, mimo że w napędzie CD-ROM znajdowała się właściwa płyta Ubuntu Desktop CD, wówczas należy zmienić w systemie BIOS kolejność uruchamiania urządzeń. Kolejność rozruchu określa, jak po kolei sprawdzane są urządzenia dostępne po włączeniu komputera w poszukiwaniu systemu operacyjnego. Aby wprowadzić zmiany, należy włączyć komputer i wcisnąć klawisz, który pozwoli uzyskać dostęp do systemu BIOS. W większości współczesnych komputerów jest to zwykle klawisz *Del* (*Delete*) lub *F2*; informacja o tym, który to klawisz, wyświetlana jest w dolnym lewym rogu ekranu podczas uruchamiania komputera.

W zależności od typu systemu BIOS zainstalowanego w komputerze, kolejność rozruchu zwykle można zmienić w menu *Advanced Settings* lub *Boot*, do których można przejść za pomocą klawiszy ze strzałkami umieszczonych na klawiaturze. Po podświetleniu odpowiedniej pozycji należy wcisnąć klawisz *Enter*, aby przejść do szczegółowego menu. Następnie przy użyciu klawiszy ze strzałkami trzeba przejść do pozycji *First Boot Device* lub *CD Drive* i sprawdzić za pomocą wskazówek znajdujących się po prawej stronie ekranu, w jaki sposób ustawić napęd CD jako pierwsze urządzenie startowe. Po dokonaniu zmiany trzeba wcisnąć klawisz *Escape*, aby wyjść z menu, a następnie *F10*, wtedy nowe ustawienia zostaną zapisane. Teraz wystarczy ponownie uruchomić komputer.

Podczas uruchamiania systemu z płyty Ubuntu Desktop CD zostanie wyświetlony ekran widoczny na rysunku 2.1.

To menu startowe¹ umożliwia użytkownikowi podjęcie decyzji, w jaki sposób chce wykorzystać płytę *live CD*: czy chce uruchomić system operacyjny, naprawić istniejący system, uruchomić system z dysku twardego, tym samym ignorując obecność płyty w napędzie, czy też przetestować płytę lub pamięć operacyjną komputera.

Aby uruchomić Ubuntu, należy wcisnąć klawisz *Enter* lub poczekać 30 sekund, po upływie tego czasu rozpocznie się automatyczne uruchamianie systemu. Standardowy pulpit Ubuntu przedstawiono na rysunku 2.2.

Po uruchomieniu Ubuntu można zainstalować system na dysku zgodnie z zamieszczonymi niżej wskazówkami, ale można też zapoznać się z systemem, używając do tego celu menu umieszczonych w górnym lewym rogu ekranu oraz folderu *Examples* zawierającego przykładowe pliki (zawartość folderu została opisana w podrozdziale niniejszego rozdziału

¹ Polscy użytkownicy mogą uruchomić Ubuntu w polskiej wersji językowej, wystarczy na ekranie widocznym na rysunku 2.1 wcisnąć klawisz *F2* i z wyświetlonej listy wybrać język polski — *przyp. tłum.*



Rysunek 2.1. Opcje uruchamiania systemu dostępne w menu startowym płyty Desktop CD

zatytułowanym „Testowanie Ubuntu”). Podczas zapoznawania się z systemem można sprawdzić działanie różnych programów, a nawet przetestować ich instalowanie i konfigurowanie.

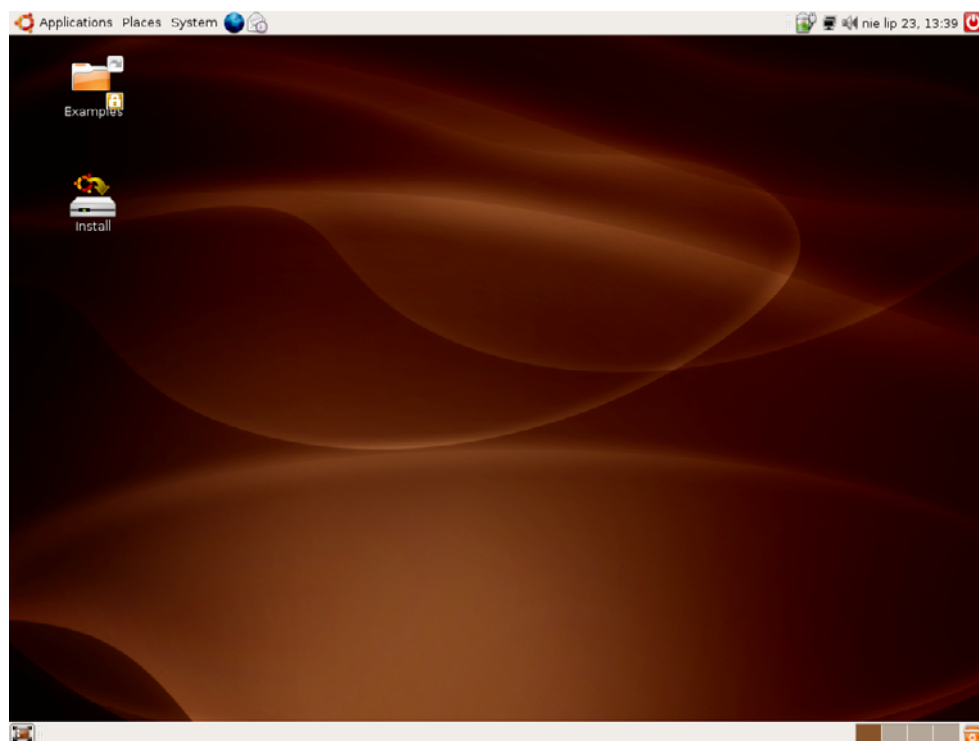
Instalacja Ubuntu z płyty Desktop CD

Po uruchomieniu systemu z płyty Desktop CD można użyć graficznego programu instalacyjnego, aby na stałe umieścić Ubuntu na dysku komputera. Instrukcje zamieszczone w niniejszej książce opisują, jak zainstalować biurkową wersję Ubuntu, która znajduje się na płycie CD dołączonej do książki. W rozdziale 3., „Instalowanie Ubuntu jako systemu specjalnego zastosowania”, zamieszczono informacje na temat instalowania innych wersji systemu, takich jak przeznaczone na serwery czy do dystrybucji typu OEM. Wymaga to jednak pobrania odpowiedniego rodzaju pliku ISO ze stron projektu Ubuntu. Przegląd różnych rodzajów obrazów płyt CD z Ubuntu można znaleźć w podrozdziale rozdziału 1. zatytułowanym „Płyty instalacyjne Ubuntu”.



Dla zamieszczonych poniżej wskazówek przyjęto założenie, że instalacja Ubuntu będzie przeprowadzana na nowym komputerze lub takim, na którym nie będzie potrzeby zachowania istniejących danych, plików czy programów znajdujących się na dysku (lub dyskach). Jeżeli na komputerze jest już zainstalowany inny system operacyjny, który należy zachować, wskazówki instalacyjne znajdzie czytelnik w rozdziale 3., w podrozdziale „Tryb podwójnego rozruchu — informacje”.

Podczas procesu instalacji w celu poprawnego skonfigurowania systemu użytkownik będzie musiał podać kilka informacji dotyczących używanego języka i lokalizacji. Aby zainstalować Ubuntu, należy wykonać następujące czynności.

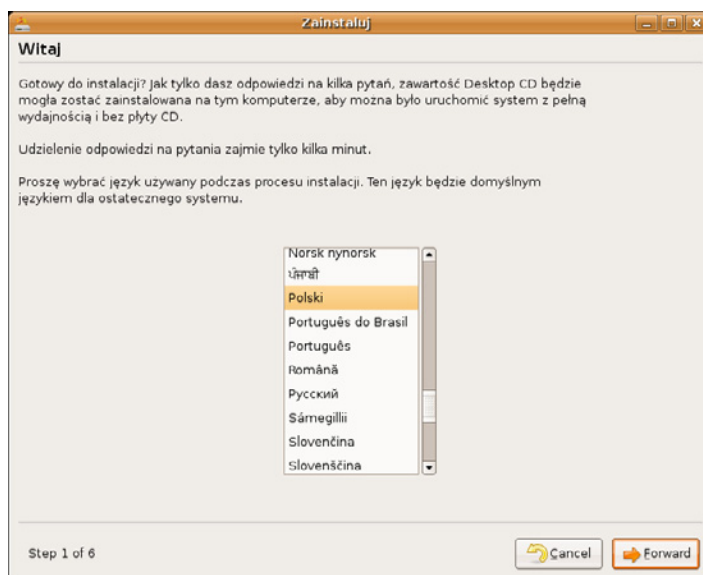


Rysunek 2.2. Standardowy pulpit Ubuntu dostępny po uruchomieniu systemu z płyty Desktop CD

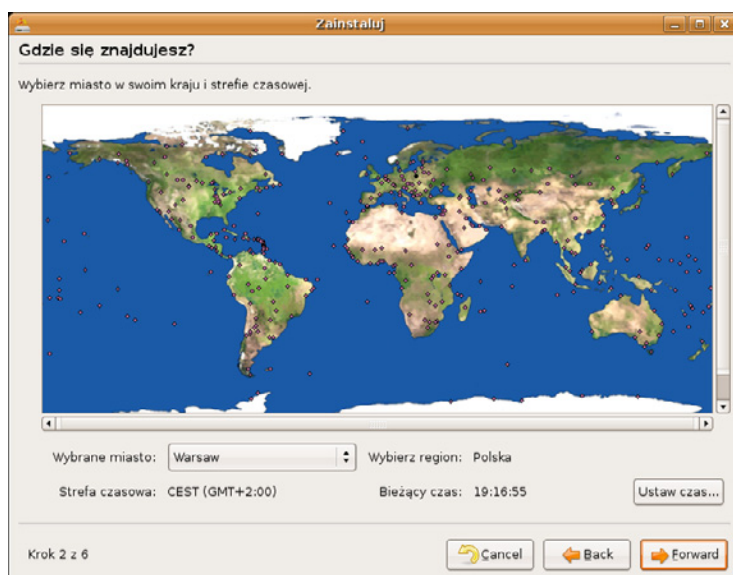
1. Aby rozpocząć proces instalacji, trzeba kliknąć dwukrotnie ikonę *Install* znajdującą się na pulpicie (patrz rysunek 2.2). W oknie dialogowym widocznym na rysunku 2.3 można wybrać język, w którym prowadzony będzie proces instalacji. Do wyboru odpowiedniego języka można użyć klawiszy ze strzałkami.
2. W oknie dialogowym widocznym na rysunku 2.4 można określić lokalizację, w której znajduje się komputer. Ta informacja zostanie wykorzystana do dalszego dostosowywania ustawień systemu: formatu daty, formatu liczb, waluty oraz lokalizacji geograficznej. Jeżeli miasto wskazane jest niewłaściwe, trzeba zmienić wybór, rozwijając listę, z której należy wybrać takie, jakie znajduje się najbliżej Twojego miejsca zamieszkania (warto zwrócić uwagę, aby znajdowało się ono w tej samej strefie czasowej).
3. Jeżeli wyświetlana godzina lub strefa czasowa są niewłaściwe (rysunek 2.4), należy kliknąć przycisk *Ustaw czas* (patrz rysunek 2.5). W innym przypadku trzeba przejść do następnego etapu, klikając przycisk *Forward*. W razie konieczności dokonania zmiany godziny lub strefy czasowej należy kliknąć przycisk *Ustaw czas*, zostanie wówczas wyświetlona mapa, na której będzie można wskazać miasto znajdujące się najbliżej. Do wprowadzenia poprawnego czasu można też użyć pola tekstowego znajdującego się u góry okna lub kliknąć przycisk *Synchronize Now*, aby program instalacyjny połączył się z internetowymi serwerami i ustawił czas właściwy dla danej strefy. Aby kontynuować, trzeba kliknąć przycisk *Forward*.

Rysunek 2.3.

*Wybór języka,
który będzie używany
w procesie
instalacyjnym*

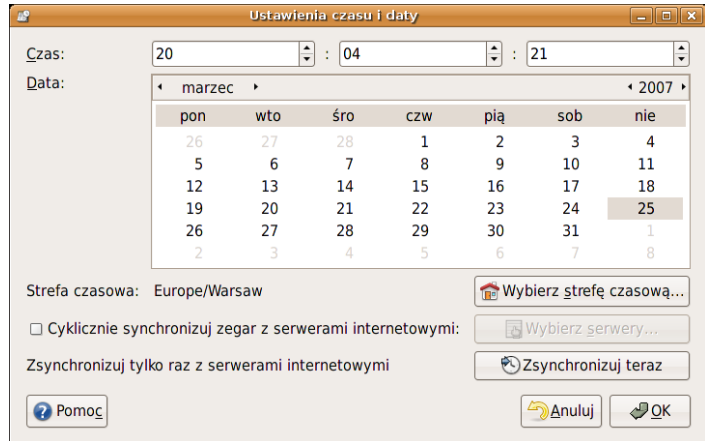
**Rysunek 2.4.**

*Określanie lokalizacji
geograficznej*

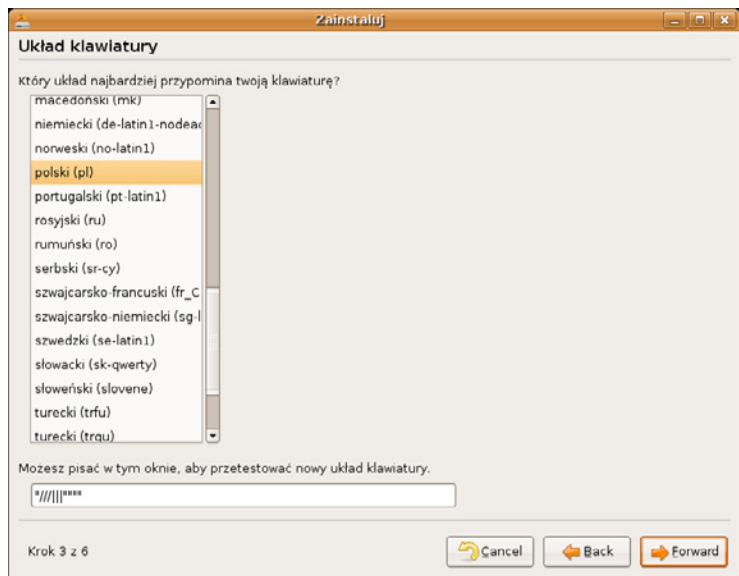


4. Okno dialogowe przedstawione na rysunku 2.6 umożliwia ustawienie odpowiedniego układu klawiatury. Wartość domyślna jest określona na podstawie informacji wprowadzonych przez użytkownika w poprzednim oknie dialogowym — jeżeli układ jest nieodpowiedni, należy wybrać właściwy z listy. Aby kontynuować, trzeba kliknąć przycisk *Forward*.
5. Okno dialogowe widoczne na rysunku 2.7 umożliwia wprowadzenie informacji o użytkowniku, na podstawie których zostanie w procesie instalacji utworzone konto. To konto pozwala na wykonywanie w systemie poleceń, dla których

Rysunek 2.5.
Ustawianie strefy
czasowej i godziny



Rysunek 2.6.
Określanie układu
klawiatury



niezbędne są specjalne uprawnienia (więcej informacji znajduje się w rozdziale 4., w punkcie „Wykonywanie czynności wymagających specjalnych uprawnień”. Po wprowadzeniu potrzebnych informacji trzeba kliknąć przycisk *Forward*.

6. W oknie dialogowym widocznym na rysunku 2.8 można określić, gdzie ma zostać zainstalowany Ubuntu. Wygląd okna może się różnić, wszystko zależy od sposobu konfiguracji partycji oraz ich zawartości. Wybór odpowiedniej opcji jest uzależniony od konfiguracji komputera oraz tego, gdzie ma zostać zainstalowany Ubuntu.
 - ♦ Podczas instalacji na nowym komputerze lub w sytuacji, gdy nie ma potrzeby zachowywania istniejących danych, programów czy plików, trzeba wybrać opcję *Wyczyść całą dysk*.

Rysunek 2.7.
Wprowadzanie informacji o użytkowniku

Kim jesteś?

Jak się nazywasz?

Jakiej nazwy chcesz używać podczas logowania się?

Jeśli komputer będzie używany przez wiele osób, można dodać kolejne konta po zakończeniu instalacji.

Ustaw hasło, aby twoje konto było bezpieczne.

Podaj to samo hasło dwukrotnie, aby uniknąć pomyłki przy wprowadzaniu.

Jaka ma być nazwa tego komputera?

Nazwa zostanie użyta, jeśli uczynisz ten komputer widocznym w sieci.

Krok 4 z 6

Cancel Back Forward

Rysunek 2.8.
Wybór lokalizacji, w której ma zostać zainstalowany Ubuntu

Przygotowywanie miejsca na dysku

Czy chcesz zmienić układ partycji na tym dysku?

☒ Zmień rozmiar IDE1 master, partycja #1 (hda1) i wykorzystaj uwolnione miejsce

☐ Wyczyść cały dysk: IDE1 master (hda) - 82.0 GB Maxtor 6Y080L0

☐ Użyj największej, ciągłej, wolnej przestrzeni

☐ Modyfikuj ręcznie tablicę partycji

Nowy rozmiar partycji: 60% (44.5 GB)

Krok 5 z 6

Cancel Back Forward

- ♦ Podczas instalowania Ubuntu na komputerach x86 lub 64-bitowych, na których znajduje się już Microsoft Windows i nie ma wolnej przestrzeni na dysku, należy wybrać opcję *Zmień rozmiar* i przesunąć suwak znajdujący się na dole okna tak, aby wskazywał żadaną wielkość partycji.



Suwak i opcja *Zmień rozmiar* są dostępne tylko podczas instalacji Ubuntu na komputerach x86 lub 64-bitowych, na których znajduje się już system Microsoft Windows.

- ♦ Gdy w komputerze znajduje się kilka dysków twardych, wszystkie zostaną wyświetlone w oknie dialogowym. Trzeba wybrać pierwszy na liście dysk niezawierający żadnych danych, które powinny zostać zachowane. Podczas wybierania odpowiedniego dysku należy zachować ostrożność!
- ♦ Gdy na dysku komputera dostępna jest pusta, nieprzydzielona przestrzeń (o takim przypadku wspomniano w rozdziale 3.), należy wybrać opcję *Użyj największej, ciąglej wolnej przestrzeni*.
- ♦ Jeśli jesteś doświadczonym użytkownikiem Linuksa i chcesz samodzielnie określić sposób partycjonowania dysku, wybierz opcję *Modyfikuj ręcznie tablicę partycji* i sprawdź dodatkowe informacje na temat używania graficznego programu do partycjonowania dysków w ramce pt. „Samodzielne partycjonowanie dysku”.



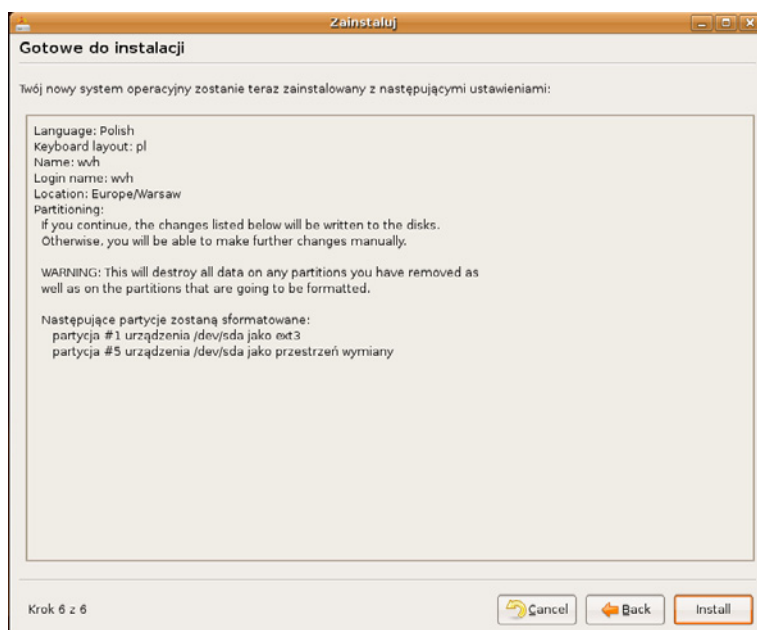
Jak wspominałem wcześniej, program instalacyjny Ubuntu dla komputerów PPC nie jest w stanie automatycznie zmienić rozmiaru istniejących partycji systemu Mac OS X, tak jak ma to miejsce w przypadku komputerów z rodziny x86 i 64-bitowych. W tej sytuacji są dwa rozwiązania: albo zezwolimy programowi instalacyjnemu na automatyczne usunięcie istniejącej partycji z systemem Mac OS X, albo samodzielnie zmniejszymy jej rozmiar przed rozpoczęciem instalowania Ubuntu. Informacje na temat zmiany wielkości partycji systemu Mac OS X można znaleźć w rozdziale 3., w podrozdziale „Tryb podwójnego rozruchu — informacje”.

Po ustaleniu, która opcja będzie najlepsza dla używanego systemu, trzeba kliknąć *Forward*.

7. Na rysunku 2.9 przedstawiono ostatnie okno programu instalacyjnego Ubuntu, w którym wyświetlone zostały informacje o wszelkich wprowadzonych zmianach i ustawieniach. Aby rozpocząć proces instalacji, trzeba kliknąć *Install*.

Rysunek 2.9.

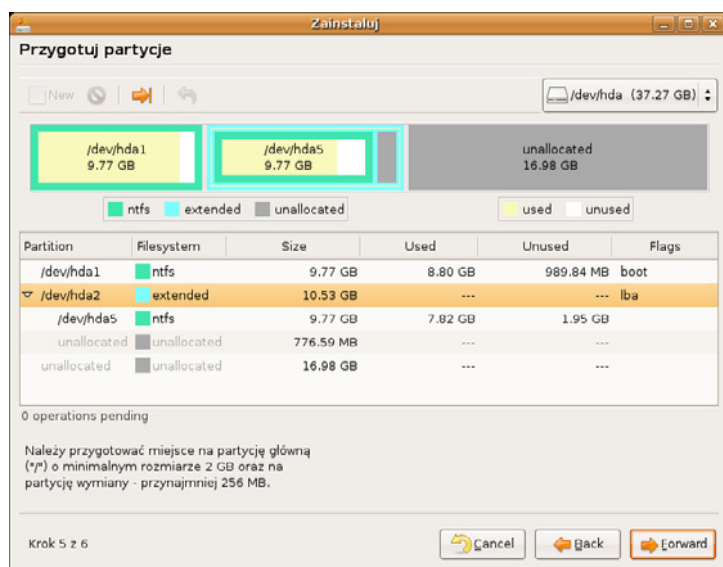
Okno dialogowe podsumowujące ustawienia, które będą zastosowane w procesie instalacji Ubuntu



Samodzielne partycjonowanie dysku

Zaawansowani użytkownicy Linuksa oraz osoby posiadające szczególne powody mogą podzielić dysk (lub wykorzystać opcję podwójnego rozruchu) zgodnie ze swoimi potrzebami. Dla tych, którzy nie chcą samodzielnie partycjonować dysku, program instalacyjny przeprowadzi całą operację automatycznie, tworząc nowe partycje (czyli partycję wymiany, tzw. *swap*, której wielkość jest wielokrotnością ilości dostępnej w systemie pamięci operacyjnej oraz partycję podstawową). Oczywiście, użytkownik może podzielić dostępną przestrzeń dyskową w inny sposób, nie tak, jak zaproponuje to program instalacyjny. Informacje na temat dysków i partycji można znaleźć w rozdziale 4., w punkcie zatytułowanym „Dyski, partycje i punkty montowania”.

Jeżeli czytelnik wybierze opcję samodzielnej modyfikacji tablicy partycji (rysunek 2.8), należy kliknąć przycisk *Forward*, zostanie wówczas wyświetlone kolejne okno dialogowe.

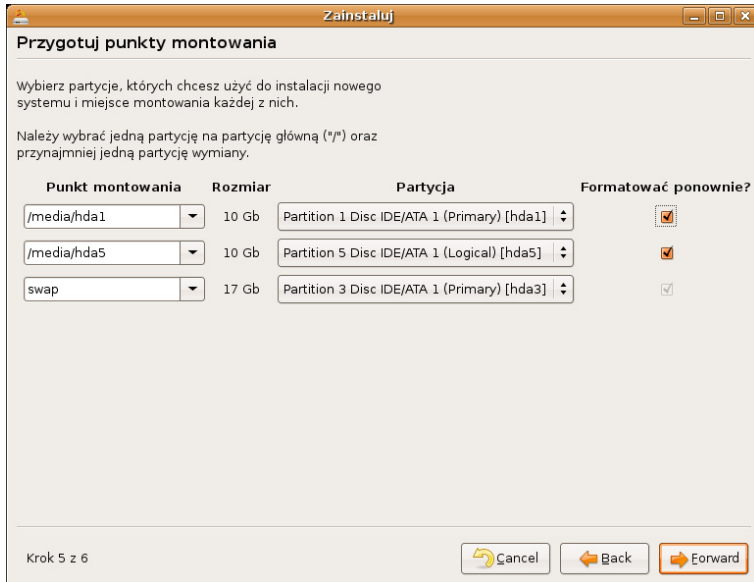


Samodzielne partycjonowanie dysku z użyciem Ubuntu

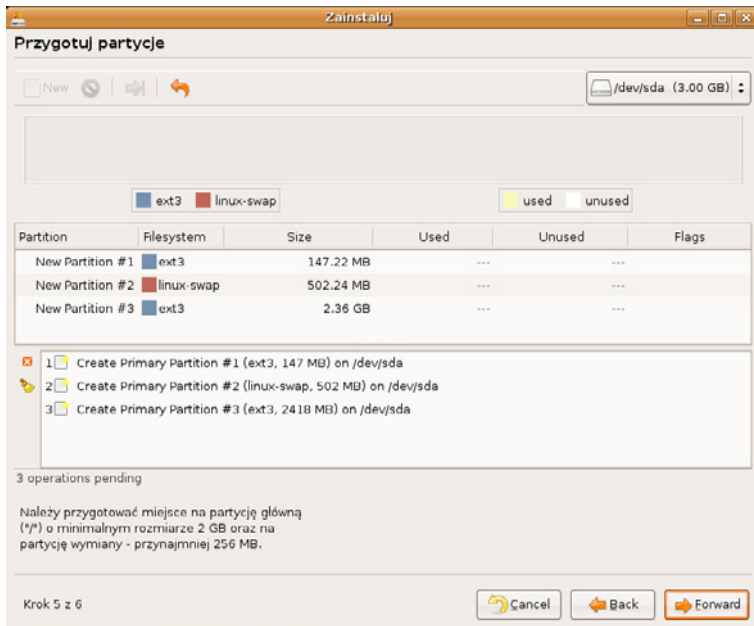
Jeżeli na dysku komputera znajdują się partycje, a użytkownik chciałby usunąć jedną lub więcej z nich, należy wybrać i podświetlić odpowiednią pozycję i kliknąć przycisk *Delete*. Trzeba zachować ostrożność, ponieważ wymazanie partycji spowoduje trwałe usunięcie wszelkich danych, które się na niej znajdowały. Następnie należy zaznaczyć nieprzydzieloną przestrzeń na dysku i kliknąć przycisk *New*, dzięki temu będzie można określić parametry nowej partycji. Zostanie wyświetlone następujące okno dialogowe.

Należy wprowadzić rozmiar partycji, która ma zostać utworzona, sprawdzić, czy wybrany został odpowiedni jej rodzaj (fizyczna lub rozszerzona) i format (zwykle ext3 lub swap), a następnie kliknąć przycisk *Add*, aby zaplanować jej utworzenie na dysku komputera. Powtórnie zostanie wyświetlone pierwsze okno dialogowe, tym razem z informacjami o partycjach, które właśnie zostały zdefiniowane przez użytkownika. Jak wynika z rysunku, wszelkie uprzednio zdefiniowane partycje nie są tworzone natychmiastowo, ich formatowanie będzie wykonane podczas kolejnego etapu instalacji. Również miejsce ich montowania będzie można wskazać w następnym etapie.

Powyższe czynności należy powtórzyć dla każdej partycji, która ma zostać utworzona. Trzeba utworzyć przynajmniej jedną partycję wymiany (*swap*), która będzie używana przez system do wspomaganie pamięci wirtualnej; jej wielkość jest wielokrotnością dostępnej w systemie pamięci operacyjnej.

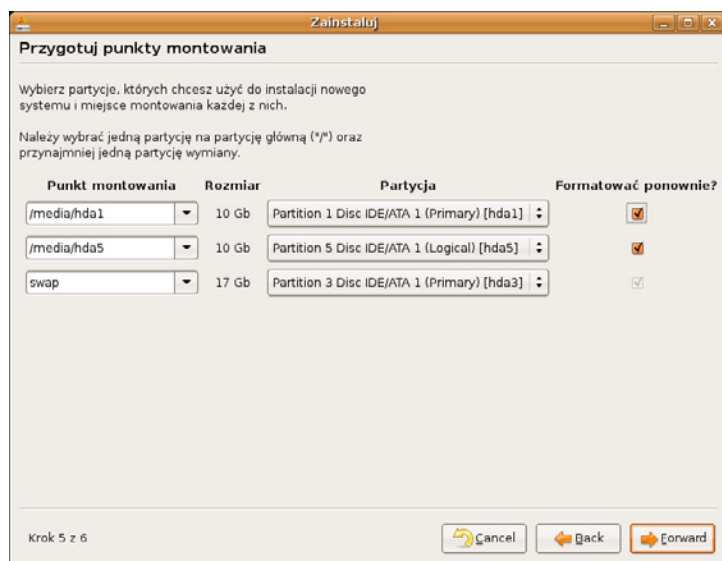


Określanie rozmiaru partycji podczas samodzielnego podziału dysku



Partycje, których utworzenie zostało zaplanowane do wykonania w kolejnym etapie instalacji systemu

Należy też utworzyć partycję dla głównego systemu plików, powinna ona mieć co najmniej 2 GB wielkości. Po zdefiniowaniu parametrów wszystkich partycji, które mają zostać utworzone, należy kliknąć przycisk *Forward*. Program partycjonujący wyświetli komunikat z prośbą o potwierdzenie zmian, które mają zostać wprowadzone. Aby rozpocząć tworzenie partycji, należy kliknąć przycisk *Apply*, kliknięcie przycisku *Cancel* spowoduje anulowanie procesu. Po kliknięciu przycisku *Apply* program utworzy określone wcześniej partycje i wyświetli kolejne okno dialogowe (patrz rysunek).

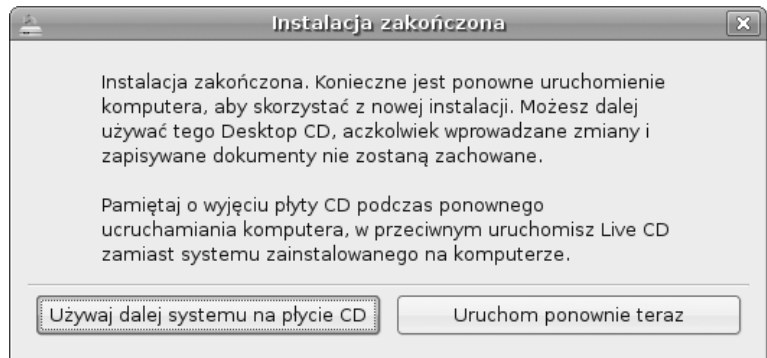


Przypisywanie punktów montowania podczas procesu samodzielnego partycjonowania dysku komputera

Powyższe okno dialogowe umożliwia użytkownikowi określenie, gdzie tworzone partycje zostaną zamontowane, dzięki czemu będą dostępne dla systemu operacyjnego. Za pomocą list rozwijanych *Mount Point* i *Partition* można się upewnić, czy wszystkie zaplanowane partycje zostały uwzględnione, czy nie zostanie sformatowana partycja zawierająca dane, które powinny zostać zachowane, oraz czy poprawnie zdefiniowano i wskazano punkt montowania partycji wymiany. Nim użytkownik potwierdzi wprowadzone ustawienia, należy je sprawdzić dwukrotnie, ponieważ tym razem kliknięcie przycisku *Forward* spowoduje faktyczne wprowadzenie zmian na dysku twardym. Kiedy użytkownik upewni się, że ustawienia są poprawne, należy kliknąć przycisk *Forward*.

8. Program instalacyjny wyświetli kilka okien informujących o postępie procesu partycjonowania dysku, instalacji Ubuntu, utworzeniu konta użytkownika i inne standardowe informacje konfiguracyjne. Po zakończeniu instalacji systemu zostanie wyświetlone okno przedstawione na rysunku 2.10. Aby uruchomić system, który właśnie został zainstalowany, należy kliknąć przycisk *Restart*, usunąć płytę z napędu CD-ROM (po tym, jak tacka zostanie wysunięta) i wcisnąć *Enter*, aby ponownie uruchomić komputer. Można też wybrać przycisk *Continue using the Live CD* i skorzystać z systemu uruchomionego z płyty CD, np. po to, aby zapisać utworzone wcześniej pliki. Informacje o tym, jak zapisać pliki utworzone podczas korzystania z systemu *live CD*, można znaleźć w podrozdziale niniejszego rozdziału zatytułowanym „Testowanie Ubuntu”.

Rysunek 2.10.
*Ostatnie okno
dialogowe programu
instalacyjnego*



Instalacja Ubuntu została zakończona — następnym razem system zostanie uruchomiony już z dysku komputera (po wyjęciu płyty CD). Gratulacje!

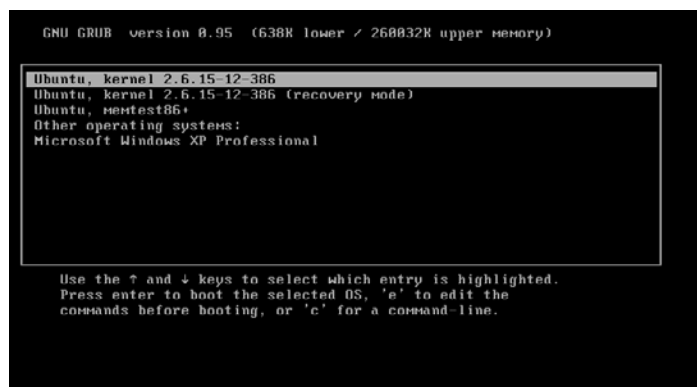
Uruchamianie Ubuntu

Jeżeli Ubuntu został zainstalowany jako jedyny system operacyjny lub na nowym komputerze, wówczas podczas wczytywania danych z dysku twardego użytkownik na monitorze zobaczy komunikat *GRUB loading, please wait... Press ESC to access the menu*. Uruchamianie Ubuntu rozpocznie się automatycznie po 5 sekundach, jeżeli w tym czasie użytkownik wciśnie jakikolwiek klawisz, wówczas zostanie wyświetlone menu programu GRUB, za pomocą którego będzie można wybrać wersję jądra, jaka ma zostać uruchomiona (jeżeli zainstalowanych jest ich więcej), uruchomić jądro systemu w innym trybie lub przetestować pamięć operacyjną. Więcej informacji na temat programu GRUB oraz jego opcji można znaleźć w punkcie zatytułowanym „Program rozruchowy”, w rozdziale 19.

Uruchamianie Ubuntu w trybie podwójnego rozruchu

Jeżeli Ubuntu został zainstalowany na komputerze, na którym był już inny system operacyjny, wówczas w menu programu GRUB będzie można dokonać wyboru, który z nich ma zostać uruchomiony. Domyślnie po upływie 30 sekund zostanie uruchomiony Ubuntu. Menu programu GRUB na komputerach z rodziny x86 można zobaczyć na rysunku 2.11.

Rysunek 2.11.
*Wybór systemu
operacyjnego w trybie
podwójnego rozruchu
na komputerach
o architekturze x86*



W komputerach typu PPC zostanie wyświetlony następujący komunikat system:

```
First Stage Ubuntu Bootstrap
```

```
Press: l for GNU/Linux,  
       x for MacOSX,  
       c for CDROM
```

```
Stage 1 Boot:
```

```
Welcome to yaboot version 1.3.13  
Enter help to get some basic help information
```

```
boot:
```

Dla komputerów PPC z systemem Mac OS X i opcją podwójnego rozruchu należy wpisać literę *L* i wcisnąć *Return*, aby uruchomić Ubuntu, lub wcisnąć *X*, aby uruchomić Mac OS X.

Pierwsze uruchomienie Ubuntu

Podczas pierwszego uruchomienia Ubuntu użytkownik może zaktualizować system. Dzięki temu można się upewnić, że w systemie zainstalowana jest najnowsza i najlepsza dostępna wersja oprogramowania łącznie z poprawkami bezpieczeństwa, uaktualnieniami i ulepszeniami. Więcej informacji na temat uaktualniania systemu można znaleźć w podrozdziale rozdziału 20., pt. „Wykorzystanie Menedżera aktualizacji”.

Testowanie Ubuntu

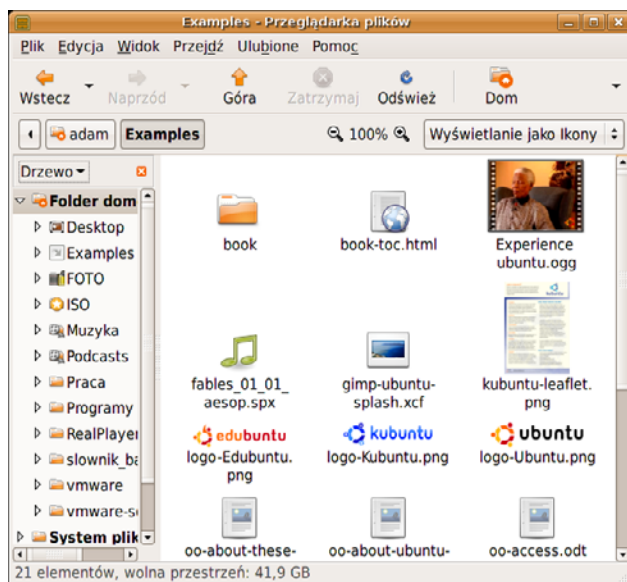
Na rysunku 2.2 przedstawiono standardowy pulpit Ubuntu, który wyświetlany jest po uruchomieniu systemu z płyty Ubuntu Desktop CD dołączonej do niniejszej książki. Z programami dostępnymi w Ubuntu Desktop CD można zapoznać się na dwa sposoby: po prostu kolejno je uruchomić i przetestować lub skorzystać z folderu znajdującego się na pulpicie, który zawiera dokumenty, pliki graficzne i multimedialne, dzięki którym można sprawdzić różne aplikacje. Ubuntu Desktop CD to kompletny system, a korzystając ze wskazówek zawartych w niniejszej książce, można go używać bez instalacji na dysku komputera. Niedogodnością takiego korzystania z systemu jest to, że wszelkie utworzone pliki czy konfiguracja oprogramowania zostaną utracone po ponownym uruchomieniu komputera.

W kolejnym podrozdziale omówiłem zawartość folderu *Examples*, który dostępny jest na pulpicie Ubuntu, oraz aplikacji obsługujących znajdujące się w nim pliki. Opisałem też wiele sposobów zachowania utworzonych danych, począwszy od zapisania ich w istniejącym systemie plików aż po użycie specjalnie sformatowanej pamięci USB, dzięki czemu ma się do dyspozycji prawdziwie przenośny system oparty o Ubuntu.

Zawartość folderu Examples

Ubuntu Desktop CD zawiera folder z przykładami dokumentów, plików graficznych itp., co ułatwia zapoznanie się z różnymi dostępnymi w systemie programami, za których pomocą można przeglądać, edytować i zapisywać wspomniane pliki. Po dwukrotnym kliknięciu folderu jego zawartość zostanie wyświetlona w przeglądarce plików Nautilus (patrz rysunek 2.12).

Rysunek 2.12.
Zawartość folderu
Examples



Oto dokumenty i inne pliki znajdujące się w folderze Examples.

- ♦ *Experience ubuntu.ogg*: przykładowy plik wideo zakodowany kodekiem o otwartym kodzie Theora w formacie Ogg. Podwójne kliknięcie tego pliku spowoduje wyświetlenie jego zawartości w multimedialnym odtwarzaczu Totem.
- ♦ *book-toc.html*: zawartość trzech przykładowych rozdziałów książki *Ubuntu. Oficjalny podręcznik*, które znajdują się podfolderze. Podwójne kliknięcie otworzy plik w przeglądarce internetowej Firefox.
- ♦ *fables_01_01aesop.spx*: przykładowy plik audio skompresowany kodekiem o otwartym kodzie w formacie Ogg. Podwójne kliknięcie tego pliku spowoduje odtworzenie jego zawartości w multimedialnym odtwarzaczu Totem.
- ♦ *gimp-ubuntu-splash*: przykładowy plik graficzny w otwartym formacie używanym w programie graficznym GIMP. Podwójne kliknięcie otworzy plik w programie graficznym GIMP.
- ♦ *kubuntu-leaflet.png*: przykładowy dokument o Kubuntu przygotowany w formacie PNG (ang. *Portable Network Graphics*). Podwójne kliknięcie otworzy plik w programie graficznym GIMP.
- ♦ *logo-Edubuntu.png*: przykładowe logo Edubuntu w formacie PNG. Podwójne kliknięcie otworzy plik w programie graficznym GIMP.

- ♦ *logo-Kubuntu.png*: przykładowe logo Kubuntu w formacie PNG. Podwójne kliknięcie otworzy plik w programie graficznym GIMP.
- ♦ *logo-Ubuntu.png*: przykładowe logo Ubuntu w formacie PNG. Podwójne kliknięcie otworzy plik w programie graficznym GIMP.
- ♦ *oo-about-these-files.odt*: przykładowy dokument pakietu OpenOffice.org zawierający informacje o plikach znajdujących się w folderze *Examples*. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org.
- ♦ *oo-about-ubuntu-ru.rtf*: przykładowy dokument zapisany cyrylicą w języku rosyjskim w formacie RTF (ang. *Rich Text Format*) Microsoftu. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org.
- ♦ *oo-access.odt*: przykładowy dokument pakietu OpenOffice.org zawierający informacje o dostępności i technologiach Ubuntu. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org.
- ♦ *oo-cd-cover.odg*: przykładowy plik OpenOffice.org Draw zawierający projekt okładek dla płyt z Ubuntu. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org.
- ♦ *oo-derivatives.doc*: przykładowy dokument w formacie pakietu Microsoft Office/Word opisujący dystrybucje pochodne Ubuntu: Edubuntu, Kubuntu i Xubuntu. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org.
- ♦ *oo-maxwell.odt*: przykładowy dokument pakietu OpenOffice.org ilustrujący możliwości edytora równań i omawiający równania Maxwella. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org.
- ♦ *oo-payment-schedule.ods*: przykładowy plik arkusza kalkulacyjnego OpenOffice.org ilustrujący obliczenia i inne możliwości. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org Calc.
- ♦ *oo-presenting-kubuntu.odp*: przykładowa prezentacja wykonana w pakiecie OpenOffice.org poświęcona Kubuntu. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org Impress.
- ♦ *oo-presenting-ubuntu.odp*: przykładowa prezentacja wykonana w pakiecie OpenOffice.org poświęcona Ubuntu. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org Impress.
- ♦ *oo-trig.xls*: przykładowy plik programu Excel pakietu Microsoft Office prezentujący obliczenia i graficzne przedstawienie wyników. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org Calc.
- ♦ *oo-welcome.odt*: przykładowy plik pakietu OpenOffice.org zawierający grafikę i wprowadzenie do Ubuntu. Podwójne kliknięcie spowoduje otwarcie pliku w programie OpenOffice.org.
- ♦ *ubuntu Sax.ogg*: przykładowy plik Vorbis (kodek o otwartych źródłach) w formacie Ogg. Podwójne kliknięcie tego pliku spowoduje wyświetlenie jego zawartości w multimedialnym odtwarzaczu Totem.

Podwójne kliknięcie któregośkolwiek z wymienionych plików daje możliwość zapoznania się z programami służącymi do ich obsługi. Tych plików nie można modyfikować, ponieważ znajdują się na płycie CD. Aby je edytować, należy skopiować je do odpowiedniego folderu i zmienić ich właściwości tak, aby można było je zapisywać. Więcej informacji na temat zmieniania atrybutów plików w Linuksie można znaleźć w podrozdziale rozdziału 4., zatytułowanym „Podstawowe informacje o uprawnieniach w systemach Linux”.

Uzyskiwanie dostępu do dysku twardego z systemu Desktop CD

Uruchamianie Ubuntu na komputerze z działającym systemem Windows lub Apple to przyjemność, ale w przypadku wykonania poważnej pracy może zaistnieć konieczność zapisania jej efektów, aby można było z nich skorzystać w przyszłości. Zarówno Microsoft Windows, jak i Macintosh używają specjalnych typów systemów plików do przechowywania danych użytkownika i systemu. Ponieważ Ubuntu Desktop CD jest uruchamiany w pamięci (na tzw. RAM dysku), aby zapisać dane utworzone podczas korzystania z systemu *live CD*, możliwe jest uzyskanie dostępu do istniejącego systemu plików z poziomu Ubuntu i zapisanie plików. Inną metodą jest użycie zewnętrznego nośnika pamięci i zapisanie danych wprost na nim, omówiono to w punkcie „Tryb persistence”. Aby uzyskać dostęp do systemu plików znajdującego się na dysku lokalnym, należy go udostępnić, czyli zamontować. Do tego celu potrzebne jest specjalne narzędzie.



Niestety, z powodu własnościowego charakteru systemów plików używanych w Microsoft Windows obecnie Ubuntu nie oferuje obsługi zapisywania danych we wszystkich systemach.

Poniżej omówiłem rodzaje systemów plików używane w Microsoft Windows i Apple Macintosh oraz opisałem, jak uzyskać do nich dostęp i kopiować pliki oraz dane utworzone w działającym Ubuntu Desktop CD na inny komputer dostępny w sieci w przypadku, gdy nie można z jakiegokolwiek powodu uzyskać dostępu do zasobów lokalnych.

Uzyskiwanie dostępu do partycji Microsoft Windows z systemu Desktop CD

W czasie, gdy pisałem niniejszą książkę, w większości systemów Microsoft Windows używano NTFS (*NT File System*) lub starszego FAT32 (*32-bit File Allocation Table*). NTFS jest własnościowym i złożonym formatem, linuksowe sterowniki obsługują go właściwie tylko w trybie do odczytu. W praktyce oznacza to, że można zamontować partycję NTFS w Linuksie i odczytywać z niej dane, ale nie da się jej zmodyfikować. Nawet wtedy, kiedy pliki z tej partycji zostaną skopiowane, aby można było na nich pracować, po zakończeniu edycji nie uda się ich skopiować z powrotem przy użyciu standardowych sterowników Linuksa dla NTFS.

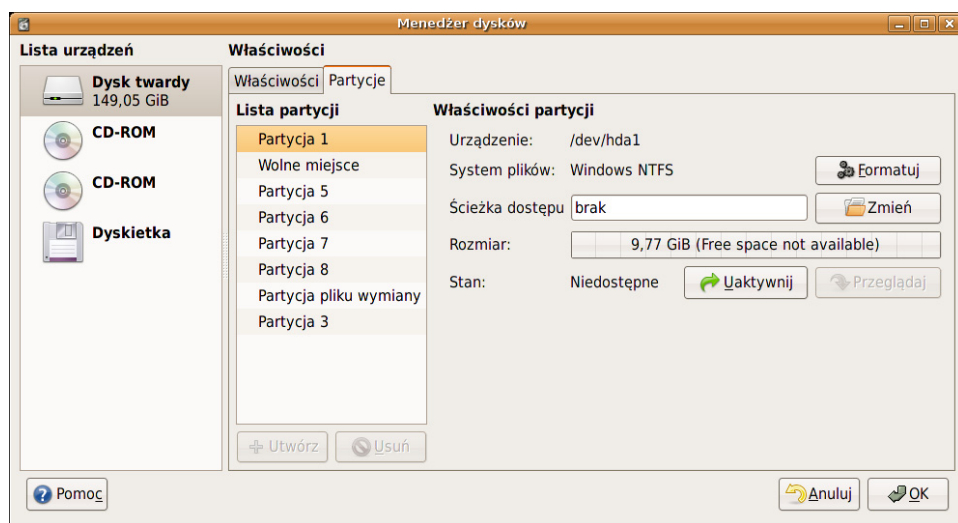


Chociaż standardowe sterowniki Linuksa w przyszłości zapewne będą obsługiwać zapis na partycjach NTFS, to dostępne są komercyjne sterowniki, które radzą sobie już z tym zadaniem, ich autorem jest firma Paragon Software (www.ntfs-linux.com). Firma Paragon dostarcza wersję demonstracyjną kodu źródłowego sterownika, którą można wkompiłować do jądra systemu (dla jąder vanilla dostarczanych z wieloma popularnymi dystrybucjami).

Sterowniki te są powolne, ale działają doskonale. Niestety, obecnie Desktop CD nie zawiera kompilatora (gcc) niezbędnego do zbudowania i zainstalowania pakietu. W przypadku konieczności zapisania zmian w plikach, w sytuacji gdy w komputerze, na którym uruchomiono Ubuntu, znajdują się tylko partycje NTFS Microsoft Windows, istnieje możliwość użycia nośnika pamięci w trybie *persistent* (szerzej objaśniono to w punkcie „Tryb *persistence*”), skopiowanie plików na zewnętrzny nośnik pamięci (jak objaśniono w punkcie „Uzyskiwanie dostępu do dysku twardego z systemu Desktop CD”) lub przeniesienie zmodyfikowanych plików (jak objaśniono w punkcie „Kopiowanie plików do innego komputera dostępnego w sieci”). Informacje na temat kompilowania, instalowania i używania specjalnych urządzeń i sterowników systemów plików w trakcie uruchamiania systemu można znaleźć w rozdziale 19., „Proces rozruchu i zamykania systemu Ubuntu”. Poniżej napisano, jak używać menedżera dysków do montowania istniejących partycji w trybie do odczytu i — jeśli to możliwe — również do zapisu.

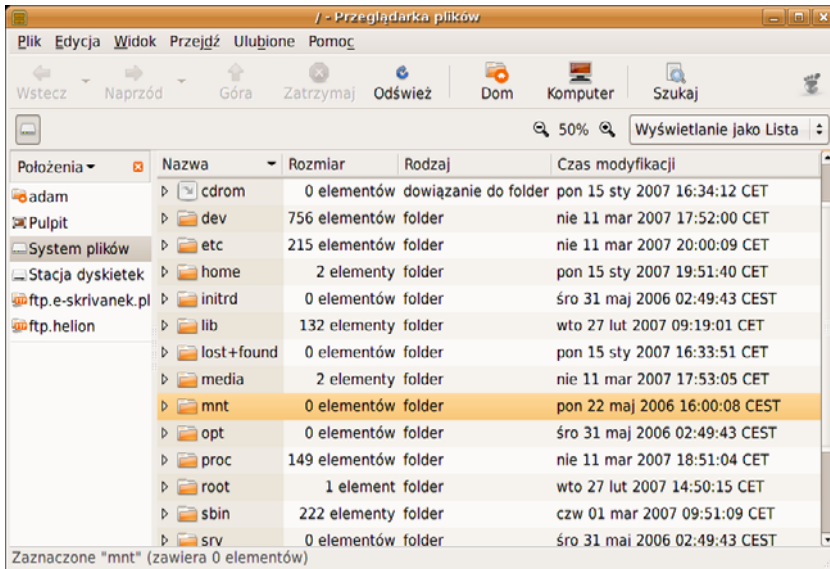
Po uruchomieniu systemu z płyty CD należy wybrać z menu *System/Administration/Disks*. Menedżer dysków wyświetli listę dostępnych dysków oraz innych urządzeń do przechowywania danych dostępnych w systemie. Aby zainstalować partycję dysku twardego, trzeba wykonać następujące czynności.

1. Należy wybrać odpowiednią partycję z listy znajdującej się po lewej stronie, a następnie kliknąć zakładkę *Partycje*, zostanie wyświetlona lista dostępnych partycji i dysków twardych. Tu trzeba wybrać partycję dysku, do której chcesz uzyskać dostęp. Zostanie wyświetlone okno, takie jak na rysunku 2.13.



Rysunek 2.13. Wybór partycji Windows w Disks Manager

2. Teraz trzeba kliknąć przycisk *Zmień* znajdujący się w części *Ścieżka dostępu*, a następnie wybrać punkt montowania wybranego systemu plików, co można zrobić, klikając *System plików* z listy znajdującej się po lewej stronie i wskazując właściwe miejsce. Ścieżka */mnt* to standardowa lokalizacja dla montowania partycji (patrz rysunek 2.14).



Rysunek 2.14. Wybór punktu montowania dla partycji systemu Windows

3. Aby wybrać wskazaną ścieżkę, należy kliknąć przycisk *Otwórz*. Okno dialogowe przedstawione na rysunku 2.14 umożliwia dokładne wskazanie punktu montowania. Aby partycja Windows stała się dostępna w */mnt*, trzeba ją zamontować, klikając przycisk *Uaktywnij*. W oknie dialogowym *Menedżer dysków* zostanie wyświetlona informacja o ilości wolnego miejsca dostępnego na zamontowanej partycji (patrz rysunek 2.15).

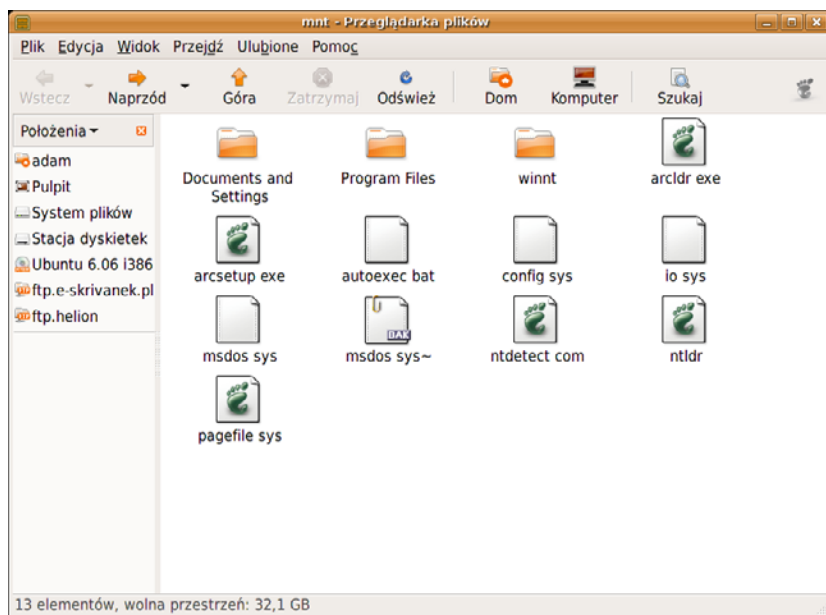


Rysunek 2.15. Informacje o zamontowanej partycji

4. Aby sprawdzić, czy uzyskano dostęp do plików na partycji Windows, trzeba kliknąć przycisk *Przeglądaj*. Zostanie uruchomiony menedżer plików Nautilus, w którym będzie można zobaczyć listę plików dostępnych na zamontowanej partycji (patrz rysunek 2.16).

Rysunek 2.16.

Przeglądanie
zamontowanej
partycji Windows



Po zamontowaniu partycji Windows można metodą „przeciągnij i upuść” przenosić pliki do działającego systemu lub pracować na nich wprost na zamontowanej partycji zwłaszcza wtedy, kiedy jest typu FAT32. Jeżeli w systemie są tylko partycje NTFS i użytkownik pracuje z tymczasowymi kopiami plików, wówczas można je zapisać na zewnętrznym nośniku pamięci, który jest dostępny z Ubuntu, można przenieść je na inny komputer dostępny w sieci (więcej na ten temat można znaleźć w punkcie „Kopiowanie plików do innego komputera dostępnego w sieci” znajdującym się dalej w tym rozdziale).

Uzyskiwanie dostępu do partycji PPC Macintosh z systemu Desktop CD

Na większości współczesnych komputerów typu PPC Macintosh używany jest system plików o nazwie HFS+, obecnie Disk Manager (o czym wspomniano wcześniej) nie obsługuje tego rodzaju partycji. Jest to niedogodność, ale nie oznacza ona, że nie uda się uzyskać dostępu do takich partycji. Jeżeli tylko użytkownik znajdzie w sobie trochę ochoty do wykonania magii w linii poleceń, wówczas uda się zamontować partycje systemu PPC Macintosh komputera, na którym został uruchomiony Ubuntu Desktop CD, a co za tym idzie, uzyskany zostanie dostęp do plików tam przechowywanych, będzie można je nawet modyfikować i zapisywać wprost na partycjach Apple.

Aby samodzielnie zamontować partycje HFS+, używając polecenia `mount`, należy wykonać następujące czynności.

1. Trzeba wybrać z menu *Applications/Accessories/Terminal*. Zostanie wyświetlone okno terminala.
2. Aby wyświetlić listę partycji dostępnych w systemie Macintosh, należy wpisać polecenie `fdisk -l /dev/hdc`, jak w poniższym przykładzie (uwzględniono tylko najważniejsze informacje):

```
sudo fdisk -l /dev/hdc
/dev/hdc
#                               type name      ...
/dev/hdc1  Apple_partition_map Apple      ...
/dev/hdc2  Apple_Bootstrap  untitled  ...
/dev/hdc3  Apple_HFS         Apple_HFS untitled  ...
/dev/hdc4  Apple_UNIX_SVR2  untitled  ...
/dev/hdc5  Apple_UNIX_SVR2  swap      ...
/dev/hdc6  Apple_Free          Extra      ...
```

1. Partycje PPC Macintosh są identyfikowane jako `Apple_HFS` widoczne w drugiej kolumnie. Można zamontować którąkolwiek z nich w katalogu `/mnt`, używając polecenia `mount`, jak w poniższym przykładzie, w którym zamontowano system plików HFS+ filesystem `/dev/hdc3` w katalogu `/mnt`:

```
sudo mount -t hfsplus /dev/hdc3 /mnt
```

Po zamontowaniu partycji HFS+ można przeglądać w katalogu `/mnt` dostępne pliki. Niestety, partycje mogą być zapisywane tylko przez użytkownika z odpowiednimi uprawnieniami, dlatego najlepiej będzie skopiować pliki i pracować na nich na pulpicie Desktop CD, a po zakończeniu edycji skopiować je za pomocą polecenia `sudo` z powrotem na partycję Macintosha. Przykładowo poniższe polecenie kopiuje pliki do katalogu domowego użytkownika `/mnt/Users/wvh`, który został zamontowany i udostępniony w Ubuntu:

```
sudo cp Desktop/the_file_I_modified.doc /mnt/Users/wvh
```

Użytkownicy, którzy skorzystają z powyższych wskazówek do montowania partycji, powinni pamiętać o tym, aby przed wyłączeniem Ubuntu Desktop CD je odmontować. Poniżej znajduje się przykładowe polecenie, za którego pomocą można odmontować partycję HFS+ zamontowaną wcześniej:

```
sudo umount /dev/hdc3
```

Tryb persistence

Płyty *live CD*, takie jak Ubuntu Desktop CD, oferują prosty sposób na łatwe czasowe uruchamianie Ubuntu na dowolnym komputerze. Jednakże większość takich systemów nie udostępnia możliwości zapisania specyficznych danych, takich jak konfiguracja systemu, programów itd. Można, co prawda, skorzystać z komputerów dostępnych w sieci czy innej partycji systemowej, ale ogranicza to użyteczność całego systemu do prostych prezentacji, jednorazowych eksperymentów, ratowania innych systemów i innych prób usunięcia awarii. Tak było do tej pory.

Ubuntu 6.06 Desktop CD został wyposażony w mechanizm służący do rozpoznawania w trakcie uruchamiania systemu zewnętrznych nośników pamięci opisanych specjalną nazwą, a znajdujące się na nich dane mogą teraz być wykorzystane do przywracania ustawień

systemu, plików konfiguracyjnych programów, a nawet zainstalowanych aplikacji. Ilość danych konfiguracyjnych oraz zainstalowanych programów jest ograniczona tylko pojemnością zewnętrznego nośnika pamięci. Oto wymagania, które należy spełnić, aby korzystać z zalet takiego rozwiązania.

- ♦ Zewnętrzny nośnik pamięci musi być rozpoznawany przez jądro systemu w trakcie jego uruchamiania.
- ♦ Na zewnętrznym nośniku pamięci musi być wykorzystywany system plików obsługiwany przez jądro systemu Desktop CD. Systemy, takie jak VFAT, NTFS i HFS+, nie będą działały, musi to być aktualny system plików Linuksa. Zalecanymi systemami dla przenośnego urządzenia są ext2 lub ext3.
- ♦ System plików na urządzeniu zewnętrznym musi mieć nazwę *casper-rw*.
- ♦ Podczas uruchamiania systemu należy dodać do jądra parametr *persistent*.

Urządzeniami wykorzystywanymi do przechowywania danych są najczęściej pamięci flash USB (znane również pod nazwą *pen drive'ów*) ze względu na swoją przenośność, choć z drugiej strony, zwykle dysponują one ograniczoną pojemnością. Jeżeli użytkownik zamierza instalować dodatkowe programy czy dane wymagające dużych ilości miejsca, lepszym rozwiązaniem będzie zewnętrzny dysk twardy podłączany za pośrednictwem portu USB.



Jeżeli Ubuntu Desktop CD jest używany do przeprowadzania prezentacji i w związku z tym nie ma potrzeby zapisywania danych, wówczas najlepszym rozwiązaniem może być utworzenie dostosowanej specjalnie do potrzeb własnej wersji Ubuntu. Na stronach projektu można znaleźć całkiem spory zestaw instrukcji, jak zbudować własną wersję Ubuntu Desktop CD na podstawie wydania 6.06 LTS: <https://help.ubuntu.com/community/LiveCDCustomization/6%2e06>.

Formatowanie zewnętrznego nośnika pamięci i nadawanie mu nazwy

Po podłączeniu pamięci do komputera należy uruchomić Ubuntu z dysku twardego lub napędu CD-ROM. W rozdziale 23., „Dodawanie sprzętu i przyłączanie urządzeń peryferyjnych”, można znaleźć informacje na temat identyfikowania i formatowania dysków w systemie. Jeżeli użytkownik korzysta z pamięci flash lub dysku USB, wówczas nazwa urządzenia powinna wyglądać mniej więcej tak: *sda*, *sdb* itd. Należy się upewnić, czy formatowany jest właściwy dysk! Jeżeli zewnętrzny nośnik pamięci posiada już system plików, należy go ręcznie zamontować (więcej informacji na ten temat znajduje się w punkcie rozdziału 4. zatytułowanym „Montowanie systemu plików”), dzięki temu będzie można się upewnić, że do formatowania wybrano właściwy dysk; przed rozpoczęciem formatowania trzeba odmontować system plików. Jak wspomniałem wcześniej, nowy system plików musi być jednym z aktualnie wykorzystywanych w Linuksie, czyli ext2, ext3, jfs, reiserfs lub xfs. Zalecam użycie systemu ext3, ponieważ jest on obsługiwany przez wszystkie dystrybucje Linuksa, co może być przydatne w sytuacji, gdy trzeba będzie wykonać kopię danych z dysku dostępną dla innego Linuksa. Z kolei system ext2 może być lepszym rozwiązaniem dla urządzeń o ograniczonej pojemności, takich jak *pendrive'y*.



Jeżeli użytkownik będzie chciał umieścić na dysku jakieś dane z innego Linuksa, może to zrobić bardzo łatwo: wystarczy podłączyć dysk do komputera, zamontować go i przekopiować dane do odpowiedniej lokalizacji. Zewnętrzne nośniki pamięci wykorzystywane do przechowywania danych w trybie *persistent* mają podobną do linuxowej strukturę katalogów. Aby np. skopiować do katalogu domowego domowego użytkownika *live CD*, wystarczy umieścić je w katalogu *home/ubuntu* na zewnętrznym nośniku pamięci, a podczas następnego uruchomienia systemu będą one widoczne w katalogu użytkownika (więcej na ten temat w dalszej części rozdziału).

Kiedy już na zewnętrznym nośniku pamięci zostanie umieszczony odpowiedni system plików, należy nadać mu odpowiednią nazwę (czyli *casper-rw*), która w Ubuntu 6.06 używana jest do rozpoznawania urządzeń działających w trybie *persistent*. Do nadawania nazwy systemom plików *ext2* lub *ext3* można użyć polecenia `e2label` w terminalu GNOME czy *xterm*. Dla innych systemów plików trzeba użyć odpowiedniego polecenia: *reiserfstune* dla *reiserfs*, *xfs_admin* dla *XFS* itd. I tak poprawne polecenie dla nadania nazwy systemowi plików *ext2* znajdującemu się na urządzeniu identyfikowanemu w systemie jako */dev/sda1* powinno wyglądać następująco:

```
$ sudo e2label /dev/sda1 casper-rw
```

Po nadaniu systemowi plików odpowiedniej nazwy można ponownie uruchomić komputer wraz z podłączonym urządzeniem pamięci zewnętrznej, co opisano poniżej.

Aktywacja trybu *persistent* podczas uruchamiania jądra systemu

Po utworzeniu systemu plików i nadaniu mu nazwy na zewnętrznym nośniku pamięci należy podłączyć go do odpowiedniego komputera, który następnie trzeba uruchomić, umieszczając w napędzie CD-ROM płytę Ubuntu Desktop CD. Po pojawieniu się ekranu startowego należy nacisnąć klawisz *F6*, następnie klawisz spacji i wpisać słowo *persistent* w wyświetlonej linii parametrów startowych jądra systemu (patrz rysunek 2.17).

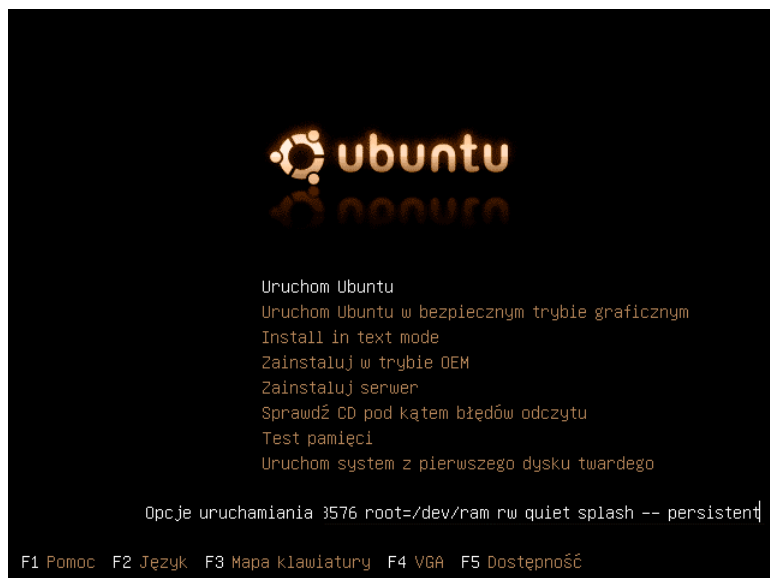
Po dodaniu odpowiedniego parametru należy wcisnąć przycisk *Enter*, aby kontynuować uruchamianie systemu z płyty CD. Kiedy system zostanie uruchomiony, można w nim dokonywać zmian konfiguracyjnych, tworzyć nowe pliki, instalować programy itd. Wszelkie wprowadzone zmiany zostaną zapisane na nośniku pamięci, dzięki czemu będą dostępne podczas kolejnego uruchomienia systemu z płyty *live CD*, warunkiem jest tylko wpisanie parametru startowego *persistent*.

Kopiowanie plików do innego komputera dostępnego w sieci

Użytkownik podczas korzystania z systemu Desktop CD może chcieć zapisać utworzone pliki lub dokonane w nich zmiany. Jeżeli system został uruchomiony z płyty CD i nie ma możliwości zapisania jakichkolwiek zmian na partycji systemowej (co opisałem wcześniej), jest inny sposób na ich zachowanie: wystarczy skopiować je na inny, dostępny w sieci komputer.

Rysunek 2.17.

Dodawanie parametru startowego do jądra systemu



Linux oferuje kilka różnych sposobów kopiowania plików z wykorzystaniem sieci. Oto kilka najpopularniejszych:

- ♦ protokół ftp (ang. *file transfer protocol*), umożliwiający połączenie się z innym komputerem i dwukierunkowy transfer plików,
- ♦ narzędzie scp (*secure copy*) umożliwiający transfer plików z jednego komputera na inny za pośrednictwem bezpiecznego, szyfrowanego połączenia,
- ♦ *System/Administration/Shared Folders* narzędzie umożliwiający zamontowanie folderów sieciowych udostępnionych na innych komputerach, z których i na które można kopiować pliki.

Wszystkie te metody zostały szczegółowo omówione w rozdziale 15., „Łączenie się z innymi systemami”. W tym miejscu wspominam o nich, aby czytelnik miał świadomość faktu, że istnieje kilka metod zapisywania danych utworzonych podczas pracy z systemem *live CD* nawet wtedy, kiedy nie ma — z różnych powodów — bezpośredniego dostępu do partycji Windows czy Apple Mac OS X.

Instalacja programów dla Windows umieszczonych na płycie CD

Ubuntu Desktop CD oferuje nie tylko możliwość wypróbowania Ubuntu, ale także kilku zaawansowanych programów o otwartym kodzie wprost w systemie Windows. Po umieszczeniu w napędzie płyty Ubuntu Desktop CD dla komputerów i386 w działającym systemie Microsoft Windows użytkownik zobaczy okno przedstawione na rysunku 2.18.

Rysunek 2.18.

*Oprogramowanie
o otwartych źródłach
znajdujące się
na Ubuntu Desktop
CD i przeznaczone
dla Windows*



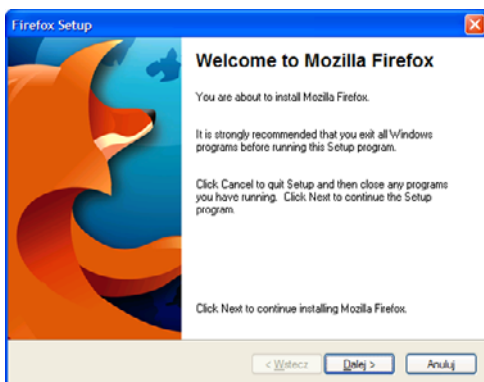
Oto aplikacje, które można zainstalować wprost w systemie Windows.

- ♦ Mozilla Firefox to popularna przeglądarka internetowa, będąca alternatywą Microsoft Internet Explorera w systemach Windows. Podstawowe możliwości programu można łatwo powiększyć, korzystając z setek rozszerzeń dostępnych bezpłatnie w internecie. Korzystanie z Firefoksa omówiono w rozdziale 9., „Surfowanie z Firefoksem”.
- ♦ Mozilla Thunderbird to program do obsługi poczty elektronicznej wyposażony w popularne i bezpieczne mechanizmy do pobierania wiadomości z serwerów pocztowych POP/POP3 i IMAP.
- ♦ AbiWord to popularny edytor tekstu będący częścią pakietu biurowego środowiska GNOME, posiada wolno dostępny kod i jest dobrą alternatywą dla kosztownych i zamkniętych programów, takich jak Microsoft Word.
- ♦ Gaim to komunikator internetowy o otwartym kodzie, za jego pośrednictwem można korzystać z takich sieci jak AOL, Yahoo, Gadu-Gadu itd.
- ♦ GIMP to popularny program do edycji grafiki, doskonały konkurent drogich programów o zamkniętym kodzie, takich jak Adobe Photoshop. Więcej informacji na temat używania GIMP-a można znaleźć w podrozdziale „Używanie programu GIMP”, w rozdziale 12.

Aby zainstalować któryś z powyższych programów, wystarczy kliknąć znajdującą się obok nich ikonę *Install*. Spowoduje to uruchomienie programu instalacyjnego dla systemu Windows; na rysunku 2.19 można zobaczyć początek procesu instalacji Firefoksa.

Po zakończeniu instalacji programy można uruchamiać wprost w systemie Windows bez konieczności umieszczania w napędzie płyty Ubuntu Desktop CD. Na rysunku 2.20 przedstawiam Firefoksa uruchomionego pod kontrolą systemu Windows.

Rysunek 2.19.
*Instalacja Firefoksa
 z płyty Desktop CD
 w systemie Windows*



Rysunek 2.20. *Zainstalowany i działający w Windows Firefox*

Podsumowanie

W niniejszym rozdziale wyjaśniłem, w jaki sposób można przetestować Ubuntu, jak zainstalować system na stałe po uruchomieniu płyty *live CD* odpowiedniej do rodzaju posiadanego komputera. Do książki dołączona jest płyta zawierająca Ubuntu Desktop CD dla komputerów typu x86. Mimo że mam nadzieję, iż Ubuntu znalazło się na komputerach czytelników na stałe, to jednak w rozdziale podałem informacje na temat różnych metod zapisywania danych uzyskanych w trakcie korzystania z systemu uruchamianego z płyty CD.

Jeżeli nawet czytelnik nie jest jeszcze gotowy do zupełnego przejścia na system Ubuntu, to płyta Desktop CD umożliwia wypróbowanie możliwości zarówno systemu operacyjnego o otwartym kodzie, jak i popularnych programów. W rozdziale opisałem, jak uzyskać dostęp do systemu plików na komputerze, na którym uruchomiono Ubuntu Desktop CD, a także jak zainstalować w systemie Windows wolne oprogramowanie zamieszczone na płycie, dzięki czemu można również na tej platformie sprawdzić możliwości programów o otwartym kodzie.

W następnym rozdziale omówię metody instalowania innych wersji Ubuntu, a także instalacje w nieco bardziej skomplikowanych konfiguracjach, np. z zachowaniem wcześniejszego systemu operacyjnego.

Rozdział 3.

Instalowanie Ubuntu jako systemu specjalnego zastosowania

W tym rozdziale:

- ♦ Konfigurowanie opcji podwójnego rozruchu w komputerach x86 i PPC
- ♦ Repartycjonowanie dysków twardych
- ♦ Pozyskiwanie innych wersji Ubuntu
- ♦ Instalacja Ubuntu w wersji serwerowej
- ♦ Rozwiązywanie problemów instalacyjnych
- ♦ Instalowanie systemu z płyty Alternate CD

W poprzednim rozdziale omówiono graficzny program instalacyjny będący częścią Ubuntu Desktop CD dołączonego do niniejszej książki. Korzystanie z niego jest standardowym sposobem instalowania systemu na komputerze. Jak jednak wspomniałem w rozdziale 1., w podrozdziale, „Płyty instalacyjne Ubuntu”, dostępne są jeszcze dwa dodatkowe rodzaje płyt przeznaczone dla innych obsługiwanych platform. Są to Server Install CD oraz Alternate Install CD. W niniejszym rozdziale opisuję, w jaki sposób można je zdobyć, jak przeprowadzić instalację systemu, korzystając z każdej z nich i jak skonfigurować podczas instalacji systemu opcję podwójnego rozruchu, dzięki czemu na jednym komputerze można uruchamiać więcej niż jeden system operacyjny (choć nie w tym samym czasie). Opcja podwójnego rozruch oznacza, że podczas uruchamiania systemu użytkownik może dokonać wyboru, na którym systemie operacyjnym chce pracować: na Ubuntu Linux czy innym, np. Microsoft Windows czy Mac OS X. To najłatwiejsze rozwiązanie w sytuacji, kiedy użytkownik nie dysponuje komputerem przeznaczonym specjalnie dla Ubuntu lub też kiedy korzysta z ważnego oprogramowania uruchamianego pod kontrolą innego systemu operacyjnego, dla którego nie ma pewności, że będzie tak samo działało w Ubuntu.

Mam nadzieję, że po lekturze tej książki czytelnikom będzie łatwiej rozstać się z Windows, ale zawsze lepiej się zabezpieczyć, niż żałować po fakcie, kiedy już niewiele da się zrobić.

Instalator graficzny znajdujący się na płycie Ubuntu Desktop CD różni się od pseudograficznego programu instalacyjnego, znanego z Debiana, który z kolei został wykorzystany w płytach Server i Alternate Install. Istnieje jednak kilka poważnych powodów jego użycia. Przyjemnie jest korzystać z fantazyjnego programu instalacyjnego, który można obsługiwać za pomocą myszy, ale warto pamiętać, że jest to aplikacja, której używa się zwykle raz dla danego systemu operacyjnego, a później odchodzi w zapomnienie. Dlatego taki program musi być solidny jak skała, łatwy do opanowania i użytkowania. W dodatku musi działać na każdym komputerze: od starych maszyn z kartami graficznymi wyświetlającymi obraz w niskich rozdzielczościach, aż po wyspecjalizowane zestawy komputerowe do gier oferujące rozdzielczości i efekty graficzne, których wiele osób nie jest w stanie nawet dostrzec. Serwery często pracują bez standardowych urządzeń wejścia/wyjścia (np. bez konsoli graficznej), dlatego szczególnie ważne jest, aby program działał dobrze również na tego typu sprzęcie.

Tryb podwójnego rozruchu — informacje

Jak wyjaśniłem we wstępie do rozdziału, **tryb podwójnego rozruchu** to termin używany do opisywania sytuacji, w której na jednym komputerze można uruchomić więcej niż jeden system operacyjny (choć nie w tym samym czasie). Podwójny rozruch to atrakcyjne rozwiązanie dla tych, którzy chcieliby eksperymentować z Ubuntu, ale muszą korzystać z aplikacji uruchamianych pod kontrolą innego systemu operacyjnego. Ponieważ Ubuntu jest przygotowane dla komputerów z rodziny x86 – 32 oraz 64-bitowych, na których najczęściej działa system Microsoft Windows, oraz dla PPC, na których zwykle uruchamiany jest Mac OS X, w niniejszym rozdziale opisałem, jak zainstalować Ubuntu jako drugi system operacyjny. Jedynym wymogiem, jaki trzeba spełnić, jest odpowiednia ilość wolnego miejsca na dysku, tak aby zmieściły się tam dwa systemy operacyjne.



Program instalacyjny Ubuntu omówiony w rozdziale 2. może automatycznie zmienić rozmiar istniejącej partycji systemu Microsoft Windows, dotyczy to systemów x86 i 64-bitowych. Jak wyjaśniłem poniżej, w większości przypadków nie trzeba samodzielnie repartycjonować dysku. Jednak na wszelki wypadek zamieszczam te informacje; być może kiedyś użytkownik zechce ręcznie zmienić układ partycji i wiadomości te okażą się przydatne. Jeżeli na dysku znajduje się wiele partycji Windows, a użytkownik chciałby je zmniejszyć, aby odzyskać trochę wolnego miejsca lub po prostu zrobić to „na wszelki wypadek”, poniżej znajdzie informacje na ten temat.

Proces uruchamiania komputera

Aby zrozumieć sposób działania trybu podwójnego rozruchu, warto wiedzieć, co dzieje się po włączeniu komputera, czyli jak przebiega sam **proces rozruchu**. Podczas uruchamiania systemu komputerowego program *Basic Input Output System* (czyli BIOS) ładuje, sprawdza i testuje zainstalowane komponenty maszyny, a następnie pobiera ustawienia konfiguracyjne zapisane w specjalnym module pamięci. W komputerach x86 ustawienia BIOS-u określają wiele parametrów konfiguracyjnych, m.in. kolejność przeszukiwania urządzeń dyskowych (dysku twardego, napędów CD-ROM i DVD czy innych nośników)

w poszukiwaniu czegoś, co można wykonać. W systemach Apple Macintosh za takie urządzenie automatycznie uznaje się partycję dysku, chyba, że użytkownik wskaże inny napęd, wciskając odpowiedni klawisz podczas uruchamiania komputera.

Kiedy system już wie, gdzie ma szukać, wówczas sprawdza wskazany napęd i wczytuje z niego główny sektor rozruchowy. Do tego momentu system nie ma żadnych informacji na temat sposobu, w jaki zorganizowany jest dysk twardy czy inny nośnik pamięci, ale zawsze potrafi odnaleźć kilka pierwszych bloków, które identyfikują dysk jako rozruchowy (lub nie). W Ubuntu MBR (ang. *Master Boot Record* — główny sektor rozruchowy) zawiera pierwszą część danych GRUB-a, czyli programu rozruchowego, która zostaje wczytana do pamięci i wykonana. Następnie zostaje wczytana druga część GRUB-a, która potrafi zidentyfikować, jaki rodzaj systemu plików znajduje się na dysku startowym.

Po wprowadzeniu danych z drugiej części GRUB-a do pamięci odczytywane są pliki konfiguracyjne znajdujące się na urządzeniu zawierającym program rozruchowy, a następnie wyświetlane są dostępne opcje uruchamiania komputera. W komputerach, na których działa tylko jeden system operacyjny, opcje rozruchowe nie są skomplikowane, wyświetlane są różne metody uruchomienia tego jednego systemu. Gdy dostępnych jest więcej systemów operacyjnych, wszystkie będą wyświetlone na liście. Oprócz możliwości wyboru różnych sposobów uruchamiania systemów, użytkownik może również wybierać pomiędzy nimi.

Wykonywanie sekwencji programów coraz bardziej skomplikowanych, począwszy od systemu BIOS czy monitora rozruchowego, skończywszy na pełnym systemie operacyjnym, to właśnie **uruchamianie**.



Więcej informacji na temat programów startowych używanych w Linuksie można znaleźć w podrozdziale zatytułowanym „Proces rozruchowy dystrybucji Ubuntu Linux”, w rozdziale 19.

Przygotowywanie systemu do uruchomienia trybu podwójnego rozruchu

Skonfigurowanie systemu tak, aby można było uruchamiać na nim więcej systemów operacyjnych, jest całkiem proste. Dzięki temu użytkownik będzie mógł uruchomić Microsoft Windows lub Mac OS X wówczas, gdy będzie musiał przeczytać wiadomość w aplikacji, która działa tylko w jednym z dwóch wymienionych systemów, natomiast Linuksa będzie mógł uruchomić wtedy, gdy będzie chciał wykonać bardziej złożone zadania lub przetestować wydajność i elegancję Ubuntu. Korzystanie z trybu podwójnego rozruchu jest też świetną metodą uczenia się Linuksa w sytuacji, kiedy użytkownik posiada tylko jeden komputer i nie ma ochoty lub umiejętności, aby wykonać od razu skok na głęboką wodę i całkowicie przejść na Ubuntu.

Podczas instalowania Mac OS X lub Microsoft Windows na komputerze z jednym dyskiem twardym systemy te zwykle tworzą też jedną partycję. To sprawia, że dodanie kolejnego systemu operacyjnego na tym samym komputerze staje się nieco bardziej skomplikowane. W takiej sytuacji użytkownik może:

- ♦ dodać kolejny dysk do komputera i zainstalować na nim Linuksa,
- ♦ zmienić rozmiar istniejącej partycji tak, aby odzyskać wolną przestrzeń, która zostanie wykorzystana do utworzenia dodatkowej partycji na zainstalowanie Linuksa.

Pierwsza opcja dotyczy zwykle komputerów biurowych i jest dostępna przy założeniu, że w komputerze jest miejsce na zainstalowanie dodatkowego dysku, a użytkownik dysponuje funduszami na jego zakup oraz umiejętnościami, aby zdjąć obudowę i poprawnie zamontować nowy podzespół. Jeżeli jednak na już działającym dysku jest wolna przestrzeń, wówczas druga opcja będzie tańsza, szybsza i prostsza. W niniejszym rozdziale omówiłem instalowanie Ubuntu w komputerze z działającym już systemem operacyjnym — instalacja w przypadku korzystania z dwóch dysków różni się tylko tym, że trzeba zachować ostrożność podczas wybierania odpowiedniego urządzenia.

Jak już wspomniałem wcześniej, za minimum potrzebne do umieszczenia Ubuntu należy przyjąć co najmniej 3 GB wolnej przestrzeni dyskowej, co umożliwi zainstalowanie systemu operacyjnego, standardowego zestawu operacji oraz zachowanie rozsądnej ilości miejsca na dane i pliki użytkownika. Po zakończeniu instalacji ze wspomnianych 3 GB nie zostanie wiele wolnego miejsca, dlatego każda większa ilość przeznaczona na Ubuntu oznaczać będzie więcej przestrzeni na tworzenie i przechowywanie plików użytkownika. Proces zmiany wielkości istniejących partycji na dysku nosi nazwę **repartycjonowania**.



Istnieje kilka darmowych oraz komercyjnych programów, które mogą ułatwić wprowadzenie zmian na istniejących partycjach dysku. Niestety, jak zawsze istnieje prawdopodobieństwo, że coś może się nie udać. Dlatego zawsze przed dokonywaniem jakichkolwiek zmian na dysku należy najpierw wykonać kopie zapasowe krytycznych danych w systemach Mac OS X i Windows!

W dziewięćdziesięciu dziewięciu przypadkach na sto repartycjonowanie jest zupełnie bezpieczne i nie powoduje żadnych strat w istniejących programach czy danych znajdujących się na dyskach systemów Windows czy Mac OS X. Jednak zapobieganie niedogodnościom spowodowanym utratą danych warto jest czasu, jaki należy poświęcić na sporządzenie kopii zapasowej ważnych plików przed wprowadzeniem jakichkolwiek zmian w układzie partycji. Jeżeli użytkownik jest kuszony wizją pominięcia tego kroku i przejścia wprost do repartycjonowania, warto się przez chwilę zastanowić, co znaczy utrata danych z komputera — przypadną wszystkie wiadomości pocztowe wymieniane z rodziną i przyjaciółmi, sporządzone pisma, fotografie cyfrowe, kolekcje muzyczne i pisana właśnie powieść — wszystko zniknie. Czy naprawdę warto tak ryzykować? Jeżeli czytelnik odpowie na to pytanie twierdząco, to oznacza, że jest odważniejszy od autora niniejszej książki.

Repartycjonowanie zamontowanego dysku

Po wykonaniu kopii zapasowej ważnych plików oraz upewnieniu się, że można ją odczytać, trzeba przejść do repartycjonowania dysku, co ma na celu uzyskanie przestrzeni potrzebnej do zainstalowania Linuksa. Pierwszym krokiem na drodze do osiągnięcia celu jest defragmentowanie dysku, co spowoduje umieszczenie plików i katalogów jak najbliżej siebie. Jeżeli ta część zawiera fragmenty plików, nie można jej tak po prostu odciać.

Oprogramowanie do wykonywania kopii zapasowych dla Microsoft Windows i Mac OS X

Chociaż nie jest to książka poświęcona Microsoft Windows czy Mac OS X, to jednak zalecenie wykonania kopii zapasowych tych systemów bez kilku informacji dodatkowych byłoby niegrzeczne. Zarówno systemy Windows, jak i Mac OS X są wyposażone w podstawowe narzędzia do wykonywania kopii bezpieczeństwa. Aby uruchomić taki program w Windows XP, należy wybrać kolejno z menu *Programy/Akcesoria/Narzędzia systemowe/Kopia zapasowa*. Podstawowy program do wykonywania kopii zapasowych dla maców dostępny jest za pośrednictwem subskrypcji internetowej (można zapisać się na bezpłatne testowanie programu na stronie .Mac, jeżeli użytkownik nie chce stałej subskrypcji). Obie wspomniane aplikacje mogą wykonywać kopie na przenośnych dyskach.

Dla obu platform dostępne są również bardziej skomplikowane i droższe rozwiązania. W przypadku systemów Windows dobre opinie można usłyszeć o programach SyncBackSE (www.2brightsparks.com) czy Genie-Soft Backup Manager (www.genie-soft.com); sam używam komercyjnego Veritas Backup Exec (www.backupexec.com), choć dla zwykłych użytkowników komputerów stacjonarnych i przenośnych może być zbyt zaawansowany. W przypadku systemów Mac OS X popularnością cieszy się Lacie Silverkeeper (www.lacie.com/silverkeeper), który jest darmowy, a do instalacji wymaga jedynie zarejestrowania się na stronie. Kolejna, niesłychanie popularna aplikacja to Bombich Software Carbon Copy Cloner (www.bombich.com/software/cccl.html).

Podczas korzystania z różnorodnych systemów dobrym rozwiązaniem może być wieloplatformowy EMC Insignia Retrospect (www.emcinsignia.com), umożliwiający wykonywanie kopii zapasowych w Windows, Mac OS X ale także w Linuksie, pod warunkiem zamontowania danej partycji w systemie Mac poprzez NFS lub SMB. Podobny, choć o nieco większych możliwościach, jest Tolis Group BRU (www.tolisgroup.com) działający w Windows, Mac OS X i Linuksie.

Defragmentacja w systemach Microsoft Windows

Systemy Windows posiadają wbudowane narzędzie do defragmentowania dysków, które można uruchomić, wybierając kolejno z menu *Programy/Akcesoria/Narzędzia systemowe/Defragmentator dysku*. W przypadku tego narzędzia (patrz rysunek 3.1) najpierw należy wybrać literę dysku przypisaną do partycji, która ma zostać zmniejszona, a następnie kliknąć przycisk *Defragmentuj*. Program wykona swoje zadanie i wyświetli obraz stopnia fragmentacji dysku przed i po całym procesie. Wystarczy jednokrotne uruchomienie defragmentatora, nawet, jeśli wyświetli on informację, że pozostały jeszcze jakieś pofragmentowane dane.

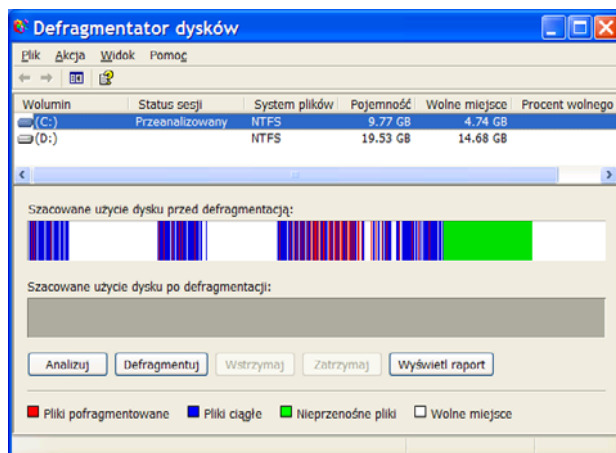
Być może czytelnik zastanawia się, dlaczego część plików widocznych na rysunku 3.1 nie została przeniesiona bliżej początku dysku. Defragmentator nie przenosi plików systemowych Microsoft Windows, takich jak plik stronicowania, ponieważ sposób ich użytkowania wyklucza modyfikowanie po tym, jak zostały utworzone. Mogą jednak zostać utworzone ponownie, kiedy zostaną wykryte jakieś problemy, dlatego jeżeli do utworzenia nowej partycji zostanie wykorzystana część dysku, na której się one znajdują, system utworzy je ponownie.



Linuksowe narzędzie, które zostanie użyte do repartycjonowania, jest przygotowane do radzenia sobie z fragmentacją i potrafi przenieść pliki z nowo tworzonej partycji na partycję Windows. Jednak defragmentacja dysku wcześniej na pewno nie zaszkodzi, dlatego zalecam wykonanie tej czynności przed kolejnymi etapami instalacji.

Rysunek 3.1.

Porządkowanie partycji dysku w systemie Windows za pomocą programu Defragmentator dysków



Po defragmentacji dysku następnym krokiem w procesie repartycjonowania jest samodzielne wydzielenie przestrzeni dyskowej za pomocą narzędzi znajdujących się na płycie Ubuntu Desktop CD. Użytkownicy Windows mogą teraz przejść wprost do podpunktu zatytułowanego „Repartycjonowanie systemu Microsoft Windows z użyciem płyty Ubuntu Desktop CD”. Korzystanie z Ubuntu Desktop CD zostało omówione w rozdziale 2., gdzie w punkcie „Uzyskiwanie dostępu do dysku twardego z systemu Desktop CD” skupiłem się na użyciu płyty live CD do dokonania prostych zmian w układzie partycji systemu Microsoft Windows.

Defragmentowanie dysku w systemach Mac OS X

Apple Disk Utility to doskonałe narzędzie do sprawdzania spójności partycji HFS i HFS+ w systemach Mac OS X. Niestety, za jego pomocą nie można zdefragmentować partycji; do tego celu czy w ogóle jakichkolwiek modyfikacji układu dysku potrzebny będzie program firm zewnętrznych.

System Mac OS X 10.3 (Panther) i późniejsze korzystające z systemu plików HFS+ z kroniką automatycznie defragmentują pliki mniejsze niż 20 MB, jeżeli zostaną w nich dokonane jakieś zmiany. W systemie tym i w przypadku partycji HFS+ jest też używany mechanizm znany jako *Hot-File-Adaptive-Clustering*, który przenosi często używane pliki o rozmiarze mniejszym niż 10 MB do lokalizacji o nazwie **hot band**, dzięki czemu uzyskuje się optymalną wydajność opartą o charakterystyki danego dysku. Pliki umieszczane w tym miejscu są defragmentowane w trakcie procesu przenoszenia. Choć jest to opcja, dzięki której można uzyskać wyraźny wzrost wydajności, korzystanie z *Hot-File-Adaptive-Clustering* komplikuje działanie narzędziom do defragmentacji.

Jednakowoż jest to kolejna sytuacja, w której lepiej najpierw się zabezpieczyć, niż później żałować. Dla maców pracujących na komputerach PPC narzędziem dyskowym (którego też używam) jest Micromats TechTool Pro (dostępne na stronie www.micromat.com), wykonuje ono defragmentację dysków podczas procesu optymalizacji.

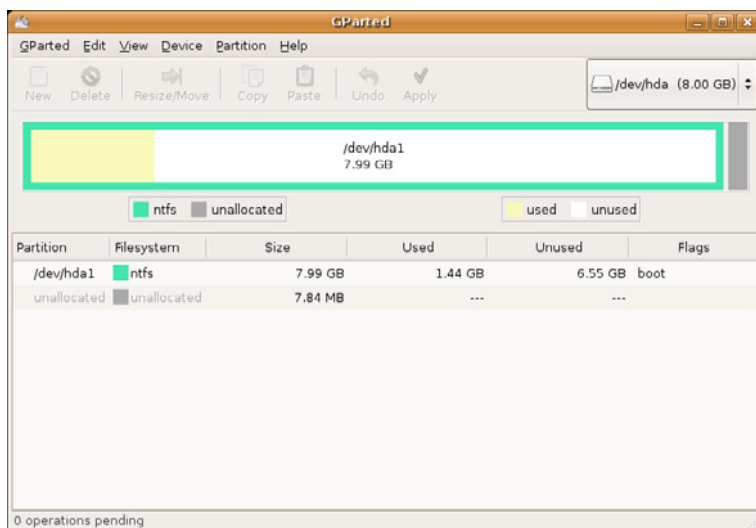
Repartycjonowanie systemu Microsoft Windows z użyciem płyty Ubuntu Desktop CD

Teraz rozpoczyna się zabawa! Ubuntu Desktop CD zawiera graficzne narzędzie o nazwie GNOME Partition Editor, które zmianę rozmiaru partycji Windows czyni zupełnie prostą. Obecnie w większości systemów Windows wykorzystuje się system plików NTFS, choć w użytku jest również 32-bitowy FAT32, znany również jako VFAT. GNOME Partition Editor potrafi zmienić rozmiar partycji wykorzystujących oba te systemy plików.

Aby skorzystać z GNOME Partition Editor w celu zmiany partycji systemu Microsoft Windows, należy uruchomić system z płyty Ubuntu Desktop CD zgodnie ze wskazówkami zamieszczonymi w rozdziale 2. Kiedy zostanie wyświetlony pulpit GNOME, należy z menu wybrać kolejno *System/Administration/GNOME Partition Editor*. Zostanie wyświetlone okno dialogowe przedstawione na rysunku 3.2.

Rysunek 3.2.

Ekran startowy programu GNOME Partition Editor

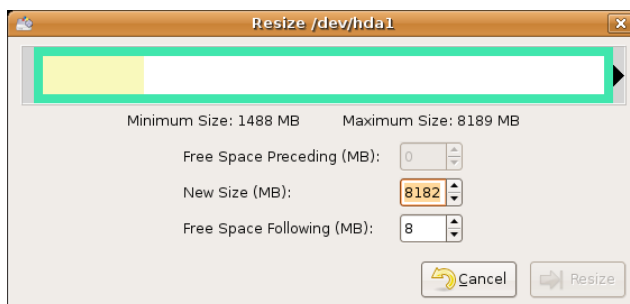


Od tego momentu trzeba kolejno wykonać następujące kroki.

1. Należy wybrać z listy partycję dysku twardego, która ma zostać zmniejszona, a następnie kliknąć przycisk *Resize/Move* znajdujący się na pasku narzędziowym powyżej listy dostępnych partycji. Okno to widoczne jest na rysunku 3.3.

Rysunek 3.3.

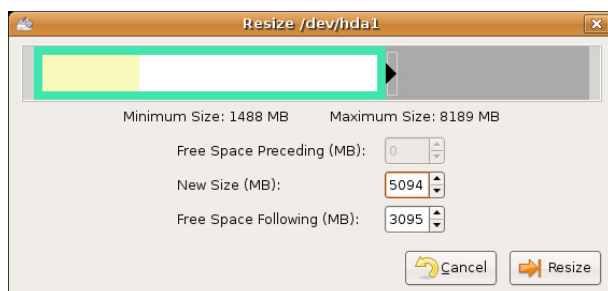
Okno dialogowe Resize



2. Teraz trzeba kliknąć lewym przyciskiem myszy strzałkę znajdującą się po prawej stronie, za jej pomocą można dopasować rozmiar partycji. Podczas przeciągania będzie zmieniała się wielkość wyrażona w megabajtach, co umożliwi dokładne jej dopasowanie. W tym momencie na dysku nie zostaną wprowadzone żadne zmiany, ustala się tylko ich zakres, dlatego jeżeli trzeba będzie je zmodyfikować, nie będzie to stanowiło żadnego problemu. Na rysunku 3.4 widoczne jest okno dialogowe wyświetlane po wprowadzeniu nowego rozmiaru partycji, która została teraz równomiernie podzielona na część przeznaczoną dla systemu plików NTFS i część, na której można zainstalować Ubuntu.

Rysunek 3.4.

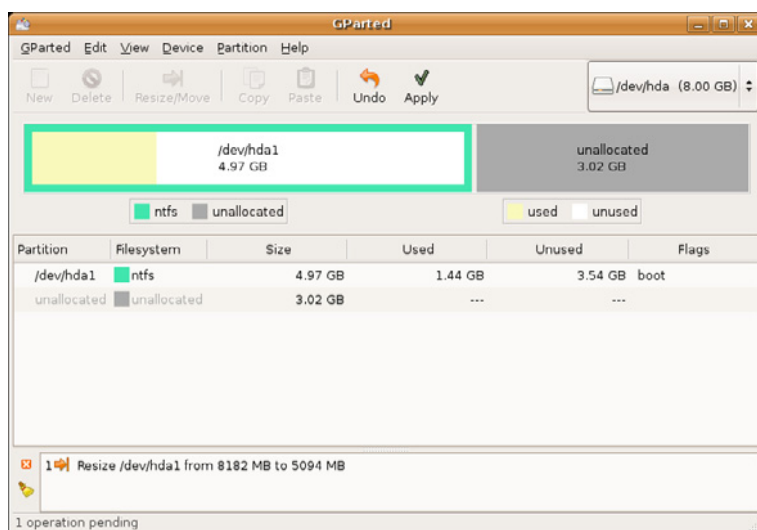
Okno dialogowe *Resize* po wprowadzeniu nowych ustawień partycji



3. Aby wprowadzić zmianę rozmiaru do harmonogramu zadań, należy kliknąć *Resize* i zamknąć okno dialogowe. Zostanie ponownie wyświetlone główne okno dialogowe edytora z listą dostępnych partycji w dole (patrz rysunek 3.5).

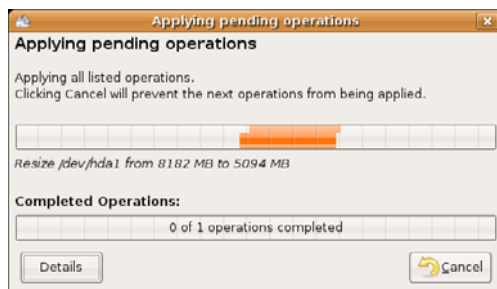
Rysunek 3.5.

Nowy podział dysku na partycje w programie GParted



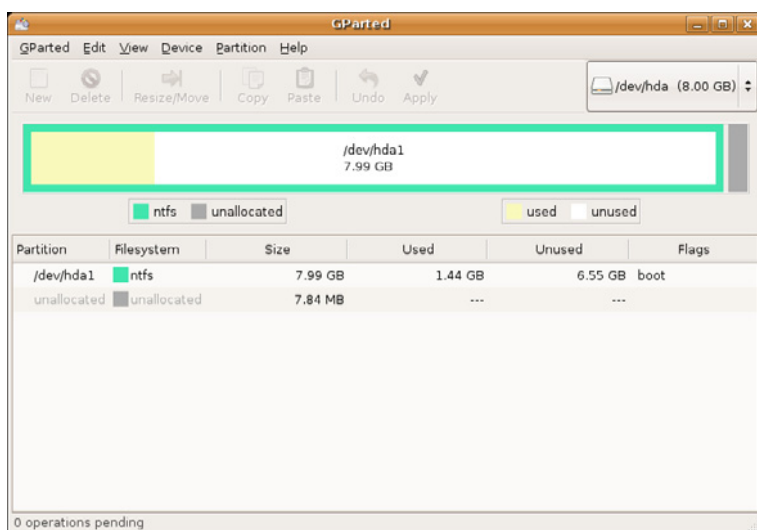
4. W tej chwili trzeba kliknąć przycisk *Apply*. Program poprosi o potwierdzenie wykonania zaplanowanych operacji. Na rysunku 3.6 widać animację potwierdzającą wykonywanie operacji zmiany wielkości partycji.
5. Kiedy operacja zmiany wielkości partycji zakończy się, zostanie wyświetlone okno dialogowe, w którym widoczne będą wprowadzone zmiany, czyli zmniejszona partycja NTFS oraz wolne miejsce, które można wykorzystać

Rysunek 3.6.
Operacja zmiany
wielkości partycji



na zainstalowanie Ubuntu (patrz rysunek 3.7). Aby wyłączyć program, należy wybrać z menu polecenie *GParted/Quit*, a następnie *System/Administration/Log Off*, co spowoduje zakończenie sesji live CD. W oknie dialogowym, które zostanie wyświetlone, trzeba wybrać opcję *Restart*.

Rysunek 3.7.
Okno dialogowe
programu GParted
po dokonaniu zmian
rozmiaru partycji



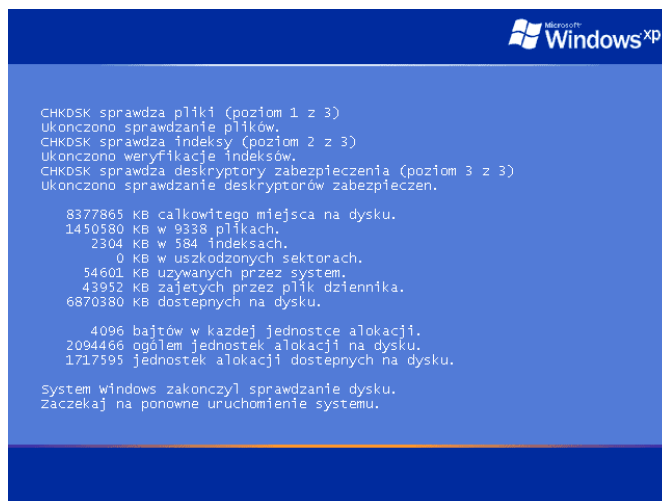
Gratulacje! Proces zmiany wielkości partycji zakończył się powodzeniem, a dla Ubuntu zostało przygotowane miejsce na dysku. Aby uruchomić system Microsoft Windows w celu sprawdzenia, czy wszystko działa właściwie, należy usunąć płytę CD z napędu i wcisnąć przycisk *Enter*. Po upewnieniu się, że Windows działa właściwie, można przejść do instalowania Ubuntu w wersji Desktop, Server lub Alternate Install CD.



W trakcie pierwszego uruchamiania po zmianie rozmiaru partycji system Windows może wyświetlić komunikat, że wykrył zmiany na dysku i uruchomi program CHKDSK, aby zaktualizować informacje o partycjach, a w razie potrzeby utworzy również plik stronicowania. Zostanie wyświetlony ekran widoczny na rysunku 3.8. Nie ma powodów do obaw, należy umożliwić systemowi przeprowadzenie tej operacji — zostanie sprawdzona partycja, dostosowane jej parametry, po czym Windows zostanie poprawnie uruchomiony.

Rysunek 3.8.

System Microsoft
Windows sprawdza
partycję, której
rozmiar został
zmieniony



Repartycjonowanie systemu PPC Macintosh przy użyciu płyty Ubuntu Desktop CD

Choć to może wydawać się dziwne, jednak graficzne narzędzie do repartycjonowania dysków nie potrafi poradzić sobie z systemami plików Macintosha HFS czy HFS+. Na szczęście, nie jest to powód do zmartwień, ponieważ Ubuntu Desktop CD zawiera narzędzie linii poleceń o nazwie *parted*, które łatwo poradzi sobie z partycjami HFS i HFS+. Choć może nie jest tak elegancko jak program omówiony powyżej, to można potraktować jego użycie jako wstęp do zgłębiania tajników potężnej konsoli — narzędzia tekstowego dostępnego w Linuksie. Zwykle użytkownik nie będzie musiał go używać, ale warto wiedzieć, że w razie potrzeby jest dostępne.

W systemie Mac OS X 10.2 wprowadzono do tradycyjnego systemu plików HFS **kronikowanie**. Jest to technika polepszająca przepustowość oraz skracająca czas uruchamiania systemu; polega ona na zapamiętywaniu zmian w plikach na specjalnej części partycji zwanej **kroniką** przed zapisaniem ich na dysku. Zestaw zmian zostaje zastosowany i usunięty z kroniki dopiero wówczas, gdy cała operacja zakończy się sukcesem. Podczas uruchamiania, system sprawdza kronikę każdego systemu plików i wprowadza dokonane zmiany, dzięki czemu można mieć pewność, że dane na partycji są aktualne; minimalizuje to także prawdopodobieństwo niespójności, która może być spowodowana nie w pełni wprowadzonymi zmianami.

Przed zmianą rozmiaru partycji HFS+ należy wyłączyć kronikowanie, żeby mieć pewność, że dane są aktualne; dzięki temu nie będzie żadnych zmian czekających na wprowadzenie na dysk. Należy to zrobić, kiedy działa system Mac OS X. Aby tego dokonać, trzeba uruchomić terminal (menu *Applications/Terminal*) i w jego oknie wykonać następujące polecenie:

```
$ diskutil list
/dev/disk0
#:
```

	type name	size	identifier
--	-----------	------	------------

Problemy z korzystaniem z graficznych programów do repartycjonowania?

W niektórych przypadkach omówiony powyżej graficzny program do repartycjonowania dysków może nie zadziałać na systemach plików Windows. Choć to uciążliwe, da się temu zaradzić; za pomocą polecenia `parted` można samodzielnie dokonać zmian w układzie partycji, szczegóły podałem w części dotyczącej repartycjonowania w systemach Apple. Różnić się będą jedynie nazwy i numery partycji. Poniżej znajduje się zapis procesu samodzielnego zmieniania rozmiaru partycji Windows za pomocą polecenia `parted` — szczegółowe instrukcje można znaleźć w następnej części, wystarczy tylko w miejsce słów `PPC Macintosh`, `Apple Macintosh`, `Mac OS X` itp. wstawić `Microsoft Windows`.

```
(parted) p
Disk geometry for /dev/hda: 0kB - 8590MB
Disk label type: msdos
Number  Start  End      Size     Type     File system  Flags
1       32kB   8579MB  8579MB  primary  fat32        boot, lba
(parted) resize 1 32kB 4096MB
(parted) p
Disk geometry for /dev/hda: 0kB - 8590MB
Disk label type: msdos
Number  Start  End      Size     Type     File system  Flags
1       32kB   4096MB  4096MB  primary  fat32        boot, lba
(parted) quit
```

Część piękna i potęgę Linuksa to fakt, że zawsze jest kilka sposobów, jakimi można osiągnąć ten sam cel. Zwykle różnica pomiędzy nimi polega na tym, czy użytkownik skorzysta z programu z graficznym interfejsem, czy też z linii poleceń.

0:	Apple_partition_scheme	*9.6 GB	disk0
1:	Apple_partition_map	31.5 KB	disk0s1
2:	Apple_HFS Boot	9.4 GB	disk0s3

Jak można odczytać z danych wyjściowych, aktywna partycja systemu Mac OS X to trzecia partycja na urządzeniu `/dev/disk0`, która jest identyfikowana jako partycja `/dev/disk0s3` (trzecia część dysku 0). Aby wyłączyć kronikowanie w systemie plików, należy wykonać następujące polecenie:

```
$ diskutil disableJournal /dev/disk0s3
Journaling has been disabled on /dev/disk0s3
```

Jak można odczytać z danych wyjściowych, kronikowanie dla tej partycji zostało wyłączone. W napędzie CD-ROM należy umieścić płytę instalacyjną Ubuntu przeznaczoną dla komputerów PPC, a następnie wyłączyć system Mac OS X i ponownie uruchomić komputer oraz włączyć Linux zgodnie ze wskazówkami zamieszczonymi w rozdziale 2.

Teraz rozpoczyna się prawdziwa zabawa. Korzystając z menu *Applications/Accessories/Terminal*, należy uruchomić terminal. W oknie terminala trzeba wykonać następujące czynności.

1. Najpierw wykonać polecenie `parted`, podając nazwę dysku, który ma zostać poddany ponownemu partycjonowaniu. Dla większości komputerów Mac nazwa dysku twardego będzie wyglądała następująco: `/dev/hdc`. Dane wyjściowe, które zostaną wyświetlone, powinny przypominać dane zmieszczone poniżej:

```
# parted /dev/hdc
GNU Parted 1.6.21 with HFS shrink patch 16
Copyright (C) 1998 - 2004 Free Software Foundation, Inc.
This program is free software, covered by the GNU General Public License.
This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY
or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License
for more details.
```

```
Using /dev/hdc
(parted)
```

2. Następnie należy wpisać polecenie `p`; (*print* — drukuj), co spowoduje wyświetlenie informacji na temat danej partycji dysku, co powinno wyglądać mniej więcej tak:

```
(parted) p
Disk geometry for /dev/hdc: 0.000-9787.148 megabytes
Disk label type: mac
Minor   Start      End        Filesystem  Name              Flags
1        0.000      0.031      Apple       Apple              (boot)
3       128.031   9787.140   hfs+        Apple_HFS_Untitled_1
```

Use the `resize` command to reduce the size of the HFS+ partition.

3. Tej komendy można użyć do zmniejszenia rozmiaru partycji HFS+. Dla tego polecenia należy podać trzy parametry: numer partycji, która ma być zmniejszona; blok początkowy partycji oraz nowy blok końcowy. Aby zmniejszyć rozmiar partycji HFS+ do ok. 4 GB, należy wykonać następujące polecenie:

```
(parted) resize 3 128.031 5600
```

4. Ponieważ obsługa systemu plików HFS+ w programie `parted` nadal jest w fazie testowej, użytkownik może zobaczyć następujący komunikat:

```
Warning: You have an HFS+ file system that has a feature that I haven't seen
used anywhere. Parted can theoretically handle it, but the corresponding code
has never been tested, so this might be risky. Please e-mail me so I can see how
it works! <xilun666@libertysurf.fr>
Ignore/Cancel?
```

5. Teraz należy wpisać literę `i` i wcisnąć klawisz *Enter*, rozpocznie się proces zmiany rozmiaru partycji. Na ekranie będzie widoczna informacja:

```
shrinking... NN% (time left MM:SS)
```

6. Po zakończeniu procesu w programie `parted` zostanie wyświetlony znak zachęty. Aby zakończyć działanie programu, należy wprowadzić polecenie `quit` i wcisnąć klawisz *Enter*.

Żeby uruchomić system Mac OS X w celu sprawdzenia, czy partycja HFS+ działa właściwie, należy usunąć płytę CD z napędu i wcisnąć przycisk *Enter*. Po upewnieniu się, że Mac OS X działa prawidłowo, można przejść do instalowania Ubuntu w wersji Desktop, Server lub Alternate Install CD.

Pozyskiwanie innych płyt instalacyjnych

Ubuntu nie można kupić w sklepach, ale nie jest to przeszkodą, ponieważ płyta CD, której używamy do testowania systemu czy instalacji, jest dołączona do niniejszej książki, a pozostałe wersje instalacyjne są dostępne w internecie. Jeżeli użytkownik posiada łącze szerokopasmowe oraz nagrywarkę CD, najprostszym sposobem na zdobycie Buntu, w wersji innej niż dołączona do książki, będzie pobranie z internetu odpowiedniego pliku ISO. Obrazy ISO to pliki zawierające „zdjęcie” zawartości płyt CD lub DVD w standardzie zatwierdzonym przez International Standards Organization pod numerem 9660. Więcej informacji na temat rodzajów dostępnych płyt innych niż dołączona do książki można znaleźć w rozdziale 1., w podrozdziale „Płyty instalacyjne Ubuntu”. Kiedy już użytkownik zdecyduje, która wersja Ubuntu odpowiada jego potrzebom, wystarczy udać się na stronę www.ubuntu.com/download, wybrać odpowiedni serwer lustrzany znajdujący się w kraju położonym najbliżej, wybrać wersję instalacyjną odpowiednią do posiadanej platformy sprzętowej i rodzaj systemu, jaki chce się zainstalować. Następnie należy zapisać plik na dysku, nagrać go na płytę CD i gotowe!

Powolne łącze internetowe lub brak nagrywarki CD nie są powodami do paniki. Jak wspomniano wcześniej, ludzie z Ubuntu dostarczą odpowiednią płytę CD, choć może potrwać to kilka tygodni, ponieważ będzie wysyłana z Holandii. Nie jest to najlepsze rozwiązanie dla tych, którzy potrzebują natychmiastowego rozwiązania, ale czasem może okazać się jedyne. Aby złożyć zamówienie na płytę z Ubuntu, należy przejść na stronę <https://shipit.ubuntu.com>, utworzyć konto, podając swój adres oraz hasło, i zamówić jeden lub więcej nośników z odpowiednim wydaniem. Płyty zostaną dostarczone w atrakcyjnym, podwójnym opakowaniu zawierającym zamówioną wersję instalacyjną oraz wersję Desktop CD dla każdego wybranego typu systemu. Można jednocześnie zamówić darmowe płyty dla różnych rodzajów komputerów.

Płyty w wersji Server i Alternate Install CD — uruchamianie systemu

Uruchamianie systemu Ubuntu Server czy Alternate Install CD jest łatwe: wystarczy umieścić odpowiednią płytę w napędzie, uruchomić ponownie komputer, ustawiając go jednocześnie tak, aby uruchamiał system z płyty CD, a nie z partycji dysku twardego. W tradycyjnych systemach PC o kolejności wczytywania systemu operacyjnego decyduje BIOS, podczas gdy w systemach PPC ustawienia te można zmienić, przytrzymując wciśnięty klawisz *C* podczas uruchamiania komputera.

Kiedy system zostanie uruchomiony z płyty Ubuntu Server lub Alternate Install CD, konieczne będzie wprowadzenie kilku podstawowych informacji służących do poprawnego skonfigurowania systemu, instalacji plików lokalizacyjnych oraz aktywowania konta użytkownika. W kolejnych dwóch częściach rozdziału omówiłem opcje instalacyjne wersji Server i Alternate, w których wykorzystano ten sam pseudograficzny program instalacyjny omówiony we wstępie do niniejszego rozdziału oraz zamieszczono wskazówki, jak przebrnąć przez proces instalacji.



Jeżeli komputer typu x86 uruchomił system operacyjny z partycji dysku, mimo że w napędzie CD-ROM znajdowała się właściwa płyta Ubuntu CD, wówczas należy zmienić w systemie BIOS kolejność uruchamiania urządzeń. Kolejność rozruchu to układ, w jakim dostępne urządzenia są po włączeniu komputera sprawdzane w poszukiwaniu systemu operacyjnego. Aby wprowadzić zmiany, należy włączyć komputer i wcisnąć klawisz, który pozwoli uzyskać dostęp do systemu BIOS. Zwykle jest to klawisz *Del* (*Delete*) lub *F2*; informacja o tym, który to klawisz, wyświetlana jest w dolnym lewym rogu ekranu podczas uruchamiania komputera.

W zależności od typu systemu BIOS zainstalowanego w komputerze, kolejność rozruchu zwykle można zmienić w menu *Advanced Settings* lub *Boot*, do których można przejść za pomocą klawiszy ze strzałkami umieszczonych na klawiaturze. Po podświetleniu odpowiedniej pozycji należy wcisnąć klawisz *Enter*, aby przejść do szczegółowego menu. Następnie za pomocą klawiszy ze strzałkami trzeba przejść do pozycji *First Boot Device* lub *CD Drive* i sprawdzić za pomocą wskazówek znajdujących się po prawej stronie ekranu, w jaki sposób ustawić napęd CD jako pierwsze urządzenie startowe. Po dokonaniu zmiany należy wcisnąć klawisz *Escape*, aby wyjść z menu, a następnie *F10*, dzięki czemu nowe ustawienia zostaną zapisane. Teraz wystarczy ponownie uruchomić komputer.



W programie instalacyjnym jest błąd, który powoduje, że Ubuntu instalowane z niektórych płyt *Server* i *Alternate* może nie być odpowiednio zabezpieczone, jeżeli użytkownik w czasie instalacji w ostatnim ekranie wybierze opcję *Go Back* zamiast *Continue* (patrz rysunek 3.25). Jej wybranie powoduje, że nie zostanie ustawione hasło głównego użytkownika, co oznacza, że każdy może się zalogować do systemu właśnie jako użytkownik *root* lub użyć polecenia *su* do zdobycia jego uprawnień. Jeżeli użytkownik będzie chciał powrócić do wcześniejszego etapu instalacji, wówczas lepiej samodzielnie ustawić lub wyłączyć hasło dla konta użytkownika z uprawnieniami administracyjnymi. Być może po wydaniu tej książki błąd ten został już usunięty, ale lepiej się zabezpieczyć, niż później żałować.

Menu instalacyjne płyty *Server Install CD*

Jak łatwo można zgadnąć, płyta *Server Install CD* służy do instalowania systemu przeznaczonego do działania jako serwer. Ponieważ serwery często są umieszczane w obudowach montowanych w stojakach i nie podłącza się do nich monitora, dlatego w Ubuntu *Server* nie umieszczono graficznego interfejsu, takiego jak system *X Window* czy *GNOME*. Brak tych pakietów w ustawieniach domyślnych zmniejsza ilość przestrzeni dyskowej potrzebnej na instalację bazową oraz redukuje ilość oprogramowania, które trzeba aktualizować.



Jeżeli użytkownik zechce, zawsze może później dodać graficzny interfejs do systemu bazowego, dzięki czemu będzie można skorzystać z dodatkowych narzędzi czy poprawić wygodę użytkowania. I tak, aby dodać środowisko *Xfce*, należy zainstalować pakiet *xubuntu-desktop*; żeby dodać kompletne środowisko *GNOME*, trzeba zainstalować pakiet *ubuntu-desktop*; pakiet *kubuntu-desktop* potrzebny jest, by korzystać z *KDE desktop*, a jeżeli użytkownik będzie chciał samodzielnie zdecydować i określić narzędzia graficzne, wówczas powinien zainstalować pakiet *xserver-xorg*. Dodawanie pakietów oprogramowania omówiono szczegółowo w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Na rysunku 3.9 widoczne jest menu startowe wyświetlane po uruchomieniu systemu z płyty CD (aby zastosować język polski, należy wcisnąć klawisz *F2* i wybrać język z listy).

Rysunek 3.9.

Menu instalacyjne
płyty Server Install CD



Oto dostępne opcje menu instalacyjnego płyty Server Install CD.

- ♦ **Zainstaluj na twardym dysku:** instalacja podstawowego serwera Ubuntu na dysku komputera za pomocą pseudograficznego programu instalacyjnego. Proces ten omówiłem szczegółowo w dalszym punkcie pt. „Instalacja serwera Ubuntu”.
- ♦ **Zainstaluj serwer LAMP:** instalacja na dysku komputera podstawowego serwera Ubuntu LAMP (Linux, Apache, MySQL i Perl) za pomocą pseudograficznego programu instalacyjnego. Inaczej niż w poprzednim przypadku zostaną zainstalowane wszystkie pakiety niezbędne do uruchomienia serwera LAMP. W procesie instalacyjnym serwera LAMP używany jest ten sam program i wyświetlane są te same ekrany instalacyjne, które opisałem poniżej w punkcie „Instalacja serwera Ubuntu”. Dodatkową konfigurację omówiłem w punkcie zatytułowanym „Instalowanie serwera LAMP”.
- ♦ **Sprawdź CD pod kątem błędów odczytu:** testowana jest spójność plików na płycie oraz ich zawartość.
- ♦ **Napraw uszkodzony system:** wyświetlonych zostaje kilka informacji, takich jak język, w którym mają być wyświetlane komunikaty, położenie geograficzne, a następnie z płyty uruchomiony zostaje system, dzięki któremu można naprawić uszkodzone dyski, poprawić lub zainstalować ponownie program startowy GRUB itp.



W razie wystąpienia problemów z dostępem do trybu ratunkowego można uruchomić system z dołączonej do książki płyty; oferuje ona takie same możliwości usunięcia usterek, dodając świetny interfejs graficzny.

- ♦ **Test pamięci:** wykonane zostają testy sprawdzające pamięć systemową oraz podręczną. Po wciśnięciu klawisza z literą *C* zostanie wyświetlone menu konfiguracyjne umożliwiające uruchomienie wybranych testów. Po zakończeniu testów pamięci należy wcisnąć klawisz *Escape*, co spowoduje ponowne uruchomienie systemu i powrót do menu instalacyjnego.

- ♦ **Uruchom system z pierwszego dysku twardego:** system nie jest uruchamiany z płyty, ale z dysku komputera. Opcja ta jest przydatna w sytuacji, kiedy płyta Ubuntu Server Install CD przypadkowo została pozostawiona w napędzie, choć użytkownik nie planował uruchamiania sesji live CD.

Wybranie opcji instalacji na dysku lub instalacji systemu serwera LAMP spowoduje rozpoczęcie procesu instalacji systemu na komputerze. W dalszej części rozdziału omówiono proces instalowania systemu Ubuntu Server po wybraniu opcji *Zainstaluj na twardym dysku*. W punkcie „Instalowanie serwera LAMP” znajdują się szczegółowe informacje na temat instalacji oprogramowania serwera LAMP.

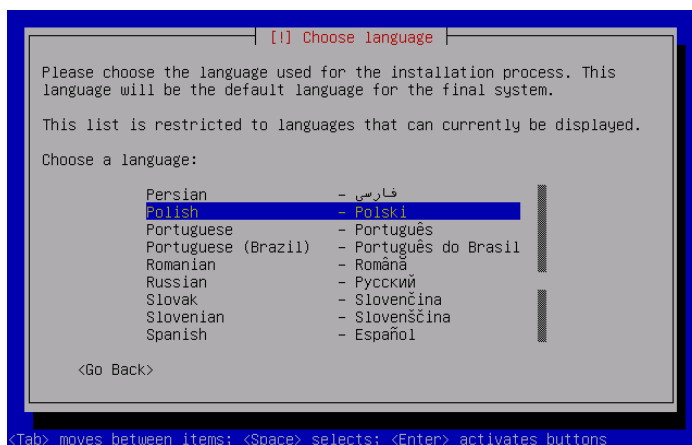
Instalacja serwera Ubuntu

Czas instalacji jest zależny od wydajności systemu komputerowego oraz przepustowości łącza. Oczywiście, warto go poświecić w oczekiwaniu na ostateczny efekt. Aby rozpocząć proces instalacji, należy umieścić w napędzie odpowiednią płytę i ponownie uruchomić komputer. Oto, co należy kolejno robić.

1. Po uruchomieniu komputera zostanie wyświetlone menu instalacyjne widoczne na rysunku 3.9. Aby kontynuować, należy podświetlić pozycję *Zainstaluj na twardym dysku* i wcisnąć przycisk *Enter*.
2. Na rysunku 3.10 wyświetlone jest menu, w którym można wybrać język, w jakim będą wyświetlane kolejne komunikaty instalacyjne. Aby wybrać odpowiedni język, należy użyć klawiszy ze strzałkami, a po znalezieniu odpowiedniej pozycji podświetlić ją i wcisnąć klawisz *Enter*.

Rysunek 3.10.

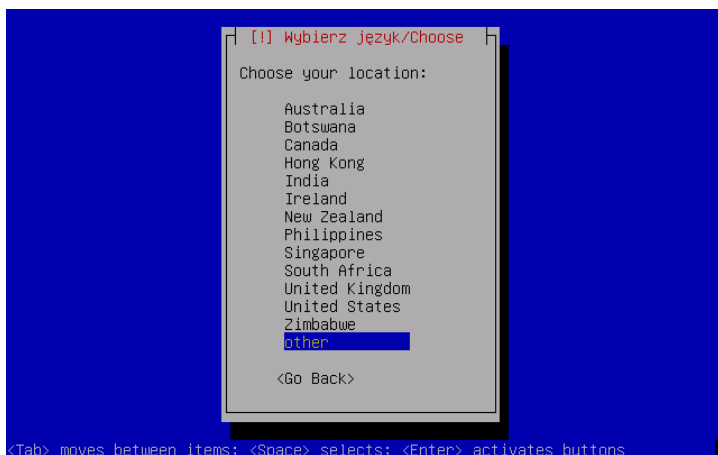
Wybór języka



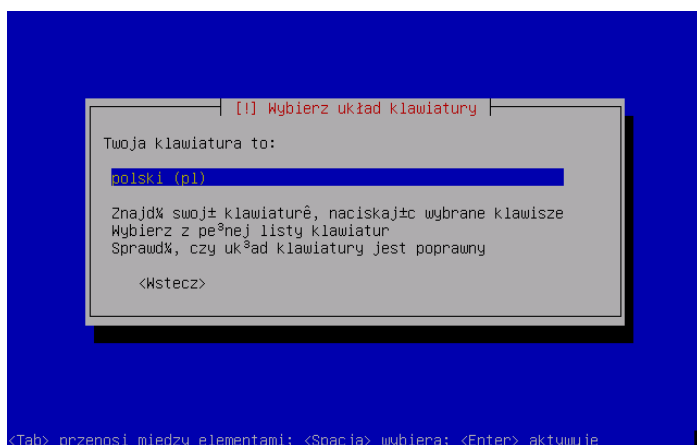
3. W oknie dialogowym widocznym na rysunku 3.11 można określić lokalizację geograficzną komputera. Ta informacja zostanie wykorzystana do dalszego dostosowywania ustawień systemu: formatu daty, formatu liczb, waluty oraz lokalizacji geograficznej.
4. Okno dialogowe przedstawione na rysunku 3.12 umożliwia ustawienie odpowiedniego układu klawiatury. Wartość domyślna jest określona na podstawie informacji wprowadzonych przez użytkownika w poprzednim oknie dialogowym

Rysunek 3.11.

Określanie lokalizacji geograficznej

**Rysunek 3.12.**

Określanie układu klawiatury



— jeżeli układ jest nieodpowiedni, wówczas można wybrać właściwy z obszernej listy. Jeżeli używany jest inny układ niż U.S., najłatwiej odnaleźć go za pomocą klawiszy ze strzałką. Należy wcisnąć klawisz *Enter* i wybrać odpowiedni układ z listy.

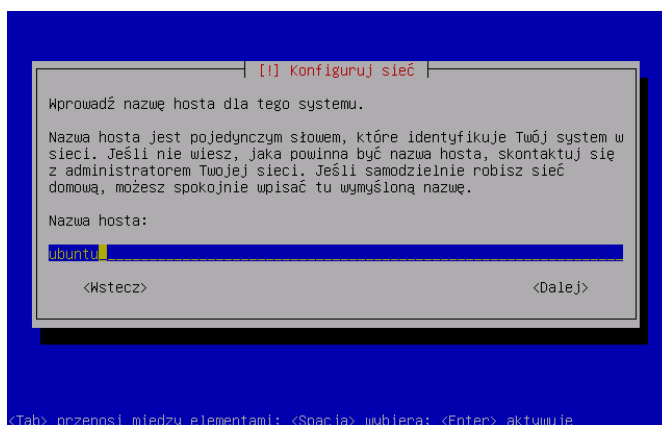
Następnie zostanie wyświetlonych kilka ekranów informujących o działaniach podejmowanych przez komputer: przeszukiwaniu napędu CD-ROM, wykrywaniu podzespołów komputera oraz identyfikacji i konfiguracji ustawień sieci. Użytkownik nie może w tym czasie dokonywać zmian.

Jeżeli program napotka na kłopoty z konfiguracją dostępu do sieci, zostanie wyświetlony odpowiedni komunikat; więcej informacji na ten temat czytelnik znajdzie w podpunkcie niniejszego rozdziału zatytułowanym „Rozwiązywanie problemów instalacyjnych”.

5. W kolejnym ekranie instalacyjnym można wprowadzić nazwę komputera (patrz rysunek 3.13). Ta składająca się z jednego wyrazu nazwa (nie należy używać nazwy domeny) jest używana do identyfikowania komputera w sieci lokalnej. Podczas korzystania z trybu podwójnego rozruchu warto używać tej samej nazwy

Rysunek 3.13.

Określanie nazwy komputera

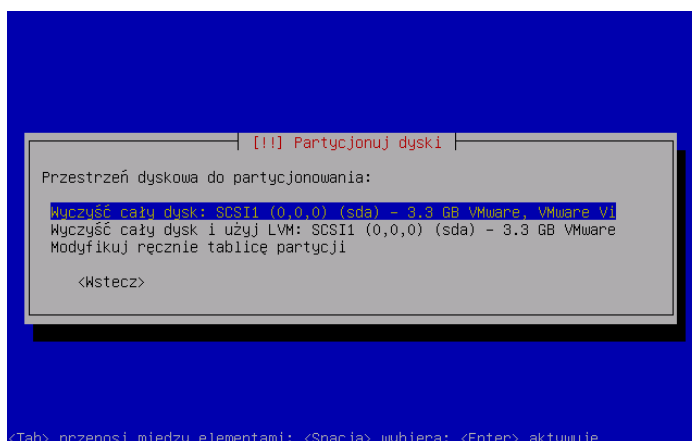


dla obu systemów, dzięki czemu można uniknąć zamieszania. Jeżeli użytkownik zdecyduje się pozostawić domyślną nazwę (ubuntu) lub wprowadził nową, należy następnie wcisnąć klawisz *Enter*.

6. Następnie wyświetlonych zostanie kilka ekranów informujących o wykrywaniu i identyfikowaniu sprzętu, m.in. dysków twardych. Po zakończeniu testów wyświetli się menu programu do partycjonowania dysku (patrz rysunek 3.14).

Rysunek 3.14.

Menu programu do partycjonowania dysków



7. Na kolejnym ekranie programu instalacyjnego (patrz rysunek 3.14) widać jedyny skomplikowany etap całej instalacji, czyli partycjonowanie dysku. Gdy Ubuntu instalowane jest w komputerze z jednym dyskiem, który w dodatku w całości ma być poświęcony na ten system, wówczas użytkownik powinien zobaczyć układ podobny do widocznego na rysunku 3.14 (może z wyjątkiem nazw i rozmiarów dysków). W takim przypadku dostępne będą trzy opcje:

- ♦ *Wyczyść cały dysk...*,
- ♦ *Wyczyść cały dysk i użyj LVM...*,
- ♦ *Modyfikuj ręcznie tablicę partycji...*

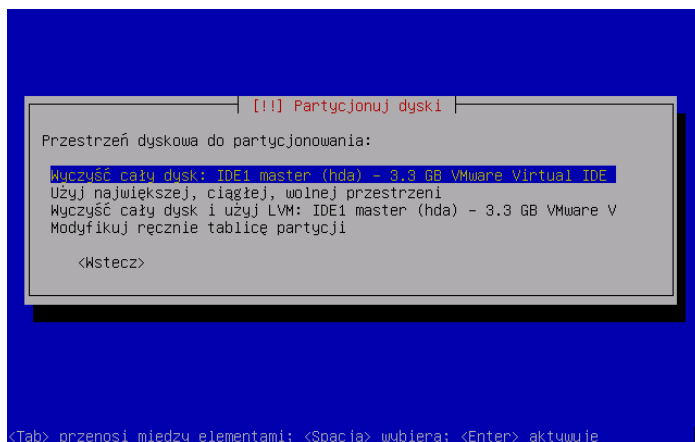
Jeżeli czytelnik zdecyduje się na samodzielne partycjonowanie dysku, powinien sprawdzić informacje na ten temat zamieszczone w tym rozdziale, w punkcie „Określanie układu partycji”. W przeciwnym razie należy wybrać jedną z dwóch pierwszych opcji. Przewagą użycia LVM jest to, że kiedy na skutek braku miejsca do systemu zostanie dodany kolejny dysk, wówczas będzie można dołączyć dostarczoną przez niego przestrzeń do istniejącego dysku logicznego. Istnieje mnóstwo sposobów na osiągnięcie tego samego efektu za pomocą woluminów logicznych, dlatego można zaakceptować domyślne ustawienia i wcisnąć klawisz *Enter*, aby przejść do następnego etapu — chyba, że użytkownik chce wprowadzić inne ustawienia.



Jeżeli Ubuntu instalowany jest z wykorzystaniem opcji podwójnego rozruchu na komputerach x86 lub 64-bitowych, wówczas ekran, który zostanie wyświetlony, będzie podobny do tego, jaki widać na rysunku 3.15. W tym przypadku można pozwolić systemowi automatycznie zmienić rozmiar partycji, można też nakazać wykorzystanie największej wolnej przestrzeni, gdy użytkownik wcześniej zmienił rozmiar partycji lub dostępne jest miejsce na instalację Ubuntu. Jeżeli użytkownik wybierze opcję zmiany rozmiaru, wówczas w kolejnym etapie będzie musiał określić, do jakiej wielkości ma zostać zmniejszona partycja Windows. Nowy rozmiar można określić zarówno ilościowo (np. 10 GB), jak i procentowo (np. 25%).

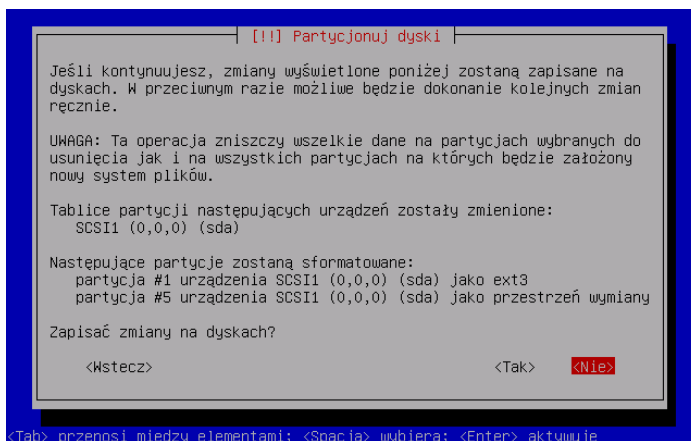
Rysunek 3.15.

Partycjonowanie
dysku w trybie
podwójnego rozruchu

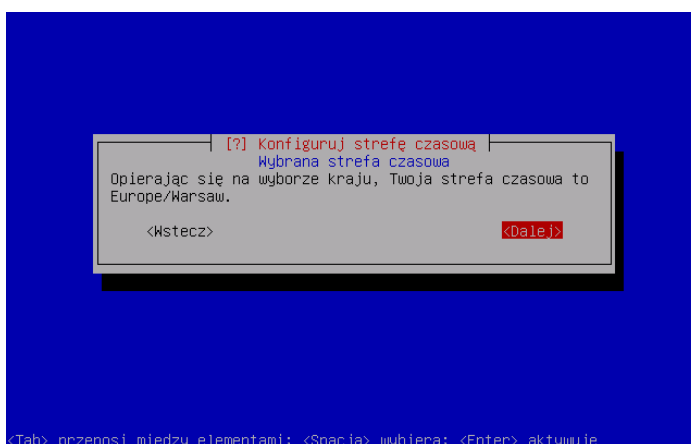


8. Po zakończeniu konfigurowania nowego ustawienia partycji zostanie wyświetlony ekran podsumowujący (patrz rysunek 3.16). Zgodnie z zamieszczonymi informacjami, operacja nie może być cofnięta. Dlatego należy dwa razy sprawdzić, czy wprowadzone ustawienia są poprawne. Jeżeli ustawienia są odpowiednie, wówczas za pomocą klawisza *Tab* trzeba podświetlić pole *Tak* i wcisnąć przycisk *Enter*. Aby wprowadzić zmiany w stosunku do wyświetlonego układu, należy podświetlić pole *Nie* i wcisnąć klawisz *Enter*. Zostanie wyświetlony ekran programu do partycjonowania dysków, omówiony w punkcie pt. „Określanie układu partycji”. Przed przejściem do kolejnego etapu instalacji zalecam, by czytelnik zapoznał się z zamieszczonymi tam informacjami.
9. Menu przedstawione na rysunku 3.17 umożliwia określenie strefy czasowej używanej przez system. Nawet w przypadku korzystania z komputera przenośnego należy wprowadzić strefę czasową, która będzie traktowana jako podstawowa,

Rysunek 3.16.
Partycjonowanie
dysku



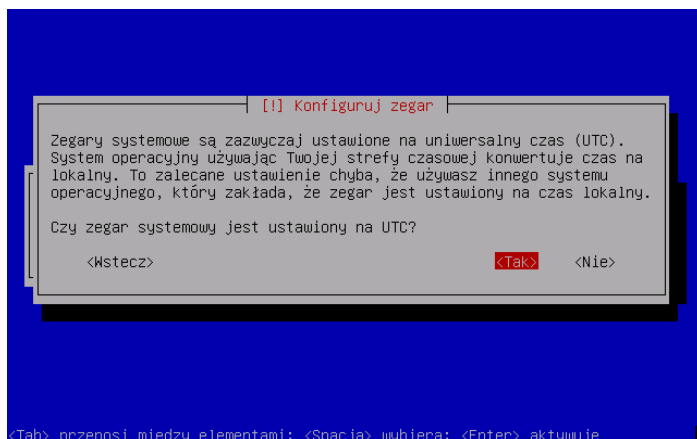
Rysunek 3.17.
Określanie strefy
czasowej



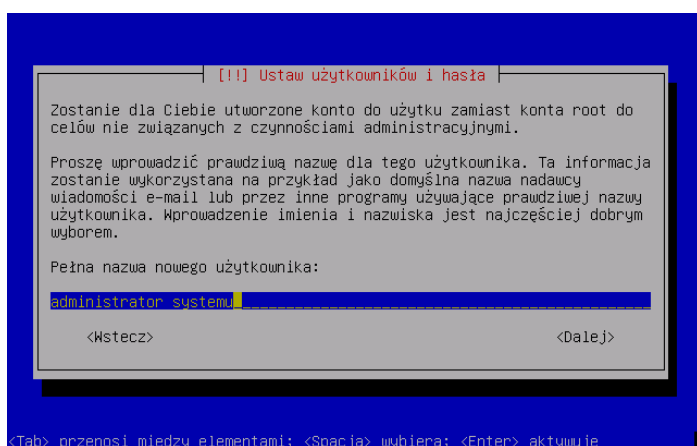
zawsze można ją zmienić w późniejszym czasie. Za pomocą klawiszy ze strzałkami trzeba wskazać właściwą pozycję i wcisnąć klawisz *Enter*.

10. W menu przedstawionym na rysunku 3.18 użytkownik może zdecydować, czy w systemie ma być używany czas UTC, czy lokalny. Właściwą odpowiedź należy podświetlić za pomocą klawisza *Tab* i wcisnąć klawisz *Enter*.
11. Następny ekran programu instalacyjnego umożliwia wprowadzenie nazwy użytkownika do systemu (patrz rysunek 3.19). Zgodnie z informacjami zamieszczonymi w rozdziale 4., „Podstawowe założenia systemu Linux”, w Ubuntu używany jest specjalny schemat uprawnień umożliwiający domyślnemu użytkownikowi wykonywanie zadań administratora komputera. Użytkownik, którego nazwa zostanie tu wprowadzona, będzie mógł wykonywać zadania administratora systemu, dlatego powinna to być nazwa głównego użytkownika (patrz rysunek 3.19). Należy wprowadzić poprawną nazwę i wcisnąć klawisz *Enter*.
12. W następnym ekranie należy wprowadzić nazwę użytkownika, która będzie wykorzystywana w systemie. Poprzednio wprowadzona była pełna nazwa, teraz należy podać jej skróconą wersję (patrz rysunek 3.20). Program podpowie nazwę opartą na wprowadzonej poprzednio. Jeżeli czytelnik chciałby ją zmienić, należy

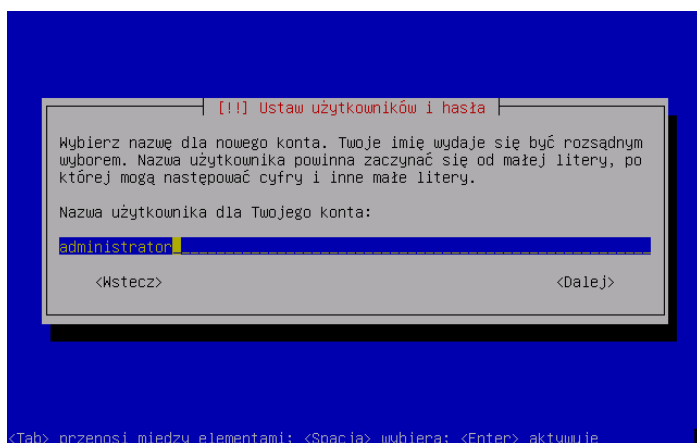
Rysunek 3.18.
Konfigurowanie czasu
systemowego



Rysunek 3.19.
Tworzenie konta
domyślnego
użytkownika

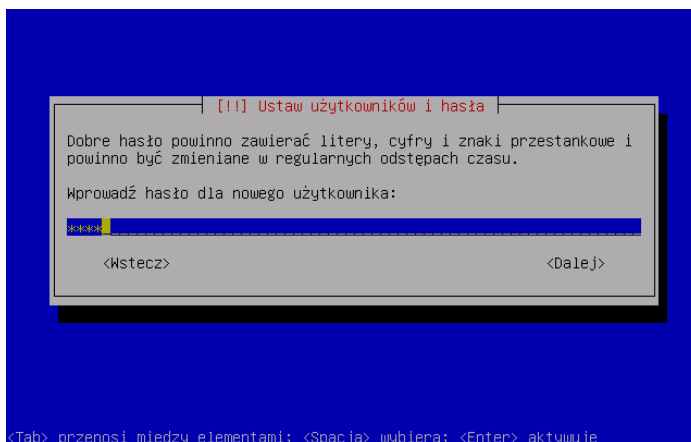


Rysunek 3.20.
Wprowadzanie nazwy
użytkownika systemu

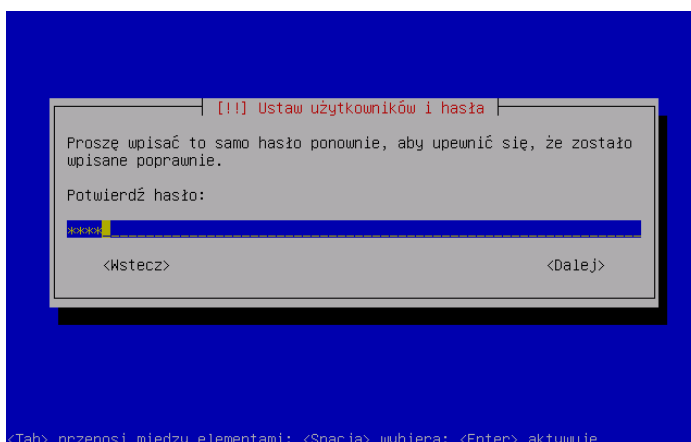


usunąć propozycję za pomocą klawisza *Backspace*, a następnie wprowadzić nową. Po wprowadzeniu odpowiedniej nazwy należy przejść dalej, wciskając klawisz *Enter*.

13. Następnie zostanie wyświetlone okno widoczne na rysunku 3.21. Należy wprowadzić hasło dla tworzonego konta użytkownika, pamiętając o zasadach tworzenia bezpiecznych haseł, takich jak używanie dużych i małych liter oraz cyfr (np. dat urodzin współmałżonka czy dzieci). Aby przejść dalej, należy wcisnąć klawisz *Enter*.

Rysunek 3.21.*Wprowadzanie hasła*

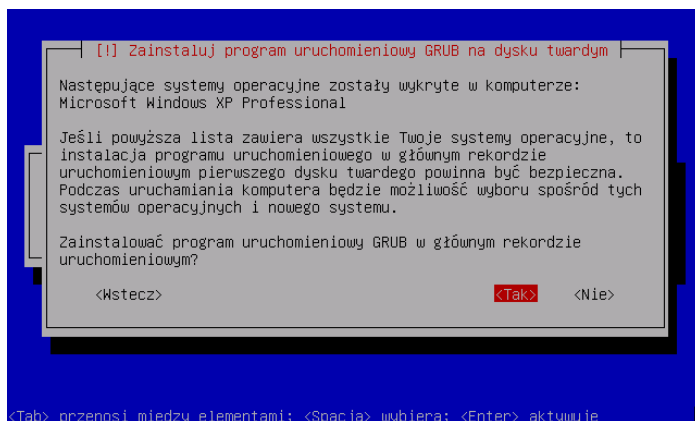
14. Po wprowadzeniu hasła użytkownik zostanie poproszony o powtórne jego wpisanie w celu potwierdzenia zgodności (patrz rysunek 3.22). Należy wprowadzić to samo hasło i wcisnąć klawisz *Enter*.

Rysunek 3.22.*Potwierdzanie hasła*

15. W tym momencie rozpocznie się kopiowanie plików z płyty CD na dysk komputera. Zostanie również skonfigurowany system aktualizowania oprogramowania, używany do pobierania i instalowania programów z repozytoriów znajdujących się najbliżej położenia, które użytkownik wskazał we wcześniejszych etapach instalowania systemu. Systemy aktualizowania oprogramowania w Ubuntu, takie jak *apt-get*, *aptitude* i *Synaptic*, zostały omówione w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.
16. Jeżeli Ubuntu Server instalowany będzie z opcją podwójnego rozruchu, wówczas podczas instalacji pakietów zostanie wyświetlony komunikat widoczny na

rysunku 3.23. GRUB, program rozruchowy używany w Ubuntu, domyślnie jest instalowany w głównym sektorze rozruchowym podstawowego dysku twardego. Aby zainstalować GRUB-a w tej lokalizacji, należy wcisnąć klawisz *Enter*.

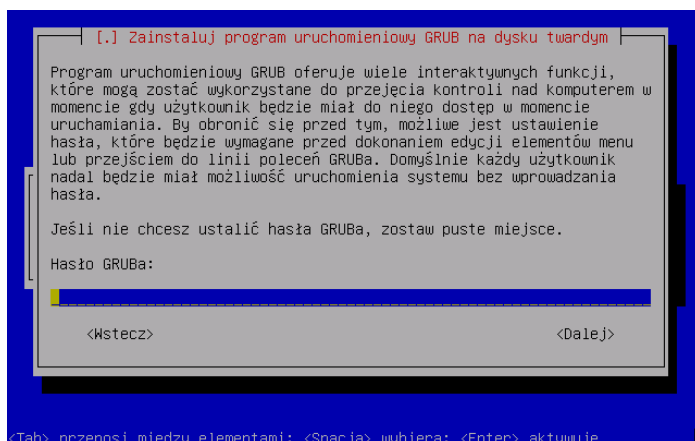
Rysunek 3.23.
*Instalowanie GRUB-a
na komputerze
z opcją podwójnego
rozruchu*



Jeżeli podczas instalowania systemu zostaną wyświetlone komunikaty błędów dotyczące repozytoriów, informacje na temat rozwiązania kłopotów znajdzie czytelnik w podpunkcie „Rozwiązywanie problemów instalacyjnych”. Nie są to błędy krytyczne, da się je usunąć.

Aby zainstalować program GRUB w innym miejscu, należy użyć klawisza, aby je wskazać, i wcisnąć klawisz *Enter*. Na rysunku 3.24 widać dostępne lokalizacje, w których można zainstalować program GRUB. Należy wybrać odpowiednią pozycję i wcisnąć klawisz *Enter*.

Rysunek 3.24.
*Menu instalacyjne
programu
rozruchowego GRUB*

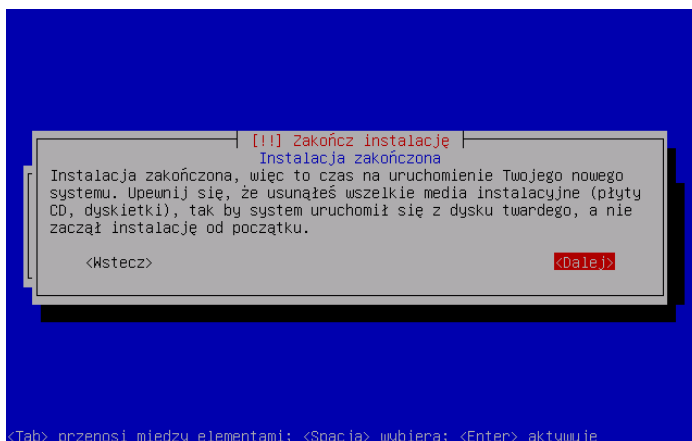


17. Po skonfigurowaniu kilku dodatkowych pakietów zostanie wyświetlony komunikat o zakończeniu instalacji (patrz rysunek 3.25).

Kiedy znajdzie się on na ekranie, należy usunąć płytę instalacyjną z napędu i wcisnąć klawisz *Enter*, wtedy komputer zostanie uruchomiony ponownie, a wraz z nim nowo zainstalowany system Ubuntu Linux Server! Więcej informacji można znaleźć w punkcie „Pierwsze uruchomienie systemu”.

Rysunek 3.25.

Ostatni ekran
instalacyjny



Określanie układu partycji

Samodzielne określanie układu partycji może być użyteczne, ale jest także czasochłonne oraz wymaga znajomości zasad działania systemu plików w Linuksie. Ręczne partycjonowanie umożliwia umieszczenie odpowiednich katalogów systemu plików Linuksa na osobnych partycjach, co może poprawić wydajność, zminimalizować wielkość kopii zapasowych systemu oraz uprościć proces aktualizowania czy ponownego konfigurowania systemu.



Domyślnie program instalacyjny Ubuntu automatycznie tworzy całkiem rozsądny schemat, który wyznacza jedną dużą partycję przeznaczoną na pliki użytkownika i systemowe oraz drugą, mniejszą partycję wymiany *swap*, czyli wirtualną pamięć systemu. Poniżej opisano alternatywny wobec proponowanego przez program instalacyjny schemat podziału dysku na partycje. Czytelnik może pominąć ten fragment, jeżeli zapozna się z książką systematycznie i ma już zainstalowany system operacyjny.

Jeżeli jednak czytelnik zamierza zainstalować Ubuntu po raz pierwszy lub też jest początkującym użytkownikiem Linuksa, wówczas zalecamy pozostawienie partycjonowania programowi instalacyjnemu. Automatyczne partycjonowanie na pewno zajmie mniej czasu, nie wymaga też znajomości przeznaczenia różnych katalogów systemu plików Linuksa. W miarę stawania się coraz bardziej zaawansowanym użytkownikiem Ubuntu można z czasem wykonać kopię zapasową swojego konta, a następnie przeinstalować system, partycjonując dysk zgodnie ze swoimi potrzebami.

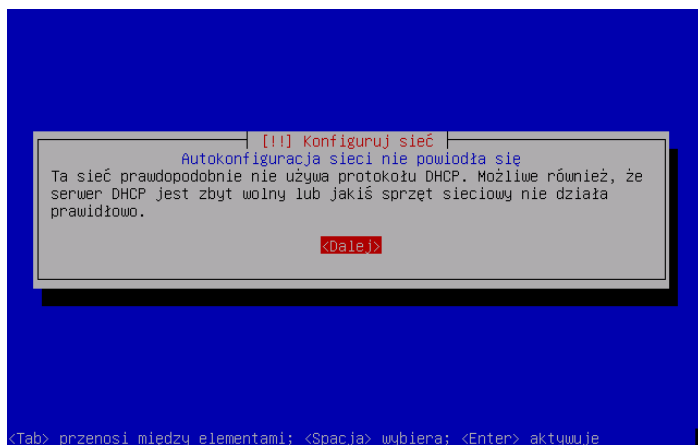
Oto katalogi, które można znaleźć na osobnych partycjach.

- ♦ `/`: Najwyższy poziom w systemie plików Linuksa. Każda partycja musi być dostępna w katalogu `/`.
- ♦ `/boot`: W tym katalogu przechowywane są jądro systemu oraz powiązane z nim pliki używane podczas procesu uruchamiania systemu.
- ♦ `/home`: Katalog zawierający pliki i konta użytkowników.
- ♦ `/opt`: Katalog zawierający dodatkowe programy i związane z nimi pliki.
- ♦ `/tmp`: Katalog używany do przechowywania tymczasowych plików tworzonych przez użytkowników oraz procesy systemowe.

Rozwiązywanie problemów instalacyjnych

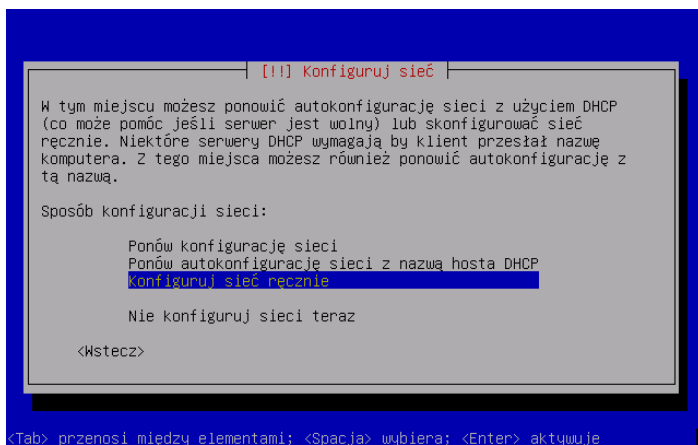
Większość błędów pojawiających się podczas instalacji jakiegokolwiek systemu, włącznie z Ubuntu Linux, to błędy krytyczne uniemożliwiające dokończenie procesu. Wszelkie problemy związane z odczytywaniem i dostępem do danych z nośników, z partycjami, formatowaniem dysków komputera i innym sprzętem należy rozwiązać przed zakończeniem instalowania systemu. Na szczęście, dwa najczęściej pojawiające się błędy są dość łatwe do usunięcia. Pierwszy z nich dotyczy automatycznej konfiguracji połączeń sieciowych; drugi — dostępu do internetowych repozytoriów Ubuntu oraz poprawnego skonfigurowania systemu aktualizacji oprogramowania.

Jeżeli program instalacyjny Ubuntu nie może automatycznie skonfigurować sieci, zostanie wyświetlony komunikat i użytkownik będzie miał możliwość samodzielnego wprowadzenia właściwych ustawień (patrz rysunek poniżej).



Informacja o nieudanej automatycznej konfiguracji dostępu do sieci

Aby przejść do następnego etapu, należy wcisnąć klawisz **Enter**.



Menu umożliwiające samodzielną konfigurację ustawień sieciowych

Użytkownik może zdecydować, co system powinien zrobić dalej: czy ma ponowić próbę automatycznej konfiguracji, rozpocząć konfigurację poprzez DHCP z określoną nazwą hosta, czy umożliwić samodzielną konfigurację, czy w końcu pominąć ten etap instalacji systemu. Przed wykonaniem kolejnego kroku należy sprawdzić, czy wszystkie kable są odpowiednio podłączone, zwłaszcza w przypadku korzystania z DHCP.

Można wybrać każdą z dostępnych możliwości, wskazane jest jednak samodzielne skonfigurowanie ustawień i podanie statycznego adresu IP oraz pozostałych informacji. Z doświadczenia autora wynika, że niewiele serwerów posiada dynamiczne adresy IP, a to dlatego, że utrudnia to połączenie się z nimi w przypadku wystąpienia problemów, bowiem użytkownicy nie znają ich adresów IP.

Jeżeli użytkownik wybierze opcję samodzielnego konfigurowania połączenia sieciowego, wówczas program instalacyjny wyświetli kolejne cztery ekrany, na których będzie można wprowadzić dodatkowe informacje instalacyjne: adres IP, maskę sieciową, adres IP bramy oraz nazwę serwera w sieci (gdyby użytkownik chciał wprowadzić kilka nazw, wówczas należy oddzielić je spacją, nie przecinkiem). Po wprowadzeniu tych informacji interfejs sieciowy jest skonfigurowany i nastąpi powrót do programu instalacyjnego.

Kolejnym problemem, na który można natknąć się podczas instalacji Ubuntu, jest nawiązanie połączenia oraz sprawdzenie wszystkich dostępnych repozytoriów, czyli stron internetowych zawierających nowe i zaktualizowane pakiety oprogramowania dla różnych wydań Ubuntu. Jeżeli problem wystąpi podczas próby połączenia się z repozytoriami, wówczas — podczas procesu instalacji — zostanie wyświetlony odpowiedni komunikat. Nie powinien być on jednak powodem do szczególnych zmartwień. To zwykle przejściowy problem z połączeniem sieciowym lub pobraniem danych. Po ponownym uruchomieniu systemu należy usunąć znak komentarza umieszczony przy odpowiednim repozytorium (patrz rysunek 3.26).

Najczęściej problemy te dotyczą repozytoriów bezpieczeństwa; aby usunąć znak komentarza, należy za pomocą terminala uruchomić ulubiony edytor tekstu (oto przykład polecenia dla programu vi: `sudo vi /etc/apt/sources.list`). Następnie należy usunąć znaki komentarza (#) z początku każdej linii odpowiadającej danemu repozytorium, zapisać wprowadzone zmiany i wykonać polecenie `sudo apt-get update`, które powoduje sprawdzenie, czy można połączyć się z nowymi repozytoriami. W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, znaleźć można więcej informacji na temat repozytoriów Ubuntu oraz poleceń służących do aktualizowania oprogramowania, takich jak `apt-get`.

- ♦ */usr*: W katalogu tym przechowywane są programy, pliki systemowe i biblioteki używane w systemie Linux. W Ubuntu na katalog */usr* należy przeznaczyć ok. 1,25 GB przestrzeni dyskowej.
- ♦ */usr/local*: W katalogu tym przechowywane są programy, pliki systemowe i biblioteki używane w konkretnym systemie, co oznacza, że w innych dystrybucjach może ich nie być.

Dystrybucje, które stosują osobne partycje dla różnych katalogów, zwykle umieszczają na nich: */*, */boot*, i */home*. Więcej informacji o tych katalogach oraz o strukturze systemu plików można znaleźć w rozdziale 4., „Podstawowe założenia systemu Linux”.

Partycje linuksowe można podzielić na dwa typy: partycje wymiany używane do wspomagania zarządzania pamięcią oraz partycje przeznaczone dla danych, na których przechowywane są pliki i katalogi. Aby działać poprawnie, system Linux musi mieć przynajmniej jeden obszar wymiany — zwykle jest to przeznaczona do tego celu partycja, ale można

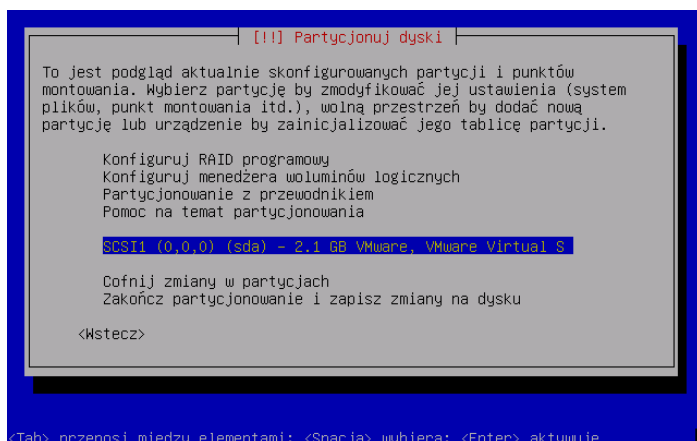
też wykorzystać specjalny plik. W rozdziale 4. umieściłem więcej informacji na temat partycji i wirtualnej pamięci. Poniżej opisałem, w jaki sposób utworzyć kilka partycji do przechowywania danych oraz jedną partycję wymiany — ma to zobrazować schemat tworzenia partycji; to, jaki czytelnik wybierze, zależy tylko od niego.

Samodzielne partycjonowanie dysku składa się z następujących etapów.

1. Na ekranie widocznym na rysunku 3.14 należy wybrać opcję *Modyfikuj ręcznie tablicę partycji*. Zostanie wówczas wyświetlone okno widoczne na rysunku 3.26.

Rysunek 3.26.

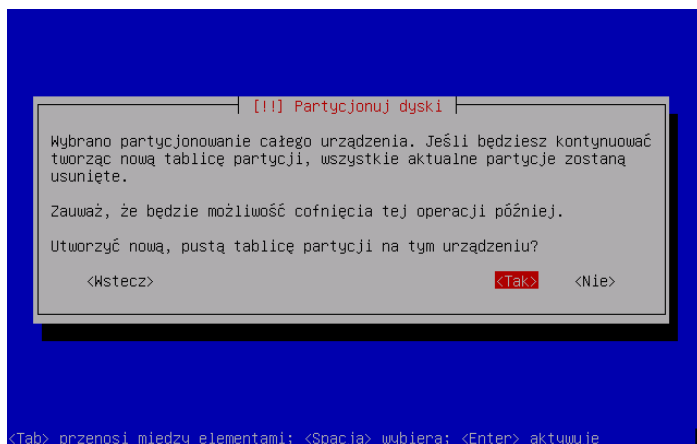
Ręczne
partycjonowanie
dysku komputera



2. Za pomocą klawiszy ze strzałkami należy wskazać odpowiedni dysk i wcisnąć klawisz *Enter*. Zostanie wyświetlone okno widoczne na rysunku 3.27.

Rysunek 3.27.

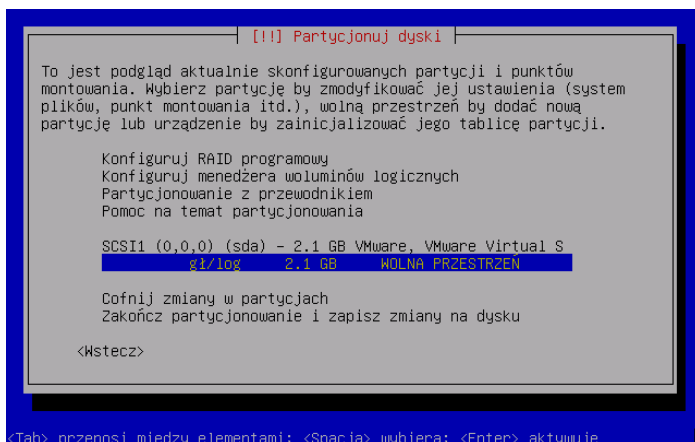
Ekran informacyjny
przed rozpoczęciem
samodzielnego
partycjonowania
dysku



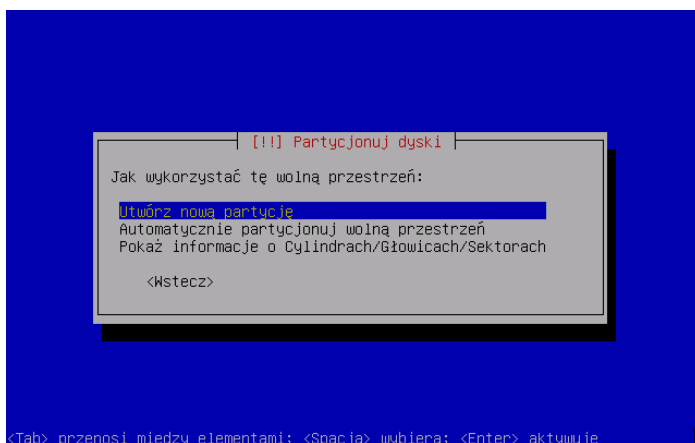
3. Za pomocą klawisza *Tab* należy podświetlić pozycję *Tak* i wcisnąć klawisz *Enter*. Dzięki temu na dysku zostanie utworzona duża, wolna przestrzeń (patrz rysunek 3.28).
4. Za pomocą klawiszy ze strzałkami należy wskazać odpowiednią pozycję i wcisnąć klawisz *Enter*. Zostanie wyświetlone okno widoczne na rysunku 3.29.

Rysunek 3.28.

Wskazanie przestrzeni dyskowej, która zostanie wykorzystana w dalszym etapie

**Rysunek 3.29.**

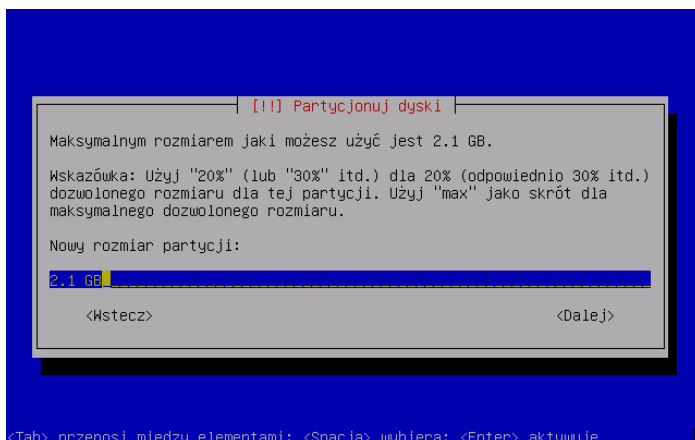
Początek definiowania nowej partycji



5. Teraz trzeba podświetlić pozycję *Utwórz nową partycję* i wcisnąć klawisz *Enter*. Zostanie wyświetlone okno widoczne na rysunku 3.30.

Rysunek 3.30.

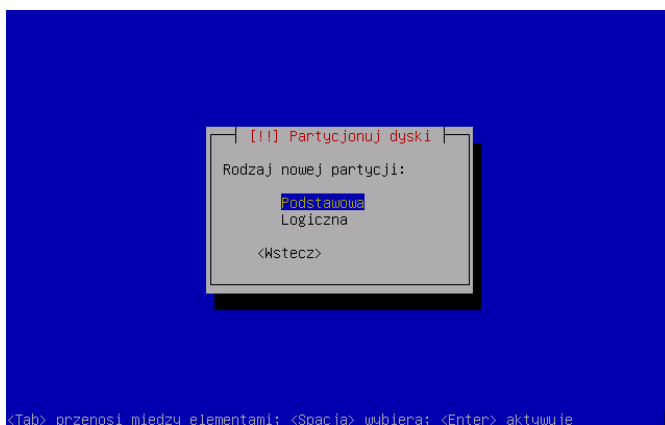
Określanie rozmiaru nowej partycji



6. Należy wprowadzić rozmiar nowej partycji. Domyślnie wyświetlana jest całkowita ilość dostępnej przestrzeni. Można podać wielkość partycji jako wartość bezwzględną (np. 100 MB), określić ją jako procent dostępnej przestrzeni (np. 10%) lub też utworzyć partycję, wykorzystując całą dostępną przestrzeń. Po wprowadzeniu wielkości partycji należy wcisnąć klawisz *Enter*. Zostanie wyświetlone okno widoczne na rysunku 3.31.

Rysunek 3.31.

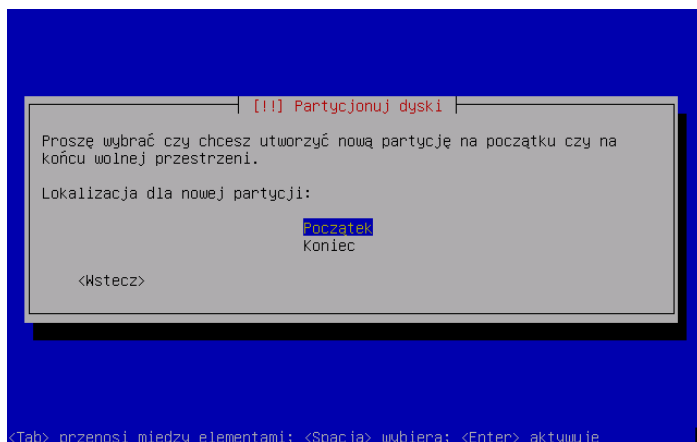
Określanie typu nowej partycji



7. Trzeba określić typ tworzonej partycji. Na dysku może znajdować się do czterech partycji podstawowych, każda kolejna musi być partycją logiczną. Partycja logiczna może zawierać inne partycje, podczas gdy partycja podstawowa to po prostu wydzielony kawałek dysku, którego nie można już bardziej podzielić. Jeżeli użytkownik nie planuje utworzenia więcej niż czterech partycji, wówczas może wybrać opcję *Podstawowa*, a następnie wcisnąć klawisz *Enter*. Zostanie wyświetlone okno widoczne na rysunku 3.32.

Rysunek 3.32.

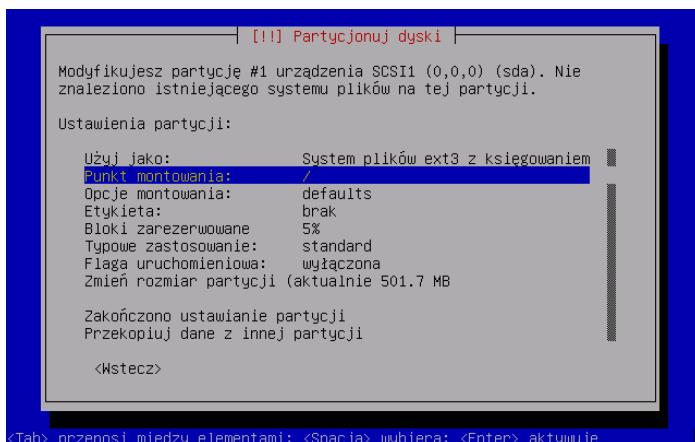
Określanie położenia nowej partycji



8. Jeżeli nie ma specjalnych przesłanek dla umieszczenia partycji w określonym miejscu dysku, wówczas należy zaakceptować domyślną wartość (*Początek*) i wcisnąć klawisz *Enter*. Zostanie wyświetlone okno widoczne na rysunku 3.33.

Rysunek 3.33.

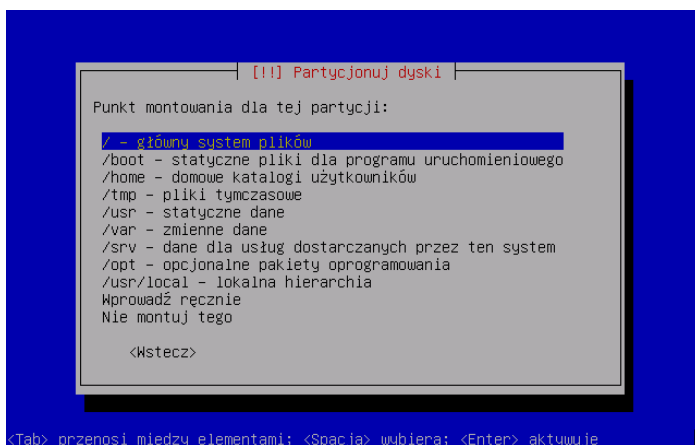
*Określanie typu
i punktu montowania
nowej partycji*



9. **Punkt montowania** partycji to inaczej katalog, w którym dostępna będzie dla systemu operacyjnego zawartość partycji. W rozdziale 4., „Podstawowe założenia systemu Linux”, można znaleźć dodatkowe informacje na temat punktów montowania i katalogów systemu Linux. Lista partycji i powiązanych z nimi punktów montowania została przedstawiona na początku niniejszego podrozdziału. Jedynym obowiązkowym katalogiem powiązany z punktem montowania jest /, główny katalog systemu plików. Za pomocą klawiszy ze strzałkami należy wskazać odpowiedni punkt montowania i wcisnąć klawisz *Enter*. Zostanie wyświetlone okno widoczne na rysunku 3.34.

Rysunek 3.34.

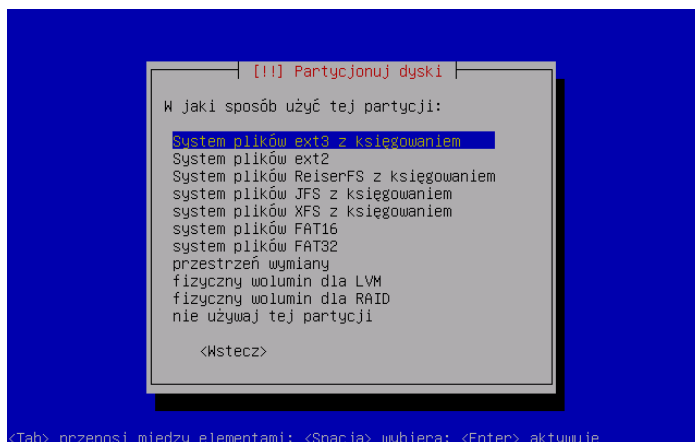
*Określanie punktu
montowania nowej
partycji*



10. W tym przykładzie nowa partycja została zamontowana w katalogu */boot*, w którym umieszczone jest jądro systemu oraz powiązane z nim pliki. Za pomocą klawiszy ze strzałkami należy przejść do katalogu */boot*, a następnie wcisnąć klawisz *Enter*. Zostanie wyświetlone okno widoczne na rysunku 3.35.
11. Linux obsługuje kilka różnych systemów plików. Każdy z nich posiada charakterystyczne właściwości, które zostały omówione w rozdziale 4., „Podstawowe założenia systemu Linux”. W tym przypadku partycja */boot* służy do przechowywania

Rysunek 3.35.

Określanie typu partycji

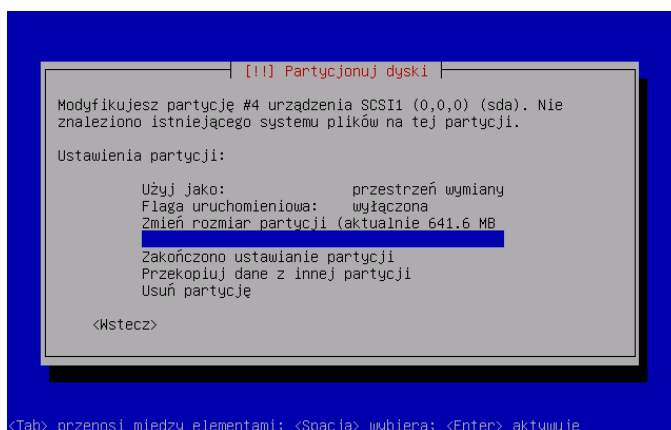


danych, a domyślny system plików ext3 to najpopularniejszy format dla tego typu partycji i ten należy wybrać. Zostanie wyświetlone okno widoczne na rysunku 3.27.

12. Następnie należy powtórzyć etapy od 3. do 11. i utworzyć kolejną partycję. Jak wspominałem wcześniej, oprócz partycji przeznaczonej na przechowywanie danych (w tym przypadku /boot) oraz partycji /i /home, potrzebna będzie jeszcze partycja wymiany. Tworzenie innych partycji przebiega zgodnie z procesem opisanym powyżej. Aby utworzyć partycję wymiany, należy powtórzyć opisaną powyżej procedurę z tą różnicą, że na ekranie instalacyjnym widocznym na rysunku 3.35 trzeba wybrać opcję „przestrzeń wymiany”. Na rysunku 3.36 przedstawiłem konfigurowanie partycji wymiany. Jest to tylko potwierdzenie, że tworzona jest przestrzeń wymiany, na której nie będzie można przechowywać żadnych danych.

Rysunek 3.36.

Szczegółowe informacje na temat tworzonej partycji wymiany



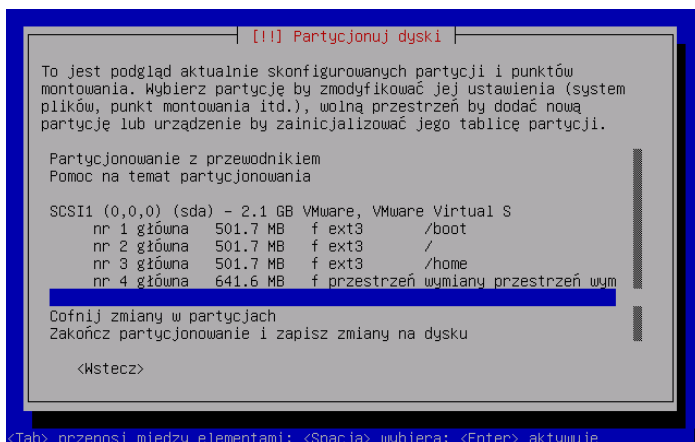
13. Po rozdysponowaniu całej dostępnej przestrzeni dyskowej pomiędzy partycje zostanie wyświetlony ekran widoczny na rysunku 3.37.



Jeżeli użytkownik planuje utworzenie więcej niż czterech partycji, wówczas piąta i każda kolejna muszą być partycjami rozszerzonymi, w których tworzone będą kolejne.

Rysunek 3.37.

Ekran
podsumowujący
wprowadzone zmiany

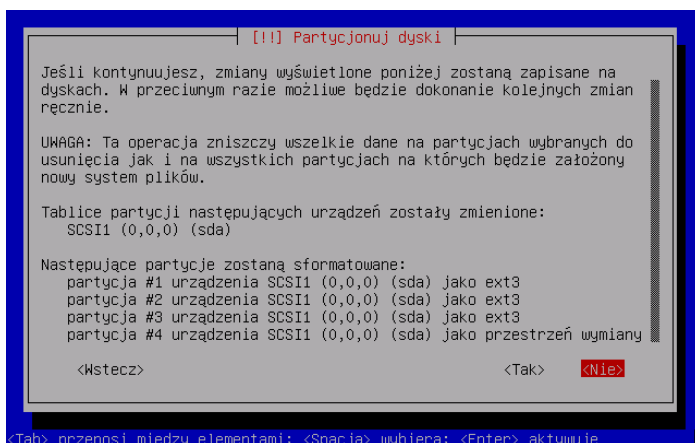


Aby się upewnić, że podczas instalowania systemu nie pojawiają się kłopoty, należy sprawdzić, czy partycja montowana w `/` ma rozmiar co najmniej 3 GB. Stosowanie tej zasady powinno zagwarantować poprawną instalację systemu, chyba że użytkownik zdefiniował inne partycje, takie jak `/usr`, z niewystarczającą ilością miejsca do przechowywania plików, które zostaną tam zapisane.

14. Na rysunku 3.37 widać podsumowanie zmian wprowadzonych przez użytkownika w układzie partycji. Należy pamiętać, że musi być jedna partycja zamontowana w punkcie `/` oraz jedna partycja wymiany. Aby zatwierdzić wprowadzone zmiany, należy za pomocą klawiszy ze strzałkami podświetlić opcję *Zakończ partycjonowanie i zapisz zmiany na dysku*, a następnie wcisnąć klawisz *Return*, zostanie wówczas wyświetlony kolejny ekran programu instalacyjnego (patrz rysunek 3.38).

Rysunek 3.38.

Ostateczne
potwierdzenie
dokonanych zmian



15. Ponieważ zastosowanych zmian nie można cofnąć, system wymaga jeszcze jednego potwierdzenia ze strony użytkownika, że zmiany mają być wprowadzone. Aby je zatwierdzić i przejść do następnego etapu procesu instalacji, należy za pomocą klawisza *Tab* podświetlić opcję *Tak* i wcisnąć klawisz *Enter*.

Teraz można przejść do etapu 9. instalacji Ubuntu opisanego w poprzednim punkcie. Gratulacje — kolejny etap szkolenia w zakresie Linuksa został zakończony.

Instalowanie serwera LAMP

Instalowanie Ubuntu w wersji serwera LAMP przebiega w taki sam sposób jak instalowanie zwykłego serwera, co opisałem wcześniej w punkcie „Instalacja serwera Ubuntu”. Dodatkowo zostaną zainstalowane Apache 2, MySQL i Perl.

Po zakończeniu procesu instalacji oraz po pierwszym zalogowaniu się (co opisałem poniżej), użytkownik zapewne będzie chciał dopasować system do swoich potrzeb. W rozdziale 26., „Konfiguracja serwera WWW”, można znaleźć dodatkowe informacje na temat konfigurowania Apache 2. Kolejną czynnością będzie zapewne ustawienie hasła dla głównego użytkownika w MySQL, można to zrobić w następujący sposób:

```
$ mysql -u root mysql
mysql> update user set Password=password('nowe_hasło') where user='root';
mysql> flush privileges;
mysql> quit
```

Polecenie `update` zastąpi ciąg `nowe_hasło` tym, które zostanie podane przez użytkownika. Po ustaleniu i wprowadzeniu nowego hasła można sprawdzić, czy zostało wpisane poprawnie. W tym celu należy w linii poleceń wykonać następujące polecenie:

```
$ mysql -u root mysql -p
Enter password:
```

Jeżeli nowe hasło jest poprawne, wówczas użytkownik zostanie połączony z bazą MySQL.

Pierwsze uruchomienie systemu

Po zakończeniu procesu instalacji systemu w wersji przeznaczonej na serwer zostanie wyświetlony ekran widoczny na rysunku 3.23. Należy wówczas usunąć płytę instalacyjną z napędu i wcisnąć klawisz *Enter*, co spowoduje ponowne uruchomienie komputera i nowo zainstalowanego systemu.



Proces uruchamiania systemu zależy od tego, czy jest to jedyny system operacyjny na dysku, czy też wybrano tryb podwójnego rozruchu. Choć wydaje się mało prawdopodobne, aby użytkownik chciał instalować serwer Ubuntu wraz z innym systemem, to informacje na ten temat można znaleźć w rozdziale 2., w punkcie „Uruchamianie Ubuntu”.

Podczas uruchamiania Ubuntu zobaczymy kilka komunikatów dotyczących testowania sprzętu i oprogramowania. Po zakończeniu całego procesu pojawi się linia poleceń, a użytkownik będzie miał możliwość zalogowania się do systemu (patrz rysunek 3.39). Należy wówczas podać nazwę użytkownika i hasło ustalone podczas procesu instalowania systemu.

Gratulacje, serwer Ubuntu został uruchomiony. Więcej informacji na temat Linuksa czytelnik może znaleźć w rozdziale 4., „Podstawowe założenia systemu Linux”. Jeśli chciałby np. uruchomić jedną z setek wydajnych aplikacji dostarczanych wraz z Ubuntu, wystarczy sprawdzić ich spis dostępny na końcu książki i przejść wprost do rozdziału, w którym jest opisana.

Rysunek 3.39.

Logowanie
do serwera Ubuntu

```
* Setting up LVM Volume Groups... [ ok ]
* Starting Enterprise Volume Management System... [ ok ]
* Checking all filesystems... [ ok ]
* Configuring network interfaces... [ ok ]
* Setting up general console font... [ ok ]
* Setting up per-UC fonts...
* /dev/tty2
* /dev/tty3
* /dev/tty4
* /dev/tty5
* /dev/tty6
[ ok ]

* INIT: Entering runlevel: 2
* Loading ACPI modules... [ ok ]
* Starting ACPI services... [ ok ]
* Starting system log... [ ok ]
* Starting kernel log... [ ok ]
* Starting RAID monitoring services... [ ok ]
* Starting deferred execution scheduler... [ ok ]
* Starting periodic command scheduler... [ ok ]
* Running local boot scripts (/etc/rc.local) [ ok ]

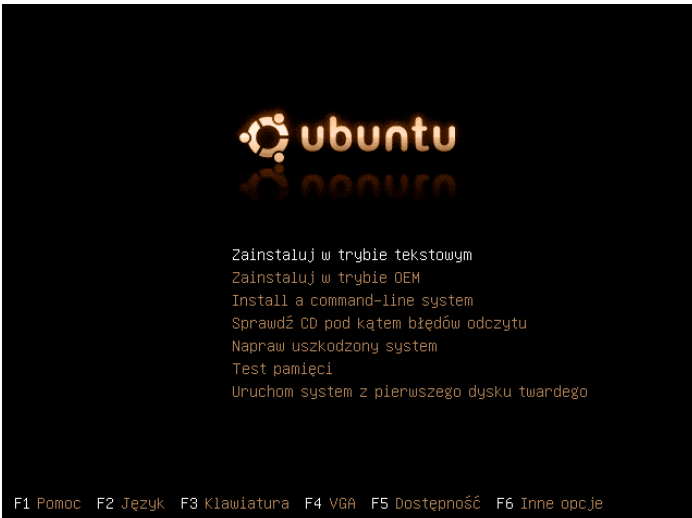
Ubuntu 6.06 LTS ubuntu tty1
ubuntu login:
```

Opcje instalacyjne dla płyty Alternate Install CD

Na rysunku 3.40 widoczny jest ekran, który pojawi się po uruchomieniu komputera z płyty Ubuntu Alternate Install CD.

Rysunek 3.40.

Menu instalacyjne
płyty Ubuntu
Alternate Install CD



```

ubuntu

Zainstaluj w trybie tekstowym
Zainstaluj w trybie OEM
Install a command-line system
Sprawdź CD pod kątem błędów odczytu
Napraw uszkodzony system
Test pamięci
Uruchom system z pierwszego dysku twardego

F1 Pomoc F2 Język F3 Klawiatura F4 VGA F5 Dostępność F6 Inne opcje
```

Oto kolejne opcje dostępne w tym menu.

- ♦ **Zainstaluj w trybie tekstowym:** system zostanie zainstalowany na dysku komputera z wykorzystaniem pseudograficznego programu instalacyjnego. Ten proces został opisany w punkcie pt. „Instalowanie systemu Ubuntu Desktop w trybie tekstowym”.
- ♦ **Zainstaluj w trybie OEM:** system zostanie zainstalowany na dysku komputera z wykorzystaniem pseudograficznego programu instalacyjnego. Odmienne niż w poprzednim przypadku będzie można zdefiniować użytkownika OEM i dostosować

system do potrzeb użytkownika końcowego. Następnie zostanie uruchomione polecenie, za którego pomocą będzie można usunąć użytkownika OEM i utworzyć standardowego użytkownika Ubuntu z przywilejami administratora systemu. Instalowanie systemu w wersji OEM omówiłem w punkcie pt. „Instalowanie Ubuntu w trybie OEM”.

- ♦ **Zainstaluj serwer:** system zostanie zainstalowany na dysku komputera z wykorzystaniem pseudograficznego programu instalacyjnego. Ta opcja została omówiona wcześniej w punkcie pt. „Instalacja serwera Ubuntu”.
- ♦ **Sprawdź CD pod kątem błędów odczytu:** zostanie sprawdzona spójność danych znajdujących się na płycie CD umieszczonej w napędzie, a także ich zawartość.
- ♦ **Napraw uszkodzony system:** zostanie wyświetlonych kilka informacji na temat języka, w którym mają być wyświetlane komunikaty oraz lokalizacji geograficznej, a następnie Ubuntu zostanie uruchomiony z płyty CD, dzięki czemu będzie można naprawić uszkodzone dane na dysku, zmodyfikować lub zainstalować ponownie program GRUB itp.



Jeżeli pojawiają się problemy z przejściem do trybu ratunkowego, wówczas można uruchomić system z płyty dołączonej do niniejszej książki, oferuje ona te same możliwości, ale dodatkowo dostępny jest graficzny interfejs.

- ♦ **Test pamięci:** zostanie wykonanych kilka testów sprawdzających pamięć systemową i podręczną. Aby uzyskać dostęp do menu konfiguracyjnego, należy wcisnąć klawisz z literą C, wówczas będzie można określić rodzaj testów. Po ich zakończeniu należy ponownie uruchomić komputer, wciskając klawisz *Escape*.
- ♦ **Uruchom z dysku twardego:** system zostanie uruchomiony z dysku komputera, z pominięciem zawartości płyty Ubuntu Alternate Install CD. Ta opcja jest używana, gdy przypadkiem w napędzie zostanie pozostawiona płyta, mimo że użytkownik nie planował instalowania systemu czy przeprowadzenia testów.

Użycie każdej z opcji menu *Zainstaluj w trybie tekstowym*, *Zainstaluj w trybie OEM* czy *Zainstaluj serwer* spowoduje zainstalowanie Ubuntu na dysku komputera. Poniżej omówiłem szczegółowo wszystkie i podkreśliłem różnice pomiędzy instalacją w trybie tekstowym (uruchamianą w tym przypadku) a instalacją z wykorzystaniem pseudograficznego instalatora, który został omówiony w punkcie pt. „Instalacja serwera Ubuntu”.

Instalowanie systemu Ubuntu Desktop w trybie tekstowym

Jeżeli użytkownik wybierze taką opcję z menu programu instalacyjnego, wówczas będzie mógł zainstalować system w trybie tekstowym, a takie rozwiązanie może być wskazane w kilku przypadkach. Oto one.

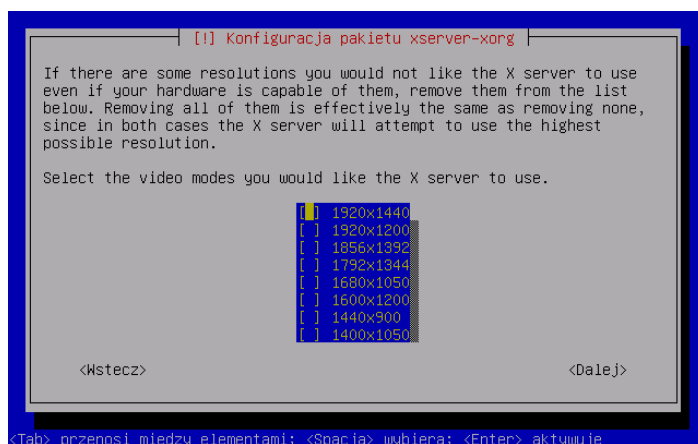
- ♦ Wystąpiły problemy z uruchomieniem sesji live CD z powodu konfliktów lub kłopotów z kartą graficzną.
- ♦ Użytkownik chce użyć woluminów logicznych zamiast zwykłych partycji.

- ♦ Użytkownik chce zainstalować system, ale chce też zmienić domyślne miejsce instalowania programu rozruchowego GRUB (czyli sektor rozruchowy pierwszego dysku).

Instalowanie systemu w trybie tekstowym polega na przejściu przez takie same etapy, które opisano wcześniej w punkcie „Instalacja serwera Ubuntu”. Zostanie zainstalowany nieco inny zestaw oprogramowania, choć pakiety GNOME, X Window i inne narzędzia graficzne pojawią się w systemie. Jedyna różnica między takim sposobem instalacji a omówionym wcześniej polega na wyświetleniu planszy (tuż przed wyjęciem płyty CD z napędu i ponownym uruchomieniu komputera), z której będzie można wybrać rozdzielczość, w jakiej monitor wyświetli informacje (patrz rysunek 3.41).

Rysunek 3.41.

*Określanie
rozdzielczości
monitora*



Kiedy ten obraz pojawi się na ekranie, należy wskazać rozdzielczość, w której ma być wyświetlane środowisko graficzne GNOME. Aby wybrać odpowiednią pozycję, należy podświetlić ją za pomocą klawiszy ze strzałkami i wcisnąć klawisz spacji. Spowoduje to jednocześnie usunięcie zaznaczenia poprzednio wybranej wartości. Domyślnie zaznaczone są pozycje 1024×768, 800×600 i 640×480, ponieważ większość współczesnych komputerów wyświetla obraz właśnie o takich parametrach. Jeżeli komputer może wyświetlać wyższe rozdzielczości, należy je zaznaczyć, ich lista jest określana na podstawie testów przeprowadzonych na karcie graficznej. System spróbuje użyć najwyższej wybranej wartości, jeżeli wystąpią jakieś problemy, wówczas skorzysta z niższych ustawień. Aby przejść dalej, należy wcisnąć klawisz *Enter*.

Następnie trzeba usunąć z napędu płytę CD, po uruchomieniu komputera dostępne będzie Ubuntu ze środowiskiem graficznym.

Instalowanie Ubuntu w trybie OEM

Po wybraniu z menu instalacyjnego tej opcji będzie można zainstalować Ubuntu z wykorzystaniem trybu tekstowego, ale w systemie zostanie utworzone tymczasowe konto użytkownika OEM. Umożliwia to dodatkową konfigurację systemu, zainstalowanie oprogramowania, a także w razie potrzeby usunięcie pewnych pakietów. Tak przygotowany system można następnie redystrybuować do końcowych użytkowników. Aby przekształcić

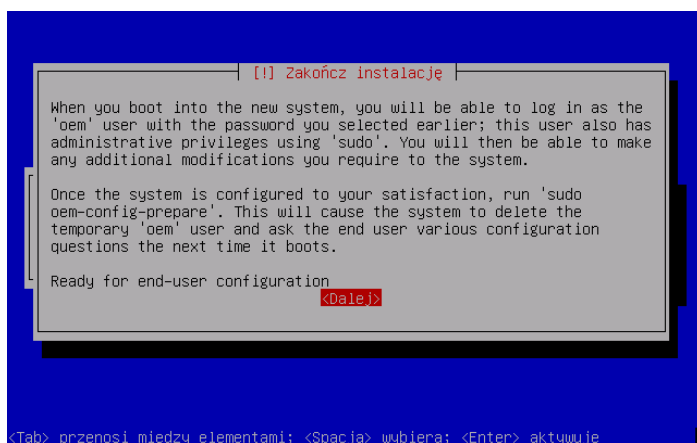
ten system w standardowe Ubuntu, z kontem użytkownika utworzonym przez odbiorcę końcowego, należy wykonać polecenie `sudo oem-prepare`, co spowoduje usunięcie konta użytkownika OEM. Po ponownym uruchomieniu systemu pojawi się możliwość utworzenia nowego konta użytkownika i wprowadzenia kolejnych zmian w konfiguracji systemu.

Podobnie jak poprzednio, pseudograficzny program instalacyjny wyświetli taki sam zestaw ekranów, jak ten omówiony w punkcie „Instalowanie systemu Ubuntu Desktop w trybie tekstowym”. Ponieważ podczas instalacji w trybie OEM tworzony jest domyślny użytkownik, nie pojawi się pytanie o nazwę użytkownika, a tylko o hasło. Podobnie jak w omawianym poprzednio przypadku, zostanie wyświetlone pytanie o rozdzielczość ekranu.

Tuż przed ostatnim, podsumowującym ekranem instalacyjnym zostaną wyświetlone informacje na temat korzystania z konta użytkownika OEM (patrz rysunek 3.42).

Rysunek 3.42.

*Podsumowanie
procesu instalacji
w trybie OEM*



Po wyświetleniu powyższego okna procedura instalacji systemu będzie kontynuowana; użytkownik zostanie poproszony o usunięcie płyty z napędu i ponowne uruchomienie komputera. Po zalogowaniu się do systemu graficznego będzie można dokończyć instalację systemu przed rozdyskrebowaniem komputerów do użytkowników końcowych.

Instalowanie serwera z płyty Alternate Install CD

Wybranie z menu tej opcji umożliwia zainstalowanie serwera za pomocą programu instalacyjnego korzystającego z trybu tekstowego i dostarczającego dodatkowe możliwości związane z tym trybem. Sam proces przebiega tak samo jak w przypadku instalacji z płyty Server Install CD, co opisano w punkcie „Instalacja serwera Ubuntu”.

Podsumowanie

W rozdziale poruszyłem wiele zagadnień, wyjaśniłem niektóre tematy, które pomogą z sukcesem zainstalować Ubuntu jako alternatywny lub podstawowy system w komputerze. Omówiłem specjalne opcje rodem z Debiana dostępne w programie instalacyjnym. W poruszonych tematach znalazły się takie jak: proces uruchamiania systemu, samodzielne

partycjonowanie dysków i kilka innych specjalnych opcji instalacyjnych. Niezależnie od rodzaju instalowanego systemu, po przeczytaniu niniejszego rozdziału użytkownik będzie mógł zainstalować dowolną jego wersję.

W rozdziale 4. omówię podstawowe koncepcje Linuksa, które mogą być interesujące dla początkujących użytkowników tego systemu. Natomiast w rozdziałach 5. i 6. opiszę korzystanie z Ubuntu przy użyciu interfejsu graficznego i linii poleceń, co jest dobrym wstępem dla osób, które znają już Linuksa i chcą pracować dalej z systemem.

Część II

Korzystanie z Ubuntu

W tej części:

Rozdział 4. Podstawowe założenia systemu Linux

Rozdział 5. Środowisko GNOME

Rozdział 6. Korzystanie z linii poleceń

Rozdział 7. Praca z plikami tekstowymi w Ubuntu

Rozdział 8. Obsługa poczty elektronicznej za pomocą programu Evolution

Rozdział 9. Surfowanie z Firefoksem

Rozdział 10. Tworzenie i publikowanie dokumentów

Rozdział 11. Pozostałe składniki pakietu Office: arkusz kalkulacyjny Calc i program do tworzenia prezentacji

Rozdział 12. Praca z grafiką

Rozdział 13. Multimedia

Rozdział 14. Zagrasz?

Rozdział 15. Łączenie się z innymi systemami

Rozdział 16. Przesyłanie i udostępnianie plików w Ubuntu

Rozdział 17. Urządzenia elektroniczne w Ubuntu

Rozdział 18. Tworzenie oprogramowania w Ubuntu

Rozdział 4.

Podstawowe założenia systemu Linux

W tym rozdziale:

- ♦ Informacje o plikach i katalogach
- ♦ Informacje o dyskach i partycjach
- ♦ System plików w Linuksie
- ♦ Podstawowe informacje o użytkownikach i grupach
- ♦ Uprawnienia użytkowników w Ubuntu

W niniejszym rozdziale omówiłem podstawowe założenia systemu Linux ze szczególnym uwzględnieniem sposobów, w jakie są przechowywane i organizowane dane, a także kontroli ich udostępniania oraz wykonywania poleceń, dla których konieczne są specjalne uprawnienia. Gdy pominiemy fragmenty zawierające informacje, jak rozwiązano w Ubuntu kwestię uprawnień użytkowników, okaże się, że pozostałe części tego rozdziału dotyczą większości systemów typu Linux.

Niniejszy rozdział nie grzeszy ostrożnością. Choć brzmi to nieprawdopodobnie, to jednak zanim zdecydowałem się na studiowanie informatyki na uniwersytecie, nigdy wcześniej nie używałem komputera (o tyle jestem usprawiedliwiony, że było to dawno temu, u początku historii komputerów osobistych, nim pojawiły się DOS, Microsoft Windows, Apple Macintosh itp.). Wciąż mam w pamięci pierwsze zajęcia w klasie, kiedy prowadzący powiedział: „Po napisaniu programu proszę zapisać go do pliku” i myśl w głowie: „O czym on mówi?”.

Dlatego ten rozdział zawiera podrozdziały, takie jak „Praca z plikami i katalogami”, choć dla części użytkowników zamieszczanie tak podstawowych zagadnień może wydawać się obraźliwe. Czytelnicy, dla których są to sprawy oczywiste, mogą je pominąć, ale ci, którzy właśnie przynieśli Ubuntu do domu ze sklepu czy od znajomych, po przeczytaniu mogą z zadowoleniem stwierdzić: „Ach, teraz rozumiem!”.

Praca z plikami i katalogami

Plik to nic innego jak pojedynczy zestaw informacji, z którego korzystają system operacyjny oraz programy. Pliki to pojemniki na różnego rodzaju dane, mogą zawierać listy do mamy, zwolnienie lekarskie, kopię jednej z ulubionych piosenek (legalną oczywiście), cyfrowe zdjęcie czy informacje używane w arkuszu kalkulacyjnym do sprawdzania stanu swoich finansów. Zastosowanie takiej koncepcji pliku jako pojemnika na dane umożliwia „systemowi operacyjnemu oraz odpowiednim programom, łatwe wyszukiwanie plików i korzystanie z nich. Dlatego właśnie menu *Plik*, które jest obecne w większości używanych dzisiaj programów graficznych, zawiera polecenia umożliwiające otwieranie istniejących plików, tworzenie nowych oraz zapisywanie zmian wprowadzonych podczas pracy z nimi.

Pliki są niezbędne — system operacyjny musi mieć możliwość identyfikowania oraz dostępu do danych, programy potrzebują plików, aby działać, korzystają też z plików konfiguracyjnych itd. Podobnie ma się rzecz z komputerem, który po uruchomieniu musi znaleźć system operacyjny oraz opcje konfiguracyjne niezbędne do jego uruchomienia. Chociaż potrzeba coraz więcej plików, które używane są do różnorodnych celów i tworzone przez różnych użytkowników, zapoznanie się z mechanizmem ich organizowania nie zajmie dużo czasu, a będzie to dobry wstęp do omówienia katalogów. Katalogi to z kolei proste pojemniki na zbiory powiązanych plików; mogą one również zawierać inne katalogi (czyli podkatalogi). Jest to hierarchiczny układ plików (zwany czasem strukturą drzewiastą), w którym położenie każdego pliku opisywane jest za pomocą listy katalogów, które kolejno go zawierają. Katalogiem, od którego należy rozpocząć poszukiwania w Linuksie, jest tzw. **katalog główny**; to inaczej punkt startowy przy opisywaniu, jak odnaleźć dany plik, który jest przechowywany lokalnie (na danym komputerze).

Zwykle układ plików i katalogów porównuje się do kartotek w szufladach — aby odnaleźć jakiegokolwiek z nich, należy przejść przez następujący proces: trzeba otworzyć odpowiednią szufladę, następnie odnaleźć odpowiednio opisaną teczkę (np. imieniem i nazwiskiem) i sprawdzić jej zawartość w poszukiwaniu dokumentu z wynikami badań. Analogicznie, w przypadku komputera proces wyszukiwania może wyglądać następująco: najpierw należy przejść do głównego systemu plików (/), następnie do katalogu z plikami użytkowników (*home*), później do katalogu *wvh* (nazwa użytkownika) i odszukać plik o nazwie *wyniki_badania.txt*.

W systemach Linux używa się znaku / do oddzielenia od siebie kolejnych elementów, dlatego często pojawiają się zapisy typu „przejdź do pliku */home/wvh/wyniki_badania.txt*”. Seria kolejnych katalogów prowadzących do danego pliku często nazywana jest „ścieżką do pliku”.

Systemy linuksowe wyposażone w graficzny interfejs użytkownika (GUI), taki jak GNOME, oferują również graficzne sposoby nawigowania pomiędzy katalogami, zwykle poprzez klikanie, co powoduje wyświetlenie okna z ich zawartością w postaci ikon, które odpowiadają znajdującym się tam plikom. W każdym Linuksie w terminalu można użyć polecenia *cd* (ang. *change directory*), które umożliwia przechodzenie pomiędzy kolejnymi lokalizacjami (więcej informacji na ten temat można znaleźć w rozdziale 6., „Korzystanie

z linii poleceń”. Można przejść do innego katalogu, podając po kolei katalogi lub też od razu określić miejsce docelowe. Innymi słowy, poniższe sekwencje poleceń dadzą ten sam efekt:

```
cd /  
cd home  
cd wvh  
  
i  
  
cd /home/wvh
```

Linux oferuje też wiele różnorodnych narzędzi do wyświetlania zawartości katalogów oraz szczegółowych informacji na temat mieszczących się tam plików oraz katalogów. Więcej informacji na ten temat podaję w rozdziale 6., teraz więcej uwagi poświęcę opisowi organizacji plików i katalogów w systemach Linux.

Katalogi systemu Linux

Wszystkie systemy linuksowe posiadają standardowy zestaw podstawowych katalogów. Poniższe katalogi są używane do przechowywania programów i plików konfiguracyjnych, bibliotek i tymczasowych plików tworzonych i używanych podczas uruchamiania systemu i działania różnych programów.

- ♦ */*: to katalog znajdujący się na samej górze struktury plików, jest niezbędny, ponieważ w nim są umieszczane kolejne katalogi.
- ♦ */bin*: w tym katalogu przechowywane są podstawowe aplikacje systemu Linux.
- ♦ */dev*: w tym katalogu przechowywane są pliki specjalne, znane jako węzły urządzeń, dzięki którym uzyskuje się dostęp do urządzeń dostępnych w systemie.
- ♦ */etc*: w tym katalogu przechowywane są informacje o konfiguracji systemu, pliki opisujące sekwencje programów uruchamianych podczas procesu rozruchowego systemu, a także pliki konfiguracyjne dla niektórych programów uruchamianych przez system.
- ♦ */lib*: w tym katalogu przechowywane są biblioteki funkcji, które mogą być wywołane przez inne programy.
- ♦ */proc*: w tym katalogu przechowywane są informacje na temat procesów wykorzystywane przez jądro systemu.
- ♦ */sbin*: w tym katalogu przechowywane są programy, które uruchamiać może tylko użytkownik z uprawnieniami administratora.
- ♦ */sys*: w tym katalogu przechowywane są informacje na temat statusu urządzeń oraz ich interfejsów.
- ♦ */tmp*: w tym katalogu przechowywane są tymczasowe pliki tworzone przez różne programy podczas działania systemu.

Te katalogi można znaleźć w większości standardowych systemów Linux, bez względu na rodzaj używanej dystrybucji oraz rozmiar dysku.

Inne katalogi systemu Linux

W zależności od ilości zainstalowanych plików oraz ich układu, w systemie prawdopodobnie trzeba będzie utworzyć kilka kolejnych katalogów. Oto niektóre inne katalogi systemu Linux.

- ♦ */home*: w katalogu tym przechowywane są podkatalogi z plikami użytkowników. I tak większość plików (jeżeli nie wszystkie), których właścicielem jest *wvh*, jest przechowywana w katalogu */home/wvh* i znajdujących się w nim podkatalogach. Katalogi używane przez indywidualnych użytkowników, do przechowywania ich osobistych plików, noszą nazwę „katalogów domowych” — nazwa prowadzi wprost do katalogu */home*, co w języku angielskim oznacza dom. W starszych systemach uniksowych katalogi użytkowników były przechowywane w katalogu */usr*, teraz jednak, dla uproszczenia umieszcza się je w lokalizacji */home*.
- ♦ */opt*: w tym katalogu zwykle instaluje się oprogramowanie firm zewnętrznych. Nazwa nawiązuje do przechowywanej zawartości, „optional” i znaczy tyle, co dodatkowe (w tym znaczeniu niesystemowe); oprogramowanie to może różnić się, w zależności od komputera.
- ♦ */usr*: znajdują się tam pliki, z których użytkownicy korzystają podczas pracy w systemie Linux. W katalogu */usr/bin* znajdują się aplikacje, do których uruchomienia potrzebne będą biblioteki umieszczone w */usr/lib* itp.
- ♦ */var*: w katalogu tym znajdują się inne podkatalogi z różnorodną zawartością, np. w katalogu */var/log* zapisywane są dzienniki systemowe. Dzienniki te są tworzone podczas działania systemu i z czasem mogą się bardzo rozrosnąć.

Jak widać, Linux oferuje rozbudowaną, hierarchiczną strukturę katalogów i plików zorganizowanych w taki sposób, aby ułatwić odnajdywanie plików danego typu, np. plików wykonywalnych, bibliotek, plików konfiguracyjnych, zawierających informacje itd. Teraz podam kilka informacji o tym, w jaki sposób w Linuksie wykorzystywane są różne rodzaje urządzeń pamięci masowej i jak pobiera się z nich dane.



Wymienione i opisane powyżej katalogi można uznać za standardowe dla większości systemów linuksowych, ale ich zawartość, zależnie od dystrybucji, może nieco się różnić. Projekt Linux Standard Base (LSB) został powołany do ustandaryzowania katalogów i ich zawartości dla różnych dystrybucji Linuksa. Więcej informacji na jego temat można znaleźć na stronie internetowej www.linuxbase.org.

Wprowadzenie do systemu plików w Linuksie

Omówione wcześniej katalogi to typowy sposób organizowania plików i katalogów niezbędnych do uruchomienia systemu, działania podstawowych aplikacji itd. Warto zwrócić uwagę, że ścieżka dostępu dla wszystkich rozpoczyna się od znaku */*. To jednak nie oznacza, że wszystkie są zlokalizowane na tym samym dysku twardym czy innym urządzeniu pamięci masowej. Odmienne niż w systemach Windows, w których każdemu dyskowi lub partycji przypisana jest niepowtarzająca się litera alfabetu (*C:*, *D:* itd.), czy w systemach

Mac OS X, w których każdy dysk lub partycja przedstawiane są jako osobne wolumeny na pulpicie, w systemach linuksowych korzysta się z „jednolitej przestrzeni nazw” dla wszystkich urządzeń pamięci masowej oraz partycji, do których system ma dostęp.

W tym podrozdziale omówię metody zarządzania i uzyskiwania dostępu do dysków przez system, powiem, czym są partycje oraz jak łączy się je w strukturę katalogów, do których użytkownik ma dostęp. Struktura plików i katalogów w Linuksie ogólnie powiązana jest z systemem plików, chociaż, zgodnie z informacjami podanymi poniżej, pojęcie „system plików” jest często używane do określania sposobu organizacji danych na określonej partycji.

Dyski, partycje i punkty montowania

Głównym zadaniem systemów komputerowych jest przechowywanie, uzyskiwanie, obrabianie i zapisywanie informacji. Pracując również jako administrator systemów komputerowych, często jestem zaskakiwany przez znajomych, znających się na zagadnieniach komputerowych, takimi zdaniami: „Twój komputer przenośny jest taki powolny. Dlaczego go nie rozbudujesz?”. Gdybym mógł pisać szybciej, niż byłby to w stanie obsłużyć używany laptop z procesorem Pentium 1,4 GHz, wówczas mógłbym występować w gabinecie osobliwości Ripleya i podróżować z cyrkiem po świecie, zamiast ślęczeć przed monitorem. Z drugiej jednak strony, gdyby zapisanie rozdziału miało zajmować pięć minut lub gdyby okazało się, że zabrakło miejsca na zapisanie dokonanych zmian, wówczas zapewne komputer zakończyłby żywot w rzece, a na jego miejscu stanąłby zupełnie nowy, bardziej błyszczący i współdziałający z Linuksem laptop. W mojej pracy dużo ważniejsze od prędkości procesora są szybki zapis i odczyt danych.

Każdy system komputerowy obejmuje jakieś urządzenia pamięci masowej, takie jak dyski, napędy DVD i CD, karty CompactFlash itp., na których przechowywane są system operacyjny oraz programy, one też dostarczają przestrzeń dla plików tworzonych przez użytkownika. Jedno z moich ulubionych zdań na temat komputerów przeczytałem w podręczniku obsługi komputera Hewlett-Packard z systemem Unix (HP-UX), który dotarł do mnie w połowie lat 80.; brzmiało ono następująco:

„Na pustym dysku można szukać całą wieczność”.

W języku geeków oznacza to, że jeżeli dyskowi nie zostanie nadana jakaś struktura, system operacyjny nie będzie umiał odnaleźć tam żadnych plików, katalogów czy w ogóle czegokolwiek. Aby przechowywać dane i później je udostępniać, dysk musi być przygotowany w taki sposób, aby umożliwić systemowi operacyjnemu i różnym programom dostęp do nich, ich odczyt i zapis oraz inne operacje. Takie przygotowanie zwykle składa się z dwóch etapów. Oto one.

1. Podzielenie dysku na jedną lub więcej części, które będą odpowiednio identyfikowane przez system operacyjny. Każda z takich części nosi nazwę partycji.
2. Sformatowanie partycji, czyli utworzenie na niej systemu plików, dzięki czemu system będzie mógł uzyskać do niej dostęp oraz tworzyć pliki i katalogi.

Dyski partycjonuje się z kilku powodów, niektóre przedstawiam poniżej.

- ♦ Aby zredukować czas potrzebny na znalezienie określonej porcji danych na dysku. Po prostu przeszukanie mniejszej ilości przestrzeni zajmuje mniej czasu.
- ♦ Aby ograniczyć ilość danych, które mogą być uszkodzone lub utracone w przypadku awarii dysku lub partycji.
- ♦ Aby przyspieszyć zadania administracyjne, takie jak defragmentowanie, sprawdzanie spójności i — w razie potrzeby — naprawy.
- ♦ Aby uprościć czynności administracyjne, takie jak wykonywanie kopii zapasowych. Łatwiej wykonać kopię partycji, która w całości zmieści się na taśmie czy innym nośniku danych, ponieważ nie ma wówczas potrzeby interwencji operatora kopii (np. zmiany taśmy czy płyty). Większa ilość partycji umożliwia również instalowanie systemu plików i programów na innych partycjach, niż te, na których przechowywane są dane użytkownika. Dzięki temu relatywnie często można wykonywać kopię danych bez zapisywania dużej ilości plików wykonywalnych, plików systemowych i innych, które nie uległy zmianie itp.

Jak już wspominałem, pojęcie „system plików” często potocznie używane jest na określenie całej struktury plików i katalogów składających się na system Linux. Tutaj jednak „system plików” używany jest w swoim podstawowym znaczeniu, czyli określa sposób, w jaki formatowane są partycje, aby umożliwić przechowywanie i używanie danych.

Po sformatowaniu partycji i utworzeniu na niej systemu plików trzeba jeszcze sprawić, by stała się dostępna dla systemu. Ta operacja nosi nazwę **montowania** systemu plików, co oznacza włączenie systemu plików do którejś z części struktury plików, dzięki czemu można korzystać z danego urządzenia do przechowywania danych. W Linuksie system plików montowany jest w katalogach i staje się częścią przestrzeni nazw, co wyjaśnię dokładniej w punkcie „Montowanie systemu plików”. Każdy katalog w Linuksie może być punktem montowania systemu plików, wiele omówionych wcześniej katalogów faktycznie jest takimi punktami dla oddzielnych systemów plików, których zawartość jest przechowywana właśnie w tych katalogach. Standardowy katalog linuksowy `/mnt` zawiera specjalne podkatalogi, które są stosowane jako punkty montowania dla używanych od czasu do czasu systemów plików.

Lokalne systemy plików: standardowy i z kroniką

Lokalny system plików umieszczony jest na urządzeniach pamięci dyskowej, które są fizycznie podłączone do komputera. Dzięki takiemu bezpośredniemu podłączeniu dostęp do nich jest stosunkowo szybki. Z drugiej strony, lokalne systemy plików są użyteczne jedynie wówczas, gdy użytkownik, który potrzebuje dostępu do danych, może podłączyć się do komputera, na jakim są one przechowywane. Dzięki dzisiejszym środowiskom sieciowym podłączenie się do komputera, na którym przechowywane są dane, nie stanowi kłopotu, wystarczy uruchomić SSH (ang. *Secure SHell*) lub telnet i połączyć się odpowiednim komputerem (warunek jest jeden, sieć musi działać prawidłowo).

Najczęściej używanym systemem plików w Linuksie jest ext2 (*Extended Filesystem*, wersja 2). Przyjęto określenie standardowy, ponieważ jest on wykorzystywany w większości podstawowych (*vanilla*) systemów. Podczas aktualizowania pliku zmiany są wprowadzane bezpośrednio w systemie plików. W przypadku awarii systemu w momencie

zapisu danych mogą one nie znaleźć się na dysku. Podczas ponownego uruchamiania systemu zostanie przeprowadzone sprawdzenie spójności systemu plików. Więcej informacji na temat rozpoznawania systemu plików w Linuksie można znaleźć w punkcie „Automatyczne montowanie systemów plików podczas uruchamiania komputera”). Narzędzie używane do sprawdzania spójności systemu plików ext2 to *fsck* (ang. *file system consistency check*). *fsck* sprawdza całą strukturę danych, aby upewnić się, że wszystkie dane w systemie są odpowiednio powiązane z plikami i katalogami.

W przypadku dużych partycji program *fsck* może pracować dłuższą chwilę, co oznacza, że przez jakiś czas system może być niedostępny dla użytkowników. Aby skrócić czas uruchamiania systemu, eliminując potrzebę włączania *fsck* (lub też redukując ilość zadań, które program powinien wykonać), zaprojektowano nowszą wersję systemu plików wyposażoną dodatkowo w **kronikowanie**. W takich systemach wszelkie zmiany wprowadzone w plikach najpierw zapisywane są w specjalnej części systemu, zwanej kroniką (czasem używa się określenia dziennik), w postaci transakcji. Zmiany zapisane w kronice są następnie wprowadzane do systemu asynchronicznie i transakcyjnie. Zapis o wprowadzonych zmianach pozostaje w dzienniku dopóty, dopóki same zmiany nie zostaną z powodzeniem wprowadzone w systemie plików, po czym zostają usunięte.

Systemy plików z kroniką skracają czas przeładowania systemu, ponieważ system plików zachowuje spójność podczas wszelkich operacji na nim wykonywanych. Dlatego też uruchomienie programu *fsck* generalnie sprowadza się do wprowadzenia zmian zapisanych w dzienniku systemowym. Systemy plików z kroniką są też wydajniejsze od systemów jej pozbawionych, ponieważ w tym przypadku nie ma potrzeby czekania na wprowadzenie wszystkich zmian. Po zapisaniu zmian w kronice system może kontynuować pracę, bowiem może być uaktualniany asynchronicznie. Ponieważ dziennik jest przechowywany w odrębnej części systemu o wysokiej wydajności i specjalnym formacie, może być uaktualniany znacznie szybciej aniżeli system plików. Zwykle zmiany w pliku muszą być wprowadzone, zanim zostaną dokonane następne, dlatego przed modyfikacją pliku konieczne jest upewnienie się, że poprzednie zmiany zostały zastosowane.

W Linuksie najczęściej używanym systemem plików z kroniką jest ext3, oparty o wersję ext2, ale poszerzoną o możliwość stosowania kroniki. Ten typ systemu plików zyskał sobie popularność, ponieważ łatwo można do niego przekonwertować system ext2, dzięki czemu wszelkie zmiany, naprawy plików i analizy programów wykonane w ext2 będą wprowadzone w ext3. Inne popularne systemy plików z kroniką to: ReiserFS i Reiser4 (oba zaprojektowane przez Hansa Reisera i przyjaciół z namesys.com), JFS (oryginalnie stworzony przez IBM dla OS/2 i AIX, ale udostępniony jako projekt open source, <http://jfs.sourceforge.net/>) oraz XFS (oryginalnie napisany przez SGI dla IRIX-a i również opublikowany przez SGI jako oprogramowanie o otwartym kodzie, <http://oss.sgi.com/projects/xfs/>).

Oto próbką humoru ze strony podręcznika dla narzędzia *fsck* dla systemu plików XFS:

```
fsck.xfs(8)                                fsck.xfs(8)
NAZWA
fsck.xfs - nie rób nic - z sukcesem

STRESZCZENIE
fsck.xfs [ ...]
```

OPIS

fsck.xfs to nazwa programu startowego fsck(8) przeznaczonego do sprawdzania i naprawiania systemu plików XFS. XFS to system plików z kroniką i dokonuje napraw podczas montowania(8)
- jeżeli to konieczne, dlatego fsck.xfs ma a zerową liczbę wyjść.

PLIKI

/etc/fstab

ZOBACZ RÓWNIEŻ

fsck(8), fstab(5), xfs(5).

Cóż, to faktycznie skróci czas działania programu fsck i ponownego uruchomienia systemu!

Sieciowe systemy plików

Sieciowe systemy plików to systemy przechowywane na zdalnych komputerach (zwykle nazywanych serwerami plików), które mogą być montowane w komputerach jako lokalne systemy plików. Sieciowe systemy oferują użytkownikom swobodę dostępu do informacji z każdego komputera, na jakim mogą się zalogować i połączyć z zasobami sieciowymi.

Sieciowe systemy plików mają kilka przewag nad systemami lokalnymi.

- ♦ Zmniejszają ryzyko utraty dostępu do danych w przypadku awarii jednego komputera. Większość takich systemów umożliwia zalogowanie się z wielu różnych komputerów dokładnie w taki sam sposób.
- ♦ Oferują centralne położenie dla danych, które muszą być współdzielone przez użytkowników.
- ♦ Upraszczają dostęp do danych za pośrednictwem szybszych systemów. Załóżmy, że czytelnik musi napisać program do testowania procesora i wydajności pamięci lub też, że praca albo wynik badań zależą od intensywnych obliczeń przeprowadzanych przez procesor czy pamięć. Uruchomienie programu na szybszej czy wydajniejszej maszynie jest łatwiejsze, podobnie jak jej uruchomienie z sieciowego systemu plików.
- ♦ Oferują one również centralne administrowanie zadaniami, takimi jak wykonywanie kopii zapasowych.
- ♦ Systemy te stawiają na interoperacyjność i elastyczność. Można z nich korzystać, używając systemów, takich jak Linux, Microsoft Windows, Mac OS X itp. Dzięki temu najłatwiej korzystać ze sprzętu i oprogramowania najlepiej spełniającego wymogi użytkownika i wciąż mieć dostęp do zgromadzonych plików.

Dla Linuksa istnieje wiele różnorodnych rodzajów systemów plików sieciowych. Najpopularniejszy z nich to NFS (ang. *Network File System*) stworzony przez firmę Sun Microsystems, obsługiwany niemal w każdym systemie operacyjnym wywodzącym się z Uniksa. NFS jest obsługiwany przez Mac OS X, a klienty i serwery NFS są dostępne nawet w systemach Microsoft Windows. Innym powszechnie wykorzystywanym systemem jest format Microsoftu SMB (ang. *Server Message Block*), który został przemianowany na CIFS

(ang. *Common Internet File System*) w celu ukrycia jego własnościowych korzeni oraz faktu, że aby korzystać z niego w Linuksie i podobnych systemach, należało skorzystać z inżynierii wstecznej. SMB/CIFS jest obsługiwany (zarówno jako klient, jak i serwer) w systemach linuksowych dzięki pakietowi oprogramowania znanego pod nazwą Samba (www.samba.org). Klient Samby umożliwia z poziomu Linuksa zamontowanie i dostęp do systemu plików SMB/CIFS eksportowanych przez systemy Microsoft Windows. Z kolei serwery Samby umożliwiają eksportowanie systemu plików z Linuksa tak, aby można było z nich korzystać w różnych wersjach Windows. Linuksa można też połączyć z systemem plików Novell Netware, a to dzięki narzędziu Novell Core Protocol (www.novell.com/coolsolutions/feature/15350.html), a także z systemem Pre-OS X MacOS Apple przy użyciu netatalk i afpfs (<http://netatalk.sourceforge.net/> lub www.anders.com/projects/netatalk/).

Praca z partycjami i systemami plików

Ubuntu, podobnie jak większość systemów linuksowych, automatycznie identyfikuje dyski w systemie i proponuje pewien schemat podzielenia ich na partycje podczas procesu instalowania systemu operacyjnego. Podobnie rzecz ma się podczas podłączania nowego dysku do systemu, kiedy to Ubuntu wyświetla komunikat z pytaniem o szczegóły jego konfiguracji. Partycjonowanie podczas procesu instalowania omówiłem w rozdziale 2., „Instalowanie Ubuntu”, a korzystanie z graficznego narzędzia do partycjonowania dysków opisałem szczegółowo w rozdziale 23., „Dodawanie sprzętu i przyłączanie urządzeń peryferyjnych”.

Nawet po utworzeniu partycji lub podłączeniu dysku, na którym są one już utworzone, nie będzie do nich dostępu, dopóki nie zostaną gdzieś zamontowane. W tym podrozdziale omówię montowanie partycji z użyciem polecenia `mount`, opiszę najpopularniejsze opcje dostępne dla tej komendy, a także metody montowania wybranych dysków podczas uruchamiania systemu, co można osiągnąć, zmieniając wpisy w pliku konfiguracyjnym `/etc/fstab`.

Montowanie systemu plików

Po podłączeniu dysku zawsze można sprawdzić, jakie systemy plików się na nim znajdują, wystarczy skorzystać z polecenia `fdisk` z opcją `-l` (ang. *list*), tak jak w poniższym przykładzie:

```
$ sudo fdisk -l
```

```
Disk /dev/sda: 250.0 GB, 250059350016 bytes
255 heads, 63 sectors/track, 30401 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1		1	1306	10490413+	83	Linux
/dev/sda2		1307	2090	6297480	82	Linux swap
/dev/sda3	*	2091	2122	257040	83	Linux
/dev/sda4		2123	30399	227135002+	f	W95 Ext'd (LBA)
/dev/sda5		2123	2645	4200966	83	Linux
/dev/sda6		2646	6562	31463271	83	Linux

```

/dev/sda7          6563          10478      31455238+ 83 Linux
/dev/sda8          10479          30399      160015401 83 Linux

```

```

Disk /dev/hde: 250.0 GB, 250059350016 bytes
255 heads, 63 sectors/track, 30401 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

```

```

      Device Boot      Start         End      Blocks   Id  System
/dev/hde1                1         30401     244196001   83   Linux

```

```

Disk /dev/hdf: 250.0 GB, 250059350016 bytes
255 heads, 63 sectors/track, 30401 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

```

```

      Device Boot      Start         End      Blocks   Id  System
/dev/hdf1                1         30401     244196001   83   Linux

```

Domyślnie polecenie `fdisk` wyświetla listę wszystkich partycji na wszystkich dyskach, które system mógł wykryć. Polecenia `fdisk` można również użyć do wyświetlenia listy partycji znajdujących się na określonym urządzeniu:

```

$ sudo fdisk -l /dev/sda
Password:

```

```

Disk /dev/sda: 250.0 GB, 250059350016 bytes
255 heads, 63 sectors/track, 30401 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

```

```

      Device Boot      Start         End      Blocks   Id  System
/dev/sda1                1          1306     10490413+ 83   Linux
/dev/sda2                1307         2090      6297480    82   Linux swap
/dev/sda3      *         2091         2122       257040    83   Linux
/dev/sda4                2123         30399     227135002+  f   W95 Ext'd (LBA)
/dev/sda5                2123         2645       4200966    83   Linux
/dev/sda6                2646         6562      31463271    83   Linux
/dev/sda7                6563         10478      31455238+  83   Linux
/dev/sda8                10479         30399     160015401    83   Linux

```

Należy zwrócić uwagę, że podstawowym argumentem dla tego polecenia jest nazwa dysku, którego dotyczy zapytanie, a nie nazwa określonej partycji. Informacje podobnej treści, choć w nieco mniej użytecznej formie, można uzyskać po przejrzeniu pliku `/proc/partitions`, który jest wykorzystywany przez system operacyjny do śledzenia wszystkich partycji i dysków aktualnie używanych. Oto zawartość tego pliku zaczerpnięta z systemu, z którego korzystano w poprzednim przykładzie:

```

$ cat /proc/partitions
major minor #blocks name
 8      0 244198584 sda
 8      1 10490413 sda1
 8      2 6297480 sda2
 8      3 257040 sda3
 8      4 1 sda4
 8      5 4200966 sda5
 8      6 31463271 sda6
 8      7 31455238 sda7
 8      8 160015401 sda8
33      0 244198584 hde

```

```
33    1 244196001 hdel
33    64 244198584 hdf
33    65 244196001 hdf1
```

Po zidentyfikowaniu partycji na nowym dysku i podjęciu decyzji, które z nich mają zostać użyte, zamontowanie ich za pomocą polecenia `mount` jest już całkiem proste. Zawsze trzeba zamontować system plików, którego nie ma w pliku `/etc/fstab` (więcej szczegółów na ten temat podaję w kolejnym punkcie).

Oto przykład: poniższe polecenie spowoduje zamontowanie partycji `/dev/sda5` w punkcie montowania `/mnt/tmp` (punkt ten musi istnieć już w systemie):

```
$ sudo mount /dev/sda1 /mnt/tmp
```

W przypadku montowania partycji zawierającej system plików `ext2` lub inny, który może zostać automatycznie rozpoznany przez system operacyjny, nie ma potrzeby określania jego typu. Jeżeli jest to jakiś niestandardowy format, wówczas należy zidentyfikować go za pomocą opcji `-t`, po której musi pojawić się nazwa systemu plików; w poniższym przykładzie montowany jest system plików `XFS` na dysku `/dev/sda1` w punkcie montowania `/mnt/xfs`:

```
$ sudo mount -t xfs /dev/sda1 /mnt/tmp
```

Opcja `-t` wymusza wyszukanie w pierwszej kolejności określonego systemu plików, czyli `/sbin/mount.typ`, gdzie `typ` jest nazwą tego systemu. Tego typu polecenia związane z systemami plików są zwykle instalowane wraz z innymi narzędziami do administrowania za pomocą menedżera pakietów `Synaptic` lub `apt-get`, więcej szczegółów na ten temat znajduje się w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Systemy plików, dla których można użyć opcji `-t`, to: `dfs`, `affs`, `autofs`, `coda`, `coherent`, `cramfs`, `devpts`, `efs`, `ext`, `ext2`, `ext3`, `hfs`, `hpfs`, `iso9660`, `jfs`, `minix`, `msdos`, `ncpfs`, `nfs`, `ntfs`, `proc`, `qnx4`, `ramfs`, `reiserfs`, `romfs`, `smbfs`, `sysv`, `tmpfs`, `udf`, `ufs`, `umsdos`, `usbfs`, `vfat`, `xenix`, `xfs` i `xiafs`.

Niektóre systemy plików, takie jak `NFS` czy `SMB/CIFS`, wymagają dodatkowych opcji. Ponieważ są one specyficzne dla każdego rodzaju systemu, nie będę nimi zanudzać czytelników. Skupię się na opcjach dostępnych dla systemu `SMB/CIFS`, ponieważ jest to prawdopodobnie najczęściej używany sieciowy system plików, z którym będzie miał do czynienia użytkownik. Niektóre spośród innych, często używanych w pliku konfiguracyjnym `filesystem/mount` opcji montowania omawiam poniżej.

Podczas montowania sieciowego systemu plików `SMB/CIFS` należy określić nazwę użytkownika `Windows`, który ma mieć dostęp do systemu, hasło dostępu do `Windows`, a także tożsamość użytkownika `Linuxa`, który jest właścicielem plików, dzięki czemu będzie można dokonywać zmian w systemie plików. W poniższym przykładzie widoczne są wspomniane dane konieczne dla polecenia `mount`:

```
$ sudo mount -t smbfs -o username=vwj,password=moje_haslo,uid=vwj \
//192.168.6.66/share /mnt/terastation
```

To polecenie spowoduje zamontowanie systemu plików `smbfs` o nazwie `share` na komputerze `192.168.6.66` w katalogu `/mnt/terastation`, wykorzystana zostanie nazwa użytkownika

Windows wvh, którego hasło to *moje_haslo*. Właścicielem zamontowanego systemu plików będzie użytkownik wvh, co oznacza, że będzie mógł zapisywać pliki w systemie, który nie jest zabezpieczony przez mechanizmy Windows ACL lub podobne.

Automatyczne montowanie systemów plików podczas uruchamiania komputera

Podczas uruchamiania Ubuntu program rozruchowy najpierw identyfikuje używany system plików, na jego podstawie są konfigurowane parametry jądra. W starszych systemach używano programu rozruchowego o nazwie LiLo (*Linux Loader*), w Ubuntu i w większości współczesnych systemów stosuje się program o nazwie GRUB (ang. *Grand Unified Boot Loader*).

W czasie uruchamiania systemu, główny system plików jest montowany w trybie tylko do odczytu, dzięki czemu możliwe jest sprawdzenie jego spójności. Po wykonaniu tej operacji system jest montowany w trybie do odczytu i zapisu, po czym następuje sprawdzenie spójności innych używanych systemów plików. Lista tych, które są dostępne, znajduje się w pliku */etc/fstab* (*File System Table*).

Każda linia w pliku */etc/fstab* zawiera informacje na temat jednego systemu plików spośród tych, które są dostępne w systemie. Aby dodać nowy system plików, który będzie automatycznie montowany podczas uruchamiania komputera, lub dostosować działanie systemu operacyjnego w sytuacji, gdy podłączane jest kolejne urządzenie z systemem plików, wystarczy do pliku */etc/fstab* wprowadzić odpowiednie wpisy. Oto zawartość przykładowego pliku */etc/fstab* zaczerpnięta z mojego komputera (dla każdego komputera plik ten może wyglądać inaczej):

```
$ cat /etc/fstab
LABEL=/      /          ext3        noatime,acl,user_xattr 1 1
/dev/sda3    /boot      ext3        acl,user_xattr      1 2
/dev/sda8    /home      ext3        noatime,acl,user_xattr 1 2
/dev/sda5    /tmp       ext3        acl,user_xattr      1 2
/dev/sda6    /usr       ext3        noatime,acl,user_xattr 1 2
/dev/sda7    /usr/local ext3        noatime,acl,user_xattr 1 2
/dev/sda2    swap       swap        sw                  0 0
/dev/hde1    /opt2      ext3        noatime,acl,user_xattr 0 0
devpts       /dev/pts   devpts      mode=0620,gid=5     0 0
proc         /proc      proc        defaults            0 0
sysfs        /sys       sysfs       noauto              0 0
/dev/hdd     /media/cdrom0 udf,iso9660 user,noauto          0 0
/dev/fd0     /media/floppy0 auto        rw,user,noauto      0 0
```

Oto opis zawartości poszczególnych pól w pliku */etc/fstab*.

- ♦ Pierwsze pole to nazwa urządzenia lub zdalnego systemu plików, który ma zostać zamontowany. Jest to zwykle plik urządzenia dla partycji, która ma zostać zamontowana, ale może to być też wpis w rodzaju *hostname:directory* dla sieciowych systemów plików, takich jak NFS. Systemy plików *ext2* i *ext3* mogą być również identyfikowane według nazwy, która została im przypisana jako etykieta podczas tworzenia. Przykładowo wpis *LABEL=/* w powyższym przykładzie pliku */etc/fstab* może zostać zastąpiony wpisem */dev/sda1*, ponieważ jest to partycja

dysku, na której znajduje się główny system plików. Używanie etykiet jest jednak bardziej elastyczne od stosowania nazw plików urządzeń, ponieważ nazwy przypisane do konkretnych partycji mogą się zmieniać, jeżeli dany dysk zostanie przeniesiony do innego systemu lub też do systemu zostanie dodany dodatkowy dysk, na którym znajduje się już układ partycji.

- ♦ Drugie pole wskazuje katalog, w którym jest montowany dany system plików. W przypadku specjalnych partycji, które nie powinny zostać zamontowane (np. partycja wymiany), w tym polu powinien znajdować się zapis *none*.
- ♦ W trzecim polu identyfikuje się system plików. Zwykle znajdują się tu wpisy, takie jak *ext2* (standardowy system plików w Linuksie), *ext3* (wersja systemu *ext2* wzbogacona o kronikę), *vfat* (32-bitowa partycja FAT systemów Windows), *iso9660* (standardowy system plików wykorzystywany na płytach CD-ROM), *nfs* (sieciowy system plików wykorzystywany w protokole NFS firmy Sun) i *swap* (przestrzeń wymiany). Jeżeli dany system nie jest aktualnie wykorzystywany, ale użytkownik chciałby zachować informacje o nim w pliku */etc/fstab*, wystarczy umieścić w opisywanym polu opcję *ignore*, która spowoduje, że dany system nie będzie montowany, nie będzie sprawdzana jego spójność itd.

Informacje o typach systemów plików, których obsługa kompilowana jest do jądra, można znaleźć w pliku */proc/filesystems*, choć mogą one wprowadzić w błąd. Jądro systemu zwykle może obsługiwać również inne systemy plików, ale odbywa się to raczej na zasadzie ładowania odpowiednich modułów, a nie stałych wpisów. Oto przykład pliku */proc/filesystems* z odpowiednimi wpisami:

```
ext3
nodev sysfs
nodev rootfs
nodev bdev
nodev proc
nodev sockfs
nodev debugfs
nodev securityfs
nodev pipefs
nodev futexfs
nodev tmpfs
nodev inotifyfs
nodev eventpollfs
nodev devpts
ext2
nodev ramfs
nodev hugetlbfs
minix
iso9660
nodev nfs
nodev nfs4
nodev mqueue
nodev rpc_pipefs
reiserfs
nodev usbfs
nodev subfs
vfat
hfsplus
nodev smbfs
```

Systemy plików, których wpisy poprzedzone są słowem `nodev`, nie są skojarzone z żadnym fizycznym urządzeniem, korzystają z nich jednak różne programy oraz system operacyjny.

- ♦ W czwartym polu można znaleźć listę oddzielonych przecinkami opcji polecenia `mount`, które mają zostać zastosowane podczas montowania danego systemu plików. Wiele z nich posiada specyficzne właściwości, niżej podaję najczęściej używane:
 - ♦ `async`: zapis w systemie plików powinien być wykonywany asynchronicznie,
 - ♦ `auto`: system ma być montowany automatycznie po wykryciu lub po wykonaniu polecenia takiego jak `mount -a`,
 - ♦ `defaults`: jako domyślne używane są następujące opcje: `-async`, `auto`, `dev`, `exec`, `nouser`, `rw`, `suid`,
 - ♦ `dev`: urządzenie znakowe lub blokowe zawierające system plików znajduje się lokalnie w systemie,
 - ♦ `exec`: w systemie można wykonywać wszelkie programy, skrypty i inne, których uprawnienia wskazują na to, że są wykonywalne,
 - ♦ `gid=value`: podczas montowania systemu plików ustawia ID grupy montowanego systemu plików na określone ID grup numerycznych,
 - ♦ `noauto`: system nie zostanie zamontowany automatycznie po wykryciu ani też po wykonaniu polecenia `mount -a`; ta opcja jest zwykle wykorzystywana do obsługi wymiennych nośników danych, takich jak dyskiety czy CD,
 - ♦ `nouser`: system plików może zamontować tylko użytkownik `root`, zwykli użytkownicy nie będą mogli tego zrobić,
 - ♦ `owner`: właścicielem systemu plików zostaje użytkownik, który go zamontował; jeżeli system został zamontowany automatycznie, wówczas jego właścicielem jest użytkownik `root`,
 - ♦ `ro`: system jest montowany w trybie tylko do odczytu,
 - ♦ `rw`: system jest montowany w trybie do odczytu i zapisu,
 - ♦ `suid`: umożliwia programom zmianę ID użytkownika lub grupy, kiedy są one wykonywane, jeżeli taką operację umożliwiają znaczniki specjalne; używając tej opcji podczas importowania systemu plików, należy zachować ostrożność, ponieważ uruchamianie programów, które ustawiają UID na użytkownika `root`, to popularny sposób włamywania się do systemu,
 - ♦ `uid=value`: podczas montowania systemu plików ustawia ID użytkownika montowanego systemu plików na określone ID użytkowników numerycznych.

Więcej informacji o opcjach dostępnych dla polecenia `mount` można znaleźć na stronach podręcznika `man` dostępnego w systemie Linux, zwłaszcza w części 8. (wystarczy użyć polecenia `man 8 mount`).

- ♦ Piąte pole używane jest przez polecenie `dump` służące do wykonywania kopii zapasowych w systemach Linux i Unix; jeżeli przy danym systemie plików wprowadzono ten zapis, podczas wykonywania programu `dump` zostanie

wykonana jego kopia zapasowa. Jeżeli w tym polu umieszczona została wartość 0 (lub też nie ma tam żadnego wpisu), wówczas podczas wykonywania kopii zapasowej przez polecenie `dump` dany system plików (powiązany z odpowiednim wpisem w pliku `/etc/fstab`) zostanie pominięty.

- ♦ Szóste pole jest używane przez systemy Linux i Unix podczas sprawdzania spójności systemów plików (więcej informacji można znaleźć w dalszej części rozdziału). Na tej podstawie identyfikowane są systemy, dla których należy przeprowadzić takie sprawdzanie podczas ponownego uruchamiania systemu. Jeżeli w polu znajduje się wartość 0 lub nie ma żadnego wpisu, wówczas program `fsck` przyjmie, że system plików skojarzony z danym wpisem w pliku `/etc/fstab` nie wymaga przeprowadzenia sprawdzania spójności danych.

Początkowo zawartość pliku może wydawać się skomplikowana, ale z czasem użytkownik przywyknie do tworzenia nowych wpisów i edytowania już istniejących. Istnieje wiele narzędzi graficznych, które umieszczają odpowiednie wpisy w pliku, wyręczając użytkownika. Polecam też metodę, z której sam korzystam: wystarczy skopiować już istniejący, choć prosty wpis, a następnie dostosować go do potrzeb nowo tworzonej lub dodawanej partycji.

Automatyczne montowanie systemów plików znajdujących się na przenośnych nośnikach danych

Oprócz wpisów w pliku `/etc/fstab`, Ubuntu posiada również specjalne wsparcie mające uprościć korzystanie z przenośnych nośników danych. Choć ten temat został omówiony szerzej w innej części książki, warto o nim wspomnieć w tym miejscu, aby uzupełnić informacje na temat urządzeń służących do przechowywania danych, z których można korzystać w Ubuntu.

Automatyczne rozpoznawanie wymiennych nośników danych w Ubuntu jest obsługiwane przez program `gnome-volume-manager application`, który można skonfigurować poprzez `gnome-volume-properties`. Aby uzyskać dostęp do menu konfiguracyjnego, wystarczy z menu wybrać kolejno *System/Preferencje/Napędy i nośniki wymienne*.

Raz skonfigurowany `gnome-volume-manager` zdecydowanie ułatwia automatyczne rozpoznawanie płyt CD, DVD, kart CompactFlash i innych przenośnych nośników danych oraz montowanie ich w odpowiednim punkcie systemu (czyli katalogu `/media`). Korzystanie z programu `gnome-volume-manager` oraz dostosowywanie go omówiłem szczegółowo w rozdziale 23., „Dodawanie sprzętu i przyłączanie urządzeń peryferyjnych”.

Program `gnome-volume-manager` umożliwia obsługę urządzeń, które są często podłączane do systemu w trakcie działania, dotyczy to nie tylko nośników danych, ale też urządzeń peryferyjnych, takich jak drukarki, myszy, klawiatury, tablety itp. Ułatwia też systemowi automatyczne rozpoznawanie innych elektronicznych urządzeń, np. aparatów cyfrowych czy palmtopów, dzięki czemu można łatwo transferować dane między nimi a komputerem, a zatem można z nich korzystać jak z dysków. Ten aspekt działania `gnome-volume-manager` omówiłem szerzej w rozdziale 17., „Urządzenia elektroniczne w Ubuntu”.

Podstawowe informacje o uprawnieniach w systemach Linux

Kiedy Ubuntu jest współdzielony pomiędzy kilku użytkowników, warto wiedzieć, w jaki sposób system chroni pliki i foldery, tak aby dostęp miały do nich tylko uprawnione osoby. Gdy czytelnik zainstalował Ubuntu i jest odpowiedzialny za opiekę nad nim (czyli mówiąc fachowo, będzie „administrował systemem”), musi wiedzieć, w jaki sposób system będzie decydował, którzy użytkownicy będą mogli wykonywać zadania wymagające specjalnych uprawnień. Byłoby bardzo niewygodne, ale i niebezpieczne, gdyby dowolny użytkownik mógł formatować dyski, zmieniać ustawienia połączenia z internetem itp. Jeżeli nawet przyjmujemy, że nikt nie będzie chciał specjalnie wyrządzić żadnych szkód, to jednak podczas korzystania może się zdarzyć, iż użytkownik zmieni ustawienia systemu, nawet o tym nie wiedząc (bo np. sądził, że kliknięcie przycisku *OK* spowoduje tylko zamknięcie programu).

Linux obsługuje tradycyjny system uprawnień znany z systemów uniksowych, na który składają się użytkownicy i grupy; wprowadzono doń jednak kilka zmian, które mają na celu głównie ułatwienie użytkowania. Czytelnicy, którzy wcześniej korzystali już z innych Linuksów, nie będą mieli kłopotów z przejściem na system stosowany w Ubuntu. Nowi użytkownicy będą mogli się przekonać, że system uprawnień jest prosty i łatwy.



W Ubuntu dostępny jest prosty, graficzny program do tworzenia użytkowników i zarządzania nimi. Poniżej omówiłem założenia samego systemu uprawnień w Ubuntu — korzystanie z tego programu oraz tworzenie użytkowników i zarządzanie nimi omówiłem szczegółowo w rozdziale 21., „Zarządzanie użytkownikami, grupami i zaawansowane uprawnienia”.

Podstawowe koncepcje: użytkownicy i grupy

Wszystkie systemy linuksowe wyposażone są w dwa podstawowe sposoby zarządzania, które wykorzystywane są do określania, kto może mieć dostęp do jakich plików oraz kto może wykonywać operacje wymagające specjalnych uprawnień. Każdy użytkownik logujący się w systemie robi to za pośrednictwem konta użytkownika, na które składa się nazwa oraz hasło. Dla każdego konta tworzony jest specjalny katalog domowy, w którym przechowywane są dane konfiguracyjne oraz wszelkie pliki tworzone przez użytkownika. W pliku */etc/passwd* (to skrót od *password file*, tylko po co męczyć się wpisywaniem niepotrzebnych liter, jeśli nie trzeba tego robić) znajduje się lista wszystkich użytkowników posiadających konta na danym komputerze.

Każdy użytkownik systemu Linux należy do jednej lub wielu grup. **Grupa** to jednostka administracyjna, za której pomocą można określonej grupie użytkowników łatwo udostępnić określony zestaw plików. Informacje o grupach zdefiniowanych w systemie oraz o należących do nich użytkownikach można znaleźć w tekstowym pliku */etc/group*, który ma łatwą do przyswojenia strukturę. Jak wyjaśniam poniżej, uprawnienia do plików i katalogów mogą być ustawiane w taki sposób, aby użytkownicy będący członkami danych grup mogli otwierać lub (i) zapisywać pliki we współdzielonych katalogach, do których nie mają dostępu inni użytkownicy systemu. Dzięki temu można łatwo współdzielić różne pliki czy inne informacje z wybranymi użytkownikami systemu.



Większość systemów linuksowych do uwierzytelniania i uzyskiwania informacji o członkach grup korzysta z lokalnych plików `/etc/passwd` i `/etc/group`, ale systemy stosujące mechanizmy uwierzytelniania w sieci nie mogą uzyskać takich informacji z innych komputerów połączonych w sieć. Dla przykładu w systemach wykorzystujących protokół NIS (ang. *Network Information Service*) jest używany w wielu środowiskach linuksowych opartych na systemie NFS pliki zawierające informacje o grupach i hasłach mogą zawierać wpisy, które powodują sprawdzanie przez NIS informacji dotyczących uwierzytelniania użytkowników i grup. Linux oferuje też obsługę mechanizmu PAM (ang. *Pluggable Authentication Modules*), który umożliwia systemowi pobieranie informacji ze źródeł znajdujących się w sieci, takich jak serwery LDAP (ang. *Lightweight Directory Access Protocol*), serwery Kerberos, a nawet serwery Microsoft Windows. Mechanizm PAM nie został tutaj opisany, ponieważ dotyczy specjalnych i zaawansowanych rozwiązań, a zamierzeniem niniejszego rozdziału jest jedynie wprowadzenie do podstawowych zagadnień bezpieczeństwa w Ubuntu i Linuksa. Protokół NIS omówiłem w rozdziale 21., „Zarządzanie użytkownikami, grupami i zaawansowane uprawnienia”.

Wpis w pliku `/etc/passwd` w moim systemie Ubuntu wygląda następująco:

```
wvh:x:1000:1000:William von Hagen,,,:/home/wvh:/bin/bash
```

Oto omówienie jego zawartości po kolei.

- ♦ Nazwa użytkownika.
- ♦ Pole hasła. W tym przypadku znak `x` oznacza, że informacje o hasle są przechowywane w pliku `/etc/shadow`.
- ♦ ID użytkownika, jest to wartość numeryczna powiązana z nazwą użytkownika.
- ♦ ID grupy, której domyślnym członkiem jest dany użytkownik.
- ♦ Pole zawiera pełną nazwę użytkownika oraz inne informacje tekstowe, takie jak lokalizacja biura oraz numery telefonów do biura i do domu.
- ♦ Katalog domowy użytkownika.
- ♦ Aplikacja uruchamiana po zalogowaniu się użytkownika; w tym przypadku powłoka (*shell*).

Oto kilka przykładowych wpisów z pliku `/etc/group` znajdującego się w tym samym systemie:

```
adm:x:4:wvh,juser  
cdrom:x:24:hal,wvh,juser  
lpadmin:x:106:wvh,juser  
admin:x:112:wvh,juser  
wvh:x:1000:
```

Każdy wpis w pliku `/etc/group` rozpoczyna się od nazwy grupy, pola hasła (w powyższym przykładzie wskazuje ono, że wszelkie hasła grup są przechowywane w pliku `/etc/shadow`), numerycznego identyfikatora powiązanego z daną grupą oraz listy użytkowników należących do danej grupy.

Jak widać w powyższym przykładzie, jeden użytkownik może należeć do wielu grup, jest to sytuacja całkiem zwyczajna. Każda z grup umożliwia dostęp do określonych zasobów,

takich jak pliki, katalogi, i pozwala na wykonywanie konkretnych czynności administracyjnych. Listę grup, do których należy użytkownik, można sprawdzić, wykonując w terminalu polecenie `groups`.

W trakcie procesu instalowania systemu tworzy się dla każdego użytkownika konto oraz domyślnie grupę. Jest to grupa, której domyślnym członkiem staje się użytkownik podczas logowania do systemu. Programy, które określają uprawnienia administracyjne na podstawie przynależności do określonych grup, najpierw sprawdzają je w pliku `/etc/group`, dopiero później są wykonywane. Aby uzyskać dostęp do współdzielonych zasobów, być może trzeba będzie zmienić grupę, do której należy użytkownik. Tę operację można przeprowadzić w terminalu, korzystając z polecenia `newgrp`.

W Ubuntu numery grup mniejsze od 1000 są zarezerwowane dla celów administracyjnych. Różne programy korzystające z tych grup automatycznie sprawdzają listę członków w trakcie działania. Nie ma potrzeby zapamiętywania, która z grup jest domyślna dla użytkownika, ponieważ programy wymagające specjalnych uprawnień przyznanych użytkownikowi same dodadzą go do właściwej. Więcej szczegółów na ten temat można znaleźć w punkcie „Wykonywanie czynności wymagających specjalnych uprawnień”.

Uprawnienia do plików i katalogów w Linuksie

W niniejszym punkcie opisuję, w jaki sposób działają (i jak użytkownik może z nich korzystać) uprawnienia do plików i katalogów w systemach Linux. Choć ten temat został szerzej omówiony w rozdziale 6., „Korzystanie z linii poleceń”, warto wprowadzić go już teraz, aby zilustrować, w jaki sposób tożsamość użytkownika czy grupy umożliwiającą dostęp do plików, katalogów i innych zasobów oraz w jaki sposób zmieniać te ustawienia.

Najprostszym poleceniem, za którego pomocą można sprawdzić uprawnienia do danego pliku czy katalogu, jest komenda `ls` (*list*), efektem jej działania jest wyświetlenie informacji na temat plików i katalogów znajdujących się w określonej lokalizacji.

I tak polecenie `ls -ld /home/wvh` dostarczy informacji na temat zawartości katalogu domowego użytkownika `wvh`, w którym znajdują się dane na temat aktualnych uprawnień do tego katalogu:

```
drwxr-xr-x 145 wvh wvh 7728 2006-03-05 10:54 /home/wvh
```

Skupmy się na pierwszym polu, w którym wyświetlane są aktualne uprawnienia do tego katalogu (w rozdziale 6., „Korzystanie z linii poleceń”, zamieszczono dodatkowe informacje na temat zawartości wszystkich pól, a także polecenia `ls` oraz jego opcji).

Uprawnienia wyświetlone za pomocą polecenia `ls` można podzielić na cztery części.

- ♦ Pierwszy znak identyfikuje typ obiektu. Jeżeli jest to zwykły plik, wówczas zostanie wyświetlona litera `a`; litera `d` — gdy będzie to katalog i `b` — w przypadku urządzenia węzła, do którego dostęp można uzyskać, tak jak do urządzenia blokowego.
- ♦ Trzy zestawy po trzy znaki, które definiują uprawnienia właściciela, grupy oraz innych użytkowników do danego pliku czy katalogu.

Najczęściej spotykanymi wartościami dla tych trzech pól są: *r* — co oznacza, że dany plik lub katalog jest do odczytu; *w* — oznaczające, że plik lub katalog można zapisywać oraz *x* — oznaczające (w przypadku pliku), że jest to plik wykonywalny, a dla katalogu, że może on być przeszukiwany. Jeżeli któraś z wartości nie będzie ustawiona, wówczas w danej pozycji pojawi się kreska. Bit wykonywalności dla użytkownika lub grupy może być ustawiony na *s*, co oznacza „wykonaj plik z uprawnieniami właściciela (użytkownika lub grupy)”. Pozwala to zwykłym użytkownikom na uruchamianie programów wymagających specjalnych uprawnień.

Nadawanie uprawnień domyślnych podczas tworzenia plików i katalogów

Klasycznym unikowym mechanizmem służącym do ustawiania trybu ochrony plików i folderów tworzonych przez użytkownika jest *umask*. Jest to czterocyfrowa liczba ósemkowa, której wartość w przypadku tworzenia plików to 0666, a w przypadku katalogów 0777. W większości dystrybucji Linuksa domyślna wartość *umask* to 0002, co oznacza, że każdy tworzony przez użytkownika plik posiada tryb 0664, co z kolei skutkuje tym, że użytkownik oraz członkowie grupy mogą odczytywać i zapisywać plik, pozostali natomiast mogą go tylko odczytywać. Podobnie jest w przypadku katalogów tworzonych w trybie ochrony 0775: użytkownik i członkowie grupy mogą tworzyć w nich pliki, natomiast pozostali użytkownicy mogą tylko przeglądać ich zawartość oraz przeszukiwać je. Aby sprawdzić ustawienia maski *umask*, należy w linii poleceń wykonać polecenie *umask*.

Większość użytkowników ustawia *umask* w pliku konfiguracyjnym powłoki (zwykle *~/.bashrc*) na wartość 0022, dzięki czemu tworzone pliki mogą być odczytywane tylko przez właściciela (np. są tworzone z opcją 0644), podobnie jest w przypadku katalogów, które mogą być zapisywane tylko przez właściciela (np. mają opcję 0755). Możliwość definiowania wartości *umask* jest wbudowana we wszystkie powłoki uniksowe. Więcej informacji na temat jej ustawiania można znaleźć w dokumentacji systemowej (powłoka *bash* została omówiona w rozdziale 6. „Korzystanie z linii poleceń”).

Wykonywanie czynności wymagających specjalnych uprawnień

Najważniejszym użytkownikiem w systemie Linux jest użytkownik **root** (inaczej super-użytkownik), którego ID (i ID grupy) to 0.

W większości systemów Linux operacje wymagające specjalnych uprawnień wykonywane są za pomocą narzędzia *su* (*substitute user ID*), które umożliwia uzyskanie uprawnień użytkownika **root**. Jednak w Ubuntu sprawy mają się nieco inaczej. W tej dystrybucji do wykonywania zadań wymagających uprawnień administratora wykorzystuje się polecenie *sudo* (*substitute user ID do*). Korzysta ono z pliku konfiguracyjnego */etc/sudoers*, aby stwierdzić, którzy użytkownicy mogą wykonywać operacje wymagające uprawnień super-użytkownika. W Ubuntu każdy członek grupy administratorów (*admin*) może działać jako użytkownik **root**. Z polecenia *sudo* można korzystać w każdym systemie Linux, ale w Ubuntu nie można używać polecenia *su*.

Różnice pomiędzy poleceniami `su` i `sudo` są subtelne, ale znaczące.

- ♦ Gdy używamy polecenia `su` do wykonania zadań wymagających specjalnych uprawnień, najpierw wykonujemy polecenie `su` i po podaniu hasła superużytkownika uzyskujemy odpowiednie uprawnienia. Następnie można wykonać dowolną komendę: wszystkie są wykonywane przez użytkownika `root`. Po zakończeniu zamykamy konsolę, dzięki czemu nikt inny nie będzie mógł z niej skorzystać.
- ♦ Gdy używamy polecenia `sudo`, należy po nim wprowadzić nazwę i opcje polecenia, które ma zostać wykonane.



Działanie poleceń `su` i `sudo` nie jest ograniczone tylko do uzyskania uprawnień superużytkownika. Można zostać dowolnym użytkownikiem, wystarczy tylko po poleceniu `su` podać nazwę użytkownika oraz jego hasło. W podobny sposób można korzystać z polecenia `sudo` z opcją `-u`, dzięki czemu można określić nazwę użytkownika, z którego uprawnieniami ma zostać wykonane dane polecenie. Dla polecenia `su` wymagana jest znajomość hasła danego użytkownika, natomiast dla polecenia `sudo` wymagane jest tylko podanie własnego hasła.

Oto kilka przykładów, w których porównuję działanie obu tych poleceń. Poniżej prezentuję, w jaki sposób użyć polecenia `su` do wyświetlenia partycji dostępnych na dysku:

```
$ su
Password: <wprowadzić-hasło-superużytkownika>
# fdisk -l /dev/sda
[dane wyjściowe]
# exit
$
```

Aby wykonać taką samą operację, używając polecenia `sudo`, należy:

```
$ sudo fdisk -l /dev/sda
Password: <wprowadzić-hasło-użytkownika>
[dane wyjściowe]
$
```

W dystrybucjach innych niż Ubuntu można też użyć polecenia `su` z opcją `-c` (wykonywanego jako jedno polecenie) do przeprowadzenia prostych działań, np.:

```
$ su -c fdisk -l /dev/sda
Password: <wprowadzić-hasło-superużytkownika>
[dane wyjściowe]
$
```

Kluczowa różnica pomiędzy `su` i `sudo` polega na konieczności wprowadzenia hasła użytkownika `root` lub użytkownika (należącego do grupy `admin`). Jednakże korzystanie z polecenia `sudo` posiada kilka istotnych zalet.

- ♦ Trudniej zaatakować system, ponieważ najpierw należy odkryć, którzy z użytkowników mają uprawnienia do wykonywania zadań administracyjnych, dopiero potem można próbować się włamać na ich konta.
- ♦ Uprawnieni użytkownicy muszą pamiętać tylko swoje hasło, nie ma potrzeby ustanawiania odrębnego hasła dla superużytkownika.



Jeżeli użytkownik chce wykonać kilka poleceń wymagających specjalnych uprawnień, ma dwie możliwości. Po pierwsze, może wykonać polecenie `sudo` z opcją `-s`, co spowoduje, że polecenie będzie wykonane z uprawnieniami superużytkownika. Po drugie, może te same polecenia wykonać, uruchamiając komendę `sudo /bin/bash`. W obu przypadkach należy pamiętać o zamknięciu powłoki po wykonaniu wszystkich zaplanowanych działań.

Podsumowanie

W niniejszym rozdziale omówiłem pewne podstawowe założenia systemów Linux i Ubuntu. Opisałem pliki oraz katalogi, również te, które są najczęściej używane w Ubuntu. Przedstawiłem sposób, w jaki Ubuntu używa dysków, jak tworzy partycje dla osobnych systemów plików, wyliczyłem dostępne systemy plików, które użytkownik może wykorzystać, oraz powiedziałem, co trzeba zrobić, żeby były dostępne w systemie. Ostatnie podrozdziały poświęciłem podstawowym zagadnieniom dotyczącym administrowania Linuksem, a szczególnie zarządzaniu użytkownikami i grupami oraz wykonywaniu poleceń wymagających szczególnych uprawnień. W niniejszym rozdziale czytelnik zapoznał się z podstawami niezbędnymi do zrozumienia informacji podawanych w dalszej części książki.

W następnym rozdziale omówię GNOME, graficzny interfejs użytkownika instalowany domyślnie w Ubuntu. Opiszę m.in. korzystanie z różnych przycisków myszy w programach graficznych, posługiwanie się menu GNOME w celu uruchamiania aplikacji, korzystanie z paneli pulpitu, czyli podam wszelkie informacje, które mogą być pomocne czytelnikowi w rozpoczęciu pracy z nowo zainstalowanym systemem Ubuntu.

Rozdział 5.

Środowisko GNOME

W tym rozdziale:

- ♦ Używanie myszy
- ♦ Podstawy GNOME
- ♦ Korzystanie z pulpitu
- ♦ Dostosowywanie pulpitu
- ♦ Wykorzystanie menedżerów okien

Chociaż wielu wiernych fanów Uniksa długo unikało jakiegokolwiek interfejsu graficznego, to należało jednak zmierzyć się z faktem, że większość dzisiejszych użytkowników oczekuje (i spodziewa się) takiego rozwiązania. Środowisko graficzne używane w Ubuntu to GNOME. Jest to stabilne i użyteczne środowisko umożliwiające uruchamianie graficznych aplikacji i korzystanie z nich. Większość linuksowych narzędzi używanych do administrowania i konfigurowania systemu wyposażona jest w interfejs graficzny, ułatwiający wykonywanie skomplikowanych zadań. Uruchamiać je można, korzystając z jednego z podstawowych menu dostępnych w GNOME.

Niniejszy rozdział rozpoczynam od podstawowych informacji na temat środowisk graficznych używanych w systemach Linux, następnie wyjaśnię znaczenie słowa „pulpit”, a także omówię alternatywne środowiska, których można używać zamiast GNOME. Pozostała część rozdziału poświęcona jest korzystaniu ze środowiska GNOME. Jest to potężny graficzny interfejs użytkownika (angielski skrót to GUI od *Graphical User Interface*) zawierający wszystkie funkcje, jakich można oczekiwać od współczesnych środowisk, wystarczy tylko wiedzieć, gdzie ich szukać.

Co to jest pulpit?

Środowiska graficzne dla Linuksa

Niemal wszystkie operacje korzystające w wysokiej rozdzielczości są w Linuksie obsługiwane przez system graficzny o nazwie X Window, znany też jako X11 czy po prostu X. X Window jest jednym z najatrakcyjniejszych aspektów Linuksa i niemal każdy system działający obecnie z grafikami bitmapowymi korzysta właśnie z niego. Umożliwia on

pracę z graficznymi aplikacjami, podobnie jak czynią to środowiska graficzne używane w Microsoft Windows czy Mac OS X. System X Window został zaprojektowany w celu dostarczenia środowiska okienkowego dla wieloprotokółowych, połączonych siecią komputerów i obecnie jest standardem dla maszyn — na których uruchamiane są różne wersje systemów Linux czy inne uniksopodobne systemy operacyjne — a nawet dla urządzeń, z wciąż działającym systemem operacyjnym Compaq VMS. Odpowiednie wersje X Window są również dostępne dla systemów Windows (od wersji 3.1) oraz dla wszystkich wersji Mac OS X — uruchamiają one ten system w otoczeniu natywnego środowiska okienkowego odpowiedniego dla każdej z tych platform.

System X Window został zaprojektowany w Massachusetts Institute of Technology (MIT), przez pewien okres był pod nadzorem Open Group (www.opengroup.org), a obecnie kierowany jest przez fundację X.org (www.x.org). Poprzednia implementacja systemu X Window dla Linuksa, znana pod nazwą XFree86, nie jest już używana — teraz oficjalnym źródłem implementacji X dla systemów linuksowych jest X.org.

System X Window umożliwia uruchamianie na ekranie wielu aplikacji, z których każda może wyświetlać własne okno, grafiki, okna dialogowe itp. w jednym oknie na ekranie lub w wielu oddzielnie sterowanych oknach. Menedżer okien to aplikacja, która również jest uruchamiana na ekranie, ale jej zadaniem jest zarządzanie poszczególnymi oknami tworzonymi i używanymi przez programy. Menedżer okien umożliwia przesuwanie okien po ekranie, zmianę ich rozmiaru, wyświetla też granice okien, dzięki czemu możliwe jest przesuwanie okien programów, odpowiedzialny jest też za pojawianie się komunikatów po kliknięciu pustej części ekranu. Dzięki niemu można też minimalizować, maksymalizować oraz zamykać osobne okna. Najpopularniejsze menedżery okien to: After Step, Black Box, Fluxbox, FVWM, ICEwm, kwm, Metacity, twm, vtwm i Window Maker. Wszystkie oferują podobne możliwości, choć różnią się metodami zarządzania minimalizowanymi programami, rodzajami konfiguracji, sposobem wyświetlania uruchamianych i działających programów, obsługą różnych czcionek oraz stylów wyświetlania itd.



Ponieważ system X Window został zaprojektowany do działania na wielu różnych platformach sprzętowych, to jedną ze wspaniałych jego możliwości jest fakt, że można uruchomić klienta X Window na jednym urządzeniu, natomiast wyświetlać obraz z niego na innym (przy założeniu, że taka możliwość nie jest blokowana przez zapórę sieciową czy konfigurację systemu). Te wartości są ustawiane za pomocą zmiennej systemowej DISPLAY, z której korzystają programy działające zdalnie. Zmienna środowiskowa DISPLAY ma następującą postać:

host:wyświetlacz.ekran

host to nazwa i adres IP komputera, na którym mają być wyświetlane dane, po nim wstawiony jest dwukropka. Później następuje numer (*wyświetlacz*) identyfikujący ekran, na którym będzie wyświetlane dane okno. Po kropce umieszczony jest numer fizycznego ekranu podłączonego do karty. Zwykle wartości dla *wyświetlacza* i *ekranu* to 0.

W punkcie rozdziału 6. zatytułowanym „Zmienne środowiskowe” zamieszczono dodatkowe informacje na temat ustawiania zmiennych systemowych. W wielu przypadkach można również uruchomić system X Window zdalnie, wtedy otwieramy programy z poziomu linii poleceń, podając argument `-display host:wyświetlacz.ekran`.

Po menedżerze okien kolejnym etapem ewolucji systemu X Window jest menedżer pulpitu. Potocznie kojarzony z pulpitem, zwykle uruchamiany jest w ukryciu, ale oferuje:

- ♦ mechanizm „przeciągnij i upuść”,
- ♦ menedżer plików ułatwiający graficzne przeglądanie plików i katalogów,
- ♦ proste i elastyczne sposoby na łączenie programów z przyciskami, ikonami i menu,
- ♦ scentralizowany mechanizm konfigurowania wyglądu ekranu oraz wyświetlanych okien, znany pod nazwą **motywów**,
- ♦ obsługę niewielkich programów (**apletów**) służących do wykonywania określonych rodzajów zadań.

Najczęściej używanymi środowiskami graficznymi dla systemów Linux są GNOME (GNU Network Object Model Environment, www.gnome.org), KDE (K Development Environment, www.kde.org), ROX (rox.sourceforge.net) i XFCE (www.xfce.org). Zarówno GNOME, jak i KDE to dobrze wyposażone, solidne środowiska wyposażone we własne menedżery okien (w GNOME wykorzystywany jest Metacity, a w KDE — kwm). Podstawowe różnice to sposoby powiązania działania klawiszy klawiatury i przycisków myszy z programami oraz organizacja menu i biblioteki graficznej (w GNOME korzysta się z biblioteki GTK, a w KDE z QT). Środowiska ROX i XFCE charakteryzują się niewielkimi wymaganiami. Starano się w nich zachować równowagę pomiędzy sporymi wymaganiami sprzętowymi stawianymi przez potężne środowiska (takie jak GNOME czy KDE) a niskimi możliwościami starszych komputerów. Dobre wprowadzenie do większości menedżerów i środowisk graficznych dostępnych dla systemu X Window można znaleźć na stronach <http://xwinman.org>.

Pozostała część rozdziału poświęcona jest GNOME, środowisku wykorzystywanemu w Ubuntu, które jest dostępne po pierwszym zalogowaniu się do systemu. Użytkownik może zmienić menedżera okien używanego w Ubuntu, przełączyć się na środowisko KDE, ale są to nieco bardziej zaawansowane zagadnienia. Najpierw należy zapoznać się ze standardowym środowiskiem Ubuntu, może się okazać, że całkowicie spełnia ono oczekiwania użytkownika i nie ma potrzeby dokonywania zmian w konfiguracji systemu. Zawsze będzie wystarczająco dużo czasu, aby wypróbować różne sztuczki i triki związane ze środowiskiem graficznym; część z nich omówiłem w podrozdziale zatytułowanym „Korzystanie z menedżera okien”.

Korzystanie z myszy

Korzystanie w jakiegokolwiek interfejsu graficznego wymaga używania myszy i klawiatury, dzięki którym można kontrolować kursor, zaznaczać i wybierać wyświetlane elementy, uzyskiwać dostęp do menu czy wprowadzać dane do programów. Jak czytelnik przekona się w trakcie lektury niniejszej książki, działanie przypisane poszczególnym przyciskom myszy zależy m.in. od tego, co jest w danym momencie wyświetlane na ekranie komputera. Takie dostosowywanie działania w zależności od kontekstu pozwala na wykonywanie wielu różnorodnych zadań przy wykorzystaniu tylko trzech niewielkich przycisków.

Domyślnie mysz w GNOME jest skonfigurowana dla praworęcznych użytkowników, a poszczególnym przyciskom są przypisane następujące działania.

KDE w Ubuntu?

Jak wspomniałem wcześniej, KDE (czyli K Desktop Environment) to środowisko graficzne zaprojektowane tak, aby dostarczyć użytkownikom standardowe i łatwe w użytkowaniu wszechstronne narzędzie. KDE zostało zainspirowane początkowo projektem CDE (ang. *Common Desktop Environment*), który był efektem współpracy głównych dostawców Uniksa, takich jak IBM i Sun Microsystems, i działał na ich sprzęcie. CDE zostało zaprojektowane i wykonane z użyciem bibliotek Open Group Motif X Window, co oznaczało, że każdy dostawca CDE posiadał licencję pakietu Motif. Było to nie do zaakceptowania przez wolne środowisko Linuksa i idee oprogramowania o otwartym kodzie, dlatego powstało KDE — wolna alternatywa wobec CDE. Początkowo projekt KDE był sponsorowany przez kilku dostawców i fanów Linuksa, takich jak Caldera, Delix, O'Reilly Associates czy SUSE. Niestety, twórcy KDE wybrali do środowiska bibliotekę Qt, która była ograniczona pewnymi licencjami. Wszystko to zostało później poprawione, ale wówczas efektem takiego wyboru było powstanie kolejnego projektu, czyli GNOME, który z czasem stał się domyślnym wyborem wśród wielu użytkowników i dostawców Linuksa (Ubuntu, Red Hat, Yellow Dog, Mandrake itd.). Więcej informacji na temat KDE można znaleźć na stronach www.kde.org.

Ulubione środowiska graficzne są jak opinie — każdy ma jakieś. Nawet ci z użytkowników, którzy na przestrzeni lat korzystali z wielu różnorodnych środowisk, mają swoje ulubione. Aby zadowolić fanów KDE, powstała specjalna wersja Ubuntu, czyli Kubuntu, w której domyślnym środowiskiem jest właśnie KDE. Obrazy płyt Kubuntu można pobrać ze strony www.kubuntu.com, tam też można zamówić gotowe płyty (tak jak w przypadku Ubuntu). KDE można zainstalować na standardowym Ubuntu, ale dlaczego nie skorzystać z efektów pracy członków społeczności Ubuntu, którzy są fanami KDE? Podobnie jest w przypadku projektu Xubuntu, czyli wersji Ubuntu z zainstalowanym domyślnie środowiskiem o niewielkich wymaganiach, czyli XFCE.

- ♦ **Lewy przycisk:** pojedyncze kliknięcie zaznacza ikonę, menu czy też jego pozycję lub obiekt w programie. Podwójne kliknięcie ikony znajdującej się na pulpicie uruchamia skojarzony z nią obiekt. Jeżeli przycisk zostanie przytrzymany i jednocześnie kursor będzie przesuwany np. po pulpicie, spowoduje to zaznaczenie wielu elementów. Po zaznaczeniu wybranego elementu można go przesuwać po pulpicie, wystarczy tylko przytrzymać wciśnięty przycisk. Wymienione operacje można ze sobą łączyć; np. w edytorze tekstów można zaznaczać bloki tekstu, przesuwać nad nimi kursor z jednocześnie wciśniętym przyciskiem myszy, a następnie zwolnić przycisk, po czym ponownie go wcisnąć (kursor musi znajdować się nad zaznaczonym fragmentem), wówczas będzie można przesuwać blok tekstu w dowolne miejsce.
- ♦ **Środkowy przycisk:** pojedyncze kliknięcie powoduje wklejenie uprzednio wyciętego lub skopiowanego tekstu czy obiektu. Może też spowodować przeniesienie wybranego okna za te, które są aktualnie widoczne. Po kliknięciu ikony znajdującej się na pulpicie i przeciągnięciu kursora na nową pozycję zostanie wyświetlone menu umożliwiające przeniesienie ikony bez uprzedniej zmiany jej położenia, nastąpi ona dopiero po potwierdzeniu przez użytkownika zamiaru.
- ♦ **Prawy przycisk:** służy do wyświetlania menu kontekstowego zależnie od aktualnej pozycji kursora.

Użytkownik może skonfigurować działanie myszy, wygląd kursora oraz działanie poszczególnych przycisków (za pomocą specjalnego programu, więcej informacji na ten temat można znaleźć w punkcie „Dostosowywanie działania myszy”). Jeżeli np. użytkownik jest osobą leworęczną i sprawia mu kłopot korzystanie z ustawień standardowych, wówczas może zmienić konfigurację myszy tak, aby odwrócić działanie przycisków.



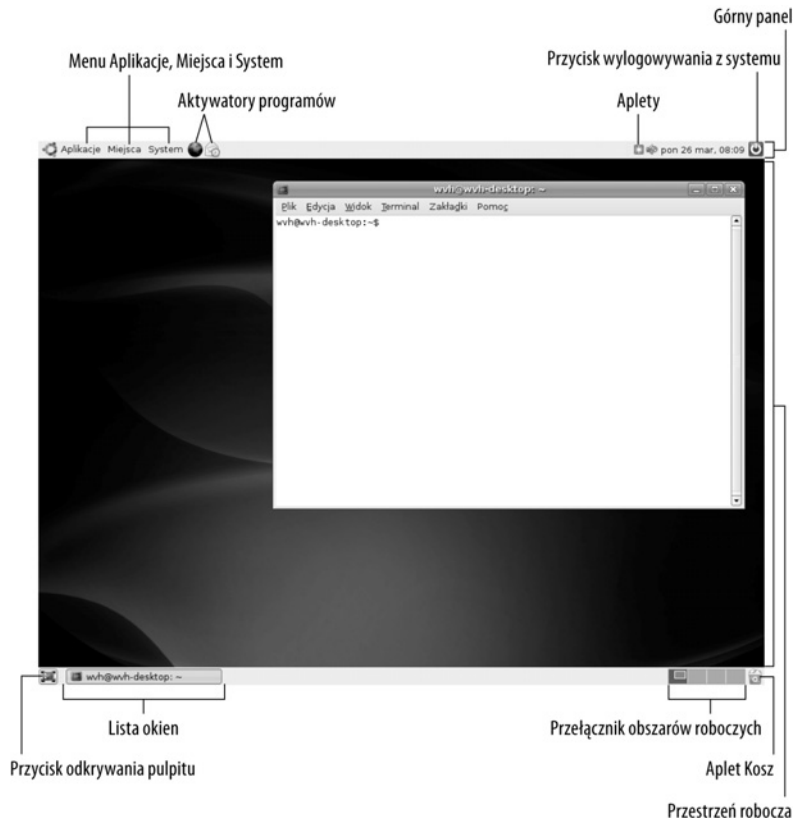
Jeżeli użytkownik ma kłopoty z obserwowaniem pulpitu czy kursora myszy, nie ma powodu do zmartwień — programiści GNOME i Ubuntu zadbali również o to. Więcej informacji można znaleźć w punkcie „Technologie wspierające w GNOME” znajdującym się w dalszej części rozdziału.

Wprowadzenie do GNOME

Na rysunku 5.1 przedstawiam domyślny pulpit GNOME w Ubuntu Linux, który użytkownik zobaczy po pierwszym zalogowaniu się do systemu. Widać okno aplikacji, które zostanie wyświetlone po wybraniu z menu *Aplikacje/Akcesoria/Terminal*, a także dodatkowe informacje, które pozwolą zapoznać się z podstawowymi funkcjami środowiska. Zespół Ubuntu włożył wiele wysiłku, aby stworzyć atrakcyjny, przyjemny dla oka zestaw czcionek, tło, dekorację okien i przyciski dla wszystkich uruchamianych programów. Zestaw takich grafik nosi nazwę **motywu**, w dalszej części rozdziału zamieszczam informację, w jaki sposób można łatwo dostosowywać motywy do własnych potrzeb.

Rysunek 5.1.

Domyślny pulpit GNOME w Ubuntu



Na rysunku 5.1 nazwałem różne części GNOME, a oto ich dokładniejszy opis.

- ♦ **Górny panel:** panel to specjalna część ekranu kontrolowana przez program `gnome-panel`, udostępniono na niej wiele możliwości korzystania z pulpitu

i uruchamiania różnych aplikacji. GNOME może wyświetlać panele na każdej krawędzi ekranu, ale domyślna konfiguracja przewiduje ich umiejscowienia na górze i na dole. Domyślnie górny panel zawiera następujące elementy.

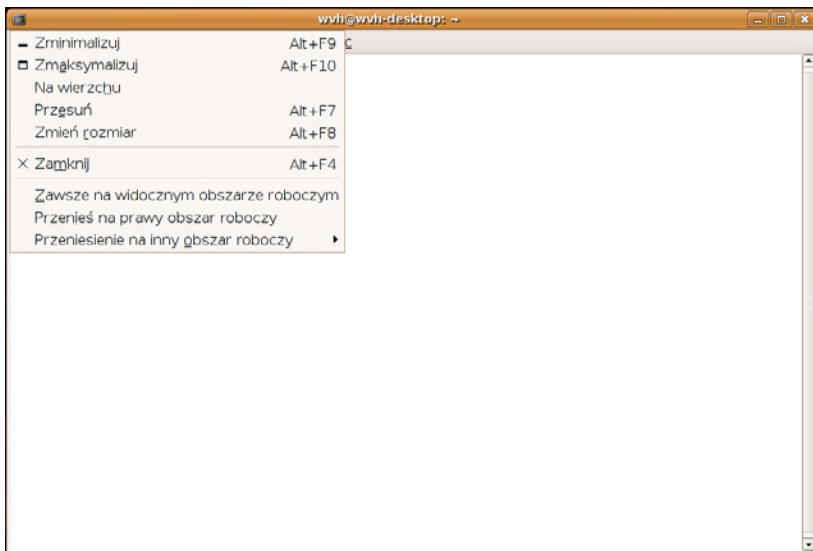
- ♦ **Menu *Aplikacje, Miejsca i System***: te menu składają się z listy skrótów do określonych poleceń, zadań czy aplikacji. Te trzy menu umożliwiają łatwy, graficzny dostęp do programów, miejsc w komputerze lub sieci czy zadań. Każde menu opisałem szczegółowo w podrozdziale pt. „Menu w GNOME”. Aby wyświetlić zawartość poszczególnych menu, wystarczy kliknąć je lewym przyciskiem myszy, zostanie wówczas wyświetlona lista rozwijana. Aby ją zamknąć, wystarczy kliknąć przyciskiem myszy z kursorem ustawionym na wolnym polu na pulpicie.
- ♦ **Aktywatory programów**: po kliknięciu takiej ikony zostanie uruchomiony odpowiedni program.
- ♦ **Aplety**: te ikony uruchamiają niewielkie programy oferujące przeróżne możliwości. Dodawanie i konfigurowanie apletów zostało omówione w punkcie „Dostosowywanie paneli”.
- ♦ **Przycisk wylogowywania z systemu**: służy do szybkiego kończenia sesji i wyświetlania ekranu logowania.
- ♦ **Przestrzeń robocza**: w tej części uruchamiane są programy oraz wyświetlane okna dialogowe i komunikaty.
- ♦ **Dolny panel**: ten panel jest wyświetlany przy dolnej krawędzi ekranu i zawiera następujące elementy.
 - ♦ **Przycisk odkrywania pulpitu**: po jego kliknięciu wszystkie programy i okna dialogowe zostaną zminimalizowane, dzięki czemu pulpit stanie się widoczny.
 - ♦ **Lista okien**: każdy uruchomiony program będzie widoczny na przestrzeni roboczej oraz na wydzielonym fragmencie panelu dolnego. Po kliknięciu tego obszaru prawym przyciskiem myszy będzie można zminimalizować lub zmaksymalizować okno, natomiast kliknięcie lewym przyciskiem myszy spowoduje wyświetlenie menu kontekstowego, za którego pomocą będzie można lepiej dostosować zachowania okna.
 - ♦ **Przełącznik obszarów roboczych**: to specjalny aplet umożliwiający zarządzanie obszarami roboczymi, na którym w miniaturze wyświetlana jest zawartość każdego pulpitu. Obszary robocze są oddzielnymi, wirtualnymi pulpitemi dostępnymi w GNOME. Na każdym obszarze można uruchamiać programy, przenosić między nimi okna aplikacji itp. Za pomocą kilku obszarów roboczych można uruchamiać różne rodzaje aplikacji na osobnych pulpitych, dzięki czemu będą one widoczne dopiero wówczas, kiedy użytkownik przełączy się na odpowiedni obszar. Częstym przykładem wykorzystywania obszarów roboczych są gry: można na jednym z pulpitów uruchomić jakąś i grać, kiedy nikt nie patrzy przez ramię.
 - ♦ **Aplet *Kosz***: jest to specjalny aplet, umożliwiający dostęp do Kosza, czyli specjalnej przestrzeni w GNOME, w której są przechowywane pliki przed ich ostatecznym usunięciem.

GNOME jest niesłychanie konfigurowalnym środowiskiem, dlatego istnieje całkiem spora szansa, że nie ma na świecie dwóch tak samo wyglądających pulpitów GNOME — i bardzo dobrze. Sposób, w jaki użytkownik skonfiguruje swój system, zależy całkowicie od niego, powinien on tylko odpowiadać potrzebom i oczekiwaniom osób, które z niego korzystają. Zadaniem takich środowisk graficznych jest umożliwienie i maksymalne ułatwienie użytkownikowi zadań, które ma wykonać.

Okno aplikacji GNOME

Okno aplikacji GNOME przedstawione na rysunku 5.1 można kontrolować na wiele sposobów: przesuwać je po ekranie, zmieniać rozmiar itd. Lokalizacja służących do tego elementów oraz ich możliwości są zależne od wykorzystywanego menedżera okien. Elementy te można dostrzec w górnej części obramowania okna zwykle znanej pod nazwą paska tytułowego.

Menu kontrolne można wyświetlić, klikając pole znajdujące się po lewej stronie paska tytułowego. Jego zawartość zależy od stosowanego menedżera okien. Menu dostępne domyślnie w GNOME (w którym używany jest menedżer okien Metacity) przedstawiam na rysunku 5.2.



Rysunek 5.2. Menu kontrolne okna w menedżerze Metacity

Za pomocą trzech kontrolkek okna znajdujących się po prawej stronie paska tytułowego tegoż okna można wykonać następujące czynności:

- ♦ zminimalizować dane okno,
- ♦ przełączać okno pomiędzy rozmiarem pełnoekranowym a ustawionym przez użytkownika,
- ♦ zamknąć dane okno, przerywając jednocześnie wszelkie wykonywane operacje.

Wszystkie te kontrolki są standardowe i można je znaleźć w większości pasków tytułowych okien wyświetlanych przez współczesne menedżery okien.

Menu w GNOME

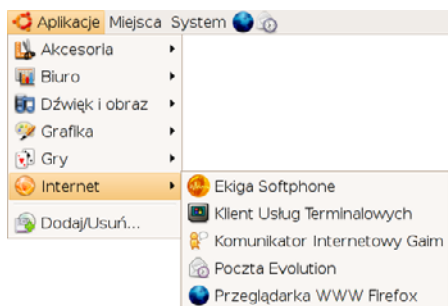
Jak wspominałem wcześniej, w GNOME dostępne są dwa rodzaje menu: te zlokalizowane na panelach, które umożliwiają łatwy dostęp do różnych aplikacji, lokalizacji i funkcji systemu, oraz menu kontekstowe udostępniające funkcje zależne od miejsca, w jakim zostały aktywowane. W niniejszej części zamieszczam więcej informacji na temat obu tych rodzajów menu, sposobów ich wyświetlania, udostępnianych przez nie funkcji oraz metod dostosowywania ich do własnych potrzeb.

Menu na panelach

Menu to jest wyświetlane po kliknięciu lewym przyciskiem nazwy danego menu. Menu te mogą zawierać skróty do pewnych operacji, lokalizacji w systemie, mogą też zawierać inne menu. Menu znajdujące się w innych menu noszą nazwę **podmenu** (czasami używa się angielskiej nazwy **submenu**); np. na rysunku 5.3 widać domyślną zawartość menu *Aplikacje* oraz podmenu *Internet*, które omówię w dalszej części rozdziału.

Rysunek 5.3.

Menu i podmenu w GNOME



Poniżej omawiam szczegółowo zawartość standardowych menu znajdujących się na panelu w Ubuntu.

Menu Aplikacje

Menu *Aplikacje* umożliwia łatwe uruchamianie różnych programów dostępnych w GNOME, zostały one pogrupowane w podmenu, które można wyświetlić, podświetlając odpowiednio pozycje kursorem myszy. W menu *Aplikacje* znajdują się następujące pozycje:

- ♦ **Akcesoria**: to kategoria zawierająca niewielkie, często używane programy, takie jak kalkulator, słownik, edytor tekstu i terminal.
- ♦ **Biuro**: znajduje się tu lista standardowych programów potrzebnych w biurze, które są jednocześnie dostępne w systemie. Domyślnie w menu znajduje się Evolution, zaawansowany program do obsługi poczty zintegrowany z kalendarzem, oraz programy wchodzące w skład pakietu OpenOffice.org, czyli zaawansowany edytor tekstu (więcej informacji można znaleźć w rozdziale 10., „Tworzenie

i publikowanie dokumentów”), arkusz kalkulacyjny, a także aplikacje do tworzenia baz danych i prezentacji (więcej informacji można znaleźć w rozdziale 11., „Pozostałe składniki pakietu Office: arkusz kalkulacyjny Calc i program do tworzenia prezentacji”).

- ♦ **Dźwięk i obraz:** udostępnia programy do obsługi audio i wideo, m.in. aplikacje do nagrywania dźwięków, konwertowania (legalnego) płyt CD-audio do innych formatów, odtwarzania audio i wideo, a także skomplikowane programy do tworzenia różnorodnych typów płyt CD, VCD i DVD. W rozdziale 13., „Multimedia”, czytelnik znajdzie więcej informacji na temat korzystania z różnych typów programów oraz plików multimedialnych w systemie Ubuntu.
- ♦ **Grafika:** udostępnia programy służące do tworzenia grafiki, wykonywania zrzutów ekranu monitora, obróbki zeskanowanych obrazów, a także zgrabny program do przeglądania zgromadzonych plików. W rozdziale 12. „Praca z grafiką”, czytelnik znajdzie więcej informacji na temat oprogramowania graficznego w Ubuntu.
- ♦ **Gry:** zawiera dokładnie to, czego można się spodziewać po nazwie. Ubuntu zaprojektowano dla ludzi, więc — oczywiście — można w systemie znaleźć kilka zainstalowanych gier. Więcej informacji na temat gier w Ubuntu czytelnik znajdzie w rozdziale 14., „Zagraasz?”.
- ♦ **Internet:** stąd można uruchomić aplikacje ułatwiające korzystanie z internetu, np. program do wykonywania połączeń telefonicznych (ang. Voice-Over-IP), klienta poczty Evolution, (szerzej omówionego w rozdziale 8., „Obsługa poczty elektronicznej za pomocą programu Evolution”), przeglądarkę internetową Firefox (szerzej omówioną w rozdziale 9., „Surfowanie z Firefoksem”), komunikator internetowy, klienta IRC itd.

Dodatkowo na dole menu *Aplikacje* znajduje się pozycja *Dodaj/Usuń*, dzięki której można łatwo dodawać programy do menu. Zgodnie z informacjami zamieszczonymi poniżej, ta pozycja menu jest zintegrowana z mechanizmem zarządzania pakietami w Ubuntu — jeżeli do systemu dodamy program znajdujący się w repozytoriach, zostanie on pobrany, zainstalowany, a w menu pojawi się nowa pozycja. Całkiem fajnie!

Menu Miejsca

Przy użyciu menu *Miejsca* można szybko uzyskać dostęp do plików, katalogów i innych zasobów systemowych. Po wybraniu odpowiedniego miejsca w systemie czy też urządzenia do przechowywania danych z menu *Miejsca* zostanie uruchomiony menedżer plików Nautilus, a następnie wyświetlona zawartość wybranego miejsca. Menu *Miejsca* umożliwia również wygodny dostęp do wykonywania operacji związanych z danymi, czyli z zapisem na płytach CD i DVD, łączeniem się z serwerem plików czy wyszukiwaniem określonych plików i katalogów. Więcej informacji na temat menedżera plików Nautilus można znaleźć w podrozdziale „Menedżer plików Nautilus — wprowadzenie.”

Zawartość menu *Miejsca* będzie się zmieniać podczas dodawania lub odłączania przenośnych nośników pamięci i korzystania ze zdalnych serwerów plików.



Oprócz skrótów do standardowych miejsc w systemie, w menu *Miejsca* można znaleźć jeszcze dwa podmenu: *Zakładki*, wyświetlający wszystkie miejsca, do których skróty utworzył użytkownik w menedżerze plików Nautilus, oraz *Ostatnie dokumenty*, zawierające listę 10 plików, które były ostatnio otwierane. Te menu znacznie ułatwiają pracę nad tymi samymi plikami za każdym razem, kiedy użytkownik loguje się do systemu.

Menu System

Menu *System* udostępnia opcje, za których pomocą można oddziaływać na GNOME i system Ubuntu, takie jak ustawianie osobistych preferencji wyglądu i działania GNOME oraz wykonywania zadań administracyjnych (konfigurowanie sprzętu, uaktualnianie systemu, konfigurowanie usług sieciowych, uzyskiwanie pomocy, wylogowywanie się oraz wyłączanie komputera).

W menu *System* znajdują się następujące pozycje:

- ♦ **Preferencje:** z tego miejsca można uzyskać dostęp do ogromnej ilości programów, dzięki którym można zmodyfikować sposób działania GNOME i Ubuntu czy np. dostępne połączenia sieciowe (np. z internetem czy siecią domową). Jeżeli użytkownikowi nie podoba się sposób, w jaki działa jakaś część systemu, wówczas warto sprawdzić, czy za pomocą programów dostępnych w menu *Preferencje* nie uda się tego zmienić — wystarczy uruchomić odpowiednią aplikację i trochę poeksperymentować. W podrozdziale „Dostosowywanie pulpitu”, znajdującym się poniżej, objaśniam, jak przy użyciu właśnie tego menu wykonać wiele najpowszechniejszych czynności konfiguracyjnych.
- ♦ **Administracja:** z tego miejsca można uruchomić programy służące do konfiguracji aplikacji zainstalowanych w Ubuntu, a także dostępnych w systemie urządzeń, uruchamianych usług i podstawowych informacji, takich jak czas, data, lista użytkowników, lista użytkowników ze specjalnymi uprawnieniami itd.
- ♦ **Pomoc:** znajdują się tu odnośniki do różnych rodzajów pomocy, jaką można uzyskać. Dostępna jest dokumentacja zainstalowana w systemie, jest tu łącze do dokumentacji Ubuntu znajdującej się na stronie zespołu odpowiedzialnego za ten aspekt, a nawet jest łącze do Ubuntu Marketplace (omówionego w rozdziale 1.), gdzie można znaleźć firmy, które pomogą rozwiązać problemy lub skonfigurować system, jeżeli ta książka nie okaże się wystarczająca.
- ♦ **Informacje o GNOME:** korzystając z tej pozycji menu, można wyświetlić informacje na temat używanej wersji GNOME, a także o wszystkich osobach i organizacjach zaangażowanych w różnorodny sposób w sam projekt. Informacje o używanej wersji GNOME będą potrzebne podczas zgłaszania raportu o błędach, zapytań czy kontaktu z firmami, które znaleźć można na Ubuntu Marketplace.
- ♦ **O Ubuntu:** z tego menu można wyświetlić dokumentację Ubuntu znajdującą się w systemie, informacje o samym projekcie i używanej wersji systemu. Podobnie jak w przypadku informacji o wersji środowiska GNOME, zawarte tutaj dane potrzebne będą podczas zgłaszania raportów o błędach, zapytań czy kontaktu z firmami, które znaleźć można na Ubuntu Marketplace.
- ♦ **Zakończ:** po użyciu tej opcji zostanie wyświetlona plansza umożliwiająca wykonanie kilku operacji związanych z dostępem do systemu. Użytkownik może wylogować

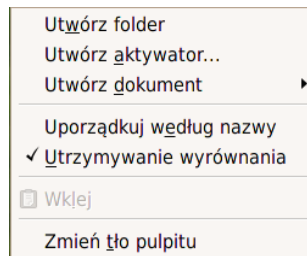
się z systemu (zostanie wówczas uruchomiony menedżer logowania), zablokować ekran (wygodne rozwiązanie, kiedy trzeba na chwilę odejść od komputera i jednocześnie uniemożliwić innym dostęp do pulpitu), przełączyć użytkownika (dzięki temu można łatwo przełączyć się na inne konto bez konieczności wylogowywania się ze swojego konta), hibernować system (opcja przydatna dla użytkowników komputerów z systemem ACPI, umożliwia „zamrożenie” systemu, dzięki czemu jego ponowne uruchomienie zajmuje znacznie mniej czasu), uruchomić system ponownie (opcja przydatna np. po zainstalowaniu istotnych poprawek w systemie) czy w końcu wyłączyć komputer. Więcej informacji na temat tego, co się dzieje podczas uruchamiania i wyłączania systemu można znaleźć w rozdziale 18., „Tworzenie oprogramowania w Ubuntu”.

Menu kontekstowe

O tym menu wspominałem już wcześniej, można je wyświetlić, klikając prawym przyciskiem myszy element ekranu, podmenu itd. Jego zawartość zależy od tego, co aktualnie dzieje się w systemie, i od elementu, który został wskazany podczas kliknięcia myszą. Właśnie ze względu na to, że zawartość tego menu zależy od kontekstu, w jakim zostanie ono wyświetlone, nosi ono nazwę **menu kontekstowego**. Na rysunku 5.4 przedstawiam menu, które jest wyświetlane po kliknięciu pustego pulpitu Ubuntu.

Rysunek 5.4.

Menu kontekstowe
pulpitu GNOME



Na rysunku 5.5 przedstawiam przykład, w jaki sposób menu kontekstowe zależy od otoczenia, w jakim zostało wywołane. Można na nim zaobserwować menu kontekstowe wyświetlane w przeglądarce internetowej Firefox po kliknięciu prawym przyciskiem myszy.

Zawartość menu kontekstowego to funkcje udostępniane przez daną aplikację, a to oznacza, że można tę zawartość dostosowywać poprzez instalowanie dodatkowych programów, apletów czy — jak ma to miejsce w przypadku takich programów jak Firefox czy Nautilus — rozszerzeń lub wtyczek, które dodają nowe możliwości do programów i nowe pozycje w menu.

Dostosowywanie menu

W Ubuntu istnieją dwa sposoby na dostosowywanie zawartości podstawowych menu dostępnych w GNOME. Za pomocą pozycji *Dodaj/Usuń* dostępnej w menu *Aplikacje* można łatwo dodawać nowe programy do każdej z istniejących już pozycji tego menu, co wiąże się też z zainstalowaniem danego programu (jeżeli jeszcze nie ma go w systemie). W menu *Akcesoria* dostępna jest pozycja *Edytor menu Alacarte*, umożliwiającą łatwe



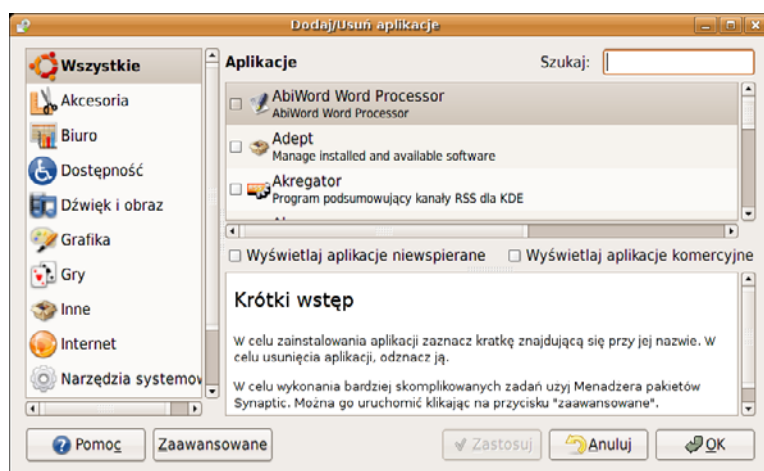
Rysunek 5.5. Menu kontekstowe w przeglądarce internetowej Firefox

do dodawania nowych grup (podmenu), skrótów czy wpisów aplikacji do standardowego menu GNOME, co jednak nie spowoduje zainstalowania jakichkolwiek programów. Poniżej omawiam sposoby korzystania z tych dwóch narzędzi.

Dodawanie programów do menu Aplikacje

Po wybraniu pozycji *Dodaj/Usuń* znajdującej się na dole menu *Aplikacje* zostanie wyświetlone okno dialogowe programu `gnome-app-install` (patrz rysunek 5.6).

Rysunek 5.6.
*Dodawanie aplikacji
dostępnych
w systemie Ubuntu*



Oto elementy okna *Dodaj/Usuń aplikacje*:

- ♦ **Lista kategorii oprogramowania**: wyświetlane są na niej wszystkie zdefiniowane typy oprogramowania rozpoznawane przez program *Dodaj/Usuń*.
- ♦ **Okno *Aplikacje***: lista wszystkich programów dostępnych w aktualnie wybranej kategorii oprogramowania.
- ♦ **Pole *Szukaj***: jeżeli użytkownik nie wie, do jakiej kategorii należy dany program, może wówczas wyszukać go, korzystając z jego nazwy.
- ♦ **Pole wyboru *Wyświetlaj aplikacje niewspierane***: po zaznaczeniu tego pola zostaną wyświetlone programy, które nie są oficjalnie wspierane przez Canonical Ltd, ale są dostępne w repozytoriach *Universe*. Więcej informacji o repozytoriach oraz klasyfikowaniu oprogramowania można znaleźć w rozdziale 11., „Pozostałe składniki pakietu Office: arkusz kalkulacyjny Calc i program do tworzenia prezentacji”.
- ♦ **Pole wyboru *Wyświetlaj aplikacje komercyjne***: podobnie jak w przypadku *Wyświetlaj aplikacje niewspierane*, po zaznaczeniu tego pola zostaną wyświetlone programy, które posiadają ograniczenia dotyczące kopiowania, redystrybuowania, ulepszania i dlatego nie mieszczą się w definicji wolnego oprogramowania Ubuntu i Debiana. Takie oprogramowanie dostępne jest w repozytoriach *Multiverse*. Więcej informacji na ten temat klasyfikowania oprogramowania można znaleźć w rozdziale 11.
- ♦ **Okno opisu programu**: wyświetlane są w nim informacje o aktualnie zaznaczonej aplikacji.
- ♦ **Przyciski**: umożliwiają wykonanie następujących czynności:
 - ♦ ***Pomoc***: zostanie wyświetlona pomoc na temat aplikacji `gnome-app-install` dostępna w systemie.
 - ♦ ***Zaawansowane***: zostanie uruchomiony Menedżer Pakietów Synaptic, który oferuje większe możliwości od aplikacji `gnome-app-install`.
 - ♦ ***Zastosuj***: po kliknięciu tego przycisku zostaną zainstalowane wszystkie wybrane przez użytkownika programy, a następnie ponownie uruchomiony program `gnome-app-install`.
 - ♦ ***Anuluj***: program zostanie zamknięty bez wprowadzania jakichkolwiek zmian w menu *Aplikacje*. Zmiany wprowadzone przed kliknięciem tego przycisku nie zostaną cofnięte.
 - ♦ ***OK***: kliknięcie tego przycisku spowoduje wprowadzenie zaplanowanych zmian, a następnie wyłączenie programu.

Przyjmijmy, że użytkownik chciałby dodać do menu *Biuro* arkusz kalkulacyjny *Gnumeric*. Oto, co należy wykonać:

1. Wybrać z menu *Aplikacje* pozycję *Dodaj/Usuń*.
2. Wybrać pozycję *Biuro* z listy kategorii oprogramowania, a następnie w oknie *Aplikacje* odszukać pozycję *Gnumeric*. Można też skorzystać z pola wyszukiwania i tam wpisać `gnumeric`, a następnie wcisnąć klawisz *Enter*.

3. Zaznaczyć pole wyboru znajdujące się po lewej stronie pozycji *Gnumeric* w oknie *Aplikacje*.
4. Kliknąć przycisk *Zastosuj*, aby zainstalować program; jeżeli użytkownik nie chce wprowadzać kolejnych zmian, wówczas można kliknąć przycisk *OK*, program zostanie zamknięty.
5. Podać hasło i wcisnąć przycisk *Enter*.

Program `gnome-app-install` automatycznie pobiera i instaluje wybrane oprogramowanie oraz inne pakiety niezbędne do spełnienia zależności. Jeżeli zostaną wykryte jakiegokolwiek konflikty z już zainstalowanymi programami, zostanie wyświetlone okno dialogowe, w którym użytkownik będzie mógł usunąć pakiety będące przyczyną kłopotów.

Jak wspominałem wcześniej, program *Dodaj/Usuń* oferuje wygodny mechanizm instalowania większości standardowych pakietów. W przypadku bardziej skomplikowanego oprogramowania należy użyć menedżera Synaptic, jednak podczas instalowania popularnego oprogramowania trudno przebić prostotę użytkowania omawianego programu.

Dodawanie nowych pozycji do innych menu

Omówiony wcześniej program `gnome-app-install` jest niesłychanie wygodny, gdy trzeba dodać kolejne programy do menu *Aplikacje*, ale nie jest już tak przydatny, kiedy chcemy dodać własne menu lub dostosować już istniejące wpisy dla jakiejś aplikacji. Aby to zrobić, należy z menu *Aplikacje* wybrać pozycję *Edytor menu Alacarte*, który umożliwi dodanie niemal dowolnej pozycji do istniejących menu i podmenu w GNOME.

Oto przykład. Przypuśćmy, że użytkownik chciałby utworzyć swoje własne menu o nazwie *Ulubione* i umieścić w nim odwołania do kilku ulubionych programów. Aby to wykonać, (użyjemy edytora `emacs` jako przykładu), należy kolejno:

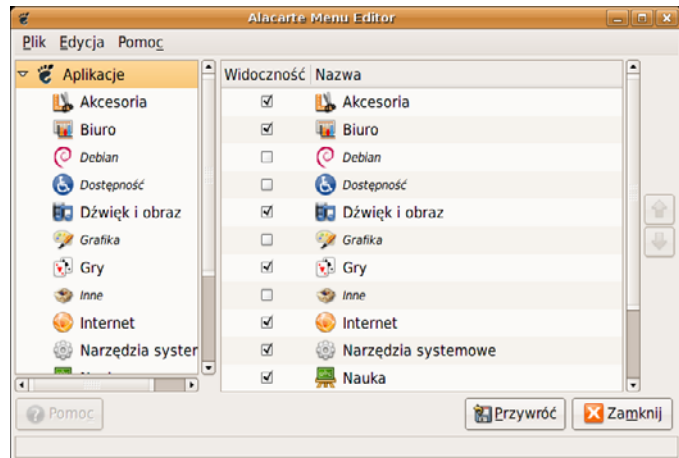
1. Uruchomić program, wybierając kolejno z menu *Aplikacje/Akcesoria/Edytor menu Alacarte*. Ten edytor menu ma dwa główne okna — jedno z głównymi pozycjami menu, a drugie ze szczegółową zawartością każdej pozycji. Jeżeli menu *Aplikacje* wyświetlane jest w wersji skróconej i w oknie ze szczegółową zawartością nie widać wszystkiego, wówczas wystarczy kliknąć mały trójkąt znajdujący się w oknie po lewej stronie obok każdej pozycji; umożliwi on zwiłanie i rozwijanie menu. Edytor menu powinien wyglądać tak jak na rysunku 5.7.



Użytkownik zapewne zauważył, że niektórych pozycji wyświetlanych w oknie po lewej stronie nie widać w menu systemu, dzieje się tak, dlatego że nie zostały oznaczone jako widoczne w panelu po prawej stronie (patrz rysunek 5.7).

2. Wybrać z menu kolejno *Plik/Nowe menu*. Zostanie wyświetlone następne okno dialogowe (patrz rysunek 5.8).
3. Wprowadzić nazwę dla nowo tworzonego menu (w tym przykładzie użyto nazwy *Ulubione*), można też dodać treść komentarza, który będzie wyświetlany po najechaniu kursora na menu, a następnie kliknąć *OK*. Nowe menu zostanie dodane do menu *Aplikacje*, zostanie również zaktualizowana lista pozycji wyświetlanych w oknie programu *Alacarte*. Następnie należy zaznaczyć nowo

Rysunek 5.7.
Edytor menu Alacarte
w systemie Ubuntu

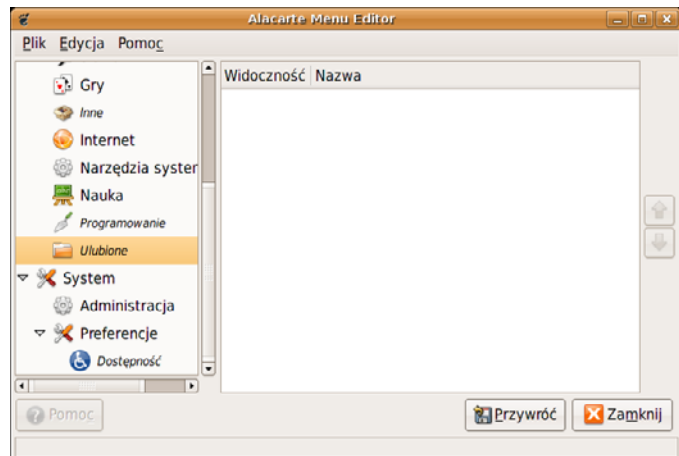


Rysunek 5.8.
Dodawanie nowego
menu do systemu
Ubuntu



utworzoną pozycję, dzięki czemu w prawym oknie zostanie wyświetlona jej zawartość, chociaż w chwili obecnej nie będzie tam żadnych wpisów (patrz rysunek 5.9).

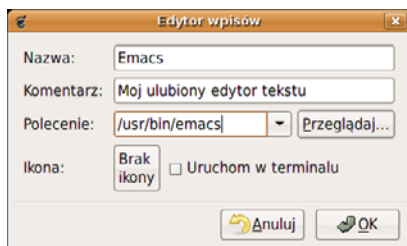
Rysunek 5.9.
Nowe menu
w systemie Ubuntu



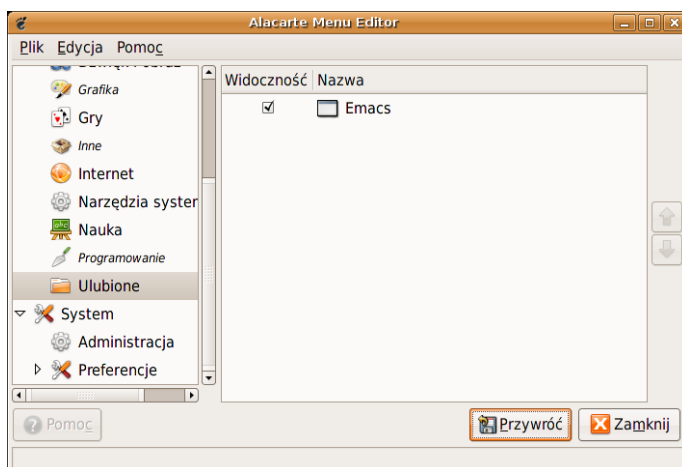
4. Wybrać z menu *Plik/Nowa pozycja*. Zostanie wyświetlone kolejne okno dialogowe (patrz rysunek 5.10).
5. Wprowadzić informacje dotyczące nowego wpisu (w tym przypadku jest to edytor emacs) i kliknąć przycisk *OK*. Okno dialogowe zostanie zamknięte, a w menu *Ulubione* pojawi się nowy wpis (patrz rysunek 5.11).

Rysunek 5.10.

Dodawanie nowego wpisu do menu w systemie Ubuntu

**Rysunek 5.11.**

Nowa pozycja w menu

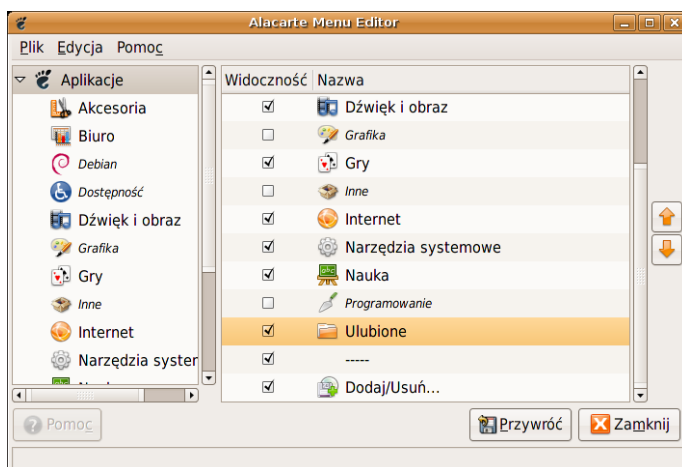


Jeżeli użytkownik nie wie, gdzie dokładnie jest zlokalizowana aplikacja, która ma zostać dodana, wówczas należy kliknąć przycisk *Przeglądaj*, spowoduje to otwarcie okna przeglądarki plików, dzięki czemu będzie można dokładnie wskazać położenie wybranego programu.

- Upewnić się, czy zostało zaznaczone pole wyboru *Widoczność* przy nowo utworzonym wpisie (patrz rysunek 5.12). To zaznaczenie jest konieczne; jeżeli go nie będzie, wówczas w menu *Aplikacje* nie zostanie wyświetlona pozycja *Ulubione*.

Rysunek 5.12.

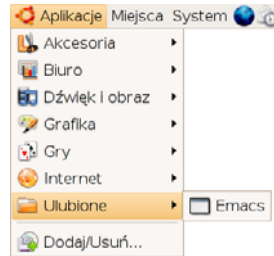
Należy upewnić się, czy nowe menu będzie widoczne



7. Kliknąć przycisk *Zamknij*, spowoduje to zakończenie działania programu Alacarte oraz wprowadzanie wszystkich zmian do menu. Menu *Aplikacje* zostanie zaktualizowane, a nowo dodany wpis stanie się dostępny (patrz rysunek 5.13).

Rysunek 5.13.

Nowa pozycja
w menu *Aplikacje*
w GNOME



Jeżeli użytkownik zdecyduje, by nie zapisywać wprowadzonych zmian, wówczas należy kliknąć przycisk *Przywróć*. Spowoduje to przywrócenie domyślnych ustawień; następnie trzeba kliknąć *Zamknij*.

Dzięki elastyczności takiego sposobu dodawania nowych menu, łatwo umieszczać kolejne wpisy dla kompilowanych samodzielnie programów, dodawać skróty do skryptów powłoki itd. (patrz rysunek 5.10).



Nowe menu w GNOME można dodawać też przy użyciu menedżera plików Nautilus, choć — oczywiście — używanie aplikacji zaprojektowanej do edytowania menu jest łatwiejsze do opanowania. Więcej informacji na temat tworzenia menu za pomocą programu Nautilus można znaleźć w dokumentacji GNOME znajdującej się w systemie.

Dostosowywanie pulpitu

Już wcześniej pisałem, że dzięki GNOME można łatwo dostosowywać menu i podmenu, ale tak naprawdę każda część tego środowiska graficznego jest prosta w konfigurowaniu. Poniżej objaśniam, w jaki sposób wykonać większość czynności, które użytkownicy zwykle podejmują w celu dostosowania i spersonalizowania GNOME. Liczba dostępnych możliwości jest bardzo duża, ale aby omówić każdą z nich, potrzebna byłaby osobna książka! Przyjrzyjmy się zatem najważniejszym sprawom.

Dostosowywanie działania myszy

Jedną ze zmian najczęściej dokonywanych przez użytkowników jest dostosowanie sposobu działania myszy. Oto dostępne opcje.

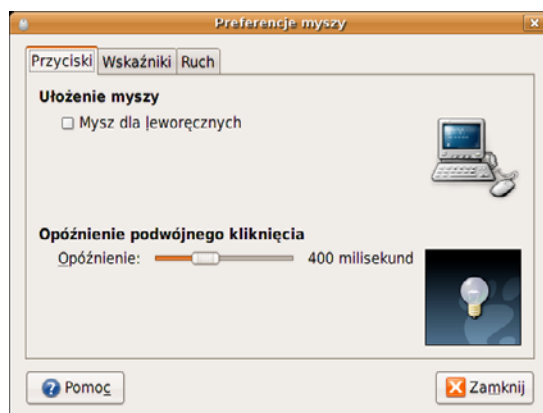
- ♦ Konfiguracja działania dla użytkowników praworęcznych lub leworęcznych. Pierwsza opcja jest włączona domyślnie, ale może nie odpowiadać użytkownikom leworęcznym lub tym, którzy używają lewej ręki do obsługi tego urządzenia peryferyjnego.
- ♦ Prędkość poruszania się kursora po ekranie komputera.
- ♦ Czas, jaki może upłynąć pomiędzy dwoma kliknięciami przycisku myszy, aby zostały one zidentyfikowane jako podwójne kliknięcie (czyli takie, które

np. otwiera folder), a nie pojedyncze (kiedy jakiś element jest tylko zaznaczany). Jeżeli użytkownik ma trudności z szybkim klikaniem, wówczas może zwiększyć długość trwania przerwy pomiędzy pierwszym i drugim kliknięciem.

Wszystkie te aspekty łatwo ustawimy w oknie *Preferencje myszy*, które można uruchomić, wybierając z menu *System/Preferencje/Mysz* (patrz rysunek 5.14).

Rysunek 5.14.

Okno dialogowe
Preferencje myszy

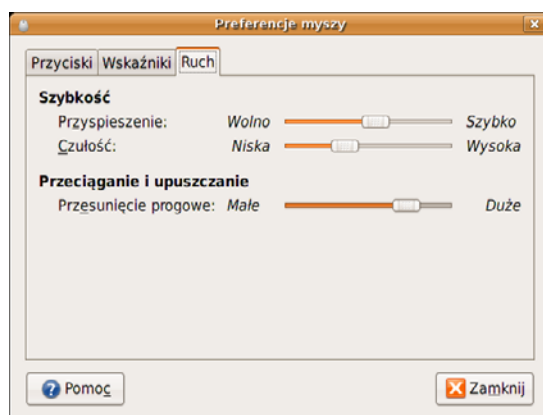


W oknie dostępne są trzy zakładki umożliwiające konfigurowanie różnych aspektów korzystania z myszy, takie jak działanie przycisków i kursora (określanego też nazwą **wskaźnik myszy**) oraz prędkość i czułość urządzenia. Zakładka *Przyciski* widoczna jest na rysunku 5.14. Aby przejść do zakładki znajdującej się obok, wystarczy wskazać ją kursorzem i wcisnąć przycisk myszy.

Ponieważ opcje konfiguracyjne dostępne w tym oknie są proste, nie będziemy ich tu omawiać. Kilku słów wymaga jedynie zakładka umożliwiająca konfigurowanie ruchu i czułości działania urządzenia. Opcje te zostały umieszczone w zakładce *Ruch* okna dialogowego *Preferencje myszy* (patrz rysunek 5.15).

Rysunek 5.15.

Konfigurowanie
ruchu myszy



Oto opis dostępnych tam opcji.

- ♦ **Przyspieszenie:** opcja odpowiada za to, jak szybko kursor myszy porusza się po ekranie, reagując na ruchy użytkownika; można ją kontrolować za pomocą suwaka. Dzięki tej możliwości można uniknąć sytuacji, w której podkładka pod mysz musiałaby być rozmiarów monitora — przyspieszenie umożliwia pokrycie większego obszaru przy jednostajnym ruchu myszą. Przesunięcie suwaka bliżej pozycji *Wolno* oznacza, że droga pokonywana przez kursor będzie bliższa drodze pokonywanej przez mysz; natomiast umieszczenie suwaka bliżej pozycji *Szybko* spowoduje, że ruchy myszą będą mogły być krótsze. Innymi słowy, kursor będzie przyspieszał wraz z ciągłym ruchem myszy. Aby zmienić to ustawienie, należy kliknąć suwak i przeciągnąć go do odpowiedniej pozycji.
- ♦ **Czułość:** ta opcja decyduje o czułości reakcji kursora na ruchy myszą. Jeżeli ustawiona wartość *Przyspieszenia* będzie wysoka, wówczas kursor może „skakać” po ekranie. Aby zniwelować to zjawisko, wystarczy zmniejszyć czułość myszy.
- ♦ **Przeciąganie i upuszczanie:** ustawienie decyduje o tym, jaki dystans musi pokonać kursor myszy po wybraniu danego obiektu, aby rozpocząć operację typu „przeciągnij i upuść”. Przykładowo podczas podwójnego kliknięcia może się zdarzyć, że pomiędzy jednym a drugim wciśnięciem klawisza myszy kursor zostanie nieznacznie przesunięty; jeżeli opcja *Przesunięcie progowe* będzie ustawiona na niskim poziomie, wówczas takie przesunięcie zostanie zinterpretowane jako operacja „przeciągnij i upuść”, a nie podwójne kliknięcie. W takim przypadku należy przesunąć suwak w kierunku pozycji *Duże*.

Zmiany ustawień myszy są wprowadzane natychmiast, a zatem łatwo eksperymentować w poszukiwaniu ustawień najlepiej odpowiadających preferencjom użytkownika.

Konfigurowanie rozdzielczości wyświetlania

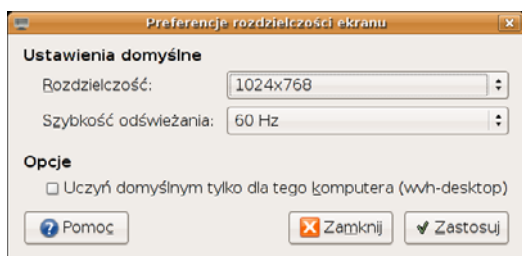
Podczas instalowania systemu Ubuntu jedno z pytań, na które musi odpowiedzieć użytkownik, dotyczy rozdzielczości, która będzie używana w systemie. To miłe, że program instalacyjny zadaje takie pytanie, ale akurat w przypadku rozdzielczości dość trudno wyobrazić sobie, jak będzie wyglądał obraz przy określonych parametrach. W konsekwencji może się okazać, że jeśli nawet użytkownik korzysta z 30-calowego monitora i najlepszej na całym świecie karty graficznej, to Ubuntu i tak nalega na wyświetlanie wszystkiego w rozdzielczości 1024×768, w której ikony na ekranie będą wielkości grejpfrutów. Prawdopodobnie nie takie rozwiązanie miał na myśli użytkownik, a już z pewnością nie jest to najlepsze wykorzystanie posiadanego sprzętu. Poniżej zamieszczam informacje na temat rozwiązywania problemów z wyborem wyższej rozdzielczości. Oznacza to, że monitor może wyświetlać więcej pikseli, a elementy pulpitu będą mniejsze i wyraźniejsze. W przemyśle komputerowym nosi to nazwę *the right thing*TM.

Rozdzielczość ekranu w GNOME można kontrolować w oknie dialogowym *Preferencje rozdzielczości ekranu*, które wywołujemy, wybierając kolejno z menu *System/Preferencje/Rozdzielczość ekranu* (patrz rysunek 5.16).

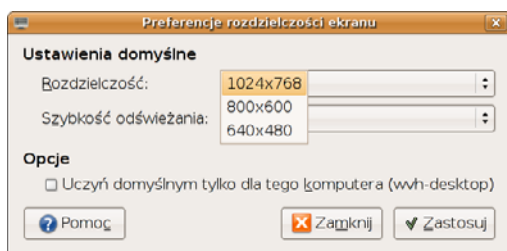
Po wyświetleniu okna można sprawdzić listę dostępnych rozdzielczości, wystarczy tylko rozwinąć listę *Rozdzielczość* (patrz rysunek 5.17).

Rysunek 5.16.

Program
do konfigurowania
rozdzielczości ekranu

**Rysunek 5.17.**

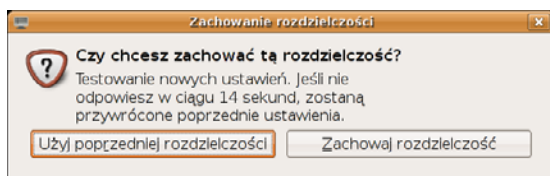
Wybór innej
rozdzielczości



Po wybraniu nowej rozdzielczości należy kliknąć przycisk *Zastosuj*, wówczas zostanie ona wprowadzona. Po dokonaniu zmiany GNOME wyświetli jeszcze okno komunikatu, w którym użytkownik będzie mógł potwierdzić swoją decyzję (patrz rysunek 5.18).

Rysunek 5.18.

Potwierdzenie zmiany
rozdzielczości
wyświetlania



Ponieważ istnieje możliwość, że użytkownik wybierze rozdzielczość, która nie jest obsługiwana, w przypadku braku potwierdzenia zmiany system po upływie 20 sekund automatycznie przywróci poprzednie ustawienia. Jeżeli więc po zmianie ustawień nic nie pojawi się na ekranie, wystarczy odczekać chwilę, a wszystko wróci do normy.

Jeżeli użytkownik jest zadowolony z nowej rozdzielczości, wówczas należy kliknąć przycisk *Zachowaj rozdzielczość*. Okno zostanie zamknięte, nowe ustawienia zachowane, a podczas każdego kolejnego logowania się do systemu właśnie taka rozdzielczość będzie używana do wyświetlania grafiki.

Jeżeli ekran w nowych ustawieniach wygląda dziwnie lub elementy są zbyt małe, wówczas należy kliknąć przycisk *Użyj poprzedniej rozdzielczości*. Następnie można wypróbować inne ustawienia w poszukiwaniu rozdzielczości, która będzie odpowiednia.

Dostosowywanie paneli

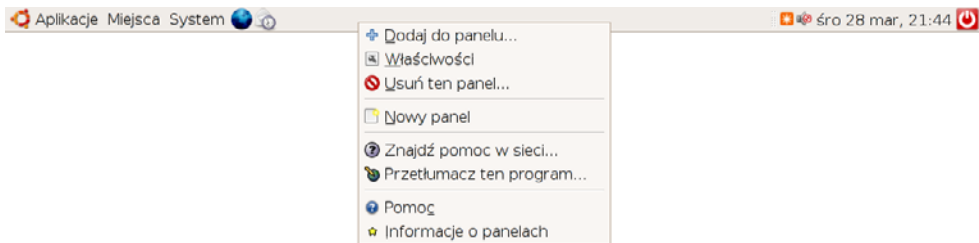
Jak powiedziałem wcześniej, panele to obszary pulpitu, za których wyświetlanie odpowiedzialny jest program `gnome-panel`; umożliwiają one różne dodatkowe sposoby korzystania i uruchamiania programów. Aplikacja `gnome-panel` może wyświetlać panele na każdej krawędzi ekranu, ale domyślne ustawienia w Ubuntu to wyświetlanie paneli przy górnej i dolnej krawędzi.



Jeden uruchomiony program `gnome-panel` jest odpowiedzialny za wyświetlanie wszystkich paneli w systemie, dlatego fakt wyświetlania wielu różnych paneli nie oznacza, że uruchomiono więcej niż jedną kopię programu `gnome-panel`.

W ustawieniach domyślnych panele GNOME są wyświetlane na górze oraz na dole ekranu i znajdują się na nich menu, aplety i przyciski omówione w podrozdziale „Wprowadzenie do GNOME”. Użytkownik może konfigurować nie tylko to, jak i gdzie są wyświetlane panele, ale również dostosowywać zawartość każdego z nich. GNOME w Ubuntu jest instalowane z wieloma różnymi apletami, przyciskami i kontrolkami, które użytkownik może umieścić na panelach.

Wystarczy kliknąć prawym przyciskiem myszy panel, a zostanie wyświetlone menu kontekstowe widoczne na rysunku 5.19.



Rysunek 5.19. Menu kontekstowe programu `gnome-panel`

Oprócz standardowych pozycji, czyli informacji na temat wersji programu oraz dostępu do pomocy systemowej, menu kontekstowe programu `gnome-panel` zawiera również dodatkowe pozycje; oto opis ich działania.

- ♦ **Dodaj do panelu:** umożliwia dodanie do wyświetlanego na pulpicie panelu nowego apletu, przycisku czy innego obsługiwanego elementu. Więcej informacji na ten temat dodawania do paneli nowych elementów można znaleźć w podpunkcie „Dostosowywanie zawartości paneli”.
- ♦ **Właściwości:** wyświetla okno dialogowe *Właściwości panelu*, w którym użytkownik może skonfigurować dany panel. Więcej informacji na ten temat dostosowywania istniejących paneli można znaleźć w podpunkcie „Dostosowywanie właściwości paneli”.
- ♦ **Usuń ten panel:** usuwa z pulpitu wybrany panel, a także wszystkie dane konfiguracyjne znajdujące się w systemie.
- ♦ **Nowy panel:** tworzy nowy panel przy następnej wolnej krawędzi ekranu. Można utworzyć nieskończenie wiele nowych paneli — każdy nowy będzie tworzony (po wyczerpaniu wolnych krawędzi) obok panelu, w którym wyświetlono menu kontekstowe. Uważam, że dwa panele to całkiem odpowiednia liczba, ale użytkownik może mieć własny pogląd na ten temat.

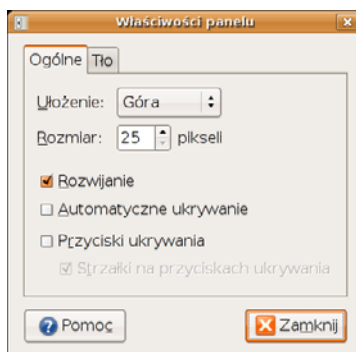
W następnych dwóch podpunktach omówiłem sposób, w jaki można zmodyfikować właściwości paneli ekranowych, dodać do nich aplety, przyciski i kontrolki. Opisałem także najbardziej przydatne i zabawne elementy dodatkowe dostępne w systemie Ubuntu Linux.

Dostosowywanie właściwości paneli

Po wybraniu z menu kontekstowego pozycji *Właściwości* zostanie wyświetlone okno dialogowe widoczne na rysunku 5.20.

Rysunek 5.20.

Okno dialogowe
programu
gnome-panel
Właściwości panelu



To okno dialogowe posiada dwie zakładki: *Tło*, dzięki której użytkownik może decydować o kolorze i stopniu przezroczystości danego panelu lub też wybrać grafikę wykorzystaną jako jego tło, oraz *Ogólne* (patrz rysunek 5.20), która umożliwia ustalenie pozycji oraz działanie panelu. Oto własności, które użytkownik może modyfikować.

- ♦ **Ułożenie:** umożliwia określenie krawędzi, przy której panel ma być wyświetlany. Zmiana krawędzi spowoduje natychmiastowe przeniesienie panelu do nowej lokalizacji.



Panel można przenieść w nowe miejsce również w inny sposób: wystarczy tylko kliknąć i przytrzymać lewy przycisk myszy z kursorem umieszczonym na tle panelu, a następnie przeciągnąć do innej krawędzi.

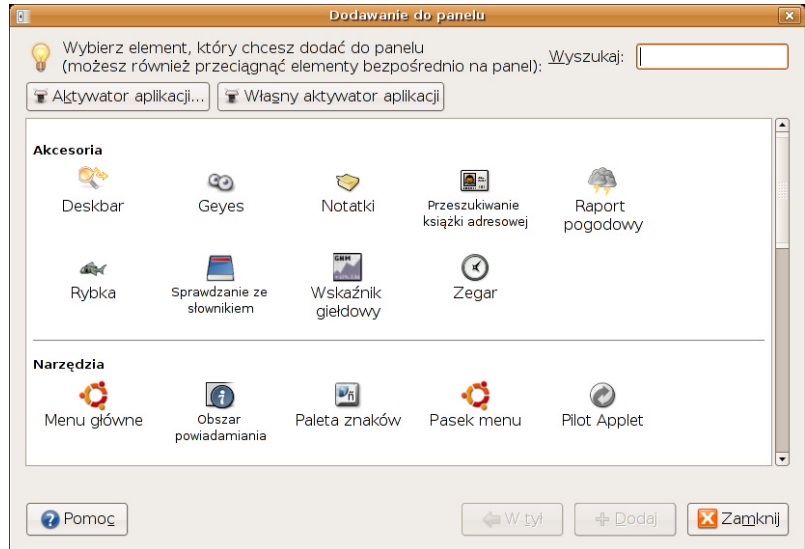
- ♦ **Rozmiar:** umożliwia określenie podawanej w pikselach wysokości (lub szerokości w przypadku paneli zorientowanych pionowo) danego panelu.
- ♦ **Rozwijanie:** zaznaczenie lub anulowanie zaznaczenia tego pola decyduje o tym, czy panel będzie zajmował całą szerokość ekranu, czy tylko obszar niezbędny do wyświetlenia znajdujących się na nim elementów.
- ♦ **Automatyczne ukrywanie:** zaznaczenie lub anulowanie zaznaczenia tego pola decyduje o tym, czy panel będzie wyświetlany tylko w chwili, kiedy kursor myszy zostanie przesunięty do odpowiedniej krawędzi ekranu. Jeżeli pole nie jest zaznaczone, wówczas panel będzie wyświetlany przez cały czas.
- ♦ **Przyciski ukrywania:** zaznaczenie lub anulowanie zaznaczenia tego pola decyduje o tym, czy na każdym z końców panelu zostaną umieszczone przyciski, za których pomocą będzie można ukryć (zwinąć) dany panel. Po zaznaczeniu tej opcji zostanie dodatkowo aktywowana opcja *Strzałki na przyciskach ukrywania*, dzięki której można łatwo określić kierunek, w który zostanie przesunięty i ukryty dany panel.

Zmiany dokonywane we właściwościach paneli wprowadzane są natychmiast. Aby zamknąć okno dialogowe *Właściwości panelu*, należy kliknąć przycisk *Zamknij*.

Dostosowywanie zawartości paneli

Po wybraniu z menu kontekstowego opcji *Dodaj do panelu* zostanie wyświetlone okno dialogowe widoczne na rysunku 5.21.

Rysunek 5.21.
Okno dialogowe
Dodawanie do panelu



To okno dialogowe umożliwia dodawanie aktywatorów aplikacji, przycisków i apletów do każdego panelu wyświetlanego na ekranie monitora. Wszystkie dostępne przyciski i aplety są wyświetlane w oknie, które można przewijać. Aby dodać któryś z tych elementów do widocznego panelu, należy najechać na niego kursorem, wcisnąć i przytrzymać lewy przycisk myszy, a następnie przesunąć element na panel w miejsce, w którym powinien się znaleźć. Po zwolnieniu przycisku myszy element zostanie umieszczony na nowym miejscu.



Aplety po dodaniu ich na którykolwiek z paneli można przesunąć metodą „przeciągnij i upuść” lub za pomocą prawego przycisku myszy, wciskając go i wybierając z menu kontekstowego *Przesuń*. Po wybraniu opcji *Przesuń* aplet będzie się przemieszczał zgodnie z ruchami kursora myszy. Po umieszczeniu go w odpowiednim miejscu należy kliknąć lewy przycisk myszy, wówczas aplet zostanie zablokowany. Aplety mogą być przenoszone pomiędzy różnymi panelami wyświetlanymi na ekranie monitora.

Aplety wyświetlane w oknie dialogowym *Dodawanie do panelu* są zorganizowane w cztery główne kategorie: *Akcesoria*, *Narzędzia*, *Pulpit i okna*, *System i sprzęt*. Oto krótki opis kilku moich ulubionych apletów.

- ♦ ***Sprawdzanie ze słownikiem***: umieszczony w kategorii *Akcesoria* umożliwia szybki i prosty sposób wyszukiwania słów w internetowym słowniku.
- ♦ ***Szuflada***: umieszczona w kategorii *Pulpit i okna* to świetny sposób na korzystanie z wielu apletów czy przycisków bez zbędnego zużywania przestrzeni na panelu. W istocie *Szuflada* jest rozszerzeniem panelu, w którym można przechowywać inne obiekty. Kliknięcie jej spowoduje rozwinięcie i wyświetlenie zawartości, dzięki czemu użytkownik będzie mógł uruchomić daną aplikację tak, jak gdyby znajdowała

się na panelu. Jeżeli użytkownik często korzysta z apletów, ale nie chce mieć zbyt wielu paneli, wówczas *Szuflada* będzie znakomitym rozwiązaniem.

- ♦ **Notatki:** aplet umieszczony w kategorii *Akcesoria* umożliwia tworzenie notatek i zarządzanie nimi; *Notatki* są elektronicznym odpowiednikiem znanych ze służbowych biurków żółtych, samoprzylepnych karteczek.
- ♦ **Monitor systemu:** aplet umieszczony w kategorii *System i sprzęt*, można go skonfigurować do graficznego prezentowania informacji na temat procesora, pamięci, sieci, przestrzeni wymiany i stopnia wykorzystania dysku komputera.
- ♦ **Regulacja głośności:** aplet umieszczony w kategorii *System i sprzęt*, umożliwia łatwe zwiększanie lub zmniejszanie głośności dźwięku. Ten aplet jest domyślnie instalowany w systemie Ubuntu.
- ♦ **Raport pogody:** aplet umieszczony w kategorii *Akcesoria*, umożliwia uzyskiwanie informacji o temperaturze i innych warunkach pogodowych dla określonej lokalizacji. Ponieważ wielu użytkowników w swoich biurach nie ma okien (niektórzy w ogóle nie mają biur), to narzędzie może być bardzo przydatne, kiedy trzeba wyjść na zewnątrz. Trochę nieprzyjemna jest myśl, ile razy ten aplet okazał się przydatny, bo można by było stwierdzić: „O, na zewnątrz znowu pada śnieg”.



Każdy aplet panelu posiada własne opcje konfiguracyjne; np. w *Raporcie pogodowym* należy określić własne położenie, tak aby podawane informacje były przydatne. Te dane jednak nie zostaną wykorzystane przez inne aplikacje. Aby skonfigurować jakikolwiek aplet znajdujący się na panelu, należy wcisnąć prawy przycisk myszy (wskaźnik myszy musi znajdować się nad danym apletem), a następnie z menu kontekstowego wybrać opcję *Preferencje*. Zostanie wówczas wyświetlone okno dialogowe z dostępnymi opcjami konfiguracyjnymi.

Aktywatory aplikacji to wpisy umieszczone na panelu, które uruchamiają określony program. Ubuntu posiada kilka wstępnie zdefiniowanych aktywatorów, które można umieścić na panelach i za ich pomocą uruchamiać określone programy. Aby dodać do panelu jeden ze skonfigurowanych w systemie aktywatorów, należy wcisnąć przycisk *Aktywator aplikacji* znajdujący się w oknie dialogowym *Dodawanie do panelu*. Zostanie wówczas wyświetlone okno dialogowe przedstawione na rysunku 5.22.

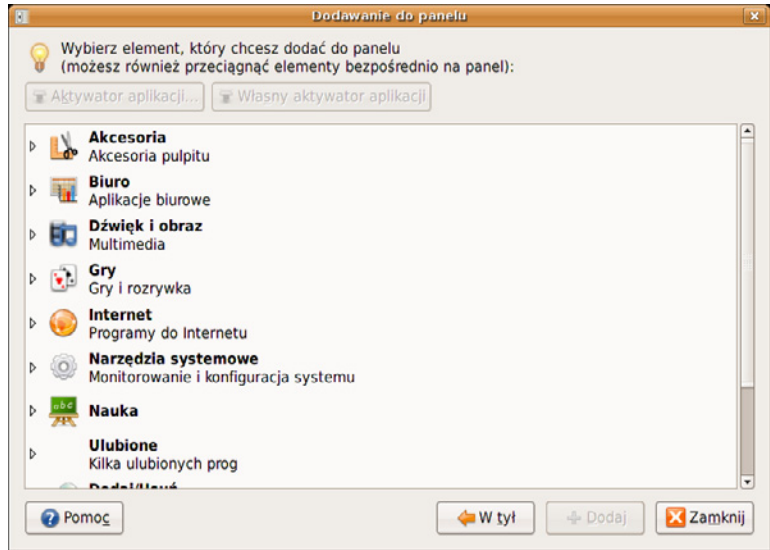
Aby dodać wybrany aktywator aplikacji, wystarczy wybrać go z rozwijanej listy znajdującej się w oknie dialogowym. Aby powrócić do głównego okna, należy wcisnąć przycisk *W tył*.

Aplikacja *Dodaj do panelu* umożliwia też użytkownikom dodawanie samodzielnie skonfigurowanych aktywatorów aplikacji. Dzięki niej można dodać do panelu wpis, który będzie uruchamiał program z opcjami zdefiniowanymi przez użytkownika. Aby dodać do panelu taki aktywator, należy wcisnąć przycisk *Własny aktywator aplikacji* znajdujący się w oknie dialogowym *Dodawanie do panelu*. Zostanie wyświetlone okno dialogowe (patrz rysunek 5.23).

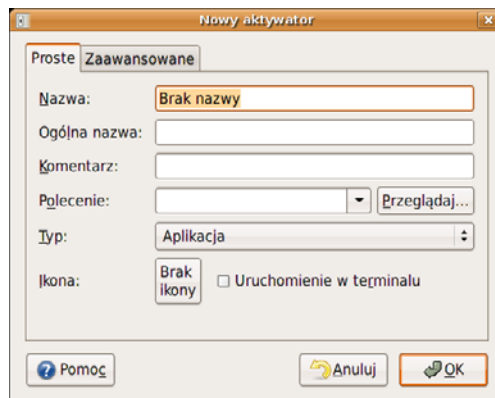
Okno dialogowe *Nowy aktywator* umożliwia ustawienie wielu różnych opcji startowych wybranego programu oraz wyświetlanie samego aktywatora na panelu. Po wprowadzeniu potrzebnych danych należy wcisnąć przycisk *OK*, wówczas aktywator zostanie utworzony. Domyślnie aktywator zostanie umieszczony na panelu, z którego uruchomiono okno *Dodawanie do panelu*.

Rysunek 5.22.

*Dodawanie do panelu
zdefiniowanych
w systemie
aktywatorów aplikacji*

**Rysunek 5.23.**

*Definiowanie
własnych
aktywatorów aplikacji*



Po zakończeniu dostosowywania apletów, przycisków i innych narzędzi do własnych potrzeb, należy wcisnąć przycisk *Zamknij*.

Aby usunąć dany aplet z panelu, trzeba najechać wskaźnikiem myszy na jego nazwę, następnie wcisnąć prawy przycisk myszy i z menu kontekstowego wybrać opcję *Usuń z panelu*.



Konfigurowanie wygaszacza ekranu

W dawnych czasach monitory komputerów były wrażliwe na wyświetlanie tego samego obrazu przez dłuższy czas. Jeżeli monitor wyświetlał tę samą aplikację przez długi okres, wówczas uszkodzeniu mógł ulec luminofor znajdujący się wewnątrz monitora, co z kolei prowadziło do pojawiania się na ekranach „obrazów-duchów” (dany obraz był widoczny na ekranie, mimo że program wyświetlał już zupełnie inną zawartość). Aby wyeliminować lub przynajmniej zredukować ten problem, wiele firm zajmujących się oprogramowaniem projektowało programy noszące nazwę wygaszaczy ekranu, które były

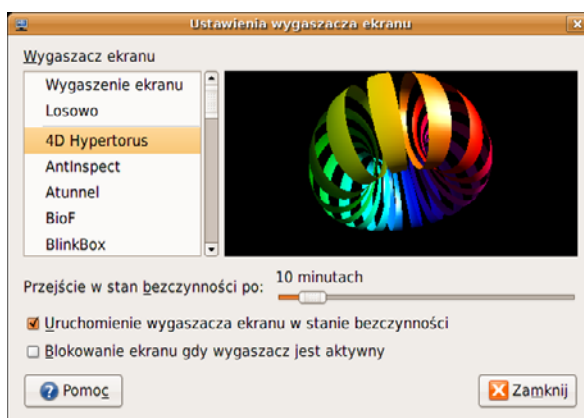
automatycznie uruchamiane, jeżeli system przez dłuższy czas był nieużywany. Programy te wyświetlały na monitorach różne obrazy, co zapobiegało wypalaniu luminoforu i uszkodzeniom urządzenia.

Produkowane obecnie monitory są o wiele bardziej zaawansowane i dużo bardziej odporne na tego rodzaju uszkodzenia. Obecnie wygaszacze ekranu są zwykle używane z powodów estetycznych lub ze względów bezpieczeństwa, ponieważ zabezpieczają komputery przed dostępem niepowołanych użytkowników.

Domyślne ustawienia w Ubuntu powodują uruchomienie wygaszacza ekranu po dziesięciu minutach braku aktywności systemu. Użytkownik może samodzielnie wybrać wygaszacz, który ma być wyświetlany (są ich setki), określić czas, po jakim ma zostać włączony, i zdecydować, czy wraz z jego uruchomieniem ma zostać zablokowany dostęp do systemu. Wszystkie te opcje można skonfigurować w oknie dialogowym *Ustawienia wygaszacza ekranu*, dostępnym w menu *System/Preferencje/Wygaszacz ekranu* (patrz rysunek 5.24).

Rysunek 5.24.

Konfigurowanie
wygaszacza ekranu



Kiedy to okno zostanie wyświetlone, użytkownik może przejrzeć wszystkie wygaszacze ekranu: ich lista dostępna jest w części okna o nazwie *Wygaszacz ekranu*. W prawej części okna wyświetlany jest podgląd działania zaznaczonego na liście wygaszacza.

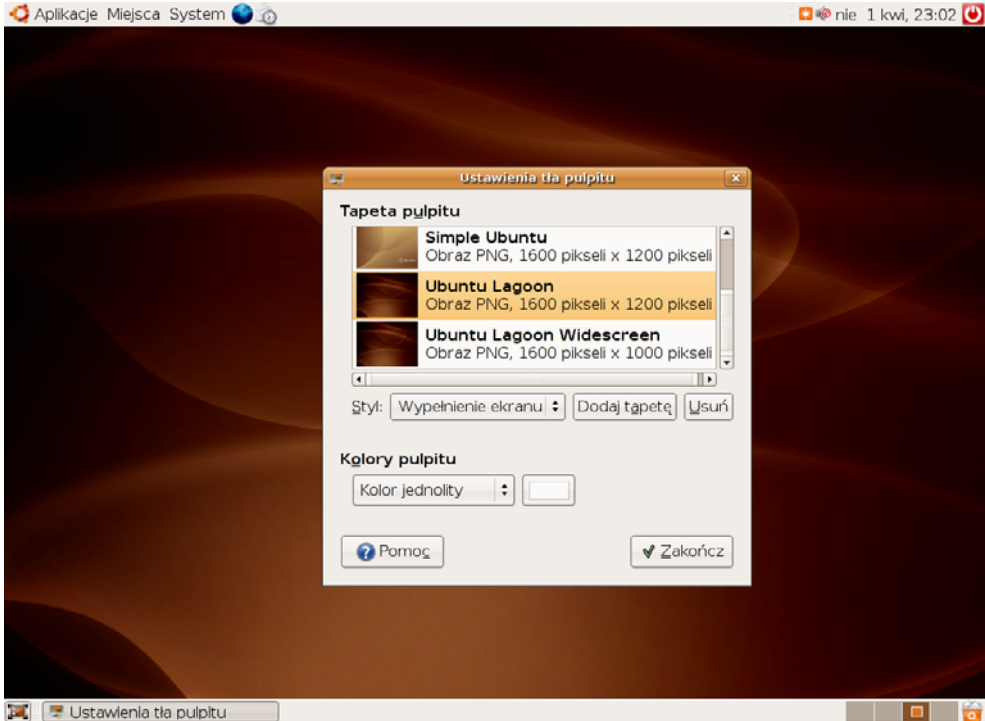
Po wybraniu odpowiedniego rodzaju wygaszacza można skonfigurować jego parametry działania dostępne w oknie *Ustawienia wygaszacza ekranu*. Aby zmienić czas, po jakim wygaszacz jest aktywowany, należy przesunąć w odpowiednim kierunku suwak znajdujący się w polu *Przejście w stan bezczynności po*. Aby całkowicie wyłączyć wygaszacz, należy usunąć zaznaczenie pola *Uruchomienie wygaszacza ekranu w stanie bezczynności*.

Poruszenie myszy lub wciśnięcie dowolnego klawisza spowoduje zakończenie działania wygaszacza oraz wyzerowanie jego licznika. Aby skorzystać z możliwości połączenia blokowania systemu z uruchamianiem wygaszacza ekranu, należy zaznaczyć opcję *Blokowanie ekranu gdy wygaszacz jest aktywny*. Aby wyświetlić ponownie pulpit, konieczne będzie podanie hasła dostępu do systemu.

Po zakończeniu konfiguracji wygaszacza ekranu należy wcisnąć przycisk *Zamknij*, okno dialogowe zostanie wyłączone.

Zmiana tła pulpitu

GNOME umożliwia również proste zmienianie tła pulpitu wyświetlanego w systemie. Tło to stały kolor lub obraz wyświetlany na pulpicie, który widać pod ikonami czy uruchomionymi programami. Aby dokonać jego zmiany, należy wybrać kolejno z menu *System/Preferencje/Tło pulpitu*, zostanie wówczas wyświetlone okno dialogowe *Ustawienia tła pulpitu* (patrz rysunek 5.25).



Rysunek 5.25. Konfigurowanie tła pulpitu

Zmiana tła pulpitu jest jedną z modyfikacji najczęściej dokonywanych przez użytkowników podczas dostosowywania systemu do własnych potrzeb. Dostosowując pulpit, można zmienić kolor tła, wybrać obraz z listy *Tapeta pulpitu* lub wcisnąć przycisk *Dodaj tapetę*, dzięki czemu będzie można wskazać określony plik graficzny, który ma zostać wyświetlony (np. zdjęcie z wakacji).

Po wybraniu obrazu należy określić styl jego wyświetlania; oto dostępne możliwości.

- ♦ **Wyśrodkowany:** wybrana tapeta będzie wyświetlana w samym środku ekranu w oryginalnej rozdzielczości.
- ♦ **Wypełnienie ekranu:** wybrana tapeta zostanie powiększona do takich rozmiarów, aby wypełniła cały ekran monitora. Oryginalne proporcje mogą nie zostać zachowane.
- ♦ **Przeskalowany:** wybrana tapeta zostanie rozciągnięta tak, aby wypełnić cały ekran monitora z zachowaniem oryginalnych proporcji.

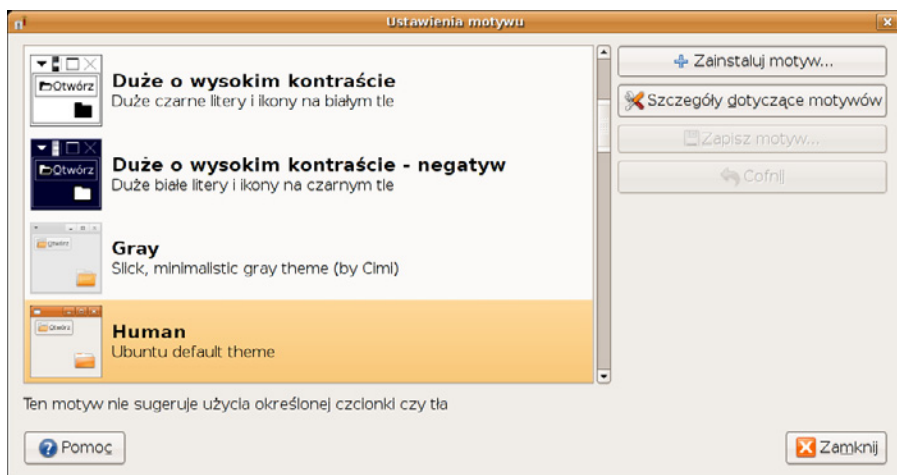
- ♦ **Powiększenie:** wybrana tapeta wypełni cały obraz. Jeżeli obraz nie będzie miał dokładnie takich samych rozmiarów jak monitor, wówczas pewne jego części mogą zostać przycięte.
- ♦ **Sąsiadujaco:** wybrana tapeta zostanie wielokrotnie zduplikowana na ekranie monitora w swoim oryginalnym rozmiarze, tak aby całkowicie go wypełnić.

Po wybraniu odpowiedniej grafiki, stylu jej wyświetlania i innych opcji zmiany zostaną wprowadzone natychmiast. Jeżeli będą one odpowiadać użytkownikowi, może zamknąć program, używając przycisku *Zakończ* znajdującego się w oknie dialogowym *Ustawienia tła pulpitu*.

Przełączanie motywów

Motyw to określenie dla kombinacji okien i kontroltek, schematu kolorystycznego oraz zestawu ikon, przy którym ustawieniu pulpit oraz okna w systemie działają i są wyświetlane w określony sposób. Również motywy posiadają swoje własne tapety pulpitu, które pomagają osiągnąć to, co zamierzył sobie autor motywu. Eksperymentowanie z różnymi motywami może być dobrą zabawą, ale jest również bardzo przydatne, ponieważ może poprawić czytelność zawartości ekranu po dobraniu odpowiednich wielkości elementów oraz kontrastu.

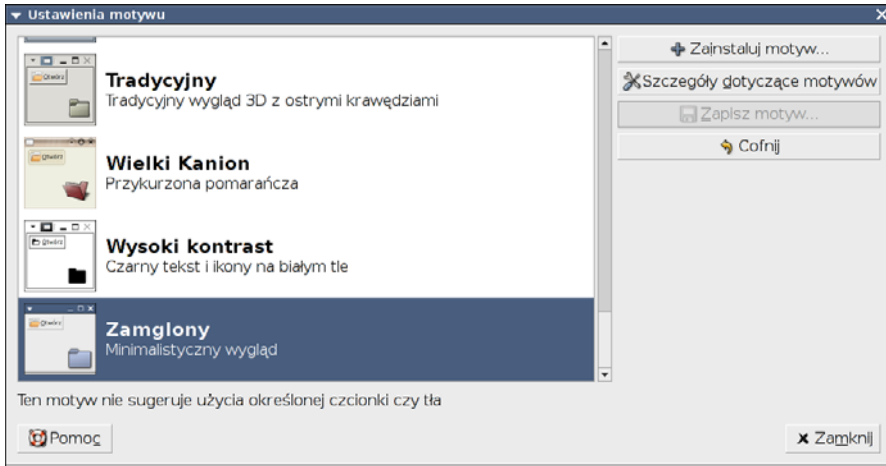
Aby zmienić konfigurację używanego w Ubuntu motywu, należy z menu wybrać *System/Preferencje/Motyw*, zostanie wówczas wyświetlone okno dialogowe *Ustawienia motywu* widoczne na rysunku 5.26.



Rysunek 5.26. Określanie ustawień motywu

GNOME w systemie Ubuntu zawiera wiele atrakcyjnych motywów. Domyślnie używany jest temat o nazwie Human, który został zaprojektowany tak, aby był atrakcyjny, przyjazny dla oczu oraz łatwy w użytkowaniu. Aby zastosować inny temat dostępny w Ubuntu, należy wybrać któryś z wyświetlanych na liście znajdujących się w lewej części okna *Ustawienia motywu*. W tym oknie można zobaczyć podgląd każdego z tematów, ale i tak

zmiany będą wprowadzane natychmiast, dzięki czemu użytkownik od razu będzie mógł zweryfikować ostateczny efekt. Na rysunku 5.27 przedstawiam okno dialogowe po zmianie tematu na *Mist*. Choć tego nie widać, użytkownik może uwierzyć mi na słowo, że teraz tło dekoracji okien jest niebieskie, widać natomiast różnice w rozmieszczeniu czcionek, kontrolki okna, kształtu przycisków systemowych itd.



Rysunek 5.27. Wybór nowego tematu

Po wybraniu tematu, który będzie najodpowiedniejszy w danej chwili, należy wcisnąć przycisk *Zamknij*; okno dialogowe *Ustawienia motywu* zostanie zamknięte, a nowy temat będzie teraz na stałe używany w systemie (dopóki użytkownik go nie zmieni).



Chociaż w Ubuntu znajduje się duża kolekcja przyjemnych tematów, to artystycznie uzdolnieni użytkownicy wciąż tworzą nowe. Wiele znakomitych (lub po prostu interesujących) można znaleźć na stronach internetowych. Moje ulubione strony to GNOME Art (<http://art.gnome.org/>), GNOME-Look.org (www.gnome-look.org/) i sekcja poświęcona tematom na Freshmeat (<http://themes.freshmeat.net/browse/58/>). Krótkie poszukiwanie w sieci z hasłem w wyszukiwarce „GNOME tematy” (lub w wersji angielskiej „GNOME themes”) przyniesie kolejne strony. Aby umieścić nowy temat w systemie, należy go pobrać, a następnie użyć przycisku *Zainstaluj motyw*, po czym wskazać położenie pliku. Temat zostanie dodany do listy i będzie można go używać tak jak pozostałych. Gdzieś tam na pewno jest ten idealny temat!

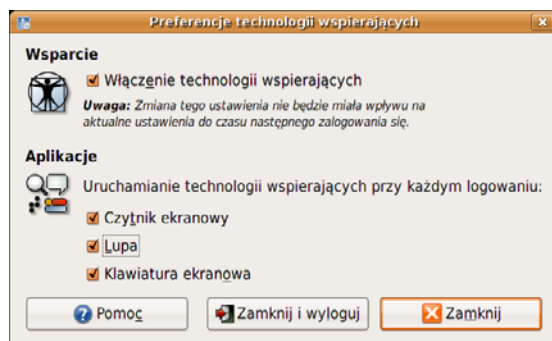
Technologie wspierające w GNOME

Programiści z Ubuntu uwzględnili potrzeby osób, które mogą mieć problemy z odczytywaniem zawartości wyświetlanych na monitorach czy używaniem klawiatury i myszy. GNOME i Ubuntu wyposażone są w *technologie wspierające*, dzięki którym łatwiej używać komputera, czy to dzięki powiększeniu zawartości ekranu, klawiaturze ekranowej czy odczytywaniu głosowych komunikatów. Te funkcje dostępne są w pakietach *gok* (GNOME On-Screen Keyboard), *gnopernicus* i *gnome-mag* instalowanych standardowo w systemie.

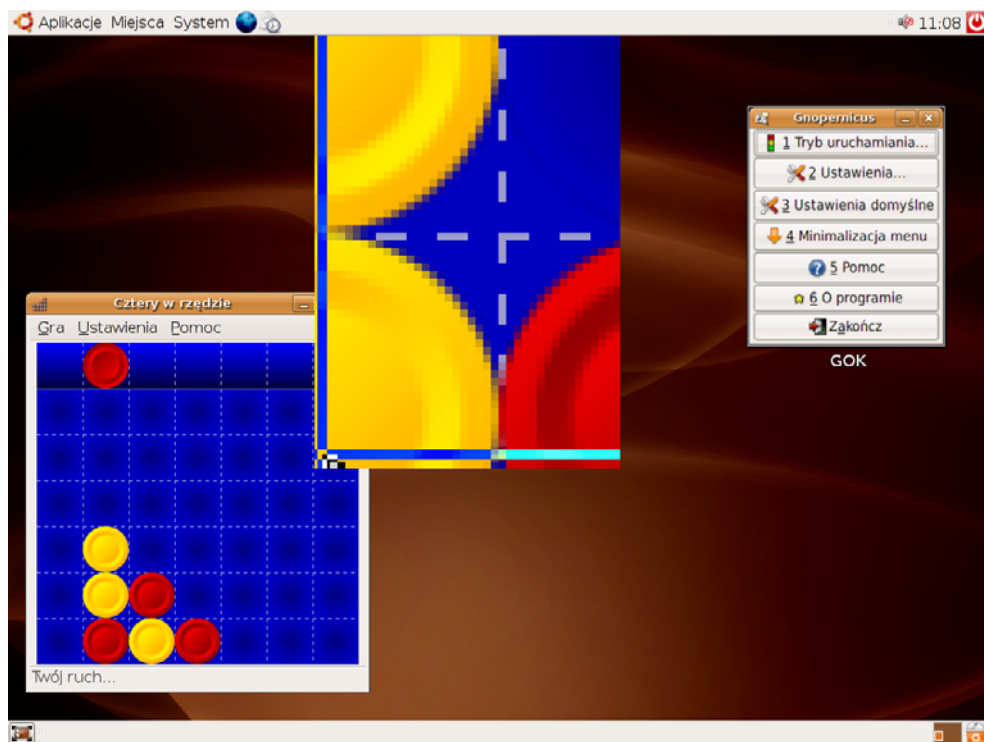
Można je uruchomić, wybierając z menu kolejno *System/Preferencje/Technologie wspierające*, zostanie wówczas wyświetlone okno dialogowe (patrz rysunek 5.28), umożliwiające aktywowanie tych technologii; wystarczy zaznaczyć odpowiednie pole i zalogować się ponownie do systemu.

Rysunek 5.28.

Włączanie technologii wspierających w GNOME



Programy te, jeszcze niedoskonałe, są aktywnie rozwijane, aby każdemu zapewnić korzystanie z możliwości oferowanych przez Ubuntu. Na rysunku 5.29 widać przykład działania jednego z tych programów, który służy do powiększania wyświetlanej zawartości ekranu w GNOME.



Rysunek 5.29. Klawiatura ekranowa i powiększanie ekranu w GNOME



Nie wszystkie biblioteki używane przez technologie wspierające w GNOME obsługują pakiety, takie jak *gok* i *gnome-mag*; w przypadku kilku przyjęto, że już znajdują się w systemie. Jeżeli podczas próby ich uruchomienia zostanie wyświetlony komunikat następującej treści:

```
Failed to load module "atk-bridge": libatk-bridge.so \
cannot open shared object file: No such file or directory
```

wówczas należy użyć Synaptica (lub innego programu do instalowania pakietów) i upewnić się, czy pakiet *at-spi* znajduje się w systemie.

Skróty klawiaturowe w GNOME

Jeżeli użytkownik ma problemy z korzystaniem z myszy lub po prostu wolałby nie odrywać rąk od klawiatury, może zastosować dostępne w GNOME specjalne sekwencje klawiszy, dzięki którym użyje klawiatury do wykonania większości standardowych operacji, do jakich zwykle używa się myszy. Te kombinacje noszą nazwę **skrótów klawiszowych**. Aby użyć któregoś z nich, należy wcisnąć i przytrzymać pierwszy klawisz kombinacji, a następnie wcisnąć drugi. Oto kilka najpopularniejszych w GNOME skrótów.

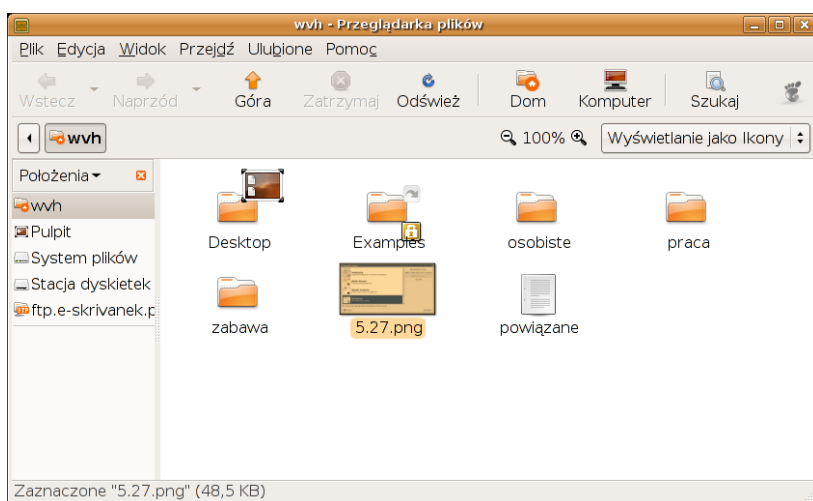
- ♦ **Alt+F1**: wyświetla menu *Aplikacje*.
- ♦ **Alt+F2**: wyświetla okno dialogowe *Uruchom program*, gdzie można wpisać nazwę programu, który ma zostać uruchomiony.
- ♦ **Alt+F4**: zamyka aktywne okno.
- ♦ **Alt+F5**: zmniejsza rozmiar aktywnego okna, jeżeli jest zmaksymalizowane.
- ♦ **Alt+F7**: umożliwia zmianę położenia aktualnie wybranego okna. Po wciśnięciu skrótu klawiaturowego można użyć klawiszy ze strzałkami do przeniesienia okna, każdy inny klawisz wyłączy tę funkcję.
- ♦ **Alt+F8**: umożliwia zmianę rozmiaru aktywnego okna, po użyciu kombinacji należy użyć klawiszy ze strzałkami do zmiany aktualnego rozmiaru, każdy inny klawisz wyłączy tę funkcję.
- ♦ **Alt+F9**: minimalizuje aktywne okno.
- ♦ **Alt+F10**: maksymalizuje aktywne okno.
- ♦ **Alt+Tab**: umożliwia przełączanie się pomiędzy wszystkimi wyświetlanymi na ekranie oknami. Podczas przełączania kolejne okna będą podświetlane, po wybraniu właściwego należy zwolnić klawisze, dane okno zostanie wówczas wyświetlone przed innymi.
- ♦ **Alt+klawisz spacji**: wyświetla menu kontekstowe przypisane do aktualnie wyświetlanego okna.
- ♦ **Control+L**: jeżeli skrót zostanie użyty podczas wskazywania pliku za pomocą polecenia menu *Plik/Otwórz*, wówczas wyświetlone zostanie okno dialogowe *Otwarcie położenia*, w którym można wpisać ścieżkę dostępu. W ten sposób łatwo zaoszczędzić sobie wielu kliknięć myszą. Podczas wpisywania program będzie automatycznie uzupełniał ścieżkę — dlatego jest to jedna z moich ulubionych funkcji GNOME.

W GNOME dostępne są jeszcze inne skróty klawiszowe, większość z nich umożliwia wykonanie standardowych czynności w odniesieniu do uruchomionych programów. Kompletna lista dostępnych skrótów dostępna jest w menu *System/Pomoc/Dokumentacja systemu*.

Menedżer plików Nautilus — wprowadzenie

Menedżer plików Nautilus, znany też po prostu jako Nautilus, to wygodne narzędzie dla użytkowników korzystających z interfejsu graficznego, które umożliwia przenoszenie plików i folderów w systemie komputerowym. Nautilus jest odpowiednikiem programu Explorer w systemach Microsoft Windows oraz Finder w Mac OS X i z podobną łatwością działa na plikach i katalogach. Na rysunku 5.30 przedstawiam przykładowe okno Nautilusa.

Rysunek 5.30.
Eksplorowanie katalogu domowego w Nautilusie



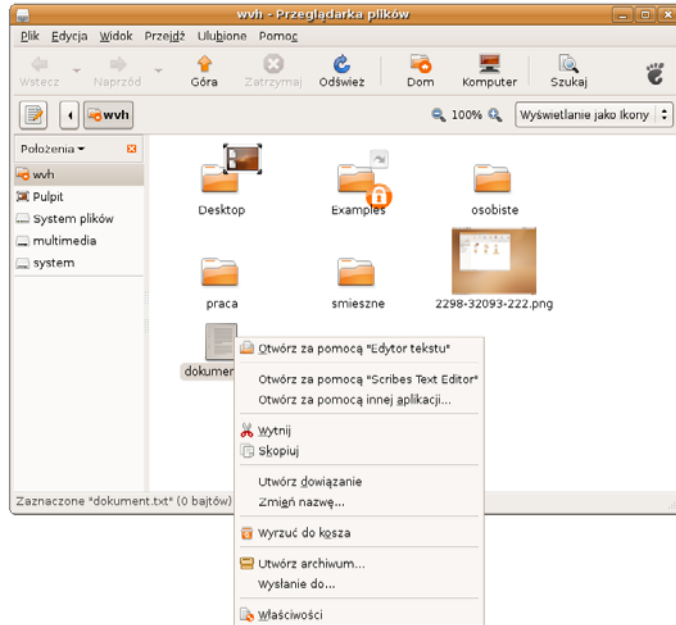
Jak wspominałem wcześniej, menu *Miejsca* znajdujące się na górnym panelu, a instalowane domyślnie w Ubuntu, jest właściwie zbiorem skrótów umożliwiających eksplorowanie za pomocą Nautilusa różnych części systemu oraz sieci. Zawartość powyższego folderu (patrz rysunek 5.30) wyświetlono, korzystając z menu *Miejsca/Folder domowy*.

Podstawowe operacje w Nautilusie

Tak jak inne menedżery okien, również Nautilus ułatwia zarządzanie plikami i katalogami. Większość poleceń służących do obsługi plików i katalogów znajduje się w menu *Edycja*, są one również dostępne z menu kontekstowego wyświetlanego po kliknięciu prawym przyciskiem myszy dowolnego obiektu wyświetlanego w Nautilusie. Na rysunku 5.31 widoczne jest menu kontekstowe wyświetlone po kliknięciu prawym przyciskiem myszy pliku o rozszerzeniu *.txt* znajdującego się w folderze domowym.

Jak widać, za pomocą menu można wykonać standardowe działania, jakie znajdują się również w innych programach tego typu. Oto one.

Rysunek 5.31.
Menu kontekstowe
w Nautilusie



- ♦ Otwieranie pliku za pomocą aplikacji skojarzonej z danym typem pliku.
- ♦ Otwieranie pliku za pomocą innej aplikacji skojarzonej z danym typem pliku.
- ♦ Wycięcie lub skopiowanie pliku, dzięki czemu można go przenieść (lub skopiować) w nowe miejsce.
- ♦ Utworzenie dowiązania do danego pliku.
- ♦ Usunięcie pliku poprzez przeniesienie do *Kosza*, który jest tymczasowym magazynem dla plików przed ich ostatecznym usunięciem z systemu.
- ♦ Wysłanie pliku do określonego urządzenia lub utworzenie archiwum poprzez skompresowanie pliku do jednego z dostępnych formatów. Więcej informacji na temat archiwizowania można znaleźć w rozdziale 22., „Wykonywanie i przywracanie kopii zapasowych”.
- ♦ Sprawdzenie lub zmodyfikowanie właściwości pliku, łącznie z jego nazwą, uprawnieniami i sposobem wyświetlania w Nautilusie.

Przykłady użycia Nautilusa

Wcześniej opisałem operacje najczęściej wykonywane za pomocą Nautilusa, w tym punkcie zamieszczam kilka przykładów, jak rozpocząć pracę z graficznym menedżerem plików, są one przydatne zwłaszcza dla osób, które wcześniej z niego nie korzystały.

Eksploracja systemu plików

Menu *Miejsca* w GNOME zawiera kilka wstępnie skonfigurowanych skrótów do różnych lokalizacji w systemie lokalnym oraz sieci. Przy odrobinie szczęścia katalogi i lokalizacje

sieciowe, które użytkownik chciałby otwierać, również się tam znajdują. Jeżeli nie, można do nich przejść samodzielnie, wykonując następujące czynności.

1. Należy otworzyć Nautilusa w zdefiniowanej wstępnie lokalizacji, korzystając z menu *Miejsca*.
2. Teraz trzeba przejść do wybranego podkatalogu w danej lokalizacji — wystarczy dwukrotne kliknięcie odpowiedniego folderu.
3. Następnie przejść wyżej (czyli do katalogu zawierającego przeglądany aktualnie katalog) — wystarczy użyć przycisku *Góra* znajdującego się na pasku narzędzi Nautilusa.



Można uruchomić Nautilusa tak, aby od razu wyświetlał wybrany przez użytkownika katalog, w tym celu należy otworzyć go z linii poleceń, podając odpowiedni argument. Więcej informacji na ten temat zamieszczono w rozdziale 6., „Korzystanie z linii poleceń”.

Kopiowanie plików i katalogów

Aby skopiować plik lub katalog za pomocą Nautilusa, należy:

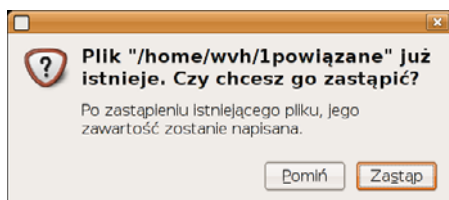
1. Przejść do katalogu zawierającego plik, który ma być kopiowany.
2. Zaznaczyć go, klikając lewy przycisk myszy.
3. Wywołać menu kontekstowe, wciskając prawy przycisk myszy, i wybrać z niego polecenie *Skopiuj*, można też po zaznaczeniu pliku wybrać to samo polecenie z menu *Edycja*.
4. Przejść do katalogu, w którym plik lub katalog ma zostać umieszczony.
5. Wybrać z menu kontekstowego lub menu *Edycja* polecenie *Wklej* (lub *Wklej do folderu*).

Po chwili w katalogu, do którego był kopiowany dany plik, pojawi się jego ikona, co będzie oznaczać, że operacja została zakończona. Kopiowanie katalogów zawierających wiele plików może zająć więcej czasu.

Nie należy martwić się, jeżeli zdarzy się kopiowanie do folderów plików o tych samych nazwach; jeżeli w katalogu docelowym znajdują się pliki mające takie same nazwy jak te, znajdujące się w katalogu źródłowym, wówczas zostanie wyświetlony odpowiedni komunikat informujący o tym użytkownika (patrz rysunek 5.32). Będzie on mógł zdecydować, czy istniejące pliki mają zostać zastąpione.

Rysunek 5.32.

Komunikat informujący o powtarzającej się nazwie pliku



Zaznaczanie wielu plików i katalogów

Podczas pracy z Nautilusem może się okazać, że użytkownik chce jednocześnie skopiować wiele plików i katalogów. Jest to całkiem prosta operacja, wystarczy tylko wcześniej zaznaczyć wszystkie pliki i katalogi, które mają zostać skopiowane. Zaznaczenie wielu elementów można w Nautilusie wykonać na kilka sposobów.

- ♦ Aby zaznaczyć wiele plików lub katalogów, należy podczas wybierania kolejnych trzymać wciśnięty klawisz *Ctrl*.
- ♦ Aby zaznaczyć grupę plików lub katalogów (według kryterium alfabetycznego), należy zaznaczyć pierwszy plik lub katalog, następnie wcisnąć przycisk *Shift*, po czym zaznaczyć ostatni plik lub katalog, który ma zostać zaznaczony.
- ♦ Aby graficznie zaznaczyć pliki znajdujące się obok siebie w oknie Nautilusa, należy wcisnąć lewy przycisk myszy (z kursorem umieszczonym na wolnej przestrzeni w oknie programu), a następnie z wciśniętym przyciskiem zaznaczyć odpowiedni zakres plików, przesuwając mysz.

Te sposoby można również łączyć ze sobą; i tak np. zaznaczyć grupę plików, trzymając wciśnięty klawisz *Shift*, a następnie wcisnąć klawisz *Ctrl* i dodać (lub odznaczyć) kolejne pliki do już wybranych.

Przenoszenie plików i katalogów

Aby przesunąć plik lub katalog, należy:

1. Przejść do katalogu zawierającego plik lub katalog, który ma zostać przeniesiony.
2. Zaznaczyć ikonę pliku lub katalogu, używając lewego przycisku myszy.
3. Z menu *Edycja* lub menu kontekstowego wybrać polecenie *Wytnij*, zaznaczona ikona pliku lub katalogu zostanie usunięta z okna Nautilusa.
4. Przejść do katalogu, w którym ma znaleźć się wycięty uprzednio element.
5. Wybrać z menu *Edycja* lub menu kontekstowego polecenie *Wklej*.

W tym momencie ikona symbolizująca plik lub folder powinna pojawić się w katalogu docelowym, będzie to znak, że operacja przenoszenia zakończyła się poprawnie. Przenoszenie katalogów zawierających dużo plików może zająć dłuższą chwilę.

Zmiana nazwy pliku lub katalogu

Aby zmienić nazwę pliku lub katalogu, należy:

1. Przejść do katalogu zawierającego element, którego nazwa ma zostać zmieniona.
2. Za pomocą prawego przycisku myszy zaznaczyć plik lub katalog, którego nazwa ma zostać zmieniona, i z menu kontekstowego wybrać pozycję *Zmień nazwę*. Nazwa pliku zostanie podświetlona i udostępniona do edycji.
3. Należy wprowadzić nową nazwę i wcisnąć klawisz *Enter*. Zostanie wyświetlona nowa nazwa.

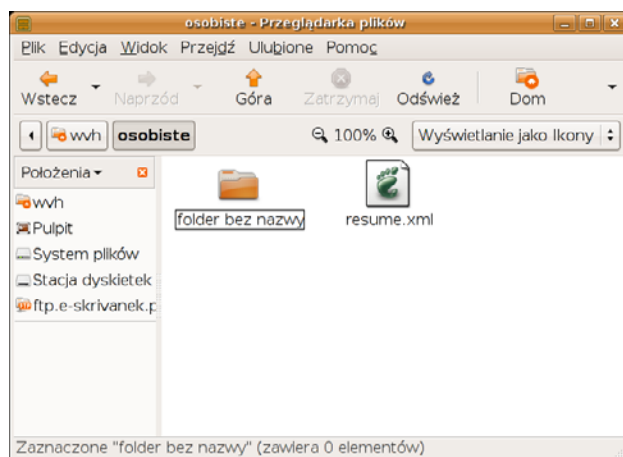
Tworzenie katalogów

Pliki tworzy się zwykle z poziomu danej aplikacji, zapisując efekt pracy pod określoną nazwą i w konkretnym miejscu. Katalogi można też tworzyć z linii poleceń (więcej informacji na ten temat podaję w rozdziale 6.). Oczywiście, katalogi można też tworzyć — szybko i łatwo — korzystając z Nautilusa.

Aby utworzyć katalog w Nautilusie, należy:

1. Przejść do katalogu, w którym ma zostać utworzony podkatalog.
2. Wcisnąć prawy przycisk myszy na pustym polu w oknie Nautilusa i z menu kontekstowego wybrać pozycję *Utwórz folder*. Można też skorzystać z menu *Plik/Utwórz folder*. Zostanie utworzony folder o nazwie *folder bez nazwy* (patrz rysunek 5.33).

Rysunek 5.33.
Tworzenie nowego
folderu w Nautilusie



3. Warto zwrócić uwagę, że podczas tworzenia nowego folderu jego nazwa jest podświetlona. Dzięki temu można szybko i łatwo zmienić nazwę zgodnie ze wskazówkami zamieszczonymi powyżej.

Dodatkowe informacje o Nautilusie

Nautilus posiada obszerną i doskonałą dokumentację, do której dostęp można uzyskać, wybierając z menu programu kolejno *Pomoc/Zawartość*. Ponieważ program działa, podobnie jak menedżery plików w innych systemach, korzystanie z niego nie powinno przysporzyć użytkownikowi żadnych trudności.

Korzystanie z menedżera okien

W podrozdziale „Co to jest pulpit? Środowiska graficzne dla Linuksa” omówiłem różnicę pomiędzy menedżerem okien a pulpitem, a także zwróciłem uwagę na fakt, że istnieje wiele różnorodnych menedżerów. Te najpopularniejsze to: After Step, Black Box,

enlightenment, Fluxbox, Fvwm, ICEwm, kwm, Metacity, twm, vtwm i Window Maker, ale jest ich dużo więcej.

Kiedy już użytkownik zapozna się z GNOME, może być ciekaw, jak wyglądają i działają inne menedżery okien. Połowa przyjemności korzystania z tak elastycznych środowisk jak Linux czy X11 bierze się z eksperymentowania z nowym oprogramowaniem tylko dlatego, „że jest”. Wypróbowaniem alternatywnych menedżerów okien mogą być zwłaszcza zainteresowani użytkownicy, którzy używają wolniejszych komputerów, z ograniczoną ilością pamięci. Dotyczy to starszych komputerów przenośnych, ponieważ takie menedżery potrzebują zwykle znacznie mniej pamięci i innych zasobów komputera aniżeli GNOME czy KDE.

Zgodnie z potencjalnymi oczekiwaniami użytkowników, przełączanie się pomiędzy GNOME i innymi środowiskami jest całkiem proste. Najpierw należy uruchomić repozytorium *Universe* (więcej informacji na ten temat zamieszczono w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”), ponieważ tam właśnie zlokalizowane są pakiety z dodatkowymi menedżerami okien. Następnie należy zainstalować jeden lub więcej menedżerów, do tego celu należy użyć jednego z narzędzi opisanych w rozdziale 20. Po zainstalowaniu pakietów trzeba wylogować się z systemu, teraz można wybrać już jeden z dodatkowych menedżerów, wystarczy wybrać przycisk *Opcje* znajdujący się po lewej stronie ekranie logowania i wskazać go w menu *Wybór sesji*. Na rysunku 5.34 widoczne jest przykładowe menu z kilkoma dodatkowymi (Fvwm, Iwcem i Fluxbox) zainstalowanymi menedżerami okien.



Rysunek 5.34. Ekran logowania GNOME i menu *Sesje*

Wystarczy wybrać odpowiedni menedżer, który użytkownik chciałby sprawdzić i użyć przycisku *Zmiana sesji*, aby zamknąć menu. Podczas logowania do systemu zostanie wyświetlone okno dialogowe (patrz rysunek 5.35), w którym użytkownik będzie mógł zdecydować, czy zmiana ma być trwała.



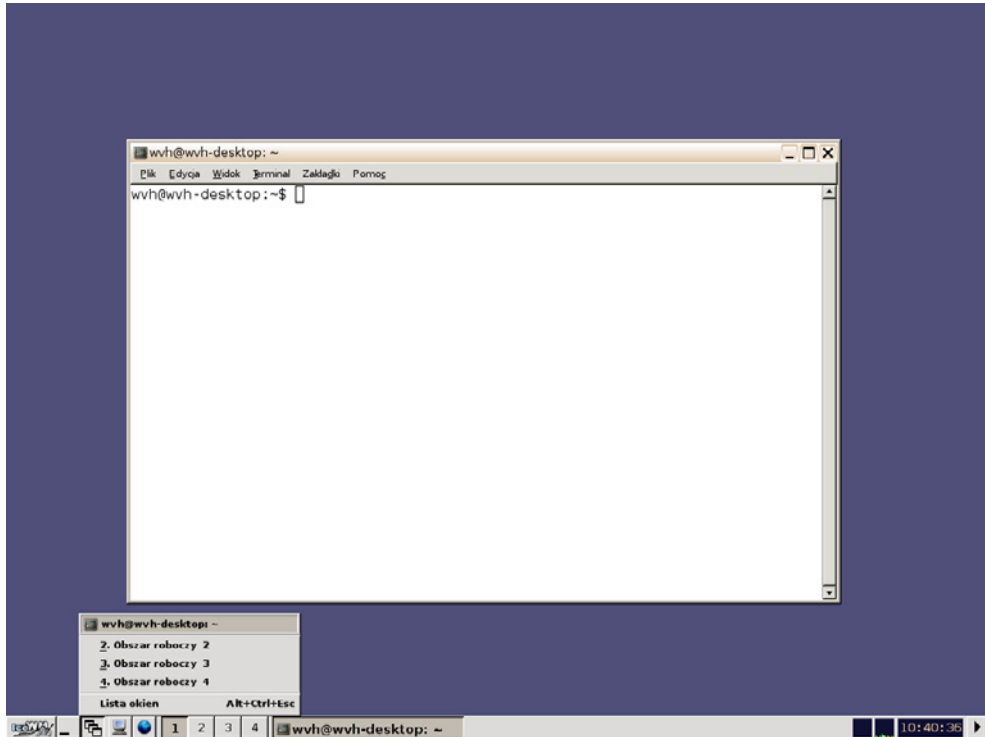
Rysunek 5.35. *Zmiana menedżera okien*

W tej chwili należy wcisnąć przycisk *Tylko dla tej sesji*. Jeżeli użytkownik zdecyduje się zmienić menedżer na stałe, wówczas podczas następnego logowania do systemu powinien wcisnąć przycisk *Uczyń domyślnym*, dzięki czemu zmiany zostaną zapisane w plikach konfiguracyjnych w katalogu domowym (to zagadnienie zostało omówione szerzej w dalszej części książki). Po kliknięciu przycisku *Tylko dla tej sesji* zostanie uruchomiony wybrany menedżer okien. Na rysunku 5.36 przedstawiam działający pod kontrolą Ubuntu menedżer IceWm.

Aby na stałe zmienić używany menedżer okien, można użyć menu *Sesje* widocznego na rysunku 5.34 oraz okna dialogowego z rysunku 5.53, ale można też utworzyć plik tekstowy o nazwie `.xsession` w katalogu domowym i umieścić w nim następującą linię:

```
exec /usr/bin/icewm
```

Plik musi stać się wykonywalny (`chmod 755 ~/.xsession`), następnym razem po zalogowaniu się do systemu jako domyślny menedżer okien zostanie uruchomiony IceWM zamiast GNOME. Aby zachować zgodność z innymi metodami uruchamiania środowiska X Window, należy też utworzyć dowiązanie symboliczne do pliku o nazwie `.xinitrc`.



Rysunek 5.36. Menedżer IceWM używany w Ubuntu

znajdującego się w katalogu domowym użytkownika. Można to zrobić za pomocą następującego polecenia:

```
ln -s ~/.xsession ~/.xinitrc
```

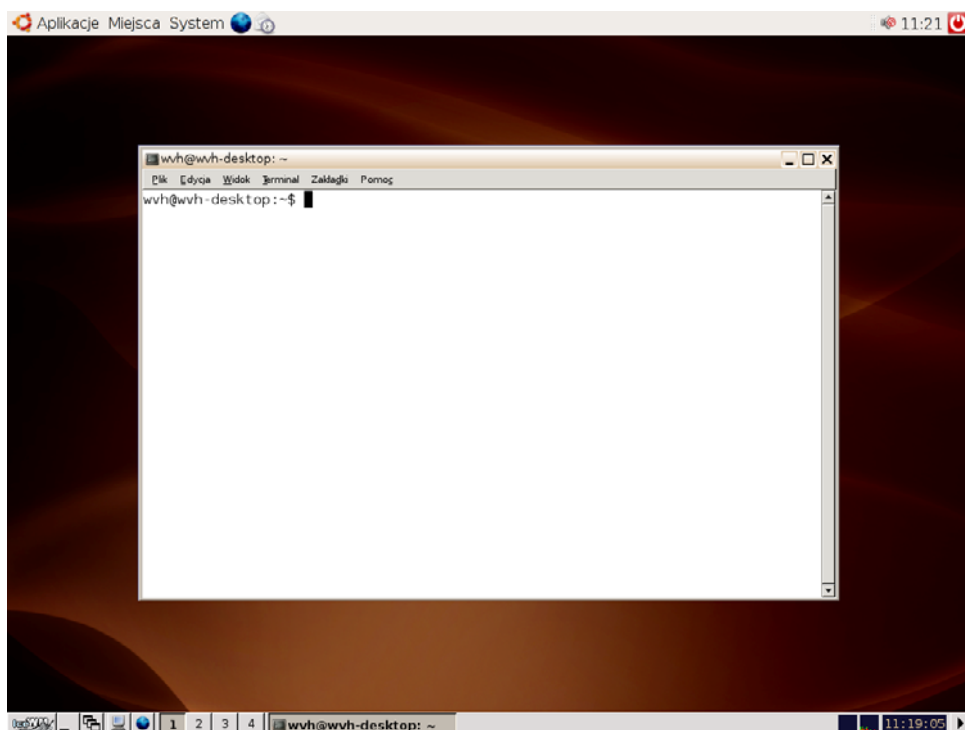
Eksperymentowanie z innymi menedżerami okien jest ciekawym doświadczeniem, ponieważ każdy z nich prezentuje inne podejście do kwestii, w jaki sposób użytkownik „powinien” korzystać z systemu graficznego, uruchamiać programy itd. Sam używam systemu X Window, począwszy od wersji 1.0; na poznawaniu, konfigurowaniu i dostosowywaniu menedżerów okien napisanych w ciągu ostatnich 20 lat spędziłem tyle czasu, że trudno byłoby go zliczyć. Ale to dobra zabawa.



Użytkownik, mając niewielką ilość pamięci czy innych zasobów systemowych (jak ma to często miejsce w starszych komputerach przenośnych), może korzystać z Linuksa bez graficznego interfejsu. System X Window można ponownie uruchomić, używając polecenia `startx` lub `xninit`, które również spowodują wykonanie pliku i dowiązania symbolicznego utworzonych wcześniej, dzięki temu zostanie też uruchomiony menedżer okien. Więcej informacji na temat dokonywania zmian podczas uruchamiania systemu z interfejsem graficznym lub w trybie tekstowym zamieszczono w rozdziale 19., „Proces rozruchu i zamykania systemu Ubuntu”.

Dzięki poprzednim instrukcjom użytkownik mógł zdecydować, czy ma zostać uruchomiony GNOME czy inny menedżer okien. Jednak zgodnie z zamieszczonymi wcześniej informacjami, wszystkie systemy uruchamiają menedżery okien w ukryciu. Domyślnie

GNOME uruchamia Metacity, KDE uruchamia kwm itd. Jeżeli użytkownik chciałby korzystać z GNOME, ale zmienić domyślny menedżer okien, musi użyć takiego, który jest zgodny ze standardem Inter-Client Communication Conventions Manual (ICCCM) 2.0 (<http://en.wikipedia.org/wiki/ICCCM>) opublikowanym przez X Consortium, ale też ze wskazaniem EWMH (ang. *Extended Window Manager Hints*). Pierwsza pozycja to oficjalna specyfikacja, określająca sposób, w jaki powinny działać klienty systemu X Window korzystające z tego samego serwera, drugie to nieoficjalny zestaw rozszerzeń do ICCCM 2.0 zdefiniowanych przez FreeDesktop.org definiujących dodatkowe menedżery okien, okna i narzędzia wspierające mechanizmy interakcji. Menedżery okien spełniające te kryteria są często określane mianem „zgodnych z GNOME”. Niektóre spośród menedżerów „zgodnych z GNOME” to: After Step, Enlightenment, Fvwm2, IceWM, Metacity i WindowMaker. Na rysunku 5.37 przedstawiam menedżer IceWM działający w GNOME



Rysunek 5.37. Menedżer okien IceWM działający w GNOME

Istnieje wiele sposobów na zmianę używanego w GNOME menedżera okien, niektóre są bardziej skomplikowane, inne mniej. Najprostszym jest zmiana zawartości pliku `~/.xsession` tak, aby najpierw był uruchamiany wybrany menedżer okien, dopiero później środowisko GNOME. Przykładowy plik `~/.xsession` uruchamiający najpierw menedżera okien IceWM, a dopiero później GNOME wygląda następująco:

```
/usr/bin/icewm&
exec gnome-session
```


Jest to sztuczka wykorzystująca fakt, że jednorazowo może być uruchomiony tylko jeden menedżer okien. Dlatego uruchomienie IceWM przed GNOME uniemożliwia włączenie Metacity, dzięki czemu GNOME zmuszone jest do korzystania z IceWM.

Uruchamiany menedżer okien powinno udać się zmienić, umieszczając poniższą linię w pliku ~/.gnomerc:

```
export WINDOW_MANAGER=/usr/bin/icewm
```

Niestety, ten sposób u mnie nie zadziałał. Przeszukując internet w poszukiwaniu rozwiązania można znaleźć wiele wskazówek, np. aby zmodyfikować gconf, skrypty startowe menedżera okien GNOME (np. gnome-wm), pliki kontrolne sesji korzystające ze skryptów GNOME i różne kombinacje powyższych. Ponieważ jednak jestem zwolennikiem rozwiązań, które „po prostu działają”, dlatego też polecam własne rozwiązanie.

Podsumowanie

W niniejszym rozdziale omówiłem graficzne interfejsy użytkownika ze szczególnym uwzględnieniem GNOME, który jest standardowo instalowany w Ubuntu. Po opisanu sposobu, w jaki zorganizowany jest pulpit Ubuntu, główną część rozdziału poświęciłem metodom dostosowywania oraz konfigurowania domyślnych ustawień GNOME w systemie Ubuntu. Spersonalizowanie GNOME to pierwszy krok na drodze do tego, by sprawić, że komputer będzie w pełni należał do użytkownika. GNOME oferuje takie same możliwości dostosowywania oraz obsługuje taki sam poziom możliwości personalizowania jak inne systemy operacyjne, takie jak Microsoft Windows czy Mac OS X — tyle tylko, że nie kosztuje tyle, ile one.

W rozdziale 6. uzupełnię informacje na temat możliwości graficznych GNOME i systemu X Window o wiedzę na temat linii poleceń. Fakt, że tak wiele programów Linuksa ukierunkowanych jest na linię poleceń, jest jedną z przyczyn, dla których zaawansowani użytkownicy Windows odnoszą się z lekceważeniem do tego systemu, co jest po prostu niemądre, ponieważ Linux oferuje zarówno interfejs graficzny, jak i potężną linię poleceń. Wystarczy zerknąć na system Mac OS X (który skrywa w sobie Uniksa i okno Terminala), aby się przekonać, jak można połączyć wspaniały interfejs graficzny użytkownika oraz potężną linię poleceń w komercyjnym produkcie. Ubuntu oferuje tę samą moc (jeżeli nie większą), tyle że za darmo.

Rozdział 6.

Korzystanie z linii poleceń

W tym rozdziale:

- ♦ Wykorzystanie linii poleceń
- ♦ Podstawowe operacje wykonywane w linii poleceń
- ♦ Konfiguracja powłoki bash

Ponieważ korzenie Linuksa wywodzą się ze starszego systemu operacyjnego Unix, dlatego też wiele programów dla niego przeznaczonych zostało zaprojektowanych do wykonywania w linii poleceń. Linia poleceń to tradycyjny interfejs spotykany w starszych komputerach, zwykle niekorzystający z wysokiej rozdzielczości czy zalet zaawansowanych graficznie monitorów, których oczekują dzisiejsi użytkownicy. W tym modelu działania system uruchamia program, znany jako **interpreter linii poleceń**, który robi dokładnie to, co sugeruje jego nazwa. Interpreter linii poleceń odczytuje wpisywane przez użytkownika komendy, odnajduje właściwe programy i uruchamia je zgodnie z wpisanymi instrukcjami. Po zakończeniu wykonywania polecenia interpreter wyświetla sekwencję znaków określanych nazwą **znaków zachęty**, które oznaczają, że program jest gotowy do przyjęcia kolejnego polecenia.

Linux nie jest zależny od linii poleceń tylko z powodu swojego pochodzenia; podczas uruchamiania systemu konieczne jest wykonanie kilku poleceń na długo przed tym, nim dostępny stanie się interfejs graficzny. Sam interfejs jest uruchamiany z linii poleceń, dopiero po tym dostępny jest przyjazny tryb graficzny.

Dla wielu osób wyświetlanie prostej linii poleceń na zaawansowanych monitorach z mnóstwem megabajtów pamięci na kartach graficznych jest marnotrawieniem możliwości sprzętu. Jednak Linux oferuje tysiące potężnych i wydajnych aplikacji linii poleceń, dlatego nawet w interfejsach graficznych (takich jak GNOME czy system X Window) dostępna jest ona w osobnym oknie, dzięki czemu użytkownik może czerpać korzyści z obu środowisk. W jednym oknie można używać Writera z pakietu Open Office czy GIMP, korzystając z ich graficznych możliwości, a w drugim równocześnie pracować z narzędziami linii poleceń.

W niniejszym rozdziale wyjaśnię podstawowe idee stojące za narzędziami linii poleceń i wykonywanymi w niej komendami, omówię główne interpretery linii poleceń w Linuksie, powiem też, jak uzyskać do nich dostęp, a także zamieszczę przegląd najpopularniejszych narzędzi działających w środowisku tekstowym w Ubuntu.



Większość przykładowych poleceń używanych w tym rozdziale jako przykłady rozpoczyna się od znaku \$, który jest domyślnym znakiem zachęty używanym w standardowo instalowanym w Ubuntu interpretatorze linii poleceń (znak zachęty to znak (lub znaki) wyświetlany przez program w celu zasygnalizowania użytkownikowi, że czeka na jego działania). Dlatego podczas pracy w środowisku tekstowym nie należy wpisywać tego znaku samodzielnie, tylko poczekać, aż zostanie wyświetlony przez program.

Po co używać linii poleceń?

W Linuksie istnieją tysiące narzędzi linii poleceń, począwszy od prostych programów do tworzenia, sprawdzania i modyfikowania plików czy uprawnień do plików, aż do skomplikowanych programów dostosowywania wydajności podzespołów komputera czy parametrów podstawowych części składowych systemu, takich jak systemy plików albo połączenia sieciowe. Wiele tych narzędzi posiada graficzne odpowiedniki, dotyczy to zwłaszcza programów poziomu użytkownika czy służących do konfigurowania systemu. Wybierając z menu *System/Administracja/Sieć*, użytkownik uruchamia program *network-admin*, za którego pomocą można wykonać te same czynności udostępniane przez narzędzie *ifconfig*. Podobnie rzecz ma się w przypadku menedżera plików *Nautilus*, który ogólnie jest odpowiednikiem interpretera linii poleceń, ponieważ umożliwia analizowanie plików oraz katalogów i działania na nich, wykonywanie programów itd.

Korzystanie z linii poleceń nie jest czymś zupełnie obcym nawet dla użytkowników takich systemów jak Microsoft Windows czy Mac OS X (choć łatwiej o to tym drugim). Wielu użytkowników Windows, szczególnie dotyczy to administratorów systemu, często dochodzi do wniosku, że używanie narzędzi linii poleceń jest po prostu wygodne, tak jest np. w programie do konfigurowania sieci i monitorowania statusu o nazwie *ifconfig*. Koncepcja linii poleceń może w pewnym stopniu wywoływać zdziwienie u doświadczonych użytkowników systemów Mac OS X, chociaż była ona prawdziwą alternatywą dla interfejsu graficznego, kiedy wprowadzono system operacyjny oparty na Uniksie.

Jeżeli nawet użytkownik nie chce korzystać z linii poleceń, jest jeden przypadek, kiedy może być do tego zmuszony, dzieje się tak wtedy, kiedy podczas uruchamiania, system Ubuntu wykryje jakieś problemy związane ze sprzętem czy ogólną konfiguracją. Jeśli podczas uruchamiania systemu okaże się, że główny system plików jest uszkodzony lub niedostępny, użytkownik zobaczy linię poleceń, zanim zdąży pomyśleć: „Hmm, ciekawe czy to poważny problem”. Wszystkie systemy linuksowe odsyłają do linii poleceń, jeżeli podczas rozruchu zostanie wykryty błąd, dzięki temu użytkownik ma dostęp do narzędzi, które można wykorzystać do usunięcia usterki i poprawnego uruchomienia systemu. Proces uruchamiania systemu przebiega w linii poleceń, wykonywane są w odpowiednim porządku pliki konfiguracyjne, z których ostatnie włączają system X Window oraz GNOME (czy inny menedżer okien, jeżeli użytkownik zmienił konfigurację). Jeśli nawet system plików jest w dobrej kondycji, a pliki konfiguracyjne urządzeń nie zawierają żadnych błędów, to zawsze może znaleźć się jakiś problem w konfiguracji systemu X Window; podczas roz-

ruchu systemu jest to jedna z najczęstszych przyczyn uruchamiania linii poleceń zamiast ekranu logowania. Więcej informacji na temat procesu uruchamiania Ubuntu zamieszczono w rozdziale 19., „Proces rozruchu i zamykania systemu Ubuntu”.

Wykonywanie komend w linii poleceń

Aplikacje zaprojektowane do wykonywania w linii poleceń systemu Linux często są określane nazwą „narzędzia linii poleceń”. Wszystkie takie narzędzia mają taką samą strukturę: zaczynają się od nazwy polecenia, które ma być wykonane, po czym następuje ewentualne określenie sposobu, w jaki ma to zostać osiągnięte.

Wszystko, co następuje po nazwie polecenia, które ma zostać wykonane, nosi nazwę **argumentu** polecenia. Argumenty dla każdego polecenia zależą właśnie od samego polecenia. Oto interesujący przykład rekurencji: argumenty narzędzi linii poleceń znane są też pod nazwą **opcji**, z których każda może być argumentem.

Czytelnicy, którzy poczuli się zakłopotani i dla których to wszystko jest nowością, nie powinni się przejmować. Kilka przykładów pomoże szybko zrozumieć temat i pozwoli na swobodne korzystanie z jej dobrodziejstw. Dobrym przykładem jest polecenie `ls`, ponieważ posiada chyba najwięcej opcji spośród wszystkich komend; zostało wprowadzone w rozdziale 4., w części poświęconej uprawnieniom użytkowników.

Polecenie `ls` wyświetla informacje o plikach i katalogach w systemie Linux. Jest to uaktualniona wersja klasycznego polecenia o tej samej nazwie, znanego z Uniksa. W prawdziwym uniksowym stylu nikt sobie nie zwraca głowy wpisywaniem dodatkowych liter „i” i „t”¹, dlatego też to polecenie zostało skrócone do łatwiejszej do wpisywania postaci `ls`.

Efektem użycia polecenia `ls` jest wyświetlenie zawartości aktualnego katalogu, oto przykład:

```
$ ls
boot_services.txt  hello.c  hello.o  include_example.c  include_test
hello              hello.foo  hello.s  include_example.out
```

Można też użyć nazwy określonego pliku czy folderu jako argument dla polecenia `ls`, tak jak w poniższym przykładzie:

```
$ ls hello
hello
```

Używanie polecenia `ls` do wyświetlania listy plików, która jest właściwie znana, nie jest specjalnie interesujące (choć może być bardzo przydatne w połączeniu ze znakami wieloznacznymi, co omówiłem w dalszej części rozdziału). Można jednak wyświetlić zawartość danego katalogu, tak jak poniżej:

```
$ ls include_test
libxml2  netdev  system
```

¹ Pełna nazwa polecenia w języku angielskim brzmi *list* — *przyp. tłum.*

Dane wyjściowe z tego polecenia informują, że katalog `include_test` zawiera trzy inne pliki lub katalogi. Jeżeli użytkownik nie jest pewien, jakiego rodzaju zawartość znajduje się w danym katalogu, może użyć polecenia `ls` z opcją `-F`, dzięki czemu uzyska potrzebne informacje. Oto przykład:

```
$ ls -F
boot_services.txt  hello.c    hello.o    include_example.c  include_test/
hello*            hello.foo  hello.s    include_example.out
```

Polecenie `ls` użyte z opcją `-F` spowoduje umieszczenie dodatkowych znaków, za których pomocą będzie można zidentyfikować obiekty niebędące plikami tekstowymi. Gwiazdka obok nazwy informuje, że jest to plik wykonywalny; ukośnik (/) z kolei umieszczany jest przy nazwach katalogów.

Polecenie `ls` użyte z opcją `-F` jest bardzo przydatne, ale — zgodnie z duchem Linuksa — nie jest jedynym sposobem uzyskania szczegółowych informacji na temat obiektów w danym katalogu. Podobne informacje można też zdobyć, używając innych opcji polecenia `ls`. Inną powszechnie używaną opcją polecenia `ls` jest `-l`, która oznacza mniej więcej „wyświetl dane w długim formacie”. Użycie jej spowoduje wyświetlenie wielu różnorodnych informacji o obiektach znajdujących się w katalogu bieżącym, oto przykład:

```
$ ls -l
razem 44
-rw-r--r-- 1 vvh users 783 2006.03-15 06:36 boot_services.txt
-rwxr-xr-x 1 vvh users 9249 2006.03-15 06:37 hello
-rw-r--r-- 1 vvh users 60 2006.03-15 06:37 hello.c
-rw-r--r-- 1 vvh users 60 2006.03-15 06:37 hello.foo
-rw-r--r-- 1 vvh users 2504 2006.03-15 06:37 hello.o
-rw-r--r-- 1 vvh users 857 2006.03-15 06:37 hello.s
-rw-r--r-- 1 vvh users 202 2006.03-15 06:37 include_example.c
-rw-r--r-- 1 vvh users 736 2006.03-15 06:37 include_example.out
drwxr-xr-x 5 vvh users 4096 2006.03-15 06:37 include_test
```

Jak widać, ta opcja umożliwia uzyskanie bardziej szczegółowych informacji o plikach i katalogach znajdujących się w katalogu bieżącym. Oto, co można znaleźć w tych danych (od lewej do prawej): uprawnienia, ilość dowiązań twardych do pliku w systemie plików (to zagadnienie omówię w dalszej części), właściciela i grupę, rozmiar, datę i czas ostatniej modyfikacji oraz nazwę pliku lub katalogu.

Jak wspominałem wcześniej, można łączyć opcje i argumenty dla tego samego polecenia, a dzięki temu udoskonalać działanie danej komendy. I tak np., aby uzyskać dokładne informacje na temat zawartości katalogu `include_test`, należy wykonać następujące polecenie:

```
$ ls -l include_test
razem 0
drwxr-xr-x 2 vvh users 72 2006.03-15 06:37 libxml2
drwxr-xr-x 2 vvh users 80 2006.03-15 06:37 netdev
drwxr-xr-x 4 vvh users 96 2006.03-15 06:37 system
```

Kolejną popularną opcją polecenia `ls` jest `-a`, powoduje ona wyświetlenie wszystkich obiektów w bieżącym katalogu. Domyślnie polecenie `ls` nie wyświetla obiektów ukrytych, których nazwa rozpoczyna się od kropki. Jest tak dlatego, że wszystkie katalogi Linuksa zawierają dwa specjalne wpisy, na które większość użytkowników nie zwraca uwagi, a które z kolei są bardzo przydatne podczas przemierzania struktury plików

w systemie. Są to `.` (kropka), która zawsze oznacza bieżący katalog oraz `..` (dwie kropki), które z kolei oznaczają katalog nadrzędny w stosunku do katalogu bieżącego. Oto efekt użycia polecenia `ls -a` do wyświetlenia zawartości bieżącego katalogu:

```
$ ls -a
.          hello.c    include_example.c  .uruchom_mnie_teraz
..         hello.foo  include_example.out
boot_services.txt hello.o    include_test
hello      hello.s     .moja_muzyka_katalog
```

Warto zwrócić uwagę, że na początku kolumny znajdują się wpisy `(.)` i `(..)`. Pojawiły się też w katalogu dwa nowe pliki, są to `.moja_muzyka_katalog` i `.uruchom_mnie_teraz`, wyświetlone na liście uporządkowanej alfabetycznie, ponieważ polecenie `ls` podczas sortowania nazw plików ignoruje kropki znajdujące się na początku (chyba że plik nie ma innych znaków w nazwie prócz kropek, wówczas — oczywiście — zostanie ujęty na samym początku listy).



Ponieważ pliki i foldery, które na początku nazwy mają kropkę, są wyświetlane tylko po użyciu opcji `-a`, taka metoda zapisu jest stosowana w przypadku plików, które należy „ukryć”. Takie pliki i foldery są zwykle wykorzystywane do przechowywania informacji konfiguracyjnych używanych przez różne programy działające w Linuksie. Najlepszym przykładem tego typu pliku może być `.bashrc`, plik konfiguracyjny powłoki shell, którą omówię w dalszej części rozdziału.

Można łączyć razem różne litery oznaczające opcje danego polecenia, wystarczy poprzedzić je pojedynczą kreską, a dane polecenie wykona wszystkie wskazane operacje. Oto przykładowy efekt użycia polecenia `ls` z opcjami `-a` i `-F` do wyświetlenia zawartości bieżącego katalogu:

```
$ ls -aF
./          hello.c    include_example.c  .uruchom_mnie_teraz*
../         hello.foo  include_example.out
boot_services.txt hello.o    include_test/
hello*      hello.s     .moja_muzyka_katalog@
```

W danych wyjściowych tego polecenia można zauważyć dwa katalogi `(.)` i `(..)` oraz tajemniczy plik wykonywalny `.uruchom_mnie_teraz*`, którego nazwa kończy się znakiem `@`. Co to oznacza?

Użyłem poprzednio terminu „dowiązanie”, teraz jest dobry moment, aby je wyjaśnić. Dowiązania to wskaźniki do innych plików lub katalogów w systemie Linux. Istnieją dwa rodzaje dowiązań: symboliczne (zawierające nazwę innego pliku lub katalogu, do których się odwołują) i twarde (połączenia do istniejących plików i folderów). Ponieważ dowiązania twarde to połączenia do istniejącej struktury danych, obiekty, dla których tworzone są dowiązania, muszą istnieć i znajdować się na tej samej partycji dysku, co dowiązania. Jest to efekt uboczny wewnętrznej struktury danych wykorzystywanej przez system plików, ale jest to zbyt obszerne zagadnienie, aby omawiać je tutaj. Aby nie zanudzać czytelnika, nie będziemy dalej przyglądać się strukturze danych systemu plików. Odmienne niż dowiązania sztywne, dowiązania symboliczne zawierają tylko nazwę, a zatem mogą wskazywać dowolny plik na komputerze. Jak pewnie czytelnik zaczął podejrzewać, symbol `@` oznacza dowiązanie symboliczne do innego pliku lub katalogu znajdującego się gdzieś w systemie. Połączenie opcji `-a` i `-l` polecenia `ls` pozwoli uzyskać kolejne szczegółowe informacje na temat zawartości bieżącego katalogu:

```
$ ls -al
razem 45
drwxr-xr-x  3 vvh users  400 2006.03-16 06:20 .
drwxr-xr-x 13 vvh users 1376 2006.03-16 05:38 ..
-rw-r--r-  1 vvh users  783 2006.03-15 06:36 boot_services.txt
-rwxr-xr-x  1 vvh users 9249 2006.03-15 06:37 hello
-rw-r--r-  1 vvh users   60 2006.03-15 06:37 hello.c
-rw-r--r-  1 vvh users   60 2006.03-15 06:37 hello.foo
-rw-r--r-  1 vvh users 2504 2006.03-15 06:37 hello.o
-rw-r--r-  1 vvh users  857 2006.03-15 06:37 hello.s
-rw-r--r-  1 vvh users  202 2006.03-15 06:37 include_example.c
-rw-r--r-  1 vvh users  736 2006.03-15 06:37 include_example.out
drwxr-xr-x  5 vvh users  120 2006.03-15 06:37 include_test
lrwxrwxrwx  1 vvh users   11 2006.03-16 06:20 .moja_muzyka -> /opt2/muzyka
-rwxr-xr-x  1 vvh users  269 2006.03-16 05:47 .uruchom_mnie_teraz
```

Teraz wyświetliło się całkiem sporo informacji na temat wszystkich elementów znajdujących się w folderze z uwzględnieniem dowiązań. W drugiej kolumnie mamy ilość sztywnych dowiązań do każdego obiektu. I tak w drugiej kolumnie dla katalogu `.` widać, że istnieją trzy sztywne dowiązania do danego katalogu; są to: sam katalog `.`, wpis do katalogu nadrzędnego i sztywne dowiązanie do `.`, które jest widoczne w podkatalogu *include_test*, ponieważ jest to katalog nadrzędny dla *include_test*. Wpis `(.)` to sztywne łącze do katalogu nadrzędnego w stosunku do aktualnego, wydaje się popularne, ponieważ do niego istnieje aż 13 łączy. Oznacza to, że zawiera on — prawdopodobnie — kilkanaście innych katalogów. Patrząc na wpis `.moja_muzyka`, można zauważyć, że jest to faktycznie dowiązanie symboliczne, a jego nazwa wskazuje inny obiekt, który również jest dowiązaniem symbolicznym. W tym przypadku plik `.moja_muzyka` należy do katalogu, który znajduje się w innym systemie plików, zamontowanym w katalogu `/opt2/muzyka` znajdującym się w systemie.

Wszystkie omówione do tej pory opcje linii poleceń poprzedzone były pojedynczym myślnikiem. Nie jest to jednak wymóg konieczny w przypadku wszystkich poleceń i ich opcji. Istnieją komendy korzeniami sięgające pradawnych wersji Uniksa (np. polecenie `tar`), w których opcji czy grupy opcji nie trzeba poprzedzać myślnikiem. Obecnie ta antyczna konwencja nie jest już zalecana, co oznacza, że każdy, kto będzie dzisiaj próbował zaimplementować polecenie, którego opcji nie trzeba poprzedzać przynajmniej jednym myślnikiem, zostanie wykpiiony za pośrednictwem poczty elektronicznej przez członków społeczności linuksowych i otwartego oprogramowania. Wszystkie komendy uniksowe zostały zaktualizowane tak, że opcje dla nich można już poprzedzać myślnikiem. Obecnie opcje poleceń zawsze rozpoczynają się od myślnika, ale na skutek interesującego zwrotu akcji pojawiły się też takie, które mogą być poprzedzone dwoma myślnikami. Przyjęto następującą konwencję: opcje składające się z jednej litery poprzedza się pojedynczym myślnikiem, natomiast wieloliterowe (czyli „całe słowa”) poprzedza się dwoma myślnikami. Jest to konieczne z dwóch powodów.

- ♦ Większość narzędzi linii poleceń obsługuje oba rodzaje opcji: tradycyjne, jednoliterowe oraz nowsze opcje składające się z kilku znaków, który to styl wprowadzili programiści z fundacji Free Software.
- ♦ Ponieważ można łączyć jednoliterowe opcje razem w ciągu (jak czytelnicy zauważą w dalszej części rozdziału), dopóki nie zostanie użyty podwójny myślnik wskazujący, że litery tworzą jedno słowo, dopóty polecenie będzie traktować opcje jak pojedyncze znaki.

Oto przykład — próba wyświetlenia pomocy dla polecenia `ls`:

```
$ ls -help
ls: invalid option -- e
Spróbuj `ls --help' dla uzyskania informacji.
```

Ten przykład jest prosty i dobrze ilustruje omawianą zasadę. Po poprzedzeniu opcji pojedynczym myślnikiem polecenie próbowało odczytać każdą literę jako osobną opcję, a ponieważ dla komendy `ls` nie ma opcji `e`, została wyświetlona informacja na ten temat. Druga próba: uruchomienie polecenia `ls --help` spowoduje wyświetlanie następujących danych wyjściowych (skrótowych tutaj, ponieważ jest to tylko przykład):

```
$ ls --help
Składnia: ls [OPCJA]... [PLIK]...
Wypisanie informacji o PLIKACH (domyślnie w katalogu bieżącym). Sortowane
alfabetyczne, jeżeli nie jest podana żadna z opcji -cftuSUX ani --sort.

Argumenty obowiązkowe dla opcji długich obowiązują również dla krótkich.
-a, --all                bez ukrywania plików zaczynających się od .
-A, --almost-all        bez pokazania . ani ..
--author                 z -l: wypisanie autora każdego pliku
-b, --escape             wypisanie znaków niegraficznych ósemkowo
--block-size=ROZMIAR     użycie bloków o podanym w bajtach ROZMIARZE
-B, --ignore-backups     bez pokazania plików kończących się na ~
```

Co to jest powłoka?

Powłoka, czyli shell, to nazwa nadana każdemu linuksowemu interpreterowi poleceń, pochodzi z Uniksa, z którego z kolei wywodzi się Linux. Unix był pierwszym systemem operacyjnym wcielającym w życie ideę korzystania z interpretera poleceń, który nie był wbudowany w system operacyjny i nie wymagał żadnych specjalnych uprawnień do wykonywania tajemniczych zadań administracyjnych. Ta koncepcja została zachowana w każdym kolejnym systemie przyjmującym założenia Uniksa. Jednym z najbardziej interesujących aspektów powłoki jest fakt, że ponieważ jest to samodzielne narzędzie, każdy użytkownik może swobodnie wybrać powłokę, która mu odpowiada, a każda z nich może być uaktualniana, niezależnie od systemu operacyjnego.

Oryginalna powłoka dla Uniksa została napisana przez Kena Thompsona, jednego z dwóch głównych autorów Uniksa. Następnie rozbudował ją przez John Mashey, również w Bell Labs. Ta powłoka była stosunkowo prymitywna; pierwszą powłoką, która była wyposażona w funkcje znane dzisiejszym użytkownikom, była powłoka Bourne, napisana przez Stephena Bourne'a w Bell Labs w roku 1974. Powłoka, znana wśród przyjaciół jako `/bin/sh`, jest obecnie narzędziem używanym w większości systemów uniksowych, a z niej z kolei wywodzi się powłoka `/bin/bash` (znana jako *Bourne-Again Shell*) używana obecnie jako domyślne narzędzie w większości systemów linuksowych. Dla systemów linuksowych dostępnych jest wiele różnorodnych powłok (patrz niżej). Jeżeli czytelnicy korzystali wcześniej z Uniksa, wówczas dość szybko poczują się tu jak w domu (a być może nawet przeniosą pliki konfiguracyjne powłoki z Uniksa do Linuksa).

Dodatkowe informacje o poleceniach

Jak widać z poprzedniego przykładu, dla wielu poleceń dostępna jest opcja `--help`, dzięki której można wyświetlić informacje o danej komendzie. Jest to krótkie na temat używania danego polecenia, dostępnych dla niego opcji i ich znaczeń. Niestety, nie wszystkie polecenia dysponują opcją `--help`, w związku z czym nie można poznać wszystkich dostępnych dla nich opcji. Jak wspominałem w rozdziale 5., w środowisku graficznym używanym w Ubuntu dostępny jest obszerny system pomocy, w którym można znaleźć również informacje na temat poleceń używanych w trybie graficznym. Na szczęście, również w linii poleceń można znaleźć podobne systemy pomocy przy użyciu poleceń `man` i `info`.

Polecenie `man` jest najlepszym przyjacielem użytkownika podczas korzystania czy eksperymentowania z programami dostępnymi w linii poleceń. To polecenie wyświetla elektroniczne dokumenty zawierające pomoc dla programu, w dodatku sformatowane odpowiednio do monitora użytkownika. W stylu prawdziwie uniksowo-linuksowym do przeglądania kolejnych stron używa się znajomych poleceń `more` i `less`. Zawartość tej elektronicznej pomocy dostępnej za pomocą polecenia `man` jest wyświetlana w układzie „jedna strona na ekran” i odpowiada tradycyjnym, papierowym podręcznikom; elektroniczne wersje podręczników noszą nazwę **strony man**.

Oto przykład użycia polecenia `man`: aby uzyskać dodatkowe informacje na temat samego polecenia `man`, należy wpisać `man man`. Na ekranie pojawi się następująca zawartość:

```
man(1)                      Manual pager utils                      man(1)

NAME
    man - an interface to the on-line reference manuals

SYNOPSIS
    man [-c|-w|-tZHT device] [-adhu7V] [-i|-I] [-m system[...]] [-L
    locale] [-p string] [-M path] [-P pager] [-r prompt] [-S list] [-e
    extension] [[section] page ...] ...
    man -l [-7] [-tZHT device] [-p string] [-P pager] [-r prompt] file
    ...
    man -k [apropos options] regexp ...
    man -f [whatis options] page ...

DESCRIPTION
    man is the system's manual pager. Each page argument given to man is
    normally the name of a program, utility or function. The manual
    page associated with each of these arguments is then found and dis-
    played. A section, if provided, will direct man to look only in that
    section of the manual. The default action is to search in all of
    the available sections, following a predefined order and to show
    only the first page found, even if page exists in several sections.
```

[Dalsza część usunięta]

Jak można zauważyć, elektroniczne podręczniki dla Linuksa zawierają dodatkowe informacje o opcjach programu, ale dostarczają też znaczących ilości informacji na temat samego używania danego polecenia oraz prezentują wybrane przykłady.

Bardziej współczesną alternatywą dla polecenia `man` jest polecenie `info`, które ma podobne zastosowanie. Polecenie `info` używa edytora `emacs` do wyświetlania informacji oraz oferuje nieco bardziej rozbudowaną i nowszą dokumentację poleceń, niż ma to miejsce w przypadku tradycyjnego polecenia `man`. Na poniższym rysunku widoczne jest okno terminala przedstawiające informacje na temat polecenia `info`:

```
wvh@wvh-desktop: ~
Plik Edycja Widok Terminal Zakładki Pomoc
File: dir      Node: Top      This is the top of the INFO tree
This (the Directory node) gives a menu of major topics. Typing "d"
returns here, "q" exits, "?" lists all INFO commands, "h" gives a
primer for first-timers, "mTexinfo<Return>" visits Texinfo topic,
etc.
Or click mouse button 2 on a menu item or cross reference to select
it.
--- PLEASE ADD DOCUMENTATION TO THIS TREE. (See INFO topic first.) ---

In Ubuntu, Info `dir` entries are added with the command
`install-info`. Please refer to install-info(8) for usage details.

* Menu: The list of major topics begins on the next line.

Basics
* Finding files: (find).                Operating on files matching certain
                                         criteria.

Miscellaneous:
* Rluserman: (rluserman).              GNU Readline Library API

Basics
-----Info: (dir)Top, 171 linii --Top-----
Witamy w Info w wersji 4.8. ? pokazuje pomoc, m wybiera element menu.
```

Polecenie info uruchomione w oknie terminala

Polecenie `info` zostało wprowadzone przez programistów GNU po to, aby można było wyświetlać elektroniczną pomoc w lepszej postaci. Pliki tekstowe używane przez to polecenie mają bardzo bogatą składnię umożliwiającą tworzenie łączy pomiędzy różnymi częściami dokumentu; w dodatku do wyświetlania jego zawartości używany jest `emacs`, który jest bardziej elastycznym środowiskiem do przeglądania i poruszania się po dokumentach. Polecenie `info` może też wyświetlać tradycyjne strony `man`, zatem za jego pomocą można je sprawdzać i przeglądać. Polecenie `info man` jest chyba najlepszym przykładem rekursji tego rodzaju.

W trakcie korzystania z narzędzi linii poleceń komendy `man` i `info` szybko staną się najlepszymi przyjaciółmi użytkownika.



Jednym z terminów, na który można się często natknąć w trakcie prowadzenia dyskusji o Linuksie czy Uniksie, są **skrypty powłoki**. To pliki zawierające wewnętrzne polecenia powłoki (takie jak ustawienia zmiennych, wyrażenia warunkowe czy pętle) lub odnoszące się do innych poleceń Linuksa. Możliwość tworzenia bardzo skomplikowanych poleceń to jedna z funkcji Linuksa: każde narzędzie wykonuje (i to dobrze) jedno zadanie, ale wiele różnych narzędzi można połączyć razem (w potoki) i wykonywać skomplikowane działania.

Dodatkowych informacji o powłokach w systemach uniksowych można szukać na następujących stronach:

- ♦ www.softpanorama.org/People/Shell_giants/introduction.shtml,
- ♦ www.faqs.org/faqs/unix-faq/shell/shell-differences/,
- ♦ www.unix.org.ua/orelly/unix/ksh/ch01_03.htm.

Powłoki dostępne dla systemów linuksowych

W Ubuntu domyślnie instalowana jest tylko standardowa powłoka GNU bash. Ale za pośrednictwem narzędzia `apt-get` czy menedżera pakietów Synaptic dostępne są również inne. Oto te, które można znaleźć w wielu systemach linuksowych.

`/bin/ash` — powłoka Almquist, niewielkie narzędzie, którego początek to klon SYSV R4 innej wersji powłoki Bourne'a. Powłoka `ash` jest często wykorzystywana w systemach zagnieżdżonych lub podczas procesu uruchamiania albo instalowania w niektórych dystrybucjach Linuksa.

`/bin/bash` — domyślna powłoka Bourne-Again, inspirowana przez `/bin/sh` i `/bin/ksh`. Jest to domyślna powłoka instalowana i używana przez wszystkich użytkowników Ubuntu.

`/bin/csh` — po zainstalowaniu `tcsh` (patrz opis w dalszej części listy) zostanie utworzone symboliczne dowiązanie z `/bin/csh` do `/bin/tcsh`. To dowiązanie obsługuje skrypty powłoki odwołujące się do tradycyjnej lokalizacji C-Shell, standardowej powłoki używanej w systemach Berkeley Unix (BSD) i ich pochodnych. C-Shell obsługuje polecenia konfiguracyjne i korzysta ze składni języka C (stąd nazwa); została napisana przez Billy'ego Joya, późniejszego założyciela Sun Microsystems.

`/bin/dash` — powłoka Debian Almquist, wersja przeznaczona dla dystrybucji Debian Linux. Użytkownicy zainteresowani wypróbowaniem tego narzędzia mogą znaleźć je w repozytoriach Ubuntu.

`/bin/ksh` — po zainstalowaniu `pksh` (opis w dalszej części listy) zostanie utworzone symboliczne dowiązanie z `/bin/ksh` do `/bin/pksh`. Dowiązanie obsługuje skrypty powłoki odwołujące się do tradycyjnej lokalizacji powłoki Korn, standardowej powłoki używanej w systemach SYSV Unix firmy AT&T i ich pochodnych.

`/bin/nash` — kolejna niewielka powłoka wykorzystywana podczas procesu uruchamiania w wielu systemach linuksowych, szczególnie w dystrybucjach Red Hat i Fedora Core.

`/bin/pdksh` — wersja *open source* powłoki napisanej przez Davida Korna, czyli powłoki Korn. Narzędzie to zostało utworzone w firmie AT&T i było przeznaczone dla systemów uniksowych SYSV R3 (jako część „Eksperymentalnego zestawu narzędzi”) i R4 (jako w pełni obsługiwane narzędzie). Powłoka Korn jest kompletnym narzędziem kompatybilnym z oryginalną powłoką Bourne'a. Użytkownicy zainteresowani wypróbowaniem tego narzędzia mogą znaleźć je w repozytoriach Ubuntu.

`/bin/sh` — dowiązanie symboliczne do `/bin/bash`, oferuje kompatybilność z podstawowymi skryptami dla standardowych powłok dla Linuksa i Uniksa.

`/bin/tcsh` — TENEX C-Shell, zaawansowana wersja *open source* powłoki C-Shell wyposażona w rozszerzenia do edycji w linii poleceń oryginalnie wprowadzone w interpreterze używanym w systemach DEC TOPS-20 dla PDP-10, powłoka ta najpierw występowała jako BBN TENEX (dla PDP-10), następnie zmutowała do postaci TWENEX (dla TOPS-20). Użytkownicy zainteresowani wypróbowaniem tego narzędzia mogą znaleźć je w repozytoriach Ubuntu.

`/bin/zsh` — powłoka Z to potężne i posiadające ogromne możliwości rozwijania narzędzie oferujące sporo funkcji dostępnych w powłokach `bash` and `ksh`, a także wiele, wiele więcej. Użytkownicy zainteresowani wypróbowaniem tego narzędzia mogą znaleźć je w repozytoriach Ubuntu.

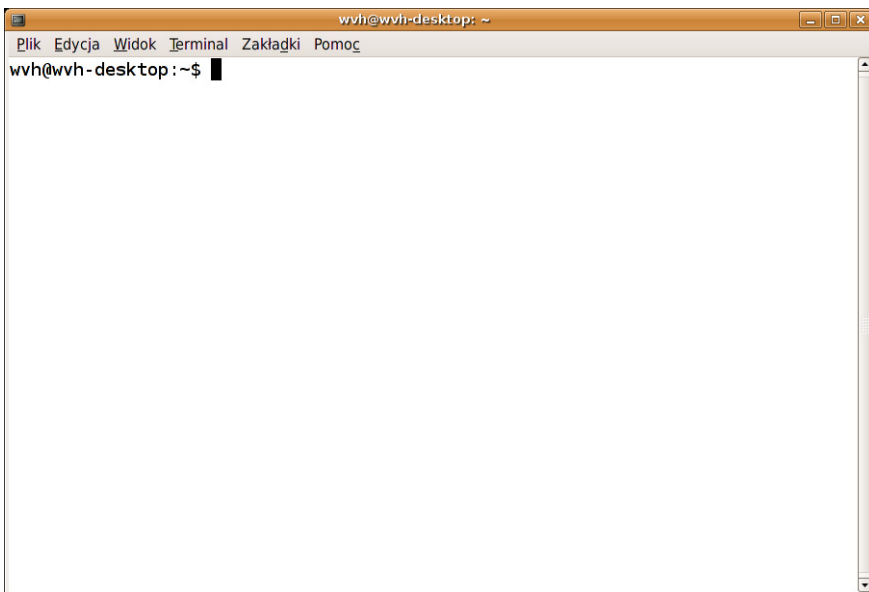
Jeżeli użytkownik zdecyduje się zmienić domyślną powłokę i używać innej niż te, które są umieszczone w pliku `/etc/passwd`, wówczas należy skorzystać z polecenia `chsh`. Nie da się zmienić logina powłoki na inny plik binarny; wszystkie programy, którymi można zastąpić powłokę, można znaleźć w pliku `etc/shells` w systemie Ubuntu. Aby ustawić inne aplikacje, należy najpierw przeprowadzić edycję pliku `/etc/shells`.

W głąb powłoki

Zakładam, że udało mi się zrobić trochę zamieszania wokół linii poleceń, zapewne teraz użytkownik, któremu system poprawnie uruchomił się w trybie graficznym, zastanawia się, w jaki sposób uruchomić powłokę, aby nieco w niej poeksperymentować. W Ubuntu dostępne są dwa podstawowe mechanizmy służące do uruchamiania powłoki; jeden z nich jest częścią systemu X Window, a drugi należy do GNOME i można go znaleźć w każdej dystrybucji korzystającej z tego środowiska. W odniesieniu do obu używa się pojęcia „aplikacja terminala”, ponieważ przypominają czasy, w których użytkownicy nie mogli korzystać z interfejsu graficznego, a dostęp do systemów komputerowych uzyskiwało się za pomocą terminali właśnie, które same w sobie nie mogły zrobić nic więcej, poza wyświetlaniem danych wejściowych i wyjściowych interpretera.

Korzystanie z terminala w GNOME

Program Terminal to najpowszechniejszy sposób na uruchomienie powłoki w graficznym systemie Ubuntu. Aby go uruchomić, należy wybrać pozycję *Terminal* z menu *Aplikacje/Akcesoria*. Na rysunku 6.1 widoczne jest okno programu Terminal.



Rysunek 6.1. Terminal w GNOME

Jak w większości aplikacji GNOME, również terminal wyposażony jest w rozbudowany system pomocy elektronicznej; aby ją uruchomić, należy z menu wybrać *Pomoc/Zawartość*.

Program Terminal ma rozbudowane menu, które daje użytkownikowi szerokie możliwości konfigurowania różnych aspektów działania programu, takie jak zmiana nazwy wyświetlanej w belce tytułowej (menu *Terminal/Zmień tytuł*), kodowanie znaków (menu

Terminal/Ustaw kodowanie znaków), a także ustawienia dotyczące rozmiaru, koloru i kroju czcionki oraz innych parametrów wyświetlania (menu *Edytuj/Bieżące profile*). Uważam, że najprzydatniejszą funkcją jest jednak możliwość uruchamiania wielu sesji terminala i zarządzania nimi, co odbywa się za pomocą zakładek. Aby otworzyć nową zakładkę, należy wybrać z menu *Plik/Otwórz zakładkę* (patrz rysunek 6.2).



Rysunek 6.2. Obsługa wielu zakładek w terminalu

Kiedy już uruchomiono wiele zakładek, można poruszać się między nimi, klikając wybraną lub wybierając odpowiednią pozycję z menu *Zakładki*. Jeżeli użytkownik chciałby, aby wybrana zakładka była wyświetlana w osobnym oknie, należy z menu wybrać *Zakładki/Odłączanie zakładki*. Aby uruchomić nowe okno terminala, można wybrać z menu już otwartego okna polecenie *Plik/Otwórz terminal*.

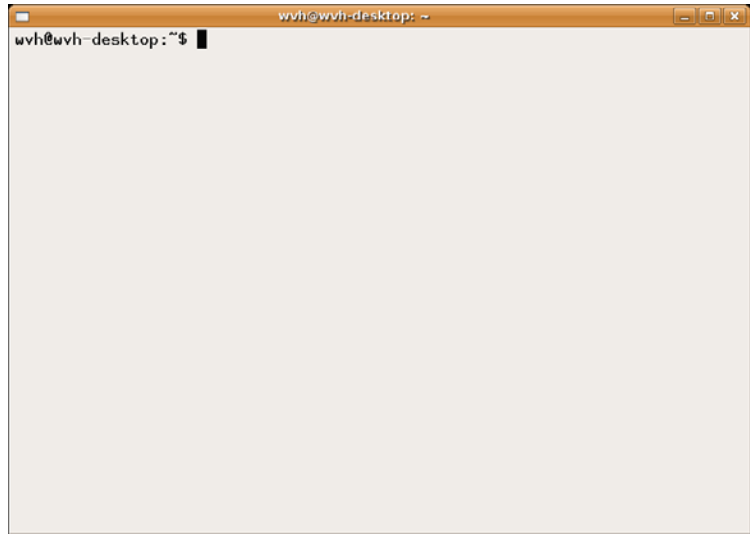
Korzystanie z terminala w systemie X Window System

Każdy system X Window jest wyposażony w niewielką aplikację terminala znaną pod nazwą *xterm*. Plik wykonywalny tego programu (widocznego na rysunku 6.3) znajduje się w */usr/bin/xterm*.

Uruchomić program *xterm* można na jeden z następujących sposobów.

- ♦ Można dodać do menu nową pozycję *xterm*, zgodnie ze wskazówkami zamieszczonymi w rozdziale 5., w podpunkcie „Dodawanie nowych pozycji do innych menu”, a następnie ją wybrać.
- ♦ Można też utworzyć na panelu (należy użyć polecenia *Dodaj do panelu*) nowy aktywator dla aplikacji *xterm*, a następnie go uruchomić. Więcej informacji

Rysunek 6.3.
Aplikacja xterm



na temat konfigurowania paneli można znaleźć w rozdziale 5., w punkcie „Dostosowywanie menu”.

- ♦ Można także utworzyć na pulpicie aktywator programu xterm; w tym celu należy kliknąć prawym przyciskiem myszy z kursorem umieszczonym na ekranie i wybrać z menu kontekstowego opcję *Utwórz aktywator*. Zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 6.4, w którym należy uzupełnić odpowiednie pola. Wtedy program xterm będzie można uruchamiać wprost z pulpitu.

Rysunek 6.4.
*Tworzenie na pulpicie
aktywatora programu
xterm*

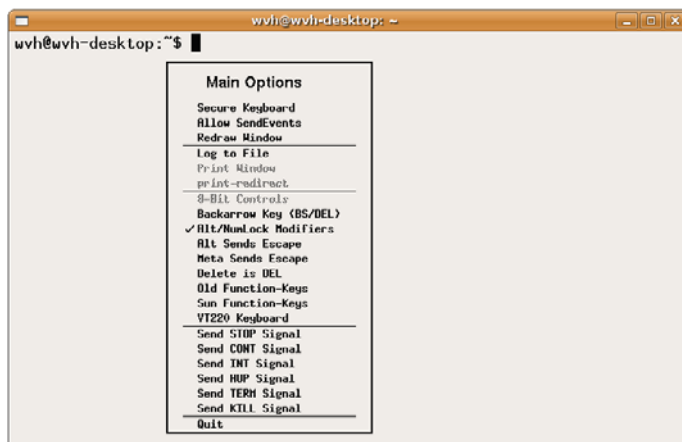


- ♦ I na koniec można wykonać w terminalu polecenie `xterm`, jest to pewna rekursja, ale całkowicie wykonalna.

Wielu nowych użytkowników Ubuntu, którzy mieli już doświadczenia z xterm, stwierdzi zapewne, że ten program bardziej im odpowiada niż terminal. Aby zapoznać się z możliwościami programu xterm, warto sprawdzić stronę podręcznika `man`; wystarczy uruchomić xterm, a następnie wykonać polecenie `man xterm`.

Ponieważ xterm jest natywnym programem systemu X Window, a nie GNOME czy KDE, dlatego też nie oferuje paska menu czy paska narzędzi ułatwiających jego konfigurację (patrz rysunek 6.3). Aby uzyskać dostęp do opcji konfiguracyjnych programu, należy wcisnąć i przytrzymać klawisz *Ctrl*, a następnie wcisnąć prawy, lewy lub środkowy przycisk myszy (kursor musi być ustawiony nad oknem programu xterm). Na rysunku 6.5 przedstawiam menu widoczne po wcisnięciu klawisza *Ctrl* i lewego przycisku myszy.

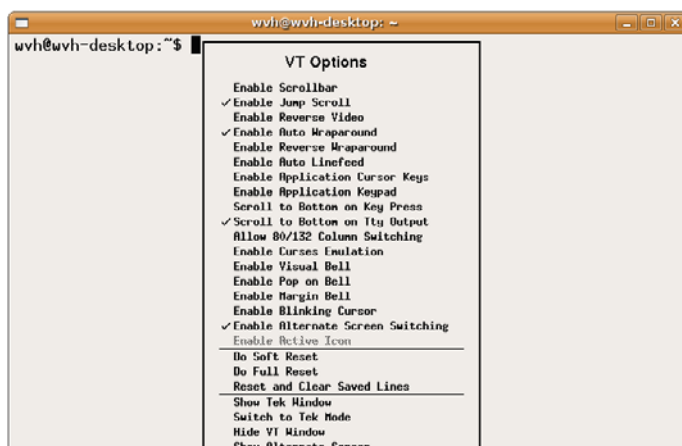
Rysunek 6.5.
Menu *Main Options*
programu xterm



Każde menu konfiguracyjne programu xterm oferuje dostęp do innych opcji. Oto one.

- ♦ ***Ctrl*+lewy przycisk myszy** — menu *Main Options* umożliwia konfigurację wysyłanych sygnałów oraz kombinacji klawiszowych w programie xterm.
- ♦ ***Ctrl*+środkowy przycisk myszy** — menu *VT Options* umożliwia skonfigurowanie opcji charakterystycznych dla emulacji terminala. Nazwa VT Options pochodzi od nazwy terminala DEC VT-100, który jest domyślnie emulowany w programie xterm. Na rysunku 6.6 przedstawiam menu *VT Options*.

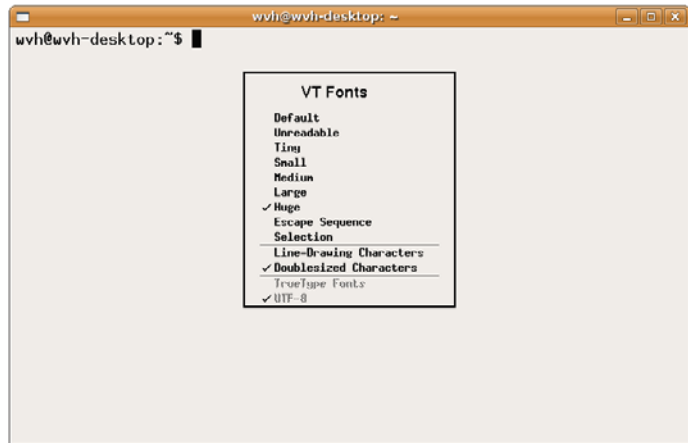
Rysunek 6.6.
Menu *VT Options*



- ♦ **Ctrl+prawy przycisk myszy** — menu *Fonts* pozwala użytkownikowi kontrolować rozmiar czcionek używanych w programie xterm. Na rysunku 6.7 widoczne jest menu *Fonts*.

Rysunek 6.7.

Menu *Fonts*
programu xterm



Domyślnie program xterm uruchamiany jest jako kopia powłoki głównej, ale jednym z najpowszechniejszych sposobów jego używania jest automatyczne uruchomienie innej aplikacji w oknie xterm; korzystamy wtedy z opcji `-e` (ang. *execute*, wykonać), po której należy podać nazwę programu, jaki ma zostać włączony. Oto przykład: poniższe polecenie uruchomi program xterm, który z kolei uruchomi polecenie `ssh` służące do połączenia się — w tym przypadku — z systemem writing.vonhagen.org:

```
$ xterm -e ssh writing.vonhagen.org
```

Użytkownik zostanie poproszony o podanie hasła, po czym uzyska dostęp do wskazanego systemu (więcej informacji na temat łączenia się z innymi systemami zamieszczam w rozdziale 15., „Łączenie się z innymi systemami”). Ponieważ w odróżnieniu od terminala GNOME program xterm nie obsługuje wielu zakładek (co objaśniłem powyżej), dlatego też polecenie `xterm -e` jest jednym z najpowszechniejszych sposobów, w jaki użytkownicy Ubuntu mogą otworzyć w osobnym oknie aplikację niekorzystającą z trybu graficznego.

Często używane komendy linii poleceń

W dalszej części rozdziału zamieszczam wprowadzenie do narzędzi linii poleceń oraz używania opcji i argumentów, za przykład posłuży standardowe polecenie systemu Linux — `ls`. Wyświetlanie listy plików i katalogów jest jedną z najczęściej wykonywanych czynności, chociaż, jak wspominałem wcześniej, w Linuksie istnieją setki różnorodnych poleceń. Niestety, nie można omówić ich wszystkich, ponieważ oznaczałoby to, że niniejsza książka musiałaby osiągnąć rozmiary małej ciężarówki. Dlatego skupię się na najczęściej wykonywanych zadaniach i związanych z nimi poleceniach. Zamieszczam też wskazówki, jak zorientować się w gąszczu zaszyfrowanych poleceń i odnaleźć właściwe narzędzie linii poleceń.

Poruszanie się po systemie plików Linuksa

Tak jak podczas pracy w trybie graficznym, również w trakcie korzystania z konsoli użytkownik zapewne będzie chciał nadać logiczną strukturę plikom i katalogom. Następnym etapem jest użycie polecenia `cd` (ang. *change directory* — zmień katalog), które robi dokładnie to, czego można by oczekiwać po nazwie. Innymi słowy, polecenie `cd` zmienia bieżący katalog na ten, który został użyty jako argument. Oto przykład: aby zmienić katalog na taki o nazwie *zastosuj_test*, należy po prostu wykonać następujące polecenie:

```
$ cd zastosuj_test
```

Kiedy się nie wie, gdzie się przebywa, sytuacja może być nieprzyjemna. Na szczęście, jeżeli użytkownik chce sprawdzić, jak nazywa się bieżący katalog, wówczas wystarczy użyć polecenia `pwd` (ang. *process working directory*). Wyświetli ono pełną ścieżkę dostępu do aktualnie przetwarzanego katalogu (w tym przypadku w powłoce), np.:

```
$ pwd  
/home/wvh/zastosuj_test
```

Aby zmienić katalog bieżący na jakikolwiek inny, wystarczy użyć polecenia `cd`, podając jako argument pełną ścieżkę dostępu do katalogu docelowego:

```
$ cd /usr/local  
$ pwd  
/usr/local
```

Jak można zauważyć, powłoka pracuje teraz w katalogu */usr/local* i wszystkie kolejne polecenia będą wykonywane w odniesieniu do niego właśnie. Wprowadzono też pewne udogodnienie: jeżeli polecenie `cd` zostanie użyte bez jakiegokolwiek argumentu, wówczas bieżący katalog zostanie automatycznie zmieniony na katalog domowy użytkownika, w którym zwykle znajduje się większość jego plików i katalogów.



Podczas pracy w linii poleceń często zdarza się, że użytkownik musi przejść do innego katalogu, wykonać jedną czy dwie operacje, a następnie powrócić do poprzedniego miejsca. Podczas częstego przechodzenia z katalogu do katalogu wygodnie mieć gdzieś zapisaną informację typu „skąd przyszedłem”. Aby zachować czas i energię umysłową użytkownika, w większości linuksowych powłok umieszczono polecenia o nazwie `pushd` i `popd`, które właśnie do tego służą. Za pomocą polecenia `pushd` można zmienić bieżący katalog, ale jego nazwa zostanie umieszczona w miejscu zwanym **stosem**, dzięki czemu po wykonaniu danej czynności wystarczy wykonać polecenie `popd`, które spowoduje zmianę bieżącego katalogu na używany poprzednio, a informacja o nim zostanie usunięta ze stosu. Użytkownicy niezaznajomieni z koncepcją stosu łatwo mogą zrozumieć jego działanie, wyobrażając sobie stosy naczyń w restauracji. Umieszczenie czegoś w stosie (w systemie) przebiega analogicznie do umieszczenia kolejnego naczynia na stercie. Usunięcie czegoś działa podobnie, przy czym uzyskuje się bezpośredni dostęp do rzeczy znajdującej się poniżej.

Kopiowanie, przenoszenie, zmiana nazwy i usuwanie plików i katalogów

Kopiowanie, przenoszenie i zmiana nazw plików to jedne z najpowszechniejszych zadań wykonywanych przez użytkowników komputerów obojętnie, czy korzystają z linii poleceń, czy pracują z graficznym interfejsem. Niezależnie od tego, czy plik jest kopiowany

po to, aby na jego podstawie rozpocząć nową pracę, czy też użytkownik chce uporządkować pliki i katalogi, aby łatwiej wyszukiwać dane elementy, kopiowanie i przenoszenie plików jest tak samo proste do wykonania zarówno w linii poleceń, jak i w środowisku graficznym. Jedyna różnica polega na tym, że w pierwszym przypadku należy wskazać i kliknąć, a w drugim — wpisać dane. W wielu przypadkach miliardy opcji oferowanych przez narzędzia linii poleceń dają większe możliwości kontroli przebiegu operacji, niż ma to miejsce podczas korzystania z interfejsu graficznego.

Polecenie `cp` służące do kopiowania plików podtrzymuje uniksową i linuksową tradycję nazywania poleceń tak, aby ich nazwa była jak najkrótsza. Aby skopiować plik, należy użyć polecenia `cp`, po którym trzeba podać nazwę istniejącego pliku oraz nazwę, jaką ma posiadać jego kopia. Aby skopiować plik o nazwie *plik.txt* i nadać kopii nazwę *nowy_plik.txt*, należy wykonać następujące polecenie:

```
$ cp plik.txt nowy_plik.txt
```



Podczas określania nazwy dla tworzonej kopii trzeba zachować szczególną ostrożność. Jeżeli w folderze, w którym wykonywana jest kopia, znajduje się już plik o nazwie takiej, jaką będzie miała kopia, wówczas ten plik, w efekcie działania polecenia `cp`, zostanie zastąpiony. W niniejszym rozdziale, w punkcie „Definiowanie i używanie aliasów”, zamieszczam informacje, w jaki sposób bezpiecznie używać polecenia `cp`.

Jeżeli użytkownik poda pełną ścieżkę dostępu do pliku źródłowego czy docelowego, wówczas polecenie `cp` będzie mogło zostać wykonane w dowolnym miejscu systemu Linux, dla którego uprawnienia do tworzenia plików będzie miał użytkownik. Aby np. skopiować plik *plik.txt* z katalogu domowego do katalogu */tmp* (w którym wszyscy mogą tworzyć pliki) i zapisać go pod nazwą *nowyplik.txt*, można wykonać następujące polecenie (zadziała ono niezależnie od bieżącego katalogu):

```
$ cp /home/wvh/plik.txt /tmp/nowyplik.txt
```

Można też użyć polecenia `cp` do kopiowania plików do innych katalogów z określeniem nazwy katalogu docelowego. A zatem jeżeli użytkownik chciałby skopiować plik *plik.txt* z katalogu domowego do katalogu */tmp*, ale chciałby zachować oryginalną nazwę pliku, powinien użyć następującego polecenia:

```
$ cp /home/wvh/plik.txt /tmp
```

To polecenie utworzy plik */tmp/plik.txt* będący dokładną kopią pliku */home/wvh/plik.txt*.

Do prostego kopiowania plików dla polecenia `cp` nie są potrzebne żadne specjalne, dodatkowe opcje. Można jednak podczas kopiowania zastosować opcję `-r` (rekursja). I tak, aby skopiować zawartość katalogu o nazwie *zastosuj_test* do nowego katalogu o nazwie *kopia*, należy użyć następującego polecenia:

```
$ cp -r zastosuj_test kopia
```

Po zakończeniu działania polecenia nowy katalog będzie dokładną kopią katalogu *zastosuj_test*. Jeżeli użytkownik chciałby mieć pełną kontrolę nad całym procesem, może wówczas dodatkowo zastosować opcję `-v`, wówczas podczas kopiowania plików za pomocą polecenia `cp` za każdym razem będzie wyświetlany komunikat o kopiowanych plikach. To polecenie wyświetli dane wyjściowe tego typu:

```
$ cp -rv zastosuj_test kopia
`zastosuj_test' -> `kopia'
`zastosuj_test/libxml2' -> `kopia/libxml2'
`zastosuj_test/libxml2/xmlops.h' -> `kopia/libxml2/xmlops.h'
`zastosuj_test/netdev' -> `kopia/netdev'
`zastosuj_test/netdev/devname.h' -> `kopia/netdev/devname.h'
`zastosuj_test/system' -> `kopia/system'
`zastosuj_test/system/libxml2' -> `kopia/system/libxml2'
`zastosuj_test/system/libxml2/xmlops.h' -> `kopia/system/libxml2/xmlops.h'
`zastosuj_test/system/netdev' -> `kopia/system/netdev'
`zastosuj_test/system/netdev/devname.h' -> `kopia/system/netdev/devname.h'
```

Jak można się spodziewać, rekursywne kopiowanie za pomocą polecenia `cp` spowoduje również wykonanie kopii wszystkich podkatalogów.

Jak wspominałem wcześniej, określanie nazwy katalogu docelowego podczas kopiowania powoduje utworzenie pliku o tej samej nazwie w katalogu docelowym. Ten sam mechanizm działa podczas określania nazwy katalogu docelowego podczas kopiowania rekursywnego. Jeżeli podczas próby skopiowania katalogu do nowego katalogu z nową nazwą okaże się, że katalog z tą nazwą już tam jest, wówczas polecenie `cp` utworzy katalog z oryginalną nazwą jako podkatalog w katalogu docelowym.

Przenoszenie plików oraz zmiana ich nazwy w linii poleceń jest równie prosta. W Linuksie nie ma obecnie osobnego polecenia do zmiany nazwy, za pomocą komendy `mv` można jednocześnie przenosić istniejące pliki i foldery do nowych miejsc, a także zmieniać ich nazwy, co jest częścią operacji ich przenoszenia. Aby np. zmienić nazwę pliku *plik.txt* na *nowyplik.txt*, należy wykonać następujące polecenie:

```
$ mv plik.txt nowyplik.txt
```

Polecenie `mv` ma takie same wymagania w zakresie podawania pełnych ścieżek dostępu i podczas określania nazw katalogów, jak ma to miejsce w poleceniu `cp`. Oto przykład, poniższe polecenie przeniesie plik */home/wvh/plik.txt* do katalogu */tmp* i zapisze pod nową nazwą *nowyplik.txt*:

```
$ mv /home/wvh/plik.txt /tmp/nowyplik.txt
```

Jeżeli plik miałby po przeniesieniu zachować oryginalną nazwę, wówczas należałoby użyć następującego polecenia:

```
$ mv /home/wvh/plik.txt /tmp
```

Inaczej niż ma to miejsce w przypadku polecenia `cp`, komenda `mv` nie wymaga żadnych specjalnych znaczników podczas przenoszenia katalogów czy zmieniania ich nazw. Aby zmienić nazwę katalogu *zastosuj_test* na *kopia*, należy użyć następującego polecenia:

```
$ mv zastosuj_test kopia
```

Tak jak w przypadku polecenia `cp`, jeżeli katalog o nazwie *kopia* już istnieje, wówczas polecenie `mv` przeniesie katalog *zastosuj_test* do katalogu *kopia*, zachowując jego oryginalną nazwę. Polecenie `mv` udostępnia też tryb `-v` (ang. *verbose*), dzięki któremu użytkownik będzie na bieżąco informowany, co dzieje się w trakcie wykonywania danej operacji, choć podczas zmiany nazwy nie jest to specjalnie ekscytujące:

```
$ mv -v zastosuj_test kopia
`zastosuj_test' -> `kopia'
```

W tym przypadku polecenie `mv` przeprowadziło tylko zmianę nazwy katalogu. Ze względu na sposób, w jaki Linux działa podczas tworzenia czy przenoszenia katalogów umieszczonych na różnych partycjach dysku, każdy plik i podkatalog najpierw jest kopiowany w nowe miejsce, a następnie usuwany ze starego, jak w poniższym przykładzie:

```
$ df . /tmp
System plików      bl.  1K B      użyte dostępne %uż. zamont. na
/dev/sda8          160010472 117909468 42101004 74% /home
/dev/sda5           4200824  984200    3216624 24% /tmp
$ mv -v zastosuj_test /tmp/kopia
`zastosuj_test' -> `/tmp/kopia'
`zastosuj_test/libxml2' -> `/tmp/kopia/libxml2'
`zastosuj_test/libxml2/xmlops.h' -> `/tmp/kopia/libxml2/xmlops.h'
`zastosuj_test/netdev' -> `/tmp/kopia/netdev'
`zastosuj_test/netdev/devname.h' -> `/tmp/kopia/netdev/devname.h'
`zastosuj_test/system' -> `/tmp/kopia/system'
`zastosuj_test/system/libxml2' -> `/tmp/kopia/system/libxml2'
`zastosuj_test/system/libxml2/xmlops.h' -> `/tmp/kopia/system/libxml2/xmlops.h'
`zastosuj_test/system/netdev' -> `/tmp/kopia/system/netdev'
`zastosuj_test/system/netdev/devname.h' -> `/tmp/kopia/system/netdev/devname.h'
`zastosuj_test/new_file' -> `/tmp/kopia/new_file'
removed `zastosuj_test/libxml2/xmlops.h'
removed directory: `zastosuj_test/libxml2'
removed `zastosuj_test/netdev/devname.h'
removed directory: `zastosuj_test/netdev'
removed `zastosuj_test/system/libxml2/xmlops.h'
removed directory: `zastosuj_test/system/libxml2'
removed `zastosuj_test/system/netdev/devname.h'
removed directory: `zastosuj_test/system/netdev'
removed directory: `zastosuj_test/system'
removed `zastosuj_test/new_file'
removed directory: `zastosuj_test'
```

Jak widać, duża część danych wyjściowych jest podobna do uzyskiwanych podczas korzystania z polecenia `cp`. W powyższym przykładzie najpierw użyłem polecenia `df`, a zatem widać, że bieżący katalog (`.`) oraz katalog docelowy są umieszczone w różnych systemach plików. Poniżej można zobaczyć różnicę w działaniu trybu *verbose* w poleceniu `mv`.

Podobną do kopiowania i przenoszenia plików operacją jest ich usuwanie. To prosta operacja, niezależnie od tego, czy użytkownik chce usunąć pliki, aby zrobić miejsce dla przyszłych projektów, usunąć kopie zapasowe projektów już ukończonych, czy usunąć pliki, które nie powinny wpaść w niepowołane ręce. Polecenie służące do usuwania plików i katalogów to `rm`, które — jak widać — zgodne jest z zasadą „im mniej wpisywania, tym lepiej”.

Aby usunąć pojedynczy plik z systemu, wystarczy wpisać polecenie `rm`, a po nim podać nazwę pliku, w przykładzie usunę *plik.txt*:

```
$ rm plik.txt
```

I plik zniknął. Podobnie jak poprzednio, polecenie posiada opcję `-r`, która umożliwia usuwanie całych katalogów. Aby definitywnie usunąć katalog o nazwie *zastosuj_test*, należy użyć następującego polecenia:

```
$ rm -r zastosuj_test
```



Aby usunąć pusty katalog, trzeba skorzystać z polecenia `rmdir`, ale nie jest to częsty scenariusz. Katalogi są zwykle tworzone, aby coś w nich przechowywać, a nie po to, żeby były puste. Do usuwania katalogów najłatwiej używać polecenia `rm -r`, dzięki czemu nie trzeba najpierw samodzielnie usuwać ich zawartości, ale to już zależy od preferencji użytkownika.

Użytkownicy, którzy wcześniej korzystali z systemów Microsoft Windows, zauważą zapewne różnicę polegającą na tym, że w Microsoft Windows odzyskanie usuniętych plików jest banalnie proste. Dzieje się tak dlatego, że podczas ich usuwania system wymazuje tylko wpisy, które identyfikowały położenia plików lub folderów. Zupełnie inaczej jest w Linuksie. Podczas usuwania plików lub katalogów cała przestrzeń dysku związana z plikiem, katalogiem (oraz jego zawartość) jest przywracana na listę wolnego miejsca dostępnego w systemie. Odzyskanie usuniętych plików czy katalogów w Linuksie również jest możliwe, ale jest dużo trudniejsze i wymaga pomocy kogoś, kto naprawdę zna się na szczegółach systemu plików.

Wadą takiego rozwiązania jest to, że w razie pomyłkowego usunięcia plików czy katalogów nie uda się ich łatwo odzyskać. Aby pomóc użytkownikowi w zabezpieczeniu się przed pomyłkowym usunięciem plików, w tym rozdziale, w punkcie „Definiowanie i używanie aliasów” zamieszczam wskazówki, jak skonfigurować polecenie `rm`, aby każde usunięcie pliku trzeba było potwierdzić.



Polecenia `cp`, `mv` i `rm` posiadają wiele opcji, które nie zostały tutaj opisane. Aby wyświetlić listę dostępnych opcji, należy dla każdego z tych poleceń użyć komendy `man` lub `info`, co spowoduje wyświetlenie systemu elektronicznej informacji.

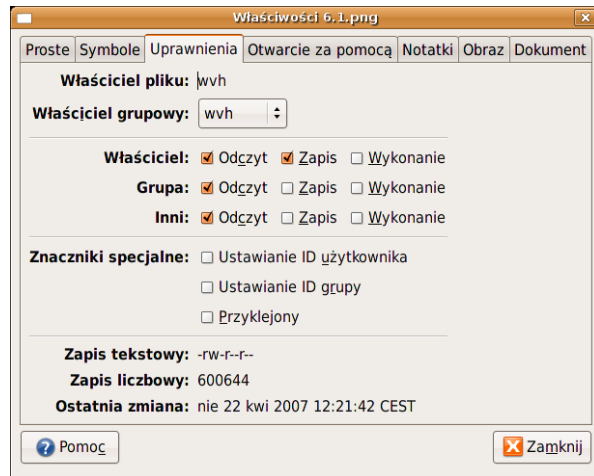
Zmiana uprawnień do plików i katalogów

W rozdziale 4. zamieściłem wstęp do linuksowej koncepcji grup i użytkowników, a także informacje wprowadzające na temat uprawnień do plików i katalogów widocznych podczas wyświetlania ich zawartości. Można było je przeczytać również w niniejszym rozdziale podczas omawiania polecenia `ls`. Możliwość ochrony plików, tak aby tylko określone użytkownicy lub grupy mieli do nich dostęp lub mogli je uruchamiać, jest bardzo wygodna, ale od razu nasuwa się pytanie, jak to zrobić. W środowisku graficznym uprawnień do plików i katalogów można łatwo zmieniać, korzystając np. z Nautilusa. Wystarczy kliknąć wybrany plik prawym przyciskiem myszy, wybrać z menu kontekstowego opcję *Właściwości* i przejść do zakładki *Uprawnienia* (patrz rysunek 6.8). Na szczęście, zmiana uprawnień do plików i katalogów wykonywana w linii poleceń jest równie prosta (jeśli nawet nie prostsza).

Polecenie `chmod` (ang. *change mode*, zmień tryb) umożliwia zmianę wszelkich uprawnień przypisanych dla danego pliku czy katalogu, którego właścicielem jest użytkownik. Polecenie `chmod` obsługuje dwie metody zmiany uprawnień do plików i katalogów:

Rysunek 6.8.

Ustawienia
uprawnień
w Nautilusie



- ♦ korzystanie z wartości ósemkowych reprezentujących bity w polu uprawnień plików lub folderów,
- ♦ korzystanie z przyjaźniejszego, literowego prezentowania uprawnień dla użytkowników będących właścicielami pliku lub folderu (u), grupy przypisanej do danego pliku lub katalogu (g), innych użytkowników systemu nienależących do tej grupy (o) oraz wszystkich użytkowników systemu (a). Do dodawania lub usuwania uprawnień służą symbole plus i minus. Ta procedura znana jest pod nazwą zmiany uprawnień do plików w trybie symbolicznym.

W tej części uwagę skupię na drugiej metodzie, ponieważ wielu użytkowników może nie pamiętać wszystkich informacji dotyczących wartości ósemkowych, a tryb symboliczny jest łatwiejszy w wizualizacji i używaniu. Jako przykład zostanie użyty plik *witaj.sh*, który jest skryptem powłoki wyświetlającym na ekranie prosty komunikat „Witaj świecie!”. Najpierw należy sprawdzić domyślne uprawnienia do pliku:

```
$ ls -l witaj.sh
-rw-r--r-- 1 wvh wvh 36 2007-05-26 16:30 witaj.sh
```

Zgodnie z wyjaśnieniami zamieszczonymi w rozdziale 4., początkowe dane wyjściowe określają aktualne uprawnienia do pliku. Można podzielić je na cztery grupy: pierwszy myślnik określa typ obiektu (myślnik oznacza, że jest to plik), następne trzy grupy znaków oznaczają kolejno uprawnienia: właściciela pliku (w tym przypadku *wvh*), grupy skojarzonej z plikiem (również o nazwie *wvh*) oraz innych użytkowników. W powyższym przykładzie możemy odczytać, że użytkownik *wvh* może odczytywać i zapisywać plik, ale pozostali członkowie grupy *wvh* mogą tylko odczytywać plik, podobnie zresztą jak pozostali użytkownicy.

Ponieważ przykładowy plik to skrypt powłoki, zapewne użytkownik będzie chciał mieć możliwość uruchamiania go. Aby nadać plikowi atrybut wykonywalności (przez właściciela), należy użyć następującego polecenia:

```
$ chmod u+x witaj.sh
```

Po ponownym wyświetleniu informacji o pliku można zaobserwować, że może on teraz być uruchamiany przez właściciela:

```
$ chmod u+x witaj.sh
$ ls -al witaj.sh
-rwxr--r-- 1 wvh wvh 36 2007-05-26 16:30 witaj.sh
```

Aby umożliwić również innym członkom grupy *wvh* oraz wszystkim innym użytkownikom komputera uruchamianie tego skryptu, należy wykonać następujące polecenie i sprawdzić jego efekt:

```
$ chmod g+x witaj.sh
$ ls -al witaj.sh
-rwxr-xr-- 1 wvh users 36 2007-05-26 16:30 witaj.sh
$ chmod a+x witaj.sh
$ ls -al witaj.sh
-rwxr-xr-x 1 wvh users 36 2007-05-26 16:30 witaj.sh
```

To całkiem sporo wpisywania i powtarzania tych samych operacji. Na szczęście, polecenie `chmod` umożliwia łączenie wielu operacji zmiany uprawnień do plików. Jeżeli użytkownik chciałby, aby tylko on sam miał możliwość uruchamiania danego pliku, wówczas powinien wykonać następujące polecenie:

```
$ chmod go-x witaj.sh
$ ls -l witaj.sh
-rwxr--r-- 1 wvh users 36 2007-05-26 16:30 witaj.sh
```

Warto zwrócić uwagę na opcję *o*, która oznacza wszystkich użytkowników spoza grupy *wvh*, jeżeli została użyta opcja *a*, oznaczałoby to wszystkich użytkowników systemu.

Za pomocą polecenia `chmod` można dodawać i usuwać prawa do odczytu i zapisu niemal każdej grupie użytkowników. Uprawnienia dla plików działają podobnie w odniesieniu do katalogów, z wyjątkiem bitu wykonywalności, który w przypadku katalogów umożliwia ich przeszukiwanie. I tak, kiedy np. użytkownik utworzy katalog, którego nikt — z wyjątkiem użytkownika — nie będzie mógł odczytać, ponieważ zostanie usunięte uprawnienie do odczytu (*r*), to jeżeli pozostawione będzie uprawnienie do przeszukiwania, inni użytkownicy będą mogli i tak wykonywać znajdujące się w nim pliki i skrypty. Nie będzie można przeglądać zawartości, ale znając nazwy plików, będzie je można uruchamiać.



Podobnymi poleceniami, których można używać w odniesieniu do plików, są `chown` (ang. *change owner*, zmień właściciela) i `chgrp` (ang. *change group*, zmień grupę). Aby jednak je wykonywać, należy użyć polecenia `sudo` do uzyskania specjalnych uprawnień. Jeżeli użytkownik nie jest administratorem systemu, wówczas zapewne rzadko będzie korzystał z tych poleceń. Na stronach pomocy elektronicznej można znaleźć dodatkowe informacje o sposobie korzystania z tych narzędzi.

Wyszukiwanie odpowiednich poleceń

Powłoka to przyjazne miejsce do wykonywania komend, jeśli tylko się wie, co zrobić i jakiego polecenia użyć. Niestety, ze względu na obecną w Linuksie tendencję do skracania długich nazw przez opuszczanie zbędnych liter, często dość trudno odgadnąć, którego polecenia użyć do danego zadania.

Na szczęście, elektroniczna pomoc dostępna za pośrednictwem polecenia `man` jest wyposażona w naprawdę wygodną opcję wyszukiwania (`-k`), za której pomocą można odnaleźć słowo kluczowe lub daną frazę. Jeżeli np. użytkownik nie zna polecenia `chmod`, a chciałby otrzymać listę poleceń, które mają coś wspólnego z uprawnieniami do plików, wówczas powinien użyć następującej komendy:

```
$ man -k permissions
chmod (1) - change file access permissions
```

Oczywiście, podobnie jak podczas wyszukiwania w internecie, sprawę utrudnia fakt, że trzeba znać odpowiednie słowo kluczowe. Ale prawdopodobieństwo trafienia zawsze jest. Można wyszukiwać pojedyncze słowa lub ciągi umieszczone w cudzysłowach, zawierające poszukiwane terminy.

Jeżeli użycie polecenia `man -k` przyniesie efekt w postaci listy poleceń, ale próba wykonania któregoś z nich zakończy się niepowodzeniem (np. zostanie wyświetlony komunikat *command not found*), oznacza to, że katalog, w którym przechowywane jest dane polecenie, nie jest jednym z tych, w jakich zwykle się one znajdują. Wtedy należy użyć polecenia `whereis`, które pomoże zlokalizować zaginioną komendę.

```
$ man -k "physical volume"
pvchange (8) - change attributes of a physical volume
pvdisplay (8) - display attributes of a physical volume
pvremove (8) - remove a physical volume
pvs (8) - report information about physical volumes
pvscan (8) - scan all disks for physical volumes
vgextend (8) - add physical volumes to a volume group

$ pvscan
bash: pvscan: command not found

$ whereis pvscan
pvscan: /sbin/pvscan /usr/share/man/man8/pvscan.8.gz
```

Polecenie `whereis` wskazało lokalizację poszukiwanej komendy oraz jej dokumentacji. Aby wykonać poszukiwane polecenie, należy wskazać pełną ścieżkę dostępu, którą w tym przypadku jest `/sbin/pvscan`.

Korzystanie z powłoki bash

Teraz, po krótkim wstępie i przedstawieniu podstawowych poleceń, czas na zapoznanie się z ciekawymi aspektami pracy z powłoką i jej konfigurowaniem. Oczywiście, można po prostu wpisywać kolejne komendy w powłoce, ale istnieje tak wiele mechanizmów ułatwiających korzystanie z niej, że warto się z nimi zapoznać i zacząć z nich korzystać. Niektóre z tych funkcji pomogą zaoszczędzić sporo czasu, pisania i ciągłego powtarzania tych samych poleceń zwłaszcza wtedy, kiedy użytkownik korzysta z podobnych komend, podczas wpisywania długich i skomplikowanych nazw czy argumentów dla poleceń itd.

Korzystanie z historii poleceń

Wszystkie powłoki linuksowe są wyposażone w mechanizm zapisywania wykonanych działań, który jest sposobem na przywołanie i ponowne wykonanie poleceń, jakie zostały już zastosowane. Aby wyświetlić listę użytych poleceń, wystarczy użyć komendy history, oto przykładowe dane wyjściowe:

```
$ history
221 ssh 192.168.0.2
222 sudo apt-get install samba
223 sudo apt-get install smbfs
224 sudo cp /etc/samba/smb.conf /etc/samba/smb.conf_backup
225 sudo gedit /etc/samba/smb.conf
226 sudo testparm
227 sudo smbpasswd -a nazwa_użytkownika
228 sudo smbpasswd -a zebii
229 sudo gedit /etc/samba/smbusers
230 sudo smbpasswd -a tomek
231 sudo gedit /etc/samba/smb.conf
232 ssh zebii
[dalsza część została usunięta]
```

Powyższa lista to przykład zaczerpnięty wprost z komputera. Ilość zapisywanych poleceń jest określona przez zmienną środowiskową HISTSIZE (więcej informacji na temat zmiennych środowiskowych zamieszczam w tym rozdziale, w punkcie „Zmienne środowiskowe”).

Istnieje kilka sposobów na przywołanie lub powtórne wykonanie już użytych poleceń:

- ♦ użycie wbudowanego w powłokę polecenia `fc`,
- ♦ użycie mechanizmu powłoki służącego do interakcji z poleceniami, które zostały już wpisane (znanego pod nazwą **biblioteki readline**), dzięki czemu można poruszać się po zapisanej historii użycia komend wprost w linii poleceń,
- ♦ użycie wykrzyknika do zidentyfikowania polecenia, które ma zostać ponownie użyte.

Ponieważ sam zwykle używam jednego z dwóch ostatnich mechanizmów, zostaną one omówione poniżej.

Można wyświetlić i wykonać ponownie każde wykorzystane już polecenie, używając skrótu klawiszowego `Ctrl+P`. Za każdym razem, kiedy zostanie zastosowany, powłoka wyświetli użyte poprzednio polecenie. Kiedy zostanie wyświetlona komenda, której użytkownik szukał, wystarczy wcisnąć klawisz `Enter`. Można też edytować takie polecenie; w tym celu można użyć kombinacji klawiszy znanych z edytora emacs: `Ctrl+B` przesuwa kursor o jeden znak do tyłu; `Ctrl+F` przesuwa kursor o jeden znak do przodu, klawiszy `Backspace` i `Delete`, które usuwają znak odpowiednio za i przed kursorem itd.

Jeżeli jednak poszukiwane polecenie znajduje się gdzieś na początku listy wpisanych komend, wówczas przemieszczanie się przez setki wpisów może być nieco nużące. W takim przypadku można użyć wykrzyknika w celu zidentyfikowania danego polecenia w pliku z historią. Można to zrobić następująco:

- ♦ **identyfikacja na podstawie kolejnego numeru** — każde polecenie na liście poprzedzone jest kolejnym numerem; po wyświetleniu listy można sprawdzić, który numer ma dane polecenie, a następnie przywołać je, wpisując przed numerem znak wykrzyknika; i tak (przy użyciu listy zamieszczonej wcześniej), wpisanie w powłoce polecenia `!222` spowoduje wykonanie polecenia `sudo apt-get install samba`,
- ♦ **identyfikacja na podstawie nazwy** — można przywołać wykonane już polecenie, używając jego nazwy, w tym celu należy ją wprowadzić i poprzedzić znakiem wykrzyknika; np. korzystając z wyświetlonej powyżej listy użytych poleceń, można wpisać w powłoce `!ssh`, zostanie wówczas wykonane polecenie `ssh zebii`.



Jeżeli użytkownik chciałby przywołać użyte poprzednio polecenie, ale przed jego wykonaniem poddać je edycji, powinien użyć skrótu klawiszowego `Ctrl+P`, który wyświetla polecenie bez jego wykonania. Po wyświetleniu polecenia można poddać je edycji i dopiero później wcisnąć klawisz `Enter`, który spowoduje jego wykonanie.

Jeżeli użytkownik wprowadził kilka podobnych poleceń, może wyświetlić ich listę, ograniczając jej zakres określonym ciągiem liter. I tak, polecenie `history | grep vi` spowoduje użycie potoku, w efekcie czego (za pomocą polecenia `grep`) zostaną wyświetlone tylko te linie, w których znajduje się ciąg `vi`. Więcej informacji na temat takiego używania poleceń zamieszczam w tym rozdziale, w punkcie zatytułowanym „Potoki i przekierowanie danych”.

Mechanizm zapisujący historię korzystania z powłoki `bash` jest niesłychanie użyteczny, powyższy opis ma na celu jedynie wskazanie kilku możliwości zapisywania, przywołania i ponownego wykonywania poleceń. Aby poznać wszelkie inne szczegóły dotyczące tego narzędzia, należy użyć polecenia `info bash`, które udostępni pomoc elektroniczną.

Uzupełnianie nazw plików

Jak można się przekonać z lektury niniejszego rozdziału, linia poleceń to bardzo pomocny i elastyczny mechanizm umożliwiający wykonywanie komend; inną rzeczą, którą można zauważyć, jest fakt, że celem wszystkich programistów Uniksa i Linuksa jest skrócenie do możliwego minimum nazw wszelkich używanych poleceń. Daje to w efekcie takie zaszyfrowane polecenia jak `ls` czy `pwd`, które z drugiej strony zaoszczędziły wiele trudu wpisującym je przez lata programistom.

Ta sama filozofia „wpisywania najmniej, jak jest to konieczne” ewoluowała w powłoce na różne sposoby. Jednym z najprzydatniejszych jest możliwość korzystania z klawisza `Tab` do uzupełniania nazw plików i poleceń. Wciśnięcie tego klawisza podczas wpisywania nazwy polecenia powoduje uzupełnienie nazwy — jeśli będzie tylko można ją jednoznacznie określić. Sam używam edytora `emacs` do wszystkich czynności związanych z plikami tekstowymi (w rozdziale 7., „Praca z plikami tekstowymi w Ubuntu”, zamieszczam więcej informacji na ten temat), zatem w ciągu dnia setki razy wpisuję polecenie `emacs to_jest_nazwa_danego_pliku.txt`. Aby zaoszczędzić sobie w prawdziwie linuksowym stylu wpisywania paru znaków, wystarczy wpisać `ema`, a następnie wcisnąć klawisz `Tab`. Powłoka uzupełni nazwę polecenia do postaci `emacs`, po czym można wstawić spację i wprowadzić nazwę pliku, który ma zostać poddany edycji.

Ale chwileczkę, jest jeszcze coś! Za pomocą klawisza *Tab* można też dopasowywać i uzupełniać nazwy plików. I tak, korzystając z powyższego przykładu, gdybym chciał edytować plik *to_jest_nazwa_danego_pliku.txt*, mógłbym wpisać kilka pierwszych znaków jego nazwy i wcisnąć klawisz *Tab*. Wówczas powłoka — miejmy nadzieję — dopasowałaby i uzupełniła nazwę pliku, dzięki czemu pozostałoby mi już tylko wcisnięcie klawisza *Enter* w celu rozpoczęcia edycji pliku. Jest to nawet częstsze wykorzystanie klawisza *Tab* niż w przypadku uzupełniania nazw poleceń, ponieważ te — jak już kilkakrotnie zwracałem uwagę — mają tendencję do bycia tak krótkimi, jak to tylko możliwe. Z drugiej jednak strony, ze względu na tę skrótowość często trudno zapamiętać ich dokładną nazwę, w takiej sytuacji klawisz *Tab* może służyć cenną pomocą.

Po wcisnięciu klawisza *Tab* powłoka dopasuje tak dużo z nazwy, jak to tylko możliwe. Jeżeli do wzorca będzie pasować więcej poleceń czy nazw plików, użycie klawisza nie przyniesie spodziewanego rezultatu, ponieważ system nie jest w stanie określić, które polecenie czy plik użytkownik ma na myśli. W takim przypadku należy dwukrotnie wcisnąć klawisz *Tab*, co spowoduje wyświetlenie listy dostępnych możliwości. Wtedy użytkownik będzie mógł wpisać kilka dodatkowych znaków, które jednoznacznie zidentyfikują odpowiedni plik lub polecenie.

Wyrażenie wieloznaczne

Innym przykładem filozofii „wpisywania najmniej, jak jest to konieczne” jest zdolność powłoki do identyfikowania grup plików, których nazwy posiadają podobny wzorec. Znaki używane do definiowania tych wzorców to **wyrażenia wieloznaczne** (ang. *wild-cards*). Powłoka obsługuje cztery główne rodzaje wyrażeń wieloznacznych:

- ♦ ? (znak zapytania) oznaczający dowolny znak,
- ♦ * (gwiazdka) oznaczająca zbiór znaków,
- ♦ {ciąg.1 ciąg2 } (nawiasy klamrowe) reprezentujące ciąg1 lub ciąg2,
- ♦ [znak1, znak2, ...] oznaczające znak lub zakres znaków podany w nawiasach kwadratowych.

Wyrażenia wieloznaczne są szczególnie przydatne podczas pracy w katalogach zawierających wiele plików o podobnych nazwach. Oto przykład katalogu zawierającego wiele plików dźwiękowych w różnych formatach:

```
$ ls
d1t01.flac  d1t04.flac  d1t07.flac  d1t10.flac  d2t02.flac
d1t01.mp3   d1t04.mp3   d1t07.mp3   d1t10.mp3   d2t02.mp3
d1t01.wav   d1t04.wav   d1t07.wav   d1t10.wav   d2t02.wav
d1t02.flac  d1t05.flac  d1t08.flac  d1t11.flac  d2t03.flac
d1t02.mp3   d1t05.mp3   d1t08.mp3   d1t11.mp3   d2t03.mp3
d1t02.wav   d1t05.wav   d1t08.wav   d1t11.wav   d2t03.wav
d1t03.flac  d1t06.flac  d1t09.flac  d2t01.flac  Horslips.info.txt
d1t03.mp3   d1t06.mp3   d1t09.mp3   d2t01.mp3   playlist.m3u
d1t03.wav   d1t06.wav   d1t09.wav   d2t01.wav
```

Czytelnik zapewne zwrócił uwagę, że nazwy plików zbudowane są według pewnej konwencji: *dysk <numer> ścieżka <numer> format*. Aby wyświetlić listę plików, które są pierwszą ścieżką z dowolnego dysku i mają format MP3, należy użyć następującego polecenia:

```
$ ls d?t01.mp3
d1t01.mp3  d2t01.mp3
```

Użyty znak zapytania odpowiada numerowi dysku, który w tym przypadku jest dowolny.

Podobnie, aby wyświetlić listę plików MP3, należy użyć symbolu `*` (gwiazdki) w następujący sposób:

```
$ ls *.mp3
d1t01.mp3  d1t03.mp3  d1t05.mp3  d1t07.mp3  d1t09.mp3  d1t11.mp3  d2t02.mp3
d1t02.mp3  d1t04.mp3  d1t06.mp3  d1t08.mp3  d1t10.mp3  d2t01.mp3  d2t03.mp3
```

Gdyby użytkownik chciał wyświetlić listę plików o rozszerzeniu FLAC i WAV, powinien użyć następującego polecenia:

```
$ ls *.{flac,wav}
d1t01.flac  d1t03.wav  d1t06.flac  d1t08.wav  d1t11.flac  d2t02.wav
d1t01.wav  d1t04.flac  d1t06.wav  d1t09.flac  d1t11.wav  d2t03.flac
d1t02.flac  d1t04.wav  d1t07.flac  d1t09.wav  d2t01.flac  d2t03.wav
d1t02.wav  d1t05.flac  d1t07.wav  d1t10.flac  d2t01.wav
d1t03.flac  d1t05.wav  d1t08.flac  d1t10.wav  d2t02.flac
```

Jeżeli użytkownik chciałby wyświetlić listę plików, które są odpowiednio siódmym i ósmym utworem z dysku pierwszego, w dodatku w formacie MP3, wówczas powinien użyć wyrażenia wieloznacznego w następujący sposób:

```
$ ls d1t0[78].mp3
d1t07.mp3  d1t08.mp3
```

Wyrażenia wieloznaczne można łączyć, co wyraźnie zwiększa ich możliwości. Oto przykład: poniższe polecenie umożliwia wyświetlenie listy plików, które są odpowiednio siódmym i ósmym utworem z dysku pierwszego i zapisane są w formatach FLAC i WAV:

```
$ ls d1t0[78].{flac,wav}
d1t07.flac  d1t07.wav  d1t08.flac  d1t08.wav
```

To kolejny bardzo użyteczny mechanizm, w który wyposażona jest powłoka. Wyrażenia wieloznaczne, takie jak te wymienione, są szczególnie wygodne podczas przenoszenia, kopiowania czy usuwania charakterystycznych grup plików.



Jeżeli użytkownik korzysta z wyrażen wieloznacznych w innych systemach operacyjnych, należy zachować ostrożność, ponieważ mogą one działać w inny sposób. Przykładowo zapis `ren *.txt *.foo` może spowodować zmianę nazwy grupy plików z rozszerzeniem `.txt` na pliki o tych samych nazwach, ale z rozszerzeniem `.foo`. To nie stanie się w Linuksie, ponieważ system wyrażen wieloznacznych zostanie uruchomiony przed wykonaniem polecenia, a polecenie `*.foo` nie znaczy nic, ponieważ nie istnieją pliki z takim rozszerzeniem.

Potoki i przekierowanie danych

Zgodnie z informacjami zamieszczonymi w rozdziale 1., podstawowym elementem filozofii Uniksa, a później też i Linuksa, jest założenie, że dane narzędzie ma wykonywać jedno zadanie, ale robić to dobrze. Dzięki temu, że dane uzyskane z jednego programu są przekazywane do następnego, można uniknąć niepotrzebnego dublowania kodu. Przykładowo

użytkownik chciałby wyświetlić wszystkie pliki w danym katalogu, a następnie je policzyć. Najprostszym sposobem jest zapisanie wyniku działania polecenia `ls` do jakiegoś pliku tymczasowego, a następnie zliczenie plików za pomocą polecenia `wc -l`. Linux ułatwia to zapisanie danych poprzez mechanizm, który nazywa się **przekierowaniem danych** i jest reprezentowany przez graficzny symbol `>`. Oto przykład polecenia, które zapisze dane z komendy `ls` do pliku tymczasowego `/tmp/lista_plikow.txt`:

```
$ ls > /tmp/lista_plikow.txt
```

Teraz będzie można użyć polecenia `wc -l`, które zliczy ilość wierszy:

```
$ wc -l /tmp/lista_plikow.txt
8
```

Najwyraźniej w katalogu bieżącym znajduje się osiem plików. Jednakże korzystanie z pliku tymczasowego wymaga dodatkowego wpisywania poleceń, a później również posprzątania po sobie. Na szczęście, Linux ponownie przychodzi z pomocą, oferując specjalny znak, poziomą kreskę, która przekształca dane wyjściowe z jednego programu w dane wejściowe kolejnego programu. Ten symbol znany jest pod nazwą **potoku** (ang. *pipe*) ma następujący kształt: `|`. Potoków można używać w celu wyeliminowania plików tymczasowych, tak jak w poniższym przykładzie:

```
$ ls | wc -l
8
```

Jeżeli użytkownik wykonuje dużo działań w linii poleceń, szybko stwierdzi, że przekierowanie danych i potoki to bardzo wygodne rozwiązania. Przekierowanie danych wyjściowych to prosty sposób na śledzenie zawartości plików i katalogów, zapisywania danych o systemie itp.

Jeszcze bardziej interesującą funkcją oferowaną przez przekierowanie danych jest możliwość dodawania danych do istniejących plików. Użyty w poprzednim przykładzie matematyczny symbol „jest większe niż” spowoduje utworzenie określonego pliku, jeśli ten nie istnieje, ale też nadpisanie zawartości pliku już istniejącego. Ale być może użytkownik wolałby, żeby zamiast zastępować zawartość danego pliku, polecenie dodało do niego nowe wpisy. Można to łatwo osiągnąć, stosując zamiast jednego znaku większości dwa, jak w poniższym przykładzie:

```
$ ls > /tmp/lista_plikow.txt
$ ls >> /tmp/lista_plikow.txt
$ wc -l /tmp/lista_plikow.txt
16
```

Jak widać, plik tymczasowy zawiera teraz dwa razy więcej linii, co wynika z faktu, że wykonano dwukrotnie tę samą operację.

Powłoka `bash` umożliwia również przekierowanie danych wejściowych, dzięki czemu można odczytać dane z jednego pliku i użyć ich jako danych wejściowych do kolejnego polecenia. Pełny zakres informacji na temat przekierowania danych można znaleźć w elektronicznej pomocy dostępnej dla powłoki `bash`. Użytkownik dowie się, że korzysta z przekierowania danych i potoków znacznie częściej, niż mu się wydawało...

Wprowadzenie do kontroli zadań

Wszystkie opisywane w niniejszym rozdziale polecenia musiały zostać wpisane do powłoki, a następnie należało poczekać, aż zakończy się ich działanie. Jest to dobry sposób dla poleceń, które działają szybko, ale co zrobić z tymi, które działają przez długi czas? Powłoka oferuje również dobre rozwiązanie tego problemu poprzez możliwość uruchamiania poleceń, które następnie będą działały w tle. Działa to w ten sposób, że po wpisaniu polecenia i określeniu, że ma ono działać w tle, dany proces będzie działał nawet wówczas, kiedy znak zachęty zostanie ponownie wyświetlony w bieżącym oknie powłoki. Załóżmy np., że użytkownik chciałby wykorzystać edytor emacs do edycji pliku *mojplik.txt*, ale też chciałby mieć możliwość korzystania z okna terminala w GNOME czy xterm, podczas gdy emacs działałby w swoim własnym oknie. Aby uruchomić dane polecenie tak, aby działało w tle, należy użyć znaku `&`, tak jak w poniższym przykładzie:

```
$ emacs mojplik.txt &  
[1] 7539
```

Takie uruchomienie polecenia określane jest nazwą zadania działającego w tle i powoduje wyświetlenie przez powłokę dwóch informacji:

- ♦ numeru zadania — w nawiasach kwadratowych,
- ♦ numeru identyfikacyjnego procesu, który został uruchomiony do wykonania danego polecenia.

Numer zadania jest przydatny, ponieważ możliwe jest uruchomienie wielu zadań działających jednocześnie w tle. Ponadto numer zadania dostarcza unikalnej metody identyfikowania poszczególnych poleceń działających w tle. Numer zadania to identyfikator zadań działających w tle w kontekście bieżącej powłoki. Natomiast numer procesu to unikalny numer identyfikujący dany proces w kontekście działania całego systemu.

Dlaczego w ogóle zajmować się identyfikowaniem procesów? Przede wszystkim dlatego, że dzięki temu można przerwać dany proces, jeżeli coś pójdzie źle. Załóżmy, że użytkownik przypadkowo uruchomił w tle polecenie, które teraz wymaga podania danych wejściowych. Aby dane zadanie przenieść z działania w tle na pierwszy plan (żeby pojawiło się w powłoce), należy po prostu podać jego numer poprzedzony znakiem `%`, tak jak w poniższym przykładzie:

```
$ %1
```

To spowoduje przywrócenie emacsa do działania na pierwszym planie, wtedy wszystko, co zostanie wpisane w powłoce, będzie przesłane do programu. Załóżmy teraz, że użytkownik utracił okno emacsa z pulpitu i chciałby samodzielnie przerwać działanie tego zadania w tle. Można to zrobić, używając polecenia `kill`, dla którego należy podać również numer identyfikacyjny danego zadania, tak jak w poniższym przykładzie:

```
$ kill %1
```

Znajomość nie tylko numeru zadania dla działającego w tle procesu, ale również numeru procesu w systemie umożliwia użycie polecenia `kill` do przerywania procesu z innej powłoki. Ponieważ numery zadań są unikalne dla danej powłoki, a numery procesów dla

całego systemu, polecenia `kill` wraz z numerem zadania można użyć do przerwania procesu z dowolnego miejsca, tak jak w poniższym przykładzie:

```
$ kill 7539
```

Możliwość wykonywania w tle długo działających poleceń jest szczególnie ważna w wielookienkowych środowiskach graficznych, ponieważ użytkownik zwykle chce mieć możliwość korzystania z aktualnej powłoki, podczas gdy dane zadania powinny działać samodzielnie.

Jak zawsze, kontrola zadań to bardzo potężne i elastyczne narzędzie, które zostało tu opisane bardzo powierzchownie. Aby uzyskać kompletne informacje na ten temat, należy użyć polecenia `info`, które udostępni system pomocy elektronicznej.

Eksploracja pliku konfiguracyjnego powłoki

Podczas uruchamiania powłoki w Ubuntu system odczytuje dane konfiguracyjne z różnych plików tekstowych znajdujących się w systemie. Wymieniłem je poniżej w porządku, w jakim wczytuje je system operacyjny:

- ♦ `/etc/profile` (tylko powłoka logowania),
- ♦ `/etc/bash.bashrc` (wszystkie powłoki bash),
- ♦ `~/.bash_profile` (tylko powłoki logowania),
- ♦ `~/.bashrc` (wszystkie powłoki).



Tylda `~` oznacza skrót do katalogu domowego użytkownika. Dlatego zapis `~/.bashrc` to plik znajdujący się w katalogu domowym aktualnego użytkownika, a zapis `~juser/.bashrc` to odwołanie do pliku znajdującego się w katalogu domowym użytkownika o nazwie `juser`.

Pierwsze dwa wymienione pliki to pliki startowe powłoki, które są odczytywane, kiedy dowolny użytkownik loguje się do systemu (`/etc/profile`) lub uruchamia powłokę (`/etc/bash.bashrc`). Kolejne dwa to osobiste pliki konfiguracyjne przechowywane w katalogu domowym danego użytkownika. Podobnie jak w plikach konfiguracyjnych systemu, plik `~/.bash_profile` jest odczytywany przez powłokę logowania użytkownika, podczas gdy plik `~/.bashrc` odczytywany jest przez powłoki uruchamiane przez użytkownika, takie jak terminal czy xterm. Z tych dwóch plików ważniejszy jest `.bashrc`, ponieważ:

- ♦ `~/.bash_profile` to domyślny plik odczytywany w systemach Ubuntu nawet w kontekście powłoki logowania,
- ♦ uruchomienie powłoki to proces znacznie bardziej skomplikowany, niż logowanie się. Użytkownik loguje się tylko raz, choć może otworzyć wiele różnych okien terminala w GNOME czy xterm.

Plik `.bashrc` umożliwia każdemu użytkownikowi dostosowywanie wielu aspektów działania powłoki, takich jak lokalizacja poleceń, wyświetlany znak zachęty, liczba zapamiętywanych poleceń i mnóstwo innych. Plik ten umożliwia też tworzenie własnych skrótów do określonych poleceń (jeżeli tylko są one zdania, że te, z których korzystają, mają zbyt wiele znaków), a także kreowanie własnych poleceń. Linie w każdym pliku konfiguracyjnym

powłoki (a także w plikach skryptów) rozpoczynające się od znaku # są rozpoznawane jako komentarze i nie są przetwarzane przez interpreter.



Wszystkie powłoki linuksowe oferują język poleceń, konstruowanie pętli i obsługę wyrażeń wieloznacznych, co ma na celu ułatwienie tworzenia skryptów. Wyjaśnienie zagadnień dotyczących programowania w powłoce wykracza poza tematykę niniejszej książki. Dwoma źródłami informacji na ten temat, polecanyymi przeze mnie, są strony internetowe Linux Shell Scripting Tutorial (www.freeos.com/guides/lsst/) oraz strony projektu Linux Documentation; Advanced Shell Scripting Guide (www.tldp.org/LDP/abs/html).

Podczas tworzenia konta użytkownika w Ubuntu powstają standardowe pliki *.bashrc* i *.bash_profile*, a dokładniej, z katalogu */etc/skel* kopiowane są ich szablony zapisane pod tymi samymi nazwami. Poniższy fragment to początkowy fragment pliku *.bashrc* używanego dla wszystkich kont użytkowników w Ubuntu:

```
# ~/.bashrc: executed by bash(1) for non-login shells.
#
# If not running interactively, don't do anything
[ -z "$PS1" ] && return
[dalsza część usunięta]
```

Ostatni fragment powyższego przykładu zatrzymuje odczytywanie przez powłokę domyślnego pliku *.bashrc*, jeżeli nie jest on uruchomiony w trybie interaktywnym. Dzieje się to dzięki użyciu polecenia `[` do sprawdzenia wartości zmiennej środowiskowej *PS1*, która określa znak zachęty użytkownika oraz powrót do powłoki, jeżeli ta zmienna nie jest ustawiona.

Cóż to jest zmienna środowiskowa? Dobrze, że padło to pytanie.

Zmienne środowiskowe

Zmienne środowiskowe to zmienne definiowane w kontekście powłoki, które muszą być odczytywane i używane przez aplikacje w systemie, włącznie z samą powłoką. Jedną z najczęściej używanych zmiennych środowiskowych jest zmienna *PATH*, zawierająca listę oddzielonych dwukropkami katalogów, w których powłoka wyszukuje (w określonym porządku) polecenia. Zmienne środowiskowe można sprawdzić, korzystając z jednego z trzech mechanizmów:

- ♦ z polecenia *set*, które wyświetli wszystkie aktualnie zdefiniowane zmienne powłoki,
- ♦ z polecenia *printenv*, które wyświetli listę wybranych (lub wszystkich) aktualnie zdefiniowanych zmiennych,
- ♦ z polecenia *echo*, które wyświetli wartości określonej zmiennej środowiskowej.

Ponieważ polecenia *set* i *printenv* produkują duże ilości danych wyjściowych, najczęściej używanym poleceniem służącym do wyświetlania wartości danej zmiennej środowiskowej jest polecenie *echo*, tak jak w poniższym przykładzie:

```
$ echo $PATH
/home/wvh/bin:/usr/local/Adobe/Acrobat7.0/bin:/usr/local/gcc_svn/bin:\
/home/wvh/BitTorrent/src/BitTorrent:/home/wvh/cxoffice/bin:\
/usr/lib/mit/bin:/usr/lib/mit/sbin:/usr/NX/bin:/usr/local/sbin:\
/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/bin/X11:/usr/games:
```



Wartość zmiennej środowiskowej PATH to pojedynczy ciąg zawierający oddzielone dwukropkiem nazwy katalogów. W powyższym przykładzie linie zostały rozdzielone, aby ułatwić ich formatowanie.

Podczas używania polecenia `echo` do wyświetlania wartości danej zmiennej należy jej nazwę poprzedzić znakiem `$`. Dzięki temu powłoka dostanie sygnał, że ma wyświetlić zawartość zmiennej, a nie wykonać powtórkę ciągu, jak stałoby się, gdyby znak dolara nie został wpisany:

```
$ echo PATH
PATH
```

Podana powyżej przykładowa wartość zmiennej PATH została wcześniej zmieniona. Standardowe ustawienia tej zmiennej w Ubuntu są następujące:

```
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:\
/usr/bin/X11:/usr/games
```

Aby dodać do tej listy nowy katalog, w którym również ma być prowadzone przeszukiwanie, należy w linii poleceń wykonać następującą komendę:

```
$ export PATH=/usr/wvh/bin:$PATH
```

To polecenie ustawia zmienną środowiskową i eksportuje wartość do środowiska powłoki. W tym przykładzie dodano katalog z plikami binarnymi użytkownika jako pierwszy element listy katalogów, w których powłoka będzie szukać poleceń do wykonania. Aby wprowadzić tę zmianę na stałe, należy dodać tę linię na końcu pliku `~/.bashrc` danego użytkownika.

Polecenie `echo` jest użyteczne, kiedy trzeba wyświetlić wartość pojedynczej zmiennej środowiskowej. Do tego samego celu można też użyć polecenia `printenv`, jak w poniższym przykładzie:

```
$ printenv PATH
/home/wvh/bin:/usr/local/Adobe/Acrobat7.0/bin:/usr/local/gcc_svn/bin:\
/home/wvh/BitTorrent/src/BitTorrent:/home/wvh/cxoffice/bin:\
/usr/lib/mit/bin:/usr/lib/mit/sbin:/usr/NX/bin:/usr/local/sbin:\
/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/bin/X11:/usr/games:
```

Ponieważ polecenie `printenv` jest zaprojektowane do wyświetlania zmiennych środowiskowych, używając go, nie trzeba poprzedzać nazwy danej zmiennej znakiem dolara.

Polecenia `printenv` i `set` są częściej używane do wyświetlania wszystkich aktualnie zdefiniowanych zmiennych środowiskowych. Następnie można wyszukać odpowiedni ciąg, kierując za pomocą potoku dane wyjściowe do polecenia `grep`, tak jak w poniższym przykładzie:

```
$ printenv | grep HIST
HISTCONTROL=ignoreboth
HISTFILE=/home/wvh/.bash_history
HISTFILESIZE=1000
HISTSIZE=1000
```

To przykładowe polecenie wyświetliło nazwy i wartości wszystkich zmiennych środowiskowych zawierających ciąg „HIST”, który jest zmienną powiązaną z mechanizmem zapisywania historii działań w powłoce (co zostało omówione wcześniej).

Definiowanie i używanie aliasów

Kolejną interesującą funkcją dostępną w powłoce jest możliwość definiowania aliasów, czyli nazw zastępczych dla różnych poleceń. Alias to ciąg, który jest rozpoznawany i rozwijany do postaci określonego polecenia. Niektóre z najpowszechniejszych aliasów używanych w systemach linuksowych dotyczą poleceń `cp` i `rm` i powodują ich wykonanie wraz z dodatkowymi opcjami, a są stosowane w celu zapobieżenia przypadkowemu nadpisaniu lub usunięciu plików, co może się przydarzyć podczas korzystania z nich bez opcji potwierdzania działania. Aby definiować aliasy za pomocą powłoki, należy użyć następującego polecenia (można też dokonać edycji pliku `~/.bashrc`):

```
$ alias cp="cp -i"
$ alias rm="rm -i"
```

Jeżeli po zdefiniowaniu tych dwóch aliasów użytkownik wprowadzi polecenie `cp` lub `rm`, powłoka automatycznie zastąpi je poleceniami, które zostały wpisane powyżej.

Aby uprościć definiowanie aliasów (i zminimalizować ilość zmian wprowadzanych w pliku `~/.bashrc`), w pliku `~/.bashrc` domyślnie używanym w Ubuntu zamieszczono wpis, który w ustawieniach domyślnych jest oznaczony znakiem komentarza:

```
#if [ -f ~/.bash_aliases ]; then
#    . ~/.bash_aliases
#fi
```

Po usunięciu znaku komentarza z początku linii użytkownik będzie mógł definiować wszystkie własne aliasy w pliku `~/.bash_aliases`, który zlokalizowany będzie w jego katalogu domowym.

Podsumowanie

W niniejszym rozdziale opisałem wykorzystanie linii poleceń w Linuksie, wyjaśniłem terminologię używaną w kontekście interpreterów linii poleceń oraz omówiłem podstawowe możliwości oferowane przez powłokę, czyli domyślny interpreter używany w Ubuntu. Napisałem też, jak uzyskać dostęp do powłoki, a także podałem kilka powszechnie używanych poleceń. Rozdział zakończyłem wyjaśnieniami na temat plików konfiguracyjnych wykorzystywanych przez powłokę oraz ich zawartości.

Naturalną konsekwencją korzystania z linii poleceń jest edycja plików tekstowych, którą omówię w rozdziale 7. Choć obecnie wiele edytorów tekstu przechowuje własne pliki w specjalnych, binarnych formatach, to jednak pliki tekstowe są powszechne w internecie i w Linuksie. Dokumenty tworzone z wykorzystaniem języków znaczników, takich jak HTML, SGML czy XML, to pliki tekstowe, które można tworzyć i edytować w najprostszym edytorze tekstowym. Większość serwerów linuksowych, włącznie z tymi, które zostały

omówione w niniejszej książce, przechowuje swoje ustawienia konfiguracyjne w plikach tekstowych. Z tych i innych powodów umiejętność korzystania z edytorów tekstu jest jedną z podstawowych podczas korzystania z Linuksa.

Rozdział 7.

Praca z plikami tekstowymi w Ubuntu

W tym rozdziale:

- ♦ Korzystanie z popularnych edytorów uniksowych
- ♦ Praca z edytorem vi
- ♦ Używanie edytora emacs
- ♦ Wykorzystanie edytora graficznego gedit
- ♦ Przegląd pozostałych edytorów dostępnych dla systemu Ubuntu

Większość użytkowników przywykła do pracy z plikami o określonych formatach w różnych aplikacjach, które często identyfikowane są przez specjalne rozszerzenie lub wyświetlaną ikonę. Znamy pliki w formacie *.doc*, które powstają w graficznych edytorach tekstu, takich jak Microsoft Word, pliki PDF tworzone i wyświetlane w programie Adobe Reader i innych czytnikach, pliki *.ppt* kreowane za pomocą programu Microsoft PowerPoint itd. We wszystkich tych plikach znajdują się zakodowane, często binarne informacje, a ich format decyduje o tym, za pomocą której aplikacji można je uruchamiać, co jednocześnie ogranicza możliwości wykorzystania innych programów do ich obróbki.

Znamy też pliki, które można uruchamiać niezależnie od aplikacji, czyli tekstowe lub — bardziej precyzyjnie — pliki w formacie ASCII. Zawartość takich plików stanowią standardowe symbole, cyfry i litery — wszystkie znaki, które można znaleźć na klawiaturze komputera. Zaletą takich plików jest to, że są łatwe do odczytania czy wykonania; generalnie praca z nimi jest prosta.

Niezależnie od tego, do czego użytkownik wykorzystuje swoje Ubuntu, niemal pewne jest, że prędzej czy później będzie musiał edytować pliki tekstowe. Jeżeli pisze kody źródłowe, niemal pewne jest, że korzysta z formatu ASCII, ponieważ oferuje on najmniejszy wspólny mianownik, który można wykorzystać w każdym kompilatorze. Jeżeli natomiast użytkownik jest pisarzem, być może będzie chciał korzystać z tego formatu, ponieważ można go używać w każdym systemie operacyjnym i w każdym edytorze tekstów — tak samo łatwo pracuje się na tych plikach w Ubuntu, jak i w systemie Mac OS X.

Co jeszcze istotniejsze, większość programów narzędziowych systemu Linux korzysta z jednego lub wielu plików do przechowywania różnego rodzaju danych konfiguracyjnych czy takich, które są wykorzystywane podczas ich uruchamiania. Chociaż większość narzędzi linuksowych można obecnie konfigurować za pomocą interfejsu graficznego, to edytowanie plików tekstowych często okazuje się szybszym i prostszym rozwiązaniem od mozolnego poszukiwania właściwego narzędzia, a następnie odpowiedniej zakładki i opcji w danym programie.

Niniejszy rozdział rozpoczyna się od krótkiego opisu historii uniksowych edytorów plików tekstowych, a także poważnego sporu, dotyczącego „właściwego” podejścia do zagadnienia edycji plików, a który toczy do dziś w kręgach związanych z Linuxem. Później podaję szczegółowe omówienie metod pracy z edytorami vi i emacs, dwoma podstawowymi narzędziami używanymi w Ubuntu. Kilka słów poświęcę też edytorowi ed, prostemu i szybkiemu w działaniu programowi, który jest bardzo przydatny wówczas, gdy trzeba wprowadzić drobne zmiany w plikach. Na końcu omówię też gedit, graficzny edytor tekstowy dostępny w Ubuntu, który jest ogniwem pośrednim pomiędzy narzędziami tekstowymi a programami z interfejsem graficznym.

Wprowadzenie do edytorów tekstowych w Linuksie

W pierwszej wersji Uniksa dla PDP-11, która była dystrybuowana przez firmę Bell Telephone Laboratories, znajdował się edytor tekstu o nazwie ed, napisany przez Kena Thompsona. Wielu użytkowników ulepszało ten program, wśród nich był też George Coulouris z *Queen Mary College* (QMC) w Wielkiej Brytanii; dodał on możliwość interaktywnego edytowania poszczególnych linii, która znalazła się w jego wersji programu o nazwie em (ang. *editor for mortals* — edytor dla śmiertelników). Program, wraz ze swoim autorem, zawędrował w roku 1976 na uniwersytet w Berkeley. W tym samym czasie Ken Thompson znalazł się w Kalifornii na tej samej uczelni i podjął współpracę z Billem Joyem oraz Chuckiem Halem nad ulepszaniem Uniksa, której efektem było powstanie BSD (ang. *Berkeley Standard Distribution*).

Haley i Joy pracowali nad edytorem em; stworzyli własną wersję o nazwie en, a następnie przeszli przez kolejne litery alfabetu, aż dotarli do wersji ex, która wciąż była ukie-
runkowana na wiersze. W roku 1976 Joy stworzył edytor vi (nazwa zaczerpnięta ze słowa *visual*), który miał wspólny z edytorem ex zestaw komend, a jego powstanie wiązało się bezpośrednio z przekazaniem uniwersytetowi w Berkeley (UCB) większej liczby terminali ADM-3A. Elastyczność programu dostępna dzięki możliwości korzystania z całego ekranu (efekt wykorzystania pliku *etc/termcap* — *terminal capabilities*, czyli możliwości terminala) była ważnym krokiem, a edytowanie plików w Uniksie nie miało już nigdy powrócić do dawnej formy.

Mniej więcej w tym samym czasie w zupełnie innej części Stanów Zjednoczonych hakerzy Lipa i ITS/TOPS-20/TWENWX zatrudnieni w laboratorium sztucznej inteligencji MIT pracowali nad ulepszeniem edytora o nazwie TECO (ang. *Text Editor and Corrector*) stosowanego w systemach PDP-10. Zainspirowani graficzną edycją tekstu prowadzoną w laboratoriach w Stanford, Richard Stallman (późniejsza sława GNU i FSF) oraz Guy

Steel zaprojektowali wersję TECO, do której użytkownicy mogli dodawać własne polecenia; następnie umieścili zestaw poleceń dla TECO w jednym pakiecie, w którym znalazło się także wsparcie do edycji na ekranie. W pakiecie poleceń dla TECO znalazł się także zestaw makr (czyli sekwencji poleceń wykonywanych jednocześnie) o nazwie *Editing MACroS*, czyli w skrócie EMACS.

Te programy były przeznaczone dla terminali DC PDP-10 i zapewne emacs nie dotarłby tam, gdzie znajduje się w tej chwili, gdyby nie wysiłki Dave'a Conroya, który napisał wersję emacsa o nazwie MicroEmacs dla wszystkich innych urządzeń, oraz Jamesa Goslinga (znanego z późniejszych osiągnięć w dziedzinie Javy), który z kolei w roku 1981 napisał wersję dla Uniksa. Inaczej niż makra programu EMACS, które były inspiracją, był to skompilowany program, który można było uruchamiać w dowolnym systemie jako skompilowane binaria. Wersja programu napisana przez Goslinga i sprytnie nazwana „emacs Goslinga”, czy prościej gosmacs, posłużyła jako prototyp dla wersji o otwartych źródłach (GNU emacs) stworzonej przez Richarda Stallmana, którą zaczął ją pisać dla systemów Unix w roku 1984. Ze względów prawnych Stallman zastąpił cały kod autorstwa Goslinga, podobnie jak język MockLisp, dojrzalszym językiem Lisp z własnym interpreterem, aby parsować pliki konfiguracyjne i inne dane startowe.



Z biegiem lat pojawiło się kilka alternatywnych propozycji odczytywania nazwy emacs, oto kilka ciekawszych propozycji: „Escape, Meta, Alt, Control, Shift” odwołujący się do roli klawiszy *Escape* i *Control* w zestawie poleceń programu; czy „Eight Megabytes And Constantly Swapping”, czyli w wolnym tłumaczeniu: osiem megabajtów i stała wymiana, co brzmiało nieco zabawnie w czasach, kiedy 8 MB to była spora ilość pamięci w przypadku komputera i odnosiło się do sporych wymagań sprzętowych (jak na owe czasy). Dzisiaj 8 MB pamięci ma zapewne każde urządzenie do podnoszenia drzwi garażowych.

Czytelnicy zainteresowani dalszymi szczegółami historii edytowania tekstu w systemach Unix i Linux, mogą znaleźć kolejne informacje na wymienionych niżej stronach.

- ♦ Historia edytora ed: www.english.uga.edu/hc/unixhistory.html (Peter Salus, A Quarter history of UNIX, Reading, MA: Addison-Wesley, 1994).
- ♦ George Colouris i i edytor em: www.dcs.qmul.ac.uk/~george/history/.
- ♦ Wywiad z Billem Joyem: www.cs.pdx.edu/~kirkenda/joy84.html (po raz pierwszy opublikowany w magazynie Unix Review, sierpień 1984).
- ♦ Lemacs/FSFmacs Schism: www.jwz.org/doc/lemacs.

Korzystanie z edytora vi

Ponieważ obecnie kod źródłowy „prawdziwej” wersji vi należy do firmy AT&T, fani tego edytora przepisali program zupełnie od nowa. Aktualnie dostępne są dwa popularne klony (czyli wersje): nvi (nowy vi) oraz vim (vi *improved* — rozbudowany). W systemach linuksowych zwykle używa się wersji vim, z kolei nvi jest często wykorzystywany w systemach z rodziny BSD, takich jak OpenBSD, FreeBSD itp. W tym rozdziale, pisząc o vi, odwołuję się do edytora vim, ponieważ jest kompatybilny z vi, a pakiet, z którego instalowany jest vim, zawiera również alias vi. Poza wszystkim vim to przecież vi, tylko rozbudowany. Jeżeli któraś z uwag będzie odnosiła się tylko i wyłącznie do oryginalnego vi, zostanie to zaznaczone w treści.

Edytory modalne kontra niemodalne

Jak już wcześniej wspominałem, pomiędzy użytkownikami Linuksa i Uniksa nie ma dłużej trwającej debaty niż ta, która dotyczy wyższości edytora vi nad emacssem (lub odwrotnie). Ilość atramentu, jaką przelano w tej dyskusji, mogłaby posłużyć do zapewnienia kilku obszernych tomów. Nowi fani Ubuntu zapewne słyszeli co nieco na ten temat, dlatego przyda im się kilka informacji rzucających więcej światła na to zagadnienie.

Główna różnica w użytkowaniu vi i emacsa sprowadza się do tego, że emacs to edytor niemodalny, co oznacza, iż jakkolwiek standardowy znak wpisywany przez użytkownika jest wprowadzany w każdym momencie edycji, natomiast w przypadku vi ma to miejsce tylko w trybie wstawiania, który jest jednym z dostępnych trybów. Oba te programy są niesłychanie konfigurowalne, chociaż vi nie zwiera implementacji Lispa, dzięki któremu tworzenie własnych funkcji jest prostsze. Ale z drugiej strony, dzięki temu program jest mniejszy i działa szybciej niż emacs. Emacs jest bardziej rozbudowany, ponieważ dodatkowo zawiera również klienta poczty, czytnik grup dyskusyjnych, a nawet kilka gier. Ale być może użytkownik potrzebuje tylko edytora tekstu, a nie środowiska typu „wszystko w jednym”.

Często można spotkać się z opiniami, że zarówno vi, jak i emacs są trudne dla początkujących użytkowników. Z mojego doświadczenia wynika, że zależy to od momentu, w jakim znalazł się ów użytkownik. Szansa, że początkujący użytkownik wpadnie na to, iż aby zapisać plik i zamknąć emacs, należy wcisnąć klawisz *Ctrl*, a następnie użyć kombinacji dwóch innych klawiszy, jest raczej niewielka. Z kolei dopóki w edytorze vi nie uruchomi się trybu wstawiania, jest on mało przydatny, a poszukiwanie sposobu na wyjście z tego trybu i zapisanie pliku przywiodło wielu początkujących użytkowników Linuksa i Uniksa do teź.

Używałem wielu różnych wersji emacsa, począwszy od gosmacsa, niniejsza książka została napisana za pomocą emacsa. Jednak do wykonywania większości zadań administracyjnych stosuje edytor vi zwłaszcza wtedy, jeśli ma to miejsce na cudzych komputerach. Dlaczego? Dlatego, że vi uruchamia się szybciej niż emacs, ale przede wszystkim dlatego, że fani emacsa często modyfikują skróty klawiszowe. Kombinacja klawiszy, która w jednym komputerze wyrównuje akapit, w innym może umieszczać wiadomość na grupie dyskusyjnej.



Chociaż niniejszy rozdział jest znakomitą źródłem informacji o programie vim, sam edytor również jest takim źródłem. Domyślna instalacja pakietu zawiera program o nazwie vimtutor, który jest samouczkiem przeznaczonym dla użytkowników. Jeżeli czytelnik chciałby zapoznać się z przykładami użytkowania programu oraz dostępnymi w nim poleceniami, jest to doskonały punkt startowy.

Jak wcześniej wspominałem, vim to edytor modalny (informacja dla tych, którzy pominęli poprzedni podrozdział: oznacza to, że za pomocą tych samych klawiszy wykonuje się różne operacje, zależy to od trybu, w jakim działa program). Oto podstawowe tryby pracy programu.

- ♦ **Tryb normalny** — jest to tryb uruchamiany podczas pierwszego włączenia programu; można w nim wprowadzać polecenie służące do przemieszczania kursora oraz usuwania tekstu, nie uda się natomiast wprowadzić żadnego tekstu do edytowanego pliku. Jest to pierwsze źródło zamieszania w głowach nowych użytkowników edytora vi, ponieważ większość z nich spodziewa się, że po uruchomieniu programu będą mogli od razu rozpocząć edytowanie tekstu.
- ♦ **Tryb wstawiania** — w tym trybie do tekstu wprowadzane są za pomocą klawiatury standardowe znaki, cyfry i litery; właśnie tego trybu spodziewa się większość użytkowników po włączeniu programu. Drugą przyczyną zamętu

w głowach nowych użytkowników jest fakt, że aby przełączyć się z trybu wstawiania do normalnego, należy wcisnąć klawisz *Esc*.

- ♦ **Tryb linii poleceń** — w tym trybie kursor jest umieszczany w dolnej linii interfejsu programu, co umożliwia użytkownikom wprowadzanie tekstu, który będzie interpretowany jako polecenie dla programu, np. polecenie wyszukania czy wykonania zewnętrznej komendy na fragmencie edytowanego pliku.

Idea programu pracującego w różnych trybach może początkowo wydawać się niepokojąca, ale to właśnie sprawia, że *vi* jest szybkim, wydajnym i potężnym edytorem. Kiedy już użytkownik dowie się, jak przechodzić z jednego trybu w inny oraz popracuje przez chwilę z programem, sam się przekona, jak znakomitym programem jest *vi*. To wszystko, a także fakt, że edytor *vi* można znaleźć w każdym systemie uniksopodobnym powoduje, że warto poświęcić czas na naukę jego obsługi.

Jedną z najfajniejszych opcji programu jest to, że prawie każde polecenie można poprzedzić argumentem składającym się z liczby, określającym część czy zakres, dzięki czemu dane polecenie zostanie wykonane określoną ilość razy, w określonej części pliku lub w określonym zakresie. Poniżej zamieszczam przykłady takiego zastosowania oraz inne wskazówki zebrane na podstawie wieloletniej pracy z programem.



Wszystkie przedstawione w tym rozdziale przykłady działania *vi* są widoczne nie w oknie terminala GNOME, ale *xterm*, ma to podkreślić, że sam program działa w linii poleceń i nie oferuje domyślnie menu. Informacje na temat graficznych wersji edytora *vi* zamieszczono w punkcie „Graficzne wersje edytora *vi*”.

Uruchamianie i wyłączanie edytora *vi*

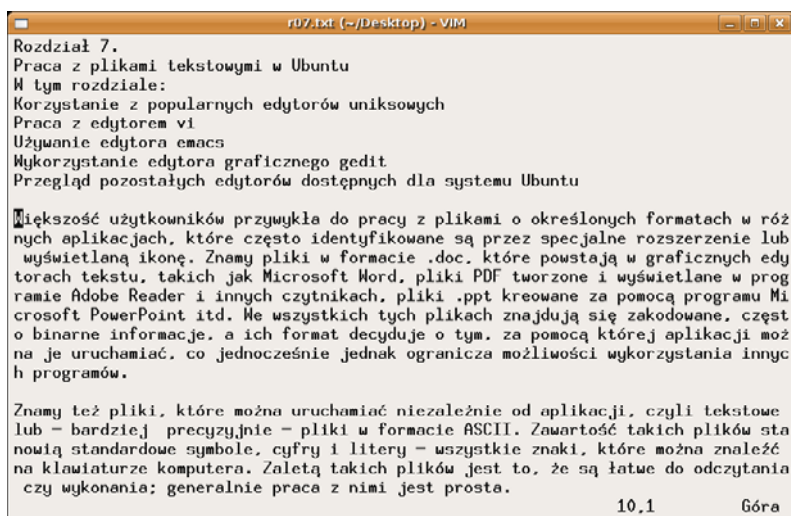
Podstawowymi czynnościami wykonywanymi w większości programów jest ich włączanie i wyłączanie. Jak już wspominałem, pierwsze wyłączenie edytora *vi*, ze względu na działanie w trybach, może wydawać się skomplikowane, ale później stanie się już drugą naturą użytkownika.

Ponieważ *vi* jest narzędziem linii poleceń, uruchamia się go zwykle, wpisując komendę *vi* w terminalu GNOME lub oknie *xterm*. Można też dodać nazwę pliku, który użytkownik chce edytować, zostanie on wówczas automatycznie otwarty w edytorze. Na rysunku 7.1 przedstawiam tekst niniejszego rozdziału edytowany w programie *vi* w oknie *xterm*.

Zakończyć działanie programu i powrócić do linii poleceń można na kilka sposobów, ale najpierw należy przejść do trybu normalnego. Jeżeli podczas edycji pliku użytkownik chce przejść do trybu normalnego, wówczas musi po prostu wcisnąć klawisz *Esc*. Jeżeli w edytowanym pliku zostanie wyświetlona sekwencja *^*, oznacza to, że należy ponownie wcisnąć *Esc*. Teraz można zamknąć program na kilka sposobów.

- ♦ Aby wyłączyć program bez zapisywania wprowadzonych zmian, należy użyć kombinacji *:q!* lub *ZQ*. Polecenie *:q* to standardowe polecenie zakończenia działania programu, co może się zemścić na użytkowniku, jeżeli wprowadził zmiany w pliku, ale ich nie zapisał. Polecenie *:q!* oznacza mniej więcej: „Tak, naprawdę chcę wyłączyć program i wiem, co robię”.

Rysunek 7.1.
xterm i edytor vi



- ♦ Aby zapisać wszelkie wprowadzone w pliku zmiany oraz wyłączyć program, należy użyć polecenia `:x` lub `:wq`. Polecenie `:x` jest charakterystyczne dla programu vim, natomiast `:wq` to standardowa komenda edytora vi, oznaczająca mniej więcej: „Write and then quit”, czyli „Zapisz i zamknij”.
- ♦ Aby określić nazwę pliku, pod jaką ma zostać zapisany, należy użyć polecenia `:w nazwa_pliku`, gdzie *nazwa_pliku* oznacza faktyczną nazwę, pod jaką dany plik ma zostać zapamiętany. Jeżeli plik jest już zapisany pod inną nazwą, wówczas należy użyć polecenia `:w! nazwa_pliku`, które w wolnym tłumaczeniu znaczy: „Tak, naprawdę chcę zapisać ten plik i wiem, co robię”.

Wstawianie tekstu w edytorze vi

Kiedy już użytkownik uruchomi edytor vi, z pewnością będzie chciał napisać jakiś program, list czy inny dokument. Jak wspominałem wcześniej, program jest uruchamiany w trybie normalnym, co oznacza, że można przesuwac kursor oraz wprowadzać polecenia, ale nie można wpisywać tekstu.

Aby zacząć wprowadzać tekst, należy wpisać polecenie `i`. Po uruchomieniu trybu wstawiania każdy wprowadzony znak pojawi się w pliku. Dla przypomnienia: najprostszym sposobem, aby program zakończył tryb wstawiania, jest wciśnięcie klawisza `Esc` znajdujące się na klawiaturze.

Użycie polecenia `i` spowoduje, że tekst będzie wstawiany w aktualnej pozycji kursora, co może być zaskakujące. Jeżeli tekst ma być wprowadzany za kursorem, wówczas należy użyć polecenia `a`. Jeżeli tekst ma być wprowadzany na końcu bieżącej linii, należy użyć polecenia `A`. Komendy `a` i `A` to wyspecjalizowane polecenie służące do wprowadzania tekstu w określonym miejscu. Aby wyłączyć tryb stawiania po użyciu tych poleceń, należy wcisnąć klawisz `Esc`.

Przemieszczanie kursora w edytorze vi

W edytorze vi do przesuwania kursora używa się następujących poleceń:

- ♦ h lub klawisz ze strzałką w lewo — przesunięcie kursora do wstecz o jeden znak,
- ♦ l lub klawisz ze strzałką w prawo — przesunięcie kursora do przodu o jeden znak,
- ♦ j lub klawisz ze strzałką w dół — przesunięcie kursora o jedną linię w dół z zachowaniem (jeżeli to możliwe) kolumny, w której się znajduje,
- ♦ k lub klawisz ze strzałką do góry — przesunięcie kursora o jedną linię w górę z zachowaniem (jeżeli to możliwe) kolumny, w której się znajduje.

Przemieszczanie kursora za pomocą poleceń, takich jak: h, k, j i k, może się wydawać dziwne, ale należy pamiętać, że vi początkowo przeznaczony był dla terminali, w których często nie było klawiszy ze strzałkami. Warto zapytać o to dziadka.



Jeżeli program będzie działał w trybie wstawiania, wówczas użycie tych poleceń spowoduje po prostu wstawienie odpowiednich liter do pliku. Wtedy jednak zawsze można skorzystać z klawiszy ze strzałkami, które będą działały zgodnie z oczekiwaniami.

Wcześniej powiedziałem, że można nakazać programowi, aby przemieścił jednocześnie kursor o określoną ilość pozycji, wystarczy tylko poprzedzić polecenie argumentem liczbowym. I tak, aby np. przesunąć kursor o 5 znaków do przodu, należy wpisać 5l, o pięć linii do góry — 5k itd. To wygodna opcja, ale wymaga uprzedniego obliczenia nowej pozycji kursora. Jeżeli użytkownik chciałby przesunąć kursor do następnego lub poprzedniego słowa, czy przejść do jakiegoś innego określonego miejsca w pliku, potrzebne jest inne rozwiązanie. Aby uprościć takie operacje, vi oferuje kolejny zestaw poleceń służących do przemieszczania kursora:

- ♦ b — przesunięcie wstecz o jedno słowo,
- ♦ w — przesunięcie do przodu o jedno słowo,
- ♦ ^ — przesunięcie na początek wiersza,
- ♦ \$ — przesunięcie na koniec wiersza,
- ♦ (— przesunięcie na początek zdania,
- ♦) — przesunięcie na koniec zdania,
- ♦ { — przesunięcie na początek akapitu,
- ♦ } — przesunięcie na koniec akapitu,
- ♦ 1G — przesunięcie do pierwszej linii w bieżącym pliku,
- ♦ 0G — przesunięcie do ostatniej linii w bieżącym pliku, wcisnięcie G w edytorze vim przyniesie ten sam efekt, ale może nie zadziałać w programie vi.

Ostatnie dwa polecenia są świetnymi przykładami na to, jak może działać standardowa komenda przejścia do linii uzupełniona o argument liczbowy. Można np. przejść do wiersza 14., używając polecenia 14G, a jeżeli użytkownik chciałby umieścić kursor na końcu 3. akapitu, wówczas wystarczy użyć polecenia 3}.

Uzupełniając polecenia służące do przemieszczania kursora po pliku, dodam, że w edytorze vi dostępne są również komendy umożliwiające przesuwanie wyświetlanej zawartości pliku z zachowaniem pozycji kursora (oczywiście, jeżeli to konieczne). Oto te polecenia:

- ♦ *Ctrl+d* (wcisnąć i przytrzymać klawisz *Ctrl*, a następnie wcisnąć dodatkowo klawisz *d*), przesunięcie wyświetlanej zawartości danego pliku o jeden ekran w dół,
- ♦ *Ctrl+e* (wcisnąć i przytrzymać klawisz *Ctrl*, a następnie wcisnąć dodatkowo klawisz *e*), przesunięcie wyświetlanej zawartości danego pliku o jedną linię w dół,
- ♦ *Ctrl+u* (wcisnąć i przytrzymać klawisz *Ctrl*, a następnie wcisnąć dodatkowo klawisz *u*), przesunięcie wyświetlanej zawartości danego pliku o jeden ekran do góry,
- ♦ *Ctrl+y* (wcisnąć i przytrzymać klawisz *Ctrl*, a następnie wcisnąć dodatkowo klawisz *y*), przesunięcie wyświetlanej zawartości danego pliku o jedną linię do góry.

Polecenia *Ctrl+d* i *Ctrl+u* wyświetlą nową zawartość, ale pozycja kursora na ekranie zostanie zachowana nawet wtedy, kiedy zmieni się tekst. Wykonanie poleceń *Ctrl+e* i *Ctrl+y* spowoduje zachowanie pozycji kursora w tekście, dopóki dana linia nie zniknie z ekranu.

Usuwanie i zmienianie tekstu w edytorze vi

Podobnie jak w większości innych operacji, również w edytorze vi można usuwać tekst za pomocą kilku różnych poleceń. Aby usunąć znak, w którym aktualnie znajduje się kursor, należy użyć polecenia *x*. Można, oczywiście, usunąć jednocześnie więcej niż jeden znak, wystarczy tylko poprzedzić polecenie parametrem liczbowym; i tak, aby usunąć 14 znaków, począwszy od bieżącej pozycji kursora, wystarczy użyć polecenia *14x*.

Chociaż polecenie *x* jest przydatne w sytuacji, kiedy trzeba usunąć kilka znaków, to jednak większość użytkowników myśli raczej w kategoriach: „usunąć całe słowo”, „usunąć wszystko do końca bieżącej wiersza” czy „usunąć wszystko do końca bieżącego akapitu”. Aby wykonać takie działanie, należy użyć polecenia *d* poprzedzonego znakiem wskazującym, gdzie komenda ma zakończyć swoje działanie. Oto przykłady: polecenie *dw* spowoduje usunięcie reszty słowa, począwszy od bieżącej pozycji kursora; polecenie *d\$* spowoduje usunięcie reszty tekstu, począwszy od bieżącej pozycji kursora, aż do końca bieżącego wiersza, polecenie *d}* spowoduje usunięcie reszty tekstu, począwszy od bieżącej pozycji kursora, aż do końca bieżącego akapitu. Jak zapewne czytelnik zauważył, drugi znak każdego polecenia to opisane wcześniej komendy służące do przemieszczania kursora.

Podczas edytowania tekstu zupełnie zwyczajną czynnością jest usuwanie go i zastępowanie nowym. Można to wykonać, używając połączenia polecenia służącego do usuwania tekstu i komendy do wstawiania tekstu. W takim przypadku należy użyć polecenia *c*, dzięki któremu można jednocześnie usunąć teks i wprowadzić na jego miejsce nowy. I tak, aby zastąpić tekst, począwszy od bieżącej pozycji kursora, aż do końca słowa, należy użyć polecenia *cw*; aby zastąpić tekst, począwszy od bieżącej pozycji kursora, aż do końca wiersza, należy użyć polecenia *c\$*; aby zastąpić tekst, począwszy od bieżącej pozycji kursora, aż do końca akapitu, należy użyć polecenia *c}* itd. Po zakończeniu wprowadzania tekstu należy przejść do trybu normalnego, używając do tego klawisza *Esc*.

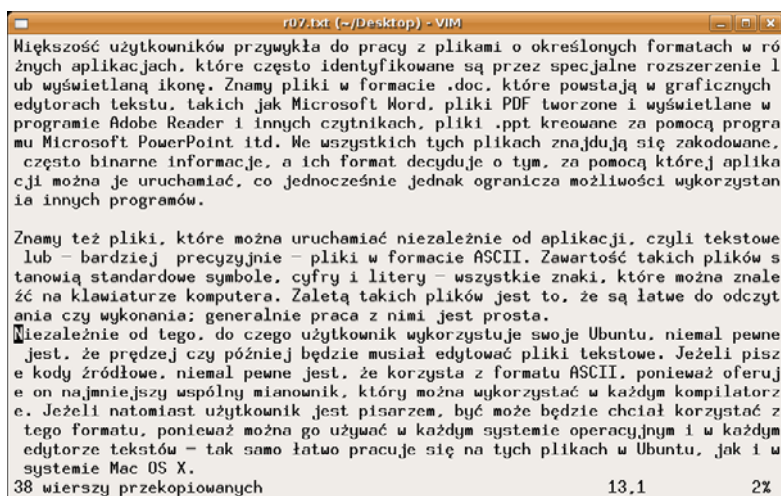
Wycinanie, kopiowanie i wstawianie tekstu w edytorze vi

W graficznych edytorach tekstu bardzo łatwo wycinać i wklejać tekst, a to za sprawą udostępnienia tych funkcji dla myszy komputerowej, a także specjalnego zestawu kombinacji klawiszy. Można to samo wykonać również w programie vi, ale wtedy wymagana jest nieco większa wiedza na temat samego edytora.

Wycinanie tekstu jest realizowane za pomocą polecenia służącego do jego usuwania (omówionego nieco wcześniej), które z kolei umieszcza tekst w przestrzeni zwanej buforem, z której można go na powrót umieścić w pliku. Program dostarcza też specjalnego polecenia y, za którego pomocą można skopiować określony fragment tekstu do bufora, a następnie umieścić go w dowolnym miejscu. Podobnie jak w przypadku polecenia służącego do usuwania, również tu można ograniczyć obszar, z którego ma być kopiowany tekst. I tak za pomocą polecenia y\$ zostanie skopiowany do bufora tekst, począwszy od bieżącej pozycji kursora, aż do końca wiersza, polecenie y} spowoduje kopiowanie do bufora tekstu, począwszy od bieżącej pozycji kursora, aż do końca akapitu. Jeżeli zostanie skopiowany więcej niż jeden wiersz, wówczas ich liczba zostanie wyświetlona na dole okna programu vi. Na rysunku 7.2 przedstawiam program vi działający w oknie xterm po skopiowaniu bieżącego akapitu do bufora.

Rysunek 7.2.

*Status kopiowania
w edytorze vi*



Oczywiście, samo wycinanie czy kopiowanie tekstu nie byłoby tak interesujące, gdyby nie możliwość wklejenia tekstu w określonym miejscu pliku; do tego celu służą polecenia p oraz P. Aby wkleić pojedynczy wiersz z bufora w miejscu, w którym znajduje się kursor, należy użyć polecenia p. Jeżeli jednak do bufora zostało skopiowanych wiele wierszy, wówczas należy polecenie poprzedzić liczbą wskazującą numer wiersza, który ma być wklejony. Jeżeli jednak użytkownik nie pamięta, który wiersz powinien zostać wklejony lub też chciałby je wkleić wszystkie, wówczas należy po prostu użyć polecenia P.

Wyszukiwanie i zastępowanie tekstu w edytorze vi

W edytorze vi dostępne są osobne polecenia do wyszukiwania treści od początku, jak i od końca pliku. Użycie polecenia `/`, po którym następuje ciąg znaków, jakie mają być wyszukane, i wciśnięcie klawisza *Enter* spowoduje, że program przeszuka plik od bieżącej pozycji kursora do końca, aż do odnalezienia (lub nie) danego tekstu. Na rysunku 7.3 przedstawiam program vi tuż przed przystąpieniem do wyszukiwania tekstu. Jeżeli dany ciąg zostanie znaleziony, wówczas kursor zostanie ustawiony przed nim. Jeżeli tekstu nie da się znaleźć w pliku, zostanie wyświetlony stosowny komunikat.

Rysunek 7.3.

Wyszukiwanie
ciągów tekstowych
w edytorze vi

```

r07.txt (~/.Desktop) - VIM
Rozdział 7.
Praca z plikami tekstowymi w Ubuntu
W tym rozdziale:
Korzystanie z popularnych edytorów uniksowych
Praca z edytorem vi
Używanie edytora emacs
Wykorzystanie edytora graficznego gedit
Przegląd pozostałych edytorów dostępnych dla systemu Ubuntu

Większość użytkowników przywykła do pracy z plikami o określonych formatach w róż-
nych aplikacjach, które często identyfikowane są przez specjalne rozszerzenie lub
wyświetlaną ikonę. Znamy pliki w formacie .doc, które powstają w graficznych edy-
torach tekstu, takich jak Microsoft Word, pliki PDF tworzone i wyświetlane w prog-
ramie Adobe Reader i innych czytnikach, pliki .ppt kreowane za pomocą programu Mi-
crosoft PowerPoint itd. We wszystkich tych plikach znajdują się zakodowane, częst-
o binarne informacje, a ich format decyduje o tym, za pomocą której aplikacji moż-
na je uruchamiać, co jednocześnie jednak ogranicza możliwości wykorzystania innych
programów.

Znamy też pliki, które można uruchamiać niezależnie od aplikacji, czyli tekstowe
lub – bardziej precyzyjnie – pliki w formacie ASCII. Zawartość takich plików sta-
nowią standardowe symbole, cyfry i litery – wszystkie znaki, które można znaleźć
na klawiaturze komputera. Zaletą takich plików jest to, że są łatwe do odczytania
czy wykonania; generalnie praca z nimi jest prosta.
/melon

```

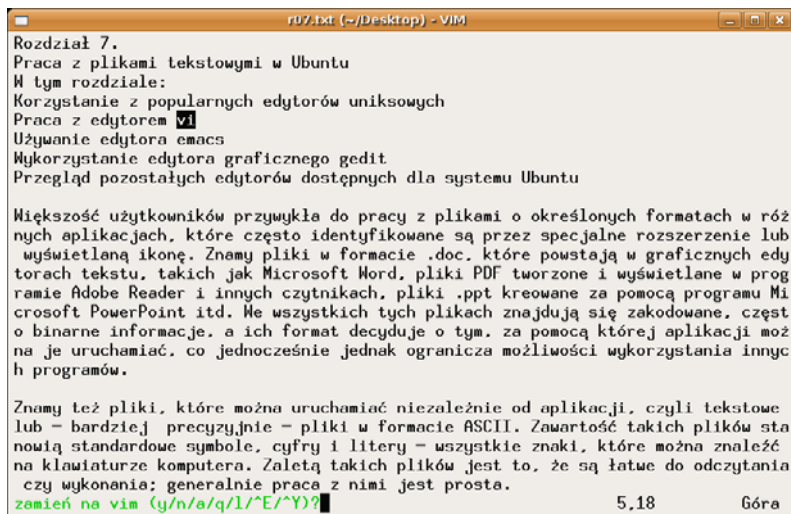
Polecenie `?`, odwrotnie niż polecenie `/`, służy do wyszukiwania zadanego ciągu wstecz, do początku pliku. Obie te komendy to przykłady korzystania z linii poleceń programu vi, wprowadzane są na dole okna programu.

Tryb linii poleceń edytora vi udostępnia również wygodne polecenie służące do wyszukiwania danego tekstu i jednoczesnego zastępowania go innym, jest to polecenie `:%s/tekst1/tekst2/`, gdzie *tekst1* to ciąg, który ma być znaleziony, a *tekst2* to ten, który zostanie wstawiony w miejsce pierwszego. Po ostatnim znaku `/` można umieścić kolejne opcje; `g` oznacza, że zmiany zostaną wprowadzone w całym pliku (całe polecenie ma wówczas postać `:%s/tekst1/tekst2/g`), a `gc` spowoduje, że program będzie prosił użytkownika o potwierdzenie każdej zmiany (polecenie `:%s/tekst1/tekst2/gc`). Na rysunku 7.4 przedstawiam warunkowe zastąpienie ciągu *vi* ciągiem *vim*. Operacje takie często określa się nazwą „wyszukaj i zastąp” (ang. *query and replace*).

Warto zwrócić uwagę, że na dole ekranu widoczne jest żądanie potwierdzenia operacji przez użytkownika (Y: *yes* — tak lub N: *no* — nie), ale też dostępne są dwie komendy służące do przemieszczania kursora: `Ctrl+e` (przewinięcie o jeden wiersz do góry) i `Ctrl+y` (przewinięcie o jeden wiersz na dół), dzięki którym łatwiej zobaczyć, gdzie będzie wstawiony tekst.

Rysunek 7.4.

Operacja wyszukaj
i zastąp w edytorze vi



```
Rozdział 7.  
Praca z plikami tekstowymi w Ubuntu  
W tym rozdziale:  
Korzystanie z popularnych edytorów uniksowych  
Praca z edytorem vi  
Używanie edytora emacs  
Wykorzystanie edytora graficznego gedit  
Przegląd pozostałych edytorów dostępnych dla systemu Ubuntu  
  
Większość użytkowników przywykła do pracy z plikami o określonych formatach w róż-  
nych aplikacjach, które często identyfikowane są przez specjalne rozszerzenie lub  
wyświetlaną ikonę. Znamy pliki w formacie .doc, które powstają w graficznych edy-  
torach tekstu, takich jak Microsoft Word, pliki PDF tworzone i wyświetlane w pro-  
gramie Adobe Reader i innych czytnikach, pliki .ppt kreowane za pomocą programu Mi-  
crosoft PowerPoint itd. We wszystkich tych plikach znajdują się zakodowane, częst-  
o binarne informacje, a ich format decyduje o tym, za pomocą której aplikacji moż-  
na je uruchamiać, co jednocześnie jednak ogranicza możliwości wykorzystania innych  
programów.  
  
Znamy też pliki, które można uruchamiać niezależnie od aplikacji, czyli tekstowe  
lub – bardziej precyzyjnie – pliki w formacie ASCII. Zawartość takich plików sta-  
nowią standardowe symbole, cyfry i litery – wszystkie znaki, które można znaleźć  
na klawiaturze komputera. Zaletą takich plików jest to, że są łatwe do odczytania  
czy wykonania; generalnie praca z nimi jest prosta.  
zamień na vim (y/n/a/q/l/^E/^Y)? 5,18 Góra
```

Cofanie zmian w edytorze vi

Jeżeli czytelnik należy do grona tych autorów i programistów, którzy podczas pisania popełniają bardzo niewiele pomyłek i w związku z tym wprowadzają też niewiele poprawek, zapewne rzadko musi myśleć: „A niech to, chciałem zrobić coś innego”. Na szczęście dla wszystkich pozostałych, program zapamiętuje wszystkie wprowadzone w danej sesji zmiany, dzięki czemu można kolejno cofać je aż do początkowego stanu pliku. Aby cofnąć ostatnią wprowadzoną zmianę, należy użyć polecenia *u* (*undo* — cofnij). Każde kolejne jej użycie spowoduje cofnięcie kolejnej wprowadzonej zmiany w kolejności odwrotnej do ich wykonywania. Niektóre złożone operacje (takie jak wyszukaj i zastąp) są zapamiętywane jako jedna operacja, tak też są traktowane podczas ich wycofywania, ale w przypadku większości zmian polecenie *u* spowoduje usuwanie ich pojedynczo, aż do przywrócenia pierwotnego stanu pliku.

Korzystanie z wielu okien w programie vi

Jedną z uwag dotyczących funkcjonalności programu *vi*, która towarzyszy mu najdłużej, jest zarzut, że program nie może obsługiwać wielu okien w tym samym czasie. W aplikacjach pracujących w trybie tekstowym obsługa wielu okien zwykle polega na tym, że te, które są jednocześnie używane, widoczne są w postaci osobnych pasków statusu. Warto wspomnieć o tym, że Billy Joy pracował nad wersją obsługującą wiele okien, ale kod źródłowy został zagubiony, dlatego też do czasu modernizacji edytora i powstania wersji *vim* program nie posiadał tej opcji. A tak o tym mówił sam Billy Joy w wywiadzie udzielonym magazynowi *Unix Review* w sierpniu 1984 roku:

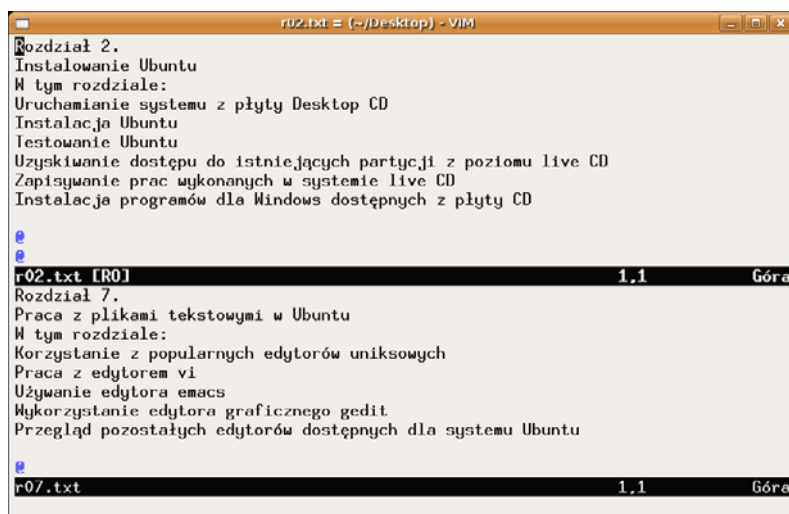
„Pracowałem nad dodaniem do *vi* obsługi wielu okien, kiedy zainstalowaliśmy nasz VAX, co miało miejsce w grudniu roku 1978. Nie mieliśmy żadnych kopii zapasowych, a napęd taśmowy zepsuł się. Kontynuowałem jednak pracę, chociaż bez możliwości wykonywania kopii zapasowych. I wtedy właśnie kod źródłowy uległ zniszczeniu i nie miałem kompletnych listingów. Musiałem przepisać na nowo niemal cały kod dotyczący

wyświetlania i obsługi okien, wtedy właśnie się poddałem. Później wróciłem do poprzedniej wersji, przygotowałem dokumentację kodu oraz podręcznik i zamknąłem wszystko. Gdyby kod nie uległ zniszczeniu (scrunch), wówczas vi byłby wyposażony w obsługę wielu okien i mógłbym w nim umieścić pewną programowalność — ale tego nie jestem pewien...”

Na szczęście, programiści vima rozwiązyali ten problem, implementując kilka komend, za których pomocą można podzielić wyświetlane okno, otworzyć inny plik w innej części okna, przełączać się pomiędzy różnymi plikami wyświetlanymi w różnych oknach itd. Na rysunku 7.5 przedstawiam edytor vim wyświetlający dwa różne pliki w dwóch osobnych oknach podczas tej samej sesji programu.

Rysunek 7.5.

*Obsługa wielu okien
w programie vim*



Oto podstawowe polecenia edytora vim służące do zarządzania wieloma oknami:

- ♦ `:hide` — zamyka bieżące okno (działa tylko wówczas, kiedy jest wyświetlanych wiele okien),
- ♦ `:only` — zamyka wszystkie okna z wyjątkiem bieżącego,
- ♦ `:split nazwa_pliku` — dzieli ekran w poziomie i wczytuje w drugim oknie wskazany plik,
- ♦ `:sview nazwa_pliku` — działa tak samo jak `split`, ale drugi plik wczytywany jest tylko w trybie do odczytu,
- ♦ `:vsplit nazwa_pliku` — dzieli ekran w pionie i wczytuje w drugim oknie wskazany plik.



Edytor vim można od razu uruchomić w trybie wielu okien, należy podczas startu użyć opcji `-o` oraz wskazać, które pliki mają zostać wczytane, np.:

```
$ vim -o plik1.txt plik2.txt plik3.txt
```

Kiedy już wyświetlimy wiele okien, do poruszania się pomiędzy nimi oraz korzystania z nich można posłużyć się następującymi kombinacjami klawiszowymi:

- ♦ *Ctrl+w*+strzałka do góry — przeniesienie kursora do górnego okna,
- ♦ *Ctrl+w+Ctrl+w* — przeniesienie kursora do innego okna (cyklicznie),
- ♦ *Ctrl+w+_* — zmaksymalizowanie bieżącego okna,
- ♦ *Ctrl+w+=* — wyrównanie rozmiarów wszystkich okien,
- ♦ *Ctrl+w++* — zwiększenie rozmiaru okna o jeden wiersz.

Po kilku eksperymentach oraz przyzwyczajeniu palców do używania nowych kombinacji klawiszy praca z wieloma oknami stanie się tak prosta jak w innych edytorach tekstu.

Dostosowywanie edytora vim

Program vim domyślnie podczas uruchamiania wczytuje trzy pliki konfiguracyjne, są to:

- ♦ plik konfiguracyjny systemu */etc/vimrc*,
- ♦ plik konfiguracyjny *.vimrc* zapisany w katalogu domowym każdego użytkownika,
- ♦ tradycyjny plik konfiguracyjny *.exrc* przypisany dla każdego użytkownika, zlokalizowany w katalogu domowym.

Jeżeli któregoś z tych plików nie będzie w systemie, nie oznacza to, że wystąpił błąd, ale wszystkie razem udostępniają bardzo elastyczny system konfigurowania programu. Każdy z nich może zawierać dowolną liczbę poniższych opcji konfiguracyjnych:

- ♦ *autocmd* — automatycznie wykonuje określone polecenie na podstawie typu edytowanego pliku,
- ♦ *iab* — umożliwia zastępowanie wyrazów innymi, przydatna jako opcja autokorekty,
- ♦ *map* — przypisuje polecenia edytora vim do poszczególnych klawiszy i umożliwia definiowanie własnych poleceń,
- ♦ *set* — przypisuje zmiennej vim wartość.

Oprócz wymienionych powyżej poleceń, vim obsługuje również obfity zestaw warunków, które umożliwiają tworzenie własnych funkcji i wykonywanie ich w różnych sytuacjach.

Oto przykład dostosowanego do potrzeb użytkownika pliku konfiguracyjnego *.vimrc*:

```
"
" wvh's .vimrc
"
" Ustawienia zmiennych
"
" nie używaj trybu kompatybilności z vi
set nocompatible
" używaj sprytnej wersji backspace
set backspace=2
" nie podświetlaj składni
syntax off
" zdefiniuj punkt, w którym wiersze są zawijane w trybie vanilla
set wrapmargin=10
" zawsze wyświetlaj informacje o wierszu i kolumnie
```

```

set ruler
" nie podświetlaj wszystkich znalezionych słów, tylko je wyszukaj
set nohlsl
" twórz kopię zapasową pliku poprzedzoną znakiem tyldy
set backuptext=
set backupext
"
" skróty
"
" autokorekta
iab użytkownik użytkownik
"
" mapowanie klawiszy
"
" użyj klawisza F5 do ponownego formatowania akapitu
map <F5> !}fmt <CR>
"
" specjalne ustawienia dla określonych typów plików
"
" ustaw specjalne opcje podczas edycji plików C
au FileType c setlocal ahiwidth=4 softtabstop=4 expandtab

```

Jak widać w powyższym przykładzie, linie rozpoczynające się od cudzysłowu zawierają komentarze wstawiane przez użytkownika.

Niestety, pełne omówienie wszystkich opcji, które można umieścić w pliku `.vimrc`, zajęłoby wystarczająco na obszerną książkę. Doskonałe przykłady zaawansowanych plików konfiguracyjnych można znaleźć na następujących stronach internetowych:

- ♦ www.stripey.com/vim/vimrc.html,
- ♦ www.cs.mcgill.ca/~navindra/editors/sven.vimrc.

Graficzne wersje edytora vi

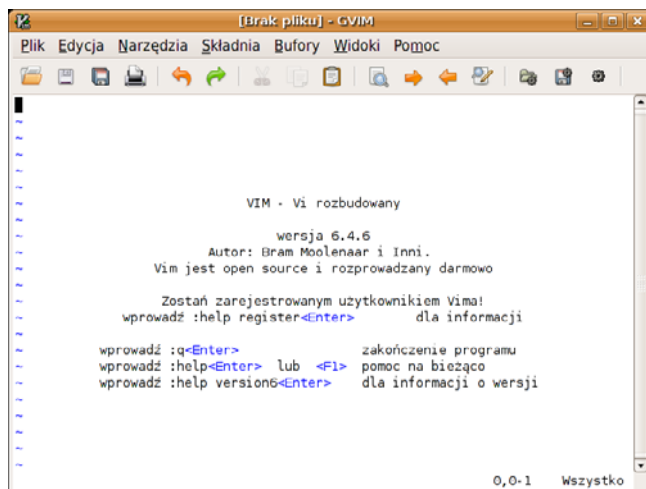
Wielu użytkowników, zwłaszcza tych, którzy wcześniej używali systemów, takich jak Microsoft Windows czy Mac OS X, oczekuje programów z graficznym interfejsem. To sprawia, że programiści pracujący nad programami, takimi jak vi czy emacs, mają do rozwiązania problem, jak przystosować dla tych użytkowników owe edytory, które zostały zaprojektowane jako aplikacje przeznaczone dla tych terminali, gdzie mysz kojarzy się tylko ze szkodnikiem. Aby nieco bardziej zaznajomić użytkowników z potężnymi narzędziami, takimi jak vi czy emacs (a następnie przeciągnąć ich, często wrzeszczących i wierzącących w XXI wiek), zaprojektowano wiele różnorodnych interfejsów graficznych. Na rysunku 7.6 przedstawiam gvim, czyli wersję edytora vim, do której dodano obsługę menu.



GNOME vim nie jest instalowany domyślnie w Ubuntu, ale łatwo można go dodać do systemu z repozytoriów za pomocą menedżera pakietów Synaptic, postępując zgodnie ze wskazówkami zamieszczonymi w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Istnieją również inne graficzne wersje edytora vim, np. kvim, który przeznaczony jest dla środowiska KDE. Z sieci dociera coraz więcej informacji, że programiści edytora kvim pracują teraz nad nowym projektem o otwartych źródłach i nazwie Yzisi; ma to być szybki,

Rysunek 7.6.
GNOME vim,
graficzna wersja
programu vim



wydajny edytor inspirowany przez vim, tworzony w systemie modułowym, dzięki czemu łatwiej będzie zaimplementować w nim graficzne rozszerzenia. Więcej informacji na temat tego projektu można znaleźć na stronie www.yzis.org.

Dodatkowe informacje na temat edytorów vi i vim

Ponieważ vi jest jednym z najstarszych edytorów dostępnych dla systemów uniksopodobnych, w sieci można znaleźć wiele różnorodnych źródeł jemu poświęconych, a uwzględniających również specyfikę vima. Oto kilka moich ulubionych stron:

- ♦ strona domowa vima: www.vim.org/,
- ♦ strona miłośników edytora vi: www.thomer.com/vi/vi.html,
- ♦ wprowadzenie do edycji plików w vi: <http://ex-vi.sourceforge.net/viin/paper-4.html>,
- ♦ wprowadzenie do opcji programu vim: www.burdell.org/articles/vim/,
- ♦ wyrażenia regularne w edytorze vim: www.geocities.com/volontir/.

Proste wyszukiwanie w internecie przyniesie setki adresów stron zawierających porady, triki czy ustawienia konfiguracyjne dla różnych wersji edytora vim. Warto pamiętać, że jedną z cennych cech programów, takich jak vim, jest fakt, że można je znaleźć w każdym systemie linuksowym, co więcej, działają one niemal tak samo w każdym z tych systemów. Kiedy już użytkownik dostosuje program do wszystkich swoich potrzeb i okaże się, że nie może bez tych zmian żyć, może to znak, że nadeszła pora na rozpoczęcie przygody z emacsem.

Korzystanie z programu emacs

Historię programu emacs opisałem na początku rozdziału, dlatego przejdę od razu do omówienia jego właściwości. Jak już wspominałem, emacs okazał się edytorem przeznaczonym właśnie dla mnie. Poglądy mogą się zmieniać, ale nie można ignorować mocy

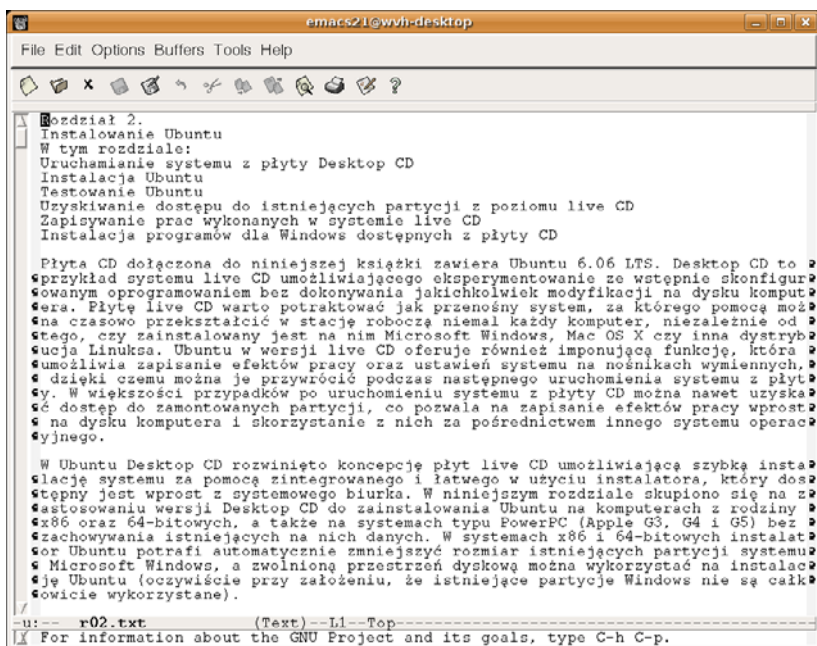
i elastyczności emacsa. Wiele starych argumentów przemawiających na niekorzyść edytora, takich jak wymagania stawiane ilości pamięci, straciły znaczenie wraz z ogólnym zwiększaniem się wymogów stawianych przez programy systemom komputerowym oraz usprawnieniami wprowadzanymi w dzisiejszych komputerach. Fakt, że emacs po uruchomieniu od razu umożliwia edycję plików (nie korzysta z trybów, jak vi), to wystarczający argument przemawiający za jego stosowaniem.



Z nieznanых powodów pakiet emacs21 nie jest instalowany domyślnie w Ubuntu. Dlatego przed rozpoczęciem jakichkolwiek testów tego edytora należy dodać go do systemu za pomocą któregoś z narzędzi do zarządzania oprogramowaniem (zgodnie z informacjami zamieszczonymi w rozdziale 20. „Dodawanie, usuwanie i aktualizacja oprogramowania”). Jest to zresztą pierwsza rzecz, którą wykonuję po zainstalowaniu Ubuntu. Z mojego punktu widzenia, Linux bez emacsa nie ma racji bytu.

Domyślna wersja programu instalowana w Ubuntu jest zintegrowana z systemem X Window, dzięki czemu po włączeniu uruchamia swoje okno z własnym menu i paskiem narzędziowym ułatwiającym wybieranie często wykonywanych poleceń. Na rysunku 7.7 przedstawiam okno edytora emacs, w którym wyświetlany jest fragment niniejszego rozdziału.

Rysunek 7.7.
Emacs w Ubuntu



Edytor emacs może być również uruchamiany w trybie tekstowym, co jest przydatne, kiedy trzeba szybko wykonać edycję pliku w trakcie połączenia ze zdalnym komputerem, kiedy nie jest potrzebne wyświetlanie okna zdalnego systemu na monitorze komputera. Aby nie uruchamiać programu w trybie okienkowym, należy dodać opcję `-nw`. Na rysunku 7.8 widoczny jest ten sam plik, ale program działa w xterm.

Jedną z najlepszych funkcji emacsa jest możliwość jego dostosowywania, która obejmuje również tryb graficzny. Na rysunku 7.9 przedstawiam okno programu z ukrytym po prawej stronie paskiem narzędziowym.

Rysunek 7.8.

Emacs w Ubuntu
i w xterm

```
File Edit Options Buffers Tools Help
Rozdział 2.
Instalowanie Ubuntu
W tym rozdziale:
Uruchamianie systemu z płyty Desktop CD
Instalacja Ubuntu
Testowanie Ubuntu
Uzyskiwanie dostępu do istniejących partycji z poziomu live CD
Zapisywanie prac wykonanych w systemie live CD
Instalacja programów dla Windows dostępnych z płyty CD

Płyta CD dołączona do niniejszej książki zawiera Ubuntu 6.06 LTS. Desktop CD to
przykład systemu live CD umożliwiającego eksperymentowanie ze wstępnie skonfigu-
rowanym oprogramowaniem bez dokonywania jakichkolwiek modyfikacji na dysku kompu-
tlera. Płyte live CD warto potraktować jak przenośny system, z którego pomocą
można czasowo przekształcić w stację roboczą niemal każdy komputer, niezależnie
od tego, czy zainstalowany jest na nim Microsoft Windows, Mac OS X czy inna dystrybucja
Linuxa. Ubuntu w wersji live CD oferuje również imponującą funkcję, która
umożliwia zapisanie efektów pracy oraz ustawień systemu na nośnikach wymiennych,
dzięki czemu można je przywrócić podczas następnego uruchomienia systemu z płyty.
W większości przypadków po uruchomieniu systemu z płyty CD można nawet uzyskać
dostęp do zamontowanych partycji, co pozwala na zapisanie efektów pracy
wprost na dysku komputera i skorzystanie z nich za pośrednictwem innego systemu
operacyjnego.

--uu:--f1 r02.txt (Text)--L1--Top-----
For information about the GNU Project and its goals, type C-h C-p.
```

Rysunek 7.9.

Dostosowywanie
okna programu
emacs

```
File Edit Options Buffers Tools Help
Rozdział 2.
Instalowanie Ubuntu
W tym rozdziale:
Uruchamianie systemu z płyty Desktop CD
Instalacja Ubuntu
Testowanie Ubuntu
Uzyskiwanie dostępu do istniejących partycji z poziomu live CD
Zapisywanie prac wykonanych w systemie live CD
Instalacja programów dla Windows dostępnych z płyty CD

Płyta CD dołączona do niniejszej książki zawiera Ubuntu 6.06 LTS. Desktop CD to
przykład systemu live CD umożliwiającego eksperymentowanie ze wstępnie skonfigu-
rowanym oprogramowaniem bez dokonywania jakichkolwiek modyfikacji na dysku kompu-
tlera. Płyte live CD warto potraktować jak przenośny system, z którego pomocą
można czasowo przekształcić w stację roboczą niemal każdy komputer, niezależnie
od tego, czy zainstalowany jest na nim Microsoft Windows, Mac OS X czy inna dystrybucja
Linuxa. Ubuntu w wersji live CD oferuje również imponującą funkcję, która
umożliwia zapisanie efektów pracy oraz ustawień systemu na nośnikach wymiennych,
dzięki czemu można je przywrócić podczas następnego uruchomienia systemu z płyty.
W większości przypadków po uruchomieniu systemu z płyty CD można nawet uzyskać
dostęp do zamontowanych partycji, co pozwala na zapisanie efektów pracy wprost
na dysku komputera i skorzystanie z nich za pośrednictwem innego systemu operac-
yjnego.

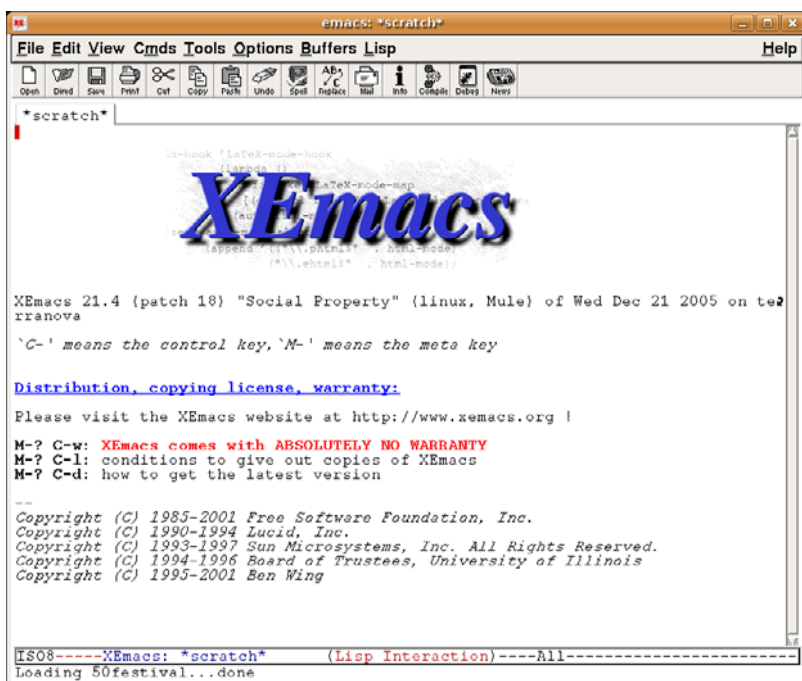
W Ubuntu Desktop CD rozwinięto koncepcję płyt live CD umożliwiającą szybką insta-
lację systemu za pomocą zintegrowanego i łatwego w użyciu instalatora, który dos-
tępny jest wprost z systemowego biurka. W niniejszym rozdziale skupiono się na
dostosowaniu wersji Desktop CD do zainstalowania Ubuntu na komputerach z rodziny
x86 oraz 64-bitowych, a także na systemach typu PowerPC (Apple G3, G4 i G5) bez
zachowywania istniejących na nich danych. W systemach x86 i 64-bitowych instalat-
or Ubuntu potrafi automatycznie zmniejszyć rozmiar istniejących partycji systemu
Microsoft Windows, a zwolnioną przestrzeń dyskową można wykorzystać na instalac-
ję Ubuntu (oczywiście przy założeniu, że istniejące partycje Windows nie są całk-
owicie wykorzystane).

Ostrzeżenie
Instalacja Ubuntu na komputerach PowerPC (PPC) z zachowaniem istniejącego system-
u i opcją podwójnego rozruchu została omówiona w rozdziale 3., w podrozdziale 3.2.
--u:* r02.txt (Text)--L3--Top-----
```

Zwolennikom graficznego dostosowywania oraz używania różnorodnych czcionek w sesjach edytora emacs warto polecić wersję programu o nazwie XEmacs. Na rysunku 7.10 przedstawiłem okno startowe tej wersji, która oferuje możliwość korzystania z różnych czcionek, rozbudowanych pasków narzędziowych oraz ikon, a nawet wyświetlanie grafik.

Pierwotnie XEmacs był projektem firmy Lucid Inc., ale teraz jest projektem o otwartym kodzie, który można znaleźć na stronie www.xemacs.org. XEmacs był projektowany jako bardziej graficzna wersja emacsa, która oferuje również obsługę myszy. Ze względu na kilka aspektów związanych z licencjami, kod XEmacs nigdy nie został włączony do

Rysunek 7.10.
Okno startowe
XEmacsa, czyli
„innego” emacsa



oryginalnego GNU emacsa i do dziś funkcjonuje jako oddzielny program. Niezależnie od tego, XEmacs to znakomita wersja emacsa, która dla wielu użytkowników jest atrakcyjniejsza od wersji oryginalnej, dlatego warto się z nią zapoznać. XEmacs nie jest oficjalnie wspierany przez programistów Ubuntu, a zatem, aby go zainstalować, należy dodać repozytoria *multiverse*, a następnie za pomocą menedżera pakietów zainstalować program zgodnie ze wskazówkami umieszczonymi w rozdziale 20. „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Kilka słów o poleceniach w edytorze emacs

Nim opowiem, jak w edytorze emacs wykonać różne działania, uważam, że warto poświęcić chwilę na zapoznanie się z poleceniami dostępnymi w edytorze.

Ponieważ emacs nie pracuje w trybach, wprowadzenie jakiegokolwiek znaku z klawiatury spowoduje jego umieszczenie w pliku. Znakomicie sprawdza się to w środowisku graficznym, w którym można korzystać z poleceń menu do wykonania większości zadań. Ponieważ jednak edytor może pracować również poza środowiskiem graficznym, musi istnieć inny sposób na wprowadzanie standardowych poleceń.

Wszystkie polecenia edytora emacs można wykonać za pomocą kombinacji klawiszy umieszczonych na klawiaturach współczesnych komputerów. Wykorzystuje się zwłaszcza klawisze, takie jak *Ctrl* (czyli *Control*), *Esc* (czyli *Escape*) oraz *Alt* i *Shift*. Jest to sposób na organizację pracy komputera, te klawisze w połączeniu ze standardowymi znakami zmieniają sygnały, które wysyłają klawisze użyte samodzielnie, bez tych specjalnych.

Sposób działania klawiszy *Ctrl*, *Esc*, oraz *Alt* jest taki sam jak klawisza *Shift*, kiedy za jego pomocą uzyskuje się duże litery — zostaje wysłany zupełnie inny sygnał i to właśnie dlatego komputer wie, co ma zrobić.

Podobnie jest z poleceniami programu *emacs*, które są wprowadzane za pomocą kombinacji klawiszy, takich jak *Ctrl+S*, *Esc+S*, *Alt+S*, *Ctrl+Shift+S* itd. Jest to jeden z tych aspektów pracy z edytorami, który może przysporzyć ich użytkownikom sporo rozrywki zwłaszcza wtedy, gdy obserwuje się kogoś korzystającego z kombinacji klawiszy w celu wstawiania tekstu. Jeden z dawnych kolegów autora, obserwując go podczas takiego pisanja, powiedział, że przypomina „epileptycznego pajaka”. Faktycznie, to określenie zawiera w sobie dużo prawdy i doskonale opisuje, jak działają skróty klawiszowe.

Klawisz *Control* modyfikuje działanie innych klawiszy, jeżeli podczas ich wciśnięcia sam jest wciśnięty. W zapisie oddaje się to zwykle jako *Control+S*, co oznacza: „Wciśnij i przytrzymaj klawisz *Control*, a następnie wciśnij klawisz *S*”. Klawisz *Ctrl* działa bardzo podobnie jak klawisz *Shift*, ale nie można włączyć go na stałe, tak jak można to zrobić za pomocą klawisza *Shift Lock*, który swoim działaniem często irytuje użytkowników.



W druku często spotyka się skróconą wersję zapisu użycia klawisza *Ctrl* w postaci symbolu \wedge . I tak np. zapisy $\wedge s$ i *Ctrl+s* to dwie metody określenia tej samej kombinacji klawiszy. Ta pierwsza jest szybsza, ponadto zapobiega ewentualnym rozterkom użytkowników, czy widoczny znak $+$ również należy wpisywać. W dalszej części książki używany będzie zapis typu *Ctrl+s*, tylko od czasu do czasu może pojawić się wersja skrócona $\wedge s$. *Emacs* sam w sobie potrafi być wystarczająco skomplikowany, nie warto utrudniać korzystania z niego, stosując kilka konwencji zapisu.

Z drugiej strony, są jednak klawisze *Esc* i *Alt*, które działają nieco inaczej; należy je wciśnąć i zwolnić, dopiero potem wprowadzać kolejne znaki. Takie kombinacje również są zapisywane w postaci *Esc+s* i *Alt+s*, ale działanie klawiatury jest już inne. Najpierw następuje wciśnięcie i zwolnienie klawisza i jest to jednocześnie sygnał dla systemu, że nastąpi wciśnięcie klawisza z właściwą komendą i nie jest to kombinacja składająca się z jednoczesnego naciskania kilku klawiszy.

Byłoby to zupełnie proste, gdyby polecenia edytora *emacs* ograniczały się do możliwości pojedynczego wciśnięcia któregoś z klawiszy *Alt*, *Ctrl*, *Esc* czy *Shift*. Ograniczyłoby to jednocześnie liczbę możliwych kombinacji do ok. 100. Jednak w większości poleceń edytora *emacs* wykorzystuje się kombinację klawiszy *Alt*, *Ctrl*, *Esc* i *Shift*, co daje właściwie nieskończoną liczbę trudnych do zapamiętania kombinacji, prowadzi do komentarzy, takich jak ten o epileptycznym pajaku, oraz zadziwia wszystkich, którzy próbują opisać kombinację typu *Esc+s+Ctrl+Shift+s*. Mimo wszystko, jest to niewielka cena za możliwość korzystania z milionów kombinacji.



Sprawą, niewątpliwie wprawiającą w zakłopotanie użytkowników innych edytorów tekstu, którzy zaczynają przygodę z *emacs*em, jest fakt, że zupełnie niestandardowo rozwiązano w nim standardowe czynności, takie jak kopiowanie, wycinanie i wklejanie tekstu. Nie jest to najszczęśliwsze rozwiązanie, ale przyczyna tkwi w tym, że kombinacje zwykle używane do takich działań zostały już wcześniej przypisane do innych operacji. Do kogo więc należało się dostosować: do tysięcy stałych użytkowników edytora czy też do zupełnie początkujących? No właśnie...

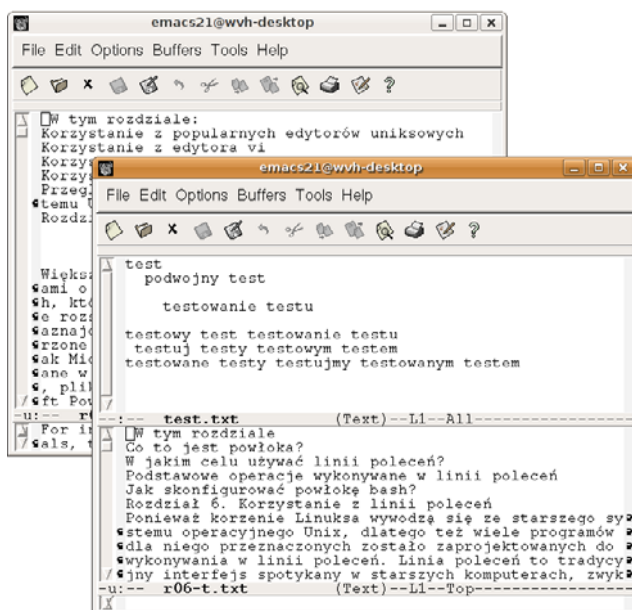
Terminologia

Ponieważ GNU emacs został zaprojektowany do pracy z graficznym menedżerem okien, do opisu zawartości ekranu należy zastosować nieco inne słownictwo. W tym punkcie objaśniam terminologię, która będzie używana w dalszej części rozdziału.

Na rysunku 7.11 przedstawiłem dwa okna z programem emacs uruchomione w Ubuntu, w każdym z nich wyświetlany jest inny plik. Każde z okien posiada swój własny pasek menu i pasek narzędzi. W oknie znajdującym się na pierwszym planie wyświetlane są dwa różne pliki, dla każdego z nich został utworzony osobny bufor. Na dole każdego z dwóch okien znajduje się pasek z informacjami na temat wyświetlanego pliku i jego statusu. Na dole każdego okna z programem jest wyświetlany pasek stanu, w którym wyświetlane są komunikaty i funkcje programu.

Rysunek 7.11.

Okienkowa terminologia dla GNU emacsa



Termin okno ma w tu inne znaczenie niż w części poświęconej edytorom vi i vim. Jest tak dlatego, że w emacsie od początku brano pod uwagę system X Window i możliwość wyświetlania wielu okien na ekranie, podczas gdy w mógł tylko dzielić okno terminala na kilka części. W tym kontekście okno w edytorze vi jest ekwiwalentem bufora w edytorze emacs.

Uruchamianie i wyłączanie edytora emacs

W edytorze emacs, podobnie jak w vi, można rozpocząć edytowanie plików na kilka sposobów.

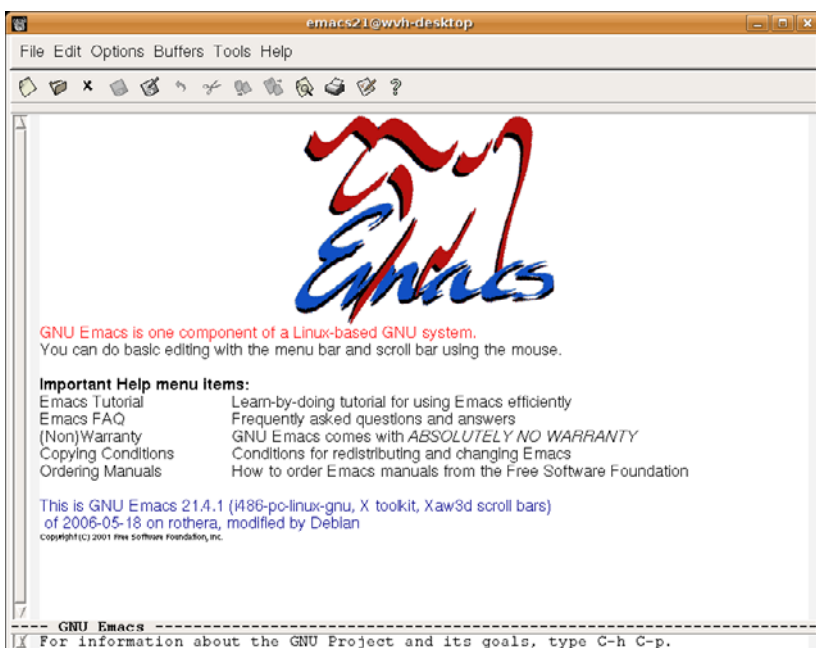
- ♦ Można wybrać z menu *Programowanie/Emacs 21 (X11)*.
- ♦ W każdym oknie xterm lub terminala GNOME można wpisać polecenie `emacs nazwa_pliku`, gdzie *nazwa_pliku* wskazuje dokument, który ma być edytowany.

Domyślnie uruchomienie edytora z terminala w Ubuntu spowoduje otwarcie nowego okna dla programu, dlatego warto podać od razu nazwę pliku, dzięki czemu będzie można od razu rozpocząć pracę z nim.

W obu przypadkach program uruchomi nowe okno, w którym wyświetli plik poddawany edycji (patrz rysunek 7.7). Jeżeli program zostanie uruchomiony z menu GNOME, wówczas ekran startowy będzie wyglądał tak, jak na rysunku 7.12.

Rysunek 7.12.

Ekran startowy
edytora GNU emacs



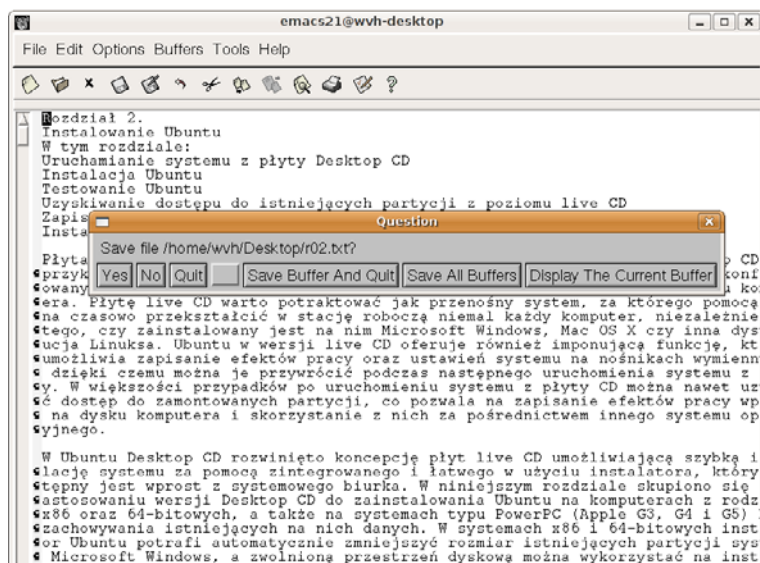
Aby zakończyć działanie programu, należy użyć polecenia *Exit Emacs* z menu *File* lub kombinacji klawiszy `^x^c` (przypominam, że oznacza to wciśnięcie i przytrzymanie klawisza *Ctrl*, a następnie wciśnięcie klawisza *x*, a później klawisza *c*, po czym można zwolnić klawisz *Ctrl*).

Jeżeli w edytowanym pliku nie zostały wprowadzone żadne zmiany lub też wszystkie zostały zapisane, wówczas program natychmiast zakończy działanie. Jeśli jednak zmiany zostały wprowadzone, a nie są zapisane, wówczas program wyświetli komunikat z pytaniem, czy zmiany mają być zapisane (patrz rysunek 7.13).

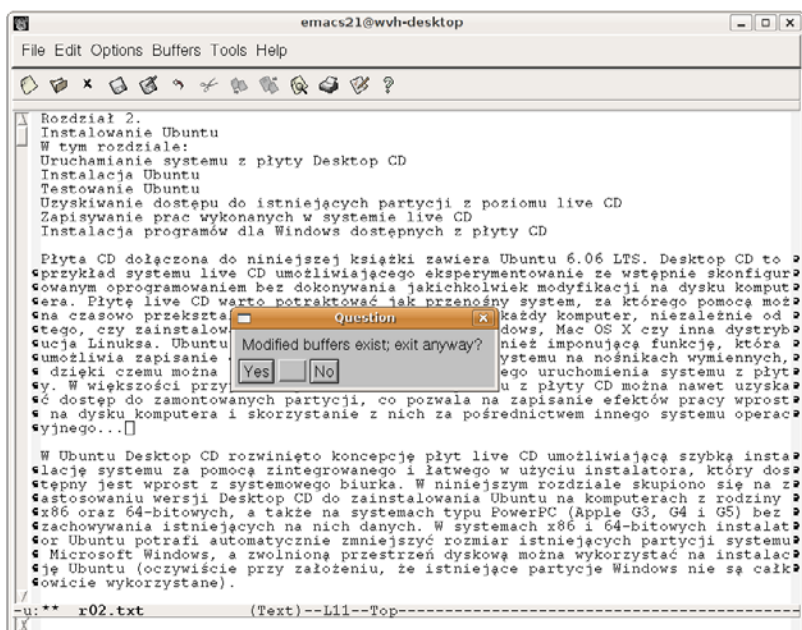
Aby wyłączyć program bez zapisywania zmian, należy wybrać opcję *No*; aby zmiany zapamiętać, trzeba kliknąć *Yes*. Po wybraniu tej drugiej opcji program zakończy działanie natychmiast po tym, jak zostaną wprowadzone zmiany do wszystkich edytowanych plików. Jeżeli użytkownik nie będzie chciał zapisać zmian i wybierze opcję *No*, wówczas zostanie wyświetlony jeszcze jeden komunikat, w którym program poprosi o ponowne potwierdzenie tego, że żadne zmiany nie mają być zapisane (patrz rysunek 7.14). Program nie będzie błagał użytkownika, aby ten zapisał jakiś plik, jest to tylko upewnienie się na wszelki wypadek, by uniknąć sytuacji, w której przez nieuwagę znikłaby wielka powieść.

Rysunek 7.13.

Potwierdzenie
zapamiętania
wprowadzonych
zmian

**Rysunek 7.14.**

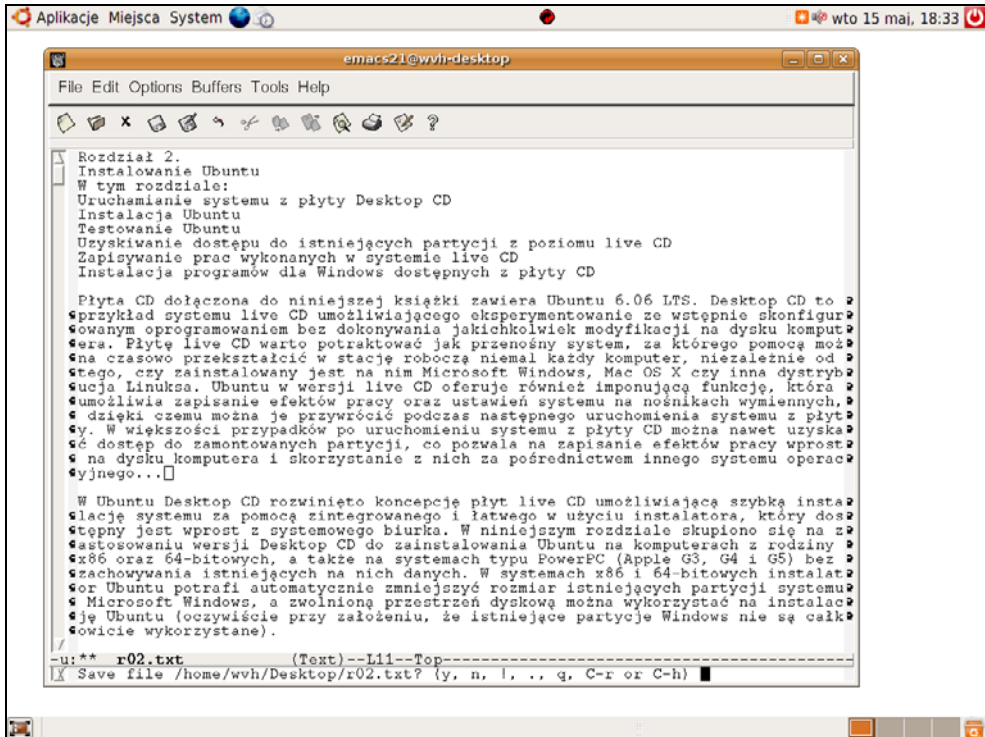
Ostateczne okno
z potwierdzeniem
faktu niezapisywania
żadnych zmian
wprowadzonych
w pliku



Taką samą prośbę o potwierdzenie zobaczymy, kiedy użytkownik będzie wyłączał program, korzystając z polecenia `^x^c`, tyle że zostanie ona wyświetlona w buforze wiadomości (patrz rysunek 7.15).

Poruszanie się w edytorze emacs

Do przemieszczania kursora w edytorze można wykorzystać wiele różnych skrótów klawiaturowych. Oto one:



Rysunek 7.15. Ostateczne okno z potwierdzeniem faktu niezapisywania żadnych wprowadzonych w pliku zmian w przypadku zamykania programu z wykorzystaniem kombinacji klawiszy

- ♦ $\wedge a$ — przejdź na początek bieżącego wiersza,
- ♦ $\wedge e$ — przejdź na koniec bieżącego wiersza,
- ♦ $\wedge f$ — przejdź do przodu o jeden znak,
- ♦ $\wedge b$ — przejdź do tyłu o jeden znak,
- ♦ $\wedge p$ — przejdź do poprzedniego wiersza,
- ♦ $\wedge n$ — przejdź do następnego wiersza,
- ♦ $\wedge v$ — przewiń wyświetlaną zawartość o jeden ekran w dół,
- ♦ $Esc + b$ — przejdź do przodu o jedno słowo,
- ♦ $Esc + f$ — przejdź do tyłu o jedno słowo,
- ♦ $Esc + v$ — przewiń wyświetlaną zawartość o jeden ekran w górę,
- ♦ $Esc + <$ — przejdź do początku bieżącego bufora,
- ♦ $Esc + >$ — przejdź do końca bieżącego bufora.

Są to najpopularniejsze komendy używane do przemieszczania kursora w obrębie bieżącego okna. Podczas korzystania z edytora w trybie graficznym, ekran można przewijać, klikając i przeciągając suwaki znajdujące się w oknie programu, lub za pomocą samego kursora myszy.

Informacje na temat poleceń używanych do przemieszczania się między wieloma oknami wyświetlanymi na ekranie monitora zostały zamieszczone w punkcie „Praca z wieloma oknami i buforami w edytorze emacs”.

Wycinanie, kopiowanie i wklejanie w edytorze emacs

Ze względu na to, że emacs jest przystosowany do pracy w środowisku graficznym, oraz na to, że do jego obsługi można używać myszy, dostępne są dwie metody służące do zaznaczania, kopiowania lub przenoszenia tekstu:

- ♦ użycie lewego przycisku myszy — wystarczy ustawić kursor w odpowiednim miejscu, wcisnąć i przytrzymać lewy przycisk myszy i przeciągnąć go po tekście, który zostanie w ten sposób zaznaczony. Jeżeli kursor zostanie przesunięty poza linię bufora emacsa, wówczas program automatycznie przewinie zawartość ekranu.
- ♦ ustawienie zaznaczenia w odpowiednim miejscu w tekście za pomocą polecenia *Ctrl+@* lub *Ctrl*+klawisz spacji, a następnie przesunięcie kursora do końca tekstu, który ma być skopiowany lub przeniesiony.

Po zidentyfikowaniu obszaru, który ma zostać skopiowany lub przeniesiony, można skopiować go do bufora, korzystając z polecenia *Esc+w*, lub wyciąć i wkleić do bufora (polecenie *Ctrl+w*). Następnie należy umieścić kursor w miejscu, gdzie ma być wklejony czy skopiowany tekst, i użyć kombinacji klawiszy *Ctrl+y*.



Jeżeli użytkownik chce usunąć tekst, począwszy od miejsca, w którym znajduje się kursor, aż do końca bieżącej linii, wówczas należy użyć polecenia *^k*, a następnie, aby tekst wkleić, *^y*.

Wyszukiwanie i zastępowanie tekstu w edytorze emacs

Aby skorzystać z bardzo wygodnego trybu wyszukiwania dostępnego w edytorze emacs, należy użyć polecenia *^s*. Po jego zastosowaniu w buforze zostanie wyświetlony komunikat *I-Search*, a program rozpocznie przeszukiwanie pliku (w dół), próbując zlokalizować pierwszy pasujący ciąg znaków według wzorca podanego przez użytkownika. Jeżeli np. użytkownik będzie chciał wyszukać tekst „melon”, to kiedy będzie wpisywał kolejne litery, program spróbuje odnaleźć pasujące ciągi znaków najpierw dla litery „m”, następnie dla ciągu „me” itd. Aby wyszukiwanie przerwać w dowolnym momencie, należy użyć polecenia *^g*, a wtedy kursor zostanie ustawiony w miejscu, w którym rozpoczęto poszukiwanie. Wprowadzenie jakiegokolwiek innego polecenia programu emacs również spowoduje przerwanie wyszukiwania i pozostawienie kursora w bieżącej pozycji w buforze.

Praca z wieloma oknami i buforami w edytorze emacs

Użytkownicy korzystający z trybu graficznego programu mogą całkiem prosto przełączać się pomiędzy buforem a oknami, korzystając z myszy — wystarczy kliknąć lewym przyciskiem myszy wybrany element i to wszystko. Warto jednak wiedzieć, jak przełączać się pomiędzy buforami w sytuacji, kiedy pracuje się w oknie, które już zostało podzielone, lub program nie działa w trybie graficznym.

Aby przełączyć się do innego bufora, który jest aktualnie wyświetlany, należy użyć kombinacji $\wedge x \wedge o$. Przełącza ona kolejne okna widoczne na ekranie.

Jeżeli użytkownik chce przełączyć się do innego bufora, niezależnie od tego, czy jest on widoczny na ekranie, czy też nie, wówczas należy wcisnąć klawisze $\wedge x + b$ ($Ctrl + x$, a następnie klawisz b) i podać nazwę bufora, który ma zostać wyświetlony.

Jeżeli użytkownik nie pamięta nazw plików, które są edytowane, lub też używa nazw tymczasowych, wówczas można użyć polecenia $\wedge x \wedge b$, które wyświetli listę dostępnych buforów, następnie wystarczy już tylko kliknąć środkowym przyciskiem myszy odpowiednią pozycję.

Dostosowywanie edytora emacs

Jeżeli czytelnik miał kiedykolwiek możliwość skorzystania z komputera należącego do wielbiciela edytora emacs, z pewnością zauważył, że program ten pracuje nieco inaczej niż w systemie używanym przez niego. Przyczyną jest fakt, że emacs jest najbardziej konfigurowalnym edytorem, jaki istnieje. Oferuje on nie tylko możliwość zmieniania kombinacji klawiszy skojarzonych z poszczególnymi poleceniami (tzw. przypisanie klawiszy), ale też pełną implementację języka programowania Lisp, dzięki czemu zaawansowani użytkownicy mogą tworzyć własne polecenia lub modyfikować już istniejące.

W większości dystrybucji Linuksa ustawienia konfiguracyjne edytora emacs dla każdego użytkownika systemowego są przechowywane w pliku konfiguracyjnym o nazwie *.emacs* znajdującym się w katalogu domowym. Jak wspominałem wcześniej, to właśnie możliwość i łatwość konfigurowania oraz dostosowywania do własnych potrzeb przyczyniły się do popularności emacsa. Poniżej opisałem, jak można dostosować program do swoich potrzeb, edytując plik konfiguracyjny. W sieci można znaleźć wiele doskonałych stron poświęconych temu zagadnieniu, warto odwiedzić chociażby <http://jeremy.zawodny.com/emacs/emacs-4.html> czy <http://linuxplanet.com/linuxplanet/tutorials/3166/4/>.

Zmiana przypisania klawiszy

Podczas wykonywania poleceń programu emacs działają funkcje języka Lisp. Wcześniej pisałem, że emacs jest skonfigurowany tak, aby wykonywać określone działania jako reakcje na wciśnięcie kombinacji klawiszy *Ctrl* i *Esc* ze zwykłymi klawiszami. Są to tzw. przypisania klawiszy, ponieważ dane funkcje są skojarzone (przypisane do) z odpowiednimi kombinacjami. Jeżeli jednak użytkownik jest już przyzwyczajony do pracy z innym edytorem, w którym również używane są kombinacje z klawiszami *Ctrl* i *Esc*, wówczas pewnym problemem może okazać się przyzwyczajanie palców do nowych układów. Dlatego jedną z najczęściej wykonywanych zmian w programie jest określenie nowych skrótów klawiszowych.

Edytor emacs umożliwia zmianę ustawienia klawiszy dla całego programu lub dla poszczególnych trybów, częściej spotyka się zmianę dotyczącą wprowadzenia nowych kombinacji tak, aby działały niezależnie od trybu, w jakim pracuje edytor. Polecenie służące do zmiany tych ustawień to *global-set-binding*. Aby np. kombinacja klawiszy *Ctrl+z* przewijała zawartość bufora o jeden wiersz do góry zamiast zawieszania działania programu, należy w pliku *~/.emacs* wprowadzić następujące polecenie:

```
(global-set-key ?\C-Z? ?scroll-one-line-up)
```

Zgodnie ze stylem języka Lisp, nazwę funkcji należy poprzedzić znakiem zapytania. Dla zainteresowanych: funkcja `scroll-one-line-up` została omówiona również poniżej w podpunkcie „Definiowanie własnych funkcji”.



Podczas definiowania dowiezań klawiszy w pliku konfiguracyjnym emacsa klawisz `Ctrl` zwykle opisywany jest jako `\C-`, a klawisz `Esc` jako `\M-`.

Ustawianie zmiennych

Do kontrolowania działania edytora emacs wykorzystuje się kilka zmiennych. Można je modyfikować za pomocą polecenia `setq`. I tak, aby np. zmusić program do nieco bardziej płynnego przewijania wiersza, można dodać do pliku `~/.emacs` następującą linię:

```
(setq scroll-step 1)
```

Kolejny przykład, zgodnie z domyślnym ustawieniem edytora, automatycznie zapisuje plik, kiedy liczba wpisanych lub zmodyfikowanych znaków przekroczy 300. Aby zmniejszyć tę ilość do 100, wystarczy do pliku `~/.emacs` dodać następujący wiersz:

```
(setq auto-save-interval 100)
```

Modyfikowanie ustawień trybów działania

Jak wspominałem wcześniej, emacs oferuje różne skróty klawiszowe oraz funkcje zależne od trybu, w jakim pracuje. Domyślny tryb, w jakim działa program po uruchomieniu bez określania pliku, który ma być edytowany, znany jest jako **tryb fundamentalny** (ang. *fundamental mode*). Użytkownik może skonfigurować edytor tak, aby niezależnie od rodzaju otwieranego pliku używał określonego trybu. Aby sprawić, że tryb tekstowy stanie się domyślny podczas każdego uruchomienia systemu, należy do pliku `~/.emacs` dodać następujący wiersz:

```
(setq default-major-mode 'text-mode)
```

Jak widać na podstawie powyższego przykładu, nazwa domyślnego trybu działania została zdefiniowana za pomocą ustawienia zmiennej programu `default-major-mode`, która odmiennie niż w poprzednim przykładzie wymaga podania nazwy funkcji, a nie wartości numerycznej.

Każdy tryb działania edytora emacs umożliwia określenie specyficznych dla niego akcji. Aby np. włączyć autouzupełnianie, należy do pliku `~/.emacs` wprowadzić następujący wiersz:

```
(add-hook 'text-mode-hook 'turn-on-auto-fill)
```

Zmiana powiązań plików

Emacs automatycznie przechodzi do odpowiedniego trybu na podstawie rozszerzenia otwieranego pliku, podobnie jak ma to miejsce w większości aplikacji z interfejsem graficznym. Użytkownik może zmienić domyślne ustawienie edytora, które powoduje uru-

chomienie danego trybu na podstawie odpowiedniego rozszerzenia pliku; np. program automatycznie uruchamia tryb tekstowy po otwarciu pliku z rozszerzeniem *.txt* lub *.text*. Co zrobić w sytuacji, kiedy użytkownik nadaje tworzonemu przez siebie plikom tekstowym rozszerzenie *.inf*?

Lista rozszerzeń plików skojarzonych z odpowiednimi trybami działania edytora emacs jest przechowywana w postaci par: rozszerzenie / nazwa trybu. Wystarczy więc dodać rozszerzenie *.inf* do trybu tekstowego, co spowoduje jego przypisanie. W tym celu należy wprowadzić następujący zapis:

```
(set auto-mode-alist
  cons '(".inf" . text-mode) auto-mode-alist))
```

W tym przykładzie użyto funkcji `cons` języka Lisp do skojarzenia nowej pary: pliku o rozszerzeniu *.inf* oraz trybu tekstowego. Wykorzystano też omówioną wcześniej funkcję `setq` oraz zmienną `auto-mode-alist` do wprowadzenia nowej pozycji na liście.

Definiowanie własnych funkcji

Fakt, że w edytorze emacs zaimplementowano język Lisp, powoduje, że tworzenie własnych funkcji jest proste; to dodatkowy argument przemawiający za tym, aby ten język opanować (oczywiście, jeżeli trzeba). Funkcje Lispa są definiowane za pomocą polecenia `defun`, co obejmuje listę pięciu wartości jako argumentów. Bez zamieniania tego fragmentu w kurs języka Lisp przedstawię krótki przykład definiowania funkcji i sprawdzania jej komponentów.

Poniższy kod napisany w języku Lisp definiuje prostą funkcję służącą do przewijania zawartości bieżącego okna w górę o określoną liczbę wierszy z domyślną liczbą 1:

```
(defun przewijanie-o-jeden-wiersz-do-gory
  (&optional arg)
  "Przewin wybrany ekran do gory o N wierszy."
  (interactive "p")
  (scroll-up (or arg 1)))
)
```

Nazwa funkcji jest definiowana w pierwszym wierszu za pomocą polecenia `defun`. Drugi argument (opcjonalny) to lista argumentów używanych przez definiowaną funkcję. W tym przypadku wykorzystano argument `&optional`. Trzeci argument polecenia `defun` to komunikat z informacją o funkcji, zamknięty w dwóch cudzysłowach. Czwarty argument określa, czy funkcja jest interaktywna (w języku Lisp `p` oznacza wartość `true`). Piąty argument to kod definiowanej funkcji, który w tym przypadku spowoduje wykorzystanie wbudowanej w emacs funkcji przewijania o ilość razy określoną przez argument numeryczny.

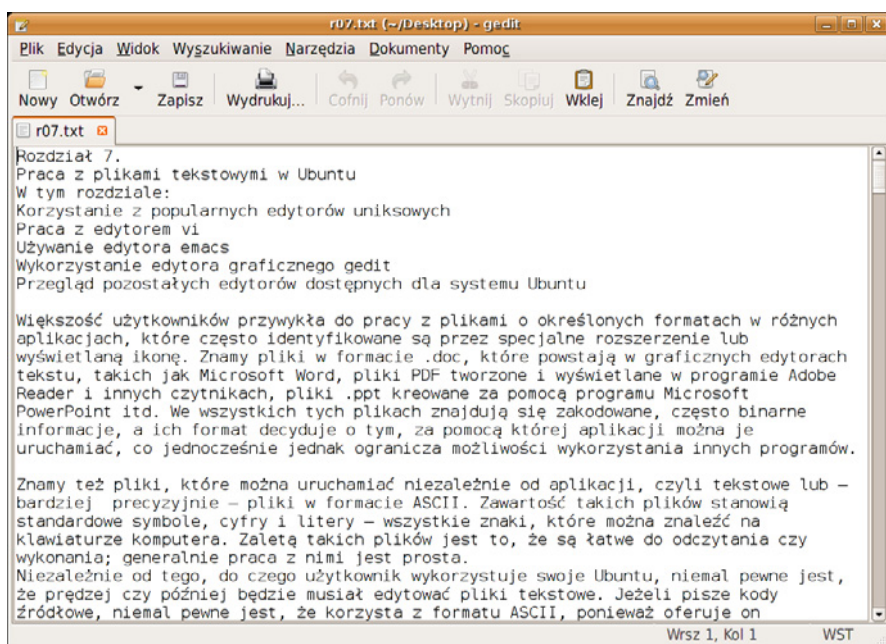
Oczywiście, powyższy fragment nie sprawi, że czytelnicy staną się specjalistami od języka Lisp, ale sama świadomość, że emacs oferuje takie możliwości, może być inspirująca. Jak widać na powyższym przykładzie, definiowanie prostych funkcji nie jest aż tak skomplikowane. Moc i elastyczność emacsa dają szansę na to, aby umieścić w edytorze tekstów funkcje, o których użytkownik zawsze marzył.

Zdobywanie dodatkowych informacji o edytorze emacs

Edytor emacs ma swój własny samouczek oraz pliki pomocy, wystarczy tylko wprowadzić polecenie `M+X`. Samouczek oferuje tryb nauczania poprzez wykonywanie konkretnych czynności. W sieci istnieje wiele różnorodnych instrukcji i przewodników, część z nich napisana jest z punktu widzenia początkujących użytkowników. Oficjalny podręcznik GNU emacs dostępny jest na stronie <http://www.gnu.org/software/emacs/manual/>. Można go zamówić również w formie książkowej. Dodatkowe informacje można też znaleźć na stronie wiki: <http://www.emacswiki.org/cgi-bin/wiki>.

Używanie programu gedit

Po omówieniu dwóch klasycznych edytorów tekstowych dla systemów Unix i Linux oraz stojącej za nimi filozofii, warto wspomnieć, że w Ubuntu dostępne są także inne edytory, których można używać bez lekcji historii Uniksa oraz przyrzekania dożgonnej lojalności Richardowi Stallmanowi czy Billemu Joyowi. Oba środowiska, GNOME i KDE, oferują łatwe w użytkowaniu edytory graficzne o nazwach gedit i kedit. Na rysunku 7.16 przedstawiam edytor gedit, w którym edytowany jest ten sam plik, który uprzednio otwierany był za pomocą `vi` i `emacs`a, czyli tekst niniejszego rozdziału.



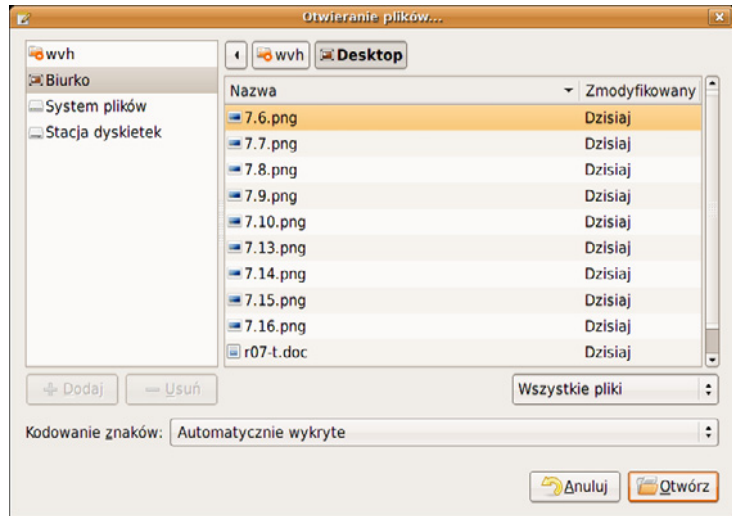
Rysunek 7.16. Edytor tekstu gedit

Edytor gedit jest całkowicie ukierunkowany na obsługę za pomocą myszy oraz menu, używa się w nim standardowych dla większości edytorów graficznych skrótów klawiszowych. Oto kilka podstawowych:

- ♦ *Ctrl+C* — kopiowanie zaznaczonego tekstu,
- ♦ *Ctrl+N* — otwarcie nowego pliku,
- ♦ *Ctrl+S* — zapisanie bieżącego pliku,
- ♦ *Ctrl+V* — wklejenie skopiowanego lub wyciętego tekstu,
- ♦ *Ctrl+X* — wycięcie zaznaczonego tekstu.

Gedit umożliwia również korzystanie z programu za pomocą wygodnych i bardziej tradycyjnych okien dialogowych, dzięki którym można zapisywać pliki, wybierać i otwierać oraz tworzyć nowe pliki i katalogi itd. Na rysunku 7.17 przedstawiam okno dialogowe uruchamiane za pomocą polecenia z menu *Plik/Otwórz*.

Rysunek 7.17.
*Okno dialogowe
Otwieranie plików
w programie gedit*



Jak w przypadku większości aplikacji środowiska GNOME, również dla edytora gedit dostępna jest obszerna dokumentacja dostępna z menu *Pomoc/Zawartość*.

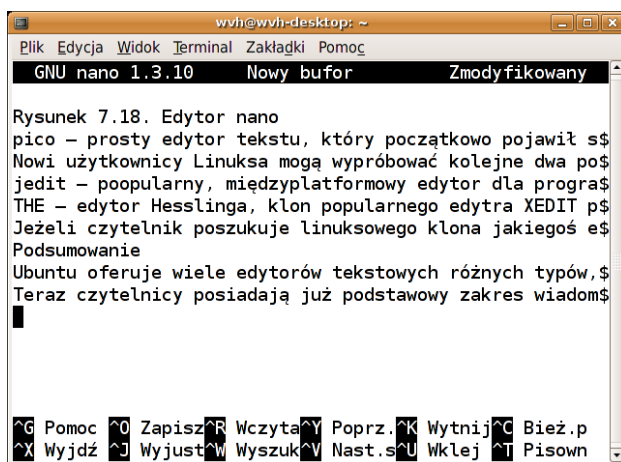
Inne edytory tekstowe w Ubuntu

W Ubuntu można skorzystać z kilku innych edytorów. Oto te, które są domyślnie instalowane w systemie:

- ♦ *ed* — prosty, wierszowy edytor inspirowany oryginalnym edytorem systemu Unix,
- ♦ *ex* — kolejny wierszowy edytor mający wiele wspólnego z *vimem*; używając edytora *ex*, można przełączyć się do pełnoekranowego trybu *vima* po wpisaniu polecenia *visual*,
- ♦ *nano* — ulepszony klon edytora *pico*, dostępny na licencji GPL (odmiennie niż *pico*); jedną z najprzyjemniejszych opcji dostępnych w obu programach (*nano* i *pico*) jest to, że podczas pracy na dole ekranu stale jest wyświetlany pasek informacyjny z najczęściej używanymi poleceniami; ponieważ program jest całkowicie

„niegraficzny”, należy go uruchamiać z terminala GNOME lub okna xterm; na rysunku 7.18 widoczny jest nano podczas edytowania niniejszego rozdziału,

Rysunek 7.18.
Edytor nano



- ♦ pico — prosty edytor tekstu, który początkowo pojawił się w systemach Unix jako część systemu do obsługi poczty elektronicznej Pine. W Linuksie jest to już tylko dowiązanie symboliczne do edytora nano, wygląda dokładnie tak samo jak na rysunku 7.18.

Nowi użytkownicy Linuksa mogą wypróbować kolejne dwa popularne edytory, które można skompilować i uruchamiać w Ubuntu:

- ♦ jedit — popularny, międzyplatformowy edytor dla programistów; więcej informacji można znaleźć na stronie www.jedit.org,
- ♦ THE — edytor Hesslinga, klon popularnego edytora XEDIT przeznaczonego dla komputerów IBM typu *mainframe*, XEDIT obsługuje różne dystrybucje Linuksa oraz język skryptowy REXX; więcej informacji można znaleźć na stronach <http://www.lightlink.com/hessling/> i http://sourceforge.net/project/showfiles.php?group_id=29648.

Jeżeli czytelnik poszukuje linuksowego klona jakiegoś edytora, warto sprawdzić <http://texteditors.org/cgi-bin/wiki.pl?EditorIndex>. Wymieniono na niej wiele różnych edytorów dla różnych systemów operacyjnych (włączając Linux) i podano wiele przydatnych informacji.

Podsumowanie

Ubuntu oferuje wiele edytorów tekstowych różnych typów, spośród których każdy użytkownik znajdzie model odpowiedni dla siebie. Dla tych, którzy korzystali już z Linuksa czy innych uniksopodobnych systemów, przeznaczony jest vim (klon edytora vi) oraz GNU emacs, które można łatwo zainstalować w systemie przy użyciu wskazówek zamieszczonych w rozdziale 20., zatytułowanym „Dodawanie, usuwanie i aktualizacja oprogramowania”. W niniejszym rozdziale szczegółowo omówiłem, jak korzystać z vi i emacsa,

jak za ich pomocą wykonać większość standardowych dla edytorów zadań oraz dostosować je do własnych preferencji. Zamieściłem tu też krótkie wprowadzenie do graficznego edytora gedit instalowanego domyślnie w systemie. Na końcu podałem informacje na temat innych edytorów dostępnych w Ubuntu, a także wskazówki, gdzie szukać kolejnych.

Teraz czytelnicy posiadają już podstawowy zakres wiadomości dotyczący korzystania z Ubuntu zarówno w trybie graficznym, jak i za pomocą narzędzi linii poleceń. Zostały omówione edytory tekstowe, podstawowe narzędzia systemowe. Teraz nadszedł czas na aplikacje z graficznym interfejsem, z którymi użytkownik będzie pracował na co dzień. W rozdziale 8. opiszę pakiet Evolution, potężny program do obsługi poczty elektronicznej i zarządzania informacjami osobistymi, przeznaczony dla środowiska GNOME.

Rozdział 8.

Obsługa poczty elektronicznej za pomocą programu Evolution

W tym rozdziale:

- ♦ Konfigurowanie programu pocztowego
- ♦ Wysyłanie i odbieranie poczty
- ♦ Organizowanie poczty
- ♦ Filtrowanie wiadomości
- ♦ Walka ze spamem

Programy do obsługi poczty elektronicznej to, zaraz po przeglądarkach internetowych, kolejne aplikacje, bez których trudno wyobrazić sobie korzystanie z internetu. I podobnie jak w przypadku przeglądarek, istnieje wiele różnorodnych programów, zwłaszcza z zakresu otwartego oprogramowania. W odróżnieniu od stron internetowych, w których język HTML jest powszechnie wykorzystywanym standardem, w poczcie elektronicznej wykorzystano wiele różnorodnych protokołów do wysyłania i odbierania poczty. Dlatego właśnie czasem wybór programu do obsługi poczty jest uzależniony od warunków technicznych oferowanych przez dostawcę usług internetowych czy też miejsca, w którym korzysta się z połączenia z siecią.

Najprostszym rozwiązaniem problemu „jaki protokół, taki klient” jest użycie programu, który może obsłużyć wszelkie znane współczesnemu człowiekowi protokoły pocztowe. Ubuntu zawiera program pocztowy o nazwie Evolution, który spełnia te oczekiwania. Evolution to klient pocztowy przeznaczony dla środowiska GNOME, pierwotnie zaprojektowany przez firmę Ximian, wykupioną przez firmę Novell, której główny produkt

SUSE Linux związany był ze środowiskiem KDE¹. Cała historia świadczy o tym, że Novell potrafi rozpoznać dobre oprogramowanie, kiedy się na nie natknie.

Evolution może obsłużyć serwery pocztowe używające wszystkich popularnych protokołów pocztowych, włącznie z POP, IMAP, IMAP4rev1, Microsoft Exchange, Novell GroupWise oraz kilku takich, których użytkowników można znaleźć tylko za pomocą Google'i (tam też znajdują się powody, dla których ich używają). Biorąc pod uwagę pocztę wychodzącą, mogę stwierdzić, że Evolution potrafi skomunikować się z każdym serwerem SMTP (ang. *Simple Mail Transport Protocol*) oraz standardowym serwerem pocztowym sendmail działającym lokalnie.

Jednak Evolution to nie tylko wieloprotokołowy program pocztowy, to nawet coś więcej niż program do zarządzania osobistymi informacjami, dla którego odbieranie i wysyłanie poczty to tylko jeden z aspektów działania. Evolution oferuje również zintegrowane usługi:

- ♦ zarządzania kontaktami przechowywanymi lokalnie za pomocą LDAP (ang. *Lightweight Directory Access Protocol*) lub na serwerach Microsoft Exchange,
- ♦ zarządzania listami zadań przechowywanymi lokalnie lub na serwerach Microsoft Exchange,
- ♦ zarządzania kalendarzami przechowywanymi lokalnie, na serwerach Microsoft Exchange i na stronach internetowych.

Być może użytkownik zauważył już pewien wątek pojawiający się często w tym wprowadzaniu — zgodność z Microsoft Exchange i usługami oferowanymi przez te serwery. Może się to podobać lub nie, ale jest faktem, że wiele firm korzysta z oprogramowania Microsoftu do obsługi planowania spotkań, operowania na rozkładach zajęć współpracowników itd. Część serwerów Microsoft Exchange korzysta po prostu ze specjalnego typu formatu IMAP, ale kalendarze i listy zadań tam dostępne są oparte już tylko na specyficznym formacie Microsoft. Dlatego właśnie interoperacyjność oferowana przez Evolution jest tak imponująca.

W niniejszym rozdziale skupię się na aspektach programu związanych z obsługą poczty elektronicznej. Objasnię, jak skonfigurować Evolution i używać go, począwszy od zakładania konta, czytania, wysyłania, skończywszy na organizowaniu i filtrowaniu poczty; omówię też możliwości oferowane przez wirtualne foldery (unikalna funkcja dostępna w Evolution). Informacje na temat dodatkowych możliwości programu obejmujących zarządzanie informacjami można znaleźć w ostatnim podrozdziale „Dodatkowe źródła informacji o Evolution”.

Uruchamianie programu Evolution

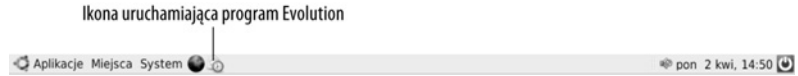
Program można uruchomić na jeden z wielu sposobów:

- ♦ klikając ikonę programu umieszczoną na górnym panelu GNOME (patrz rysunek 8.1), jest to najłatwiejsza i najczęściej wykorzystywana metoda uruchamiania programu,
- ♦ wybierając odpowiednią pozycję z menu *Aplikacje/Internet*,

¹ W kolejnych wersjach SUSE jako podstawowe środowisko zaczęto wykorzystywać GNOME — *przyp. tłum.*

Rysunek 8.1.

Ikona programu Evolution umieszczona na panelu GNOME



- ♦ wpisując polecenie `evolution` w terminalu GNOME lub oknie xterm i wciskając klawisz *Enter*.

Środowisko GNOME jest również skonfigurowane w taki sposób, że jeżeli użytkownik kliknie adres URL znajdujący się na stronie internetowej wyświetlanej w przeglądarce, dla którego przypisane jest wysyłanie wiadomości (znane pod nazwą *mailto*), wówczas również zostanie uruchomiony program Evolution lub inna aplikacja służąca do obsługi poczty elektronicznej.



Podczas uruchamiania Evolution może zostać wyświetlony komunikat informujący, że ta wersja jest jeszcze nieoficjalnym wydaniem. To, czy taka informacja się pojawi, zależy od wersji Evolution, którą programiści umieścili w Ubuntu, oraz od zainstalowanych uaktualnień. Jednym ze wspaniałych aspektów Ubuntu jest fakt, że zawsze otrzymuje się najnowsze i najlepsze oprogramowanie. Łącząc to z faktem, że oprogramowanie zamieszczane w Ubuntu jest dokładnie testowane przez programistów i społeczność, można spokojnie zignorować ten komunikat lub potraktować go jako formalność. Aby komunikat ten nie pojawiał się w przyszłości, wystarczy zaznaczyć odpowiednią opcję.

W następnym podrozdziale omówię kreator ustawień Evolution — program, który ułatwia użytkownikom przebrnięcie przez proces tworzenia pierwszego konta pocztowego w Evolution.

Kreator ustawień Evolution

Podczas pierwszego uruchomienia programu Evolution zostanie wyświetlone okno kreatora ustawień (patrz rysunek 8.2), który pomoże użytkownikowi przebrnąć przez proces tworzenia pierwszego konta pocztowego w programie.

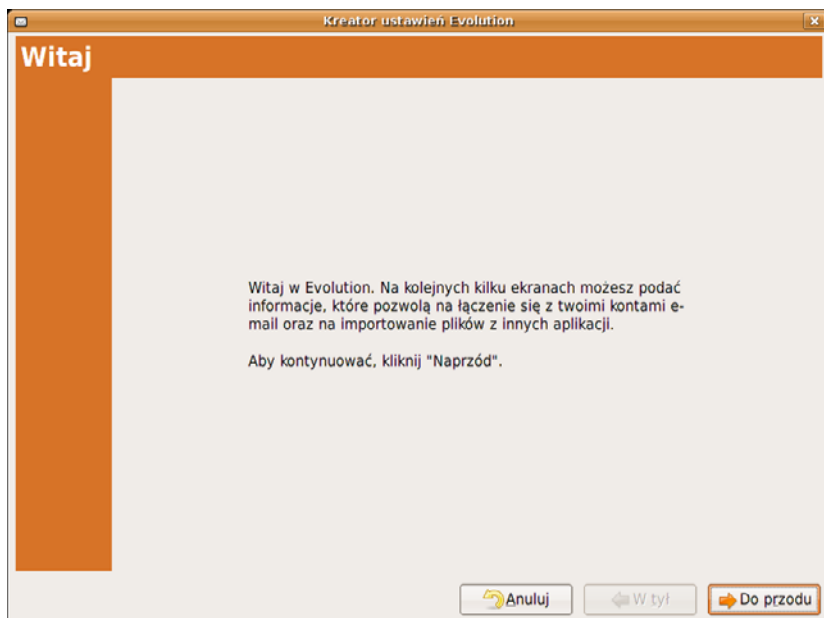
Jeszcze jedna informacja, tak na wszelki wypadek. Kreator ustawień Evolution konfiguruje program Evolution do odbierania i wysyłania poczty z już istniejącego konta; nie tworzy konta jako takiego. I tak, chociaż używam Ubuntu do odbierania i wysyłania poczty elektronicznej, to moje konto pocztowe zlokalizowane jest gdzieś na serwerze zewnętrznego dostawcy, na którym przechowywana jest domena. Za pomocą kreatora ustawień można wskazać, gdzie program ma wyszukiwać przychodzące wiadomości, a także przez jaki serwer mają być wysyłane wiadomości wychodzące. W moim przypadku komputer, na którym działa program Evolution, jest oddzielony od komputera, przez który przechodzą wysyłane i otrzymywane wiadomości.



Oczywiście, kiedy serwer pocztowy jest uruchomiony na tym samym komputerze, na którym działa Evolution, wówczas konto w Ubuntu może też służyć jako oficjalne konto pocztowe. Więcej informacji na ten temat zamieszczam w rozdziale 27., „Konfiguracja serwera poczty”.

Rysunek 8.2.

Kreator ustawień Evolution jest włączany podczas pierwszego uruchomienia programu



Ponieważ powyższe rozróżnienie może być nieco mylące, można o nim myśleć jak o otrzymywaniu i odczytywaniu wiadomości: konto pocztowe (zwykle znajduje się ono gdzieś na serwerze dostawcy usług internetowych) pozwala na odbieranie poczty, natomiast Evolution umożliwia jej odczytywanie, przechowywanie na lokalnym komputerze, usuwanie itd. Używając oficjalnego języka, można powiedzieć, że taka jest właśnie różnica pomiędzy MTA (ang. *Mail Transfer Agent*), MDA (ang. *Mail Delivery Agent*) i MUA (ang. *Mail User Agent*). MDA, ulokowany na serwerze dostawcy usług internetowych, pobiera wiadomości poprzez MTA i umieszcza je w elektronicznych skrzynkach pocztowych użytkowników. Następnie MUA (czyli w tym przypadku Evolution) pobiera wiadomości ze skrzynek i umieszcza na komputerze lokalnym (więcej informacji na ten temat zamieszczam w rozdziale 27.).

Po wyświetleniu okna widocznego na rysunku 8.2 należy przejść do następnego etapu konfigurowania programu, klikając przycisk *Do przodu*. Zostanie wyświetlone okno widoczne na rysunku 8.3.

Pole *Full name* zostanie wypełnione automatycznie, program pobierze dane konta użytkownika, na którego koncie został uruchomiony. Oczywiście, zawartość tego pola można zmienić, jeżeli użytkownik uzna, że potrzebuje innej nazwy do prowadzenia korespondencji. Krytyczne znaczenie ma pole *Adres e-mail*, musi tu zostać wpisany ten, którym ma być skojarzony z kontem. Jak wspominałem wcześniej, zwykle jest to adres podany przez dostawcę usług internetowych (lub ustalony podczas zakładania konta), a służy do obsługi poczty.

Pole *Ustaw konto jako domyślne* jest zaznaczone od razu, ponieważ kreator ustawień Evolution jest uruchamiany wówczas, kiedy system odkryje, że jest włączany po raz pierwszy. Konto domyślne jest jedynym kontem, które będzie używane w systemie, przynajmniej dopóty, dopóki użytkownik nie doda następnych.

Rysunek 8.3.
Określanie
adresu poczty
elektronicznej

W polu *Odpowiedź do* można ustalić, na jaki adres będą wysyłane odpowiedzi na listy, ale wypełnienie tego pola nie jest konieczne. Podobnie rzecz ma się z wpisem w polu *Organizacja*; umieszczenie tam jakiejś nazwy spowoduje, że w nagłówkach wysyłanych wiadomości będą umieszczane informacje zaczerpnięte również z tego pola. Większość użytkowników nie zauważa tych informacji, ale mogą być przydatne np. do filtrowania wiadomości (to zagadnienie omawiam w dalszej części rozdziału) i określania, z jakiej lokalizacji zostały wysłane. Zwykle umieszczam tam zapis w rodzaju „Komputer domowy Billa”, dzięki czemu łatwo można rozpoznać, która wiadomość została wysłana z domu.

Po uzupełnieniu informacji w tym oknie należy przejść do następnego (patrz rysunek 8.4), klikając przycisk *Do przodu*.

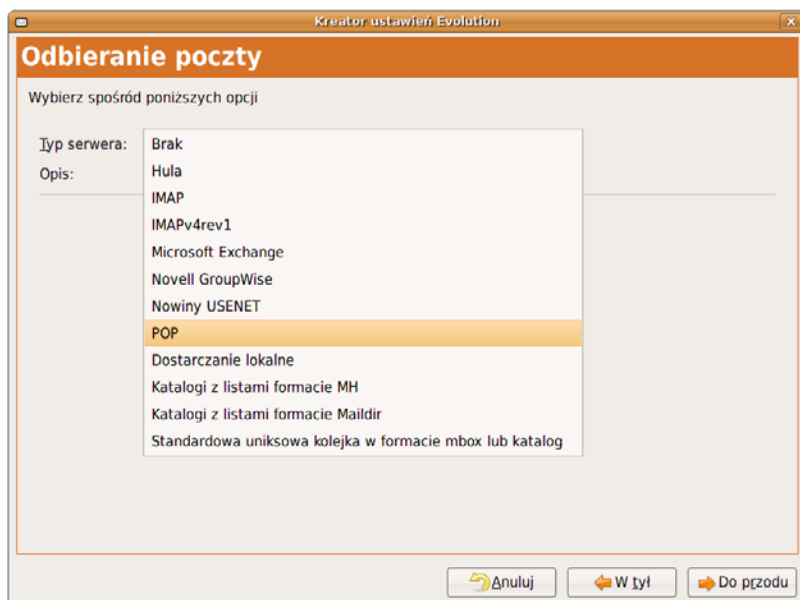
Kiedy to okno zostanie wyświetlone po raz pierwszy, będzie zupełnie puste, a na liście rozwijanej *Typ serwera* zobaczymy pozycję *Brak*. Dopiero po wskazaniu odpowiedniego (takie informacje można uzyskać od dostawcy usług internetowych lub podczas zakładania konta na wybranym portalu) typu serwera w polu *Opis serwera* zostaną wyświetlone dodatkowe informacje na jego temat. Obecnie najpopularniejsze wykorzystywane typy to POP (ang. *Post Office Protocol*) oraz jego specjalna wersja POP3, w której można znaleźć pewne rozszerzenia niewystępujące w wersji podstawowej. Drugim protokołem pod względem popularności jest IMAP, obsługujący również program Microsoft Exchange, w którym wykorzystywana jest wersja tego protokołu.

Po wybraniu z listy odpowiedniego typu serwera zostanie wyświetlona lista jego właściwości. Na rysunku 8.5 widać opcje dostępne dla serwerów POP.

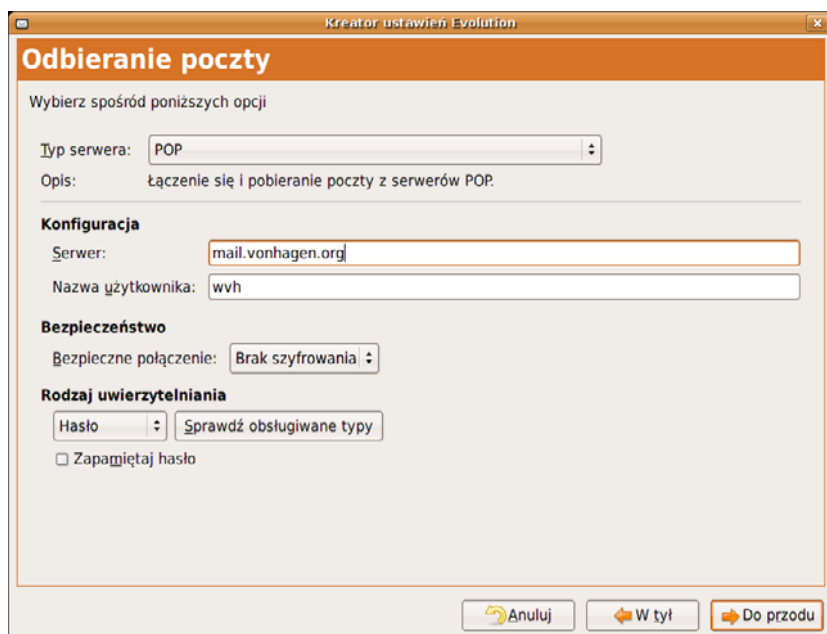
Tutaj należy kolejno wprowadzić nazwę serwera oraz nazwę użytkownika wykorzystywaną do logowania się na koncie. Warto zwrócić uwagę, że czasami nazwa użytkownika używana do logowania się w systemie poczty różni się od nazwy podawanej jako pierwszy

Rysunek 8.4.

Wybór serwera
poczty przychodzącej
(MTA)

**Rysunek 8.5.**

Określanie
parametrów
serwera poczty
przychodzącej



człon adresu e-mail. Ma to na celu zapewnienie unikalności podawanych adresów oraz możliwość obsługi wielu domen. Nazwa używana do zalogowania się na serwerze pocztowym może wyglądać następująco: *wvh@przyklad.org*, a adres pocztowy tak: *vonhagen@przyklad.org*.

Następnie należy określić zasady bezpieczeństwa używane podczas logowania się do poczty. Zawsze warto skorzystać z jakiegoś szyfrowania podczas połączenia (opcja *Bezpieczne*

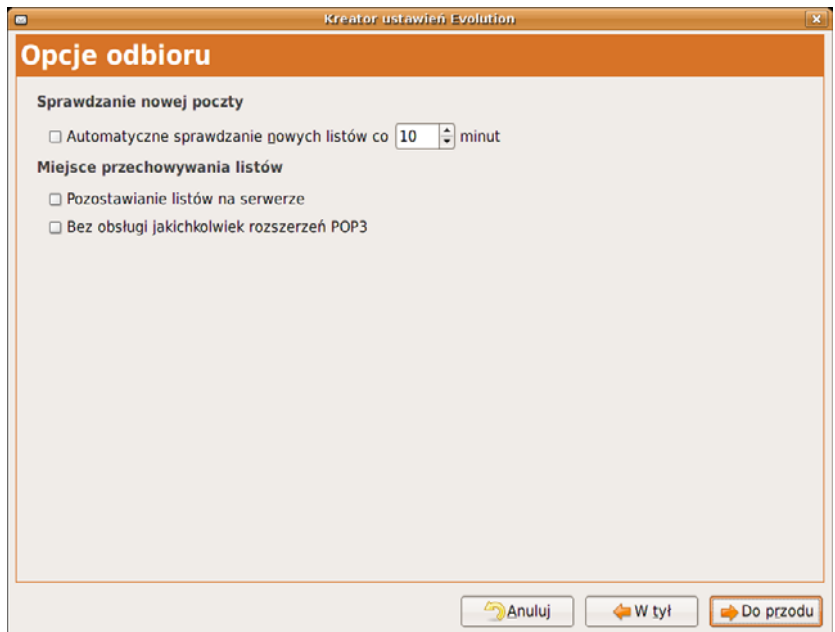
połączenie). W przypadku *Rodzaju uwierzytelniania* warto użyć przycisku *Sprawdź obsługiwane typy*, dzięki czemu program sam nawiąże łączność z serwerem (o ile podana będzie prawidłowa nazwa) i ustali, jakiego rodzaju uwierzytelnianie jest dostępne. Po sprawdzeniu zostaną przekreślone opcje, które nie będą dostępne. W większości przypadków dostępna będzie opcja weryfikacji za pomocą hasła, dzięki tej możliwości użytkownik może sprawdzić, czy serwer obsługuje bardziej zaawansowane systemy, takie jak Kerberos czy GSSAPI.

Jako opcję można zaznaczyć pole *Zapamiętaj hasło*, bo nie trzeba będzie go podawać za każdym razem podczas logowania do systemu poczty. Ale nie zalecam korzystania z tej możliwości, ponieważ w takiej sytuacji każdy, kto będzie miał dostęp do konta, będzie mógł odczytać wszystkie wiadomości, co może nie być najlepszym rozwiązaniem.

Po zakończeniu wprowadzania ustawień dla serwera należy przejść do kolejnego etapu konfigurowania programu Evolution. Po kliknięciu przycisku *Do przodu* zostanie wyświetlone kolejne okno dialogowe (patrz rysunek 8.6).

Rysunek 8.6.

Ustawianie dodatkowych parametrów serwera poczty przychodzącej



Opcje dostępne w oknie dialogowym widocznym na rysunku 8.6. pozwalają określić, w jaki sposób program będzie współdziałał z serwerem poczty przychodzącej. Niezależnie od typu serwera, użytkownik może ustalić, jak często program ma sprawdzać, czy w skrzynce pojawiły się nowe wiadomości. Po skonfigurowaniu sposobu współpracy z serwerem poczty użytkownik może określić, czy program ma usuwać kopie wiadomości. W ramce zatytułowanej „Krótki przewodnik po protokołach POP i IMAP” podałem informację, że protokoły, takie jak POP, służą do krótkoterminowego przechowywania plików, natomiast takie jak IMAP mogą służyć do długotrwałego przechowywania korespondencji. W oknie użytkownik może też ustalić, czy program ma korzystać z dodatkowych możliwości oferowanych przez protokół POP3.

Krótki przewodnik po protokołach POP i IMAP

Kluczowa różnica między protokołami POP i IMAP dotyczy miejsca, w którym są przechowywane wiadomości. W protokole POP korespondencja zwykle jest pobierana z serwera i przechowywana na komputerze, na którym działa program pocztowy. Jest to bardzo wygodna opcja dla operatorów serwerów, ponieważ nie muszą zapewnić ogromnych ilości przestrzeni dyskowej na wiadomości użytkowników — zwykle są one pobierane na dyski komputerów i usuwane z serwera. Z kolei w IMAP wiadomości są stale przechowywane na serwerach, podczas sprawdzenia poczty program pocztowy pobiera tylko nagłówki wiadomości. Nagłówki zawierają temat wiadomości, nadawcę, datę wysłania wiadomości itp. Kiedy użytkownik wybierze wiadomość, którą chce przeczytać, wówczas jej kopia zostaje pobrana na dysk lokalny, ale podstawowy plik nadal znajduje się na serwerze IMAP. Przy zastosowaniu takiego rozwiązania ilość przestrzeni dyskowej potrzebnej na przechowywanie wiadomości będzie rosła wraz z ilością przechowywanych wiadomości.

Każde z omówionych rozwiązań ma swoje wady i zalety. Protokół POP pozwala przechowywać wiadomości na lokalnym komputerze, dzięki czemu korespondencję można przeglądać również bez połączenia z internetem. Może to również być niedogodnością, ponieważ kiedy wszystkie wiadomości są przechowywane na jednym komputerze, trudno będzie odczytać starsze, korzystając z innego komputera niż aktualnie używany do tego celu. Protokół IMAP umożliwia dostęp do poczty z dowolnego komputera, jedynym wymogiem jest działające połączenie z siecią. Większość programów pocztowych obsługujących IMAP umożliwia zsynchronizowanie lokalnej skrzynki pocztowej ze znajdującą się na serwerze, dzięki czemu można mieć dostęp do starszych wiadomości bez połączenia z siecią. Operatorzy serwerów zwykle nie znoszą użytkowników, takich jak ja, którzy przechowują na kontach archiwum wiadomości sprzed dziesięciu i więcej lat, co wymaga przeznaczania na magazynowaną pocztę sporej ilości miejsca na dyskach twardych.

W większości programów do obsługi poczty elektronicznej można znaleźć opcję umożliwiającą pozostawienie wiadomości na serwerze. Wtedy do korzystania z tego samego konta pocztowego można skonfigurować wiele klientów, ale tylko jeden z nich, zwykle znajdujący się w domu, służy do pobierania wiadomości i usuwania ich z serwera. Programy korzystające z protokołu POP zapamiętują ostatnią przeczytaną wiadomość, a zatem za każdym razem, kiedy nawiązują połączenie z serwerem, nie pobierają wszystkich wiadomości, a tylko te, które rozpoznają jako nowe. Jest to wygodny kompromis pomiędzy elastycznością IMAP-a a trwałością POP-a. Podczas podróży korzystam z komputera przenośnego, zainstalowany tam program Evolution pozwala czytać pocztę, ale nie usuwa jej z serwera, to dzieje się dopiero po podłączeniu głównego komputera, znajdującego się w domu. Minusem jest brak dostępu do starszych wiadomości podczas podróży, ale w chwilach desperacji zawsze można połączyć się z komputerem domowym, żeby sprawdzić listy, lub też skorzystać z dostępu do sieci gdzieś po drodze. Rozwiązanie powolne, ale działa.

Opcje, które można konfigurować, zależą od typu serwera poczty przychodzącej. Na rysunku 8.7 można zobaczyć konfigurację serwera korzystającego z protokołu IMAP.

Łatwo zauważyć, że w tym przypadku dostępnych jest o wiele więcej opcji konfiguracyjnych, aniżeli w przypadku serwerów POP. Ponieważ serwery IMAP służą jako główne repozytorium wiadomości, w tym oknie użytkownik może skonfigurować, jak program ma łączyć się z serwerem (np. za pośrednictwem tzw. „zabezpieczonych tuneli”), jak używać folderów znajdujących się na serwerze (zarówno tych prywatnych, jak i publicznych), czy powinien synchronizować foldery z pocztą umieszczone na komputerze lokalnym ze znajdującymi się na serwerze IMAP (kwestia tworzenia kopii lokalnych wszystkich wiadomości) i filtrować niechciane (i nie tylko) wiadomości. Filtrowanie wiadomości i walka ze spamem zostały omówione w podrozdziale „Filtrowanie wiadomości przychodzących”. Jeżeli program Evolution jest konfigurowany do korzystania z serwera IMAP, warto wówczas zaznaczyć opcję *Zastosowanie filtrów do nowych listów w INBOX na tym serwerze*,

Rysunek 8.7.

Opcje konfiguracyjne
dla serwerów IMAP

w razie konieczności zawsze można ją wyłączyć później, a jej aktywacja wyraźnie pomoże w walce z niechcianymi przesyłkami — plagą czasów współczesnych.

Po wprowadzeniu ustawień konfiguracyjnych określających, w jaki sposób program pocztowy ma współpracować z serwerem poczty przychodzącej, należy skonfigurować serwer poczty wychodzącej. Po kliknięciu przycisku *Do przodu* zostanie wyświetlone kolejne okno dialogowe (patrz rysunek 8.8).

Podczas poprzednich etapów konfiguracji programu pocztowego użytkownik ustalał, jak system ma sobie radzić z pocztą przychodzącą oraz jakie ustawienia wprowadzić dla danego serwera. Na rysunku 8.8 widoczne są parametry serwera poczty wychodzącej. Serwery poczty przychodzącej i wychodzącej mogą być zupełnie różnymi komputerami (lub procesami), tak jak zupełnie inne są wykonywane przez nie zadania.

Najważniejszą informacją umieszczaną w tym oknie jest rodzaj serwera poczty wychodzącej. Program Evolution obsługuje dwa rodzaje: SMTP (ang. *Simple Mail Transport Protocol*) oraz Sendmail (najpopularniejszy serwer pocztowy uruchamiany w systemach linuksowych i uniksowych, dostępny również w Ubuntu). Dla większości użytkowników odpowiednim wyborem będzie SMTP, zwłaszcza w przypadku korzystania z poczty oferowanej przez dostawcę usług internetowych czy na którymś z popularnych portali. Jeżeli jednak w planach jest uruchomienie własnego serwera, wówczas można wybrać Sendmaila (więcej informacji na temat konfigurowania i uruchamiania własnego serwera pocztowego można znaleźć w rozdziale 27., „Konfiguracja serwera poczty”).

Jeżeli z listy zostanie wybrany serwer Sendmail, wówczas nie potrzeba dodatkowych informacji konfiguracyjnych (ponieważ działa on na tym samym komputerze, na którym uruchomiony jest program Evolution). Jeżeli jednak wybór padnie na SMTP, należy wpisać jego nazwę, a także — w razie potrzeby — określić metodę uwierzytelniania.

Rysunek 8.8.
Konfigurowanie
opcji poczty
wychodzącej

Uwierzytelnianie jest zaimplementowane w większości współczesnych serwerów SMTP, dlatego należy zaznaczyć pole *Serwer wymaga uwierzytelnienia*, określić, czy powinno być użyte bezpieczne połączenie, oraz ustalić sposób, w jaki zostanie przeprowadzone uwierzytelnianie. W tym celu najlepiej kliknąć przycisk *Sprawdź obsługiwane typy*, wtedy zostanie przeprowadzona taka sama procedura jak dla serwera poczty przychodzącej. Podobnie jak dla serwerów IMAP i POP, użytkownik może zaznaczyć opcję *Zapamiętanie hasła*, ale nie jest to zalecane, ponieważ w takiej sytuacji każdy, kto ma dostęp do konta w systemie, będzie mógł również odczytać pocztę elektroniczną.

Po odpowiednim skonfigurowaniu serwera poczty wychodzącej należy kliknąć przycisk *Do przodu*. Jeżeli użytkownik odkryje jakiś błąd we wprowadzonych danych (np. poczta nie będzie pobierana z serwera), nie ma powodu do niepokoju, zawsze można je poprawić (więcej informacji na ten temat zamieszczam poniżej). Następne okno konfiguracji programu widać na rysunku 8.9.

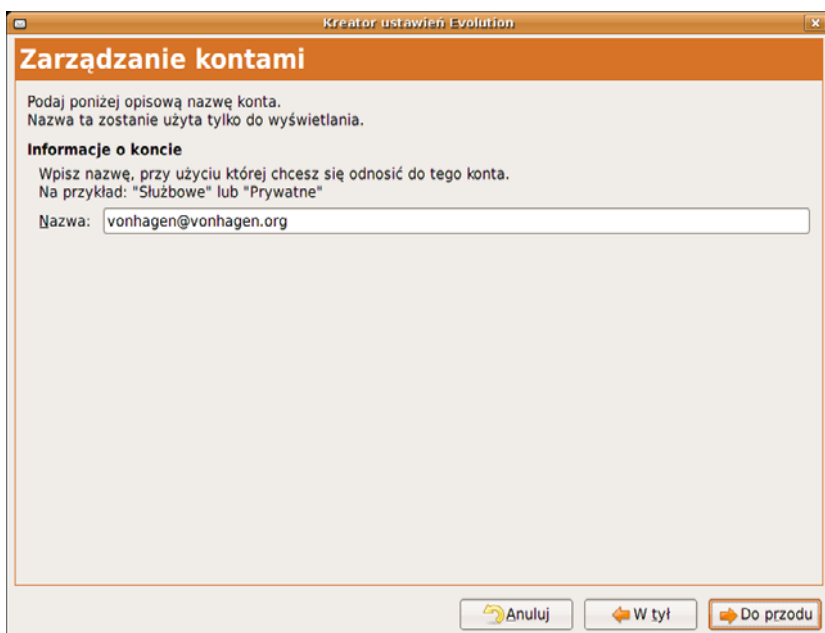
To już prawie wszystko. W oknie widocznym na rysunku 8.9 należy wprowadzić nazwę dla konta poczty elektronicznej, które właśnie zostało skonfigurowane. Będzie ona wykorzystywana w programie Evolution do identyfikowania zbioru wiadomości oraz ustawień, należy z niej korzystać również wówczas (wybierając z okna *Ustawienia Evolution*), kiedy użytkownik będzie chciał dokonać zmian w zapisanej konfiguracji. Domyślnie proponowana jest nazwa taka sama jak adres poczty elektronicznej.

Po wprowadzeniu odpowiedniego zapisu należy kliknąć przycisk *Do przodu*, zostanie wówczas wyświetlone kolejne okno konfiguracyjne programu (patrz rysunek 8.10).

Można tu określić strefę czasową odpowiednią dla komputera, na którym uruchamiany będzie program Evolution; na tej podstawie do wiadomości wychodzących dołączana

Rysunek 8.9.

Wprowadzanie
nazwy dla konta
poczty elektronicznej

**Rysunek 8.10.**

Określanie strefy
czasowej



będzie informacja o godzinie wysłania. Domyślnie wyświetlany jest czas uniwersalny koordynowany (UTC — ang. *Universal Time Coordinated*), co oznacza, że program przyjął, iż komputer stosuje właśnie ten, niezależny od położenia geograficznego, czas. Większość użytkowników woli jednak ustawiać zegary na czas lokalny, co można zrobić, wskazując odpowiednią lokalizację na wyświetlanej mapie. Wystarczy wskazać kursorem

położenie najbliższe temu, w którym znajduje się komputer, i kliknąć przyciskiem myszy. Podczas przesuwania kursora nad mapą w polu *Zaznaczenie* będą wyświetlane miasta odpowiadające pozycji wskaźnika. Kliknięcie nad dowolną częścią mapy spowoduje jej powiększenie (patrz rysunek 8.11).

Rysunek 8.11.
Powiększenie mapy



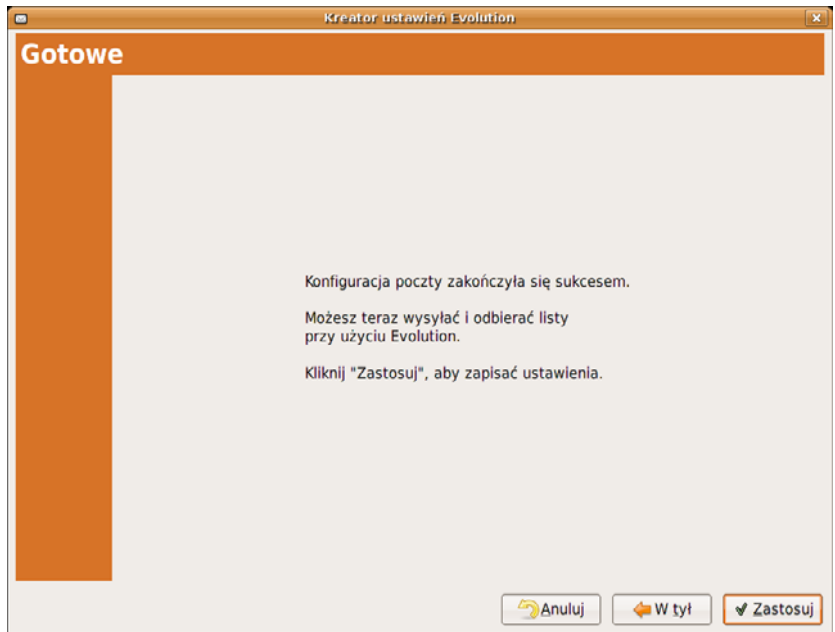
Na mapie stref czasowych miasto, w którym znajduje się użytkownik, może nie być uwzględnione. Wtedy należy wybrać położne najbliżzej (i znajdujące się w tej samej strefie czasowej), a następnie kliknąć przycisk *Do przodu*, zostanie wówczas wyświetlone kolejne okno dialogowe widoczne na rysunku 8.12.

Jak można się zorientować po zawartości okna widocznego na rysunku 8.12, określenie strefy czasowej było ostatnim etapem konfigurowania konta pocztowego w programie Evolution. Aby zapisać wprowadzone zmiany, należy kliknąć przycisk *Zastosuj*. Od tej chwili można odbierać i wysyłać pocztę w programie, więcej informacji na ten temat podaję w dalszej części rozdziału.



Evolution, odmiennie niż większość programów pocztowych, umożliwia jednocześnie wysyłanie i odbieranie poczty z wielu kont. Każdy adres poczty elektronicznej ma przypisane serwery poczty przychodzącej i wychodzącej. Kolejne konta pocztowe można konfigurować, korzystając z okna dialogowego *Ustawienia Evolution*, które można wywołać z menu *Edycja*. Warto zwrócić uwagę, że wiadomości ze wszystkich kont są przechowywane w tym samym folderze, dla każdego z nich nie jest tworzona skrzynka odbiorcza i osobny zestaw. Poniżej zamieszczam informacje, jak rozwiązać ten problem za pomocą filtrowania wiadomości lub wirtualnych folderów.

Rysunek 8.12.
Gratulacje!



Wysyłanie i odbieranie poczty elektronicznej

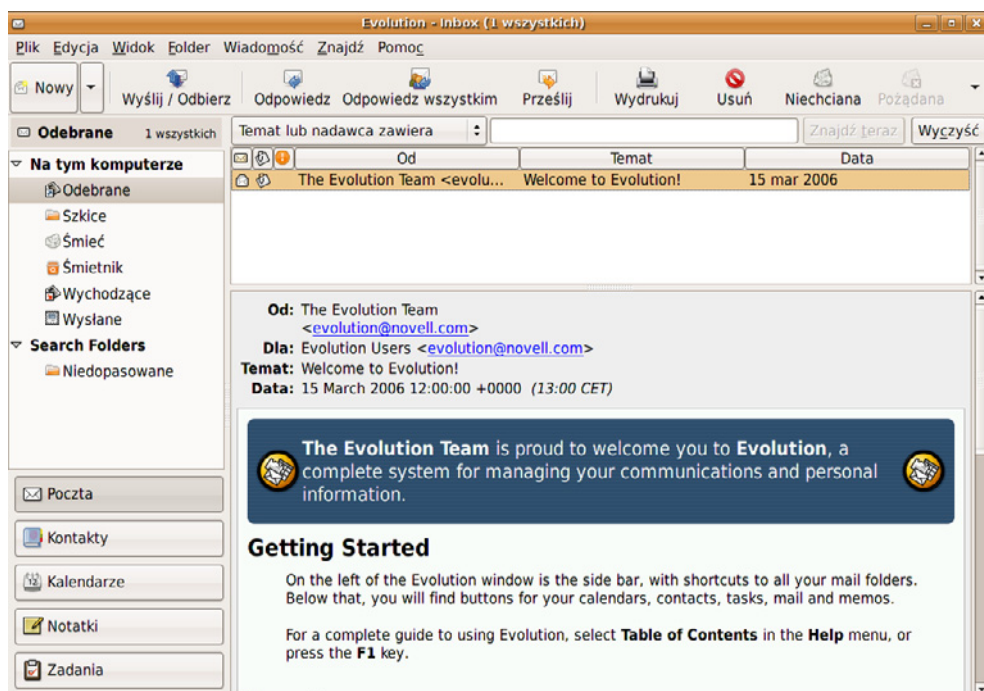
Na rysunku 8.13 widać okno wyświetlane tuż po uruchomieniu programu Evolution (już po przebrnięciu przez kreator ustawień Evolution). Czas przeczytać jakąś wiadomość!

Aby pobrać wiadomości, należy kliknąć przycisk *Wyślij/Odbierz*, znajdujący się na pasku narzędziowym. Program wyświetli komunikat z prośbą o podanie hasła, po czym nawiąże połączenie z serwerem i rozpocznie pobieranie poczty. Podczas tego procesu (a także wysyłania wiadomości) wyświetlane będzie okno informujące o postępie wykonywanej czynności (patrz rysunek 8.14).

Po odebraniu wiadomości znajdujących się na serwerze oraz wysłaniu tych, które były zakolejkowane, program zaktualizuje informacje na temat stanu wszystkich używanych folderów poczty. Aby przeczytać nowe wiadomości, należy wybrać folder *Odebrane* znajdujący się w panelu nawigacyjnym po lewej stronie, a następnie zaznaczyć odpowiednią wiadomość (patrz rysunek 8.15).

Wszelkie foldery znajdujące się w panelu nawigacyjnym, których nazwy wyświetlane są pogrubioną czcionką, zawierają wiadomości oznaczone jako nieprzeczytane. Warto zwrócić uwagę, że program Evolution wyświetla informację o liczbie takich wiadomości obok nazwy folderu w panelu nawigacyjnym, w którym się znajdują.

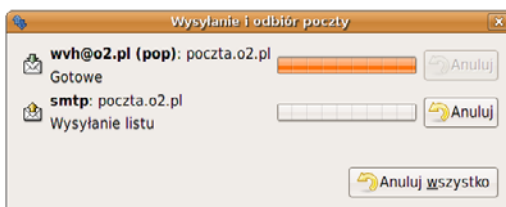
Za pomocą przycisków umieszczonych na pasku narzędziowym programu Evolution można wykonać większość najczęściej wykorzystywanych funkcji. Oto one.



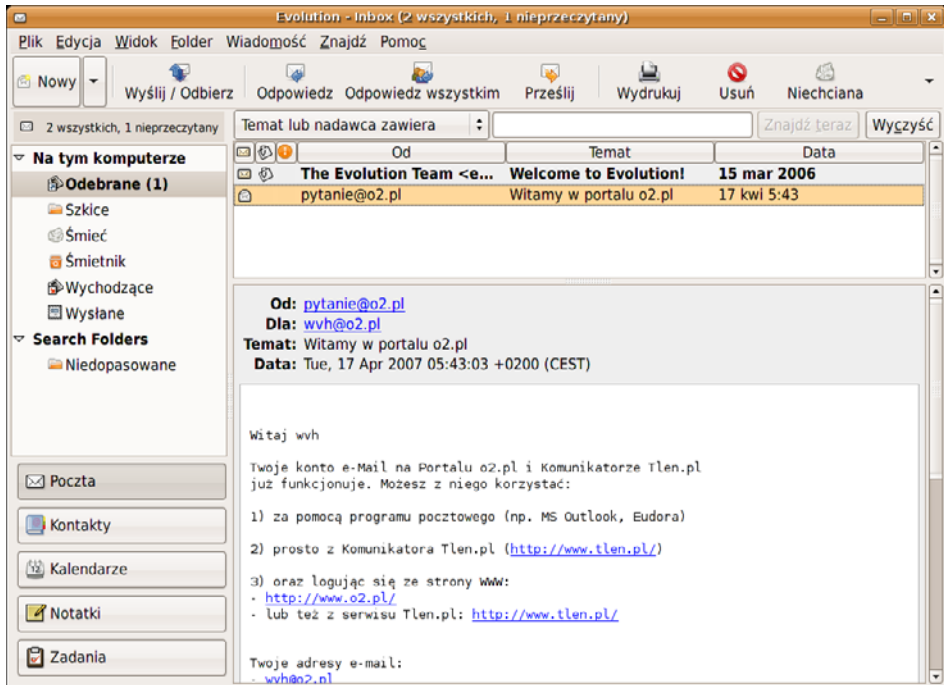
Rysunek 8.13. Główne okno programu Evolution

Rysunek 8.14.

*Informacja
o stanie procesu
pobierania
i wysłania
wiadomości*



- ♦ **Nowy** — za pomocą tego przycisku użytkownik może utworzyć nową wiadomość, dodać kontakt, termin, notatkę czy zadanie.
- ♦ **Wyślij/Odbierz** — połączenie się programu z serwerami poczty przychodzącej i wychodzącej oraz pobranie i wysłanie nowych wiadomości.
- ♦ **Odpowiedz** — tworzy nową wiadomość zaadresowaną wprost do nadawcy tej, która jest aktualnie wyświetlana.
- ♦ **Odpowiedz wszystkim** — tworzy nową wiadomość zaadresowaną wprost do nadawcy tej, która jest aktualnie wyświetlana, a także wysyła ukrytą kopię do wszystkich adresatów, znajdujących się w oryginalnej wiadomości.
- ♦ **Prześlij** — umożliwia przesłanie otrzymanej wiadomości do kolejnego adresata.
- ♦ **Wydrukuj** — pozwala wydrukować wyświetlaną wiadomość; zostaje otwarte nowe okno dialogowe, w którym użytkownik może wybrać z zainstalowanych w systemie tę drukarkę, która ma zostać użyta (dodatkowe informacje na temat



Rysunek 8.15. Wyświetlanie nowej wiadomości

konfigurowania drukarek zamieszczam w rozdziale 23., „Dodawanie sprzętu i przyłączanie urządzeń peryferyjnych”).

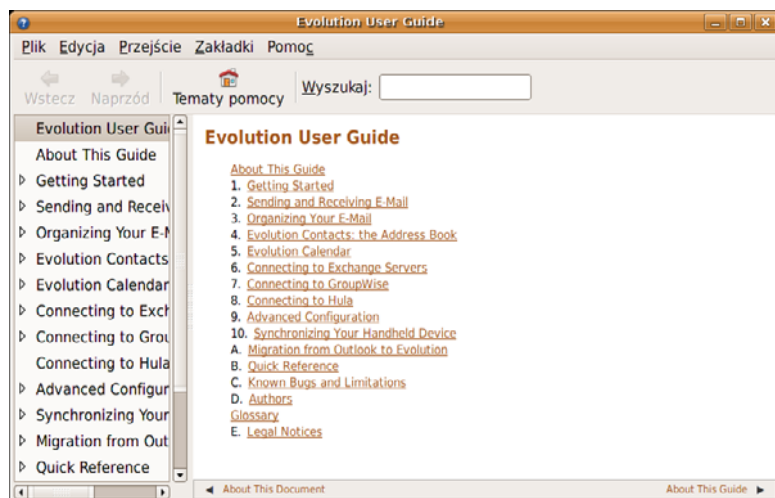
- ♦ **Usuń** — zaznacza wybraną wiadomość jako przeznaczoną do usunięcia, co powoduje usunięcie jej z bieżącej listy. Aby całkowicie usunąć wiadomość z programu, należy z menu *Plik* wybrać opcję *Opróżnij śmietnik*.
- ♦ **Niechciana** — zaznacza wybraną wiadomość jako niechcianą, co zmniejsza szansę, że w przyszłości podobne e-maile będą pojawiać się w skrzynce odbiorczej. Więcej informacji na temat radzenia sobie ze spamem można znaleźć w punkcie „Automatyczne sprawdzanie niechcianej poczty”.
- ♦ **Pożądana** — pozwala zmienić status wiadomości, która poprzednio została oznaczona jako niechciana. Więcej informacji na temat radzenia sobie ze spamem można znaleźć w punkcie „Automatyczne sprawdzanie niechcianej poczty”.

Znajdująca się na końcu paska narzędziowego strzałka skierowana w dół oznacza, że dostępne są jeszcze inne przyciski, ale monitor autora nie jest na tyle duży, aby je wszystkie wyświetlić. Żeby uzyskać do nich dostęp (są to przyciski *Anuluj*, *Poprzednia*, *Następna*), należy kliknąć strzałkę lub zmienić rozmiar okna, w którym wyświetlany jest program.

Przyciski umieszczone na pasku narzędziowym programu Evolution udostępniają większość operacji potrzebnych do korzystania z poczty. Jednak sam program to rozbudowana i wydajna aplikacja wyposażona w całe mnóstwo różnych pokręteł, przycisków i dźwigni, a ich opisanie zasługuje na osobną książkę. Na szczęście, sam program wyposażony jest w obszerny system pomocy, w którym można znaleźć informacje na temat wszystkich

możliwości programu. Aby uzyskać dostęp do pomocy, wystarczy wybrać kolejno z menu *Pomoc/Zawartość*, zostanie wówczas wyświetlone okno widoczne na rysunku 8.16.

Rysunek 8.16.
Pomoc programu
Evolution



Ponieważ książka jest poświęcona Ubuntu, a nie Evolution, w niniejszym rozdziale zamieszczam informacje, które mają tylko pomóc użytkownikowi rozpoczęcie korzystania z programu. Oprócz książek poświęconych temu programowi, w internecie można znaleźć wiele stron zawierających informacje dotyczące korzystania z tego klienta poczty elektronicznej. Poniżej zamieszczam kilka wskazówek pozwalających usprawnić korzystanie z programu i zarządzanie pocztą elektroniczną. Źródła dodatkowych informacji można znaleźć w podrozdziale „Dodatkowe źródła informacji o Evolution”.

Wysyłanie wiadomości w programie Evolution

Tworzenie nowej wiadomości w programie Evolution można rozpocząć na kilka sposobów:

- ♦ klikając przycisk *Nowy* znajdujący się na pasku narzędziowym,
- ♦ używając kombinacji klawiszy *Ctrl+Shift+M* (lub *Ctrl+N*),
- ♦ wybierając z menu *Wiadomość* opcję *Utwórz nową wiadomość*.

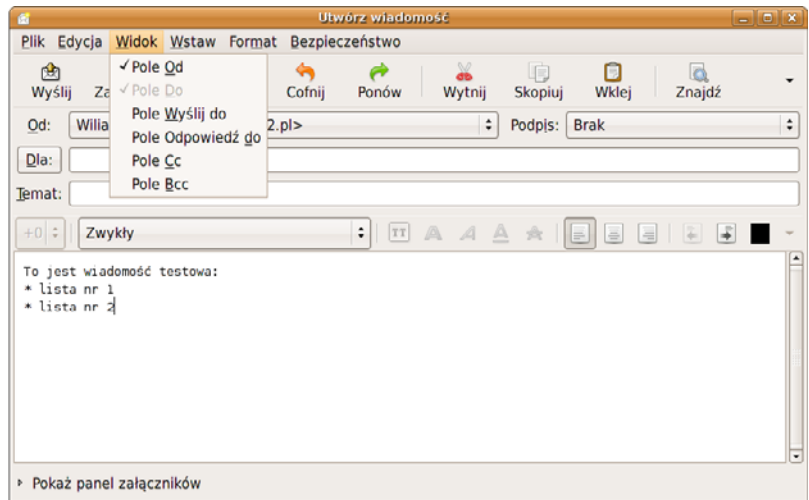
Aby odpowiedzieć na wyświetlaną wiadomość, należy:

- ♦ kliknąć przycisk *Odpowiedz*, użyć kombinacji klawiszy *Ctrl+R* lub wybrać z menu *Wiadomość/Odpowiedz nadawcy*,
- ♦ kliknąć przycisk *Odpowiedz wszystkim*, użyć kombinacji klawiszy *Shift+Ctrl+R* lub wybrać z menu *Wiadomość/Odpowiedz wszystkim*. Jeżeli użytkownik będzie jedynym odbiorcą wiadomości, wówczas efekt będzie taki sam jak w przypadku użycia opcji *Odpowiedz*.

Po wybraniu którejkolwiek z powyższych opcji zostanie wyświetlone okno dialogowe widoczne na rysunku 8.17. Jeżeli będzie to odpowiedź na otrzymaną wcześniej wiadomość, wówczas w oknie będzie również widoczna jej treść.

Rysunek 8.17.

Tworzenie wiadomości w programie Evolution



Jeżeli wyświetlana jest wiadomość wysłana na listę dyskusyjną, którą użytkownik subskrybuje, to aby odpowiedzieć wszystkim, a nie tylko osobie wysyłającej daną wiadomość, należy użyć kombinacji klawiszy *Ctrl+L* lub wybrać z menu *Odpowiedz na listę*. Aby zachować zasady netykiety, trzeba, odpowiadając na taką wiadomość, ograniczyć jej cytowanie do niezbędnego minimum.

Podczas tworzenia wiadomości, Evolution w nagłówku dla zaoszczędzenia miejsca wyświetla tylko pola *Od* i *Dla*. Często się zdarza, że użytkownik chce wysłać wiadomość również z kopią do innych osób (czy też na własne, inne konto pocztowe). Aby wyświetlić w nagłówku dodatkowe pola umożliwiające wysłanie wiadomości również do innych osób, należy przejść do menu *Widok* i wybrać inne, potrzebne pola (takie jak np. *Pole CC*, czyli kopia, oraz *Pole BCC*, czyli ukryta kopia).

Po utworzeniu wiadomości wystarczy wcisnąć przycisk *Wyślij*. E-mail zostanie umieszczony na liście przesłerek oczekujących na wysyłkę. Podczas gdy użytkownik będzie kontynuował pracę, program połączy się z odpowiednim serwerem i wyśle wiadomości tak szybko, jak to możliwe.

Konfigurowanie ustawień dla wysyłanych i odbieranych wiadomości

Do tworzenia wiadomości używane są dwa główne formaty. Pierwszy z nich to format tekstowy, jego zastosowanie oznacza, że wiadomość będzie wyglądała tak, jakby została utworzona na maszynie do pisania (jeżeli czytelnicy pamiętają takie urządzenia), czyli bez specjalnych czcionek czy formatowania. Drugim formatem jest HTML umożliwiający użycie różnych czcionek, znaków specjalnych itd. Użytkownicy zwykle preferują jeden lub drugi styl, ale to, czy adresat może odczytać wiadomość zapisaną w formacie HTML, zależy od programu używanego do obsługi poczty elektronicznej (choć potrafi to większość współczesnych klientów).



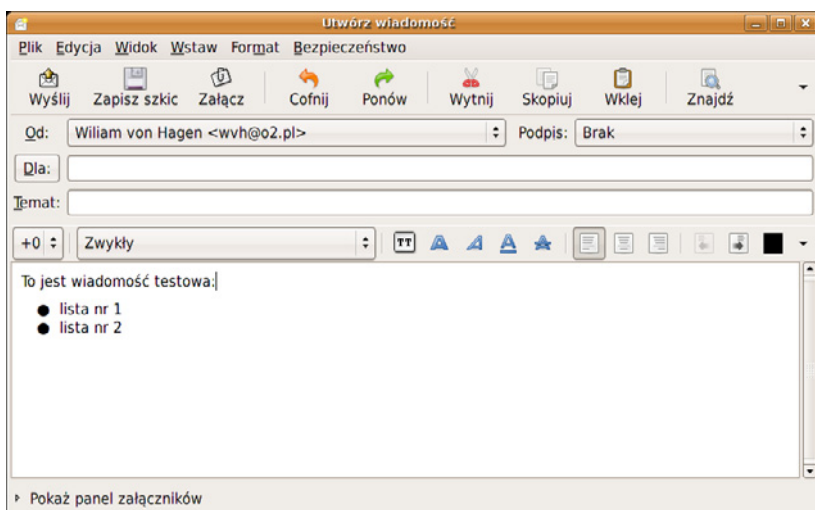
To, czy dany program może odczytać wiadomość zapisaną w formacie HTML, jest zupełnie inną kwestią od tej, czy należy takie wiadomości wysyłać. Wiele użytkowników nie chce otrzymywać lub wręcz nie przyjmuje wiadomości HTML (dotyczy to zwłaszcza list dyskusyjnych).

Jak widać na rysunku 8.17, podczas pisania pierwszej wiadomości program zastosował zwykły tekst. Użytkownik może włączyć format HTML na jeden z dwóch sposobów.

- ♦ Jeżeli sformatowana ma być tylko bieżąca wiadomość, wówczas należy wybrać z menu okna danej wiadomości polecenie *Format/HTML*. Na rysunku 8.18 przedstawiam tę samą wiadomość, którą widać na rysunku 8.17, ale wyświetlaną w HTML. Podczas tworzenia listów w tym formacie dostępne są również inne przyciski, za których pomocą można formatować wielkość czcionki, zastosowane style, odstępy itd.

Rysunek 8.18.

Wiadomość
sformatowana
w HTML



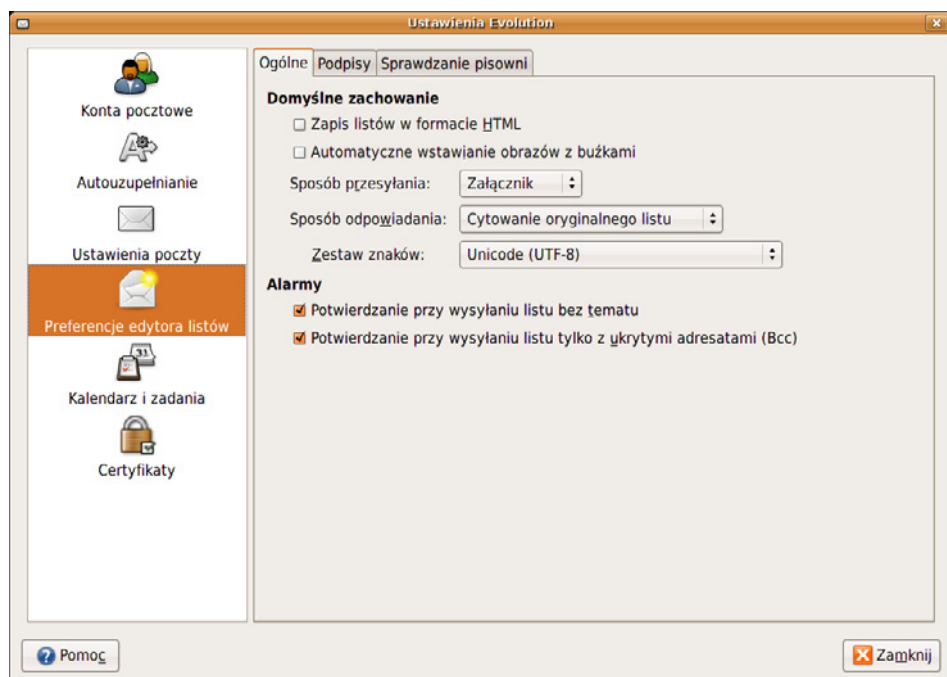
- ♦ Jeżeli wszystkie wysyłane wiadomości mają być formatowane w HTML, wówczas należy wybrać tę opcję w oknie dialogowym *Ustawienia Evolution*.

Opcje dotyczące formatowania wiadomości przychodzących i wychodzących można ustawić w oknie *Ustawienia Evolution* w działach *Ustawienia poczty* oraz *Preferencje edytora listów*. Aby wyświetlić okno *Ustawienia Evolution*, należy z menu programu wybrać *Edycja/Ustawienia*.

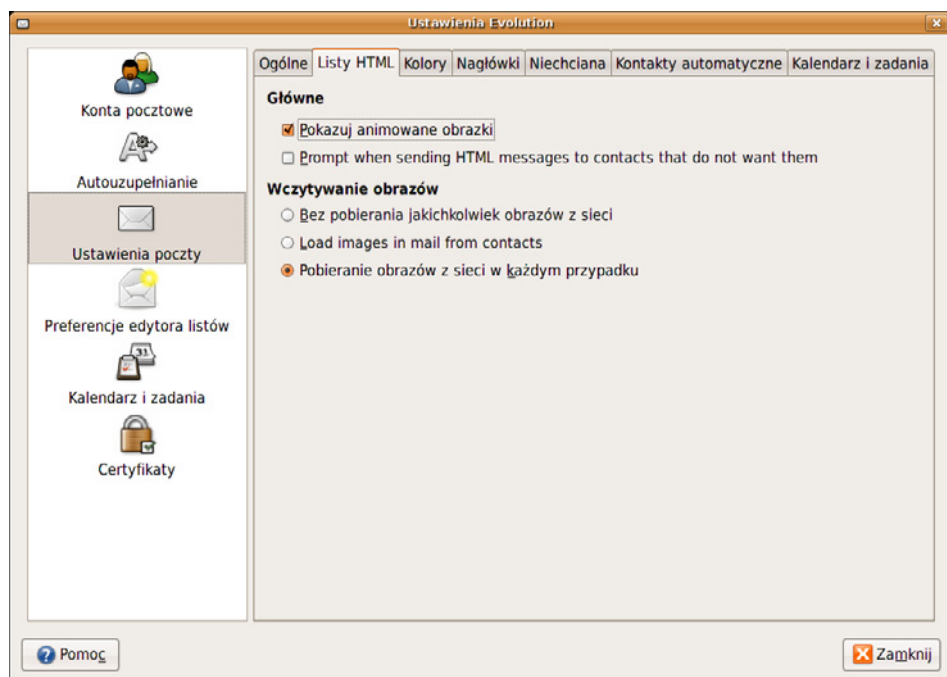
Żeby skonfigurować program tak, by dla wszystkich wysyłanych wiadomości stosowany był format HTML, należy wybrać w oknie *Preferencje edytora listów*. Zawartość okna widoczna jest na rysunku 8.19.

Aby wszystkie wysyłane wiadomości były formatowane w HTML-u, należy zaznaczyć pole *Zapis listów w formacie HTML*. Po zamknięciu okna *Ustawienia Evolution* wszystkie nowo tworzone wiadomości będą już zapisywane w formacie HTML.

Aby przychodzące wiadomości również były wyświetlane w formacie HTML, należy przejść do części *Ustawienia poczty*, a następnie do zakładki *Listy HTML* (patrz rysunek 8.20).



Rysunek 8.19. Ustawianie formatu HTML jako domyślnego dla wiadomości wychodzących



Rysunek 8.20. Opcje wyświetlania wiadomości przychodzących

Znaleźć tu można kilka opcji dla wiadomości przychodzących w formacie HTML. I tak np., jeżeli od czasu do czasu użytkownikowi zdarza się sprawdzać pocztę na komputerze innym niż domowy, z pewnością ważna jest szybkość działania. W takim przypadku można zrezygnować z pobierania obrazków, ponieważ ich wczytywanie trwa znacznie dłużej niż pobranie samej treści (niebagatelne znaczenie ma tutaj przepustowość łącza). Aby zapobiec pobieraniu obrazków, należy zaznaczyć opcję *Bez pobierania jakichkolwiek obrazków z sieci*. Żeby pobierać obrazki od znanych użytkownikowi osób (tych, które znajdują się w kontaktach), trzeba zaznaczyć opcję *Load images in mail from contacts*. Jeżeli natomiast zawsze mają być pobierane wszystkie obrazki, wówczas należy zaznaczyć opcję *Pobieranie obrazów z sieci w każdym przypadku*.

W oknie widocznym na rysunku 8.20 można również ustawić opcję, z pewnością docenioną przez osoby, które nie chcą otrzymywać wiadomości w formacie HTML; w takim przypadku wystarczy zaznaczyć pole *Prompt when sending HTML messages to contacts that do not want them*. Dzięki temu za każdym razem, kiedy będzie wysyłana wiadomość w formacie HTML na adres identyfikowany jako taki, który nie akceptuje HTML, program wyświetli komunikat z pytaniem, czy na pewno użyć takiego formatu.

Po zakończeniu wprowadzania zmian należy kliknąć przycisk *Zamknij*, okno zostanie zamknięte. Większość wprowadzanych zmian będzie zastosowana natychmiast, tylko niektóre z nich wymagają ponownego uruchomienia programu.



Evolution udostępnia użytkownikom szerokie możliwości konfigurowania sposobu, w jaki program ma współpracować z serwerami pocztowymi, wyświetlać przychodzące wiadomości, czy je tworzyć, a także wiele, wiele innych. Omówienie ich wykracza poza możliwości tego rozdziału, dlatego zachęcam użytkowników do eksperymentowania z różnorodnymi opcjami dostępnymi w wielu panelach i zakładkach okna *Ustawienia Evolution*, bowiem wtedy można skonfigurować program, aby działał dokładnie tak, jak życzy sobie użytkownik.

Odzyskiwanie usuniętych wiadomości

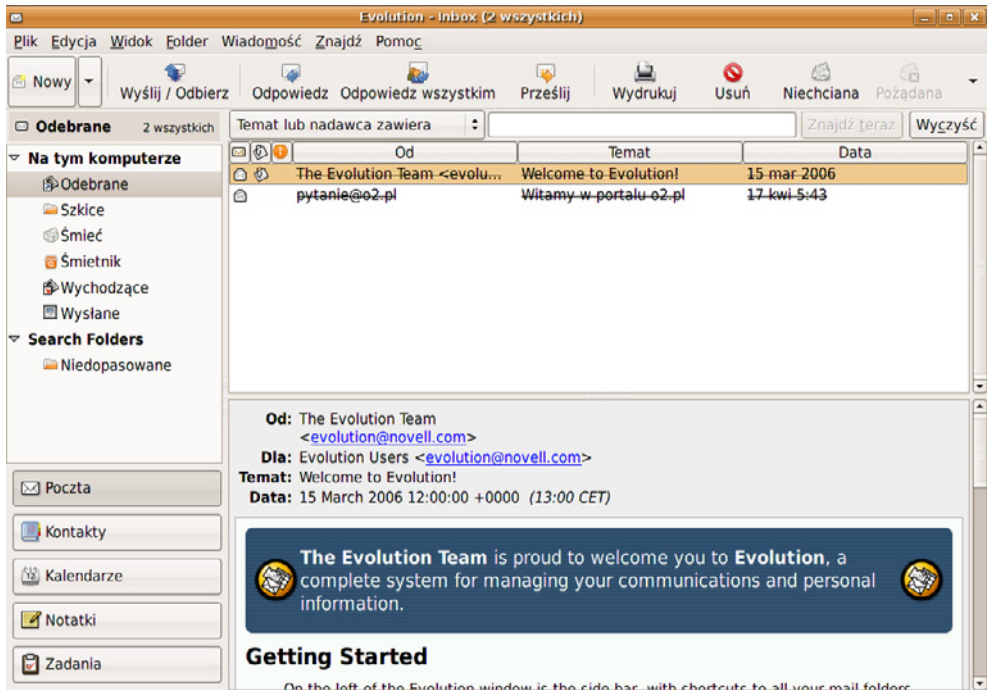
Podobnie jak w systemach DOS czy Microsoft Windows, wiadomości w programie Evolution nie są od razu całkowicie usuwane, są tylko oznaczane jako przeznaczone do usunięcia, kiedy użytkownik zdecyduje się opróżnić folder *Śmietnik* lub zamknąć program (jeżeli klient poczty jest tak skonfigurowany, aby wówczas opróżniać folder z usuniętymi wiadomościami).

Jeśli użytkownik przypadkowo usunie wiadomość, którą następnie będzie chciał odzyskać, wówczas należy z menu *Widok* wyłączyć opcję *Ukryj usunięte wiadomości*. Wtedy usunięte wiadomości zostaną wyświetlone z przekreśleniem (patrz rysunek 8.21).

Dla takich wiadomości można anulować znaczniki usunięcia, wystarczy je zaznaczyć, a następnie z menu *Edycja* wybrać opcję *Przywróć usuniętą wiadomość*.

Tworzenie i korzystanie z folderów

Po wysłaniu i odebraniu pewnej liczby e-maili skrzynka odbiorcza szybko zapełni się, a wyszukiwanie potrzebnych wiadomości będzie coraz trudniejsze, mimo że Evolution oferuje możliwość odnajdywania potrzebnych wiadomości za pomocą okna wyszukiwarki umiesz-



Rysunek 8.21. Wyświetlanie wiadomości oznaczonych jako usunięte

czonego nad wyświetlaną zawartością wybranego folderu. Ale po co wyszukiwać wiadomości, skoro można je w pierwszej kolejności lepiej zorganizować? Lista folderów, które są dostępne w programie domyślnie, jest wyświetlana w panelu nawigacyjnym widocznym po lewej stronie, ale użytkownicy mogą łatwo i szybko tworzyć foldery samodzielnie i lepiej (na stałe) porządkować posiadane wiadomości.

Podczas konfigurowania programu tworzone są następujące foldery.

- ♦ **Odebrane** — w tym folderze składowane są wiadomości po pobraniu ich z serwera.
- ♦ **Szkice** — tu są przechowywane wiadomości, nad którymi użytkownik pracował, ale nie zostały wysłane.
- ♦ **Śmieć** — tu przechowywane są wiadomości, które przez specjalny filtr programu Evolution zostały zidentyfikowane jako niechciane. Więcej informacji na ten temat można znaleźć w punkcie „Automatyczne sprawdzanie niechcianej poczty”,
- ♦ **Śmietnik** — wiadomości przeznaczone do usunięcia, kiedy użytkownik zdecyduje się wykonać operację *Opróżnij śmietnik* lub wyłączy program (w przypadku, gdy program zostanie skonfigurowany tak, aby wykonać tę operację podczas wyłączania programu),
- ♦ **Wychodzące** — folder, w którym przechowywane są wiadomości, które użytkownik wysłał, ale nie zostały one jeszcze przekazane przez program na serwer,
- ♦ **Wysłane** — w tym folderze przechowywane są kopie wiadomości, które zostały wysłane, dzięki czemu użytkownik ma stały wgląd w ich zawartość.

Z wyjątkiem folderu *Odebrane*, który zawsze jest na początku listy (aby można go było łatwo odnaleźć), pozostałe są wyświetlane w porządku alfabetycznym.

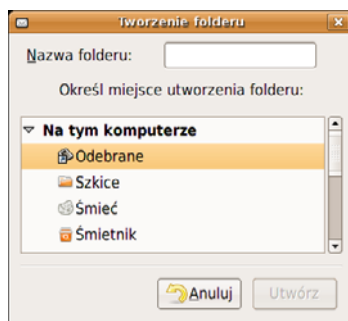
Aby utworzyć nowy folder, należy wykonać jedną z dwóch czynności:

- ♦ z menu *Folder* wybrać opcję *Nowy*,
- ♦ kliknąć prawym przyciskiem myszy ten folder, w którym ma zostać utworzony nowy i z menu kontekstowego wybrać *Nowy folder*.

Po wykonaniu jednej z tych czynności zostanie wyświetlone nowe okno (patrz rysunek 8.22).

Rysunek 8.22.

Tworzenie nowego folderu pocztowego



W tym oknie należy przejść do odpowiedniego folderu, wprowadzić nazwę dla nowo tworzonego i wcisnąć przycisk *Utwórz*.

Tworzenie nowych folderów dla wiadomości ułatwia ich porządkowanie, dzięki czemu w skrzynce odbiorczej pozostanie minimalna ich ilość, a często tylko nowe. Struktura folderów może być hierarchiczna, innymi słowami, jeden folder może zawierać inne. I tak np. można utworzyć folder o nazwie *Przyjaciele*, w jakim powstaną kolejne z nazwiskami znajomych, z którymi prowadzi się często korespondencję. Po odebraniu lub wysłaniu wiadomości do jednej z takich osób, należy przesunąć ją do odpowiedniego folderu. Wtedy w razie potrzeby można szybko przejrzeć korespondencję z wybraną osobą bez potrzeby przeglądania innej.

Korzystanie z folderów wyszukiwania

Evolution to pierwszy klient pocztowy, do którego wprowadzono koncepcję folderów wyszukiwania służących do przeglądania grupy powiązanych wiadomości. Standardowe foldery pocztowe przechowują wiadomości, foldery wyszukiwania można porównać do indeksów lub spisów zawartości odnoszących się do grupy powiązanych wiadomości. Dzięki temu wiadomości nadal można przechowywać w standardowych folderach, ale ich zawartość wyświetlać za pomocą folderów wirtualnych. W starszych edycjach Evolution używano nazwy *vfoldery* lub *foldery wirtualne*.

Oto przykład: założmy, że czytelnik subskrybuje dwie różne listy dyskusyjne, jedną poświęconą różnym dystrybucjom Linuksa, a drugą — systemom plików, a przychodzące z obu list wiadomości segreguje w oparciu o nazwę danej listy. Co zrobić, aby użytkownik mógł wyświetlać wszystkie wiadomości dotyczące danego systemu plików, niezależnie od tego,

na której z list się pojawiły? Zawsze można utworzyć filtr wiadomości (więcej na temat tego rozwiązania powiem w następnym punkcie), który będzie wyszukiwał odpowiednie i umieszczał je w przeznaczonym do tego folderze, ale będzie to marnotrawienie przestrzeni. W takiej sytuacji z pomocą przychodzą foldery wyszukiwania dostępne w Evolution — wystarczy utworzyć taki, który będzie wskazywał wszystkie wiadomości według określonego klucza bez ich kopiowania.

Jak większość operacji wykonywanych w Evolution, również tę można wykonać na kilka sposobów:

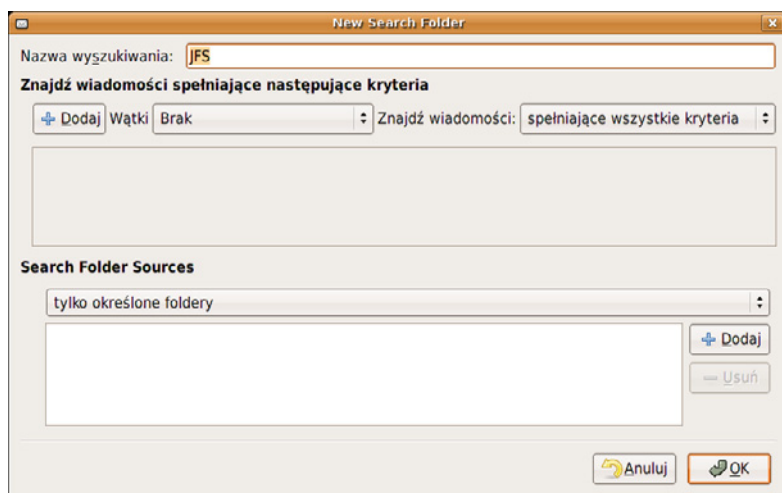
- ♦ kliknąć prawym przyciskiem myszy znajdujący się w panelu nawigacyjnym *Search Folder*, a następnie wybrać z menu pozycję *Nowy*,
- ♦ kliknąć prawym przyciskiem myszy wiadomość, spełniającą kryteria, na podstawie których ma być prowadzone wyszukiwanie, wybrać z menu kontekstowego pozycję *Utwórz regułę na podstawie listy*, a następnie jedną z opcji *Search folder from Subject* (folder wyszukiwania na podstawie tematu), *Search folder from Sender* (folder wyszukiwania na podstawie nadawcy), *Search folder from Recipient* (folder wyszukiwania na podstawie adresata), *Search folder from Mailing List* (folder wyszukiwania listy dyskusyjnej) — ostatnia pozycja pojawi się tylko wówczas, gdy program będzie w stanie rozpoznać, że dana wiadomość pochodzi właśnie z danej listy; każda z tych opcji powoduje utworzenie folderu wyszukującego określone informacje zawarte w części wiadomości, opcje te można dostosować później,
- ♦ wybrać z menu kolejno *Edycja/Search Folders*, a następnie kliknąć przycisk *Dodaj*,
- ♦ jeżeli zaznaczona jest wiadomość spełniające kryteria, które mają być zastosowane dla folderu wyszukiwania, należy wybrać kolejno z menu *Wiadomość/Utwórz regułę/Search Folder from Sender* (w zależności od potrzeb, można też użyć opcji *Search folder from Recipient* lub *Search folder from Mailing List*, ostatnia pozycja pojawi się tylko wówczas, gdy program będzie w stanie rozpoznać, że dana wiadomość pochodzi właśnie z listy dyskusyjnej). Każda z tych opcji powoduje utworzenie folderu wyszukującego określone informacje zawarte w części wiadomości, opcje te można dostosować później.

Wykonanie którejkolwiek z tych opcji spowoduje wyświetlenie okna dialogowego *New Search Folder* (patrz rysunek 8.23), w którym zawarte będą początkowo określone kryteria wyszukiwania.

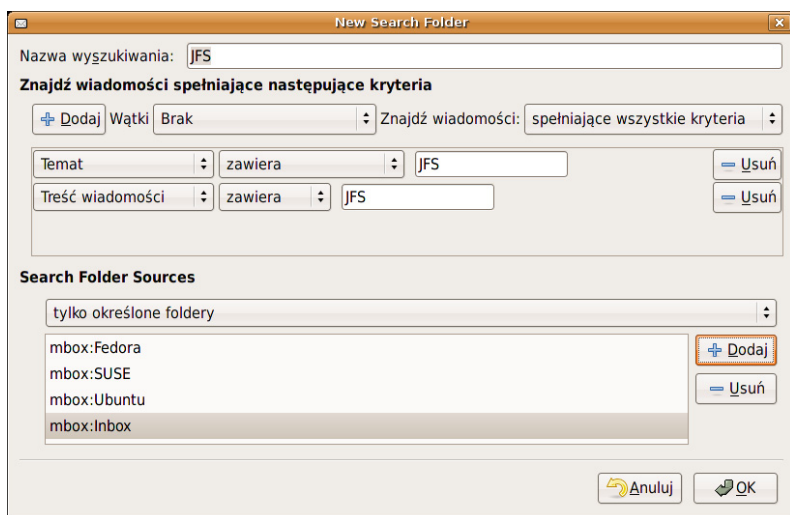
Kiedy okno zostanie wyświetlone, można wprowadzić w nim nazwę dla nowego folderu i dodać lub zmienić kryteria wyszukiwania w taki sam sposób, jak ma to miejsce w przypadku filtrów wiadomości, co omawiam poniżej. Główna różnica pomiędzy folderami wyszukiwania a regułami wyszukiwania widoczna jest w dolnej części okien dialogowych (patrz rysunki 8.23 i 8.24), gdzie dla folderów wyszukiwania określona jest lokalizacja. Użytkownik może wskazać jeden folder, który będzie sprawdzany, wybrać opcję *ze wszystkimi lokalnymi folderami*, *ze wszystkimi aktywnymi zdalnymi folderami*, *ze wszystkimi lokalnymi i aktywnymi zdalnymi folderami*. Na rysunku 8.24 przedstawiam parametry folderu wyszukiwania ograniczone do określonego zestawu folderów.

Rysunek 8.23.

Tworzenie nowego folderu wyszukiwania

**Rysunek 8.24.**

Reguła wyszukiwania przypisana do danego folderu



Po zakończeniu określania parametrów, które muszą spełniać wiadomości, aby być wyświetlane w danym folderze wyszukiwania, należy kliknąć przycisk **OK**, okno dialogowe zostanie wówczas zamknięte. Nowy folder wyszukiwania pojawi się w panelu nawigacyjnym i będą w nim wyświetlane wiadomości spełniające określone kryteria

Filtrowanie wiadomości przychodzących

Dziś do skrzynek pocztowych trafiają ogromne ilości przesyłek. Niektóre z nich są oczekiwane. Smutnym znakiem czasów jest ogromna ilość niechcianych przesyłek i spamu. Te terminy określają wiadomości, o które użytkownik nie prosił, a są rozsyłane na podstawie generowanych list wysyłkowych lub wówczas, gdy dany adres miał nieszczęście znaleźć się na jednej z list, którymi wymieniają się organizacje wysyłające takie przesyłki.

Evolution wyposażony jest w dwa zaawansowane narzędzia służące do analizowania przychodzących wiadomości i wykonywania określonych zadań, jeżeli spełniają one zadane kryteria. W przypadku właściwych wiadomości program umożliwia zastosowanie wydajnych reguł filtrowania, dzięki którym otrzymywane przesyłki będą przenoszone do odpowiednich folderów. Filtry potrafią wyłowić również wiadomości pochodzące ze źródeł znanych z rozsyłania spamu, takie, w których użyto słów charakterystycznych dla spammerów, oraz stosować inne kryteria rozpoznawania spamu.

Aby uzupełnić możliwości programu w zakresie wydajnego filtrowania wiadomości, można Evolution zintegrować z innym darmowym oprogramowaniem przeznaczonym dla Linuksa, które umożliwia zastosowanie dodatkowych metod wyłapywania spamu i automatycznego przenoszenia go do folderu *Śmieć*. Dzięki temu wystarczy tylko od czasu do czasu sprawdzić ten folder, aby upewnić się, że wszystkie wiadomości, które tam trafiły, faktycznie są niechciane, a jeżeli tak jest, po prostu je usunąć.

Poniżej omawiam sposób, w jaki należy tworzyć własne filtry wiadomości, oraz podaję, jak zainstalować dodatkowe oprogramowanie wspomagające walkę ze spamem w Evolution i Ubuntu.

Stosowanie filtrów dla poczty przychodzącej

Rozszerzając definicję filtrowania wiadomości podaną powyżej, można powiedzieć, że jest to proces analizowania zawartości w oparciu o różne kryteria, a na podstawie jego wyników wykonywane są pewne dodatkowe czynności. Filtry wiadomości są zwykle stosowane do poczty przychodzącej, przechowywanej w folderach lokalnych; gdyby użytkownik chciał zastosować je w odniesieniu do folderów zdalnych (np. na serwerze IMAP), wówczas musi określić odpowiednie opcje podczas konfigurowania konta w programie Evolution (patrz rysunek 8.7).

Jak większość operacji w Evolution, również tworzenie filtrów można wykonać na kilka sposobów.

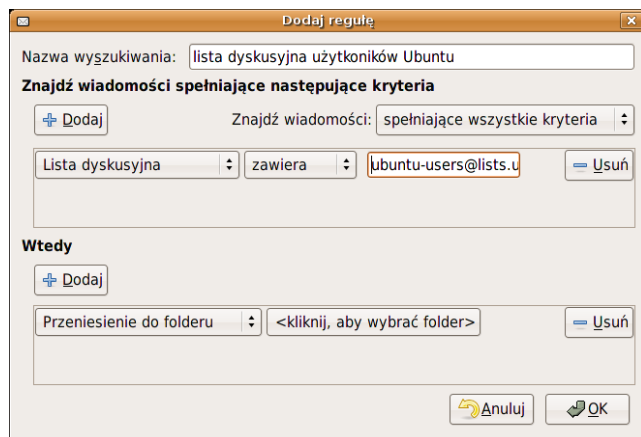
- ♦ Kliknąć prawym przyciskiem myszy wiadomość spełniającą kryteria, które mają być stosowane przez dany filtr, wybrać z menu kontekstowego *Utwórz regułę na podstawie listy*, a następnie jedną z opcji: *Filtr w oparciu o temat*, *Filtr w oparciu o nadawcę*, *Filtr w oparciu o odbiorców*, *Filtr w oparciu o listę dyskusyjną* — ostatnia pozycja pojawi się tylko wówczas, gdy program będzie w stanie rozpoznać, że dana wiadomość pochodzi właśnie z listy dyskusyjnej. Każda z tych opcji powoduje otwarcie okna dialogowego z możliwością utworzenia filtra na podstawie ustalonego zakresu informacji, opcje te można oczywiście dostosować później.
- ♦ Wybierając z menu kolejno *Edycja/Filtry wiadomości* i klikając przycisk *Dodaj*.
- ♦ Jeżeli zaznaczona jest wiadomość spełniająca kryteria, które mają być zastosowane w danym filtrze, należy wybrać z menu *Utwórz regułę na podstawie listy*, a następnie jedną z opcji: *Filtr w oparciu o temat*, *Filtr w oparciu o nadawcę*, *Filtr w oparciu o odbiorców*, *Filtr w oparciu o listę dyskusyjną* — ostatnia pozycja pojawi się tylko wówczas, gdy program będzie w stanie rozpoznać, że dana

wiadomość pochodzi właśnie z listy dyskusyjnej. Każda z tych opcji powoduje otwarcie okna dialogowego z możliwością utworzenia filtra na podstawie ustalonego zakresu informacji, opcje te można dostosować później.

Wykonanie którejkolwiek z tych opcji spowoduje wyświetlenie okna dialogowego (patrz rysunek 8.25), w którym zawarte będą początkowo określone kryteria wyszukiwania.

Rysunek 8.25.

*Tworzenie
i edytowanie
filtra wiadomości*



Podczas tworzenia filtra wiadomości podstawową czynnością, którą ma wykonać program, jest przeniesienie listu spełniającego określone kryteria do wskazanego folderu. Aby go zdefiniować, należy użyć przycisku *kliknij, aby wybrać folder*, a następnie podświetlić właściwy.

Evolution oferuje wiele skomplikowanych sposobów na oznaczenie określonych wiadomości. Poniżej zamieszczam przykłady operatorów, których można użyć, wystarczy rozwinąć listę *Nadawca* i *zawiera*.

- ♦ Na liście *Nadawca* znajdują się następujące możliwości wyboru: *Nadawca, Lista odbiorców, Temat, Określony nagłówek, Treść wiadomości, Wyrażenie, Data wysłania, Data otrzymania, Etykieta, Punkty, Rozmiar (KB), Stan, Kontynuuj, Załączniki, Lista dyskusyjna, Dopasowanie według regexp, Konto źródłowe, Przesłanie do programu, Test spamu, Pasuje do wszystkich*.
- ♦ Na liście *zawiera* znajdują się następujące możliwości wyboru: *zawiera, nie zawiera, jest, nie jest, rozpoczyna się od, nie rozpoczyna się od, kończy się na, nie kończy się na, brzmi jak, nie brzmi jak, jest przed, jest po, jest większy niż, jest mniejszy niż*. Niektóre z wartości mogą być stosowane tylko dla odpowiednich części wiadomości: np. operatory *jest przed* i *jest po* mogą zostać zastosowane dla daty wiadomości.

Podczas tworzenia wzorca dopasowania w filtrze użytkownik może korzystać z pozycji z pierwszej i drugiej kolumny w dowolnej kombinacji. Jednak nie wszystkie operatory z drugiej kolumny będą dostępne dla wartości z pierwszej, np. *jest większy niż* oraz *jest mniejszy niż* mają zastosowanie dla rozmiaru wiadomości, czyli dla wartości liczbowej.

Jako przykład stosowania kryteriów do filtrowania wiadomości przyjmijmy, że działaniu filtra mają być poddane wszystkie wiadomości pochodzące z adresu `ubuntu-users@lists.ubuntu.com`. Najpierw należy wybrać w pierwszej kolumnie pole *Lista dyskusyjna*, w drugiej kolumnie opcję *zawiera*, a w trzeciej należy wpisać adres `ubuntu-users@lists.ubuntu.com`, na koniec trzeba jeszcze wcisnąć klawisz *OK* i filtr został utworzony (patrz rysunek 8.25).



Podczas tworzenia filtrów najważniejszym aspektem jest określenie, czy dana wiadomość ma spełniać tylko jeden warunek, czy też muszą być spełnione wszystkie założenia, aby zostało wykonane zaplanowane działanie. To może decydować, czy filtr będzie działał właściwie, czy też niezgodnie z zamierzeniami użytkownika. Tę opcję można kontrolować w polu *Znajdź wiadomość* umieszczonym w górnym prawym rogu okna dialogowego służącego do tworzenia reguł filtrowania; z umieszczonej tam listy rozwijanej można wybrać opcje: *spełniające wszystkie kryteria* lub *spełniające dowolne z kryteriów*.

W tabeli 8.1 zamieszczono listę operacji, które można przypisać do wiadomości spełniającej wszystkie lub jedno kryterium filtrowania.

Tabela 8.1. Działania wykonywane po dopasowaniu wiadomości do filtra

Akcja	Działanie
<i>Brzęczyk</i>	Brak
<i>Dopasuj punkty</i>	Wprowadzić wartość od 0 do 3
<i>Odtworzenie dźwięku</i>	Wybrać plik dźwiękowy
<i>Przeniesienie do folderu</i>	Wskazać folder docelowy
<i>Przesłanie do programu</i>	Wskazać program
<i>Przypisz kolor</i>	Wybrać kolor
<i>Przypisz punkty</i>	Wprowadzić wartość od 0 do 3
<i>Skopiuj do folderu</i>	Wskazać folder
<i>Uruchamiany program</i>	Wskazać program
<i>Ustaw stan</i>	Można zaznaczyć jako: <i>Odpowiedziano do</i> , <i>Usunięty</i> , <i>Robocze</i> , <i>Ważne</i> , <i>Przeczytane</i> , <i>Niechciana</i>
<i>Usuń</i>	Brak
<i>Wyczyszczenie stanu</i>	Można anulować zaznaczenie: <i>Odpowiedziano do</i> , <i>Usunięty</i> , <i>Robocze</i> , <i>Ważne</i> , <i>Przeczytane</i> , <i>Niechciana</i>
<i>Zatrzymanie przetwarzania</i>	Brak

W tabeli 8.1 w kolumnie Akcja zamieszczono czynność, którą program ma wykonać po dopasowaniu wiadomości do reguły filtra, w kolumnie Działanie natomiast znajdują się możliwe do wyboru opcje. I tak np., jeżeli użytkownik chciałby, aby wiadomości spełniające kryteria były przenoszone do wybranego folderu, wówczas w pierwszej kolumnie powinien wybrać akcję *Przeniesienie do folderu*, a po kliknięciu drugiej (*kliknij, aby wybrać folder*) zostanie wyświetlone okno, w którym będzie można wskazać folder docelowy.

Kontynuujemy powyższy przykład z wiadomościami przychodzącymi z adresu *ubuntu-users@lists.ubuntu.com*: użytkownik może nakazać ich przenoszenie do nowo utworzonego folderu, wystarczy tylko wybrać opcję *Przeniesienie do folderu*, a następnie wskazać ten, który ma być docelowy. Można również zaplanować pewne dodatkowe operacje, takie jak np. ustawienie stanu wiadomości (opcja *Ustaw stan*) na *Przeczytane*, w przeciwnym razie nowe wiadomości będą wyświetlane pogrubioną czcionką, która informuje o pojawieniu się nieprzeczytanych elektronicznych listów; można też zastosować akcję *Zatrzymanie przetwarzania*, dzięki czemu dla danej wiadomości nie będą stosowane żadne dodatkowe filtry, ponieważ po przypisaniu takiego statusu wiadomość nie będzie już analizowana przez program Evolution, który zajmie się kolejną wiadomością.

Automatyczne sprawdzanie niechcianej poczty

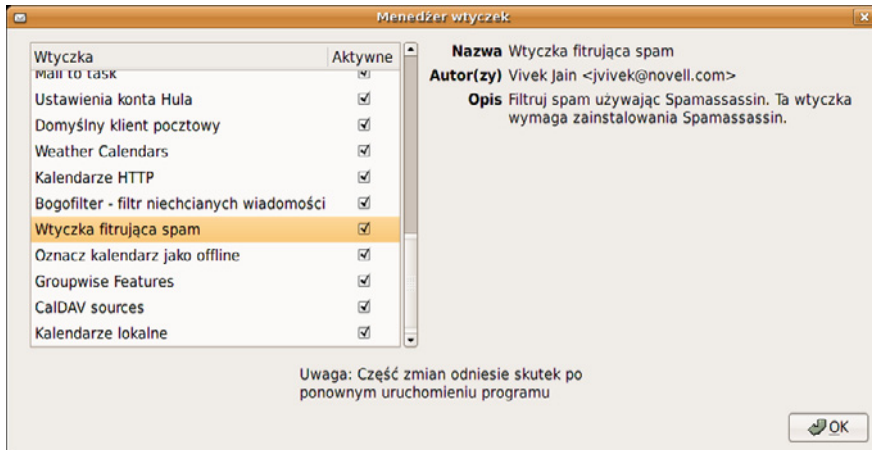
Dostępne w Evolution reguły filtrowania wiadomości oparte na definiowaniu kryteriów udostępniają ogromne możliwości, ale aby skutecznie walczyć z niechcianymi przesyłkami oraz spamem, należałoby potencjalnie utworzyć ogromną liczbę reguł. Na szczęście, Evolution zawiera zintegrowane narzędzie, które może samodzielnie analizować wiadomości i wyłapywać te niechciane. Przykładami takich popularnych narzędzi są SpamAssassin (<http://spamassassin.apache.org>) oraz Bogofilter (<http://bogofilter.sourceforge.net>).

W Evolution znajduje się wtyczka programu SpamAssassin uruchamiająca go automatycznie, dzięki czemu wykonywane jest sprawdzanie przychodzącej poczty. Ponieważ program posiada ograniczenia licencyjne, nie jest instalowany domyślnie wraz z Ubuntu, użytkownik powinien dodać go do systemu samodzielnie. Aby to zrobić, należy dodać do systemu repozytorium *Universal*, a następnie uruchomić menedżera pakietów Synaptic lub program *apt-get* i za ich pośrednictwem zainstalować SpamAssassina (więcej informacji na temat dodawania repozytoriów zarządzania pakietami zamieszczam w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”).

Po pobraniu i zainstalowaniu SpamAssassina należy sprawdzić w Evolution listę wtyczek, aby upewnić się, że został on włączony. Aby to sprawdzić, wystarczy wybrać z menu kolejno *Edycja/Wtyczki* (patrz rysunek 8.26).

Następnie należy odszukać na liście odpowiednią pozycję, a w razie potrzeby umieścić przy niej znak wyboru. Aby zamknąć okno, trzeba kliknąć przycisk *OK*.

Aby — po zainstalowaniu programu SpamAssassin i upewnieniu się, że jest on włączony — oznaczyć jakąś wiadomość jako niechcianą, wystarczy ją zaznaczyć, a następnie kliknąć przycisk *Niechciana* znajdujący się na pasku narzędziowym, spowoduje to przeniesienie jej do folderu *Śmieć*. Od tego momentu wszystkie nowe wiadomości, które będą podobne do oznaczonej, automatycznie rozpoznawane będą jako śmieci i umieszczane w folderze o tej samej nazwie. Folder *Śmieć* należy sprawdzać regularnie, aby upewnić się, że wszystkie wiadomości, które tam trafiły, faktycznie są niepożądane i usuwać je stamtąd. Jeżeli jakaś wiadomość zostanie rozpoznana jako śmieć, chociaż wcale nim nie jest, należy wówczas ją zaznaczyć, a następnie użyć przycisku *Pożądana* umieszczonego również na pasku narzędziowym. Z czasem użytkownik będzie mógł stwierdzić, że program SpamAssassin dobrze wykonuje swoją pracę (identyfikuje spam i niechciane wiadomości),



Rysunek 8.26. Włączanie wtyczki programu SpamAssassin

oszczędzając w ten sposób mnóstwo czasu, który w przeciwnym razie należałoby przeznaczyć na tworzenie nowych reguł do filtrowania niepożądanego poczty.



Jeżeli podczas instalowania czy korzystania z programu SpamAssassin użytkownik napotka jakieś trudności, należy sprawdzić, czy nie jest włączona wtyczka Bogofilter (menu *Edycja/Wtyczki*). Może to być przyczyną konfliktów.

Dodatkowe źródła informacji o Evolution

Wbrew tytułowi, zamieszczone poniżej informacje nie są przeznaczone specjalnie dla fanów Karola Darwina, ponieważ dotyczą programu do obsługi poczty elektronicznej i zarządzania informacjami, a nie teorii ewolucji.

Z racji tego, że program Evolution jest bardzo często używany jako domyślny klient pocztowy, zwłaszcza w systemach, w których korzysta się ze środowiska GNOME, można znaleźć pokaźną liczbę stron internetowych zawierających informacje na temat jego używania. Podobnie ma się rzecz z listami dyskusyjnymi, na których można stawiać pytania, czy szukać rozwiązań. Oto kilka moich ulubionych źródeł.

- ♦ Forum Evolution (<http://www.nabble.com/Gnome---Evolution-f1297.html>) — jeżeli czytelnicy wolą interfejs forów dyskusyjnych od list dyskusyjnych, na tej stronie znajdują obszerny zbiór odnośników do list poświęconych różnym aspektom użytkowania programu w przyjaznym formacie. Wiadomości tu umieszczane są wysyłane bezpośrednio na odpowiednie listy dyskusyjne.
- ♦ Kanał IRC poświęcony programowi Evolution — zwolennicy IRC-a i posiadacze odpowiedniego programu (takiego jak np. Xchat-GNOME) mogą do woli korzystać z kanału *#evolution* na *irc.gimp.org*. Przed zadaniem pytania czy włączeniem się w dyskusję warto poświęcić chwilę na obserwację, w jaki sposób kanał funkcjonuje i jak prowadzi się na nim rozmowy, dzięki temu można uniknąć oskarżeń o niewłaściwe zadawanie pytań lub stawianie takich, na które odpowiedzi już dawno znajdują się w dokumentacji.

- ♦ Lista dyskusyjna — na stronie <http://mail.gnome.org/mailman/listinfo/evolution-list> można subskrybować ogólną listę dyskusyjną przeznaczoną dla użytkowników Evolution (znaną jako *evolution-list* — brzmi znajomo, nieprawdaż?). Można też korzystać z listy przeznaczonej dla bardziej zaawansowanych użytkowników (<http://mail.gnome.org/mailman/listinfo/evolution-hackers>). Więcej informacji o listach dyskusyjnych związanych z GNOME można znaleźć na stronie <http://mail.gnome.org/mailman/listinfo/>.
- ♦ Strona domowa projektu Evolution (<http://www.gnome.org/projects/evolution/>) — jest to najlepszy punkt do rozpoczęcia poszukiwań dodatkowych informacji o programie Evolution, a także kodu źródłowego, najnowszych wersji itd.
- ♦ Evolution Wiki — (http://www.go-evolution.org/Main_Page) znakomita strona z informacjami na temat najnowszego wydania programu, można też dowiedzieć się więcej na temat funkcji dostępnych w kolejnym wydaniu Evolution, a nawet zgłosić własne propozycje.

Oczywiście, każda przyzwoita wyszukiwarka internetowa wyświetli całe mnóstwo różnorodnych stron poświęconych omawianemu programowi, ale zamieszczone powyżej mogą być dobrym początkiem poszukiwań.

Podsumowanie

W niniejszym rozdziale opowiedziałem, jak skonfigurować Evolution, wydajny program do obsługi poczty elektronicznej i zarządzania informacjami osobistymi. Skupiłem się na obsłudze poczty elektronicznej. Evolution to doskonały program obsługujący wszystkie popularne protokoły pocztowe, umożliwia też wyjątkową możliwość łączenia się z różnymi serwerami pocztowymi, np. z Microsoft Exchange.

Czytanie i wysyłanie wiadomości to jedna z najpowszechniejszych czynności wykonywanych obecnie przez użytkowników komputerów. Równie powszechne jest surfowanie w internecie, które omówię w kolejnym rozdziale. Aby zaspokoić potrzeby użytkowników w tym zakresie, w Ubuntu dostępna jest doskonała przeglądarka internetowa — Mozilla Firefox.

Rozdział 9.

Surfowanie z Firefoksem

W tym rozdziale:

- ♦ Historia Firefoksa
- ♦ Korzystanie z Firefoksa
- ♦ Konfigurowanie programu Firefox
- ♦ Praca z zakładkami
- ♦ Rozszerzenia dla Firefoksa

W Ubuntu użytkownik może zainstalować wiele różnych przeglądarek internetowych, dotyczy to zresztą całego oprogramowania. Środowiska, takie jak GNOME czy KDE, są wyposażone w uniwersalne przeglądarki (są to odpowiednio Nautilus i Konqueror), których można używać jako jednego narzędzia do wielu zastosowań — również do przeglądania internetu. W Ubuntu jednak jako domyślna przeglądarka internetowa instalowany jest produkt Mozilli, Firefox, a to z jednego, ale dobrego powodu: jest to najlepsza przeglądarka dostępna obecnie dla systemów Linux (uważam, że dotyczy to również innych systemów). Firefox jest najpopularniejszą przeglądarką o otwartym kodzie, która w dodatku jako jedyna w ostatniej dekadzie (mniej więcej) zdołała odebrać znaczącą część rynku Internet Explorerowi. Wiele usprawnień, z których mogą aktualnie korzystać użytkownicy najnowszej wersji programu Microsoftu, od kilku wydań było już dostępnych w Firefoksie.

Niniejszy rozdział rozpoczynam od krótkiego przedstawienia historii tego programu, następnie omówię uruchamianie, konfigurowanie, używanie i dostosowywanie Firefoksa.

Krótką historia Firefoksa

Firefox posiada prawdopodobnie najlepszy możliwy rodowód w historii przeglądarek — jest ostatnim potomkiem przeglądarki Mosaic, utworzonej początkowo dla systemów Unix. Mosaic powstała w roku 1992 w NCSA (ang. *National Center for Supercomputing Applications*) na uniwersytecie w Illinois w Urbana-Champaign.



Krótką historię internetu zamieszczam w rozdziale 26., w podrozdziale „Podstawy sieci WWW”.

Kiedy nastąpił okres internetowego boomu, wielu członków zespołu NCSA (włącznie z kierownikiem zespołu Markiem Andreessenem) założyło firmę Mosaic Communications Corporation. Ze względu na zastrzeżenie nazwy, firma Mosaic Communications Corporation szybko zmieniła się w Netscape Communications Corporation, a ich przeglądarka została nazwana Netscape Navigator. Wczesna wersja tego programu została opublikowana w połowie roku 1994, oficjalne wydanie miało miejsce z końcem tegoż roku.

W jednym z kontrowersyjnych rozwojów sytuacji, dzięki którym historia techniki jest ciekawa, firma o nazwie Spyglass kupiła licencję przeglądarki Mosaic od uniwersytetu w Illinois i opracowała własny program o nazwie Spyglass Mosaic, przeznaczony dla komputerów osobistych z działającym na nich systemem Microsoft Windows. W dalszej kolejności Spyglass przekazał licencję przeglądarki firmie Microsoft, która z kolei na kodzie programu Spyglass Mosaic oparła własną — Internet Explorer.

Wróćmy jednak do — w pewnym stopniu — biblijnej historii Firefoksa. Netscape Navigator był jedynie przeglądarką internetową, a stale rosnące rzesze internautów domagały się kolejnych programów powiązanych z internetem, takich jak program do obsługi poczty elektronicznej czy tworzenia stron internetowych. To doprowadziło do stworzenia przez firmę Netscape pakietu Netscape Communicator Suite.

Internet Explorer szybko zaczął zdobywać rynek kosztem Netscape — był bezpłatny i dołączany do każdego systemu operacyjnego firmy Microsoft, towarzyszył też serwerom przeznaczonym do pracy w sieci (*Internet Information Server*). W roku 1998 Netscape Network Communication praktycznie przestało istnieć jako samodzielne przedsiębiorstwo, zostało wykupione przez *America Online* (AOL) za zaskakującą sumę 4 mld dolarów.

Netscape złapał oddech, kiedy przekazał kod źródłowy przeglądarki projektowi Mozilla i towarzyszącej mu Mozilla Organization. Celem obu tych organizacji było utworzenie dla Netscape pakietu narzędzi internetowych następnej generacji. Mozilla to oryginalna nazwa kodu źródłowego przeglądarki Netscape, a wizerunek czerwonego smoka był popularnym motywem dekoracyjnym w programie. Projekt Mozilla początkowo krytykowany za chaos organizacyjny w końcu uporządkował swoje działania, następnie doprowadził do zmodularyzowania kodu i nadal kontynuuje założenia Netscape na stronie www.netscape.com.

W tym czasie projekt Mozilla opracował swoją własną przeglądarkę. Pierwotna nazwa brzmiała Phoenix, następnie Firebird, jednak ze względu na to, że już istniały programy o takich nazwach, ostateczną wersją jest Firefox.



Chociaż Firefox i jego brat Thunderbird, program do obsługi poczty elektronicznej, są podstawowymi produktami Mozilla Foundation, to pracuje ona jednocześnie nad innymi projektami. Najbardziej interesujący jest SeaMonkey, pakiet aplikacji internetowych oparty o oryginalny Mozilla Application Suite, który z kolei był następcą Netscape Communicatora. SeaMonkey zawiera przeglądarkę internetową, klienta poczty elektronicznej, czytnik grup dyskusyjnych, klienta IRC i narzędzie do tworzenia stron internetowych w HTML-u. Więcej informacji można znaleźć na stronie <http://mozillapl.org/wydania/seamoney> oraz <http://www.mozilla.org/projects/seamoney/>.

Historia Firefoksa to ciągle ulepszenia, z których korzystamy obecnie, a są to m.in. zintegrowana blokada wyświetlania wyskakujących okien, paski narzędzi, które można dostosowywać, pasek boczny, zintegrowana z paskiem narzędzi wyszukiwarka, graficzne zarzą-

dzenie plikami *cookies*, menedżer pobierania plików, motywy, możliwość dodawania rozszerzeń i przeglądanie w zakładkach.

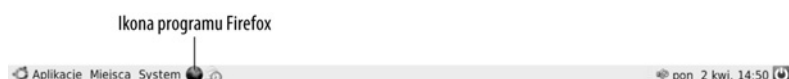
Uruchamianie programu Firefox

Program Firefox można uruchomić na wiele różnych sposobów, oto kilka najczęściej używanych.

- ♦ Użycie ikony programu znajdującej się na górnym panelu GNOME (patrz rysunek 9.1). Jest to najprostsza metoda uruchamiania Firefoksa.

Rysunek 9.1.

Ikona Firefoksa umieszczona na panelu GNOME



- ♦ Uruchomienie programu z menu *Aplikacje/Internet*.
- ♦ Wpisanie do terminala GNOME polecenia `firefox`, po czym wciśnięcie klawisza *Enter*.

GNOME jest skonfigurowany w taki sposób, że przeglądarka Firefox będzie uruchamiana również wówczas, gdy użytkownik z poziomu programu do obsługi poczty czy innego związanego z internetem kliknie adres URL (ang. *Uniform Resource Locater*).

Po pierwszym uruchomieniu programu zostanie wyświetlona domyślna strona powitalna widoczna na rysunku 9.2. Można na niej znaleźć podstawowe informacje na temat używanego wydania Ubuntu. Jeżeli Firefox został uruchomiony za pośrednictwem adresu URL, wówczas zostanie wyświetlona wskazana strona.

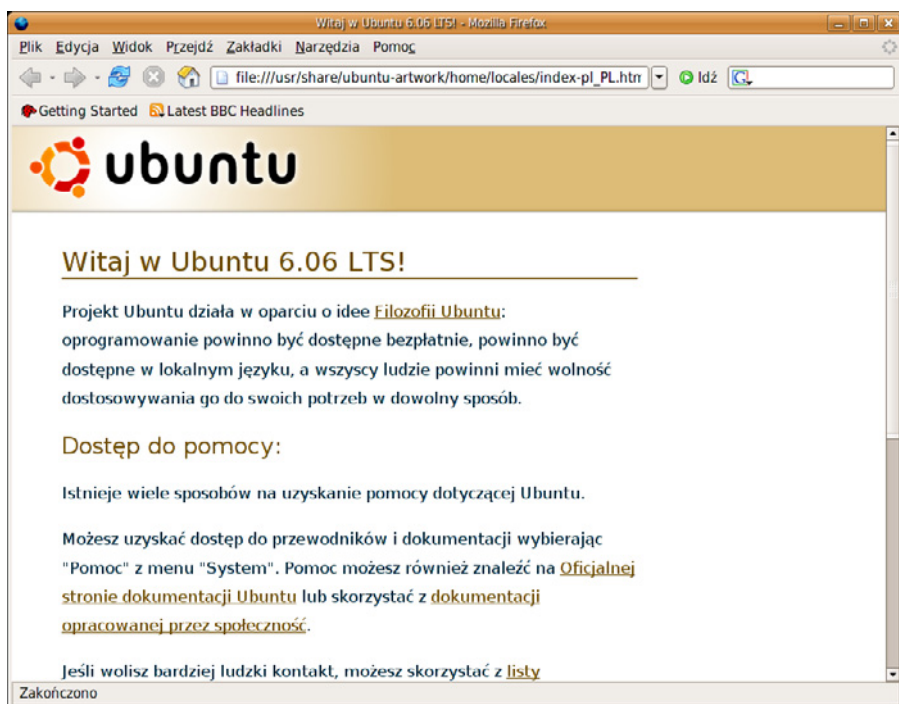
Większość użytkowników konfiguruje Firefox (i w ogóle przeglądarki internetowe) tak, aby wyświetlał stronę startową (zwaną też stroną domową) wskazaną przez internautę. Informacje, jak zmienić te ustawienia w programie, znajdują się w dalszej części rozdziału. Najpierw jednak omówię znaczenie poszczególnych elementów interfejsu programu.

Interfejs programu Firefox

Zdecydowana większość używanych dziś przeglądarek internetowych wygląda i działa podobnie, jednak na wypadek, gdyby komputer z Ubuntu miał być pierwszym takim urządzeniem użytkownika, zamieszczam trochę więcej informacji na temat interfejsu Firefoksa. Użytkownicy, którzy są zaznajomieni z przeglądarkami internetowymi i ogólnie z aplikacjami graficznymi, mogą pominąć ten fragment i przejść do punktu „Specjalne i niespecjalne funkcje programu Firefox”, w którym zamieszczam informacje na temat możliwości programu, z którymi być może nie zetknęli się, korzystając z innych aplikacji tego typu.

Standardowe elementy okna Firefoksa

Na rysunku 9.3 zaznaczyłem poszczególne elementy okna programu Firefox.

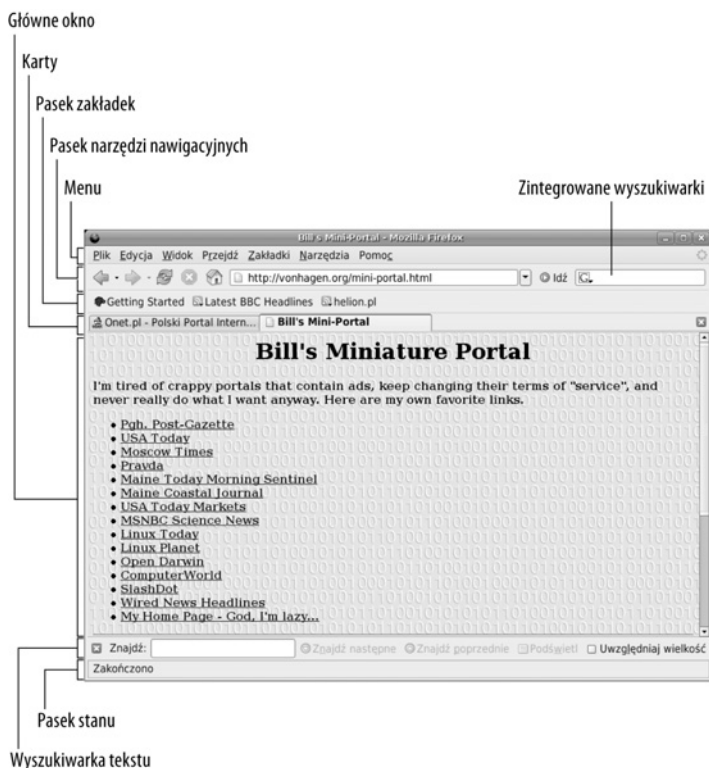


Rysunek 9.2. Pierwsze uruchomienie Firefoksa

Oto funkcje, jakie udostępniają wymienione elementy.

- ♦ **Pasek stanu:** dostarcza informacji na temat aktualnie wyświetlanej strony. Ikony wyświetlane z prawej strony informują o ewentualnych zabezpieczeniach strony, uruchomionych rozszerzeniach, postępie wczytywania strony, adresach łącz itp.
- ♦ **Główne okno:** w nim wyświetlana jest aktualnie odwiedzana strona.
- ♦ **Karty:** znajdują się na nich aktualnie wyświetlane strony internetowe. Karty pojawiają się wówczas, gdy program wyświetla więcej niż jedną stronę jednocześnie.
- ♦ **Pasek zakładek:** umożliwia szybki dostęp do najczęściej odwiedzanych stron internetowych, korzystanie z niego jest wygodniejsze niż używanie listy zakładek zapisanych w menu o tej samej nazwie.
- ♦ **Pasek narzędzi nawigacyjnych:** wyświetlane są na nim ikony umożliwiające wykonywanie podstawowych operacji podczas przeglądania internetu. W ustawieniach standardowych pasek to również łatwy dostęp do wyszukiwarki Google.
- ♦ **Menu:** po umieszczeniu nad nimi kursora myszy i wciśnięciu lewego przycisku zostają rozwinięte odpowiednie menu.
- ♦ **Wyszukiwarka:** umożliwia wyszukiwanie tekstu na aktualnie wyświetlanej stronie. To narzędzie jest wyświetlane wówczas, gdy użytkownik wybrał z menu *Edycja/ Znajdź na tej stronie* lub wcisnął kombinację klawiszy *Ctrl+F*.

Rysunek 9.3.
Elementy składowe
okna Firefoksa



Większość opisanych opcji można konfigurować zgodnie ze swoimi potrzebami (więcej informacji w dalszej części rozdziału): dla paska narzędzi nawigacyjnych i paska zakładek jest to zawartość, dla paska statusu, narzędzi nawigacyjnych i zakładek to, czy mają w ogóle być wyświetlane, a motywy pozwalają na modyfikację także ogólnego wyglądu programu.

Używanie myszy w programie Firefox

Korzystanie z myszy w programie Firefox nie różni się niczym od używania tego urządzenia peryferyjnego w innych aplikacjach z interfejsem graficznym, poza zawartością menu kontekstowego, wywoływanego wciśnięciem prawego przycisku. Oto kilka wskazówek dla osób, które rozpoczynają dopiero korzystanie z tego typu programów. Obecnie standardowa mysz posiada zwykle trzy przyciski, wciskając je (lub zwalniając), można uzyskać następujące efekty.

- ♦ **Lewy przycisk** umożliwia wybieranie różnych elementów. Kliknięcie lewym przyciskiem myszy wybranego menu spowoduje jego wyświetlenie, ponowne kliknięcie danego elementu powoduje rozwinięcie podmenu lub wykonanie polecenia. Wciśnięcie tego przycisku na którymś z przycisków paska narzędziowego spowoduje wykonanie operacji z nim skojarzonej, jeżeli użytkownik kliknie pasek wyszukiwarki Google, wówczas pojawi się w polu migający znak kursora i będzie można wprowadzić tam poszukiwany tekst.

- ♦ **Środkowy** przycisk ma działanie uzależnione od elementu, który zostanie kliknięty; jeżeli będzie to łącze do innej strony internetowej, to zostanie ona otwarta w nowej karcie; można też wkleić tekst w odpowiednie pole itd.
- ♦ **Prawy przycisk** powoduje wyświetlanie różnych menu, zależnych od elementu, który zostanie aktywowany. Dlatego noszą one nazwę **menu kontekstowych**. Na rysunku 9.4 widać dwa rodzaje menu kontekstowych wyświetlanych w programie Firefox, znajdujące się po prawej stronie pojawia się po wciśnięciu prawego przycisku, gdy kursor znajduje się nad zawartością strony internetowej; po lewej wyświetlane jest menu dostępne po kliknięciu łącza znajdującego się na danej stronie (wbrew temu, co sugeruje rysunek, nie można wyświetlić obu tych menu jednocześnie).



Rysunek 9.4. Menu kontekstowe w Firefoksie

Zawartość menu kontekstowych, takich jak widoczne na rysunku 9.4, może być zmieniana np. po zainstalowaniu dodatków i rozszerzeń do Firefoksa (więcej informacji na ten temat można znaleźć w punkcie „Instalowanie rozszerzeń do programu Firefox”). Odpowiednie rozszerzenia mogą nawet przekonfigurować działanie myszy: określają one nowe działania wykonywane po określonych ruchach urządzenia.

Specjalne i niespecjalne funkcje programu Firefox

Użytkownicy korzystający wcześniej ze zwykłych przeglądarek internetowych w innych systemach, takich jak Internet Explorer w Microsoft Windows czy we wcześniejszych wersjach Mac OS X, zauważą, że Firefox oferuje takie same możliwości jak te programy oraz posiada kilka zupełnie nowych, efektywnych i wydajnych rozwiązań. Niektóre z tych

interesujących możliwości, które jako pierwsze pojawiły się Firefoksie, nie są widoczne od razu po włączeniu programu.

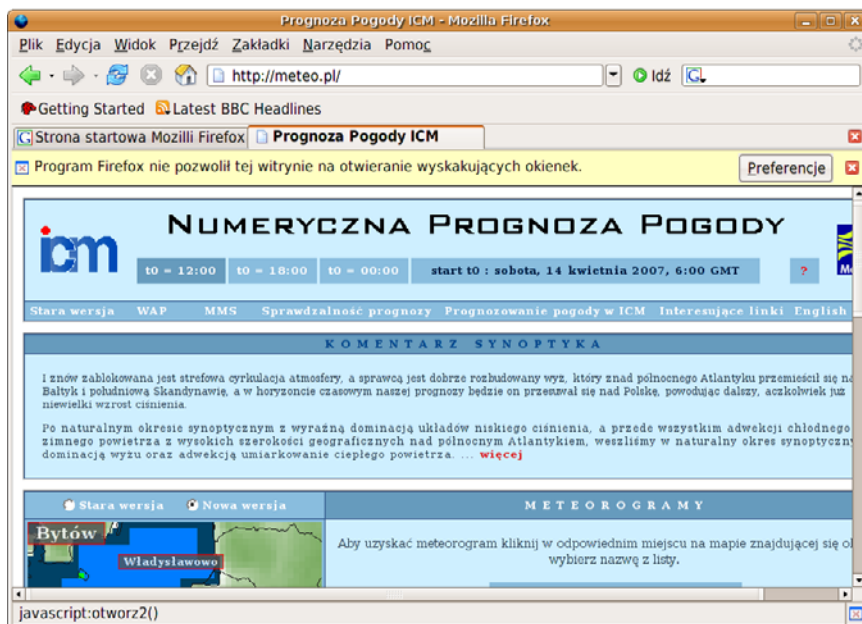
- ♦ **Przeglądanie w kartach:** umożliwia jednocześnie otwieranie wielu różnych stron internetowych bez konieczności uruchamiania osobnych okien dla każdej z nich. Każda wyświetlana strona jest reprezentowana przez osobną kartę widoczną tuż nad częścią okna, w której wyświetlana jest zawartość witryny. Można łatwo i szybko przełączać się pomiędzy kartami — wystarczy wskazać jedną z nich kursorem myszy. Ta funkcja jest tak użyteczna, że została zaimplementowana w większości używanych obecnie przeglądarek internetowych. Microsoft dodał tę opcję do Internet Explorera w wersji 7., jest też dostępna w programie Safari w Mac OS X. Aby zamknąć wybraną kartę, należy kliknąć ikonę ze znakiem „x” znajdującą się po prawej stronie panelu z kartami¹.
- ♦ **Panel boczny:** ułatwia przechodzenie pomiędzy zapisanymi stronami internetowymi. W Firefoksie dostępne są dwa różne panele boczne spełniające inne funkcje: pierwszy z nich wyświetla listę zakładek, na drugim zapisane są wszystkie odwiedzone strony (jest to tzw. **panel historii**). Panele te nie są wyświetlane domyślnie w programie; ale można je łatwo aktywować, wybierając z menu *Widok/Panel boczny*. Na rysunku 9.5 widać okno przeglądarki z wyświetlonym panelem zakładek. Aby zamknąć panel, wystarczy użyć przycisku „x” znajdującego się w jego górnej części.



Rysunek 9.5. Panel zakładek w programie Firefox

¹ W wersji 2.0 Firefoksa znak ten umieszczono na każdej uruchomionej karcie, co zwiększyło wygodę użytkowania programu — *przyp. tłum.*

- ♦ **Blokada wyskakujących okienek:** opcja zapobiegająca wyświetlaniu irytujących wyskakujących okienek, chyba że użytkownik zezwoli, aby na określonej stronie się pojawiały (można też całkowicie wyłączyć tę opcję). Na rysunku 9.6 widoczny jest komunikat o zablokowaniu wyskakującego okienka. Jeżeli użytkownik chciałby dostosować działanie programu w danym przypadku, wówczas wystarczy kliknąć przycisk *Preferencje*. Jeżeli użytkownik nie chce wprowadzać żadnych zmian w konfiguracji i wyłączyć baner, wówczas wystarczy wcisnąć przycisk „x” znajdujący po prawej stronie okna komunikatu. Metody dostosowywania wyświetlania przez Firefoksa wyskakujących okien opisuje w dalszej części rozdziału.



Rysunek 9.6. Komunikat o zablokowaniu wyświetlania wyskakujących okienek

- ♦ **Dynamiczne zakładki:** specjalny rodzaj zakładek tworzony dynamicznie na podstawie informacji z kanałów RSS dostępnych na wielu stronach. Tworzenie dynamicznych zakładek omawiam w dalszej części rozdziału.

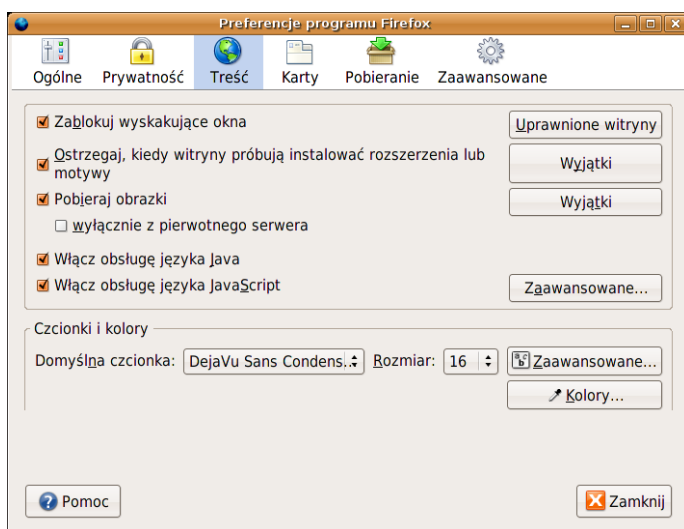
Firefox to potężna i elastyczna przeglądarka internetowa, wyposażona we wszystkie funkcje, których można oczekiwać od takiego programu. Z łatwością można by poświęcić całą książkę na opisy każdej opcji i każdego menu tego programu, ale zapewne spotkałoby się to z gwałtownym protestem wydawcy. Dlatego też zachęcamy użytkowników do samodzielnego odkrywania dostępnych funkcji i możliwości programu; niektóre z nich opisuje w dalszej części rozdziału.

Konfigurowanie Firefoksa

Firefox ma ogromne możliwości konfigurowania, posiada tyle pokręteł i przycisków, że użytkownik mógłby spędzić więcej czasu na jego ustawianiu niż na przeglądaniu internetu. Na szczęście, dzięki książkom, takim jak ta, oraz dokumentacji programu użytkownik może łatwo uniknąć takiego zagrożenia. Wiele przydatnych informacji można znaleźć na stronie <http://mozilla.org/support> (lub na stronie http://mozillapl.org/centrum_informacyjne/pomoc/).

Większość opcji konfiguracyjnych dostępna jest w oknie *Preferencje programu Firefox*, które można wywołać z menu *Edycja/Preferencje*. Na rysunku 9.7 przedstawiam to okno z programu Firefox 1.5, który jest domyślnie instalowany w Ubuntu 6.06 LTS. W kolejnych wersjach wygląd okna może się nieco różnić.

Rysunek 9.7.
Okno dialogowe
Preferencje programu
Firefox



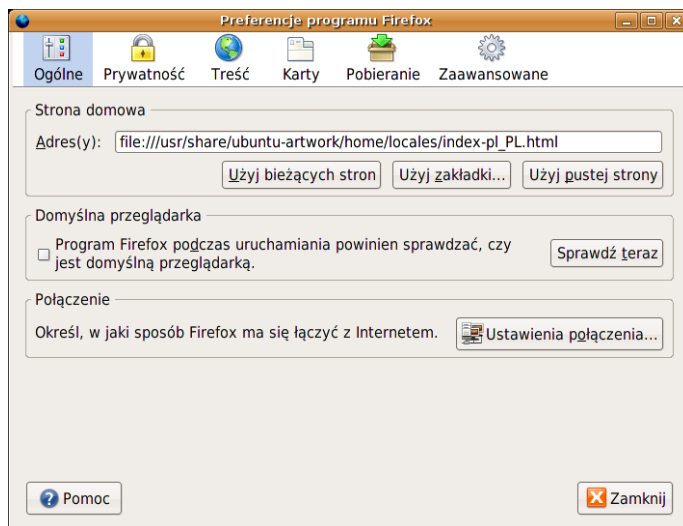
W następnych częściach rozdziału omówię podstawowe zagadnienia dotyczące konfigurowania programu z uwzględnieniem tych, które przede wszystkim mogą zainteresować czytelnika nawet wtedy, kiedy nie będzie chciał od razu ich wypróbować.

Ustawianie strony domowej

Strona domowa to ta witryna, która jest wyświetlana jako pierwsza po uruchomieniu Firefoksa (czy innej przeglądarki internetowej). Użytkownicy często ustawiają jako stronę domową witryny portali internetowych, takich jak Onet, Wirtualna Polska czy Interia, inni korzystają z serwisów z wiadomościami, dzięki czemu mają łatwy dostęp do informacji potrzebnych np. w pracy. Zależnie od potrzeb użytkownika, ustawienie strony domowej jest jednym z podstawowych aspektów personalizowania przeglądarki internetowej.

Aby ustawić stronę domową w Firefoksie, należy najpierw wyświetlić okno dialogowe *Preferencje programu Firefox* (menu *Widok/Preferencje*), następnie wcisnąć przycisk *Ogólne*, okno dialogowe powinno wówczas wyglądać tak, jak to na rysunku 9.8.

Rysunek 9.8.
Zakładka Ogólne
okna dialogowego
Preferencje programu
Firefox



Stronę domową można ustawić w tej zakładce w jeden z poniższych sposobów:

- ♦ wpisując adres strony w pole *Adres(y)*,
- ♦ wciskając przycisk *Użyj bieżących stron*, zostanie wówczas ustawiona aktualnie wyświetlana strona,
- ♦ wciskając przycisk *Użyj zakładki*, zostanie wówczas wyświetlone okno dialogowe z zapisanymi zakładkami, spośród których użytkownik będzie mógł wybrać odpowiednią stronę — wystarczy ją zaznaczyć i wcisnąć przycisk *OK*,
- ♦ wciskając przycisk *Użyj pustej strony*, co spowoduje, że Firefox nie będzie wyświetlał żadnej strony po uruchomieniu.

Po zastosowaniu którejś z powyższych metod do wskazania strony domowej należy kliknąć przycisk *Zamknij*. Wskazana strona zostanie wyświetlona podczas następnego uruchomienia programu (aby sprawdzić, czy została ustawiona odpowiednia strona, można też użyć kombinacji klawiszy *Alt+Home*).

Wyskakujące okienka, JavaScript i inne

Firefox udostępnia specjalny panel, w którym użytkownik może skonfigurować obsługę opcji oferowanych na współczesnych stronach internetowych, a także sposób ich działania po włączeniu. Można włączać, wyłączać i określać działanie następujących elementów.

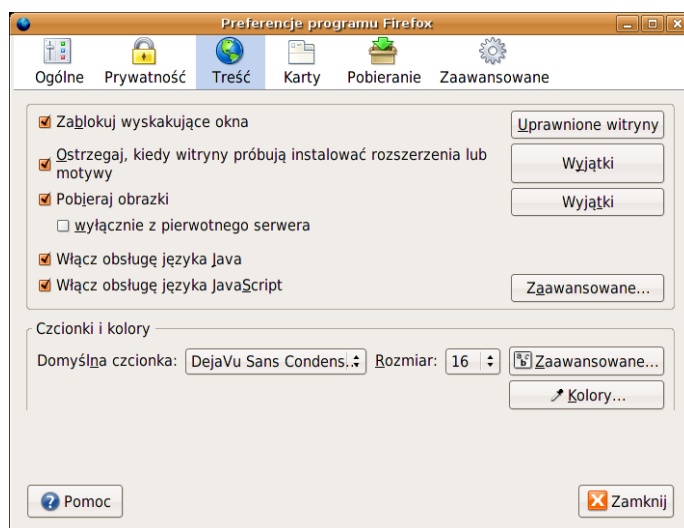
- ♦ **Zablokuj wyskakujące okienka** — zaznaczenie powoduje, że nie będą wyświetlane żadne wyskakujące okienka, z wyjątkiem tych witryn, którym użytkownik na to zezwoli.
- ♦ **Ostrzegaj, kiedy witryny próbują instalować rozszerzenia lub motywy** — decyduje o tym, czy na komputerze będzie instalowane jakiekolwiek oprogramowanie pochodzące wprost ze stron internetowych. Użytkownik może wskazać strony, dla których takie działanie będzie dozwolone.

- ♦ **Pobieraj obrazki** — określa, czy pobierać wszystkie obrazki z wyświetlanej strony internetowej, czy pobierać te, które znajdują się na innych stronach (serwerach) niż wyświetlana, oraz umożliwia wskazanie witryn, z których program będzie pobierał obrazki, jeżeli omawiana opcja zostanie wyłączona.
- ♦ **Włącz obsługę języka Java** — określa, czy program Firefox ma obsługiwać język Java, dzięki czemu ze stron będą pobierane aplety Javy, które następnie będą uruchamiane na komputerze.
- ♦ **Włącz obsługę języka JavaScript** — decyduje o włączeniu obsługi JavaScriptu, który umożliwia programowe generowanie zawartości i obsługę okien, a także określa obsługę wybranych opcji.

Aby włączyć, wyłączyć lub dostosować działanie którejs z wymienionych opcji, należy najpierw wyświetlić okno dialogowe *Preferencje programu Firefox* (menu *Widok/Preferencje*), a następnie przejść do zakładki *Treść* (patrz rysunek 9.9).

Rysunek 9.9.

*Włączanie
i dostosowywanie
opcji programu
Firefox*



Kiedy właściwa zakładka zostanie już wyświetlona, wówczas można zaznaczać (lub usuwać zaznaczenie) pola przy odpowiednich pozycjach. Aby edytować listę stron, które będą wyłączone z danej opcji, należy kliknąć przycisk *Uprawnione witryny* lub *Wyjątki* znajdujący się z prawej strony przy każdej opcji, a następnie wprowadzić nazwy witryn, które mają być wyłączone. Podobnie rzecz ma się w przypadku obsługi języka JavaScript: klikając przycisk *Zaawansowane*, użytkownik uzyska dostęp do opcji zgromadzonych w oknie *Zaawansowane ustawienia JavaScriptu*, gdzie będzie mógł szczegółowo określić ich działanie.

Po zakończeniu konfigurowania omówionych opcji należy wcisnąć przycisk *Zamknij*, wówczas zostaną zapisane. Niektóre z nich zostaną wprowadzone dopiero podczas kolejnego uruchomienia Firefoksa, dlatego warto od razu wyłączyć program i uruchomić ponownie.

Konfigurowanie ustawień z zakładki Prywatność

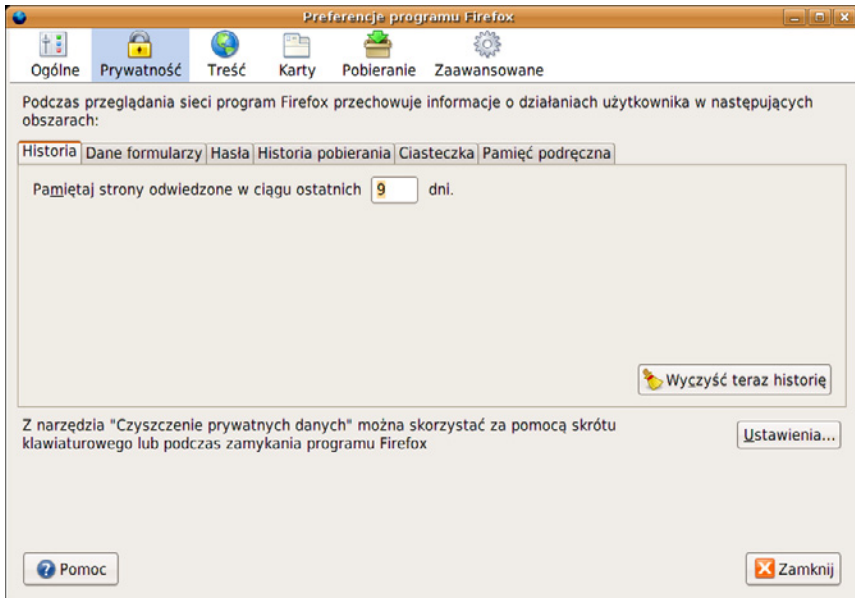
Podobnie jak w panelu *Treść*, również w przypadku ustawień dotyczących prywatności korzystania z programu użytkownik ma do dyspozycji osobną zakładkę. Jest to zakładka *Prywatność*, a poszczególne opcje zostały rozdzielone pomiędzy sześć zakładek znajdujących się w oknie. Aby zmodyfikować któreś z ustawień dotyczących prywatności korzystania z programu, należy najpierw przejść do odpowiedniej zakładki. Oto opcje dostępne w oknie *Prywatność*.

- ♦ **Historia** — umożliwia określenie, jak długo program ma przechowywać adresy URL odwiedzonych stron internetowych.
- ♦ **Dane formularzy** — użytkownik może zdecydować, czy program ma przechowywać dane umieszczane w polach tekstowych na stronach internetowych, co ułatwia późniejsze ich wypełnianie.
- ♦ **Hasła** — określa, czy Firefox ma zapamiętywać nazwy użytkownika i hasła wykorzystywane do logowania się na różnych stronach internetowych. Dostępne jest tu specjalne okno dialogowe (omówię je szerzej w dalszej części rozdziału), umożliwiające zarządzanie tymi informacjami.
- ♦ **Historia pobierania** — tu można zdecydować, czy program ma przechowywać informacje o plikach pobieranych z internetu. Dostępne opcje to: *po pomyślnym zakończeniu pobierania* (po zapisaniu pliku na dysku wpis o jego pobieraniu zostanie usunięty), *po wyjściu z programu Firefox* (wpis zostanie usunięty podczas wyłączania przeglądarki), *ręcznie* (użytkownik musi usunąć wpisy samodzielnie, w przeciwnym razie będą stale widoczne na liście).
- ♦ **Ciasteczka** — decyduje o tym, czy Firefox ma akceptować ciasteczka (można zdefiniować listę wyjątków) oraz jak długo mają być przechowywane. Pliki cookies, czyli ciasteczka, to niewielkie porcje informacji przechowywane na komputerze użytkownika, które np. umożliwiają rozpoznanie użytkownika, kiedy ten ponownie pojawi się na danej stronie. Podobnie jak w opcji zapisywania haseł, również tutaj dostępne jest okno dialogowe z dodatkowymi opcjami umożliwiającymi zaawansowane zarządzanie tymi plikami.
- ♦ **Pamięć podręczna** — opcja umożliwia zarządzanie wielkością pamięci podręcznej przeznaczonej na przechowywanie danych z odwiedzanych stron. Zapisywane są pliki HTML oraz grafiki ze stron, dzięki czemu podczas ponownego wczytywania witryny jest ona ładowana dużo szybciej, ponieważ część plików jest już na dysku komputera. Ta możliwość znacznie zwiększa prędkość działania, ale przy okazji zużywa przestrzeń dyskową.

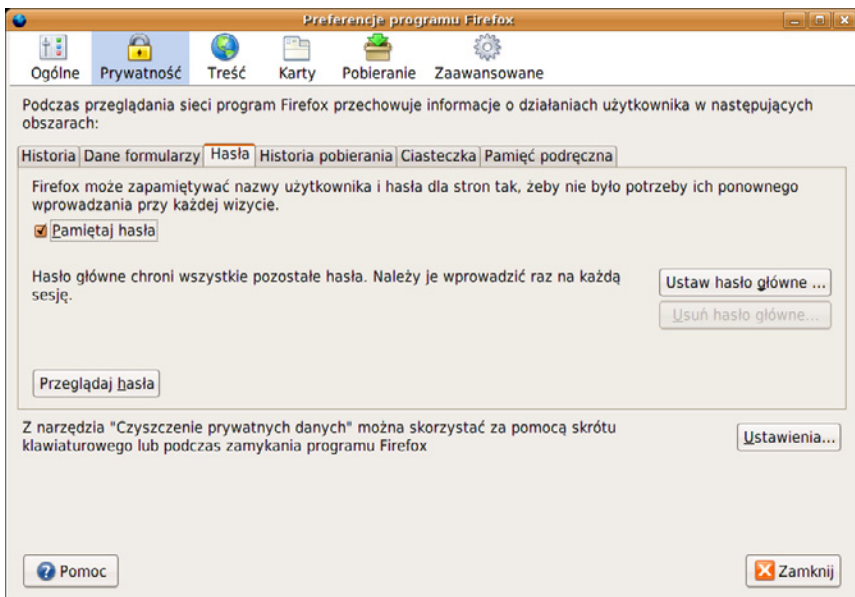
Większości tych opcji towarzyszy przycisk *Wyczyść teraz pamięć podręczną*, którego kliknięcie spowoduje szybkie usunięcie aktualnie zapisanych informacji związanych z omawianymi zagadnieniami prywatności.

Aby włączyć, wyłączyć lub zmienić ustawienia opisanych opcji, należy wybrać z menu *Widok/Preferencje*, a następnie przejść do zakładki *Prywatność* (patrz rysunek 9.10).

Po wyświetleniu zakładki użytkownik może sprawdzić, jakie ustawienia zostały wprowadzone. Na rysunku 9.11 widać zakładkę dotyczącą ustawiania haseł w programie.



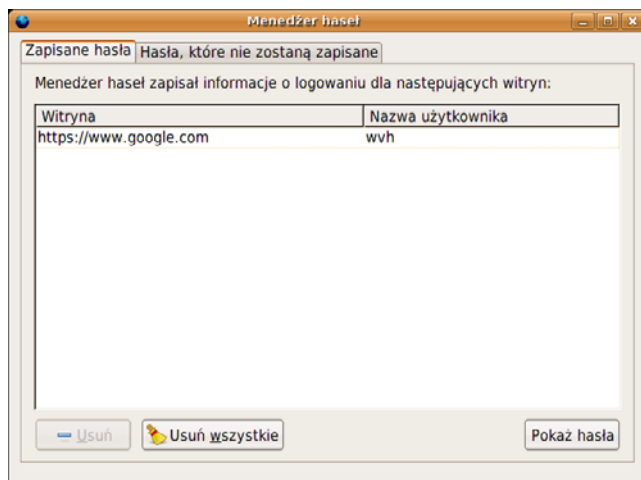
Rysunek 9.10. Dostosowywanie ustawień prywatności w programie Firefox



Rysunek 9.11. Zakładka Hasła

W tym przypadku należy kliknąć przycisk *Przeglądaj hasła* w oknie dialogowym *Preferencje programu Firefox*, zostanie wówczas wyświetlone okno *Menedżer haseł*, który umożliwia usunięcie zapisanych w programie nazw użytkownika oraz haseł (patrz rysunek 9.12).

Rysunek 9.12.
Zarządzanie hasłami
w programie Firefox



Aby usunąć nazwę użytkownika i hasło zapisane dla danej strony, należy zaznaczyć odpowiednią pozycję, a następnie kliknąć przycisk *Usuń* znajdujący się na dole okna menedżera (moja żona zrobiła tak kiedyś z danymi dostępowymi do serwisu aukcyjnego e-Bay...). Jak w większości programów z graficznym interfejsem, można zaznaczyć jednocześnie więcej niż jedną pozycję, wystarczy przytrzymać wciśnięty klawisz *Ctrl*. Po usunięciu odpowiednich wpisów można zamknąć okno, klikając kontrolkę ze znakiem „x”, zostanie na powrót wyświetlone okno *Preferencje programu Firefox*.

Kiedy użytkownik wprowadzi już wszystkie zmiany w preferencjach, można zamknąć okno, używając przycisku o nazwie *Zamknij* znajdującego się na dole. Niektóre z nich zostaną wprowadzone dopiero podczas kolejnego uruchomienia Firefoksa, dlatego warto od razu wyłączyć program i uruchomić ponownie.

Zakładki

W przeglądarkach internetowych zakładki są to zapisane adresy URL stron, które użytkownik wyświetlał i do których zamierza wracać. Po wybraniu zakładki program otworzy daną stronę — bez zbędnego zamieszania czy przeszukiwania innych witryn.



Użytkownicy Internet Explorera już pewnie zauważyli, że w tym programie również występują zakładki, tyle że pod nazwą *Ulubione*.

Zakładki, podobnie jak te w książkach, to wspaniałe rozwiązanie. Bez nich przeszukiwanie książki za każdym razem należałoby rozpoczynać od początku. Analogicznie działają zakładki w przeglądarkach internetowych; po znalezieniu ciekawej strony wystarczy ją zaznaczyć i będzie można do niej szybko powrócić w dowolnym momencie. Internauta z pewnością nie chce przerabiać kompleksowego przeszukiwania sieci czy odtwarzania całej sekwencji przechodzenia ze strony na stronę za każdym razem, kiedy chce przejść do wybranej witryny — dzięki zakładkom wystarczą dwa kliknięcia.

Firefox udostępnia dwa główne miejsca, w których można przechowywać zakładki: mogą one być dostępne w postaci rozwijanej listy znajdującej się w menu *Zakładki* lub też wprost na pasku, tuż pod narzędziami nawigacyjnymi w głównym oknie programu. Podczas gdy menu *Zakładki* udostępnia wszystkie zapisane zakładki, to pasek zakładek został zaprojektowany tak, aby ułatwić i przyspieszyć dostęp do tych często używanych. Pasek zakładek można porównać do półki z ulubionymi książkami, a menu do całej biblioteczki.

Z upływem czasu kolekcje zakładek zwykle bardzo się rozrastają, co sprawia, że zarządzanie nimi oraz odnajdywanie potrzebnych wpisów w razie potrzeby staje się krytycznym zadaniem dla wygodnego przeglądania zasobów sieci. Na szczęście, do Firefoksa dodano znakomitego menedżera zakładek, który znacznie ułatwia to zadanie. Poniżej omawiam następujące zagadnienia: jak tworzyć zakładki na różne sposoby, jak zarządzać nimi za pomocą menedżera zakładek, jak wykonać kopię zapasową zakładek i jak uzyskać do niej dostęp z innych przeglądarek internetowych, a także jak korzystać z fascynującej funkcji dostępnej w Firefoksie, czyli dynamicznych zakładek, których zawartość tworzona jest dynamicznie i automatycznie na podstawie zawartości wybranych stron internetowych.

Tworzenie zakładek w programie Firefox

Firefox umożliwia tworzenie zakładki dla aktualnie przeglądanej strony na trzy różne sposoby, są to kolejno:

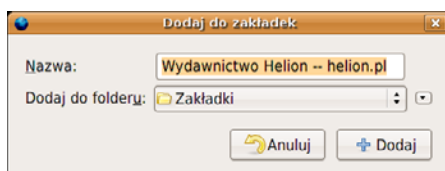
- ♦ kombinacja klawiszy *Ctrl+D* (wcisnąć i przytrzymać klawisz *Ctrl*, a następnie wcisnąć klawisz *D*), można jej użyć podczas wyświetlania strony, dla której ma zostać utworzona zakładka,
- ♦ kliknięcie prawym przyciskiem myszy wyświetlanej strony i wybranie z menu kontekstowego pozycji *Dodaj tę stronę do zakładek*,
- ♦ wybranie z menu *Zakładki* pozycji *Dodaj do zakładek*.

Można też utworzyć zakładkę z łącza znajdującego się na stronie, wystarczy kliknąć prawym przyciskiem myszy i z menu kontekstowego, które zostanie wyświetlone, wybrać pozycję *Dodaj odnośnik do zakładek*.

Niezależnie od wybranej metody, pojawi się okno dialogowe *Dodaj do zakładek* (patrz rysunek 9.13). W górnym polu wyświetlana jest nazwa, pod którą dana zakładka zostanie zapisana. Domyślnie jest to nazwa danej strony lub tekst znajdujący się w łączu, ale użytkownik może zmienić nazwę tak, aby była łatwa do zapamiętania.

Rysunek 9.13.

Okno dialogowe
Dodaj do zakładek

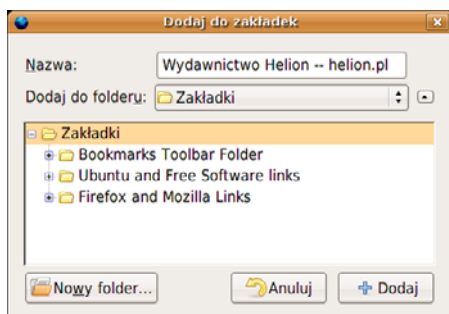


W dolnym polu wskazane jest miejsce, w którym dana zakładka zostanie zapisana. Listę można rozwinąć, zostaną wówczas wyświetlone wszystkie foldery znajdujące się w zakładkach i można będzie wybrać ten, w którym ma się znaleźć nowa pozycja. Po kliknięciu

małego trójkąta znajdującego się tuż za listą rozwijaną zostanie wyświetlona przeglądarka, za której pomocą użytkownik będzie mógł utworzyć nowy folder (patrz rysunek 9.14).

Rysunek 9.14.

Nawigowanie po folderach, w których będzie zapisana nowa zakładka



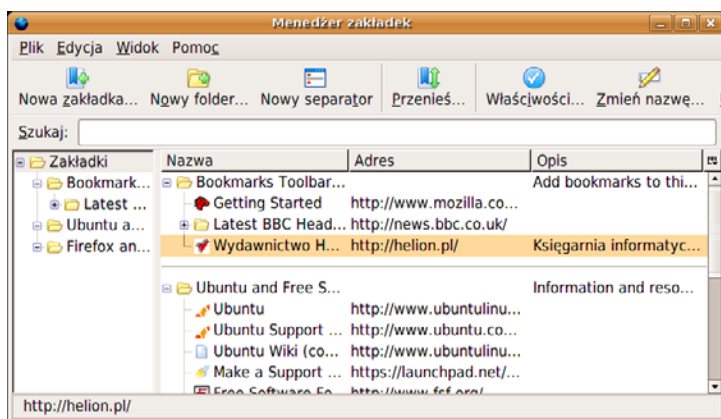
Po wprowadzeniu nowej nazwy, pod którą ma zostać zapisana nowa zakładka, oraz po wskazaniu odpowiedniego folderu należy kliknąć przycisk *Dodaj*, co zakończy całą operację. Okno dialogowe zostanie zamknięte, a użytkownik będzie mógł od tej chwili łatwo wyświetlać dodaną do zakładek stronę.

Zarządzanie zakładkami

Z upływem czasu kolekcje zakładek rozrastają się coraz bardziej, co sprawia, że odnalezienie tej, która jest właśnie potrzebna, staje się coraz trudniejsze, a to z kolei łamie główną ideę zakładek. Aby rozwiązać ten problem, w Firefoksie umieszczono doskonały menedżer zakładek, który ułatwia zarządzanie cenną kolekcją. Aby uruchomić ten program, należy z menu wybrać kolejno *Zakładki/Zarządzaj zakładkami*. Zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 9.15.

Rysunek 9.15.

Menedżer zakładek w programie Firefox



Panel widoczny po lewej stronie umożliwia zapoznanie się ze strukturą katalogów zakładek, panel po prawej stronie wyświetla szczegółowe informacje na temat każdego wpisu znajdującego się w danym folderze. Poszczególne zakładki można przesuwając między folderami metodą „przeciągnij i upuść” (w tym celu należy zaznaczyć wybraną zakładkę, wciskając lewy przycisk myszy i nie zwalniając go, przeciągnąć dany element w nowe

Kopia zapasowa zakładek

Ponieważ rozrastająca się kolekcja zakładek jest jedną z najcenniejszych rzeczy, które uzyskuje się podczas przeglądania internetu, należy się upewnić, że nie zostanie ona utracona w wyniku jakiejś awarii. W menedżerze zakładek w menu *Plik* dostępne są dwie opcje, *Import* i *Eksport*, ta druga umożliwia wykonanie kopii zapasowej aktualnie zapisanych zakładek i umieszczenie jej w osobnym pliku. Dobrym rozwiązaniem jest wykonywanie takiej kopii co tydzień lub dwa i nadawanie jej nazwy zawierającej datę utworzenia pliku (np. *zakladki-6-cze-2007*), co sprawi, że można będzie odróżnić je od siebie. Umożliwia to też odzyskanie przypadkowo usuniętej zakładki, wystarczy tylko pamiętać, kiedy ostatnio była używana.

Kopię zakładek można też wykonać wprost z linii poleceń, wystarczy skopiować plik *bookmarks.html* znajdujący się w katalogu *~/mozilla/firefox/*. Zwykle w tym katalogu jest też katalog *bookmark-backups*, w którym znajdują się kopie zakładek wykonane przez program.

miejsce). Użytkownik może też tworzyć nowe foldery, zakładki czy separatory (linie rozdzielające widoczne w menu *Zakładki*). W tym celu należy wybrać odpowiedni panel i wykonać jedną z następujących czynności:

- ♦ wybrać odpowiednią ikonę znajdującą się na pasku narzędzi menedżera zadań,
- ♦ zaznaczyć odpowiedni folder, kliknąć prawym przyciskiem myszy i wybrać odpowiednie polecenie z menu kontekstowego,
- ♦ wybrać odpowiednie polecenie z menu *Plik*.

Wszelkie wprowadzone zmiany od razu zostaną zastosowane w menu *Zakładki* oraz na pasku zakładek. Kiedy użytkownik wprowadzi już wszelkie potrzebne zmiany, należy wybrać z menu polecenie *Plik/Zamknij*, aby zamknąć program.

Niezależnie od przyjętego modelu organizowania zakładek, trzeba stosować się do generalnej zasady, aby unikać przeładowania ich listy. Warto grupować zakładki powiązane tematycznie w jednym folderze, dzięki czemu łatwiej będzie odszukać potrzebną pozycję, a także uniknąć umieszczenia dwukrotnie tego samego wpisu.

Tworzenie dynamicznych zakładek

Dynamiczne zakładki to jedno z ostatnich — i najbardziej użytecznych — usprawnień przeglądania sieci wprowadzonych przez twórców Firefoksa. Korzystają one z technologii RSS. Skrót ten, zależnie od tego, kogo zapytamy, jest rozwijany jako *Really Simple Syndication*² lub *Rich Site Summary*³. Wiele stron internetowych korzysta z technologii RSS do publikowania podsumowania wprowadzonych zmian (tzw. czytelniki RSS), dzięki czemu na innych stronach można zamieszczać bezpośrednie łącza do nich. Dynamiczne zakładki to specjalny rodzaj zakładek, w których wyświetlana jest treść nagłówków zamieszczona w kanałach RSS dostępnych z paska zakładek.

Strony wykorzystujące technologię RSS można łatwo rozpoznać po tym, że w pasku adresu, przy jego prawej krawędzi widoczna jest specjalna ikona (patrz rysunek 9.16).

² W wolnym tłumaczeniu: naprawdę proste publikowanie — *przyp. tłum.*

³ W wolnym tłumaczeniu: pełne podsumowanie strony — *przyp. tłum.*

Rysunek 9.16.

Ikona informująca o dostępności kanałów RSS na danej stronie



Aby dodać dynamiczną zakładkę, wystarczy kliknąć tę ikonę. Zostanie wówczas wyświetlone okno *Dodaj dynamiczną zakładkę* (różniące się od okna *Dodaj zakładkę* tylko nazwą), w którym użytkownik będzie mógł dopasować ustawienia. Po dodaniu zakładki i zamknięciu okna zostanie ona wyświetlona na pasku zakładek, a po kilku sekundach, potrzebnych na pobranie przez Firefoksa nagłówek wiadomości, będzie można wyświetlić jej zawartość, tak jak każde inne menu w przeglądarce (patrz rysunek 9.17).

Rysunek 9.17.

Dynamiczne zakładki w programie Firefox



Pasek osobisty jest najlepszym miejscem do umieszczenia dynamicznych zakładek, ponieważ zawarte w nich są najnowsze informacje, którymi zapewne użytkownik będzie zainteresowany.

Dynamiczna zakładka widoczna w oknie na rysunku 9.17 odpowiada stronie wyświetlanej w głównym oknie przeglądarki. Bliższe przyjrzenie się zakładce i stronie pozwoli stwierdzić, że ich treści pokrywają się ze sobą, a dokładniej, że nagłówki umieszczone na stronie są takie same jak widoczne w rozwiniętej dynamicznej zakładce. Kiedy zawartość strony zostanie zmieniona, znajdzie to odzwierciedlenie również w zakładce. Czy może być coś łatwiejszego? Może następna wersja Firefoksa będzie mogła też odczytywać treść i informować czytelnika, na które fragmenty powinien zwrócić uwagę...

Ulepszanie Firefoksa

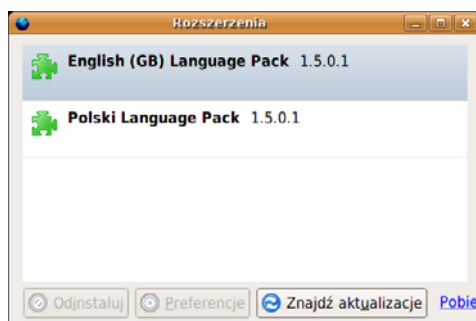
Dotychczas omówione zostały funkcje i możliwości programu, który nie został poddany żadnym zabiegom ze strony użytkownika. Jednak Firefox również w tym zakresie dostarcza narzędzi, za których pomocą można zwiększyć jego funkcjonalność. Do tego celu służą **rozszerzenia** (dodające nowe funkcje do programu) oraz **motywy** (umożliwiające zmianę wyglądu ikon, kolorów, menu, okien itd. widocznych w programie). Poniżej opisuję, w jaki sposób zainstalować te elementy, co pomoże jeszcze bardziej dostosować program do własnych potrzeb. W wielu przypadkach rozszerzenia Firefoksa dostarczają tak przydatnych możliwości, że warto byłoby przeznaczyć na nie osobny program, tu jednak udało się je świetnie zintegrować z działającą przeglądarką, podnosząc w ten sposób komfort korzystania z internetu. Niemal każdy użytkownik ma też własną wizję tego, jak powinna wyglądać przeglądarka internetowa. Motywy w Firefoksie — bardzo podobne do skórek, z którymi być może zetknął się użytkownik, korzystając z popularnych odtwarzaczy muzycznych, takich jak WinAmp czy xmms — umożliwiają dostosowanie wyglądu programu do oczekiwań użytkownika, dzięki czemu jeszcze lepiej spełnia on swoje zadania.

Instalowanie rozszerzeń do programu Firefox

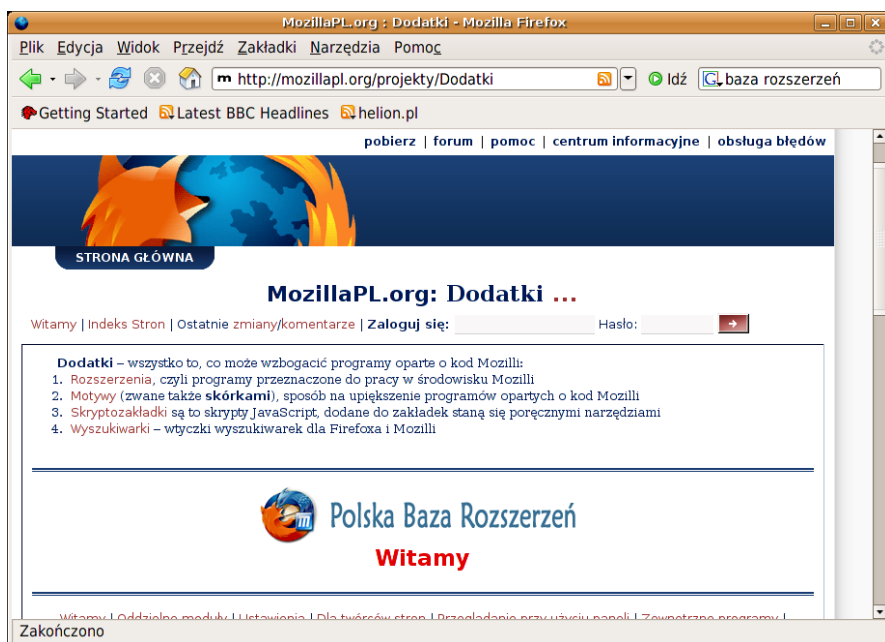
Dodawanie nowych rozszerzeń do programu jest bardzo proste, ponieważ Firefox może je automatycznie pobrać i zainstalować, użytkownikowi pozostaje jedynie wybrać te, które będą dla niego najlepsze, co ze względu na ich ogromną ilość wcale nie musi być prostym zadaniem. Ale od czego są godziny od północy do rana?

Aby dodać nowe rozszerzenie, należy wybrać kolejno z menu *Narzędzia/Rozszerzenia*. Zostanie wówczas wyświetlone okno dialogowe *Rozszerzenia* widoczne na rysunku 9.18, zawierające listę aktualnie zainstalowanych dodatków.

Rysunek 9.18.
Okno dialogowe
Rozszerzenia
w programie Firefox



Następnie należy użyć łącza *Pobierz więcej rozszerzeń* znajdującego się w prawym dolnym rogu okna, zostanie wówczas uruchomione nowe okno przeglądarki, w którym zobaczymy stronę z dodatkami do programu. Dostępna jest również polska baza rozszerzeń do programów Mozilli, wystarczy zajrzeć na stronę <http://mozillapl.org/projekty/Dodatki> (patrz rysunek 9.19).



Rysunek 9.19. Polska baza dodatków do programów tworzonych w oparciu o kod Mozilli

Na stronie zamieszczono ogólne informacje o dodatkach, które można tu znaleźć; wyjaśniono znaczenie używanych ikon oraz zawarto kilka przydatnych linków do innych stron. Zgromadzone dodatki zostały podzielone na różne kategorie (*Oddzielne moduły, Ustawienia, Dla twórców stron* itd.), dzięki czemu można szybciej odnaleźć odpowiednie rozszerzenie. Kiedy zostanie ono już znalezione, trzeba kliknąć łącze, co spowoduje wyświetlenie części strony z opisem dodatku.

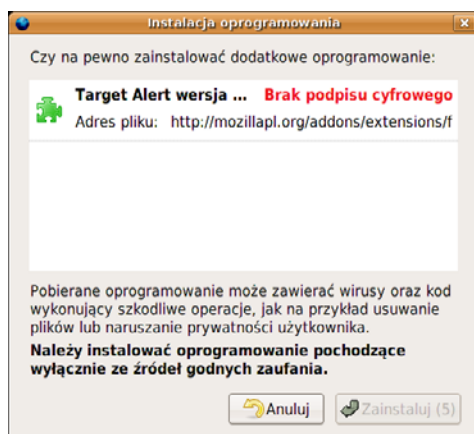
Aby go zainstalować, należy kliknąć ikonę *Instaluj*. Zostanie wówczas wyświetlona strona z obszerniejszym opisem dodatku, informacjami na temat działania i instalowania. W dolnej części znajduje się sekcja *Instalacja*, za której pomocą można dodać konkretny dodatek do przeglądarki. Wystarczy wybrać odpowiednią dla systemu i wydania programu wersję i kliknąć ikonę *Instaluj*. Zostanie wyświetlone okno dialogowe widoczne na rysunku 9.20.

Po pobraniu i zainstalowaniu rozszerzenia zawartość okna zostanie zaktualizowana (patrz rysunek 9.21). Pojawi się w nim również informacja, że dodatek zacznie działać po ponownym uruchomieniu przeglądarki.

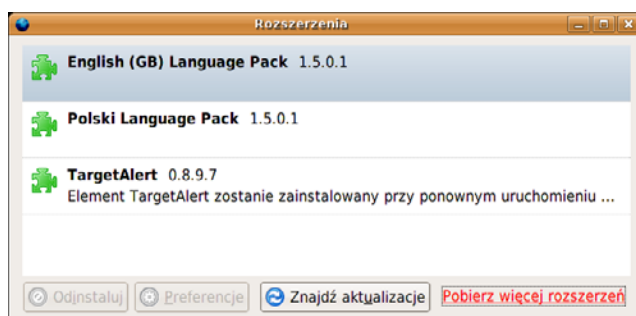
Po zakończeniu instalowania dodatku należy zamknąć okno przeglądarki, w którym wyświetlana jest strona z rozszerzeniami, a także zamknąć okno dialogowe *Instalacja oprogramowania* (należy użyć ikony ze znakiem „x” znajdującej się górnym prawym rogu

Rysunek 9.20.

Okno dialogowe
Instalacja
oprogramowania

**Rysunek 9.21.**

Zaktualizowana
zawartość okna
dialogowego
Rozszerzenia



okna). W oknie będzie wyświetlany komunikat przypominający o tym, że aby nowo zainstalowane dodatki zaczęły działać, konieczne jest ponowne uruchomienie przeglądarki. Pozwoli to również upewnić się, że Firefox działa właściwie.



Istnieje tak ogromna liczba różnorodnych rozszerzeń do Firefoksa, że bardzo kuszący jest pomysł, aby zainstalować je wszystkie jednocześnie i sprawdzić, które są przydatne. Jednak sugeruję, aby zachować kontrolę i jednocześnie instalować nie więcej niż dwa dodatki, a to z dwóch powodów. Po pierwsze, dlatego że niektóre rozszerzenia mogą powodować konflikty z innymi, co z kolei może odbić się na stabilności działania samego Firefoksa. Po drugie, po zainstalowaniu zbyt dużej ilości rozszerzeń trudno stwierdzić, które jest odpowiedzialne za konkretne zmiany wprowadzone w programie. To z kolei może skomplikować wybór odpowiednich rozszerzeń, które mają zostać zainstalowane na wielu komputerach; użytkownik straci zbyt wiele czasu na ustalaniu, których dodatków potrzebuje, a których nie.

Kiedy po zapoznaniu się z działaniem danego dodatku stwierdzimy, że nie spełnia on naszych oczekiwań, możemy anulować instalację. W tym celu należy ponownie wyświetlić okno *Instalacja oprogramowania*, zaznaczyć dane rozszerzenie, a następnie użyć przycisku *Odinstaluj*.

Dodawanie nowych motywów do programu Firefox

Dodawanie nowych motywów równie proste jak instalowanie rozszerzeń. Eksperymentowanie z nowymi motywami jest właściwie łatwiejsze od testowania rozszerzeń (co opisałem szczegółowo powyżej), ponieważ jednocześnie może być wyświetlany tylko jeden motyw.

Aby dodać nowy motyw, należy kolejno wybrać z menu *Narzędzia/Motywy*. Zostanie wówczas wyświetlone okno dialogowe *Motywy* (patrz rysunek 9.22), w którym widoczne będą wszystkie aktualnie zainstalowane.

Rysunek 9.22.

*Okno dialogowe
Motywy w programie
Firefox*



Następnie należy użyć łącza *Pobierz więcej motywów*, znajdującego się w prawym dolnym rogu okna, a zostanie otwarte nowe okno przeglądarki, w którym będzie można zobaczyć motywy aktualnie dostępne w witrynie Mozilli (należy przejść do działu *Themes*).

Ponieważ proces pobierania i instalowania motywów jest w swej istocie taki sam jak pobieranie i instalowanie rozszerzeń, nie będę zanudzać czytelników opisywaniem go w szczegółach. Główna różnica jest widoczna już po pobraniu i zainstalowaniu motywu. Ponieważ każdy z nich zawiera kompletny zestaw elementów graficznych dla programu, tylko jeden motyw może być jednocześnie wyświetlany. Aby zastosować nowy motyw, należy go zaznaczyć i użyć przycisku *Zastosuj* (patrz rysunek 9.23).

Rysunek 9.23.

*Wybieranie
nowego tematu*



Po ponownym uruchomieniu Firefoksa zostaną zastosowane zupełnie nowe ikony, wygląd menu i inne elementy graficzne. Na rysunku 9.24 widać interfejs programu Firefox po pobraniu, zainstalowaniu i zastosowaniu popularnego motywu graficznego Noia 2.0 (eXtreme). Zmiany zostały zastosowane wobec ikon, czcionek i kolorów. Przyjemne urozmaicenie.



Rysunek 9.24. Użycie nowego motywu w Firefoksie

Podobnie jak w przypadku rozszerzeń, również tematy można odinstalować, korzystając z opcji *Odinstaluj* dostępnej w oknie dialogowym *Motywy*. Nie jest dobrym pomysłem usuwanie aktualnie używanego tematu; najpierw należy włączyć inny, ponownie uruchomić Firefoksa i dopiero wówczas usunąć instalację motywu, z którego użytkownik nie chce dalej korzystać.

Podsumowanie

W niniejszym rozdziale przedstawiłem informacje na temat konfigurowania, używania i dostosowywania programu Firefox — najpopularniejszej przeglądarki internetowej dla systemów linuksowych, której udział w rynku, w odróżnieniu od innych dostępnych dla systemów Microsoft Windows czy Mac OS X, wciąż się zwiększa. Firefox to ukoronowanie lat ewoluowania i tworzenia przeglądarek przez społeczności zajmujące się otwartym oprogramowaniem. Po zapoznaniu się z Firefoksem w systemie Ubuntu czytelnik zapewne postanowi zainstalować go również na innych komputerach, z których korzysta. Używanie jednego programu na wielu komputerach i w różnych systemach operacyjnych to z pewnością wygodne rozwiązanie.

W następnym rozdziale opisuję kolejne aspekty podstawowych możliwości Ubuntu, czyli tworzenie i publikowanie dokumentów. W rozdziale 7. omówiłem korzystanie z edytorów tekstowych, zupełnie wystarczających do tworzenia notatek czy innych dokumentów, które nie wymagają używania różnych czcionek, stylów i innych elementów formatowania. Ale w pewnym momencie wszyscy chcą tworzyć wymyślne pisma z wykorzystaniem wielu krojów czcionek, stylów i formatowania. W rozdziale 10., „Tworzenie i publikowanie dokumentów”, omawiam wolne oprogramowanie dostępne w Ubuntu, które umożliwia szybkie i łatwe tworzenie dokumentów sformatowanych w zaawansowany sposób.

Rozdział 10.

Tworzenie i publikowanie dokumentów

W tym rozdziale:

- ♦ Języki znaczników TeX i LaTeX
- ♦ Korzystanie z programu OpenOffice.org
- ♦ Korzystanie z programu Scribus
- ♦ Inne procesory tekstu dla Linuksa

W rozdziale 7. omówiłem edytory tekstu, które świetnie się sprawdzają, gdy trzeba utworzyć dokument w rodzaju listy zakupów, listy spraw do załatwienia czy innych notatek. Do kreowania bardziej skomplikowanych dokumentów większość użytkowników wykorzystuje procesory¹ tekstu, czyli programy umożliwiające stosowanie różnych czcionek, różnych stylów (podkreślenia, pogrubienia czy kursywy), automatyczne tworzenie list wypunktowanych i numerowanych, a także pozwalające na edytowanie układu strony za pomocą takich elementów jak tabele, kolumny tekstu itp.

W niniejszym rozdziale objaśniam, jak korzystać z różnych wersji procesorów tekstu w systemie Ubuntu. Na początku omówię języki znaczników, czyli metody tworzenia zaawansowanych typograficznie tekstów za pomocą edytorów dostępnych w Ubuntu. Choć języki te należą do starej szkoły, wciąż są popularne, a w Ubuntu można skorzystać z systemu LaTeX do tworzenia tekstu opartego na znacznikach, w dodatku o otwartym kodzie. Można też znaleźć klon szacownego programu dla Uniksa, pochodzący z dawnych dni.

¹ Programy służące do obróbki tekstu dzielą się na edytory i procesory; te pierwsze umożliwiają edycję i wprowadzanie tekstu. Procesory tekstu z kolei to zaawansowane edytory, które dodatkowo umożliwiają zmianę wyglądu dokumentu. Nazwa edytory przyjęła się powszechnie zarówno dla właściwych edytorów, jak i procesorów. Dla potrzeb tego rozdziału zachowamy jednak właściwą terminologię — *przy. tłum.*



Jeżeli czytelnik po tych dwóch akapitach pomyśli: „O czym on do diaska mówi? Przecież wśród języków wykorzystujących znaczniki teraz liczą się tylko HTML i XHTML?” — może poczuć się rozczarowany. Ponieważ na temat tych języków można znaleźć wiele książek, a także dlatego, że są one ukierunkowane na tworzenie stron internetowych, a nie dokumentów, ich szczegółowy opis nie został zamieszczony w niniejszej książce.

Następnie omówię bardziej znajome typy graficznych procesorów, a dokładniej OpenOffice.org Writer — składnik całego pakietu OpenOffice.org. Jego poszczególne elementy zostaną opisane w kolejnych rozdziałach. OpenOffice.org to znakomity pakiet programów, a zamieszczone w niniejszym rozdziale jego omówienie powinno sprawić, że nawet zdeklarowani zwolennicy procesorów, takich jak Microsoft Word, Corel Word Perfect czy Mac OS X Nisus Writer, powinni poczuć się jak w domu

Na końcu omówię program do elektronicznego składu dokumentów — znakomity Scribus. Kiedy trzeba wykonać poważną pracę i popracować nad profesjonalnym wyglądem tworzonego dokumentu, wówczas procesory tekstu przestają być wystarczające, ponieważ nie oferują odpowiednich narzędzi do kontrolowania układu strony. Programy do składu znacząco ułatwiają tę pracę, ponieważ za ich pomocą można tworzyć dokumenty zawierające wiele różnorodnych elementów, to one umożliwiają manipulowanie tekstem, wstawianie grafiki, słowem wszystko, co sprawia, że dokument wygląda, tak jak powinien. Scribus jest relatywnie nowym programem, ale imponującym. Użytkownicy takich programów jak AdobeInDesign, Adobe PageMaker i Quark XPress — czapki z głów! Nadchodzi Scribus! W dodatku w ogóle nie trzeba przejmować się zawartością portfela.

Języki znaczników w Ubuntu

W dawnych czasach, kiedy ludzie używali terminali, do tworzenia dokumentów stosowano edytory tekstu, a w samych dokumentach umieszczano specjalne polecenia, dzięki którym inne programy wiedziały, jak mają formatować dany tekst. Format tworzenia takich dokumentów nazywamy językiem znaczników, a jego najpopularniejszym przykładem jest HTML. Oczywiście, ten język jest przeznaczony do tworzenia dokumentów, które mają znaleźć się na stronach internetowych, ale zasada działania jest taka sama. Opracowywanie dokumentów za pomocą języka znaczników wciąż jest popularne na uczelniach oraz w wydawnictwach, ponieważ wielu dziennikarzy i wiele instytucji wciąż używa takiego formatu jako najmniejszego wspólnego mianownika. Mogą łączyć dokumenty pochodzące z różnych źródeł w jedną, spójną stylistycznie publikację, bez konieczności martwienia się, której wersji Worda użyto, jakich czcionek itp.

TeX (należy wymawiać jako *tech* lub *tek*) to program do składu oraz związany z nim język znaczników, oba zaprojektowane do tworzenia bardzo wysokiej jakości dokumentów, zwłaszcza naukowych i technicznych zapisów. Wybitny naukowiec, Donald Knuth napisał TeX w roku 1977, głównie dlatego, że potrzebował odpowiedniego narzędzia do swojej trzytomowej *Art of the Computer Programming*. Jakby nie było wystarczająco fajnie napisać własny system do formatowania dokumentów, został on w dodatku napisany w języku, który również został utworzony przez Knutha, a mianowicie w języku WEB, będącym mieszanką dokumentacji i kodu Pascala połączonych w jednym pliku. Kod źródłowy jest ekstrahowany z dokumentacji WEB za pomocą programu o nazwie tangle, a sformatowany kod źródłowy oraz zintegrowana dokumentacja są ekstrahowane za

pomocą programu weave (też w formacie TeX). Oczywiście, nie ma to znaczenia, jeżeli czytelnik chce po prostu używać TeX, ale jest ciekawe i brzmi jak jeden z dawnych odcinków serialu *Komputerowi naukowcy oszaleli* (wkrótce na DVD), dlatego może być ciekawym dodatkiem do zbioru interesujących, choć niemających kluczowego znaczenia informacji.

Obecnie istnieje wiele różnych implementacji systemu TeX, wszystkie wywodzą się z oryginalnego dokumentu *tex.web*, przełożono je na inne języki oraz dodano dodatkowe możliwości. Kod źródłowy systemu TeX jest wolno dostępny. Wszystko, co wywodzi się z TeX-a i chce używać jego nazwy, musi przejść przez specjalny test formatowania, wymyślną torturę o nazwie TRIP. Umieszczona w repozytoriach Ubuntu wersja TeX-a to — jak w większości obecnych systemów Linux — *teTeX*. Nazwa (*te*) pochodzi od Thomassa Essera, który przystosował system do działania na Unikсах. Strona domowa projektu to <http://tug.org/tetex/>.

Instalowanie systemu TeX w Ubuntu

Ponieważ TeX to narzędzie dla wybranych, nie jest instalowany domyślnie w Ubuntu. Aby dodać go do systemu, należy uruchomić menedżera pakietów Synaptic (z menu *System/ Administracja*), następnie przejść do sekcji *Narzędzia TeX* znajdującej się w lewym panelu okna. Dostępnych jest wiele różnych pakietów związanych z systemem TeX, ale najważniejsze z nich to:

- ♦ *tetex-base*,
- ♦ *tetex-bin*,
- ♦ *tetex extra*.

Można również zainstalować pakiet *tetext-doc*, zawierający dokumentację typu HOWTO oraz inne dokumenty poświęcone systemowi TeX. Na rysunku 10.1 przedstawiam instalowanie trzech wymienionych wyżej pakietów.

Zaznaczenie do zainstalowania tych trzech pakietów będzie wymagało również dodania do systemu kilku innych. Kiedy już wszystkie niezbędne pakiety zostaną umieszczone w systemie, można poeksperymentować z systemem TeX.

W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, zamieszczam więcej informacji na temat instalowania programów za pomocą Synaptica.

Korzystanie z programu TeX

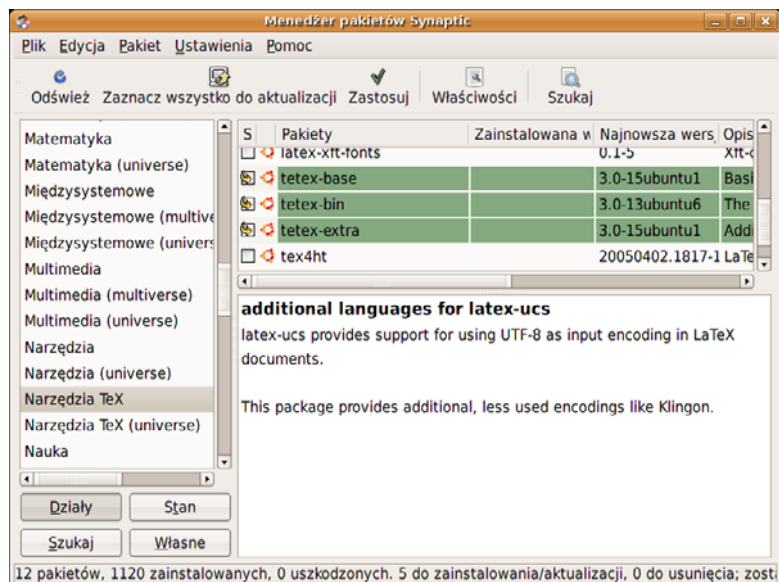
Znaczniki używane w TeX-u są proste. Oto przykład kodu dla dokumentu TeX, który spowoduje wyświetlenie napisu „Witaj świecie!”:

```
Witaj świecie!  
\bye
```

Aby sformatować ten tekst, można po prostu użyć polecenia `tex` wprost z linii poleceń Ubuntu, podając dodatkowo nazwę pliku jako argument. Na ekranie monitora powinno pojawić się coś takiego:

Rysunek 10.1.

Instalowanie systemu TeX za pomocą menedżera pakietów Synaptic



```
$ tex witaj.txt
```

```
This is TeX, Version 3.141592 (Web2C 7.5.4)
```

```
(./witaj.txt [1] )
```

```
(see the transcript file for additional information)
```

```
Output written on witaj.dvi (1 page, 228 bytes).
```

```
Transcript written on witaj.log.
```

Pliki tworzone za pomocą TeX mają format DVI (ang. *Device Independent*). Dane wyjściowe DVI zawierają informacje na temat czcionek i pozycji znaków (a także samą treść dokumentu), dlatego muszą zostać przetworzone przez inny program, który odczyta, w jaki sposób znaki mają być wyświetlane, dzięki czemu będzie można utworzyć plik w odpowiednim formacie czy języku używanym przez drukarkę.

Jak widać z przykładu „Witaj świecie”, do tworzenia prostych dokumentów wcale nie trzeba wielu poleceń. Kluczowym słowem jest prostota. Aby utworzyć plik, w którym używane są różne czcionki, odmienne położenie tekstu i jego podział, kod powinien wyglądać mniej więcej tak:

```
\centerline{To jest tytuł}
\smallskip
```

```
\centerline{Bill von Hagen}
\centerline{\tt 24 marca, 2006}
```

```
\bigskip
```

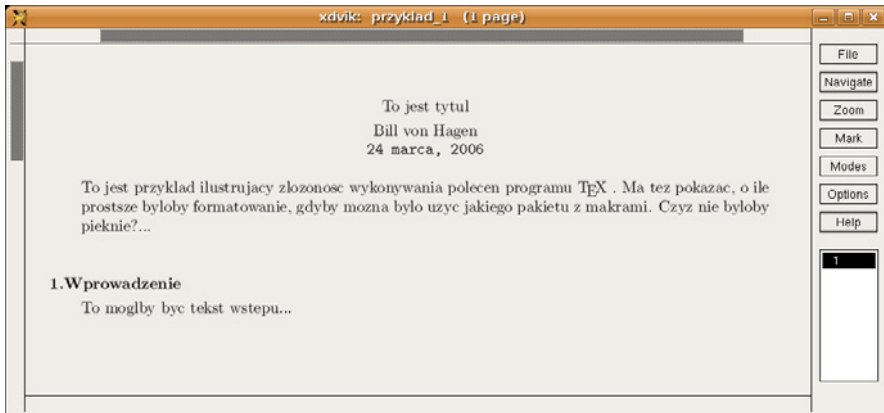
```
{\narrower\noindent
```

```
To jest przykład ilustrujący złożoność wykonywania poleceń programu \TeX\ . Ma też
pokazać, o ile prostsze byłoby formatowanie, gdyby można było użyć jakiegoś pakietu
z makrami. Czyż nie byłoby pięknie?... \par}
```

```
\bigskip
```

```
\beginsection 1.Wprowadzenie
\smallskip
To moglby byc tekst wstepu...
\bye
```

Na rysunku 10.2 przedstawiam podgląd pliku DVI utworzonego w systemie TeX.



Rysunek 10.2. Przykładowy plik wyjściowy TeX

Wprowadzenie do znaczników LaTeX

Należy zmierzyć się z prawdą: niezależnie od tego, jak dobrze wygląda plik wyjściowy, tworzenie dokumentów za pomocą poleceń niskiego poziomu w systemie TeX może być udręką zwłaszcza wtedy, kiedy dokument wymaga zaawansowanego formatowania. Aby to uprościć, powstało wiele różnych zestawów makr, najbardziej znane (o nazwie LaTeX) są autorstwa Lesliego Lamporta. Pakiety makr obsługują różne formaty dokumentów, takie jak artykuły, podręczniki, listy itd. Makra są łatwe do zapamiętania, zwłaszcza te dla tradycyjnych składników wymienionych rodzajów dokumentów. Dla wygody użytkownika pakiet LaTeX jest automatycznie instalowany w Ubuntu wraz z oryginalnym pakietem TeX.

Oto przykład wykorzystania formatu artykułu w systemie LaTeX, w wersji TeX dokument wyglądałby następująco:

```
\documentclass{article}
\begin{document}
\title{To jest tytuł}
\author{Bill von Hagen}
\date{\today}
\maketitle

\begin{abstract}
Oto przykład ilustrujący, o ile prostsze jest formatowanie tekstu w przypadku użycia
zestawu makr \LaTeX , anizeli samych znaczników \TeX\ . Jest to rozwiązane zdecydowanie
łatwiejsze dla zwykłych śmiertelników...
\end{abstract}
\section{Wprowadzenie}
```

```
Tu umieszczony zostanie tekst
\end{document}
```

Po uruchomieniu LaTeX-a dane wyjściowe będą obszerniejsze, aniżeli w poprzednim przykładzie, oto one:

```
$ latex przyklad_latex.tex
This is pdfTeX, Version 3.141592-1.21a-2.2 (Web2C 7.5.4)
entering extended mode
(./przyklad_latex.tex
LaTeX2e <2003/12/01>
Babel <v3.8d> and hyphenation patterns for american, french, german, ngerman, b
ahasa, basque, bulgarian, catalan, croatian, czech, danish, dutch, esperanto, e
stonian, finnish, greek, icelandic, irish, italian, latin, magyar, norsk, polis
h, portuges, romanian, russian, serbian, slovak, slovene, spanish, swedish, tur
kish, ukrainian, nohyphenation, loaded.
(/usr/share/texmf-tetex/tex/latex/base/article.cls
Document Class: article 2004/02/16 v1.4f Standard LaTeX document class
(/usr/share/texmf-tetex/tex/latex/base/size10.clo)) (./przyklad_latex.aux)
Overfull \hbox (4.63338pt too wide) in paragraph at lines 9--10
\OT1/cmr/m/n/9 przy-padku uzy-cia zestawu makr L[T]X , anizeli samych znaczni
kow T[X]Jest
[1] (./przyklad_latex.aux)
(see the transcript file for additional information)
Output written on przyklad_latex.dvi (1 page, 996 bytes).
Transcript written on przyklad_latex.log.
```

Zamiast polecenia `tex` wykorzystanego w poprzednim przykładzie, zostało użyte polecenie `latex`, co spowodowało uruchomienie wersji programu, która została skompilowana wraz z makrami, danymi o czcionkach i innymi informacjami. Na rysunku 10.3 przedstawiam wyjściowy plik DVI utworzony za pomocą LaTeX-a i wyświetlony w oknie przeglądarki `xdvi`. Można zauważyć, że LaTeX oferuje makra do obsługi elementów tekstu, takich jak tytuł, data, streszczenie, części (wraz z automatyczną numeracją) itd. Na rysunku widać, że tekst jest niemal taki sam, drobne różnice widoczne są w wielkości marginesów czy układzie tekstu na stronie.

Rysunek 10.3.

*Podgląd pliku
wyjściowego LaTeX*



Większość osób korzystających z systemu TeX używa obecnie LaTeX-a ze względu na wygodę obsługi makr, ale również z powodu możliwości automatycznego tworzenia elementów, takich jak spisy treści, listy rysunków, indeksy oraz inne, ważne i użyteczne składniki. Szczegółowe omówienie działania systemu LaTeX wykracza poza zakres niniejszej książki, tym bardziej, że istnieje wiele opracowań na ten temat. Osoby zainteresowane tworzeniem dokumentów za pomocą tego programu dowiedzą się, gdzie szukać dodatkowych informacji, w punkcie „Dodatkowe źródła informacji o systemach TeX i LaTeX”.

Formatowanie i wyświetlanie plików DVI

Jak już wcześniej wspominałem, formatowanie pliku w systemie TeX i LaTeX za pomocą poleceń `tex` i `latex` powoduje w efekcie utworzenie pliku DVI. Aby wyświetlić jego zawartość, można użyć programu `xdvi` (dostępnego z linii poleceń) lub skorzystać z postprocesora do utworzenia pliku odpowiedniego dla danej drukarki, która go wydrukuje. Na rysunkach 10.2 i 10.3 przedstawiam podgląd pliku w programie `xdvi`. Oto lista postprocesorów, za których pomocą można uzyskać pliki odpowiednie dla danej drukarki:

- ♦ `dvilj` — tworzy pliki przeznaczone dla drukarek Hewlett-Packard LaserJet,
- ♦ `dvilj2p` — tworzy pliki przeznaczone dla drukarek Hewlett-Packard IIp LaserJet,
- ♦ `dvilj4` — tworzy pliki przeznaczone dla drukarek Hewlett-Packard IV LaserJet,
- ♦ `dvilj6` — tworzy pliki przeznaczone dla drukarek Hewlett-Packard VI LaserJet,
- ♦ `dvips` — tworzy pliki wyjściowe przeznaczone dla drukarek postscriptowych,
- ♦ `dvipdf` — tworzy pliki PDF, które można przeglądać za pomocą programu Adobe Acrobat, `xpdf` lub innej przeglądarki,
- ♦ `dvired` — tworzy pliki, w których na jednej kartce drukowane są dwie strony, przeznaczone dla drukarek Hewlett-Packard IIp LaserJet.

Wszystkie te pakiety są instalowane jako część systemu teTeX opisanego w punkcie „Instalowanie systemu TeX w Ubuntu”.



Kiedy trzeba użyć TeX-a lub LaTeX-a do dokumentów, nad którymi użytkownik pracuje lub po prostu chce trochę poeksperymentować, w pakiecie `dviutils` dostępne są dodatkowe narzędzia do plików DVI. Pakiet można zainstalować, korzystając z menedżera Synaptic. Z kolei pakiet `dviselect` pozwala na drukowanie wybranych stron z pliku oraz zarządzanie nimi; pakiet `dviconcat` umożliwia drukowanie dwóch stron na jednej kartce, a zatem można oszczędzić nieco papieru, a przy okazji uzyskać podgląd dokumentu.

Korzystanie z narzędzi graficznych do pracy z systemami TeX i LaTeX

Jak niejednokrotnie podkreślałem w niniejszej książce, żyjemy w świecie grafiki, a Ubuntu oferuje użytkownikom wszelkie graficzne programy, jakich mogliby oczekiwać. Chociaż narzędzia linii poleceń, takie jak TeX czy LaTeX, są proste w użyciu, a tekstowy format

plików tych programów upraszcza ich wymianę pomiędzy wieloma użytkownikami różnych komputerów, to jednak trudno przebić wygodę oferowaną przez graficzny interfejs, zwłaszcza w przypadku czynności, takich jak edytowanie tekstu.

Na szczęście, programiści wolnego oprogramowania z całego świata poszli krok dalej i utworzyli graficzne rozwiązanie dla systemu LaTeX. Są to aplikacje Lyx i TeXmacs, obie dostępne w repozytoriach Ubuntu. Podobnie jak LaTeX, nie są one domyślnie instalowane w systemie, ale korzystając z menedżera pakietów Synaptic, łatwo je odnaleźć i dodać do systemu (w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, można znaleźć więcej informacji na temat instalowania programów za pomocą Synaptica). Program Lyx można uruchomić z menu *Aplikacje/Biuro*, natomiast dla TeXmacsa należy użyć w linii poleceń komendy `texmacs`.

Podczas pracy z dokumentami LaTeX trudno przebić wygodę zaimplementowaną w programie Lyx. Oferuje on wygodny, graficzny interfejs do tworzenia i edytowania dokumentów, wybierania poszczególnych elementów składowych i stylów z rozwijanych menu, formatowania ponownie wybranych znaków, akapitów itd. A wszystko bez konieczności zapamiętania wszystkich poleceń niskiego poziomu programu TeX.

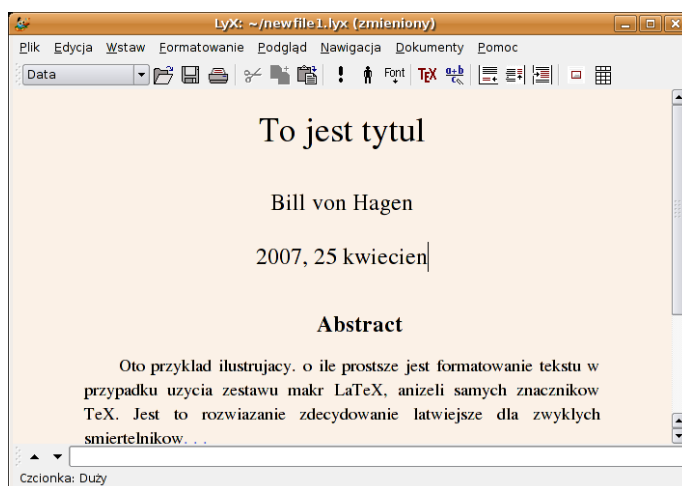


Jeżeli użytkownik będzie chciał uruchomić zainstalowany program Lyx z poziomu linii poleceń, należy użyć komendy `lyx-qt`, ponieważ wersja Lyksa znajdująca się w repozytoriach Ubuntu do wyświetlania grafiki w programie korzysta właśnie z biblioteki QT.

Na rysunku 10.4 widoczny jest ten sam dokument, którego użyto w poprzednim przykładzie (patrz rysunek 10.3), tym razem wyświetlony w programie Lyx. Od razu można zauważyć niemal identyczny wygląd, przypominający pliki wyświetlane w programie xdv, a tworzone za pomocą LaTeX-a.

Rysunek 10.4.

Przykładowy dokument wyświetlony w programie Lyx



Program ten do zapisywania dokumentów wykorzystuje własny format plików (*lyx*). Aby pracować ze standardowymi dokumentami programu LaTeX, należy najpierw zaimportować je do programu, używając polecenia z menu *Plik/Importuj*. Pliki można też eksportować, dzięki czemu możliwa jest wymiana z innymi użytkownikami korzystającymi

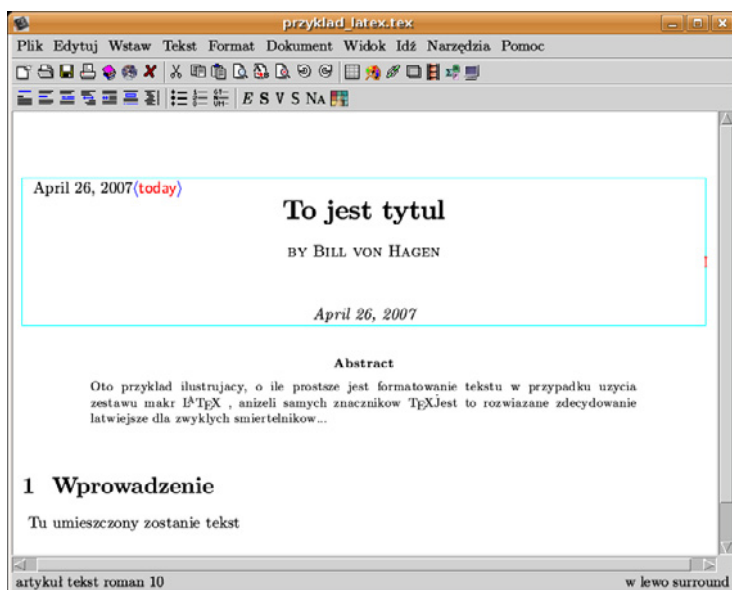
z programu LaTeX. Porównując oryginalne pliki LaTeX i utworzone w procesorze Lyx, można zauważyć, że do tych drugich doszło kilka szczegółów dotyczących formatowania, ale poza tym pliki te niczym się nie różnią.

Wygoda użytkowania oferowana przez graficzny interfejs oraz brak konieczności zapamiętania wszystkich poleceń czynią z Lyksa wygodne narzędzie, z którym warto poeksperymentować w zakresie pracy z dokumentami LaTeX. Więcej informacji o programie można znaleźć na stronie www.lyx.org.

Kolejnym rozwiązaniem oferującym graficzny interfejs do pracy z plikami LaTeX jest pakiet *texmacs*. Jest to dodatek integrujący się z programem GNU emacs. A zatem podczas pracy z plikami LaTeX można używać znanych z emacsa poleceń. Na rysunku 10.5 widoczny jest ten sam dokument (widoczny na rysunku 10.3), tym razem w oknie edytora TeXmacs. Od razu można zauważyć wygląd niemal identyczny jak wygląd plików wyświetlanych w programie xdvi, a tworzonych za pomocą LaTeX-a.

Rysunek 10.5.

Przykładowy dokument wyświetlony w programie TeXmacs



Po zainstalowaniu programu TeXmacs można go uruchomić, wpisując w terminalu polecenie `texmacs` lub wybierając z menu *Aplikacje/Akcesoria/GNU TeXmacs Editor*. Jeżeli program zostanie uruchomiony z linii poleceń, wówczas na początku zostanie wyświetlone całe mnóstwo komunikatów konfiguracyjnych, które można zignorować; podczas następnego uruchomienia programu nie będą już widoczne.

Posługiwanie się TeXmacsem jest bardzo wygodne, oferuje on kilka specyficznych dla LaTeX-a menu i ikon, dzięki którym łatwo i szybko można używać stylów, elementów tekstu, a nawet sformatować wybrany fragment dokumentu. Odmienne niż w przypadku Lyx, TeXmacs może pracować zarówno na własnych, jak i oryginalnych dokumentach LaTeX-a, nie ma potrzeby wykonywania importu do programu. Więcej informacji na temat TeXmacsa można znaleźć na stronie www.texmacs.org, program jest też wyposażony w system pomocy.

Dodatkowe źródła informacji o systemach TeX i LaTeX

Ponieważ TeX i LaTeX są używane od wielu lat na różnych systemach, w sieci można znaleźć ogromne ilości informacji na ich temat. Proste wyszukiwanie w internecie przyniesie prawdopodobnie więcej wyników, niż czytelnik mógłby oczekiwać. Aby jednak oszczędzić nieco czasu, podaję kilka moich ulubionych stron poświęconych programom TeX i LaTeX:

- ♦ <http://www.andy-roberts.net/misc/latex/index.html> — świetna seria poradników dotyczących używania LaTeX-a autorstwa Andrew Robertsa,
- ♦ <http://www.math.uiuc.edu/~hildebr/tex/course/> — wprowadzenie do programu LaTeX, krótki kurs oraz porównanie programów TeX i LaTeX przygotowane na wydziale matematyki uniwersytetu w Illinois,
- ♦ <http://www.damtp.cam.ac.uk/computing/tex/> — świetna strona uniwersytetu w Cambridge zawierająca wiele samouczków oraz przykładów tworzenia różnorodnych dokumentów za pomocą programu TeX,
- ♦ <http://www.tldp.org/HOWTO/TeX-HOWTO.html> — strona napisana kilka lat temu, ale dzięki temu, że tak stabilne programy jak TeX czy LaTeX z upływem lat niewiele się zmieniają, informacje wciąż są aktualne. Można znaleźć tu wskazówki na temat tworzenia dokumentów w programach TeX i LaTeX, a także informacje dotyczące implementacji tych aplikacji w systemach Linux.

Powstało też wiele książek poświęconych programom TeX i LaTeX, ale ponieważ używałem programu na początku lat 90., dlatego też nie przedstawię opinii na temat aktualnie wydawanych pozycji. Jednak żadna dyskusja o aplikacjach TeX i LaTeX nie będzie kompletna bez podania dwóch podstawowych dla obu programów tekstów, oto one:

- ♦ *TeXbook*, Donald E. Knuth, Addison-Wesley Professional, 1984,
- ♦ *LaTeX: A Document Preparation System*, II wydanie, Leslie Lamport, Addison-Wesley Professional, 1994.

Ponieważ autorami obu książek są autorzy opisywanych programów, można na tych publikacjach polegać całkowicie. W programie LaTeX zostały, co prawda, wprowadzone pewne zmiany (przez samego Lesliego Lamport), ale są to nadal fundamentalne pozycje dla obu aplikacji.

Edycja tekstu w programie OpenOffice.org Writer

OpenOffice.org (www.openoffice.org) to pakiet aplikacji biurowych obejmujący procesor tekstu (Writer), arkusz kalkulacyjny (Calc), program do rysowania (Draw), program do tworzenia prezentacji (Impress) i bazę danych (Base). W tym rozdziale omawiam edytor tekstu, w pozostałych znalazły się dokładne opisy arkusza kalkulacyjnego, programu do grafiki i tworzenia prezentacji. Ale najpierw kilka słów od naszego sponsora.

OpenOffice.org to jeden z najpotężniejszych i najpopularniejszych programów o otwartym kodzie źródłowym dostępnym w sieci. Osoby w dużej mierze odpowiedzialne za jego pojawienie się to pracownicy firmy Sun Microsystems. Scott McNealy, prezes Sun Microsystems, długo żywił nienawiść do Microsoftu, dlatego nie było dla nikogo zaskoczeniem, kiedy w roku 1999 Sun wykupił niemiecką firmę StarDivision, producenta pakietu oprogramowania o nazwie StarOffice. Początkowo zakup StarOffice'a był postrzegany jako sposób wywierania nacisku na Javę Sun (ponieważ zarówno StarOffice, jak OpenOffice.org w dużej mierze zależą od Javy) oraz dostarczenia użytkownikom pakietu biurowego, którego mogliby używać na swoich komputerach z oprogramowaniem Sun.

Wkrótce Sun udostępnił kod źródłowy StarOffice'a społeczności zajmującej się wolnym oprogramowaniem na warunkach licencji SISSL (ang. *Sun Industry Standard Source License*). Począwszy od wydania OpenOffice.org 2.0, cały kod dostępny jest już na warunkach licencji GNU Lesser Public License (LGPL). Sun wciąż oferuje StarOffice, dla którego dostępne są pewne funkcje nieobecne w OpenOffice.org, ale ponieważ pakiet StarOffice nie jest dostępny dla Ubuntu, nie będziemy się nimi zajmować.

Oprócz możliwości oferowanych przez pakiet OpenOffice.org oraz braku kosztów związanych z jego nabyciem, kolejnym ważnym argumentem przemawiającym za jego użytkowaniem jest zupełnie nowy format (*OpenDocument*) używany do przechowywania i wymiany dokumentów (więcej informacji można znaleźć na stronie http://oasis-open.org/committees/tc_home.php?wg_abbrev=openoffice). Do tej pory wszystkie procesory tekstu używały swoich własnych, binarnych formatów dla tworzonych plików, co znacznie utrudniało korzystanie z różnych programów czy też wymianę dokumentów z użytkownikami, którzy korzystali z innych aplikacji. Standard OpenDocument został zaprojektowany w oparciu o XML, zatem pliki w tym formacie mogą być szeroko wykorzystywane w innych edytorach obsługujących ten standard. Może się to wydawać niezbyt ważną funkcją, ale aby przekonać się, że w rzeczywistości jest dokładnie odwrotnie, wystarczy spróbować utworzyć jakiś plik zapisany parę lat temu np. w formacie WriteStar.

W dalszej części rozdziału na określenie programu OpenOffice.org Writer używana będzie skrócona nazwa Writer. Podczas odwołania do polecenia, za którego pomocą można uruchomić program z linii poleceń, zostanie użyta nazwa `oowriter`.

Instalowanie plików dla programu OpenOffice.org Writer

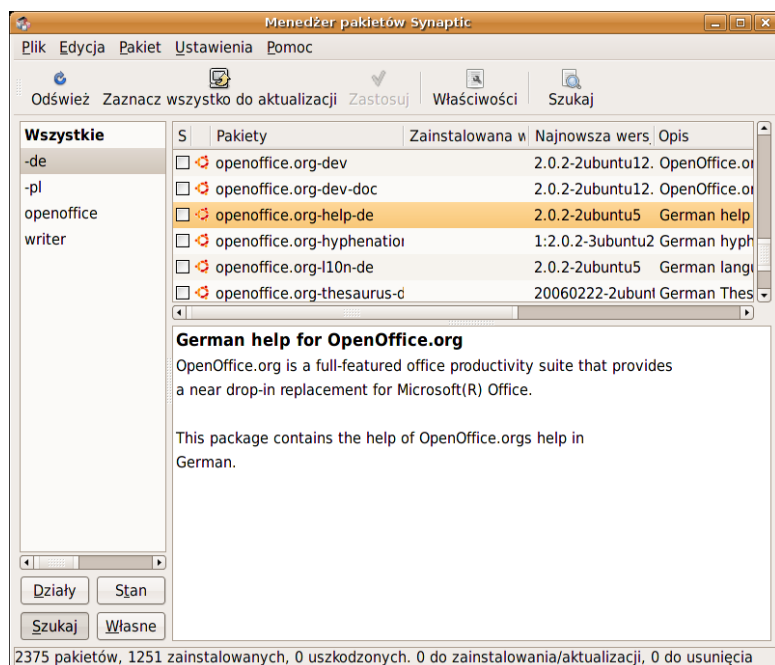
Pakiet OpenOffice.org wraz z programem Writer znajduje się w systemie, jeżeli był instalowany z płyt Live CD Desktop i Alternate Install CD (zarówno w wersji tekstowej, jak i OEM). Jeżeli użytkownik chciałby dodać pakiet OpenOffice.org wraz z procesorem tekstu Writer do Ubuntu w wersji serwerowej ze środowiskiem graficznym, wówczas powinien zainstalować pakiety przy użyciu jednego z dostępnych menedżerów plików, czyli `apt-get`, `aptitude` czy `Synaptic` (zgodnie z informacjami zamieszczonymi w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”).

Jedną z cech odróżniających Ubuntu od innych dystrybucji Linuksa jest wyraźne ukierunkowanie na użytkownika, zwłaszcza w zakresie internacjonalizacji. Jeżeli w systemie używany jest tylko jeden język, odpowiedni plik został dodany do systemu już podczas instalacji Ubuntu. Jeśli jednak użytkownik zamierza tworzyć dokumenty w różnych językach, wówczas trzeba będzie również zainstalować odpowiednie pliki lokalizacyjne

i językowe (sprawdzanie pisowni, tezaursus, dzielenie wyrazów), dzięki którym program będzie dostępny w odpowiednim języku. Pliki i pakiety lokalizacyjne można łatwo odnaleźć, używając do wyszukiwania ciągu określającego dany kraj (kod dla Polski to PL). Na rysunku 10.6 przedstawiam program Synaptic, wyświetlające pliki lokalizacyjne dla pakietu OpenOffice.org dla niemieckiej wersji językowej, identyfikowanej za pomocą ciągu -de.

Rysunek 10.6.

Dodawanie pakietów językowych dla OpenOffice.org



Jeżeli zostały już zainstalowane pakiety językowe czy lokalizacyjne dla jakiegokolwiek innego programu wchodzącego w skład pakietu OpenOffice.org, wówczas nie ma potrzeby ponownego ich instalowania. Wystarczy raz umieścić je w systemie, a będą dostępne dla każdego komponentu pakietu.

W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, zamieszczam więcej informacji na temat instalowania programów za pomocą Synaptica oraz podobnych narzędzi linii poleceń.



Użytkownik może zainstalować kompletne wsparcie językowe dla wszystkich aplikacji (bez zmiany wersji językowej samego interfejsu GNOME), wystarczy dodać odpowiedni metapaket dla odpowiednich danych lokalizacyjnych (tzw. lokali, ang. *locales*). Nazwa takiego pliku wygląda następująco: *language-support-de*, przy czym na końcu znajduje się kod kraju; w pliku znajduje się też słownik dla danego języka. Aby w pełni przetłumaczyć interfejs i menu aplikacji, należy zainstalować odpowiedni pakiet językowy. Nazwy pakietów językowych składają się z kodu języka i kodu kraju. Po zainstalowaniu odpowiednich plików językowych można dowolnie zmieniać język dla danego dokumentu, wystarczy z menu wybrać *Narzędzia/Opcje/Ustawienia językowe/Języki* i z listy rozwijanej *Domyślny język dokumentów* wybrać odpowiednią pozycję.

Rozszerzenia plików dokumentów w pakiecie OpenOffice.org

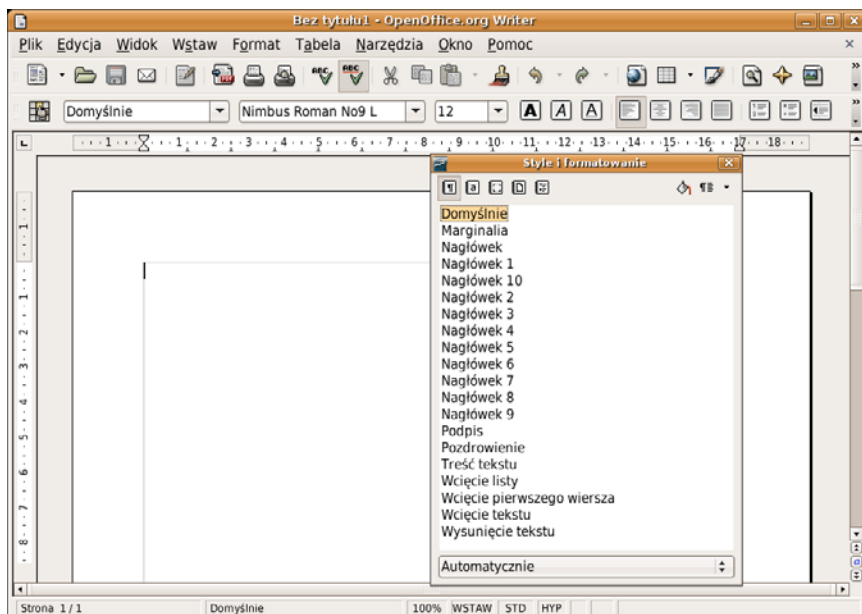
Wraz z zastosowaniem formatu OpenDocument zmianie uległy domyślne rozszerzenia plików stosowane w OpenOffice.org. Poniżej znajduje się lista wszystkich stosowanych dotychczas rozszerzeń:

- ♦ *odm* — dokument główny w OpenDocument,
- ♦ *odt* — dokument OpenDocument w OpenOffice.org 2.0 i kolejnych wersjach,
- ♦ *ott* — szablon OpenDocument w OpenOffice.org 2.0 i kolejnych wersjach,
- ♦ *sxw* — dokument programu Writer, przed wydaniem OpenOffice.org 2.0,
- ♦ *stw* — szablon programu Writer, przed wydaniem OpenOffice.org 2.0,
- ♦ *sxg* — dokument główny programu Writer, przed wydaniem OpenOffice.org 2.0.

Plikami, z którymi użytkownik będzie miał najczęściej do czynienia, będą miały rozszerzenia *.odt* i *.ott*. Mogą się zdarzyć również starsze pliki w formacie *.sxw*, jeżeli dokumenty będą tworzone w starszych wersjach OpenOffice'a. Same pliki SXW mają ciekawą budowę, ponieważ zawierają spakowane pliki XML wraz z dodatkowymi danymi. Aby korzystać z nich w programie OpenOffice.org, należy użyć kreatora importu, a następnie zapisać je już w formacie OpenDocument. Więcej informacji na ten temat można znaleźć w punkcie „Importowanie plików utworzonych w innych procesorach tekstu”.

Krótki przewodnik po programie Writer

Program można uruchomić z linii poleceń za pomocą komendy *oowriter*, ale łatwiej będzie skorzystać z menu *Aplikacje/Biuro*. Aby uruchomić program, należy wybrać pozycję *OpenOffice.org Word Processor*. Zostanie wówczas wyświetlone okno programu Writer (patrz rysunek 11.7).

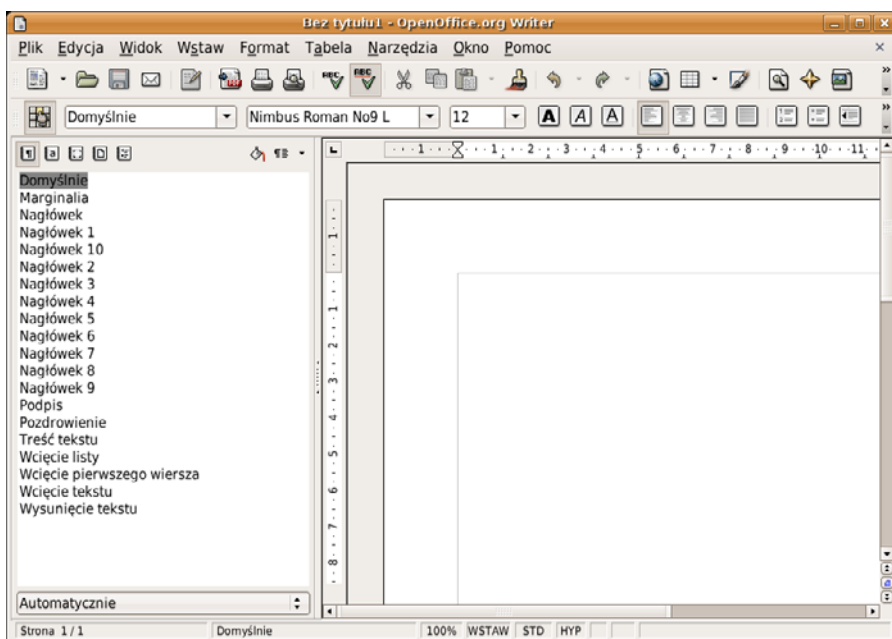


Rysunek 10.7. Uruchamianie programu Writer

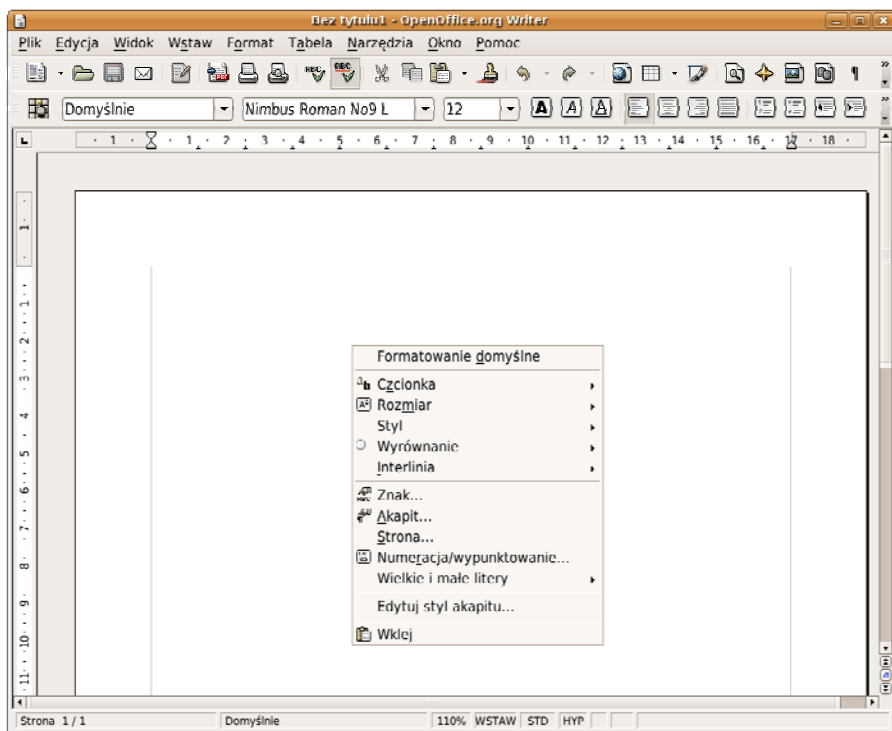
OpenOffice.org Writer wyposażony jest w doskonały system pomocy, podobnie zresztą jak cały pakiet OpenOffice.org, dlatego nie będę zanudzać czytelników opisem zawartości każdego menu. Zamiast tego proponuję krótki przewodnik po interfejsie tego programu (patrz rysunek 10.7).

- ♦ **Menu:** rozwijane menu zorganizowane są podobnie jak w Microsoft Word, dzięki czemu użytkownik nie powinien mieć kłopotów z przestawieniem się na nowy program. Pozycje menu ze strzałkami umieszczonymi po prawej stronie umożliwiają dostęp do kolejnych podmenu.
- ♦ **Paski narzędzi:** domyślnie program wyświetla paski narzędziowe: *Standardowy* i *Formatowanie*, które umożliwiają korzystanie z różnych narzędzi za pomocą ikon znajdujących się na nich. Użytkownik może modyfikować listę wyświetlanych na paskach ikon, wystarczy kliknąć kontrolkę znajdującą się przy prawej krawędzi paska narzędzi ze strzałką skierowaną w dół. Można też włączać i wyłączać wyświetlanie różnych pasków narzędzi, wybierając ich nazwy z menu *Widok/Paski narzędzi*.
- ♦ **Linijki:** wyświetlają pionowe i poziome położenie tekstu w dokumencie. Szare obszary widoczne na liniijkach oznaczają marginesy ustawione w dokumencie. Wyświetlanie linijek można wyłączyć w menu *Widok/Linijka*.
- ♦ **Okno *Style i formatowanie*:** nie jest widoczne po włączeniu programu, aby je wyświetlić, należy wcisnąć klawisz *F11* (po ponownym wcisnięciu tego klawisza okno zostanie zamknięte). Okno to umożliwia łatwy dostęp do opcji związanych ze stylami dostępnymi w danym dokumencie i jest szczególnie wygodne, gdy użytkownik nie zdecyduje się korzystać z paska narzędzi. Lista rozwijana w tym oknie umożliwia wyświetlanie dostępnych stylów (akapitu, znaków, ramek, strony i listy). Okno można zadokować w głównym oknie programu, wystarczy przytrzymać wcisnięty klawisz *Ctrl*, a następnie dwukrotnie kliknąć szary obszar znajdujący się pod belką z nazwą okna. Dzięki temu okno *Style i formatowanie* stanie się integralną częścią głównego okna programu. Aby wrócić do poprzedniego ustawienia, należy powtórzyć opisaną czynność. Na rysunku 10.8 widać, jak wygląda program z zadokowanym oknem *Style i formatowanie*.
- ♦ **Menu kontekstowe:** wystarczy ustawić kursor na pustym obszarze dokumentu i wcisnąć prawy przycisk myszy, zostanie wówczas wyświetlone menu kontekstowe Writera widoczne na rysunku 10.9. Umożliwia ono łatwy dostęp do kilku operacji związanych z formatowaniem tekstu.

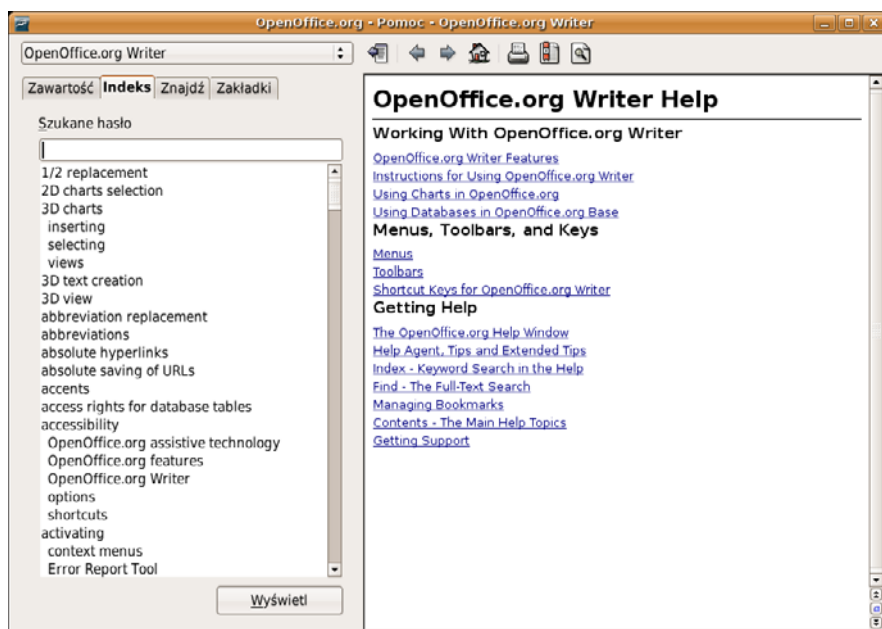
Program Writer jest też wyposażony w obszerny system pomocy, wystarczy wybrać z menu *Pomoc/Pomoc Openoffice.org* lub wcisnąć klawisz *F1* w czasie pracy z programem. Na rysunku 10.10 widać zawartość okna z pomocą programu Writer. Można w nim znaleźć wyszukiwarkę słów kluczowych, listę tematów oraz szczegółową zawartość wybranego zagadnienia (po prawej stronie okna).



Rysunek 10.8. Zadokowane okno *Style i formatowanie*



Rysunek 10.9. Menu kontekstowe programu *Writer*



Rysunek 10.10. Pomoc programu Writer

Wpisywanie danych osobistych

Pierwszą rzeczą, jaką należy wykonać po uruchomieniu programu, jest wprowadzenie pewnych informacji o użytkowniku w polu *Dane użytkownika*. Dane te będą wykorzystywane przez program do wypełniania różnych pól w dokumentach tworzonych za pomocą kreatorów pakietu OpenOffice.org (omawiam to w następnym punkcie).

Aby wprowadzić informacje, należy z menu kolejno wybrać *Narzędzia/Opcje* i przejść do pozycji *Dane użytkownika*. Następnie w wyświetlonym formularzu trzeba wypełnić odpowiednie pola (patrz rysunek 10.11), a po zakończeniu kliknąć przycisk *OK*.

Tworzenie dokumentów za pomocą kreatorów

Najprostszą metodą tworzenia dokumentów w programie Writer jest użycie kreatorów, które oferują wzory wielu różnych pism. Poniżej opisuję proces tworzenia faksu z wykorzystaniem jednego z kreatorów dostępnych w programie Writer. Niezależnie od użytego kreatora, metoda postępowania jest taka sama, ponieważ głównym celem kreatorów jest osiągnięcie założonego celu.

Aby uruchomić kreatora mającego pomóc w utworzeniu faksu, należy kolejno:

1. Wybrać z menu *Plik/Kreatory*, a następnie odpowiedni typ dokumentu, który ma zostać utworzony. Kreator umożliwia utworzenie listu, agendy, faksu, prezentacji strony internetowej itd. W omawianym przypadku należy wybrać z wyświetlanej listy *Faks*. Zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 10.12.

Opcje - OpenOffice.org - Dane użytkownika

OpenOffice.org

- Dane użytkownika
- Ogólne
- Pamięć
- Widok
- Drukowanie
- Ścieżki
- Kolory
- Czcionki
- Bezpieczeństwo
- Wygląd
- Ułatwienia dostępu
- Java
- Ładuj/Zapisz
- Ustawienia językowe
- OpenOffice.org Writer
- OpenOffice.org Writer/Web
- OpenOffice.org Base
- Wykres
- Internet

Adres

Firma: Kuźnia słów

Imię/nazwisko/inicjały: William von Hagen wvh

Ulica: Wzgórze przedrzeźniaczy 1313

Kod pocztowy/Miejscowość: 40-064 Katowice

Kraj/region: Polska

Tytuł/stanowisko:

Telefon (dom/praca):

Faks/e-mail:

OK Anuluj Pomoc Wstecz

Rysunek 10.11. Formularz Dane użytkownika

pr-bottle_f.ott(tylko do odczytu) - OpenOffice.org Writer

William von Hagen
Wzgórze przedrzeźniaczy 1313, Katowice, 40-064

Kreator faksu

Kroki

1. Projekt strony
2. Zawarte elementy
3. Nadawca i odbiorca
4. Stopka
5. Nazwa i lokalizacja pliku

Wybierz typ faksu i projekt strony

☐ Faks służbowy
Projekt strony: Modern Fax from Private

☒ Faks prywatny
Projekt strony: Bottle

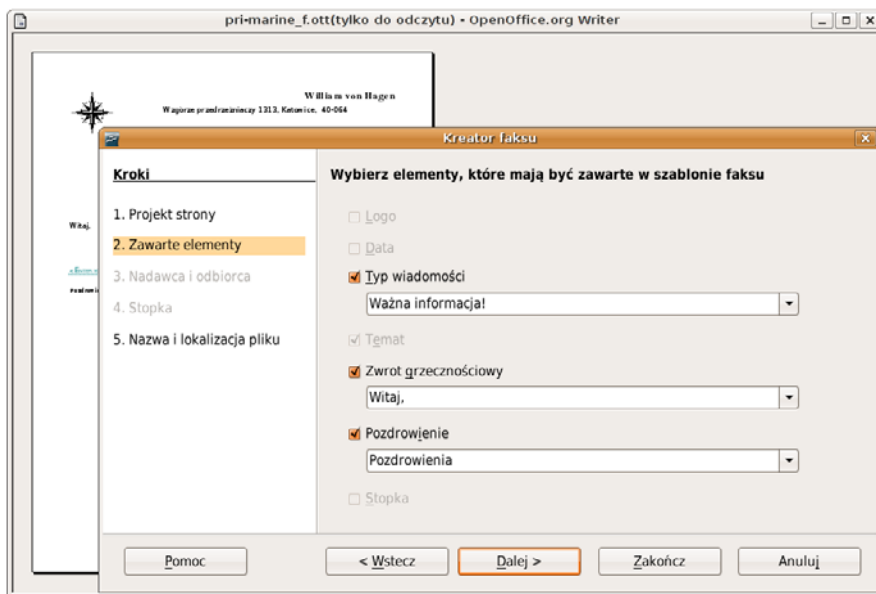
Ten kreator pomoże stworzyć szablon faksu. Utworzony szablon może być używany wielokrotnie.

Pomoc < Wstecz Dalej > Zakończ Anuluj

Rysunek 10.12. Pierwsze okno dialogowe kreatora faksu

2. Następnie zaznaczyć odpowiednią opcję, w zależności od tego, czy faks ma charakter służbowy czy prywatny. Każdy z typów ma określone domyślne style i dekoracje, które można wybrać z listy rozwijanej *Projekt strony*. W oknie wyświetlanym pod oknem *Kreator faksu* widoczny jest podgląd tworzonego dokumentu, na którego podstawie użytkownik może zdecydować, czy dany wygląd mu odpowiada. Przy bliższym przyjrzeniu się można zauważyć, że część pól na faksie została już wypełniona, zgodnie z informacjami zawartymi w punkcie „Wpisywanie danych

osobistych”. W opisywanym przykładzie został wybrany faks prywatny. Aby przejść do kolejnego okna dialogowego (patrz rysunek 10.13), po dokonaniu wyboru należy kliknąć przycisk *Dalej*.

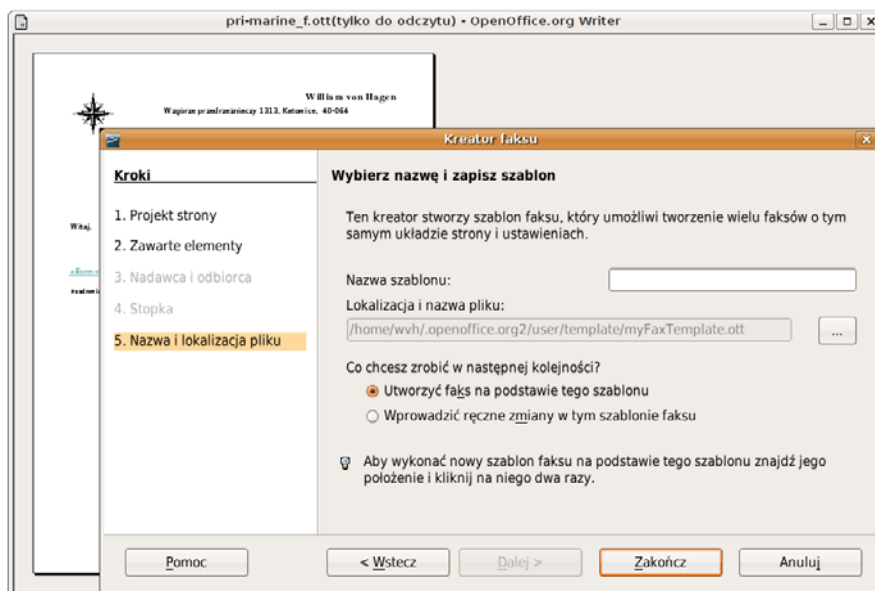


Rysunek 10.13. Drugie okno dialogowe kreatora faksu

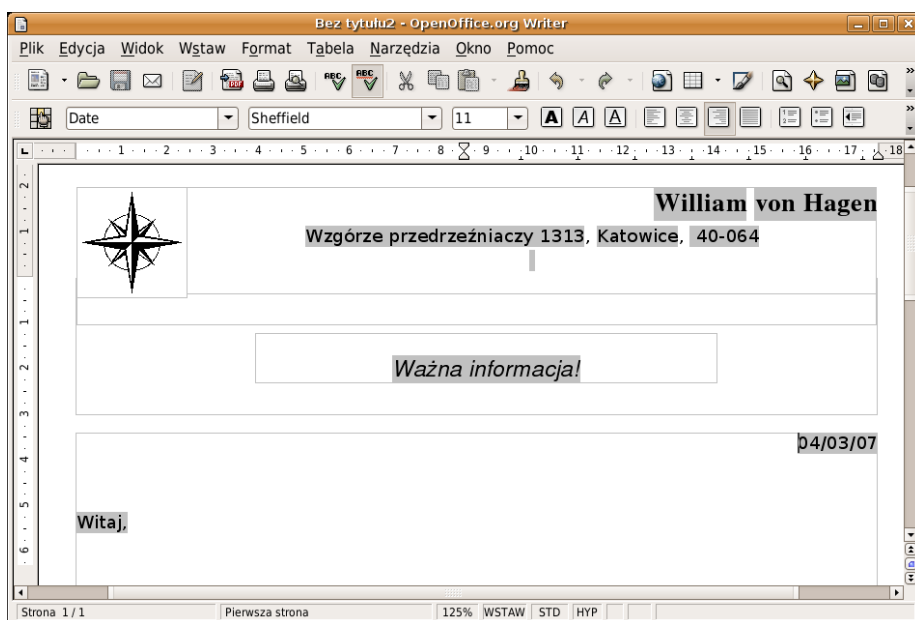
3. W następnym oknie użytkownik może wybrać, jakie standardowe elementy mają znaleźć się na fakse. Można określić, czy w dokumencie mają zostać zamieszczone: *Typ wiadomości*, *Zwrot grzecznościowy*, *Pozdrowienie*. Dodatkowo z listy rozwijanej można wybrać treść każdego z powyższych elementów. Po wprowadzeniu odpowiednich wartości należy kliknąć przycisk *Dalej*, zostanie wyświetlone kolejne okno dialogowe (patrz rysunek 10.14).
4. Jeżeli użytkownik wprowadził zmiany do szablonu i chciałby je zapisać, kolejne okno dialogowe kreatora udostępnia taką możliwość. Zmodyfikowany plik można zapisać jako nowy szablon, wówczas należy uzupełnić pole *Nazwa szablonu* oraz *Lokalizacja i nazwa pliku*. Jeżeli użytkownik nie chce wprowadzać zmian, a tylko jednorazowo utworzyć dokument, wówczas powinna zostać zaznaczona opcja *Utworzyć faks na podstawie tego szablonu*. Metody tworzenia szablonów dokumentów omówiłem w punkcie „Tworzenie i używanie szablonów”.

Po wprowadzeniu ostatnich ustawień należy kliknąć przycisk *Zakończ*, kreator zostanie wyłączony i pojawi się okno programu Writer z wzorem dokumentu, który użytkownik będzie mógł wypełnić zgodnie z potrzebami (patrz rysunek 10.15). Podczas tworzenia faksu dostępne będą jeszcze inne możliwości, np. wybranie adresu odbiorcy z listy adresowej, ale ich omówienie wykracza poza treść niniejszego rozdziału.

Jak widać, kreatory programu Writer zdecydowanie ułatwiają tworzenie różnych typów dokumentów dzięki instalowanym domyślnie szablonom. Użytkownicy mogą też tworzyć



Rysunek 10.14. Ostatnie okno dialogowe kreatora faksu



Rysunek 10.15. Faks jest gotowy do wypełnienia



Jeżeli użytkownik chciałby zmodyfikować nieco szablon dokumentu przed wypełnieniem właściwą treścią, wówczas powinien użyć opcji *Wprowadzić ręczne zmiany w tym szablonie faksu*, widocznej na rysunku 10.14. Zostanie wyświetlony szablon, w którym użytkownik będzie mógł zmodyfikować style i formatowanie strony, zgodnie ze wskazówkami zamieszczonymi w następnym punkcie.

własne szablony dokumentów bazujące na tych, które dostępne są w systemie. Więcej informacji na temat, jak to zrobić, można znaleźć w punkcie „Tworzenie i używanie szablonów”.

Modyfikowanie stylów i wyglądu dokumentów

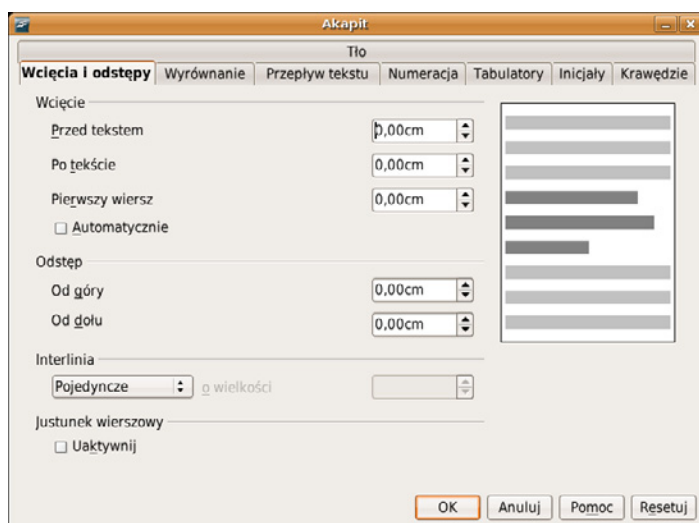
Styl jest to termin używany w procesorach tekstu do określenia ustawień dotyczących formatowania i wyświetlania wybranego fragmentu tekstu, zwykle akapitu. Podczas edytowania tekstu można zastosować istniejący styl, wybierając go z listy rozwijanej znajdującej się na pasku narzędzi *Formatowanie* lub też z okna *Style i formatowanie*, które można wyświetlić za pomocą klawisza *F11* (ten sam klawisz służy do jego wyłączenia).

Wszystkie szablony dokumentów dostępne w programie Writer posiadają zdefiniowany zestaw stylów umożliwiających formatowanie różnych części dokumentu. Aby zastosować dany styl do istniejącego akapitu, należy zaznaczyć tekst, a następnie wybrać odpowiedni styl z listy rozwijanej lub okna *Style i formatowanie*. Używamy pojęcia akapit, ponieważ style domyślnie przypisane są elementom kończącym się znakiem powrotu karetki, co automatycznie czyni z nich akapity nawet wtedy, jeśli jest związane ze stylem, takim jak nagłówek, pozdrowienie itd.

Aby sformatować wybrany akapit, należy ustawić na nim kursor, kliknąć prawym przyciskiem myszy i z menu kontekstowego wybrać pozycję *Akapit*. Zostanie wyświetlone okno dialogowe widoczne na rysunku 10.16; w nim można zmienić ustawienia dotyczące pojedynczego elementu w tekście bez modyfikowania innych elementów sformatowanych takim samym stylem.

Rysunek 10.16.

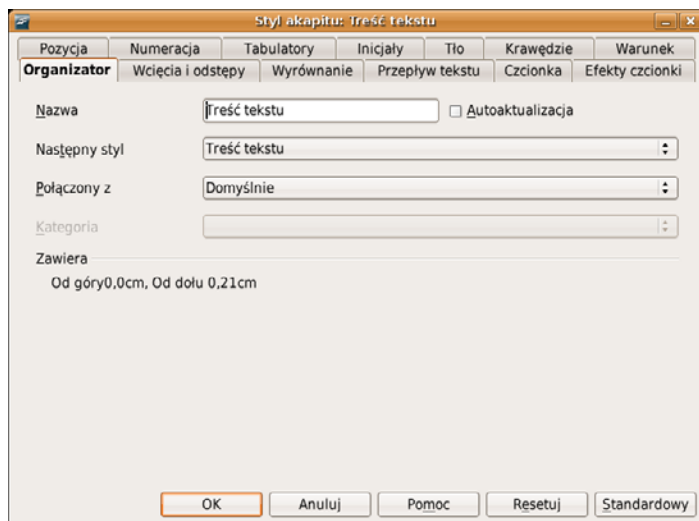
Zmiana stylu
dla pojedynczego
akapitu



Chociaż wszystkie dokumenty pakietu OpenOffice.org posiadają obszerny zestaw dostępnych stylów, każdy użytkownik może mieć inne potrzeby. Aby dostosować istniejący styl do własnych potrzeb lub też wymogów firmy, należy ustawić na odpowiednim akapicie kursor, kliknąć prawym przyciskiem myszy i z menu kontekstowego wybrać opcję *Edytuj styl akapitu*. Zostanie wyświetlone okno widoczne na rysunku 10.17.

Rysunek 10.17.

Zmiana ustawień
istniejącego stylu



Zmiana jakiegokolwiek ustawień dostępnych w tym oknie dialogowym spowoduje zmianę całego stylu, co wiąże się z aktualizacją wszystkich fragmentów tekstu sformatowanych danym stylem. W oknie *Styl akapitu: Treść tekstu* (patrz rysunek 10.17) dostępnych jest o wiele więcej opcji niż w oknie wywoływanym z menu kontekstowego poleceniem *Akapit* (patrz rysunek 10.16). Jest tak dlatego, że za pomocą okna dialogowego widocznego na rysunku 10.17 można zmodyfikować każdy aspekt formatowania danego stylu, a nie tylko zaznaczony element.

Zmiana domyślnego stylu często jest wystarczającym rozwiązaniem, ale może się okazać, że nie sprawdza się w sytuacji, kiedy użytkownik chce zmodyfikować tylko kilka elementów w dokumencie, do których zastosowano dany styl. Kiedy style domyślne są już niewystarczające, wówczas zamiast spędzić godziny na próbach ich zmodyfikowania i dostosowania do nowych potrzeb, należy utworzyć nowy styl w oparciu o jeden z już istniejących.

Aby utworzyć nowy styl w oparciu o jeden z już istniejących, najpierw należy się upewnić, czy na ekranie jest wyświetlane okno dialogowe *Style i formatowanie*; jeśli tak nie jest, należy je przywołać, korzystając z klawisza *F11*. Następnie należy ustawić kursor w akapicie sformatowanym stylem, który zostanie użyty do utworzenia nowego i z okna *Style i formatowanie* kliknąć ikonę *Nowy styl z zaznaczenia*, znajdującą się po prawej stronie na pasku narzędzi tego okna dialogowego. Z listy rozwijanej, która zostanie wyświetlona, należy wybrać opcję *Nowy styl z zaznaczenia*, co spowoduje wyświetlenie kolejnego okna dialogowego (patrz rysunek 10.18).

Trzeba wprowadzić nazwę, która zostanie przypisana nowemu stylowi, a następnie kliknąć *OK*. Na rysunku 10.18 widać, że nowemu stylowi nadano nazwę *wprowadzenie*, będzie on używany do formatowania wstępu do dłuższych artykułów. Wprowadzona nazwa zostanie dodana do listy wyświetlanej w oknie *Style i formatowanie*, a nowy styl przypisany do sposobu formatowania akapitu. Opisaną procedurę można powtarzać wielokrotnie, dzięki czemu można utworzyć dowolną liczbę stylów dostosowanych do konkretnych potrzeb.

Rysunek 10.18.

Tworzenie nowego stylu z zaznaczenia



Liczbę nowo tworzonych stylów należy — w miarę możliwości — minimalizować. Przykładowo w sytuacji, kiedy pracuje się nad firmowym dokumentem, dodawanie nowych stylów może być bardzo niewskazane i sprzeczne z wytycznymi firmy czy też zrytuje kolejne osoby, które będą pracować nad danym dokumentem. Dlatego przed utworzeniem nowego stylu warto się upewnić, czy jest to absolutnie konieczne.

Tworzenie i używanie szablonów

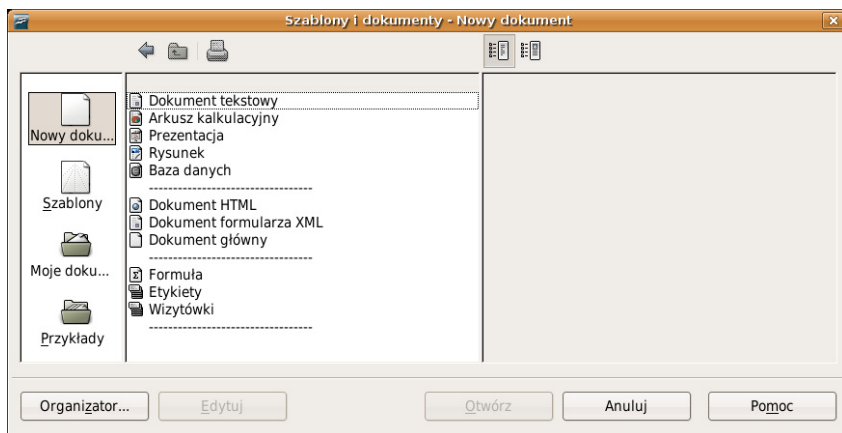
Jak już wspominałem wcześniej, Writer wyposażony jest w spory zestaw wzorów dokumentów znanych jako **szablony**, a także w kreator, który ułatwia korzystanie z nich, a nawet tworzenie nowych na podstawie już istniejących dokumentów. Powyżej opisałem sposób, w jaki można zmodyfikować styl, tak aby jego ustawienia lepiej dostosować do danego dokumentu. Jest to wygodne rozwiązanie, ale spowoduje tylko zmianę stylów skojarzonych z danym dokumentem — szablon, na którego podstawie dokument został utworzony, nadal będzie korzystał z domyślnego zestawu stylów, co utrudni wymianę dokumentów z innymi użytkownikami i wspólną nad nimi pracę. Skomplikuje to ich ponowne użycie, ponieważ każdorazowo trzeba będzie je kopiować do nowo utworzonego dokumentu. W najlepszym przypadku skończy się to bólem palców, w najgorszym okaże się zupełnie nieefektywne. W takiej sytuacji właściwym rozwiązaniem jest utworzenie nowych szablonów i zarządzanie nimi. Otwierając zaktualizowany szablon, można wygodnie korzystać z nowych stylów, a nawet wymieniać się gotowymi szablonami z innymi użytkownikami.



Tworzenie nowych szablonów przebiega właściwie tak samo jak kreowanie nowego dokumentu (w tym przykładzie bez korzystania z kreatora). Podstawowa różnica polega na tym, że nie trzeba dodawać zawartości (tylko elementy potrzebne do utworzenia nowych stylów), a także na tym, że plik zapisuje się właśnie jako szablon dokumentu, a nie sam dokument.

Aby utworzyć nowy szablon, należy wykonać poniższe czynności.

1. W oknie programu Writer wybrać z menu *Plik/Nowy/Szablony i dokumenty*, zostanie wyświetlone okno dialogowe widoczne na rysunku 10.19.
2. Z panelu znajdującego się po lewej stronie trzeba wybrać ikonę *Nowy dokument*, a następnie w panelu środkowym zaznaczyć pozycję *Dokument tekstowy*. Następnie należy kliknąć przycisk *Otwórz*, co spowoduje otwarcie nowego okna programu Writer. Tu trzeba wprowadzić dowolne elementy stylistyczne, które mają pojawić się w nowym szablonie. W poprzednim punkcie powiedziałem, jak utworzyć nowe style w oparciu o już występujące w dokumencie.

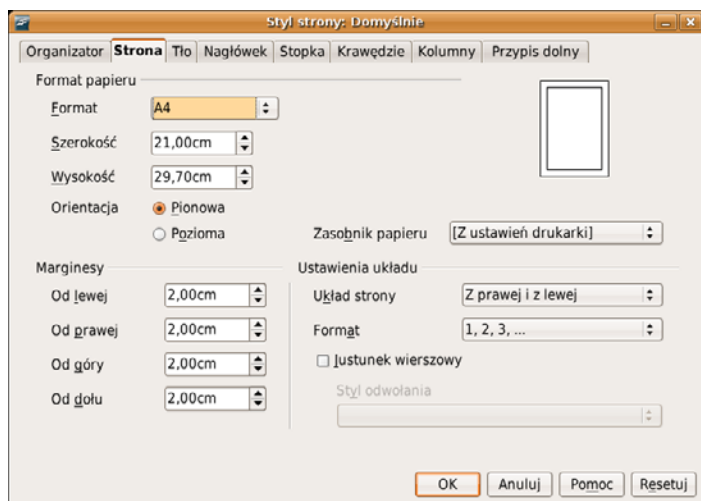


Rysunek 10.19. Tworzenie nowego szablonu

3. Następnie należy się upewnić, że ustawienia strony są odpowiednie dla nowych stylów. Inną, oprócz tworzenia nowych stylów, zmianą, którą być może użytkownik będzie chciał wprowadzić, jest zmiana domyślnych ustawień strony. Aby to zrobić, należy z menu wybrać *Format/Strona*, następnie przejść do zakładki *Strona* i tam wprowadzić odpowiednie modyfikacje dotyczące rozmiaru, marginesów itp. (patrz rysunek 10.20).

Rysunek 10.20.

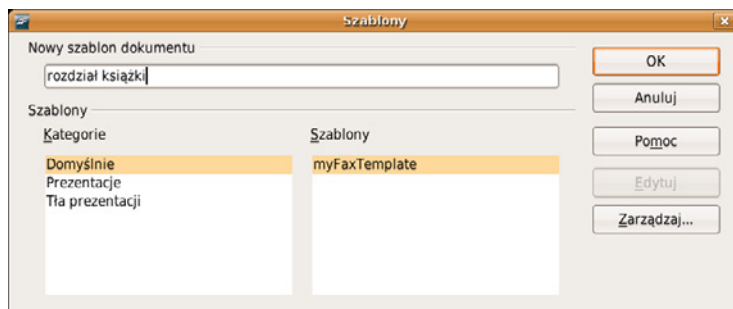
*Modyfikowanie
ustawień strony*



4. Po wprowadzeniu wszystkich zmian należy dokument zapisać jako nowy szablon; aby to zrobić, trzeba kolejno z menu wybrać *Plik/Szablony/Zapisz*, a zostanie wyświetlone nowe okno dialogowe (patrz rysunek 10.21).
5. Teraz trzeba wprowadzić nazwę nowego szablonu i kliknąć *OK*. Na rysunku 10.21 widać nowo utworzony styl o nazwie rozdział książki.

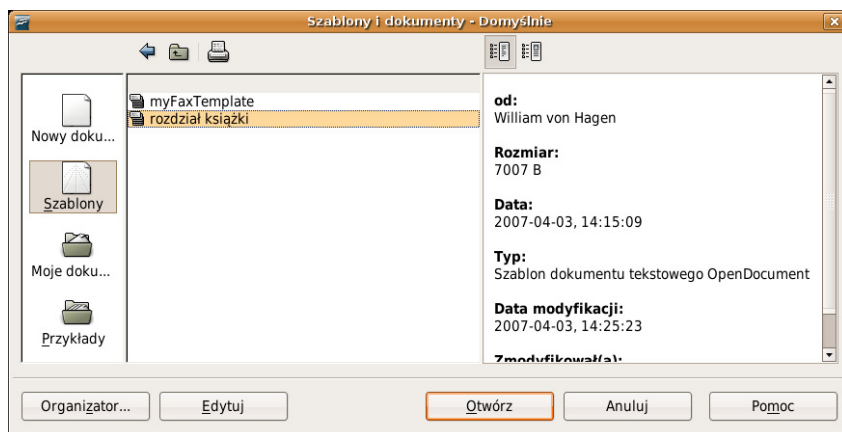
Rysunek 10.21.

Zapisywanie
nowego szablonu



Gratulacje! Właśnie został utworzony szablon dokumentu. Teraz, opierając się na nim, łatwo można tworzyć nowe dokumenty. Aby to zrobić, trzeba wykonać kolejno następujące kroki.

1. Wybrać z menu *Plik/Nowy/Szablony i dokumenty*, zostanie wyświetlone okno dialogowe widoczne na rysunku 10.19.
2. Aby utworzyć nowy dokument na podstawie istniejącego szablonu, trzeba wybrać w lewym panelu ikonę *Szablony*, a następnie wskazać miejsce, w którym szablon jest zapisany. Zostaną wówczas wyświetlone dostępne szablony. Na rysunku 10.22 przedstawiam okno dialogowe z szablonem, który został utworzony w poprzednim przykładzie.



Rysunek 10.22. Tworzenie nowego dokumentu na podstawie szablonu

3. Po zaznaczeniu odpowiedniego szablonu należy kliknąć dwukrotnie jego nazwę lub użyć przycisku *OK*.

Po zakończeniu pracy z plikiem trzeba go zapisać, używając standardowego polecenia *Plik/Zapisz*.

Tworzenie i używanie szablonów to doskonała możliwość standaryzacji dokumentów używanych w domu lub w pracy.



Wprowadzenie zmian do szablonu nie spowoduje automatycznego zaktualizowania wszystkich dokumentów utworzonych na jego podstawie. Po otwarciu takiego dokumentu program zapyta użytkownika, czy ma zaktualizować szablon. To dobre rozwiązanie i miło, że Writer sam się o to troszczy.

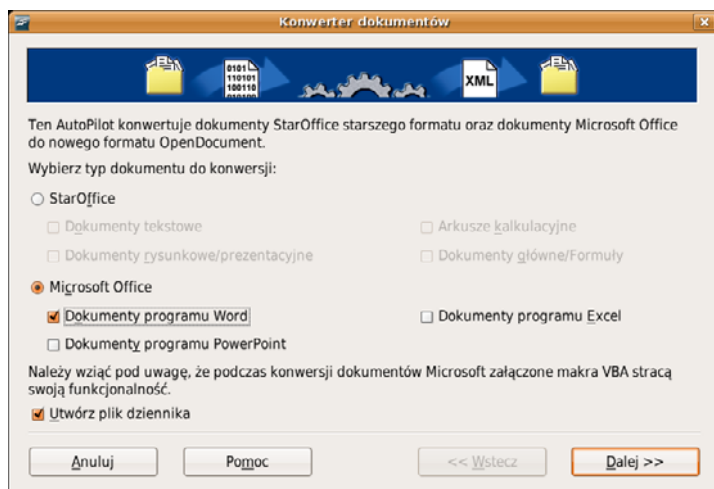
Importowanie plików utworzonych w innych procesorach tekstu

W programie Writer można otwierać dokumenty utworzone w innych edytorach, takich jak Microsoft Word, Corel WordPerfect, czy dokumenty przechowywane w formacie RTF (ang. *Rich Text Format*). Jest to wystarczające rozwiązanie, kiedy trzeba przenieść jeden lub dwa pliki, jeżeli jednak użytkownik chciałby przenieść wszystkie swoje dokumenty do Ubuntu, może się okazać, że będą ich setki. Na szczęście, w programie Writer dostępne jest specjalne narzędzie, które przeprowadzi import nie tylko samych dokumentów, ale również wykrytych w nich stylów.

Aby zaimportować zestaw dokumentów utworzonych w programie Microsoft Word (lub starszej wersji OpenOffice.org), należy kolejno:

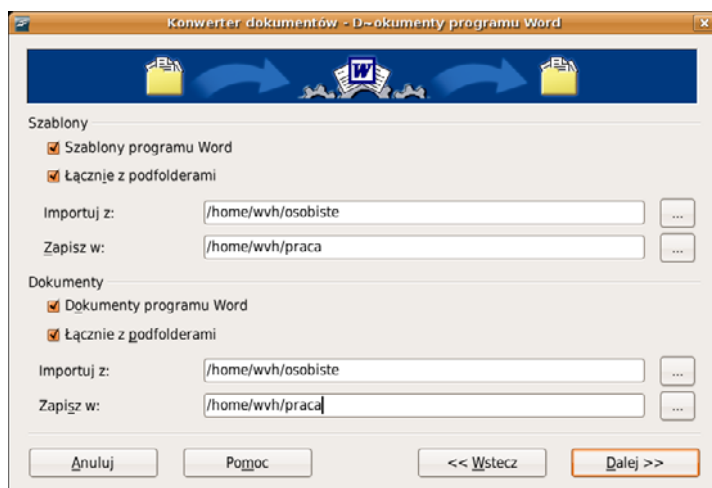
1. Wybrać z menu *Plik/Kreatory/Konwerter dokumentów*, zostanie wówczas wyświetlone okno widoczne na rysunku 10.23.

Rysunek 10.23.
Określanie formatów
wejściowych podczas
konwertowania
prezentacji



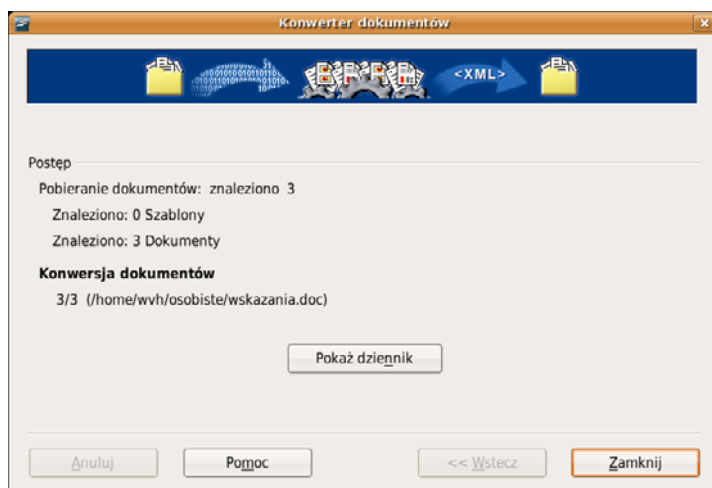
2. Aby przeprowadzić import dokumentów oraz stylów utworzonych w programie Microsoft Word, należy zaznaczyć kontrolkę *Microsoft Office* oraz pole wyboru *Dokumenty programu Word*. Aby zaimportować dokumenty z programu StarOffice lub starszej wersji OpenOffice.org, należy pozostawić zaznaczoną kontrolkę *StarOffice*. Następnie trzeba kliknąć przycisk *Dalej*. Zostanie wyświetlone okno widoczne na rysunku 10.24.
3. Następnie należy wskazać lokalizację, w której kreator ma wyszukać dokumenty, a także miejsce, w jakim mają zostać zapisane już po przeprowadzeniu procesu konwersji. Program zaproponuje foldery, ale użytkownik może określić je

Rysunek 10.24.
Określanie położenia dokumentów, które mają zostać importowane



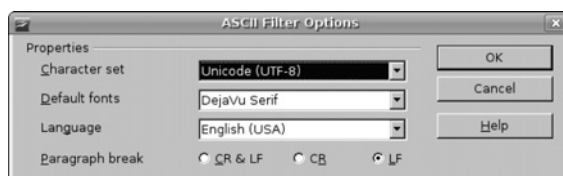
samodzielnie. Zwykle używam folderu domowego, a dopiero później przenoszę pliki do odpowiednich folderów. Zostanie wyświetlone okno dialogowe z informacjami podsumowującymi proces konwersji plików (patrz rysunek 10.25).

Rysunek 10.25.
Informacje o procesie konwersji dokumentów

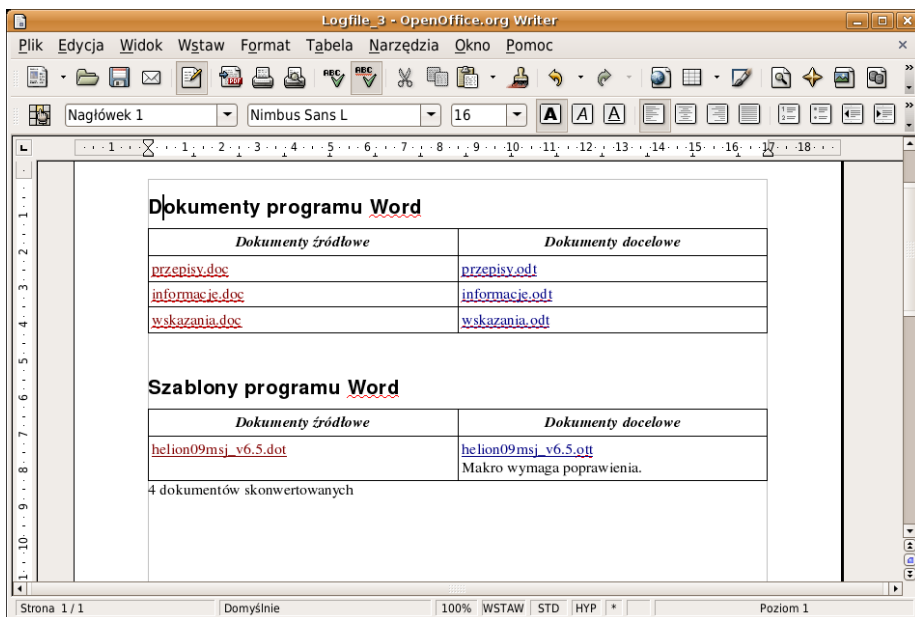


- Podczas procesu konwersji program może wyświetlić okno z informacją na temat kodowania znaków (patrz rysunek 10.26). Oznacza to zwykle, że znaleziono plik, w którym nie udało się odczytać pewnych znaków. Po zakończeniu tego procesu zostaną aktywowane przyciski w oknie widocznym na rysunku 10.25.

Rysunek 10.26.
Okno dialogowe, za którego pomocą można określić kodowanie znaków



Gratulacje! Proces konwersji został zakończony. Dokumenty i szablony są gotowe do użycia. Nim jednak użytkownik zamknie okno, dobrym pomysłem jest przejrzanie dziennika. Jak widać na rysunku 10.27, w pliku dziennika można znaleźć informację o plikach, które zostały poddane procesowi konwersji. Jeżeli znajdzie taka potrzeba, pojawi się również informacja o plikach, które wymagają dodatkowych zabiegów, nim będzie można z nich skorzystać w programie Writer. Często ma to miejsce w przypadku plików programu w formacie *.dot* Microsoft Word, które zawierają makra kompatybilne tylko z językiem Visual Basic.



Rysunek 10.27. Podsumowanie procesu konwertowania dokumentów

Dodatkowe informacje o programie Writer

Dla programu Writer dostępny jest kompletny system pomocy. Jednakże, jak w większości tego typu programów, nie znajdziemy w nim porad typu „jak zrobić...”.

Dla użytkowników, którzy potrzebują tego rodzaju pomocy, najlepszym źródłem informacji, jakie udało mi się znaleźć, będzie strona <http://tutorialsforopenoffice.org>. Zawiera znakomity zbiór poradników zapisanych w plikach PDF i formatach pakietu OpenOffice.org, a także zbiór łączy do innych stron z praktycznymi informacjami na temat korzystania z Writera. Warto odwiedzić tę stronę. Kiedy już użytkownik stanie się ekspertem w dziedzinie obsługi Writera, być może sam będzie chciał napisać jakiś poradnik i umieścić go na tej stronie, zwłaszcza że można tam znaleźć podręcznik o tworzeniu poradników.

Komputerowy skład tekstu za pomocą programu Scribus

W dwóch poprzednich podrozdziałach opisałem tworzenie w Ubuntu różnego rodzaju dokumentów za pomocą narzędzi linii poleceń oraz graficznego procesora tekstu. Umożliwiają one wykorzystanie stylów, dzięki którym można ustandaryzować wygląd tworzonych dokumentów lub wygląd powtarzających się w nich elementów. Jednak w sytuacji, kiedy wygląd tekstu jest tak samo ważny jak jego zawartość, wówczas potrzebne jest inne narzędzie. Za pomocą edytorów i procesorów tekstu można przeprowadzić wstępne formatowanie, ale zwykle oferowane możliwości ograniczają się do kontrolowania wielkości marginesów, obszaru wydruku i liczby kolumn tekstu wyświetlanych na danej stronie. Aby szczegółowo edytować układ strony, potrzebne jest narzędzie łączące w sobie możliwości procesora tekstu ze szczegółowym edytowaniem układu i formatu stron, czyli narzędzie do komputerowego składu tekstu. Na szczęście, ludzie zajmujący się wolnym oprogramowaniem rozwiązali również ten problem i opracowali doskonałe narzędzie przeznaczone do tego celu; jest to aplikacja o nazwie Scribus, którą można łatwo zainstalować w Ubuntu.

Prace nad Scribusem trwają już od kilku lat, obecnie program ten jest dostępny — bez ponoszenia żadnych kosztów — dla Linuksa, Apple Mac OS X (<http://aqua.scribus.net>) i Microsoft Windows (<http://windows.scribus.net>), a ponieważ program korzysta z biblioteki QT, wygląda i działa tak samo we wszystkich tych systemach. Strona domowa programu znajduje się pod adresem www.scribus.net, zawsze można tam znaleźć najnowsze informacje na temat Scribusa.

Podstawowym pojęciem w komputerowym składzie tekstu jest **ramka**, którą można porównać do akapitu używanego w procesorach tekstu. Ramka to kluczowy element w składzie. Dokumenty poddawane temu procesowi są komponowane z wielu różnych ramek, które mogą zawierać tekst, grafikę itp. Podzielenie dokumentu na takie składniki sprawia, że łatwo można je — ramki — przesuwać czy zmieniać, niezależnie od pozostałych elementów. Tekst w różnych ramkach może być powiązany, dlatego może przepływać z jednej ramki do innej, ale nadal pozostają one oddzielnymi elementami, podatnymi na swobodne zarządzanie.

W kolejnym podrozdziale powiem, jak zainstalować Scribusa i utworzyć dokument za pomocą szablonów; objaśnię podstawowe czynności niezbędne do utworzenia stron ze szczegółowym układem wymaganym w zaawansowanych publikacjach. Nie jest to Quark Xpress, Adobe InDesign czy Adobe PageMaker, ale, jak mawiała moja była szefowa, „to jest wystarczająco dobre dla dziewczyn, z którymi się spotykam” (nigdy nie byłam całkiem pewny, co dokładnie miała na myśli, ale to świetne powiedzenie).

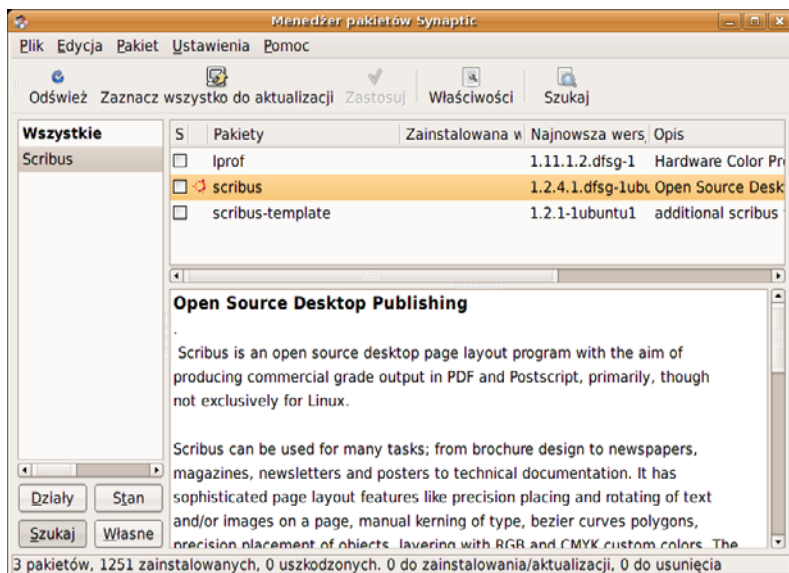
Instalowanie Scribusa w Ubuntu

Ponieważ nie jest to program, którego wszyscy potrzebują lub chcieliby używać, dlatego Scribus nie został domyślnie zainstalowany w Ubuntu. Aby go dodać do systemu, należy uruchomić menedżera pakietów Synaptic (menu *Plik/Administracja*), kliknąć przycisk *Szukaj*, wpisać w polu tekstowym Scribus i wcisnąć przycisk *Szukaj*. Po zakończeniu

wyszukiwania zostanie wyświetlone okno dialogowe z jego wynikami (patrz rysunek 10.28). Należy zaznaczyć prawym przyciskiem myszy pakiet *scribus* i z menu kontekstowego wybrać opcję *Zaznacz do instalacji*.

Rysunek 10.28.

Instalowanie programu Scribus za pomocą programu Synaptic



W podobny sposób można też dodatkowo zaznaczyć do zainstalowania pakiet *scribus-template*, zawierający dodatkowe szablony dokumentów.

Następnie należy kliknąć przycisk *Zastosuj* znajdujący się na pasku narzędzi Synaptica, co spowoduje dodanie Scribusa do systemu. Po zakończeniu procesu instalowania aplikacja jest gotowa do użycia. Aby ją uruchomić, należy z menu wybrać kolejno *Aplikacje/Biuro/Scribus*.

W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, zamieszczam więcej informacji na temat instalowania programów za pomocą menedżera pakietów Synaptic.

Krótki przewodnik po programie Scribus

Podobnie jak program OpenOffice.org Writer i większość aplikacji związanych z edytowaniem dokumentów, również Scribus oferuje na początek kilka gotowych szablonów. Kiedy używa się jakiegoś programu po raz pierwszy, łatwiej rozpocząć pracę z gotowymi plikami i wprowadzać do nich zmiany. Kiedy już użytkownik nabierze wprawy w obsłudze, zacznie samodzielnie tworzyć dokumenty, zgodnie z informacjami zamieszczonymi w dalszych punktach. Poniżej błyskawicznie omówię podstawowe funkcje Scribusa, tak aby użytkownik mógł się po nim komfortowo poruszać.

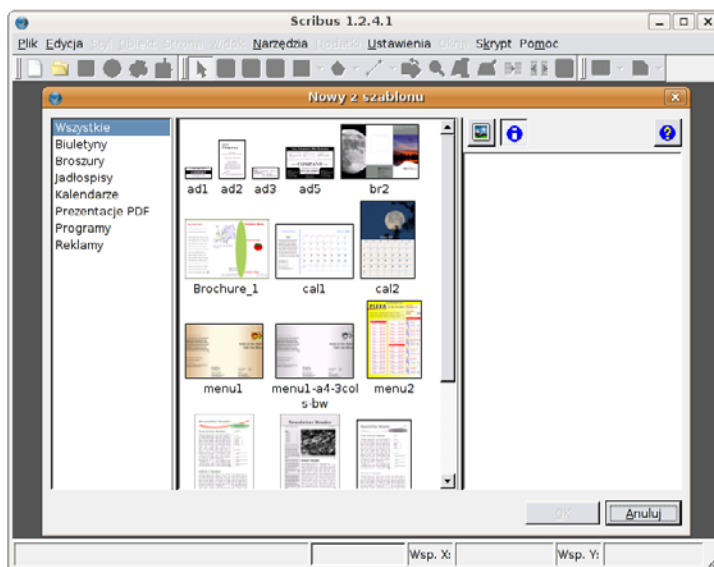
Aby uruchomić program, należy z menu wybrać kolejno *Aplikacje/Biuro/Scribus*; samo uruchomienie trwa chwilę, ponieważ program musi sprawdzić, jakie czcionki są dostępne

w systemie, bowiem jest to element o znaczeniu krytycznym dla programów do edycji układu stron. Po włączeniu programu można utworzyć nowy dokument, korzystając z jednego z dołączonych szablonów. Oto informacje, jak to zrobić.

1. Z menu należy kolejno wybrać *Plik/Nowy z szablonu*, zostanie wyświetlone okno dialogowe widoczne na rysunku 10.29.

Rysunek 10.29.

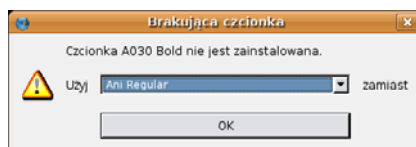
Wybór szablonu



2. Następnie należy wybrać odpowiedni szablon i kliknąć *OK*. W opisywanym przykładzie został użyty szablon *Newsletter_2*, za którego pomocą można przygotować biuletyn informacyjny zawierający grafikę. Podczas wczytywania szablonu program może wyświetlić komunikat o ostrzeżeniu, że pewne czcionki nie są dostępne w systemie (patrz rysunek 10.30). Należy zanotować każdą brakującą czcionkę i kliknąć *OK*, aby zastąpić ją inną, dostępną w systemie. Później za pomocą menedżera pakietów Synaptic można zainstalować brakujące pakiety.

Rysunek 10.30.

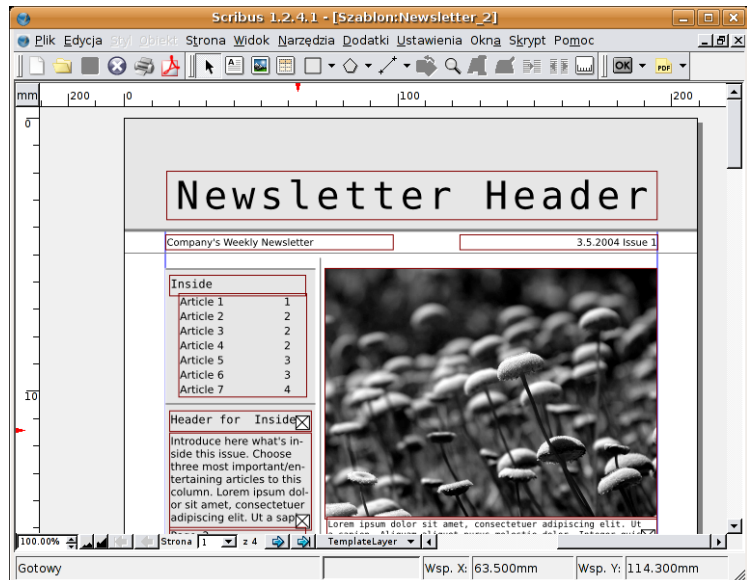
Komunikat o braku czcionek w systemie wyświetlany podczas wczytywania szablonu



3. Po zakończeniu procesu kompletowania czcionek w głównym oknie Scribusa zostanie wyświetlona kopia szablonu (patrz rysunek 10.31).
4. Teraz zaczyna robić się interesująco. Aby edytować tekst w dowolnej ramce, należy wybrać przycisk *Edytuj zawartość ramki* umieszczony na pasku narzędzi. Po jego użyciu można, do wybranej wcześniej ramki, wstawiać tekst lub go usuwać. Aby wprowadzić bardziej zaawansowane zmiany, np. dotyczące czcionek, należy wyświetlić okno dialogowe *Edytor artykułów*. Aby je przywołać, trzeba użyć przycisku *Edytuj tekst w edytorze artykułów* znajdującego się na pasku narzędzi.

Rysunek 10.31.

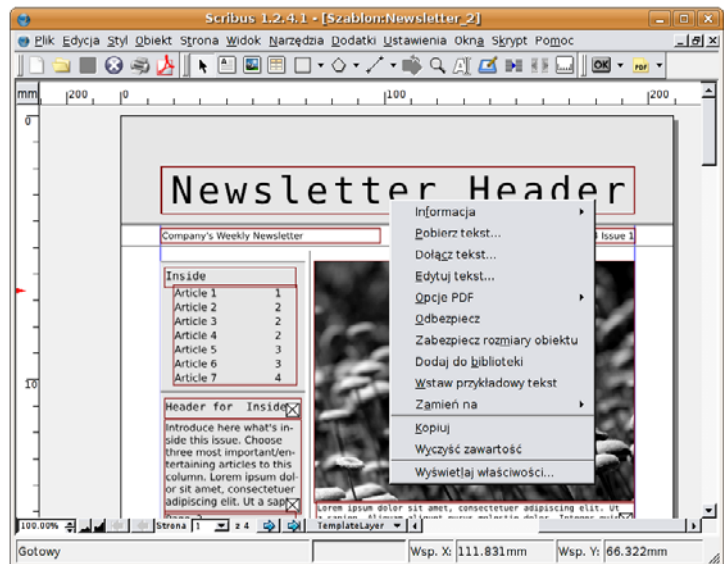
Wczytana kopia
robocza szablonu



Innym sposobem jego wyświetlenia jest użycie opcji *Edytuj tekst* dostępnej z menu kontekstowego (patrz rysunek 10.32).

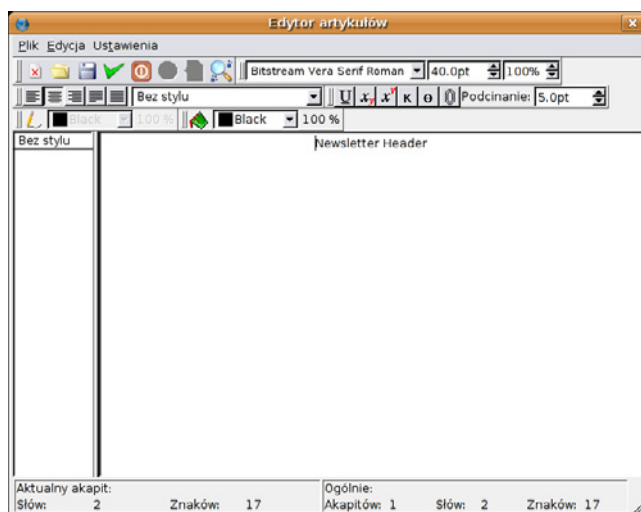
Rysunek 10.32.

Opcje edycji ramek
w programie Scribus



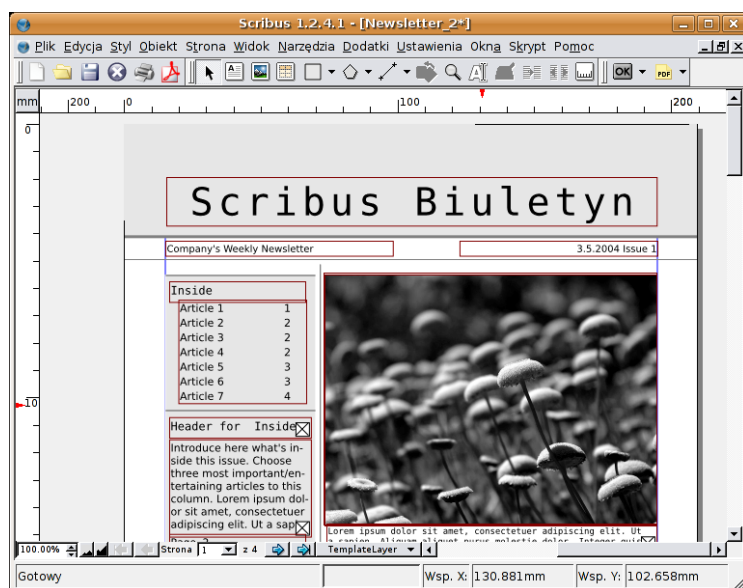
5. Aby wyświetlić okno dialogowe, w którym można formatować tekst, należy z menu wybrać *Edytuj tekst*, zostanie wyświetlone okno Edytora artykułów (patrz rysunek 10.33). Edytor ten to właściwie mały procesor tekstu wbudowany w Scribusa. Za jego pomocą można tekst zmieniać, zamieniać i dodawać do ramek. Kontrolki dostępne w programie umożliwiają inne ustawienie wyrównania tekstu w ramkach, a także użycie innych czcionek.

Rysunek 10.33.
Modyfikowanie tekstu
zamieszczonego
w ramkach



6. Po dokonaniu zmian w tekście należy kliknąć przycisk z zielnym znakiem ✓ lub wybrać z menu opcję *Aktualizuj ramkę tekstową i zakończ*. Wyświetlany plik zostanie zmodyfikowany (patrz rysunek 10.34).

Rysunek 10.34.
Zaktualizowana
wersja dokumentu

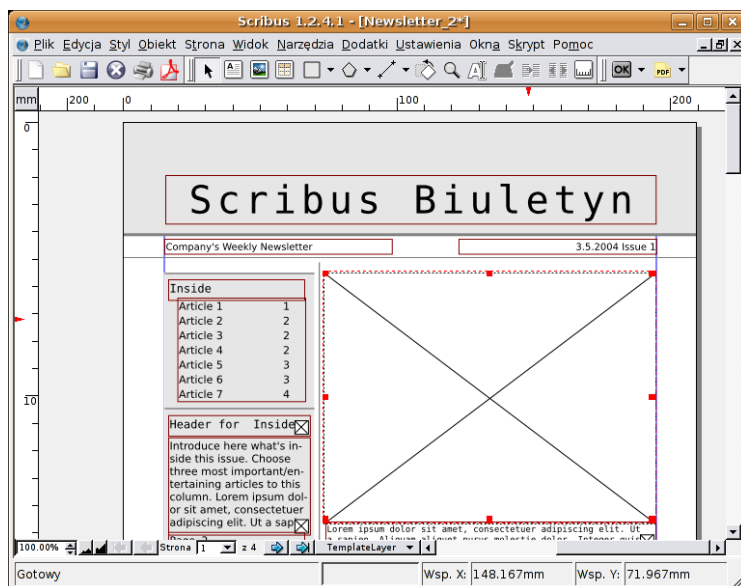


7. Praca z ramkami zawierającymi grafikę przebiega podobnie, tyle tylko, że do edycji ilustracji wykorzystywana jest zewnętrzna aplikacja (GIMP omówimy w rozdziale 12., „Praca z grafiką”). Aby np. zmienić obraz widoczny na rysunku 10.34, należy kliknąć prawym przyciskiem myszy i z menu wybrać opcję *Odblokuj*, co umożliwi edycję zawartości ramki. Podczas pracy można zabezpieczyć dowolną ramkę i zapobiec w ten sposób przypadkowej zmianie jej zawartości.

8. Aby usunąć wybrany obraz, należy z menu kontekstowego (przywoływanego wciśnięciem prawego przycisku myszy) wybrać opcję *Wyczyść zawartość*, co spowoduje usunięcie grafiki (patrz rysunek 10.35).

Rysunek 10.35.

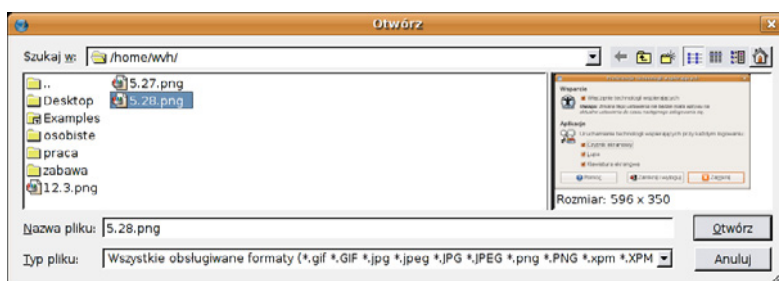
*Biuletyn
po usunięciu grafiki*



9. Aby wybrać nową grafikę, która zostanie wstawiona do ramki, należy kliknąć prawym przyciskiem myszy i wybrać z menu *Wybierz obrazek*. Zostanie wyświetlone okno dialogowe widoczne na rysunku 10.36, za jego pomocą użytkownik będzie mógł wybrać odpowiednią grafikę dostępną w systemie i umieścić ją w pliku.

Rysunek 10.36.

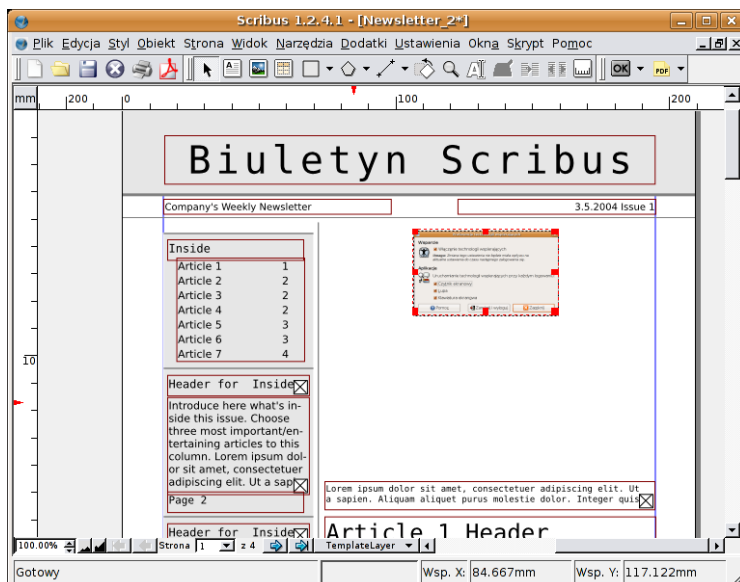
*Wybieranie
nowej grafiki
do umieszczenia
w ramce*



10. Po wybraniu nowej grafiki należy użyć przycisku *Otwórz*, a zostanie umieszczona w ramce. Na rysunku 10.37 widać wprowadzoną zmianę.
11. Ponieważ teraz grafika jest mniejsza od ramki, należy wybrać z menu kontekstowego opcję *Dopasuj ramkę do obrazka*, wtedy będzie można wykorzystać powstałe miejsce, a także przesunąć grafikę w nowe położenie.
12. Nowo powstała przestrzeń to doskonałe miejsce na nową ramkę tekstową. Aby ją dodać, należy kliknąć ikonę *Utwórz ramkę tekstową* (wygląda jak mały dokument), a następnie z wciśniętym lewym przyciskiem myszy narysować jej kształt w wolnym miejscu. W przykładzie w ramce umieściłem tekst, korzystając z opcji *Wstaw*

Rysunek 10.37.

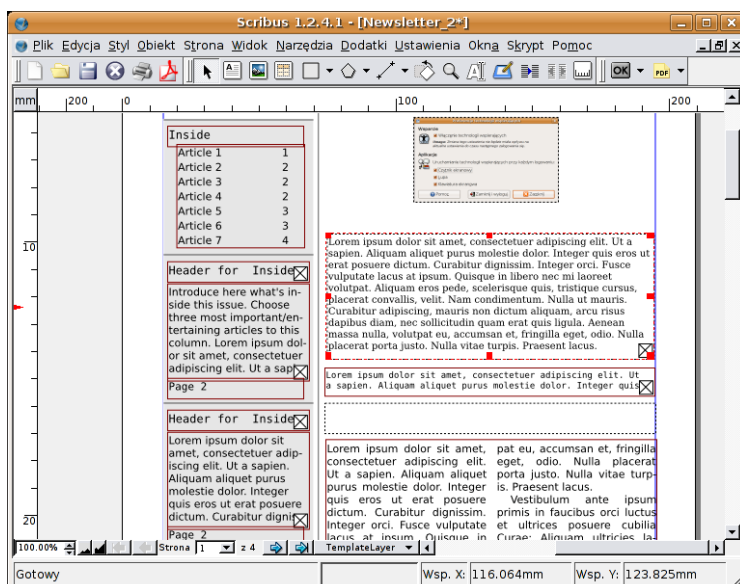
Zaktualizowany
dokument
z nową grafiką



przypadkowy tekst. Może się to wydawać dziwne, ale jest przydatne, gdy trzeba przygotować układ strony przed umieszczeniem właściwego tekstu. Po wprowadzeniu powyższych zmian dokument wygląda tak jak na rysunku 10.38.

Rysunek 10.38.

Zaktualizowany
dokument z nową
ramką tekstową

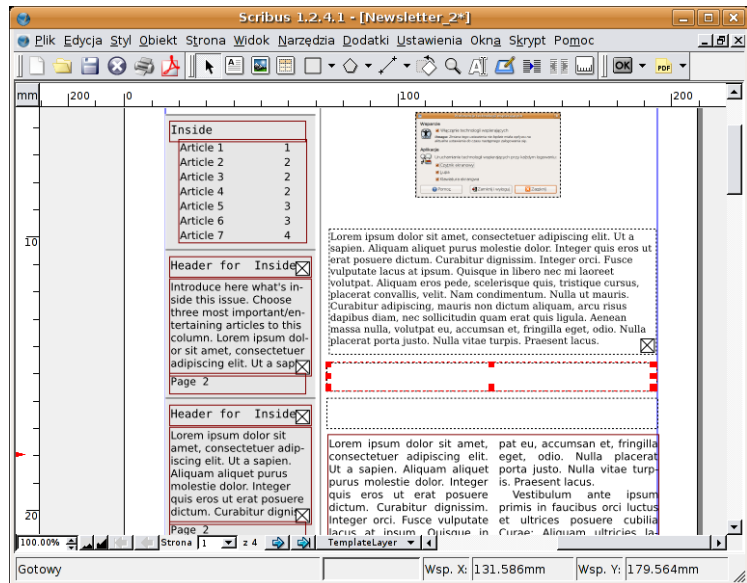


- Ostatnia czynność to połączenie dwóch ramek tekstowych tak, aby tekst swobodnie przepływał z jednej do drugiej. Można połączyć dowolne dwie ramki, ale pod warunkiem, że są puste. Aby usunąć z nich zawartość, najpierw należy je odbezpieczyć (z menu kontekstowego wybrać opcję *Odbezpiecz*), następnie trzeba wybrać opcję *Edytuj tekst*. Kiedy zawartość ramki zostanie wyświetlona w Edytorze

artykułów, należy ją usunąć i wybrać z menu *Aktualizuj ramkę tekstową i zakończ*. Ramka będzie pusta (patrz rysunek 10.39).

Rysunek 10.39.

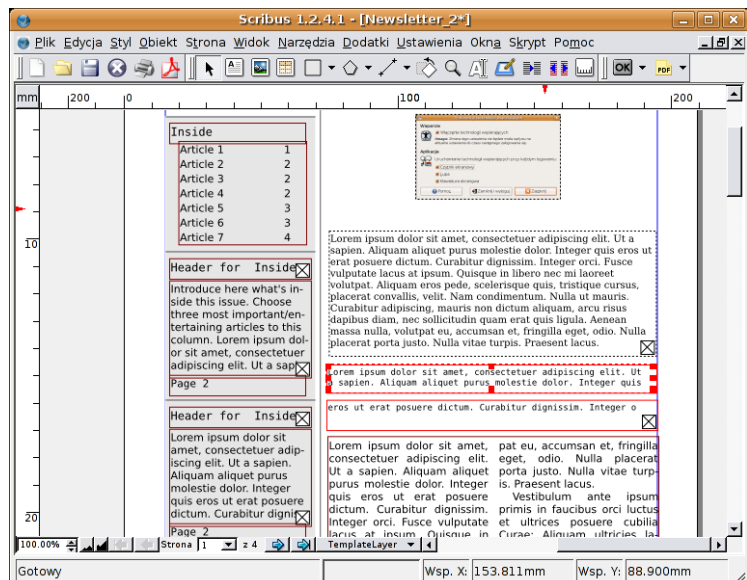
Zaktualizowany dokument z pustą ramką tekstową



14. Teraz należy zaznaczyć ramkę, z której tekst ma wypływać, a następnie kliknąć ikonę *Połącz ramki tekstowe*, znajdującą się na pasku narzędzi (to ta z dwoma kolumnami tekstu i znajdującą się między nimi strzałką, na prawo od ikony *Edytuj tekst w edytorze artykułów*). Następnie wystarczy zaznaczyć kursorem (wciskając lewy przycisk myszy) ramkę, do której tekst ma wpływać. Dokument zostanie zaktualizowany, co widać na rysunku 10.40.

Rysunek 10.40.

Zaktualizowany dokument z połączonymi ramkami



I to właściwie tyle, na tym kończę krótkie wprowadzenie. Dokument można zapisać lub wyeksportować do druku, wybierając ikonę *Zachowaj aktualny dokument jako PDF* (z symbolem plików PDF).

Scribus to użyteczna aplikacja zasługująca na osobną książkę. Tutaj opisałem podstawy tworzenia dokumentów, modyfikowania tekstu, łączenia ramek tekstowych oraz wstawiania i edycji grafiki. Scribus to imponujący przykład potęgi modelu tworzenia otwartego oprogramowania oraz determinacji kilku programistów, którzy dostrzegli potrzebę, a następnie ją zaspokoiili.

Dodatkowe informacje o programie Scribus

Scribus to stosunkowo młoda aplikacja, ale znacząca jest ilość informacji, które można znaleźć w internecie na różnych forach oraz poświęconych jej stronach. Oto kilka moich ulubionych stron.

- ♦ Strona z dokumentacją Scribusa (<http://docs.scribus.net/>) — można tu znaleźć sporą porcję informacji na temat konfigurowania programu, drukowania, zarządzania kolorami itd.
- ♦ Scribus wiki (<http://wiki.scribus.net/>) — znakomite forum do prowadzenia dyskusji i dokumentowania metod korzystania z programu.
- ♦ Samouczek, rozpoczynanie pracy ze Scribusem (<http://docs.scribus.net/index.php?lang=en&page=tutorials/freedomyug/scribustutorial>) — choć przygotowany dla starszej wersji programu i już nieco przeterminowany, nadal jest znakomitą podręcznikiem dla tych, którzy rozpoczynają swoją przygodę z programem. Na stronie dostępne są również pliki z przykładami użytymi w samouczku.

Scribus to otwarte oprogramowanie wspierane przez wolontariuszy. Każdy jego zwolennik, który będzie mógł coś wnieść, jest mile widziany.

Inne procesory tekstu i pakiety biurowe dla Linuksa

W niniejszym rozdziale omówiłem najpopularniejsze otwarte narzędzia zarówno linii poleceń, jak i graficzne, które umożliwiają edycję tekstu i układu strony. Oczywiście, istnieją inne aplikacje, ponieważ jest to wolne oprogramowanie, więc wszystko, co się robi, warto wykonać na kilka sposobów, dzięki czemu ma się możliwość wyboru. Oto kilka innych, popularnych procesorów tekstu przeznaczonych dla Linuksa.

- ♦ AbiWord — część pakietu GNOME Office (www.gnome.org/gnome-office). AbiWord to wydajny i popularny procesor tekstu dla Linuksa wyposażony w doskonałe możliwości importowania i edycji dokumentów w innych formatach (np. Microsoft Word). AbiWord jest dostępny w repozytoriach Ubuntu, a zatem łatwo dodać go do systemu, korzystając z menedżera pakietów Synaptic.

- ♦ **Applixware** — to pakiet oprogramowania biurowego zawierający również procesor tekstu. Jeden z pierwszych pakietów biurowych dla Linuksa. Początkowo był bezpośrednim rywalem produktu firmy StarDivision, czyli StarOffice’a. Z czasem przekształcił się w pakiet do projektowania i analizy aplikacji w czasie rzeczywistym. Więcej informacji na jego temat można znaleźć na stronie <http://www.vistasource.com/vs2/en/index.php>. Artykuły o instalowaniu programu w Ubuntu można znaleźć na stronie <http://groups.google.com/group/Applixware>, wystarczy poszukać hasła Ubuntu.
- ♦ **KWord** — procesor tekstu będący częścią pakietu Koffice przeznaczonego dla środowiska KDE (www.koffice.org). KWord jest wydajny i popularny, ale do działania wymaga środowiska KDE. Czytelnicy zainteresowani korzystaniem z tego programu powinni zaznajomić się z Kubuntu, czyli dystrybucją pochodną Ubuntu, w której zamiast GNOME wykorzystywane jest KDE. KWord dostępny jest w repozytoriach Ubuntu, dzięki czemu łatwo można go zainstalować, używając menedżera pakietów Synaptic. Zainstalowanie spowoduje dodanie do systemu kilku, wymaganych do uruchomienia, komponentów środowiska KDE.

Wszystkie wymienione procesory są proste w użytkowaniu i mają grono swoich fanów. Jeżeli użytkownik znajdzie coś, co mu się nie spodoba w OpenOffice.org, zawsze może zainstalować i wypróbować jeden z wymienionych programów. Ponad to, takiej ceny nie przebiję.

Podsumowanie

W rozdziale opisałem, w jaki sposób można w Ubuntu wykonać kilkanaście podstawowych operacji związanych z edytowaniem tekstu. Rozpocząłem od omówienia TeX-a i LaTeX-a, przeznaczonych dla Uniksa i Linuksa procesorów tekstu działających w linii poleceń, dla których dostępne są również narzędzia graficzne. Następnie przybliżyłem program OpenOffice.org Writer, komponent potężnego pakietu, a także wyjaśniłem, jak za jego pomocą tworzyć i edytować dokumenty. Ostatni podrozdział poświęciłem Scribusowi, programowi do komputerowego składu tekstu, który umożliwia odpowiednie przygotowanie dokumentów wówczas, gdy ich wygląd jest tak samo ważny jak zawartość. Na końcu pokrótce opisałem kilka innych procesorów tekstu, które można wykorzystać w Linuksie i tym samym w Ubuntu.

Narzędzia do tworzenia i edycji dokumentów, dostępne w Ubuntu, są znakomite, ale nie samymi dokumentami żyją użytkownicy. W następnym rozdziale omówię kolejne programy przydatne w codziennej pracy biurowej, czyli arkusze kalkulacyjne i program do tworzenia prezentacji.

Rozdział 11.

Pozostałe składniki pakietu biurowego: arkusz kalkulacyjny Calc i program do tworzenia prezentacji

W tym rozdziale:

- ♦ Wprowadzenie do arkuszy kalkulacyjnych — przewodnik
- ♦ Korzystanie z Gnumeric
- ♦ Praca z OpenOffice Calc
- ♦ Tworzenie prezentacji w programie Impress

W poprzednim rozdziale napisałem, jak w systemie Ubuntu tworzyć dokumenty. Zapewne najczęściej wykonywanymi zadaniami na nowym komputerze będą surfowanie po sieci, wysyłanie oraz odbieranie wiadomości elektronicznych, pisanie listów, tworzenie listy zadań do wykonania czy praca nad nową powieścią. Obecnie edytowanie tekstów należy do jednej z najbardziej podstawowych czynności wykonywanych w firmach, bo choć przedsiębiorstwa coraz częściej korzystają z poczty elektronicznej, zawsze może pojawić się konieczność sporządzenia sprawozdania, strategii itp.

W tym rozdziale omówię pozostałe części pakietu aplikacji biurowych dostępnych w Ubuntu, czyli arkusz kalkulacyjny oraz program do tworzenia prezentacji. Choć bardzo niewielu użytkowników domowych wykorzystuje na komputerach domowych prezentacje (może pod tytułem „Budżet domowy na bieżący miesiąc — co to dla nas oznacza?”), to już arkusz kalkulacyjny pełni tak samo ważne zadania w firmach, jak w domach. Użytkownicy pracujący w domach lub używający komputerów przenośnych dla celów zawodowych

mogą potrzebować skoroszytu arkusza kalkulacyjnego czy też prezentacji danych, a dzięki Ubuntu mogą to zrobić szybko i prosto (tak prosto, jak w każdym innym systemie operacyjnym).

Niezależnie od typu instalacji (Desktop, OEM czy instalacji w trybie tekstowym), w Ubuntu znajdzie się pakiet biurowy OpenOffice.org, a za pomocą łatwego w użyciu menedżera pakietów Synaptic, zawsze można zainstalować inne arkusze kalkulacyjne (np. równie potężny, choć o mniejszych wymaganiach Gnumeric). Oprócz tego, w systemie automatycznie zostanie umieszczony program do tworzenia prezentacji Impress, również będący częścią pakietu OpenOffice.org.



Aby zaoszczędzić sobie każdorazowego wpisywania pełnych nazw programów OpenOffice.org Impress oraz OpenOffice.org Calc, w dalszej części rozdziału będą używane ich skrócone — chociaż oficjalne — nazwy, czyli Impress i Calc.

Wprowadzenie do arkuszy kalkulacyjnych — przewodnik

Poniżej zamieszczam podstawowe informacje o tym, czym jest arkusz kalkulacyjny, jak działa oraz jak go używać. Czytelnicy, którzy już znają te informacje, mogą pominąć tę część i przejść do kolejnej. Jeżeli nie, radzę, by zapoznać się z zamieszczonymi tu informacjami.

W świecie dokumentów arkusz kalkulacyjny to płachta papieru podzielona na wiersze i kolumny. Każdy taki prostokąt znajdujący się na arkuszu to **komórka**. Każdy z wierszy dostarcza różnorodnych informacji na temat określonych transakcji czy podobnych zagadnień. Kolumny z kolei służą do porównywania, sumowania i wykonywania innych operacji matematycznych na wartościach zawartych w wierszach, a segregowanych zgodnie z kolumnami. Oto taki przykład przedstawiony w formie tekstowej:

posiłek	koszt	podatek	napiwek
śniadanie	5,79	0,41	1,25
obiad	8,48	0,60	1,75
kolacja	13,23	0,93	2,75

W tej prostej tabeli przedstawiono dzienne zestawienie posiłków. W pierwszym wierszu znajdują się nagłówki identyfikujące zawartość każdej kolumny. Kolejne wiersze dostarczają informacji na temat każdego z posiłków dla danego dnia. Dzięki takiemu układowi łatwo będzie uzyskać całkowity koszt posiłku i to na różne sposoby: np. dodając na końcu dodatkową kolumnę, w której sumowane będą wpisy w danym wierszu, dzięki czemu będzie można poznać koszt każdego posiłku. Można też dodać na dole kolejny wiersz, w jakim sumowane będą wartości zamieszczone w każdej z kolumn, które będzie można dodać do siebie. Preferuję to pierwsze rozwiązanie, czyli zastosowanie dodatkowej kolumny podsumowującej. Oczywiście, nie ma żadnych przeciwwskazań, aby każdy używał wygodnego dla siebie sposobu (lub łączył różne sposoby) — szerokie możliwości arkuszy kalkulacyjnych do wykonywania różnorodnych obliczeń są jednym z przyjemniejszych aspektów korzystania z tych aplikacji.



Poniżej zamieszczono wskazówki, jak prosto utworzyć arkusz kalkulacyjny. Instrukcje te zadziałają zarówno wtedy, kiedy użytkownik będzie korzystał z Gnumeric, jak i z OpenOffice.org Calc. Oczywiście, czytelnik nie musi zapoznawać się z wszystkimi informacjami, może przejść wprost do akapitów poświęconych używanemu przez siebie programowi. Gnumeric jest nieco mniej rozbudowany, dzięki czemu może wydawać się prostszy w obsłudze; ale jest wiele powodów przemawiających za używaniem OpenOffice.org, zwłaszcza jeżeli użytkownik będzie chciał korzystać z pozostałych części pakietu. W pierwszej części zamieściłem wskazówki, jak korzystać z Gnumeric, ale mają one też zastosowanie do Calca; jedyne różnice pomiędzy tymi programami dotyczą układu menu oraz wyglądu pasków narzędzi.

Jeżeli czytelnik nie posiada komputera lub jest biegły w wykonywaniu obliczeń matematycznych, wówczas można liczyć na kartce, jednak ja preferuję korzystanie z odpowiednich narzędzi, jeżeli tylko są dostępne — w tym przypadku kołem ratunkowym jest arkusz kalkulacyjny. Pracę z tym narzędziem należy rozpocząć od wprowadzenia do komórek odpowiednich wartości; do wybierania odpowiednich komórek, w których mają być wpisywane wartości, można użyć myszy. Na rysunku 11.1 przedstawiam arkusz z wprowadzonymi danymi.

Rysunek 11.1.

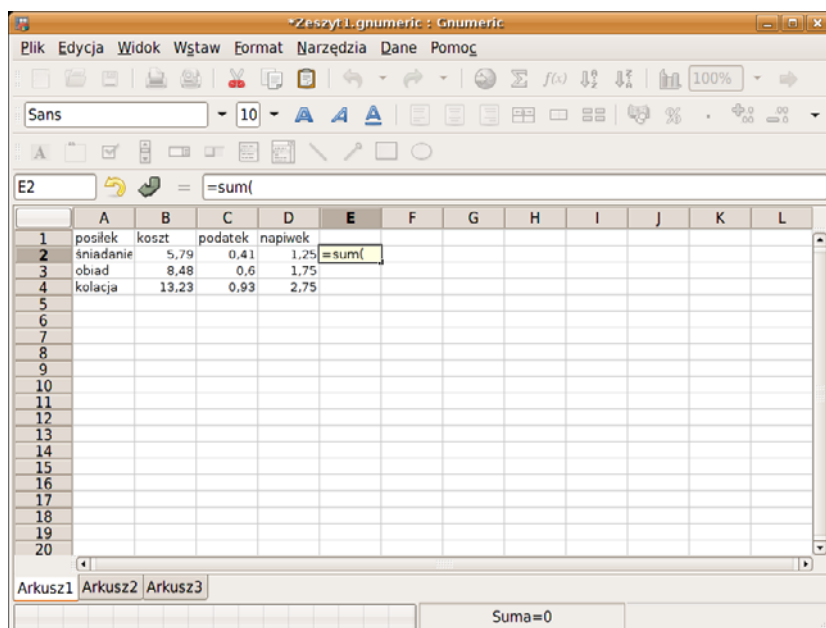
Przykładowe wartości wprowadzone do arkusza Gnumeric

	A	B	C	D	E	F	G	H	I	J	K	L
1	posilek	koszt	podatek	napiwek								
2	śniadanie	5,79	0,41	1,25								
3	obiad	8,48	0,6	1,75								
4	kolacja	13,23	0,93	2,75								
5												
6												
7												
8												
9												
10												
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												

Arkusz1 Arkusz2 Arkusz3

Suma=0

Teraz rozpoczyna się zabawa. Aby w wybranej komórce znalazła się zsumowana wartość wszystkich komórek z danego wiersza, należy zaznaczyć wybraną komórkę, a następnie wpisać `=sum(`. Znak równości oznacza, że zostanie użyta funkcja, `sum` to nazwa właśnie jednej z takich funkcji, a nawias otwierający umożliwi wprowadzenie zakresu komórek, które mają być objęte działaniem funkcji. W powyższym przykładzie można też użyć myszy do zaznaczenia zakresu komórek (więcej szczegółów na ten temat zamieszczam w dalszej części), ale wpisanie funkcji za pomocą klawiatury pozwoli szybciej przyswoić sobie zasadę ich działania. Na rysunku 11.2 znajduje się okno arkusza po wprowadzeniu tych danych.

Rysunek 11.2.Definiowanie
formuły

Następnie należy określić wartości, które mają być zsumowane. Najłatwiej to wykonać, korzystając z myszy. Aby zaznaczyć komórki, należy najpierw zaznaczyć komórkę znajdującą się w drugiej kolumnie (koszt) i z wciśniętym przyciskiem myszy przeciągnąć zaznaczenie na dwie sąsiednie komórki w kolejnych kolumnach. Zaznaczone w taki sposób komórki noszą nazwę **zakresu** komórek. Po określeniu zakresu komórek odpowiedni obszar zostanie zaznaczony przerywaną linią (obszar zwany jest często polem wyboru), dodatkowo zostanie on podany w polu formuły (patrz rysunek 11.3). Jak można zauważyć na rysunku 11.3, komórki identyfikowane są zgodnie z kolumnami i wierszami, w których się znajdują: B2 odpowiada komórce położonej w kolumnie B w wierszu 2. Podając zakres komórek dla formuły, należy wskazać pierwszą oraz ostatnią komórkę, które będą wykorzystane, i oddzielić je dwukropkiem. W tym przypadku zapis B2:D2 oznacza „wszystkie komórki pomiędzy B2 i D2, włącznie z nimi”.

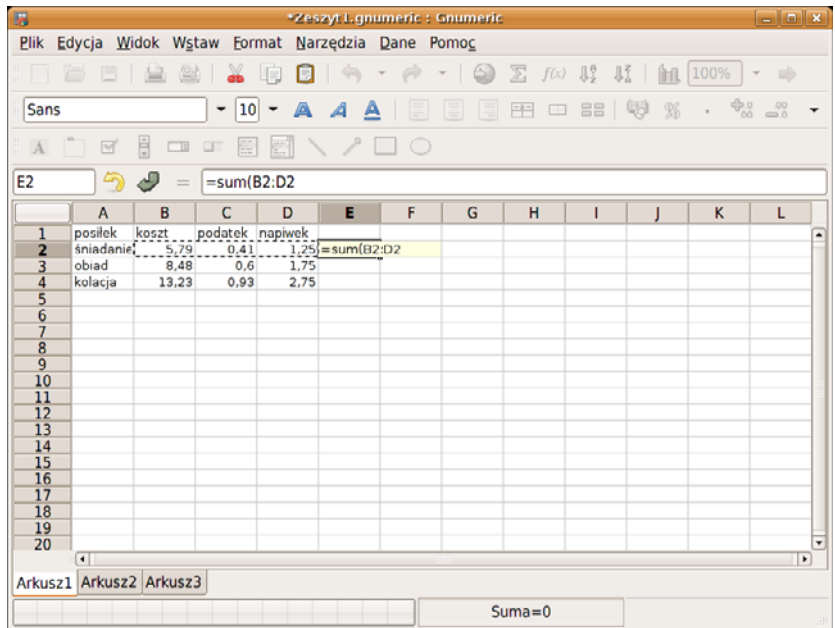
Aby po określeniu zakresu komórek dla danej formuły zaczęła ona działać, należy zamknąć nawias (patrz rysunek 11.4).

Następnie należy wcisnąć klawisz *Enter* i sprawdzić efekt działania formuły. Program zsumuje wartości w podanych komórkach, wyświetli odpowiednią wartość i przeniesie kursor do następnej komórki (patrz rysunek 11.5). To proste!

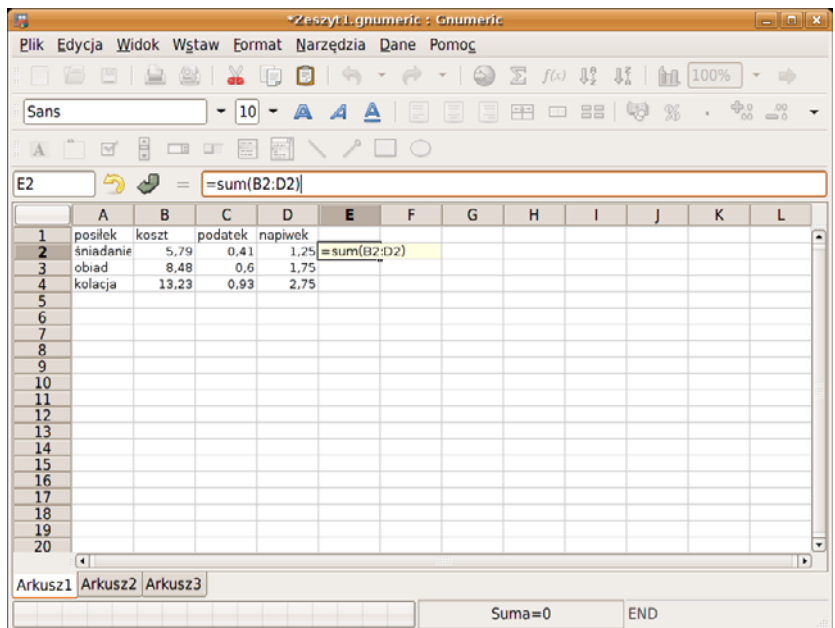
Teraz można zrobić to samo w dwóch pozostałych kolumnach; jednak można to wykonać dużo łatwiej, korzystając z możliwości arkusza kalkulacyjnego. Najprostszym sposobem na użycie w kolejnych komórkach tej samej formuły jest jej skopiowanie. Po równoczesnym skopiowaniu zawartości komórki z wynikiem działania formuły do pozostałych komórek w tej kolumnie program przyjmie, że użytkownik chce użyć w kolejnych komórkach funkcji, która została skopiowana z pierwszej komórki. A ludzie mówią, że komputery nie są sprytne.

Rysunek 11.3.

Określanie
zakresu komórek
dla formuły

**Rysunek 11.4.**

Kończenie
wprowadzania
formuły



Aby skopiować funkcję z danej komórki, należy ją zaznaczyć, a następnie wybrać z menu *Edycja/Skopiuj*. Komórka zostanie zaznaczona przerywaną linią (patrz rysunek 11.6).

Następnie należy zaznaczyć komórki, do których ma zostać wklejona formuła (trzeba to zrobić w taki sam sposób jak poprzednio podczas definiowania zakresu komórek dla funkcji). Aby to zrobić, należy przejść do komórki znajdującej się poniżej tej, której zawartość

Rysunek 11.5.
Wyświetlanie
obliczonej wartości

The screenshot shows a Gnumeric spreadsheet window titled '*Zeszyt1.gnumeric : Gnumeric'. The menu bar includes Plik, Edycja, Widok, Wstaw, Format, Narzędzia, Dane, and Pomoc. The toolbar shows various icons for file operations, editing, and formatting. The formula bar shows 'E3 ='. The spreadsheet grid has columns A through L and rows 1 through 20. The data is as follows:

	A	B	C	D	E	F	G	H	I	J	K	L
1	posilek	koszt	podatek	napiwek								
2	śniadanie	5,79	0,41	1,25	7,45							
3	obiad	8,48	0,6	1,75								
4	kolacja	13,23	0,93	2,75								
5												
6												
7												
8												
9												
10												
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												

The status bar at the bottom shows 'Suma=0'.

Rysunek 11.6.
Kopiowanie
komórek w arkuszu

The screenshot shows the same Gnumeric spreadsheet window. The formula bar now shows 'E2 = =sum(B2:D2)'. The data in the spreadsheet is the same as in the previous screenshot, but the value in cell E2 is now 7,45, which is the sum of the values in cells B2, C2, and D2.

	A	B	C	D	E	F	G	H	I	J	K	L
1	posilek	koszt	podatek	napiwek								
2	śniadanie	5,79	0,41	1,25	7,45							
3	obiad	8,48	0,6	1,75								
4	kolacja	13,23	0,93	2,75								
5												
6												
7												
8												
9												
10												
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												

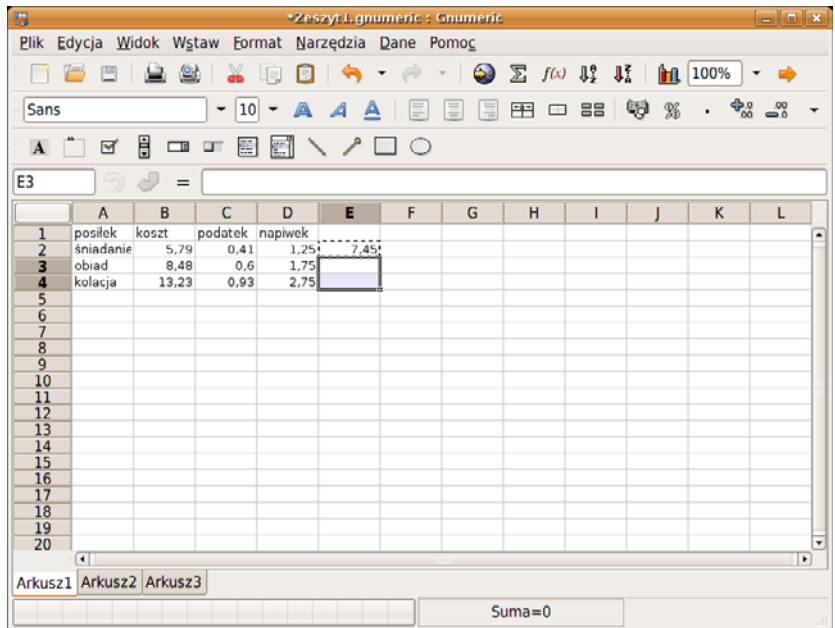
The status bar at the bottom now shows 'Suma=7,45'.

jest kopiowana, a następnie z wciśniętym przyciskiem myszy przeciągnąć zaznaczenie na kolejne komórki (patrz rysunek 11.7).

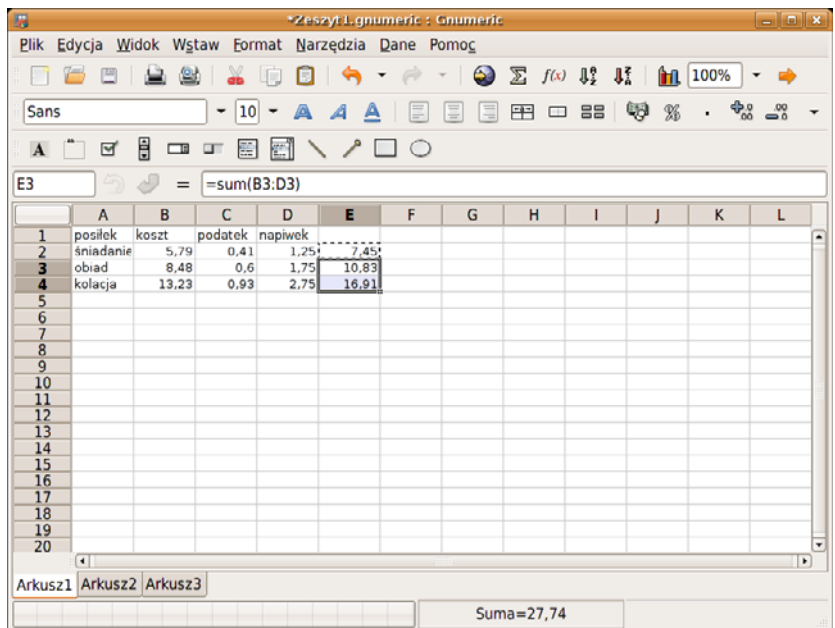
Na końcu należy wybrać z menu *Edycja/Wklej*. Funkcja została wklejona, a program automatycznie obliczy wartości dla każdego wiersza, co widać na rysunku 11.8.

Rysunek 11.7.

Zaznaczanie zakresu komórek do wprowadzenia funkcji

**Rysunek 11.8.**

Uaktualnione obliczenia w programie obejmujące dodatkowe wiersze



Świetnie, wszystko idzie zgodnie z planem. Pozostała już tylko jedna rzecz do obliczenia: zsumowanie całkowitych kosztów posiłków. Gdy znamy już omówioną wcześniej metodę, jest to całkiem proste do wykonania, należy kolejno:

1. wybrać komórkę, w której zostanie wykonane obliczenie,
2. wpisać `=sum(`, aby rozpocząć definiowanie funkcji,

3. wcisnąć lewy przycisk myszy i, trzymając wciśnięty, zaznaczyć trzy komórki, z których wartości mają być zsumowane,
4. wprowadzić nawias zamykający do funkcji i wcisnąć *Enter*, zostaną wówczas wykonane obliczenia.

Jeżeli wszystko pójdzie zgodnie z planem, arkusz kalkulacyjny powinien wyglądać teraz tak, jak na rysunku 11.9. Czytelnik ma za sobą podstawowy kurs obsługi arkusza kalkulacyjnego.

Rysunek 11.9.

Sumowanie
obliczonych
poprzednio wartości

The screenshot shows a spreadsheet window titled '*Zeszyt1.gnumeric : gnumeric'. The menu bar includes Plik, Edycja, Widok, Wstaw, Format, Narzędzia, Dane, and Pomoc. The toolbar contains various icons for file operations, formatting, and data manipulation. The formula bar shows 'E7 ='. The spreadsheet grid has columns A through L and rows 1 through 20. The data is as follows:

	A	B	C	D	E	F	G	H	I	J	K	L
1	posilek	koszt	podatek	napiwek								
2	śniadanie	5,79	0,41	1,25	7,45							
3	obiad	8,48	0,6	1,75	10,83							
4	kolacja	13,23	0,93	2,75	16,91							
5												
6					35,19							
7												
8												
9												
10												
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												

At the bottom of the window, there are tabs for 'Arkusz1', 'Arkusz2', and 'Arkusz3'. A status bar at the bottom right shows 'Suma=0'.

Oprócz faktu, że w powyższym przykładzie nie trzeba było wykonywać samodzielnie żadnych obliczeń matematycznych (hura!), kolejną wygodną sprawą jest to, że zmiana wartości w którejkolwiek z komórek ujętych w zakresie dla danej formuły spowoduje zaktualizowanie wyniku w komórce, która oblicza wynik z danego zakresu. Warto wypróbować to samodzielnie; jak każde dobre oprogramowanie, powinno po prostu działać.

Kiedy już podstawowe obliczenia są wykonane, można uzupełnić dane o opisy nowych pól, aby objaśnić ich znaczenie. Opisy są już wprowadzone w nagłówkach wierszy i kolumn, ale nie ma objaśnienia zawartości dla podsumowania dnia. Aby wprowadzić tekst do komórki arkusza kalkulacyjnego, należy zaznaczyć wybraną komórkę i po prostu wpisać odpowiedni tekst. Na rysunku 11.10 widać arkusz z opisem *Podsumowanie dnia* wpisanym do komórki D6.

Ponieważ opis *Podsumowanie dnia* nie mieści się całkowicie w komórce, należy usunąć tę drobną usterkę. W tym celu trzeba najpierw zaznaczyć całą kolumnę (w tym przypadku D), klikając jej nagłówek, następnie umieścić kursor myszy nad linią znajdującą się pomiędzy kolumnami D oraz E, wcisnąć i przytrzymać lewy przycisk myszy, i przesunąć linię tak, aby opis pola zmieścił się w komórce. Podczas przesuwania linii będzie wyświetlana informacja z aktualnym wymiarem kolumny (patrz rysunek 11.11).

Rysunek 11.10.

Dodawanie opisu do pola w arkuszu kalkulacyjnym

	A	B	C	D	E	F	G	H	I	J	K	L
1	posilek	koszt	podatek	napiwek								
2	śniadanie	5,79	0,41	1,25	7,45							
3	obiad	8,48	0,6	1,75	10,83							
4	kolacja	13,23	0,93	2,75	16,91							
5												
6				Podsumo	35,19							
7												
8												
9												
10												
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												

Rysunek 11.11.

Zmiana szerokości kolumny w arkuszu kalkulacyjnym

	A	B	C	D	E	F	G	H	I	J
1	posilek	koszt	podatek	napiwek						
2	śniadanie	5,79	0,41	1,25	7,45					
3	obiad	8,48	0,6	1,75	10,83					
4	kolacja	13,23	0,93	2,75	16,91					
5										
6				Podsumo	35,19					
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										

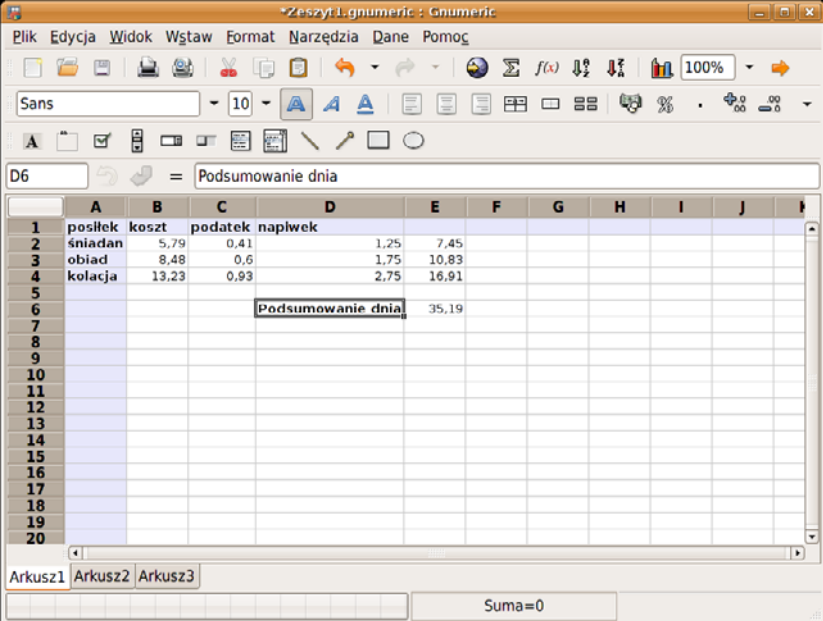
Po zmianie rozmiaru kolumny tak, aby opis pola był całkowicie widoczny, należy zwolnić przycisk myszy, cała operacja zostanie zakończona, a kolumna będzie wyświetlana na ekranie już z nową szerokością (patrz rysunek 11.12).

Teraz nadeszła pora na ulepszenie formatowania. Ponieważ zwykle opisy kolumn i wierszy różnią się nieco od reszty tabeli, można pogrubić w nich czcionkę. Łatwo to zrobić, zaznaczając odpowiednie komórki i pogrubiając czcionkę jednocześnie we wszystkich. Aby

Następnie należy użyć przycisku pogrubiania czcionki znajdującego się na pasku narzędzi. Zawartość wszystkich zaznaczonych komórek zostanie sformatowana pogrubioną czcionką, a efekt końcowy powinien wyglądać tak, jak na rysunku 11.14.

Rysunek 11.14.

Formatowanie
wybranych komórek
w arkuszu
kalkulacyjnym



The screenshot shows the Gnumeric application window titled '*Zeszyt1.gnumeric : Gnumeric'. The menu bar includes 'Plik', 'Edycja', 'Widok', 'Wstaw', 'Format', 'Narzędzia', 'Dane', and 'Pomoc'. The toolbar shows various icons for file operations, editing, and formatting. The spreadsheet has a formula bar showing 'D6 = Podsumowanie dnia'. The table data is as follows:

	A	B	C	D	E	F	G	H	I	J
1	posiłek	koszt	podatek	napilek						
2	śniadanie	5,79	0,41	1,25	7,45					
3	obiad	8,48	0,6	1,75	10,83					
4	kolacja	13,23	0,93	2,75	16,91					
5										
6				Podsumowanie dnia	35,19					
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										

The status bar at the bottom shows 'Arkusz1', 'Arkusz2', and 'Arkusz3' tabs, and a 'Suma=0' indicator.

Ostatnią czynnością, którą należy wykonać po zakończeniu pracy, jest zapisanie pliku, co można zrobić, wybierając z menu opcję *Plik/Zapisz*. Ponieważ arkusz, do którego były wprowadzane dane, posiadał dotychczas tylko tymczasową nazwę, podczas zapisywania zostanie wyświetlone pole, w którym będzie można wprowadzić ostateczną nazwę dla pliku (oraz w razie potrzeby wskazać lokalizację, w której ma zostać zapisany).

Oczywiście, za pomocą dzisiejszych arkuszy kalkulacyjnych można wykonać o wiele więcej, niż opisano w powyższym przykładzie, jednak na początek taki zestaw informacji powinien wystarczyć. Wszystkie opisane w niniejszym rozdziale arkusze kalkulacyjne posiadają doskonale opracowaną pomoc, dostępnych jest też wiele książek omawiających je szczegółowo. Choć na razie nie napisano wielu książek poświęconych dokładnie arkuszom kalkulacyjnym działającym w systemie Linux, to zarówno Gnumeric, jak i Calc zostały zaprojektowane tak, by były kompatybilne z innymi programami tego rodzaju, takimi jak np. Microsoft Excel, dzięki czemu informacje na temat korzystania z takich programów można czerpać niemal z każdej książki poświęconej arkuszom kalkulacyjnym.

Korzystanie z GnumERICA

Jak wspominałem we wstępie do niniejszego rozdziału, arkusze kalkulacyjne są obecnie równie przydatne w pracy zawodowej, jak i w domu. Wystarczy wspomnieć o rocznym rozliczeniu podatkowym, które można znacznie sobie ułatwić, korzystając właśnie z możliwości oferowanych przez tego rodzaju programy.

Gnumeric, część pakietu GNOME Office (omówionego w rozdziale 10.), jest jednym z tych arkuszy kalkulacyjnych dostępnych w Linuksie, które podobnie jak wino, z wiekiem stają się coraz lepsze.

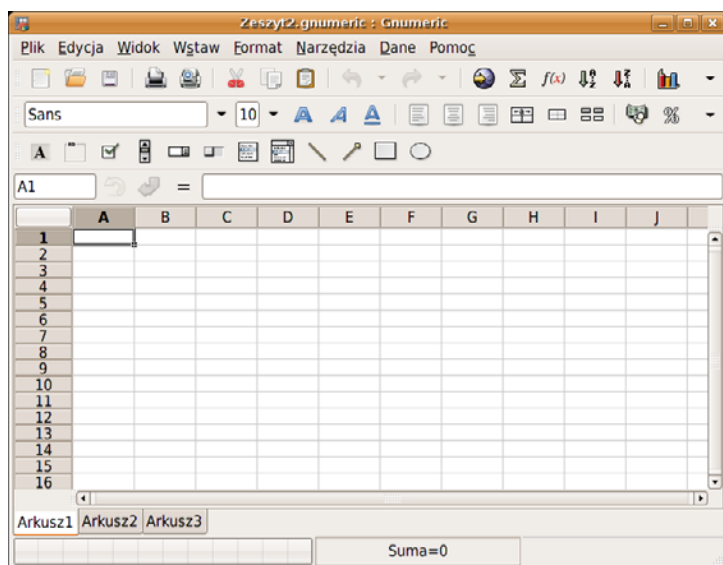


W poprzednich wersjach Ubuntu Gnumeric był instalowany domyślnie w systemie, teraz jego miejsce zajął pakiet OpenOffice.org. Aby go zainstalować, wystarczy użyć wybranego menedżera pakietów (apt-get, aptitude czy Synaptica) omówionego w rozdziale 20. Jeżeli użytkownik woli korzystać z arkusza zainstalowanego domyślnie w systemie, wówczas może pominąć tę część rozdziału i przejść wprost do akapitów poświęconych programowi OpenOffice.org Calc, znajdujących się poniżej.

Rozpoczęcie pracy z programem Gnumeric

Po zainstalowaniu programu w systemie, w menu *Aplikacje/Biuro* pojawi się odpowiednia pozycja (*Arkusz GnumERICA*), dzięki której będzie można uruchomić program. Po jego włączeniu zostanie wyświetlone okno z pustym dokumentem widoczne na rysunku 11.15.

Rysunek 11.15.
Uruchomiony program
Gnumeric



Krótki przewodnik użytkowania programu Gnumeric

Ponieważ program posiada doskonale zorganizowaną pomoc dostępną w systemie, nie będę zanudzać czytelnika opisem zawartości każdego menu. Zapewne przydatne jednak będzie krótkie omówienie interfejsu programu (patrz rysunek 11.15).

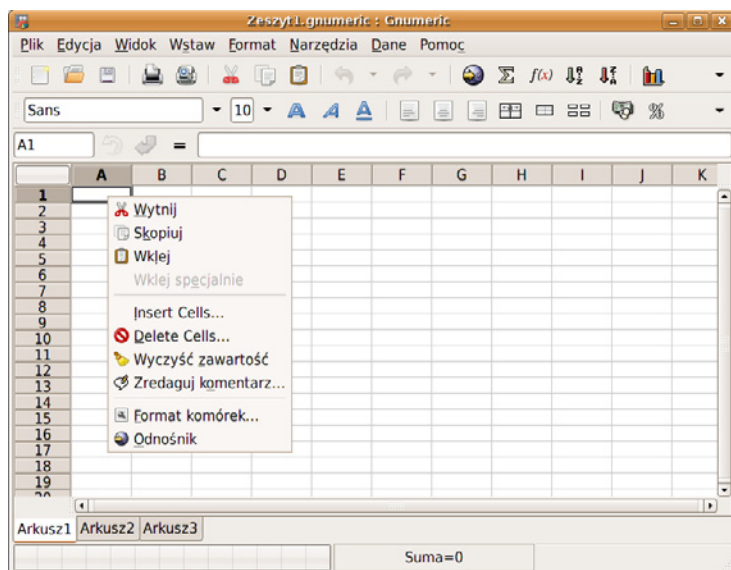
- ♦ **Menu:** rozwijane menu zorganizowane są podobnie jak w programie Microsoft Excel, dzięki czemu użytkownik nie powinien mieć kłopotów z przestawieniem się na nowy program. Pozycje menu z strzałkami umieszczonymi po prawej stronie umożliwiają dostęp do kolejnych podmenu.
- ♦ **Paski narzędzi:** domyślnie program wyświetla dwa paski narzędziowe: *Standard* i *Format*, które umożliwiają korzystanie z różnych narzędzi za pomocą ikon

znajdujących się na tych paskach. Zależnie od wielkości okna, w którym wyświetlany jest program, część narzędzi może być dostępna dopiero po kliknięciu w strzałki umieszczonej przy prawej krawędzi pasków narzędzi, a skierowanej w dół. Paski narzędzi można włączać i wyłączać, korzystając z menu *Widok/Toolbars*.

- ♦ **Nagłówki kolumn i wierszy:** te pola umożliwiają zidentyfikowanie aktualnie podświetlonej komórki. Jest to szczególnie przydatne podczas wpisywania zakresu komórek do formuły.
- ♦ **Menu kontekstowe:** po podświetleniu dowolnej komórki wystarczy wcisnąć prawy przycisk myszy, zostanie wówczas wyświetlone menu kontekstowe widoczne na rysunku 11.16. Umożliwia ono łatwy dostęp do kilku podstawowych operacji związanych z komórkami.

Rysunek 11.16.

Menu kontekstowe
w programie
Gnumeric



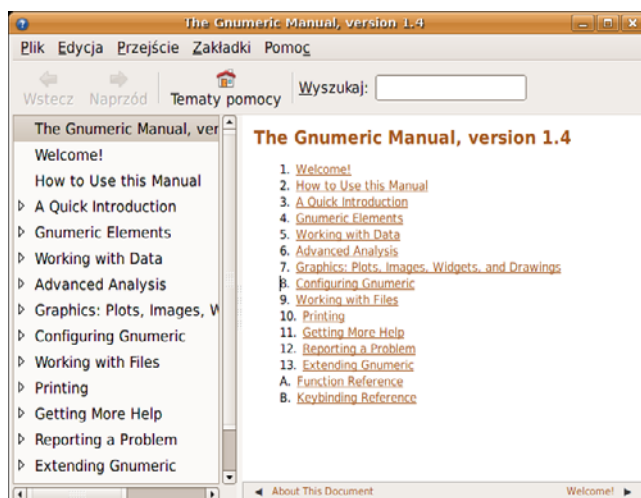
Program Gnumeric jest też wyposażony w obszerny system pomocy, wystarczy wybrać z menu *Pomoc/Zawartość* lub wcisnąć klawisz *F1*. Na rysunku 11.17 widać zawartość okna z pomocą programu Gnumeric. Można w nim znaleźć wyszukiwarkę słów kluczowych, listę tematów oraz szczegółową zawartość wybranego zagadnienia (po prawej stronie okna).

Z menu *Pomoc* można również otworzyć stronę internetową z pomocą dla Gnumerica, na której łatwo znaleźć dodatkowe informacje na temat programu, a także różnorodne materiały ułatwiające pracę z tą aplikacją (patrz rysunek 11.18).

Funkcje w programie Gnumeric

We wstępnym omówieniu działania arkuszy kalkulacyjnych funkcję wprowadzono ręcznie, co w przypadku prostych obliczeń oraz znania nazwy danej funkcji jest wystarczające. Jednak przy bardziej skomplikowanych zagadnieniach warto skorzystać z obszernego zbioru funkcji dostępnego w programie. Wystarczy wybrać z menu *Wstaw/Funkcja*,

Rysunek 11.17.
System pomocy
programu Gnumeric



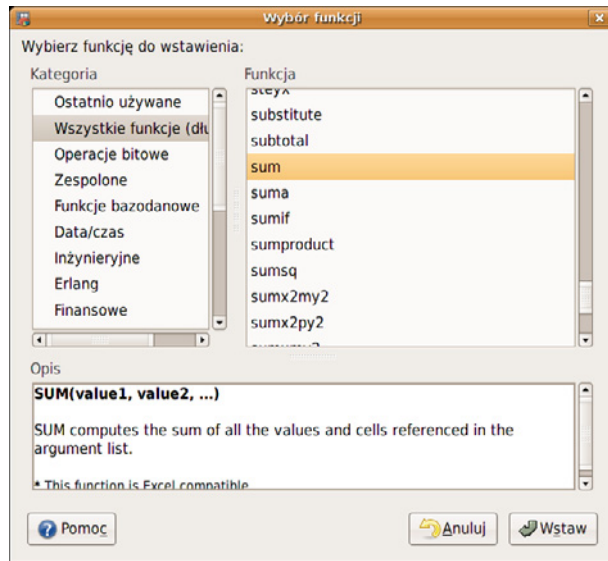
Rysunek 11.18. Strona domowa programu Gnumeric w sieci

a następnie za pomocą myszy wskazać odpowiednią opcję i wprowadzić ją do arkusza. Na rysunku 11.19 widać funkcję `sum`, którą można wybrać z odpowiedniego okna, w jakim są wyświetlane również pozostałe dostępne możliwości.

Jak widać, wybranie dowolnej funkcji spowoduje od razu wyświetlenie w dolnym panelu okna informacji na temat wybranej pozycji.

Rysunek 11.19.

*Funkcje dostępne
w programie
Gnumeric*



Określanie rodzaju danych wprowadzanych do programu Gnumeric

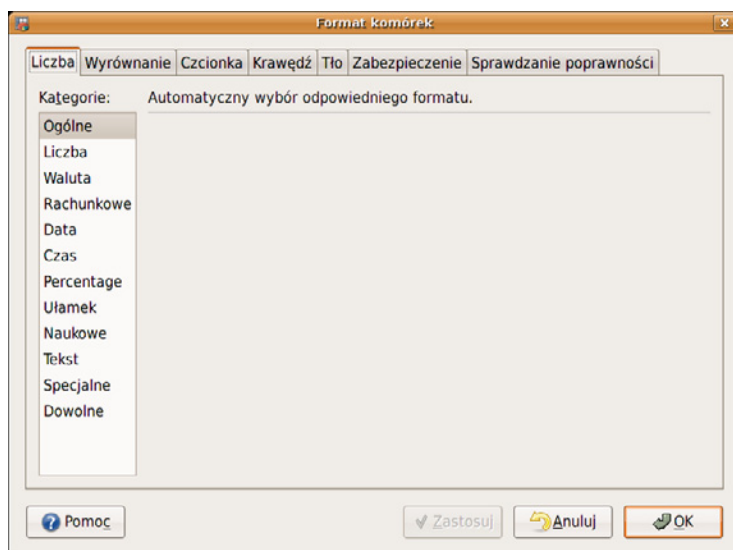
Dane wprowadzone do arkusza przedstawionego w omówieniu arkuszy kalkulacyjnych znajdującym się na początku rozdziału były łatwe do rozpoznania przez program; sekwencje znaków alfabetu i spacji to prawdopodobnie słowa, natomiast cyfry i przecinki oznaczają zapewne dane liczbowe. Zidentyfikowanie rodzaju danych (znanych też jako **rodzaj danych** lub **format danych**) wprowadzanych do komórek jest bardzo ważny zwłaszcza wtedy, kiedy ma zostać użyta odpowiednia funkcja wykonująca obliczenia z wykorzystaniem zawartości określonych komórek.

Jeżeli podczas wprowadzania danych do arkusza kalkulacyjnego, takiego jak Gnumeric, użytkownik nie zdefiniuje ich rodzaju, wówczas program przypisze im ogólny format, określony na podstawie znaków, jakie się w nich znajdują. Oczywiście, użytkownik może samodzielnie określić format danych wprowadzanych do komórki (czy komórek), wystarczy ją zaznaczyć, a następnie z menu wybrać *Format/Komórki*, co spowoduje wyświetlenie okna dialogowego widocznego na rysunku 11.20.

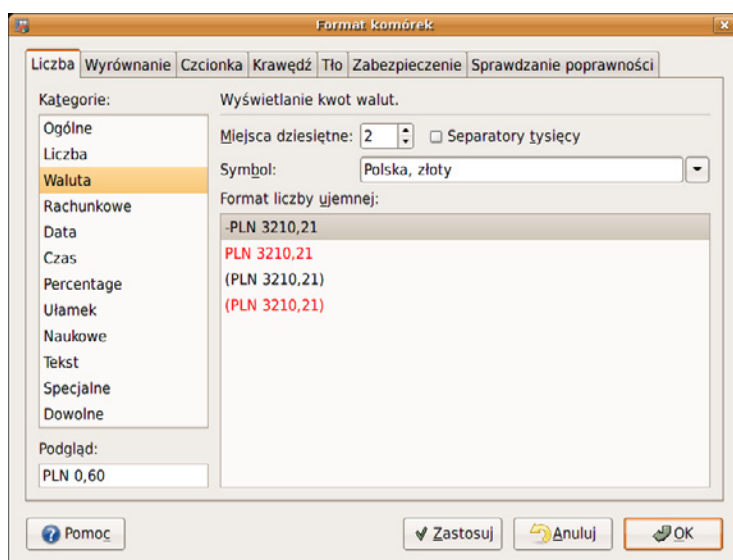
Jako przykład można potraktować komórkę C3 w przykładowym arkuszu (widać ją na rysunku 11.14), w której dane nie są wyświetlane jako kwoty pieniężne z dwoma cyframi po przecinku. Stało się tak dlatego, że choć program rozpoznał wartość ułamkową, to nie miał żadnych danych aby potraktować tę liczbę jako wartość pieniężną. Aby to poprawić, wystarczy zaznaczyć tę komórkę, z menu wybrać *Format/Komórki*, a następnie z panelu wyświetlonego okna wybrać pozycję *Waluta*. W panelu po prawej stronie okna zostaną wyświetlone kolejne pozycje z możliwymi do zastosowania formatami (patrz rysunek 11.21).

Dla tego rodzaju danych standardowym sposobem wyświetlania są dwie cyfry po przecinku, czyli dokładnie to, co należało poprawić. Teraz wystarczy już tylko użyć przycisku

Rysunek 11.20.
Określanie formatu
danych w programie
Gnumeric



Rysunek 11.21.
Określanie formatu
danych dla wybranej
komórki



OK, a wybrany format zostanie zastosowany dla komórki C3. Efekt wprowadzonych zmian można zobaczyć na rysunku 11.22.

Wszystko byłoby dobrze, ponieważ program rozpoznał, że dane wprowadzone do komórki C3 mają charakter walutowy, tyle tylko, że dane w pozostałych komórkach nadal są traktowane jak dane ogólne, co powoduje, że cały arkusz wygląda nieco dziwnie. Można to poprawić, zaznaczając wszystkie komórki z danymi liczbowymi w arkuszu (wystarczy wcisnąć i przytrzymać klawisz *Ctrl*, następnie zaznaczyć komórkę B2, rozciągnąć zaznaczenie do komórki E4, po czym zaznaczyć jeszcze komórkę E6 i dopiero wówczas zwolnić klawisz *Ctrl*), wyświetlić ponownie okno dialogowe *Format komórek*, wybrać pozycję *Waluta* i wcisnąć przycisk OK. Voila! Arkusz powinien teraz wyglądać tak, jak ten na rysunku 11.23.

Rysunek 11.22.

Arkusz
po wprowadzeniu
formatu danych
Waluta dla komórki C3

The screenshot shows a spreadsheet window titled 'jedzenie.xls : Gnumeric'. The menu bar includes 'Plik', 'Edycja', 'Widok', 'Wstaw', 'Format', 'Narzędzia', 'Dane', and 'Pomoc'. The toolbar contains various icons for file operations, editing, and formatting. The formula bar shows 'C3' and the value '0,6'. The spreadsheet grid has columns A through J and rows 1 through 20. The data is as follows:

	A	B	C	D	E	F	G	H	I	J
1	posiłek	koszt	podatek	napiwek						
2	śniadani	5,79	0,41	1,25	7,45					
3	obiad	8,48	PLN 0,60	1,75	10,83					
4	kolacja	13,23	0,93	2,75	16,91					
5										
6				Podsumowanie dnia	35,19					
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										

The status bar at the bottom shows 'Suma=0,6'.

Rysunek 11.23.

Arkusz
po formatowaniu
wprowadzonych
danych jako
walutowych

The screenshot shows the same spreadsheet window. The data is now formatted with currency symbols (PLN). The formula bar shows 'E6' and the formula '=sum(E2:E4)'. The spreadsheet grid is as follows:

	A	B	C	D	E	F	G	H	I	J
1	posiłek	koszt	podatek	napiwek						
2	śniadani	PLN 5,79	PLN 0,41	PLN 1,25	PLN 7,45					
3	obiad	PLN 8,48	PLN 0,60	PLN 1,75	PLN 10,83					
4	kolacja	PLN 13,23	PLN 0,93	PLN 2,75	PLN 16,91					
5										
6				Podsumowanie dnia	PLN 35,19					
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										

The status bar at the bottom shows 'Suma=105,57'.

Importowanie istniejących arkuszy

Użytkownicy Linuksa, tacy jak ja, mają tendencję do bycia linuksocentrykami, co jednak nie oznacza, że całkowicie ignorują rzeczywistość. Wiem, że mogę napotkać na arkusze kalkulacyjne (czy też — żeby być całkowicie szczerym — inne dokumenty pakietu

biurowego) utworzone w innym, bardziej rozpowszechnionym programie, takim jak np. Microsoft Excel, który z kolei działał w innym systemie operacyjnym, np. w Mac OS X czy Microsoft Windows. Programiści tworzący Gnumeric nie mogą ignorować tych faktów, dlatego wyposażyli swój program w szereg filtrów umożliwiających otwieranie arkuszy i skoroszytów utworzonych w Excelu.



Gdy użytkownik posiada pokaźny zbiór arkuszy kalkulacyjnych utworzonych w Excelu i zastanawia się, jak przenieść je do Ubuntu, powinien przejrzeć rozdział 16., „Przesyłanie i udostępnianie plików w Ubuntu”, w którym zamieszczono wskazówki, jak poradzić sobie z tym zadaniem. Nie ma powodu do obaw, naprawdę łatwo skonfigurować połączenie sieciowe czy wykorzystać bardziej tradycyjne narzędzia, takie jak dyskietka czy płyta CD.

I tak, aby otworzyć arkusz utworzony w programie Microsoft Excel, wystarczy wybrać z menu polecenie *Plik/Otwórz* i wskazać odpowiedni plik. Okno dialogowe, które zostanie wyświetlone, umożliwi otwarcie pliku z dowolnym rozszerzeniem, również *.xls*, które jest używane przez program Microsoft Excel.

Ten sam mechanizm jest używany do otwierania plików w innych formatach, których obsługa została zaimplementowana w programie Gnumeric. Oto formaty plików, z którymi na pewno poradzi sobie Gnumeric.

- ♦ **Applix (.as)** — pliki z tym rozszerzeniem są tworzone w pakiecie Applix Office.
- ♦ **pliki .csv** — standardowy, oparty na tekście format wymiany danych, w którym każdy wiersz wyświetlany jest w osobnej linii, a każda pozycja oddzielona jest od następnej przecinkiem. W formacie tym nie można zachować formuł, a jedynie wartości poszczególnych komórek.
- ♦ **Data Interchange Format (.dif)** — kolejny standardowy format wymiany danych oparty na tekście, również nie umożliwia przechowywania formuł, a jedynie wartości umieszczone w komórkach.
- ♦ **Gnumeric (.gnumeric)** — domyślny format zapisu danych programu Gnumeric (bez żadnych szczególnych niespodzianek).
- ♦ **Linear and Integer Program Format (.mps)** — tekstowy format wymiany danych wynaleziony przez IBM w XVI wieku. Nie umożliwia on zapisywania danych potrzebnych do obliczania zawartości komórek, a jedynie same wartości tychże komórek. Obliczenia pozostawiono mnichom, którzy zajmowali się iluminowaniem perforowanych kart.
- ♦ **Lotus 1-2-3 (.123, .wkl, .wks)** — format używany w różnych wersjach arkusza kalkulacyjnego Lotus 1-2-3.
- ♦ **Multiplan (.sylk)** — format używany w starym programie Multiplan.
- ♦ **OpenOffice.org Calc (.ods)** — otwarty format arkusza kalkulacyjnego używany w programie OpenOffice.org Calc i innych ukierunkowanych na przyszłość aplikacjach.
- ♦ **Quattro Pro (.wb1, .wb2, .wb3)** — format używany w różnych wersjach arkusza kalkulacyjnego Quattro Pro firmy Borland.

- ♦ **Xbase (.dbf)** — format bazy danych wykorzystywany w różnych aplikacjach kompatybilnych z programem Dbase III, takich jak Microsoft FoxPro, Dbase III i kolejne, itp.

Jak widać, programiści wolnego oprogramowania zajmujący się programem Gnumeric włożyli znaczący wysiłek w to, aby użytkownicy mogli importować każdy istniejący format arkusza danych do ich programu; stracić można tylko zależność od innych aplikacji.

Korzystanie z programu OpenOffice.org Calc

W poprzedniej części omówiłem program Gnumeric, potężny i mający dobrą reputację arkusz kalkulacyjny, będący częścią pakietu GNOME Office, który z kolei można znaleźć na większości komputerów działających pod kontrolą systemu Linux/GNOME. Choć ta aplikacja ma ogromne możliwości, zwiększane dodatkowo zdolnością importowania wielu różnych formatów plików, to jednak jest to pojedynczy program. Pakiet GNOME Office to zestaw kilku aplikacji wykorzystujących bibliotekę GIMP Toolkit (GTK) i jej graficzne kontrolki. Powstanie tego pakietu jest związane raczej z korzystaniem z tej samej biblioteki oraz chęci posiadania w GNOME odpowiednika pakietu biurowego dostępnego w KDE, niż z wewnętrznymi związkami między aplikacjami wchodzącymi w jego skład.

I tak oto dotarliśmy do programu OpenOffice.org Calc. Jak sugeruje nazwa, Calc jest częścią pakietu OpenOffice.org, czyli zestawu aplikacji zaprojektowanych wspólnie, dzielących ten sam kod źródłowy, podobnie wyglądających i działających, i uzupełniających wzajemnie swoje możliwości. Oto przykład: do programu Calc można za pomocą kreatora konwersji importować arkusze utworzone w innych programach, po zakończeniu tego procesu program utworzy plik zawierający raport; plik ten powstanie w formacie aplikacji OpenOffice.org Writer, dzięki czemu można od razu zapoznać się z jego zawartością. Można go obejrzeć nawet wtedy, gdy Writer nie został zainstalowany w systemie.

Zgodnie z tym, o czym przekonały się duże firmy, takie jak Microsoft, jest jeszcze wiele do powiedzenia na temat spójności powiązanych aplikacji, tak aby ich możliwości wzajemnie się uzupełniały, dzięki czemu można wykonywać skomplikowane operacje na danych, których to wymaga się we współczesnych biurach, zorientowanych na korzystanie z komputerów. Odmiennie od pakietu Microsoft Office, OpenOffice.org to przeznaczony do biura zestaw wydajnego oprogramowania, który nic nie kosztuje. Chociaż sam używam Gnumera od lat ze względu na jego niewielkie rozmiary i wymagania mniejsze od OpenOffice.org, to jednak coraz częściej korzystam z tego drugiego. Gnumeric wciąż ma zastosowanie na starszym komputerze przenośnym, który z kolei jest poręczniejszy podczas pobytu w hotelowych pokojach.

Instalowanie plików dla programu OpenOffice.org Calc

Pakiet OpenOffice.org wraz z programem Calc znajduje się w systemie, jeżeli ten był instalowany z płyt Live CD Desktop i Alternate Install CD (zarówno w wersji tekstowej, jak i OEM). Jeżeli użytkownik chciałby dodać pakiet OpenOffice.org wraz z arkuszem kalkulacyjnym Calc do Ubuntu w wersji serwerowej z zainstalowanym środowiskiem graficznym, powinien zainstalować pakiety, używając jednego z dostępnych menedżerów

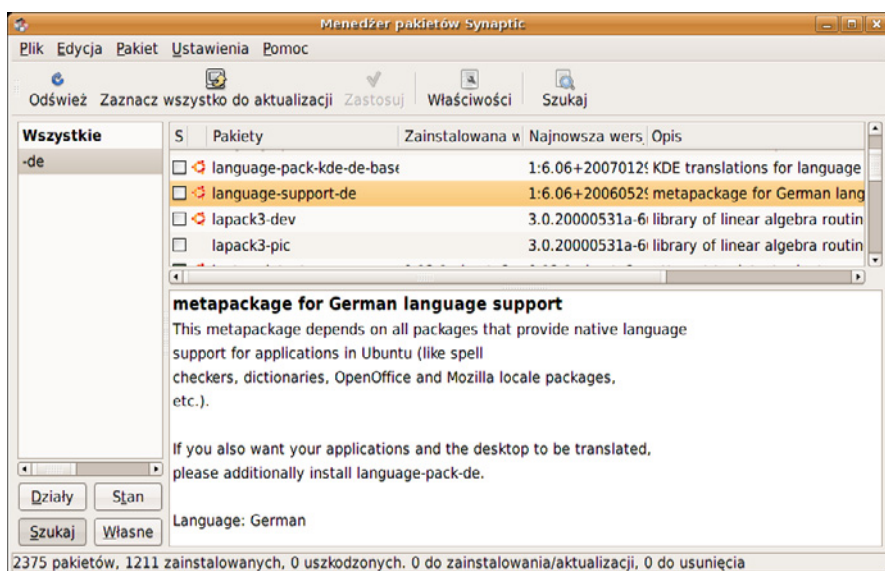
plików apt-get, aptitude czy Synaptica (zgodnie z informacjami zamieszczonymi w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”).

Jedną z cech odróżniających Ubuntu od innych dystrybucji Linuksa jest wyraźne ukięrowanie na użytkownika, zwłaszcza w zakresie internacjonalizacji. Jeżeli Calc był instalowany później niż sam system, może się okazać, że trzeba będzie również zainstalować odpowiednie pliki lokalizacyjne i językowe, dzięki którym program będzie dostępny w odpowiednim języku. Jest to szczególnie ważne, jeżeli w programie otwierane będą pliki utworzone w innych niż polska wersjach językowych programu.



Jeżeli zostały już zainstalowane pakiety językowe czy lokalizacyjne dla jakiegokolwiek innego programu wchodzącego w skład pakietu OpenOffice.org, wówczas nie ma potrzeby ponownego ich instalowania. Wystarczy raz umieścić je w systemie, a będą dostępne dla każdego komponentu pakietu.

Pliki i pakiety lokalizacyjne można łatwo odnaleźć, używając do wyszukiwania ciągu określającego dany kraj (kod dla Polski to PL). Na rysunku 11.24 przedstawiam program Synaptic, wyświetlający pliki lokalizacyjne dla pakietu OpenOffice.org dla niemieckiej wersji językowej, identyfikowanej za pomocą ciągu -de.



Rysunek 11.24. Pakiety językowe dla OpenOffice.org



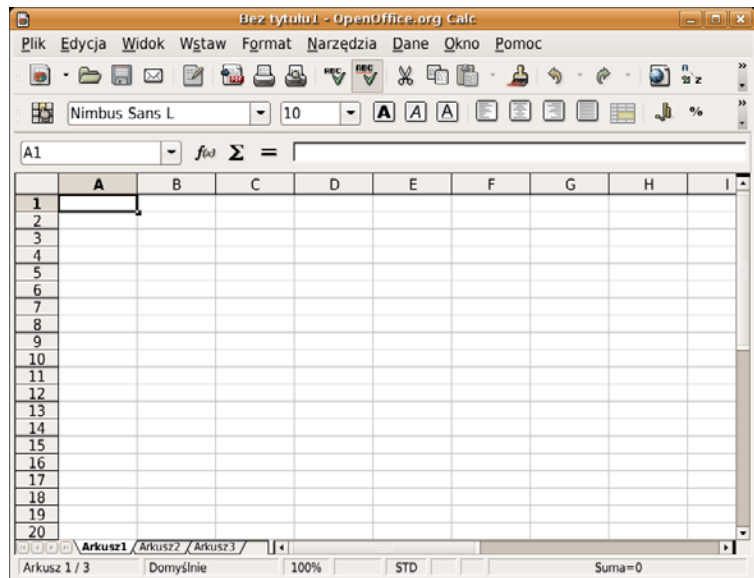
Użytkownik może zainstalować kompletne wsparcie językowe dla wszystkich aplikacji (bez zmiany wersji językowej samego interfejsu GNOME), wystarczy dodać metapaket dla odpowiednich danych lokalizacyjnych (tzw. lokali, ang. *locales*). Nazwa takiego pliku wygląda następująco: language-support-de, przy czym na końcu znajduje się kod kraju, w pliku znajduje się też słownik dla danego języka. Aby w pełni przetłumaczyć interfejs i menu aplikacji, należy zainstalować odpowiedni pakiet językowy. Nazwy pakietów językowych składają się z kodu języka i kodu kraju. Po zainstalowaniu odpowiednich plików językowych można dowolnie zmieniać język dla danego dokumentu, wystarczy z menu wybrać *Narzędzia/Opcje/Ustawienia językowe/Języki* i na liście rozwijanej *Domyślny język dokumentów* wybrać odpowiednią pozycję.

W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, zamieszczam więcej informacji na temat instalowania aplikacji za pomocą menedżera pakietów Synaptic.

Uruchamianie programu Calc

Program można uruchomić z linii poleceń przy użyciu komendy `oocalc`, ale łatwiej będzie użyć menu GNOME. Odpowiednią pozycję można znaleźć w menu *Aplikacje/Biuro*. Po wybraniu pozycji OpenOffice.org Calc zostanie wyświetlone okno programu (patrz rysunek 11.25).

Rysunek 11.25.
Program Calc



Krótki podręcznik użytkowania programu Calc

OpenOffice.org Calc wyposażony jest w doskonały system pomocy, podobnie zresztą jak cały pakiet OpenOffice.org, dlatego nie będę zanudzać czytelników opisem zawartości każdego menu. Zamiast tego proponuję krótki przewodnik po interfejsie tego programu (patrz rysunki 11.25 i 11.26).

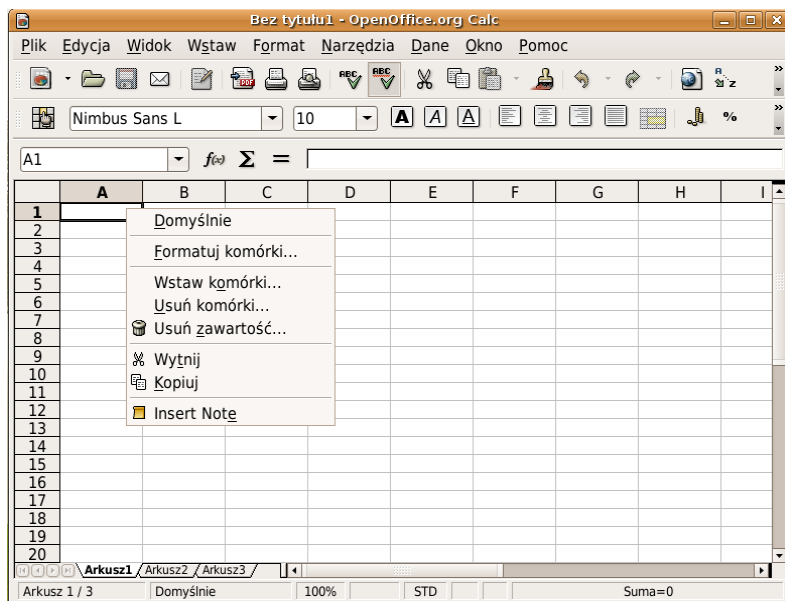
- ♦ **Menu:** rozwijane menu zorganizowane są podobnie jak w Excelu, dzięki czemu użytkownik nie powinien mieć kłopotów z przestawieniem się na nowy program. Pozycje menu z strzałkami umieszczonymi po prawej stronie umożliwiają dostęp do kolejnych podmenu.
- ♦ **Paski narzędzi:** domyślnie program wyświetla paski narzędziowe, *Standardowy* i *Formatowanie*, które umożliwiają korzystanie z różnych narzędzi za pomocą ikon znajdujących się na nich, a także *Pasek formuły*, w którym można sprawdzić i edytować funkcje i formuły zamieszczone w danych komórkach. Użytkownik może modyfikować listę wyświetlanych na paskach ikon, wystarczy kliknąć kontrolkę znajdującą się przy prawej krawędzi paska narzędzia ze strzałką skierowaną w dół.

Można też włączać i wyłączać wyświetlanie różnych pasków narzędzi, wybierając ich nazwy z menu *Widok/Paski narzędzi*.

- ♦ **Nagłówki kolumn i wierszy:** te pola umożliwiają zidentyfikowanie aktualnie podświetlonej komórki. Jest to szczególnie przydatne podczas wpisywania zakresu komórek do formuły.
- ♦ **Menu kontekstowe:** po podświetleniu dowolnej komórki wystarczy wcisnąć prawy przycisk myszy, a zostanie wyświetlone menu kontekstowe widoczne na rysunku 11.26. Umożliwia ono łatwy dostęp do kilku podstawowych operacji związanych z komórkami.

Rysunek 11.26.

Menu kontekstowe programu Calc



Program Calc jest też wyposażony w obszerny system pomocy, wystarczy wybrać z menu *Pomoc/Pomoc Openoffice.org* lub wcisnąć klawisz *F1* w czasie pracy z programem. Na rysunku 11.27 widać zawartość okna z pomocą programu Calc. Można w nim znaleźć wyszukiwarkę słów kluczowych, listę tematów oraz szczegółową zawartość wybranego zagadnienia (po prawej stronie okna).



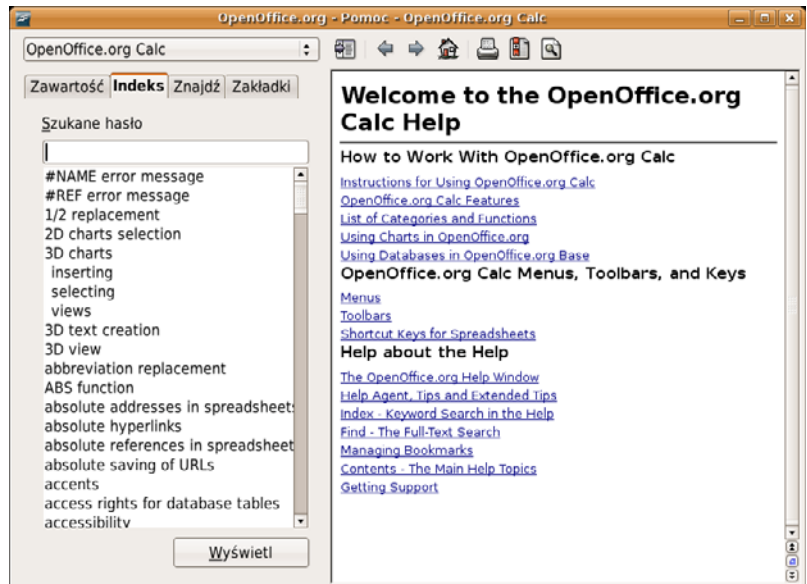
Jeżeli pojawi się jakiś problem z wyświetlaniem zawartości pomocy programu Calc, należy zainstalować program Writer. To wymusi zaktualizowanie komponentu odpowiedzialnego za wyświetlanie zawartości pomocy w pakiecie OpenOffice.org, co zwykle rozwiązuje problem.

Funkcje w programie Calc

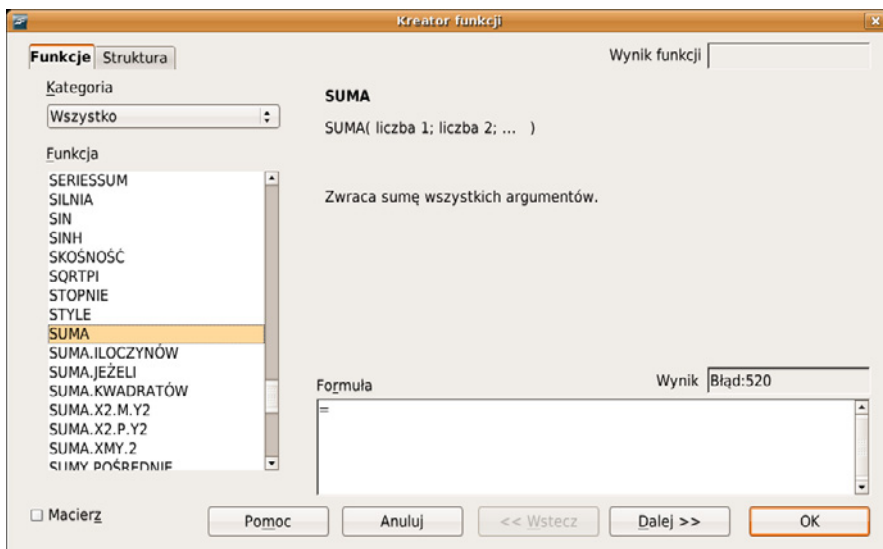
We wstępnym omówieniu działania arkuszy kalkulacyjnych funkcję wprowadzono ręcznie, co w przypadku prostych obliczeń oraz znajomości nazwy danej funkcji jest wystarczające. Jednak w czasie wykonywania bardziej skomplikowanych zagadnień warto skorzystać z obszernego zbioru funkcji dostępnego w programie. Wystarczy wybrać z menu *Wstaw/*

Rysunek 11.27.

Pomoc programu
Calc



Funkcja, a następnie za pomocą myszy wskazać odpowiednią opcję i wprowadzić ją do arkusza. Na rysunku 11.28 widać funkcję suma, którą można wybrać z odpowiedniego okna, w jakim są wyświetlane również pozostałe dostępne funkcje.



Rysunek 11.28. Funkcje dostępne w programie Calc

Jak widać, wybranie dowolnej funkcji spowoduje wyświetlenie w dolnym panelu okna informacji na temat wybranej pozycji.

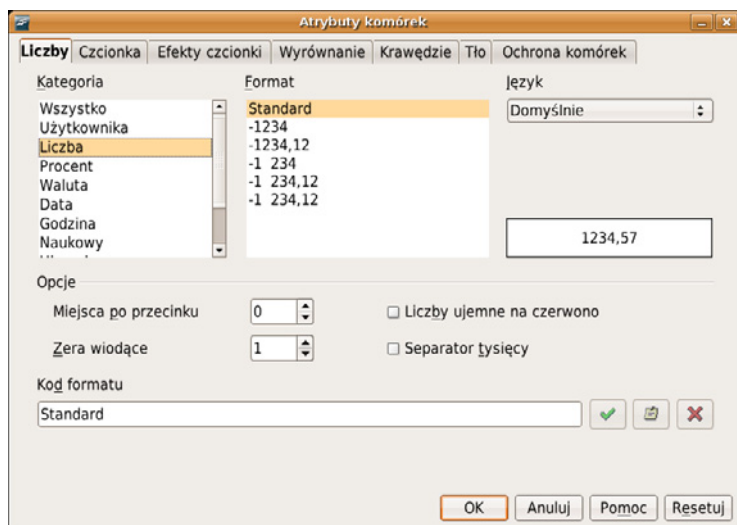
Określanie rodzaju danych wprowadzanych do programu Calc

Dane wprowadzone do arkusza przedstawionego w omówieniu arkuszy kalkulacyjnych znajdującego się na początku rozdziału były łatwe do rozpoznania przez program; sekwencje znaków alfabetu i spacji to prawdopodobnie słowa, natomiast cyfry i przecinki oznaczają zapewne dane liczbowe. Zidentyfikowanie rodzaju danych (znanych też jako **rodzaj danych** lub **format danych**) wprowadzanych do komórek jest bardzo ważny zwłaszcza wtedy, kiedy ma zostać użyta odpowiednia funkcja wykonująca obliczenia z wykorzystaniem zawartości określonych komórek.

Jeżeli podczas wprowadzania danych do arkusza kalkulacyjnego, takiego jak Calc, użytkownik nie zdefiniuje ich rodzaju, wówczas program przypisze im ogólny format, określony na podstawie znaków, jakie się w nich znajdują. Oczywiście, użytkownik może samodzielnie określić format danych wprowadzanych do komórki (czy komórek), wystarczy ją zaznaczyć, a następnie z menu wybrać *Format/Komórki*, co spowoduje wyświetlenie okna dialogowego widocznego na rysunku 11.29.

Rysunek 11.29.

Określanie formatu danych w programie Calc

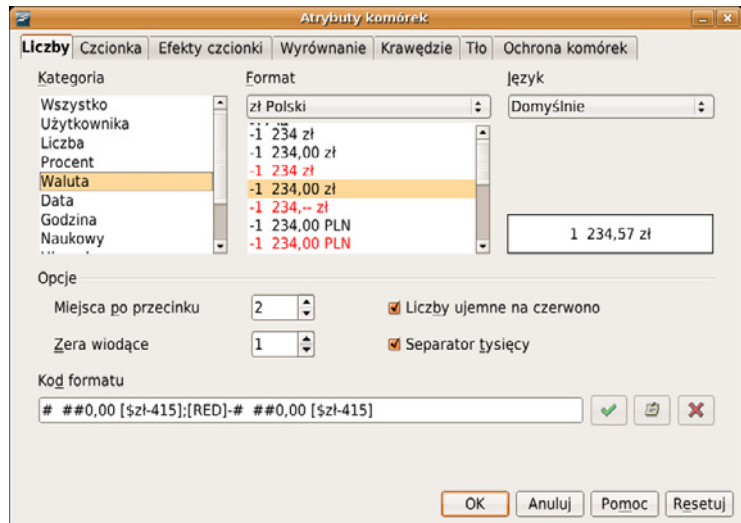


Jako przykład można potraktować komórkę C3 w przykładowym arkuszu (widać ją na rysunku 11.14), w której dane nie są wyświetlane jako kwoty pieniężne z dwoma cyframi po przecinku. Stało się tak dlatego, że choć program rozpoznał wartość ułamkową, to nie miał żadnych danych, by potraktować tę liczbę jako wartość pieniężną. Aby to poprawić, wystarczy zaznaczyć tę komórkę, z menu wybrać *Format/Komórki*, a następnie z panelu wyświetlonego okna wybrać pozycję *Waluta*. W panelu po prawej stronie okna zostaną pokazane kolejne pozycje z możliwymi do zastosowania formatami (patrz rysunek 11.30).

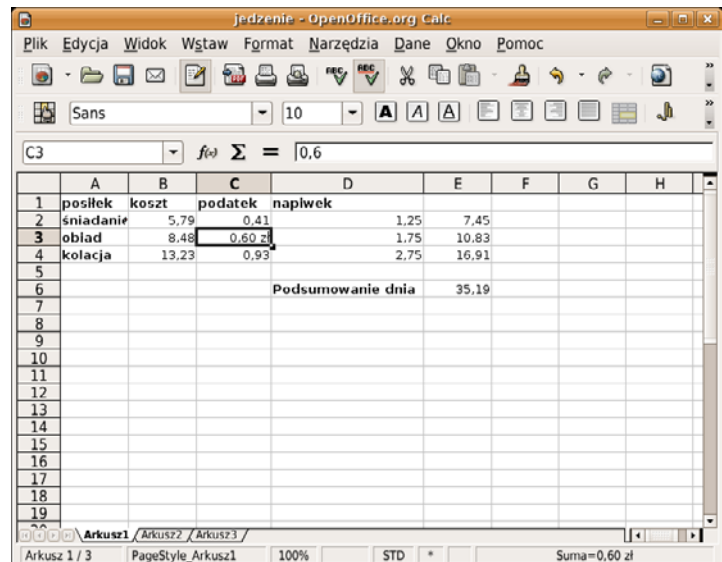
Dla tego rodzaju danych, standardowym sposobem wyświetlania są dwie cyfry po przecinku, czyli dokładnie tak, jak chcieliśmy. Teraz wystarczy już tylko użyć przycisku *OK*, a wybrany format zostanie zastosowany dla komórki C3. Efekt wprowadzonych zmian można zobaczyć na rysunku 11.31.

Rysunek 11.30.

Określanie formatu danych dla wybranej komórki

**Rysunek 11.31.**

Arkusz po formatowaniu wprowadzonych danych jako walutowych — w jednej komórce



Wszystko byłoby dobrze, ponieważ program rozpoznał, że dane wprowadzone do komórki C3 mają charakter walutowy, tyle tylko, że dane w pozostałych komórkach nadal są traktowane jak dane ogólne, co powoduje, że cały arkusz wygląda nieco dziwnie. Można to poprawić. Trzeba zaznaczyć wszystkie komórki z danymi liczbowymi w arkuszu (wystarczy wcisnąć i przytrzymać klawisz *Ctrl*, następnie zaznaczyć komórkę B2, rozciągnąć zaznaczenie do komórki E4, po czym zaznaczyć jeszcze komórkę E6 i dopiero wówczas zwolnić klawisz *Ctrl*), wyświetlić ponownie okno dialogowe *Atrybuty komórek*, wybrać pozycję *Waluta* i wcisnąć przycisk *OK*. Voila! Arkusz powinien teraz wyglądać tak, jak ten na rysunku 11.32.

Rysunek 11.32.

Arkusz
po formatowaniu
wprowadzonych
danych jako
walutowych
— we wszystkich
komórkach

The screenshot shows the OpenOffice Calc application window titled 'jedzenie - OpenOffice.org Calc'. The menu bar includes 'Plik', 'Edycja', 'Widok', 'Wstaw', 'Format', 'Narzędzia', 'Dane', 'Okno', and 'Pomoc'. The toolbar contains various icons for file operations and editing. The formula bar shows 'E6' and the formula '=SUMA(E2:E4)'. The spreadsheet has columns A through H and rows 1 through 19. The data is as follows:

	A	B	C	D	E	F	G	H
1	posiłek	koszt	podatek	napiwek				
2	śniadanie	5,79 zł	0,41 zł	1,25 zł	7,45 zł			
3	obiad	8,48 zł	0,60 zł	1,75 zł	10,83 zł			
4	kolacja	13,23 zł	0,93 zł	2,75 zł	16,91 zł			
5								
6				Podsumowanie dnia	35,19 zł			
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								
18								
19								

The status bar at the bottom shows 'Arkusz 1 / 3', 'PageStyle_Arkusz1', '100%', 'STD', and 'Suma=35,19 zł'.

Importowanie istniejących arkuszy

Użytkownicy Linuksa, tacy jak ja, mają tendencję do bycia linuksocentrykami, co jednak nie oznacza, że całkowicie ignorują rzeczywistość. Wiem, że mogę napotkać na arkusze kalkulacyjne (czy też — żeby być całkowicie szczerym — inne dokumenty pakietu biurowego) utworzone w innym, bardziej rozpowszechnionym programie, takim jak np. Microsoft Excel, który z kolei działał w innym systemie operacyjnym, np. w Mac OS X czy Microsoft Windows. Programiści tworzący Calc nie mogą ignorować tych faktów, dlatego wyposażyli swój program w szereg filtrów umożliwiających otwieranie arkuszy i skrosztów utworzonych w Excelu.



Gdy użytkownik posiada pokaźny zbiór arkuszy kalkulacyjnych utworzonych w Excelu i zastanawia się, jak przenieść je do Ubuntu, powinien przejrzeć rozdział 16., „Przesyłanie i udostępnianie plików w Ubuntu”, w którym zamieszczono wskazówki, jak poradzić sobie z tym zadaniem. Nie ma powodu do obaw, naprawdę łatwo skonfigurować połączenie sieciowe czy wykorzystać bardziej tradycyjne narzędzia, takie jak dyskietka czy płyta CD.

I tak, aby otworzyć arkusz utworzony w programie Microsoft Excel, wystarczy wybrać z menu polecenie *Plik/Otwórz* i wskazać odpowiedni plik. Okno dialogowe, które zostanie wyświetlone, umożliwi otwarcie pliku z dowolnym rozszerzeniem, również *.xls*, które jest używane przez program Microsoft Excel.

Obsługiwane formaty plików do importowania

Ten sam mechanizm jest używany do otwierania plików w innych formatach, których obsługa została zaimplementowana w programie Calc. Oto formaty plików, z którymi na pewno poradzi sobie Calc.

- ♦ **Tekst CSV (.csv)** — standardowy, oparty na tekście format wymiany danych, w którym każdy wiersz wyświetlany jest w osobnej linii, a każda pozycja oddzielona jest od następnej przecinkiem. W formacie tym nie można zachować formuł, a jedynie wartości poszczególnych komórek.
- ♦ **Data Interchange Format (.dif)** — kolejny standardowy format wymiany danych oparty na tekście, również nie umożliwia przechowywania formuł, a jedynie wartości umieszczone w komórkach.
- ♦ **Lotus 1-2-3 (.123, .wkl, .wks)** — format używany w różnych wersjach arkusza kalkulacyjnego Lotus 1-2-3.
- ♦ **Microsoft Excel (.xls, .xlt, .xlw, .xml)** — wszystkie formaty arkuszy kalkulacyjnych stosowane przez program Microsoft Excel.
- ♦ **Multipan (.sylk)** — format używany w starym programie Multipan.
- ♦ **OpenOffice.org Calc (.ods, .sdc)** — otwarty format arkusza kalkulacyjnego wykorzystywany w programie OpenOffice.org Calc i innych ukierunkowanych na przyszłość aplikacjach.
- ♦ **Quattro Pro (.wb1, .wb2, .wb3)** — format stosowany w różnych wersjach arkusza kalkulacyjnego Quattro Pro firmy Borland.
- ♦ **Pocket Excel (.pxl)** — format arkusza kalkulacyjnego używany w wersji Microsoft Excel dla urządzeń PDA.
- ♦ **Xbase (.dbf)** — format bazy danych wykorzystywany w różnych aplikacjach kompatybilnych z programem Dbase III, takich Microsoft FoxPro, Dbase III i kolejne, itp.

Jak widać, programiści wolnego oprogramowania zajmujący się programem Calc włożyli znaczący wysiłek w to, aby użytkownicy mogli importować każdy istniejący format arkusza danych do ich programu; stracić można tylko zależność od innych aplikacji.

Importowanie arkuszy za pomocą programu Konwerter dokumentów

Jak można było się przekonać w poprzedniej części, program Calc może importować arkusze kalkulacyjne utworzone w wielu różnych programach pochodzących od innych producentów. Może też bezpośrednio otwierać pliki tworzone w programie Microsoft Excel. Po zakończeniu ich edycji łatwo je zapisać w nowym, domyślnym dla programu Calc, otwartym formacie OpenDocument. Więcej na jego temat można znaleźć na stronach <http://pl.wikipedia.org/wiki/OpenDocument>.

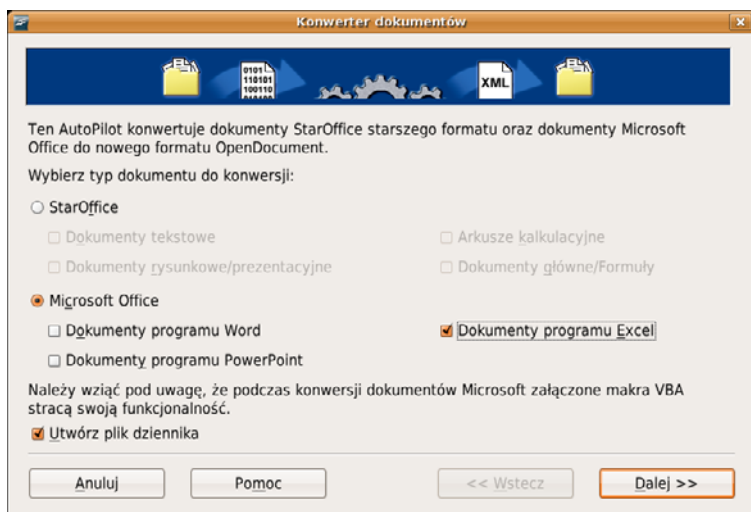
Powyższa metoda sprawdzi się w przypadku posiadania jednego czy dwóch plików, które trzeba przenieść do programu Calc, co jednak zrobić w sytuacji, gdy użytkownik chce przenieść do Ubuntu całą swoją pracę, czyli kilkaset arkuszy kalkulacyjnych? Na szczęście, program wyposażony jest w specjalny kreator, który nie tylko ułatwi przeniesienie wielu plików do nowego programu, ale też zaimportuje i przekonwertuje wszystkie szablony, które zdoła rozpoznać.

Aby zaimportować pliki Microsoft Excela lub pliki ze starszych wersji pakietu OpenOffice.org, należy wykonać następujące czynności.

1. Trzeba wybrać z menu programu opcje *Plik/Kreatory/Konwerter dokumentów*, zostanie wówczas wyświetlone okno widoczne na rysunku 11.33.

Rysunek 11.33.

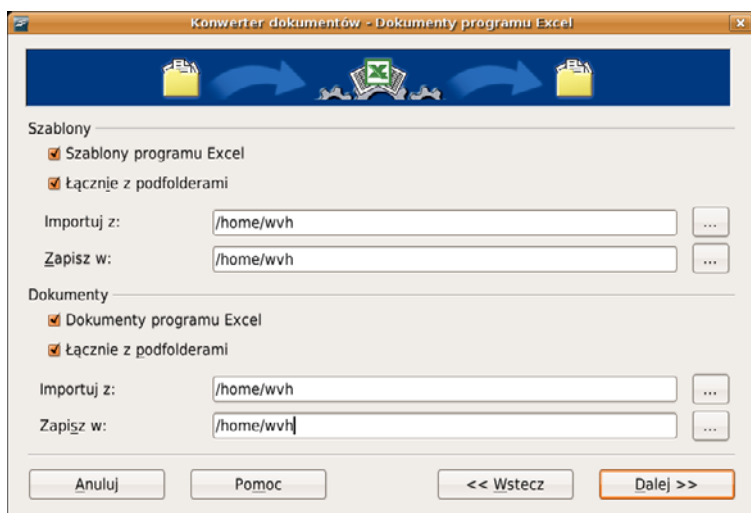
Określanie formatu danych wejściowych w procesie konwersji dokumentów



2. Aby zaimportować zarówno pliki programu Microsoft Excel, jak i szablony, należy zaznaczyć opcję *Microsoft Office* i pole wyboru *Dokumenty programu Excel*. Aby importować istniejące arkusze kalkulacyjne z programu StarOffice (format *SDC*), należy pozostawić zaznaczone pole *StarOffice* oraz zaznaczyć dodatkowo *Arkusze kalkulacyjne* oraz *Dokumenty główne/Formuły*. Następnie trzeba kliknąć przycisk *Dalej*, a zostanie wyświetlone okno dialogowe widoczne na rysunku 11.34.

Rysunek 11.34.

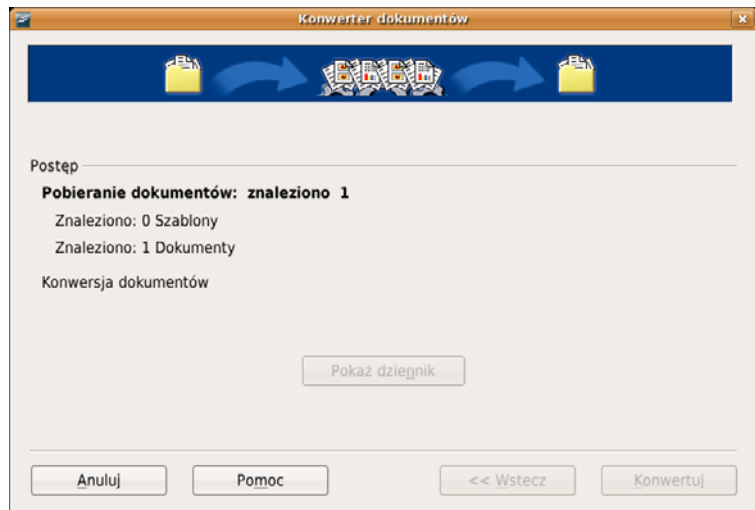
Określanie lokalizacji źródłowej i docelowej importowanych dokumentów



3. Następnie należy określić, gdzie program ma szukać plików źródłowych oraz gdzie ma zapisać wersję ostateczną. Użytkownik może zmienić katalog docelowy; sam zwykle wskazuje swój katalog domowy, następnie przenoszę pliki ręcznie. Po wskazaniu odpowiednich lokalizacji należy kliknąć przycisk *Dalej*. Zostanie wówczas wyświetlone okno z podsumowaniem wybranych ustawień. Następnie trzeba kliknąć przycisk *Konwertuj*, co spowoduje rozpoczęcie procesu. W oknie widocznym na rysunku 11.35 wyświetlany jest postęp realizacji procesu. Po jego zakończeniu staną się aktywne przyciski *Zakończ* oraz *Pokaż dziennik*.

Rysunek 11.35.

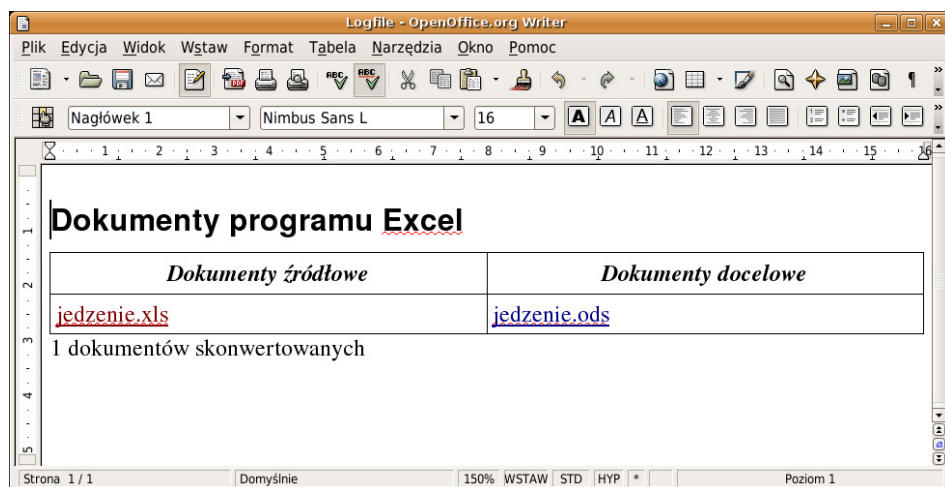
Proces konwertowania dokumentów można obserwować na bieżąco



Gratulacje! Proces konwertowania dokumentów został zakończony. Pliki oraz szablony, które zostały przeniesione do programu Calc, są teraz gotowe do użytku. Przed kliknięciem przycisku *Zakończ* warto najpierw kliknąć przycisk *Pokaż dziennik*, co spowoduje otwarcie pliku dziennika w programie Writer (jeżeli program ten nie jest zainstalowany w systemie lub niedostępny z innych powodów, wówczas przycisk *Pokaż dziennik* pozostanie nieaktywny). Jak widać na rysunku 11.36, plik dziennika zawiera informacje na temat plików, które zostały poprawnie skonwertowane, może też znajdować się tam lista tych arkuszy, które trzeba będzie poprawić lub dopracować, nim zostaną użyte w programie Calc. Tak często dzieje się w złożonych plikach ze skomplikowanymi makrami, które zawierają wpisy kompatybilne tylko z programem Visual Basic firmy Microsoft.

Korzystanie z programu OpenOffice.org Impress

Pakiet biurowy OpenOffice.org zawiera również potężne narzędzie służące do tworzenia i edytowania prezentacji — to program o nazwie Impress. Choć być może użytkownik domowy nigdy nie będzie tworzył prezentacji na swój użytek, to prezentacje są codziennym elementem pracy w biurze i żaden pakiet biurowy nie może się obyć bez programu do ich tworzenia. W dalszej części, do określenia programu OpenOffice.org Impress, stosowana będzie skrócona nazwa Impress.



Rysunek 11.36. Plik dziennika wyświetlany po zakończonym procesie konwertowania plików

Instalowanie plików dla programu Impress

Pakiet OpenOffice.org wraz z programem Impress znajduje się w systemie, jeżeli ten był instalowany z płyt Live CD Desktop i Alternate Install CD (zarówno w wersji tekstowej, jak i OEM). Jeżeli użytkownik chciałby dodać pakiet OpenOffice.org wraz z programem do prezentacji Impress do Ubuntu w wersji serwerowej z zainstalowanym środowiskiem graficznym, powinien zainstalować pakiety, używając jednego z dostępnych menedżerów plików apt-get, aptitude czy Synaptica (zgodnie z informacjami zamieszczonymi w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”).

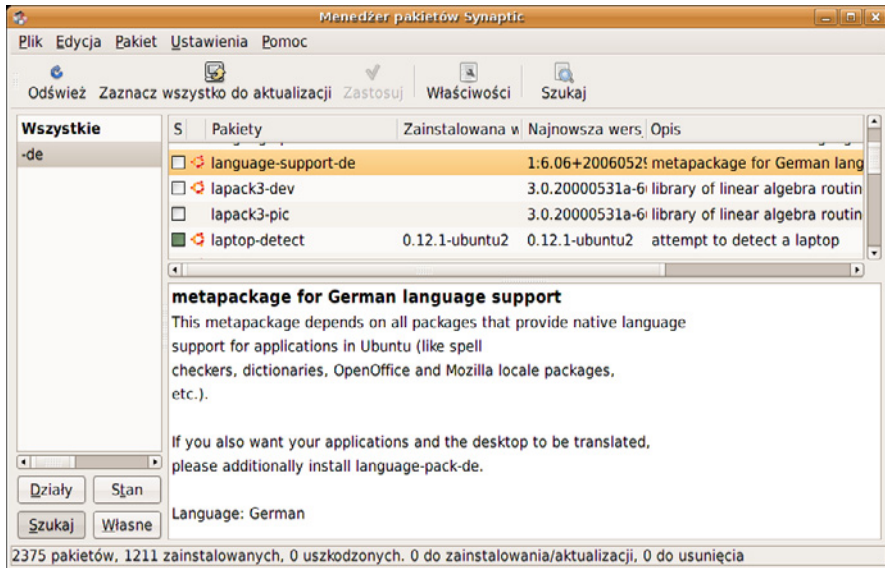
Jedną z cech odróżniających Ubuntu od innych dystrybucji Linuksa jest wyraźne ukierunkowanie na użytkownika, zwłaszcza w zakresie internacjonalizacji. Jeżeli Impress był instalowany później niż sam system, może się okazać, że trzeba będzie zainstalować odpowiednie pliki lokalizacyjne i językowe, dzięki którym program będzie dostępny w odpowiednim języku. Jest to szczególnie ważne, jeżeli w programie otwierane będą pliki utworzone w innych, niż polska, wersjach językowych programu.



Jeżeli zostały już zainstalowane pakiety językowe czy lokalizacyjne dla jakiegokolwiek innego programu wchodzącego w skład pakietu OpenOffice.org, wówczas nie ma potrzeby ponownego ich instalowania. Wystarczy raz umieścić je w systemie, a będą dostępne dla każdego komponentu pakietu.

Pliki i pakiety lokalizacyjne można łatwo odnaleźć za pomocą ciągu określającego dany kraj (kod dla Polski to PL). Na rysunku 11.37 przedstawiam program Synaptic wyświetlający pliki lokalizacyjne dla pakietu OpenOffice.org dla niemieckiej wersji językowej, identyfikowanej za pomocą ciągu -de.

W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, zamieszczam więcej informacji na temat instalowania aplikacji za pomocą menedżera pakietów Synaptic.



Rysunek 11.37. Pakiety językowe dla OpenOffice.org



Użytkownik może zainstalować kompletne wsparcie językowe dla wszystkich aplikacji (bez zmiany wersji językowej samego interfejsu GNOME), wystarczy dodać metapaket dla odpowiednich danych lokalizacyjnych (tzw. *lokalizacji*, ang. *locales*). Przykładowa nazwa takiego pliku wygląda następująco: `language-support-de`, przy czym na końcu znajduje się kod kraju, w pliku znajduje się też słownik dla danego języka. Aby w pełni przetłumaczyć interfejs i menu aplikacji, należy zainstalować odpowiedni pakiet językowy. Nazwy pakietów językowych składają się z kodu języka i kodu kraju. Po zainstalowaniu odpowiednich plików językowych można dowolnie zmieniać język dla danego dokumentu, wystarczy z menu wybrać *Narzędzia/Opcje/Ustawienia językowe/Języki* i na liście rozwijanej *Domyślny język* dokumentów wskazać odpowiednią pozycję.

Uruchamianie programu Impress

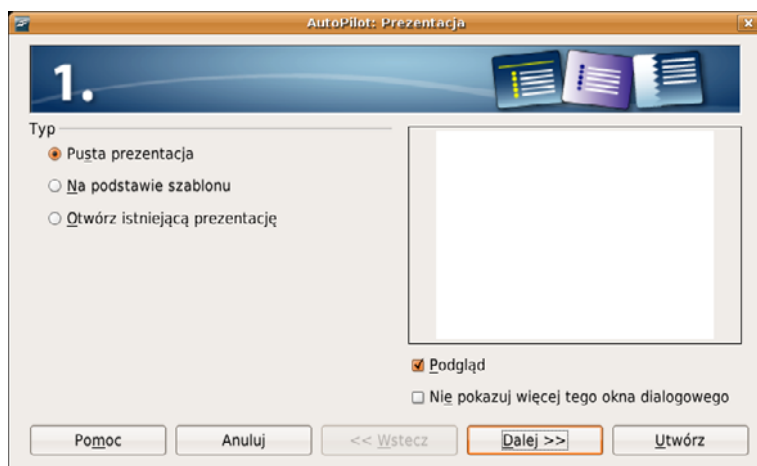
Program można uruchomić z linii poleceń za pomocą komendy `ooimpress`, ale łatwiej będzie użyć menu GNOME. Odpowiednią pozycję można znaleźć w menu *Aplikacje/Biuro*. Po wybraniu pozycji OpenOffice.org Impress zostanie wyświetlone okno programu (patrz rysunek 11.38).

Korzystanie z kreatora prezentacji

Inaczej niż w pozostałych składnikach pakietu OpenOffice.org, po uruchomieniu programu zostanie wyświetlone okno startowe kreatora tworzenia prezentacji (patrz rysunek 11.38).

Jeżeli jednak użytkownik nie wyłączył kreatora prezentacji, wówczas należy kliknąć przycisk *Dalej*, co spowoduje utworzenie nowej prezentacji. Zostanie wyświetlone kolejne okno dialogowe widoczne na rysunku 11.39.

Rysunek 11.38.
 Ekran początkowy
 kreatora prezentacji
 Impress



Rysunek 11.39.
 Określanie tła
 w kreatorze
 prezentacji



Po uruchomieniu kreatora prezentacji można przejść wprost do programu z pominięciem kolejnych etapów tworzenia prezentacji; w tym celu należy zaznaczyć opcję *Otwórz istniejącą prezentację*. Po jej zaznaczeniu użytkownik będzie mógł wskazać w oknie dialogowym prezentację utworzoną wcześniej w programie Impress lub Microsoft PowerPoint. Jeżeli dany plik znajduje się w katalogu domowym użytkownika, wystarczy wówczas kliknąć dwukrotnie jego ikonę, a zostanie otwarty; jeśli znajduje się w innym miejscu, wówczas użytkownik może wskazać odpowiednią lokalizację.

Można na stałe wyłączyć wyświetlanie kreatora prezentacji, wystarczy zaznaczyć pole *Nie pokazuj więcej tego okna dialogowego*. Program zapamięta to ustawienie i następnym razem, kiedy zostanie uruchomiony, wyświetli nową, pustą prezentację.

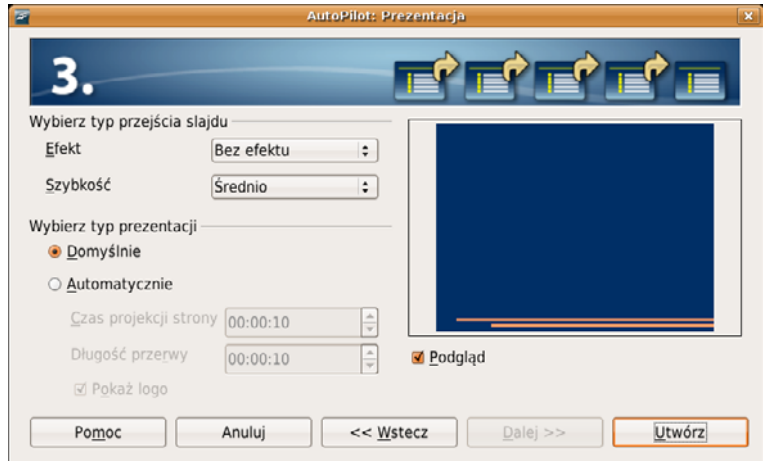
Jeżeli użytkownik nie posiada utworzonej wcześniej prezentacji, a chciałby przejść wprost do tworzenia nowej bez korzystania z kreatora, wówczas należy kliknąć przycisk *Utwórz* przy zaznaczonej opcji *Pusta prezentacja*.

Następnie należy wybrać jedną z dostępnych grafik, która zostanie użyta jako tło prezentacji (aby nie stosować żadnej grafiki, należy pozostawić oryginalne ustawienia), a także określić, na jakim medium będzie wyświetlana prezentacja (na folii, papierze, ekranie, czy slajdach). Określenie urządzenia wyjściowego spowoduje dobranie odpowiednich ustawień w celu

dopasowania do nich wyglądu i działania prezentacji. Aby po wybraniu tych ustawień (w przykładzie użyto tła *Dark Blue with Orange*) przejść do kolejnego etapu, należy kliknąć przycisk *Dalej*. Zostanie wyświetlone kolejne okno dialogowe (patrz rysunek 11.40).

Rysunek 11.40.

Ustalanie szczegółów dla tworzonej prezentacji



Na tym etapie użytkownik może wybrać rodzaj efektów, które będą towarzyszyły zmianie poszczególnych slajdów prezentacji. Dostępnych jest wiele różnych obrotów, ruchów itp. Zamiast jednak wszystkie tu opisywać, radzę, by czytelnicy sami poeksperymentowali. Wystarczy wybrać dany efekt z rozwijanej listy i upewnić się, że pole *Podgląd* jest zaznaczone (ustawienie domyślne), a wówczas wybrany efekt zostanie wyświetlony w oknie podglądu.

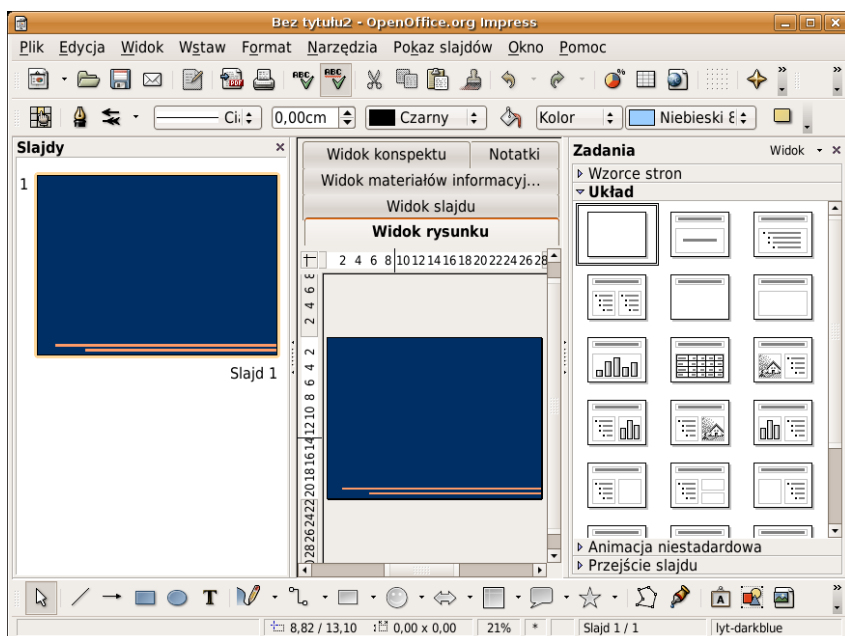
Za pomocą kolejnych ustawień można określić prędkość działania każdego efektu oraz typ tworzonej prezentacji. Domyślnie slajdy są zmieniane tylko wówczas, gdy użytkownik wcisnie odpowiedni klawisz lub inny element, który spowoduje zmianę slajdu. W prezentacji slajdy mogą być zmieniane automatycznie po upływie określonego czasu, co jest przydatne podczas używania programu w terminalach lub na targach, czyli wszędzie tam, gdzie potrzebne jest regularne zmienianie slajdów, które ma uatrakcyjnić pokaz.

Po wprowadzeniu wszystkich potrzebnych zmian należy kliknąć przycisk *Utwórz*, a zostanie zakończony proces tworzenia nowej prezentacji. Następnie zostanie wyświetlony ekran programu Impress (patrz rysunek 11.41).

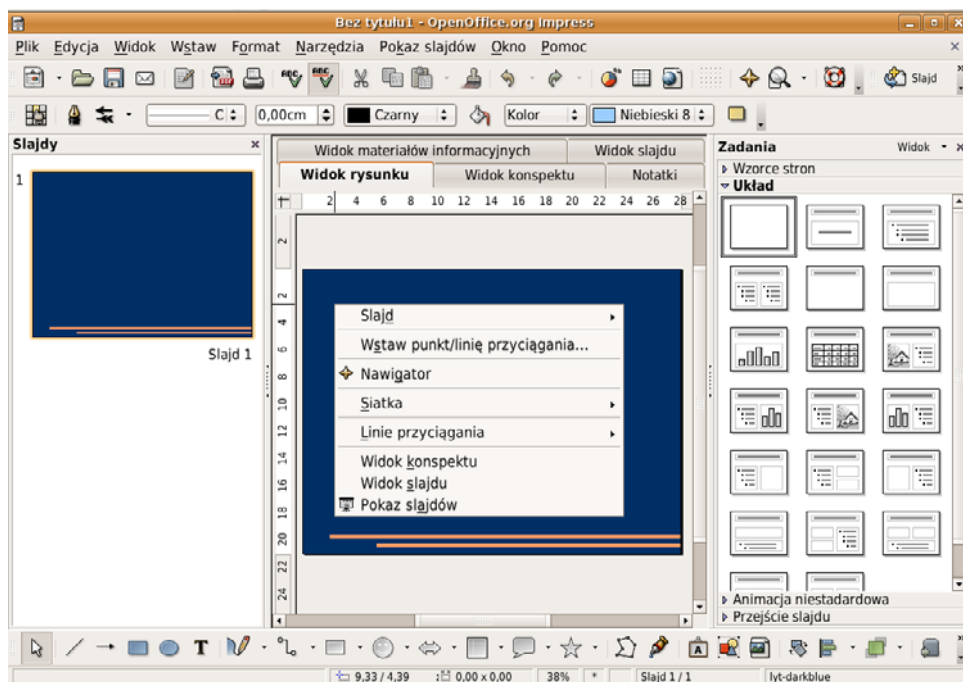
Krótki przewodnik po programie Impress

Impress wyposażony jest w doskonały system pomocy, podobnie zresztą jak cały pakiet OpenOffice.org, dlatego nie będę zanudzać czytelników opisem zawartości każdego menu. Zamiast tego proponuję krótki przewodnik po interfejsie tego programu (rysunki 11.41 i 11.42).

- ♦ **Menu:** rozwijane menu zorganizowane są podobnie jak w programie Microsoft PowerPoint, dzięki czemu użytkownik nie powinien mieć kłopotów z przestawianiem się na nowy program. Pozycje menu ze strzałkami umieszczonymi po prawej stronie umożliwiają dostęp do kolejnych podmenu.



Rysunek 11.41. Standardowe okno programu Impress, w którym wyświetlana jest nowo utworzona prezentacja



Rysunek 11.42. Menu kontekstowe programu Impress

- ♦ **Paski narzędzi:** domyślnie program wyświetla paski narzędziowe, takie jak *Standardowy* i *Prezentacja*, *Rysunek*, *Linie i wypełnienia*, które umożliwiają korzystanie z różnych narzędzi za pomocą ikon znajdujących się na nich. Użytkownik może modyfikować listę wyświetlanych na paskach ikon, wystarczy kliknąć kontrolkę znajdującą się przy prawej krawędzi paska narzędzi ze strzałką skierowaną w dół. Można też włączać i wyłączać wyświetlanie różnych pasków narzędzi, wybierając ich nazwy z menu *Widok/Paski narzędzi*.
- ♦ **Ramka slajdów:** w ustawieniu domyślnym w tym panelu wyświetlany jest podgląd wszystkich slajdów w danej aplikacji.
- ♦ **Panel edycji:** domyślnie w panelu wyświetlany jest podgląd kopii roboczej wybranego slajdu. Użytkownik może dostosować sposób wyświetlania podglądu slajdu, korzystając z zakładek umieszczonych przy górnej krawędzi panelu.
- ♦ **Ramka zadań:** w tym panelu, umieszczonym po prawej stronie okna programu, użytkownik może zdefiniować układ, który zostanie zastosowany do aktualnie wybranego slajdu. Można zmieniać wyświetlaną zawartość okna, wybierając kolejne opcje, takie jak *Wzorce stron*, *Układ*, *Animacja niestandardowa* i *Przejsście slajdu*.
- ♦ **Menu kontekstowe:** po podświetleniu dowolnego slajdu wystarczy wcisnąć prawy przycisk myszy, zostanie wówczas wyświetlone menu kontekstowe widoczne na rysunku 11.42. Umożliwia ono łatwy dostęp do kilku podstawowych operacji związanych ze slajdami.



Chociaż okno *Style i formatowanie* nie jest wyświetlane domyślnie, można je przywołać w każdej chwili, wciskając klawisz *F11* (aby wyłączyć okno, należy powtórnie wcisnąć ten sam klawisz). Okno to umożliwia łatwy dostęp do stylów dostępnych w danej prezentacji i jest szczególnie użyteczne, jeżeli nie jest wyświetlany pasek narzędziowy *Formatowanie*. Na rysunku 11.43 widoczne jest okno *Style i formatowanie*. Jak każde pływające okno, można je zakotwiczyć w głównym oknie, klikając dwukrotnie szary obszar z nazwą okna. Aby okno odłączyć od głównego okna programu, należy dwukrotnie kliknąć nagłówek zadokowanego obszaru.

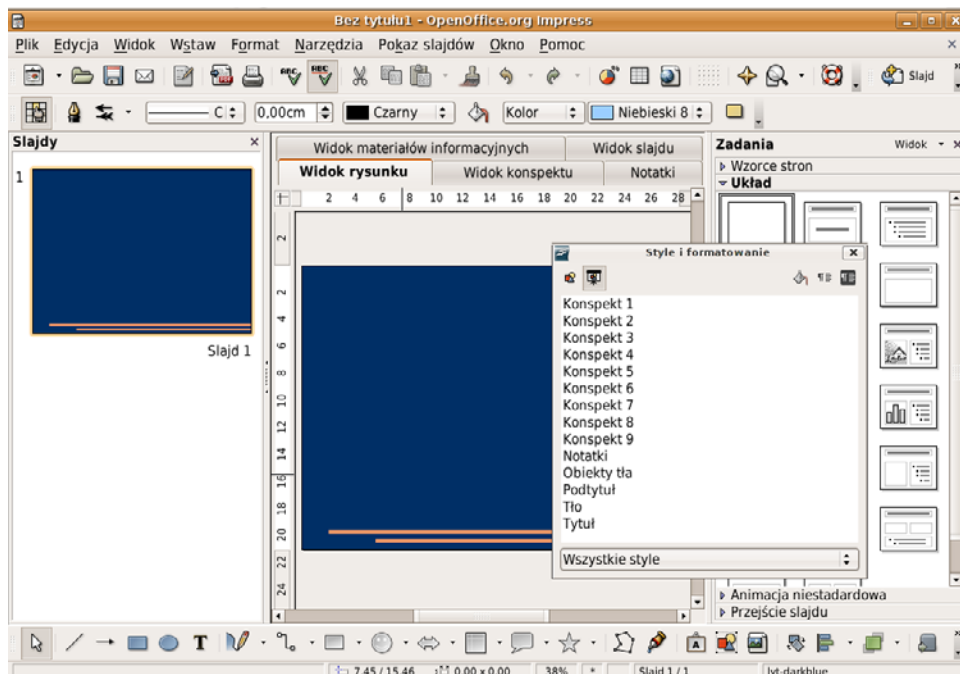
Program Impress jest też wyposażony w obszerny system pomocy, wystarczy wybrać z menu *Pomoc/Pomoc Openoffice.org* lub wcisnąć klawisz *F1* w czasie pracy z programem. Na rysunku 11.44 widać zawartość okna z pomocą programu Impress. Można w nim znaleźć wyszukiwarkę słów kluczowych, listę tematów oraz szczegółową zawartość wybranego zagadnienia (po prawej stronie okna).



Jeżeli pojawi się jakiś problem z wyświetlaniem zawartości pomocy programu Impress, należy zainstalować program Writer. To wymusi zaktualizowanie komponentu odpowiedzialnego za wyświetlanie zawartości pomocy w pakiecie OpenOffice.org, co zwykle rozwiązuje problem.

Tworzenie prezentacji

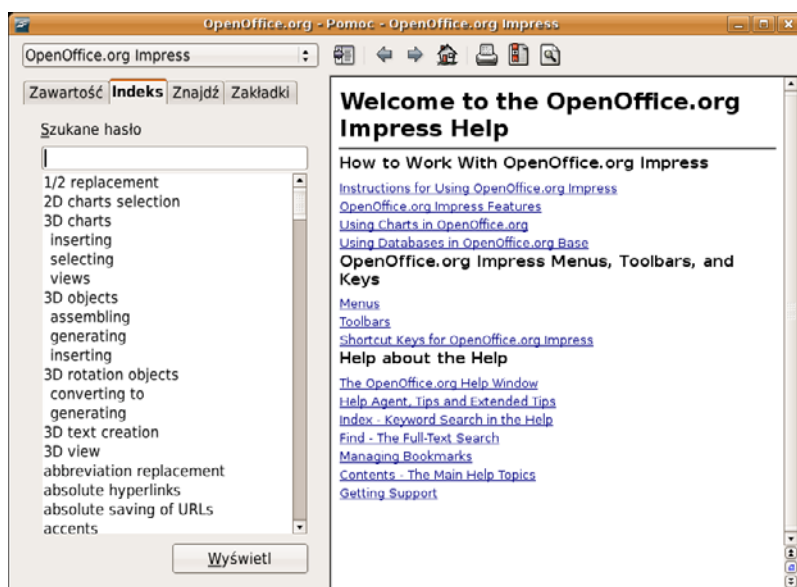
Tworzenie prezentacji za pomocą programu Impress jest bardzo proste, chociaż ustawienie wszystkich szczegółów może być takim samym wyzwaniem, jak w każdym innym programie do tworzenia prezentacji.



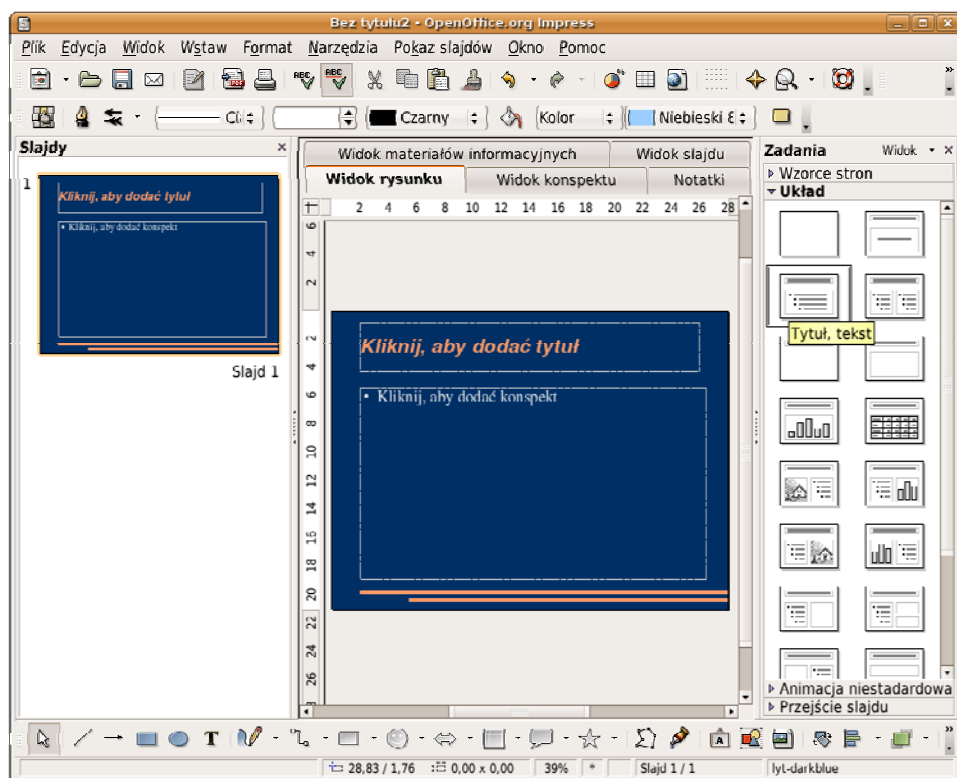
Rysunek 11.43. Okno style i formatowanie w programie Impress

Rysunek 11.44.

Pomoc programu
Impress



Aby wprowadzić do wybranego slajdu jeden z dostępnych formatów, należy go wybrać z okna *Układ*, zostanie wówczas tam wprowadzony. Następnie można kliknąć którekolwiek z dostępnych pól tekstowych i wprowadzić tekst. Na rysunku 11.45 widoczny jest slajd z zastosowanym układem *Tytuł, tekst* (prawy górny róg okna *Układ*).



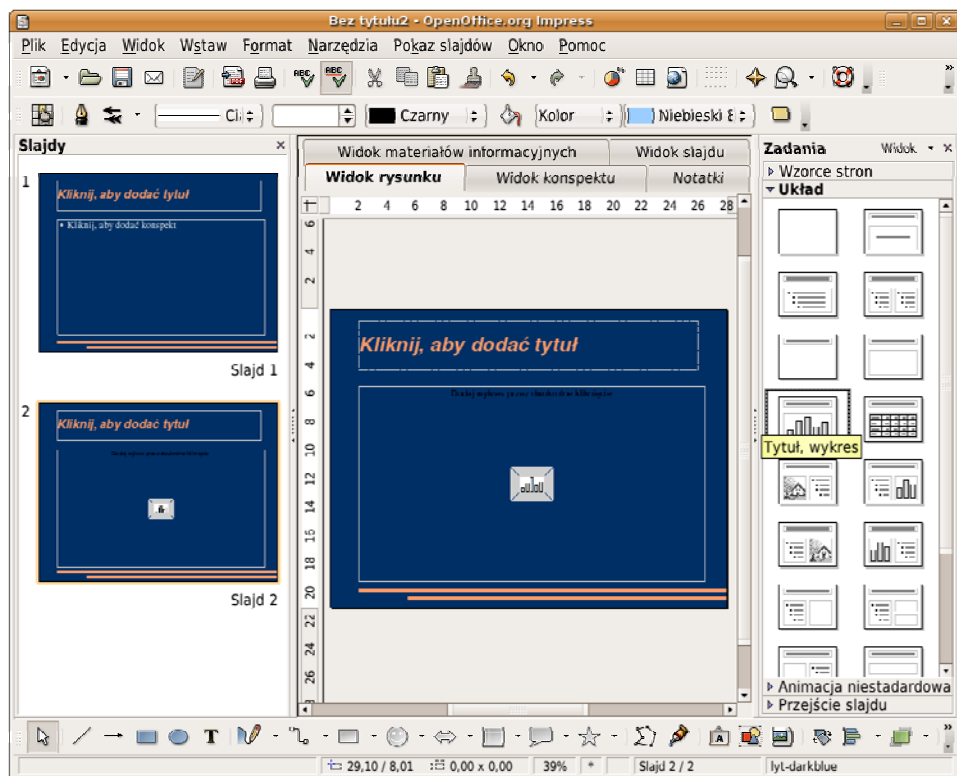
Rysunek 11.45. Tworzenie slajdu tytułowego w programie Impress

W programie Impress dostępnych jest wiele przykładowych układów, takich jak slajdy tytułowe, slajdy z nagłówkami i miejscem na grafikę itp. Poszczególne pola można identyfikować, umieszczając nad nimi kursor myszy i odczytując wyświetlane wskazówki. Przykład slajdu tytułowego przedstawiono na rysunku 11.45.

Po wprowadzeniu treści lub innego obiektu do bieżącego slajdu można dodać kolejny, wybierając z menu *Wstaw/Slajd*. Zostanie on wówczas wyświetlony w oknie edycji, zawartość okna podglądu zostanie również zaktualizowana tak, aby wyświetlać nowy slajd oraz kolejne. Na rysunku 11.46 widać nowy slajd z obiektami *Tytuł*, *wykres* i podglądem bieżącego slajdu. Na rysunku widać również wyświetlane odpowiedzi.

Kiedy do prezentacji zostanie już dodanych kilka slajdów, warto zapisać plik za pomocą polecenia *Plik/Zapisz*. Jeżeli jest to nowy plik, użytkownik będzie musiał podać dla niego nazwę.

Po zapisaniu prezentacji użytkownik będzie chciał zapewne sprawdzić, jak wygląda. Aby wyświetlić podgląd prezentacji, należy z menu wybrać kolejno *Widok/Pokaż slajdów* lub wcisnąć klawisz *F5*. Prezentacja zostanie wyświetlona wraz ze wszystkimi ustawionymi wcześniej efektami.



Rysunek 11.46. Dodawanie nowego slajdu w programie Impress



Podgląd prezentacji zostanie rozpoczęty od slajdu ustawionego w panelu podglądu. Aby wyświetlić całą prezentację, przed rozpoczęciem wyświetlania należy wybrać pierwszy slajd.

Importowanie istniejących prezentacji

Impress może otwierać prezentacje utworzone w innych programach, takich jak Microsoft PowerPoint i oczywiście StarOffice. Po zakończeniu ich edytowania można zapisać je nowym formacie OpenDocument, używanym domyślnie w programie Impress.

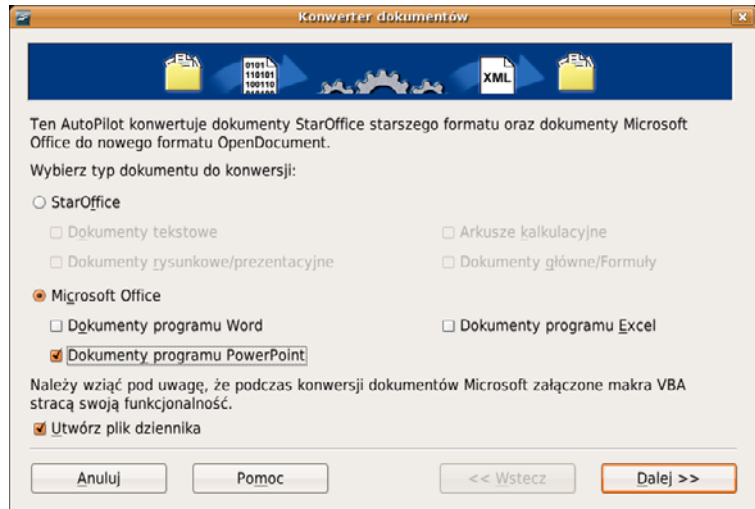
Wykonanie tego w przypadku jednej lub dwóch prezentacji może wystarczyć, jednak może to być kłopotliwe w sytuacji, gdy użytkownik chciałby przenieść do Impressa większą ilość plików. Na szczęście, program posiada kreator, który nie tylko zaimportuje pliki utworzone w programie Microsoft PowerPoint, ale też przekonwertuje ustawienia plików, które wykryje. Program ten nazywa się Konwerter dokumentów.

Konwerter ten działa identycznie jak opisany poprzednio, działający w programie Calc, dlatego nie będę marnować papieru na powtarzanie tych samych instrukcji. Więcej informacji na temat używania Konwertera dokumentów można znaleźć we wcześniejszym punkcie rozdziału zatytułowanym „Importowanie arkuszy za pomocą programu Konwerter dokumentów”.

Jedyna różnica pomiędzy importowaniem prezentacji i arkuszy kalkulacyjnych (czy raczej plików tekstowych, zgodnie z informacjami zamieszczonymi w rozdziale 10.) polega na tym, że w oknie programu należy wybrać pola *Microsoft Office* i *Dokumenty programu PowerPoint* (patrz rysunek 11.47).

Rysunek 11.47.

Określanie formatów wejściowych podczas konwertowania prezentacji



Podsumowanie

W niniejszym rozdziale omówiłem różne typy oprogramowania biurowego, którego użytkownik może potrzebować, aby całkowicie przejść na system Ubuntu. Systemy linuksowe często mają złą prasę, ponieważ uważa się, że nie zawierają wystarczającego wyboru aplikacji; w tym rozdziale pokazałem, że nie jest to prawda. Nie tylko dostępna jest szeroka gama różnorodnych i wydajnych aplikacji biurowych, ale w dodatku wolne oprogramowanie daje użytkownikom szerokie możliwości wyboru, a wszystkie za tę samą, niską cenę wynoszącą 0 złotych.

W rozdziale omówiłem dwa popularne arkusze kalkulacyjne dla Linuksa: Gnumeric i OpenOffice.org Calc. Po lekturze tego programu łatwo stwierdzić, że są podobne w wyglądzie i działaniu. Rozdział kończy się omówieniem programu OpenOffice.org Impress, potężnego i wydajnego narzędzia do tworzenia prezentacji, ich edycji i zarządzania nimi. Aplikacja ta jest częścią pakietu biurowego OpenOffice.org.

Po omówieniu zagadnień dotyczących tworzenia dokumentów, arkuszy kalkulacyjnych i prezentacji, w następnym rozdziale opiszę oprogramowanie, za którego pomocą można w systemie Ubuntu tworzyć i obrabiać różnorodne typy plików graficznych.

Rozdział 12.

Praca z grafiką

W tym rozdziale:

- ◆ Podstawowe informacje o grafice cyfrowej
- ◆ Praca z obrazami
- ◆ Korzystanie z narzędzi do rysowania w Ubuntu
- ◆ Praca z grafiką wektorową

Sam, będąc pisarzem, zawsze powtarzam stare powiedzenie, że jeden obraz wart jest tysiąc słów. Napiszę zatem tysiąc słów i przejdziemy do następnego tematu.

To — oczywiście — żart. Jest wiele sytuacji, w których obraz okazuje się być wart więcej niż słowa, choćby w przypadku zrzutów ekranu w niniejszej książce, fotografii i grafiki na okładce czy w różnych wydawnictwach specjalizujących się w gloryfikowaniu ludzkich kształtów. Nikt nigdy nie powiedział o obszernym dokumencie tekstowym, że przyciąga wzrok nawet wtedy, jeśli wysiłek osób pracujących nad wyglądem stron był ogromny. Jeżeli nawet użytkownik komputera nie jest grafikiem, to obecnie nadeszły takie czasy, w których każdy potrzebuje oprogramowania do edycji i tworzenia grafiki, choćby po to, aby usunąć efekt czerwonych oczu z wakacyjnych zdjęć lub też utworzyć prostą ilustrację do kartki pocztowej, listu czy prezentacji.

Czytelnicy, którzy pracowali z programami graficznymi, takimi jak Adobe Photoshop, Adobe Illustrator, CorelDRAW czy Paint Shop Pro (znanym też pod nazwą JASC), ucieszą się zapewne z wiadomości, że również dla Linuksa dostępne są pakiety oprogramowania umożliwiające wykonanie tego samego rodzaju pracy, w dodatku bezpłatne. Niektóre z nich są już zainstalowane w systemie, inne dostępne są w repozytoriach i wystarczy je tylko pobrać.

W niniejszym rozdziale ułatwię czytelnikowi rozeznanie się w skomplikowanej terminologii graficznej, żargonie używanym do opisywania grafiki cyfrowej i wyjaśnię, w jaki sposób używać dostępnych w Ubuntu programów, aby tworzyć i edytować rysunki, obrazy oraz inne rodzaje grafiki.

Terminologia

Jeśli czytelnik nie jest grafikiem (podobnie jak ja), to przecież każdy, kto tworzy strony internetowe czy używa aparatu cyfrowego również pracują z grafiką. Terminologicznie rzecz ujmując, grafika komputerowa to nic innego jak przechowywanie obrazów w komputerze, niezależnie od tego, czy są to zdjęcia z wakacji, zabawne przyciski i inne grafiki wykorzystywane na stronie internetowej, grafiki tworzone za pomocą odpowiedniego oprogramowania, obrazy uzyskane za pomocą skanera, obrazy pobrane wprost z ekranu monitora i właściwie wszystko to, co można przechowywać w komputerze i wyświetlać za pomocą oprogramowania graficznego. Oto krótki słownik terminów graficznych, na jakie może natknąć się czytelnik, rozpoczynając pracę z grafiką cyfrową.

- ♦ **Anti-aliasing:** technika używana do redukowania poszarpanych krawędzi obiektów graficznych i tekstu. Krawędzie te zostają rozmyte poprzez połączenie ich z kolorem, który znajduje się na obszarach położonych obok.
- ♦ **Bitmapy, mapy bitowe:** obrazy składające się z wielu elementów, każdy reprezentujący pojedynczy element obrazu. Każdy taki element posiada informację o kolorze, który jest kombinacją trzech kolorów (czerwonego, zielonego i niebieskiego). Bitmapy są jednym z dwóch głównych formatów używanych w grafice cyfrowej; drugim są obrazy wektorowe. W odniesieniu do bitmap używa się czasem określenia obrazy rastrowe.
- ♦ **BMP:** format plików graficznych, zaprojektowany przez Microsoft, obsługujący różnorodne rozdzielczości i głębie kolorów. Plik BMP zawiera pojedynczą bitmapę (obraz rastrowy).
- ♦ **Głębia koloru:** maksymalna liczba kolorów zamieszczonych na obrazie lub możliwa do wyświetlenia przez dane urządzenie (monitor czy kartę graficzną). Używane wartości to: 256 kolorów, 16 bitów (czyli 65 535 kolorów), 32 bity (czyli 4 294 967 296 kolorów, często też określany jako miliony kolorów) itd. Im więcej kolorów znajduje się na obrazie, tym większy jego rozmiar, ponieważ kolor każdego piksela wymaga odpowiedniej liczby bitów danych zawierających informacje o proporcjach użytych kolorów: czerwonego, zielonego i niebieskiego. Głębia kolorów jest często definiowana jako ilość „bitów na piksel” i tak np. dany kolor może używać 15 lub 16 bitów: 5 dla koloru czerwonego, 5 dla zielonego i 5 lub 6 dla niebieskiego.
- ♦ **Dithering:** wyświetlanie pikseli o różnych kolorach w jednej części obrazu tak, aby stworzyć wrażenie jednolitego koloru. Ta technika jest często używana w obrazach o niższej rozdzielczości, dzięki czemu można nadać im bardziej wygładzony wygląd, imitujący obraz o wyższej rozdzielczości.
- ♦ **GIF:** popularny, bezstratny format graficzny. *Graphical Interchange Format* jest szczególnie popularny na stronach internetowych, ponieważ umożliwia tworzenie plików o małych rozmiarach, a jednocześnie stosunkowo wysokiej jakości, w stosunku do których można zastosować kompresję LZW (*Lempel-Zich-Welch*), opatentowaną przez Unisys. Ze względów licencyjnych na większości stron używa się obecnie grafiki w formacie JPEG lub PNG. Podstawową przewagą plików GIF jest fakt, że jeden plik może składać się z wielu obrazów, które w dodatku można wyświetlać w określonej sekwencji. Takie obrazy określa się nazwą animowanych GIF-ów.

Aby uniknąć problemów związanych z licencjonowaniem, wielu twórców stron internetowych korzysta obecnie z formatów, takich jak Flash, QuickTime czy Shockwave, lub opracowuje animacje w środowisku Java.

- ♦ **Skala szarości:** obrazy skomponowane z różnych odcieni szarości. Ilość odcieni szarości, które mogą być wyświetlone, jest określana przez liczbę bitów przechowywujących informacje o wartości koloru wyświetlanego przez dany piksel. Jeżeli liczba bitów wynosi 1, wówczas obraz jest czarny lub biały (używa się też określenia monochromatyczny). Skala szarości jest zwykle używana w odniesieniu do gazet lub książek, które drukowane są w kolorach czarnym i białym, a półtony stosowane są do uzyskiwania różnych odcieni szarości.
- ♦ **Półtony:** służą do symulowania płynnych przejść tonalnych (cieniowane rysunki, fotografie), co uzyskuje się przez użycie punktów o różnej wielkości do różnicowania obszarów obrazu. Mniejsze punkty używane są dla obszarów o jasnych barwach, większe dla obszarów ciemniejszych. Półtony są zwykle wykorzystywane do reprodukcji czarno-białych obrazów o stosunkowo wysokiej rozdzielczości, wykorzystywanych w gazetach czy czasopismach.
- ♦ **Postrzępienie:** termin służący do określania efektu „poschodkowanych” krawędzi na obrazach, czasami określany pojęciem aliasingu. Ten nieprzyjemny efekt jest eliminowany przy pomocy technik takich jak anti-aliasing.
- ♦ **JPEG:** popularny bitmapowy format graficzny zawierający 24 bity kolorów, zaprojektowany przez Joint Photographic Experts Group. JPEG jest formatem stratnym, często używanym na stronach internetowych, oferującym dobry kompromis pomiędzy jakością obrazu a wielkością pliku.
- ♦ **Warstwy:** większość skomplikowanych plików graficznych składa się z wielu warstw, których połączenie daje ostateczny efekt. Praca na różnych warstwach obrazu (którymi mogą być tło czy poszczególne elementy obrazu) nie tylko ułatwia obróbkę wybranych elementów bez ingerencji w pozostałe, ale też przyspiesza tworzenie złożonych obrazów z oddzielnych elementów, które później można wykorzystać do kreowania kolejnych obrazów.
- ♦ **Formaty bezstratne:** formaty plików, które podczas zapisu nie tracą żadnych danych, np. prezentują materiał audio czy wideo w oryginalnej jakości, z jaką został on nagrany czy pozyskany.
- ♦ **Formaty stratne:** formaty, w których jakość obrazu (czy dźwięku) zostaje poświęcona dla zmniejszenia rozmiarów plików. Termin „stratne” odnosi się do algorytmów kompresji danych używanych do zmniejszania rozmiarów plików. Część informacji zamieszczonych w oryginalnym pliku zostaje usunięta po to, aby pliki wynikowe miały mniejszą wielkość. Z tego powodu nigdy nie uda się pliku zapisanego w formacie stratnym przywrócić do postaci oryginalnej. W grafice redukcji ulegają zwykle palety kolorów oraz liczba pikseli w obrazie. Formaty stratne są często używane w odniesieniu do plików graficznych, audio czy wideo, ponieważ umożliwiają zredukowanie rozmiarów plików poprzez usunięcie informacji, które są poza zasięgiem percepcji przez ludzki wzrok, słuch lub oba te zmysły jednocześnie.
- ♦ **Metadane:** czyli dane o danych; w plikach graficznych metadane dostarczają zwykle informacji o szczegółach i atrybutach plików (nazwa, rozmiar, położenie, głębokość kolorów, typ danych itd.) lub dodatkowych komentarzy. Metadane upraszczają

analizowanie i użytkowanie plików graficznych, dostarczając interfejsu wyższego poziomu, a nie tylko surowych plików graficznych.

- ♦ **Monochromatyczne:** tekst i obrazy wyświetlane tylko za pomocą czarnych i białych pikseli. Termin pochodzi stąd, że używana jest tylko jedna wartość chromatyczna (w tym przypadku kolor): czarny to piksel z ustawioną wartością tego koloru, podczas gdy biały to piksel o kolorze nieczarnym.
- ♦ **PCL:** język kontroli drukarki (ang. *Printer Control Language*) zaprojektowany przez firmę Hewlett-Packard (HP) i używany obecnie w tysiącach różnych modeli drukarek. Dzięki PCL pliki mogą być drukowane bezpośrednio na drukarkach HP i kompatybilnych bez wstępnego procesu obróbki.
- ♦ **Piksele:** najmniejszy element obrazu graficznego, który może być wyświetlony na monitorze komputera. Każdy piksel posiada swój własny kolor, zwykle wyrażany jako połączenie kolorów podstawowych: czerwonego, zielonego i niebieskiego. Sam termin pochodzi od słów *picture element* (element obrazu). Większość plików graficznych, np. zdjęcia cyfrowe, składa się z milionów pikseli — stąd termin megapiksele. Im więcej pikseli posiada obraz, tym więcej zawiera szczegółów i tym większą ma rozdzielczość.
- ♦ **PNG:** czyli przenośny format graficzny (*Portable Graphic Format*); powstał jako odpowiedź na problemy licencyjne z formatem GIF. PNG jest formatem bezstratnym, obsługuje kolory w głębi do 48 bitów i 16 bitów w przypadku skali szarości bez utraty jakichkolwiek szczegółów podczas kompresji.
- ♦ **PostScript:** język kontroli drukarek; zaprojektowany przez firmę Adobe i używany obecnie przez tysiące drukarek laserowych. Pliki postscriptowe obsługują zarówno bitmapy, jak i podstawowe elementy graficzne i dlatego pozwalają uzyskiwać wydruki wysokiej jakości. PostScript zainspirowany został przez wcześniejszy język obsługi drukarek, InterPress, stworzony przez Xerox PARC, kiedy nie było jeszcze tak przydatnych rzeczy jak ethernet, drukarki laserowe itd.
- ♦ **Prepress:** proces przygotowania plików do druku. Obecnie obejmuje komputerowe przygotowanie dokumentów lub obrazów do druku, dawniej dotyczył przygotowania fotograficznych negatywów, ręcznego ustawienia układu strony i czcionek, a także produkcji płyt drukowych.
- ♦ **Grafika rastrowa:** inne określenie grafiki bitmapowej. Termin raster odnosi się do pozycji na siatce współrzędnych wyświetlanej na monitorze komputera.
- ♦ **Rozdzielczość:** wyjściowa jakość obrazu definiowana w kategoriach punktów, linii czy pikseli, zależy od urządzenia wyjściowego. PPI (ang. *pixels per inch*, czyli piksele na cal) dotyczy rozdzielczości wyświetlania; DPI (ang. *dots per inch*, czyli punkty na cal) dotyczy rozdzielczości drukowania; SPI (ang. *samples per inch*, czyli próbka na cal) dotyczy rozdzielczości obrazu pobieranego ze skanera; LPI (ang. *lines per inch*, czyli linie na cal) dotyczy rozdzielczości półtonów. Wysoka rozdzielczość zawiera 300 i więcej jednostek na cal i jest zwykle stosowana do wydruków. Obrazy o niskiej rozdzielczości zazwyczaj zawierają ok. 100 lub mniej jednostek na cal i zwykle są używane w odniesieniu do stron internetowych czy generalnie treści wyświetlanych na ekranie komputera.
- ♦ **Efekt schodków:** inny termin określający zjawisko postrzępienia.

- ♦ **TIFF:** *Tagged Image File Format*, popularny bezstratny format graficzny. Pliki zapisane w tym formacie mają rozszerzenie *.TIF* lub *.TIFF*.
- ♦ **Grafika wektorowa:** grafika cyfrowa i skojarzone z nią formaty plików, w których definiuje się każdy element obrazu poprzez określenie jego cech charakterystycznych, takich jak położenie, kształt, paleta kolorów, skojarzony tekst, gradienty, wzory i elementy grupy. Grafika wektorowa nie ma ściśle określonej rozdzielczości, ponieważ w każdej chwili może zostać przerysowana w dowolnej rozdzielczości przez odpowiednią aplikację graficzną czy urządzenie.

Najlepsze źródło informacji na temat grafiki źródłowej, jakie odnalazłem w sieci, to strona About.com Graphic Software (<http://graphicsoft.about.com>), można na niej znaleźć obszerny słownik terminów graficznych, setki poradników, artykułów i dyskusji. To naprawdę zdumiewające. Jedynym problemem jest tytuł strony, czyli „O oprogramowaniu graficznym dla PC-tów i Maców” (*About Graphics Software for PC or Mac*). Hej, a co z nami? Na szczęście, jak czytelnik przekona się po przeczytaniu niniejszego rozdziału, bezpłatne narzędzia graficzne dostarczane wraz z Ubuntu są więcej niż wystarczające do wykonania większości zadań związanych z obróbką obrazów.

Używanie programu GIMP

GIMP, czyli GNU Image Manipulation Program, jest podstawowym programem graficznym o otwartym kodzie dla systemów Linux. GIMP, wcześniej znany też pod nazwą General Image Manipulation Program, został zaprojektowany przez Spencera Kimballa i Petera Mattisa. Od samego początku był przystosowany do rozbudowy za pomocą rozszerzeń tworzonych i kompilowanych osobno, ale działających na tej samej platformie, co GIMP. Jak wszystkie programy o otwartych źródłach, również GIMP czerpie nieustannie z pracy innych, dotyczy to zwłaszcza dodatków i rozszerzeń. Ten program to doskonały przykład znakomitej aplikacji, która posiada również wersje dla innych systemów operacyjnych (Microsoft Windows oraz Mac OS X) i jest tam szeroko używana. Wszelkie informacje na temat historii GIMP-a można znaleźć na stronie internetowej programu (www.gimp.org/about/ancient_history.html).

GIMP jest już też legendą oprogramowania GNU, zwłaszcza od czasu, kiedy istnieje GNOME, a to dlatego, że środowisko to korzysta z interfejsu opartego na bibliotece GTK, czyli GIMP Tool Kit. Aby nie streszczać całej historii GIMP-a, wystarczy powiedzieć, że choć program ten na początku korzystał z innej biblioteki, to jego twórcy zdecydowali się w końcu stworzyć własną, na tyle rozbudowaną i dobrze wyposażoną, że obecnie jest używana przez inne programy. Chronologicznie rzecz ujmując: GIMP doprowadził do powstania GTK, z której wywodzi się GTK+, z której wywodzi się GNOME, które teraz wspiera GIMP-a. Mam nadzieję, że jest to jasne — w dalszej części rozdziału zamieszczam quiz sprawdzający wiedzę.



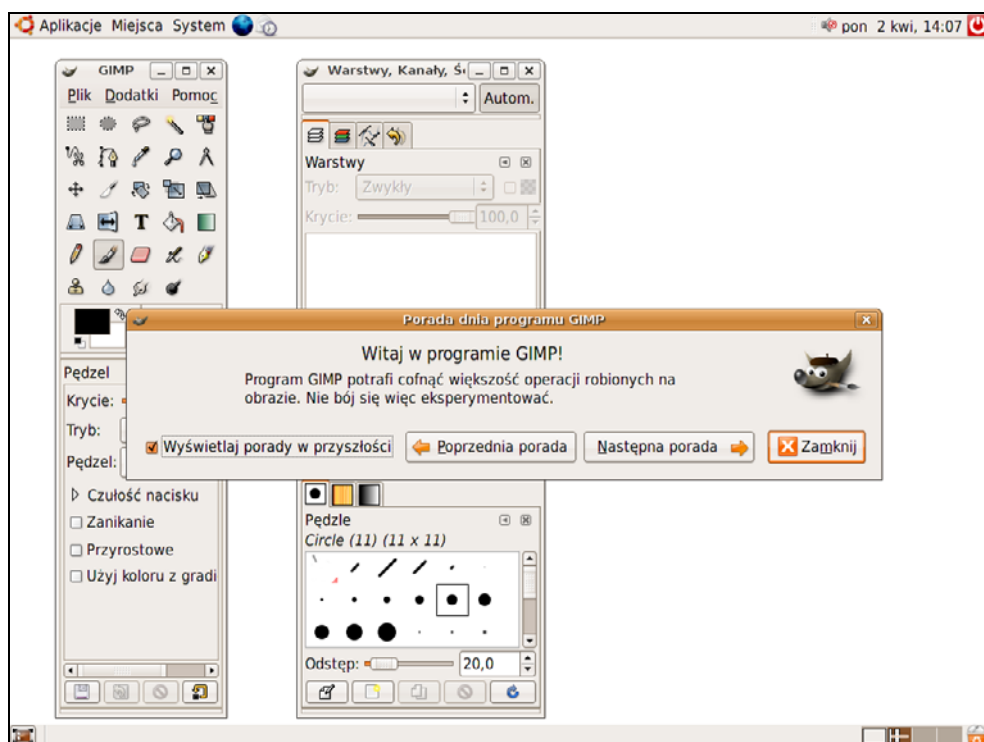
Użytkownicy często mówią o GIMP-ie po prostu gimp, a to dlatego, że jest to też nazwa polecenia, za którego pomocą można uruchomić ten program z poziomu linii poleceń. W dalszej części rozdziału nazwa GIMP będzie używana w przypadku odwoływania się do programu, natomiast gimp wówczas, gdy mowa będzie o poleceniu konsoli.

Uruchamianie programu GIMP

GIMP jest instalowany automatycznie jako część systemu Ubuntu, dlatego aby go używać, nie jest potrzebny żaden specjalny proces instalacyjny. Program można uruchomić na wiele sposobów, najpowszechniejsze to:

- ♦ wybranie z menu *Aplikacje/Grafika/Edytor obrazów GIMP*,
- ♦ wpisanie w oknie terminala GNOME polecenia `gimp`.

Po pierwszym uruchomieniu programu zostaną wyświetlone okna dialogowe i panele widoczne na rysunku 12.1.



Rysunek 12.1. Pierwsze uruchomienie programu GIMP

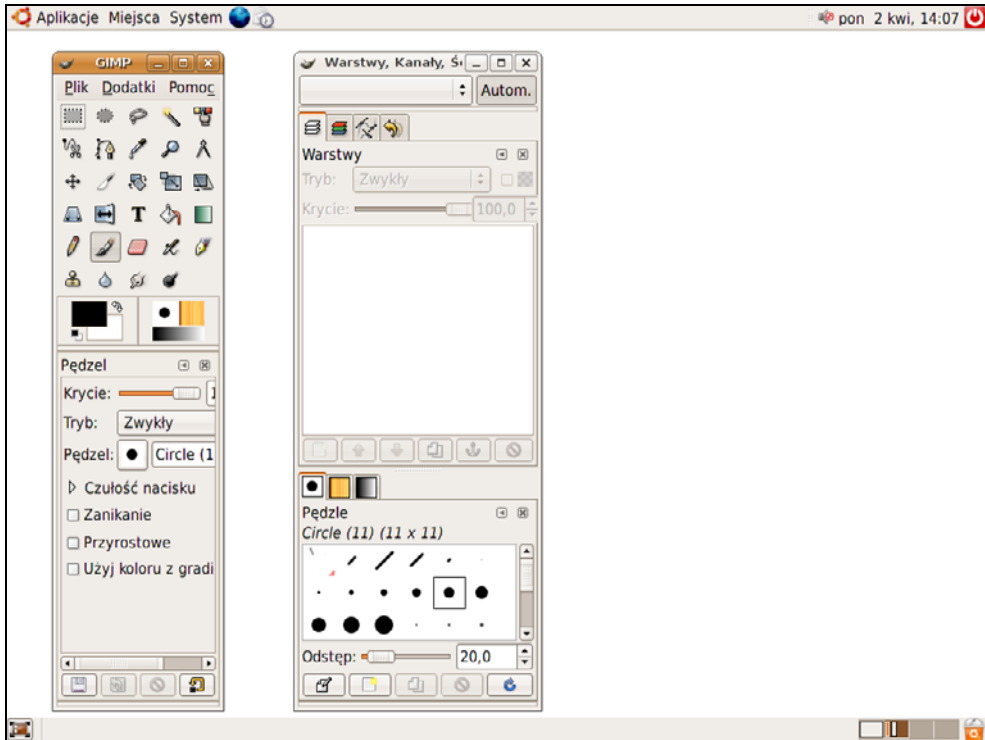


Użytkownicy wcześniejszych wersji GIMP-a zapewne pamiętają, że program przed uruchomieniem wyświetlał kilka ekranów startowych umożliwiających dopasowanie konfiguracji do potrzeb użytkownika. Ponieważ jednak znakomita większość akceptowała standardowe ustawienia, ten etap konfiguracji został usunięty w kolejnych wydaniach. Dzięki temu GIMP uruchamia się teraz trochę szybciej.

Pierwszą czynnością, którą wykonuje większość użytkowników programu po jego uruchomieniu, jest dostosowanie wyświetlanych paneli i okien. Poniżej opisuję każdy dostępny panel programu GIMP.

Krótki przewodnik po programie GIMP

Na rysunku 12.1 przedstawiam wygląd programu po jego pierwszym uruchomieniu. Zwykle wyłączam wyświetlanie porad ekranowych, dzięki czemu program po uruchomieniu wygląda tak, jak na rysunku 12.2.



Rysunek 12.2. Podstawowe okna programu GIMP

Jak widać, program wyświetla dwa główne okna. Okno zawierające wiele innych połączonych w jednym, nosi nazwę okna dokującego. Dzięki temu program jest jeszcze bardziej elastyczny, ponieważ użytkownik ma pełną swobodę w konfigurowaniu sposobu wyświetlania okien (mogą być wyświetlane osobno lub łączone w jednym, większym). Na rysunku 12.2 widać dwa okna wyświetlone domyślnie przez program. Oto krótki opis tych okien z uwzględnieniem zawartości, która może być w nich wyświetlana.

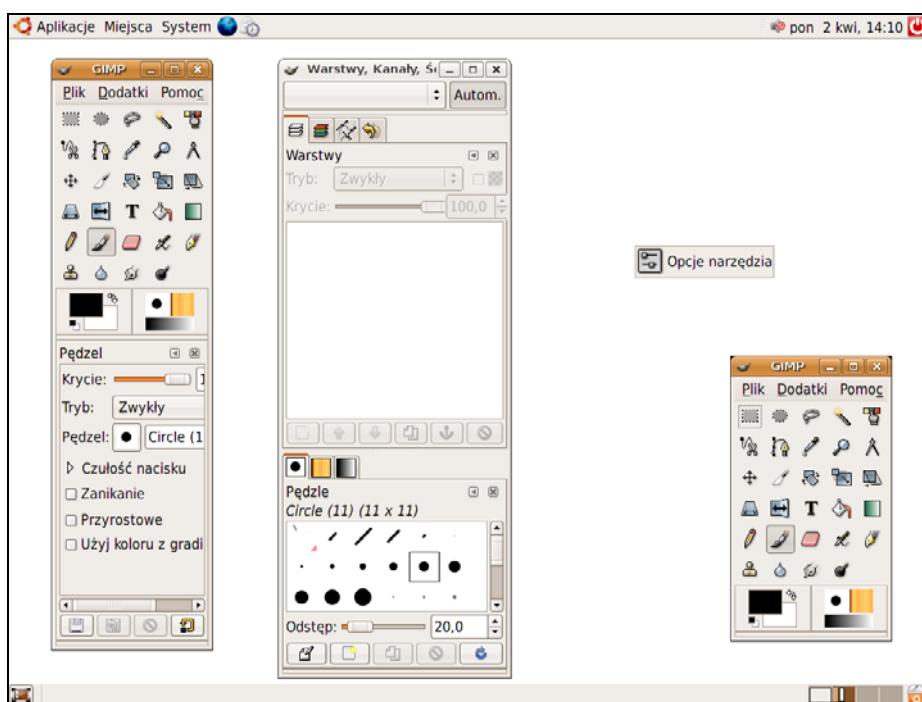
- ♦ **Główne okno narzędziowe:** okno znajdujące się po lewej stronie zawiera dwa inne: główne okno narzędziowe, a poniżej okno opcji. To pierwsze umożliwia dostęp do wszystkich poleceń i okien programu GIMP; w drugim oknie użytkownik może konfigurować aspekty działania narzędzia zaznaczonego w głównym oknie narzędziowym wyświetlanym powyżej.
- ♦ **Okno warstwy i pędzle:** okno znajdujące się po prawej stronie zawiera dwa inne: *Warstwy*, *Kanale*, *Ścieżki*, a poniżej *Pędzle*, *Desenie*, *Gradientsy*. Rzeczą, na którą warto zwrócić uwagę w tym oknie, jest fakt, że każde okno zawiera w rzeczywistości

kilka innych okien, które zostały zablokowane razem jako zakładki. W górnym oknie można znaleźć następujące zakładki: *Warstwy*, *Kanały*, *Ścieżki* i *Cofnięcie*; natomiast w dolnym są to odpowiednio: *Pędzle*, *Desenie* i *Gradienty*.



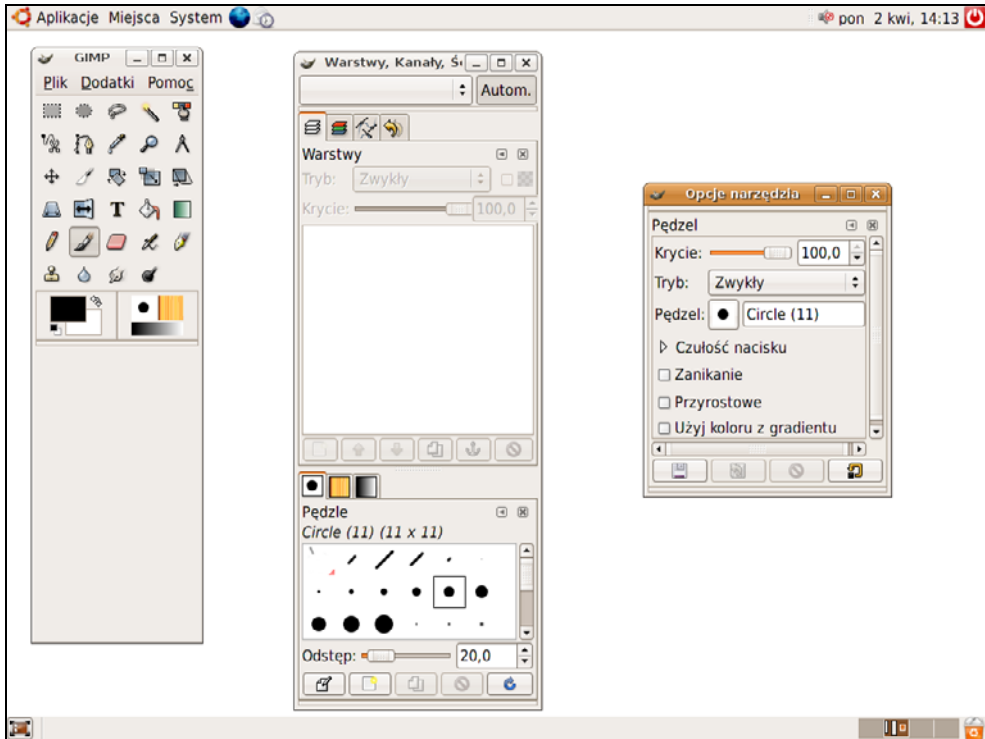
Główne okno narzędziowe programu musi być cały czas widoczne, jego zamknięcie spowoduje wyłączenie GIMP-a. Większość użytkowników uważa, że bardzo wygodne jest również stałe wyświetlanie okna *Warstwy*, dzięki czemu mają stały i łatwy dostęp do różnych elementów obrazu rozmieszczonych na różnych warstwach.

Dokowanie i odłączanie okien to bardzo prosta czynność. Aby odłączyć wybrane okno, należy za pomocą kursora myszy przeciągnąć pasek tytułowy poza okno, w którym jest ono aktualnie wyświetlane. Za kursorem będzie wyświetlana ikona identyfikująca odłączane okno, co widać na rysunku 12.3, na którym okno *Opcje narzędzia* jest przenoszone poza okno widoczne na rysunku 12.2.



Rysunek 12.3. Odłączanie okna dialogowego programu GIMP

Kiedy kursor zostanie przesunięty w miejsce, w którym użytkownik chce umieścić okno, wystarczy zwolnić przycisk myszy, a okno zostanie wyświetlone (patrz rysunek 12.4). Okno dialogowe *Opcje narzędzia* jest teraz wyświetlane osobno, dzięki czemu część głównego okna narzędziowego, która poprzednio była zajęta, teraz została zwolniona. Użytkownik może zmienić rozmiar nowo odłączonego okna, aby lepiej wykorzystać dostępną przestrzeń ekranu; w tym celu wystarczy chwycić za pomocą kursora myszy róg okna i dostosować wielkość.



Rysunek 12.4. Odlączone okno dialogowe *Opcje narzędzia* w programie GIMP

Podstawowe operacje wykonywane w programie GIMP

Jak wspominałem wcześniej, nie jestem artystą grafikiem. Wielu użytkowników GIMP-a, którzy są prawdziwymi grafikami, napisało sporo różnych poradników dla korzystających z programu i obraziłbym ich (a także czytelników), gdybym próbował powtarzać ich działania czy udawał, że jestem wykwalifikowanym grafikiem. Używam GIMP-a do szeregu różnych czynności, z których kilka okaże się — być może — interesujących dla czytelnika. Więcej informacji na ten temat można znaleźć poniżej. Adresy poradników do programu GIMP można znaleźć w punkcie „Dodatkowe informacje o GIMP-ie”.

Wykonywanie zrzutów ekranu

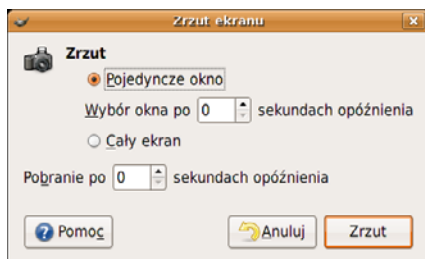
Jak czytelnicy zapewne zauważyli, w książce zamieszczono sporo zrzutów ekranu obejmujących cały ekran, wybrane okno czy określone części ekranu. Mój przyjaciel grafik ostatecznie przekonał mnie, że czasem obraz wart jest tysiąca słów, a przynajmniej powinien im od czasu do czasu towarzyszyć. Wszystkie zrzuty ekranu zamieszczone w tej książce zostały wykonane za pomocą programu GIMP, który okazał się doskonałym narzędziem do tego typu czynności. Czytelnicy mogą sądzić, że nigdy nie przyda im się umiejętność wykonywania zrzutów ekranu, ale to błąd. Zrzut ekranu to doskonałe rozwiązanie, kiedy trzeba np. zgłosić błąd programu lub zachować zawartość ekranu dla przyjaciela, aby później mu ją pokazać itd.

Wykonywanie zrzutów ekranu w GIMP-ie jest bardzo proste.

1. Z menu programu należy kolejno wybrać *Plik/Pobierz/Zrzut ekranu*, zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 12.5.

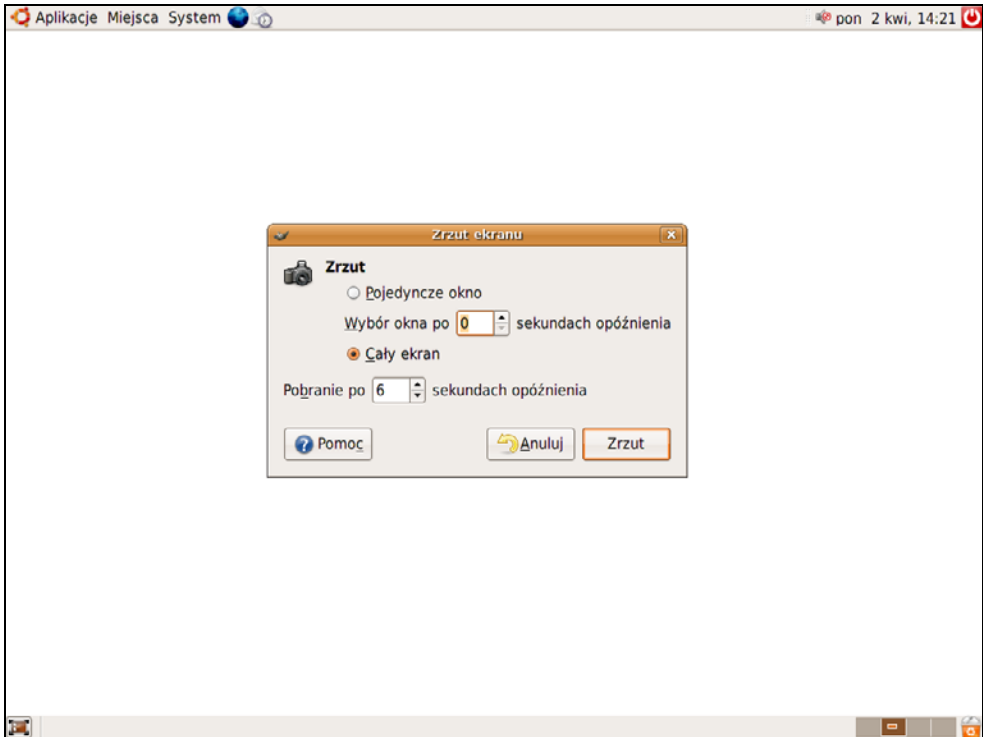
Rysunek 12.5.

Okno dialogowe pobierania zrzutów ekranu



2. Następnie należy zaznaczyć odpowiednią kontrolkę zgodnie z tym, co ma zostać uchwycone: jedno okno czy cały ekran.
3. Określić opóźnienie, po jakim zostanie wykonany zrzut całego ekranu lub tylko pojedynczego okna. Odpowiednie opóźnienie pozwoli użytkownikowi rozwinąć potrzebne menu, ukryć inne okna dialogowe itd. — jednym słowem, dostosować ekran lub okno dialogowe tak, aby na zrzucie znalazło się dokładnie to, co chciał tam umieścić.
4. Teraz trzeba wcisnąć przycisk *Zrzut*, proces pobierania obrazu zostanie rozpoczęty. Po ustawionym wcześniej opóźnieniu zrzut zostanie pobrany; jeżeli użytkownik nie ustawił żadnego opóźnienia, wówczas zrzut zostanie pobrany natychmiast po wciśnięciu przycisku. Jeżeli użytkownik chce wykonać zrzut pojedynczego okna, to po zaznaczeniu odpowiedniej opcji należy przesunąć wyświetlony krzyżyk nad odpowiednie okno i wcisnąć przycisk myszy. Po pobraniu zrzutu ekranu program wyświetli okno dialogowe (patrz rysunek 12.6), na którym widoczny będzie fragment ekranu.
5. Po wykonaniu zrzutu ekranu kolejnym krokiem jest zapisanie pliku w bezstratnym formacie, tak aby można go było dalej używać. Aby zapisać plik ze zrzutem ekranu, należy z menu okna wybrać opcje *Plik/Zapisz* (można też użyć opcji *Plik/Zapisz jako*), zostanie wówczas wyświetlone kolejne okno dialogowe (patrz rysunek 12.7).
6. Okno to umożliwia zapisanie pliku w wybranym formacie graficznym, określanym przez rozszerzenie pliku. Po nadaniu plikowi odpowiedniej nazwy oraz wyborze formatu graficznego należy kliknąć przycisk *Zapisz*, a zostanie wyświetlone jeszcze jedno okno dialogowe (patrz rysunek 12.8).
7. Dostępne opcje zależą od wybranego formatu zapisu pliku, dlatego wygląd okna przedstawionego na rysunku 12.8 może się zmieniać. Po wprowadzeniu zmian w ustawieniach zapisu pliku należy jeszcze kliknąć przycisk *OK*.

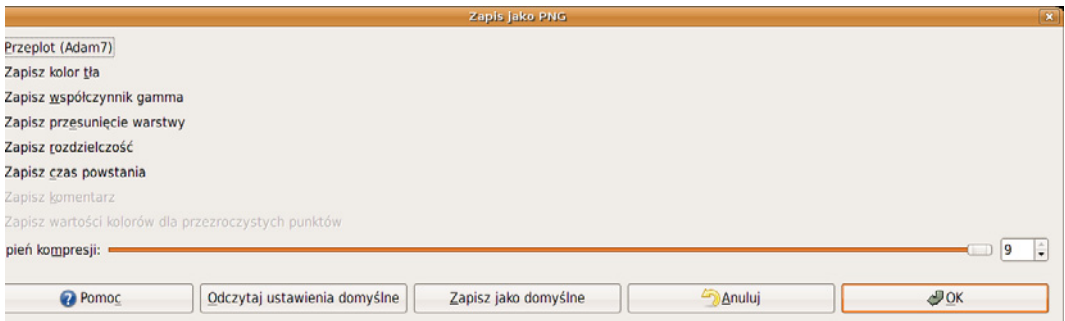
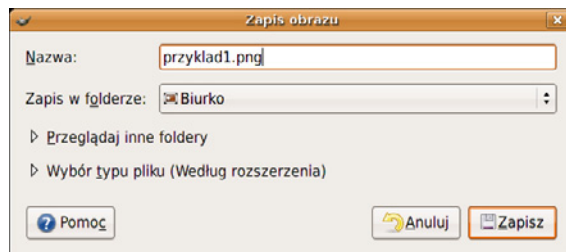
GIMP nie tylko obsługuje wiele różnorodnych formatów graficznych, ale też umożliwia łatwe wykonywanie zrzutów ekranu o wysokiej jakości oraz dostosowywanie parametrów tworzonych plików.



Rysunek 12.6. Zrzut ekranu wykonany w programie GIMP

Rysunek 12.7.

*Okno dialogowe
Zapisz jako*



Rysunek 12.8. Ustawienia szczegółowe podczas zapisywania pliku

Konwertowanie i zmniejszanie obrazów za pomocą programu GIMP

Dziś wielu użytkowników komputerów posiada własne strony internetowe, które służą do wymiany informacji i zdjęć ze znajomymi. Większość aparatów cyfrowych zapisuje zdjęcia w formacie JPEG, który oferuje doskonałą równowagę pomiędzy małym rozmiarem plików (tak aby obrazy pobierane ze stron internetowych znalazły się na dysku komputera za życia użytkownika) a dobrą jakością (dzięki czemu fotoamator uniknie podejrzenia, że nie umie posługiwać się swoim aparatem). Nadchodzą czasy, w których użytkownicy coraz częściej będą mieli pliki graficzne tych rozmiarów, że będą chcieli zmniejszać ich rozmiary i konwertować do innych formatów. Przykładem może być format TIFF, w którym najczęściej zapisywane są obrazy uzyskiwane ze skanerów. Pliki w tym formacie nie mogą być wyświetlane na stronach internetowych, poza tym są zbyt duże, aby je tam umieszczać. Podobnie jest w dzisiejszych aparatach cyfrowych, które produkują zdjęcia w megapikselach: świetne rozwiązanie w przypadku druku, ale zabójcze, gdy chodzi o strony internetowe.

Ponieważ GIMP obsługuje tak wiele różnorodnych formatów graficznych, jest to jednocześnie doskonale narzędzie do wykonywania konwersji pomiędzy różnymi formatami (np. w celu umieszczenia pliku na stronie internetowej) czy też do zmniejszenia rozmiarów pliku bez nadmiernej ingerencji w jego jakość. W ramce zatytułowanej „Formaty graficzne obsługiwane w programie GIMP” zamieszczam informacje na temat wszystkich plików graficznych, które można edytować w programie GIMP.

Oto kilka przydatnych wskazówek, które można wykorzystać podczas zmniejszania rozmiarów plików graficznych, dzięki czemu można je łatwo umieszczać na stronach internetowych, a nawet wysyłać pocztą elektroniczną.

- ♦ Warto zapisywać pliki w stralnych formatach. Choć PNG jest atrakcyjnym wyborem ze względu na obsługę 48-bitowej głębi koloru, to doświadczenie wskazuje, że najbardziej przyjaznym formatem do wymiany między użytkownikami jest jednak JPEG (w dodatku jest obsługiwany przez wszystkie przeglądarki internetowe). Należy się tylko upewnić, że nowy plik będzie zapisywany pod nową nazwą, tak aby plik oryginalny pozostał bez zmian, gdyby użytkownik chciał go wydrukować czy edytować ponownie.
- ♦ Podczas edytowania pliku JPEG warto poeksperymentować ze stopniem kompresji. Zwykle nie będziesz w stanie dostrzec różnicy, podobnie jak inni oglądający. Aby obniżyć jakość pliku JPEG, należy z menu wybrać opcję *Plik/Zapisz jako*, wprowadzić nową nazwę dla pliku i kliknąć przycisk *Zapisz*. Zostanie wyświetlone okno dialogowe widoczne na rysunku 12.9. Domyślna ustawiona wartość to 85, poniżej znajduje się informacja o rozmiarze pliku. Należy zaznaczyć opcję *Podgląd w oknie obrazu*, teraz można już przesuwając suwak, sprawdzając jednocześnie zmiany w obrazie oraz zmiany rozmiaru. Kiedy zostaną już znalezione odpowiednie wartości, wystarczy wcisnąć przycisk *OK*.
- ♦ Wskazówka dla twórców stron internetowych: rozmiar plików można zmniejszyć z tego, w jakim zostały np. zeskanowane do takiego, w jakim będą widoczne na ekranie monitora. Aby zmienić rozmiar obrazu, należy wybrać z menu programu

Formaty graficzne obsługiwane w programie GIMP

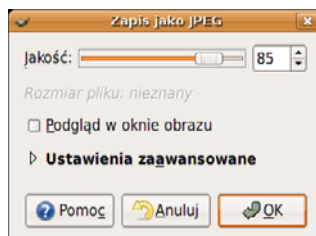
GIMP umożliwia zapisywanie plików graficznych w wielu różnorodnych formatach określanych przez rozszerzenia nadawane plikom. Oto one:

- ♦ Adobe Photoshop, obraz — *.psd*,
- ♦ Alias/Wavefront PIX obraz — *.pix*, *.mask*, *.matte*, *.alpha*, *.als*,
- ♦ ASCII, kod — *.txt*, *.ansi*, *.text*,
- ♦ Autodesk FLIC, animacje — *.fli*, *.flc*,
- ♦ Bzpip, archiwum — *.xcf*, *.bz2*, *.xcfbz2*,
- ♦ C, kod źródłowy — *.c*,
- ♦ C, nagłówki kodu, — *.h*,
- ♦ Digital Image and Communication in Medicine, obraz — *.dcm*, *.dicom*,
- ♦ Encapsulated PostScript, obraz — *.eps*,
- ♦ Flexible Image Transport System — *.fit*, *.fits*,
- ♦ GIF, obraz — *.gif*,
- ♦ GIMP Brush, pędzel — *.gbr*,
- ♦ GIMP Brush, pędzel animowany — *.gih*,
- ♦ GIMP skompresowany XJT, obraz — *.xjt*, *.xjtgz*, *.xjtbz2*,
- ♦ GIMP deseń — *.pat*,
- ♦ GIMP XCF obraz — *.xcf*,
- ♦ Gzip archiwum — *.xcf*, *.gz*, *.xcfgz*,
- ♦ HTML — *.htm*, *.html*,
- ♦ JPEG, obraz — *.jpg*, *.jpeg*, *.jpe*,
- ♦ KISS CEL — *.kiss*,
- ♦ plik ikon Microsoft Windows — *.ico*,
- ♦ MNG, animacje — *.mng*,
- ♦ PGM, obraz — *.pgm*,
- ♦ PNG, obraz — *.png*,
- ♦ PostScript, dokument — *.ps*,
- ♦ PPM, obraz — *.ppm*,
- ♦ Silicon Graphic IRIS, obraz — *.sgi*, *.rgb*, *.bw*, *.icon*,
- ♦ TarGA, obraz — *.tga*,
- ♦ TIFF, obraz — *.tif*, *.tiff*,
- ♦ Windows BMP, obraz — *.bmp*,
- ♦ X Bitmap, obraz — *.xbm*, *.icon*, *.bitmp*,
- ♦ X Pixmap, obraz — *.xpm*,
- ♦ X Window — *.xwd*,
- ♦ ZsoftPCX, obraz — *.pcx*, *.pcc*.

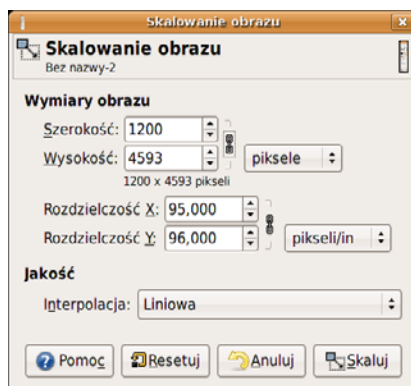
Obraz/Skaluj obraz, zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 12.10. Wystarczy wprowadzić tylko szerokość, po kliknięciu przycisku *Skaluj* wysokość zostanie obliczona automatycznie z zachowaniem proporcji oryginału.

Rysunek 12.9.

Obniżanie jakości
pliku JPEG podczas
zapisywania pliku

**Rysunek 12.10.**

Dostosowywanie
rozmiaru obrazu
do wymiarów
strony internetowej



Za pomocą tych prostych technik udało mi się zmniejszyć rozmiar zdjęcia wykonanego aparatem cyfrowym z 500 kB do 50 bez wyraźnej straty jakości. Być może ktoś z lepszym wzrokiem dostrzegłby różnice — ja ich nie widzę.



W Ubuntu dostępne są także narzędzia linii poleceń, dzięki którym można edytować pliki graficzne. Jednym z moich ulubionych poleceń jest `convert`, przy jego użyciu można zredukować głębokość kolorów obrazu. To narzędzie jest jednym z wielu dostępnych w pakiecie ImageMagick instalowanym domyślnie w systemie Ubuntu. Więcej informacji na temat narzędzia `convert` oraz pakietu ImageMagick można uzyskać, korzystając z polecenia `man`.

GIMP posiada bardzo dużo pożytecznych funkcji. Powyżej zaledwie drasnąłem powierzchnię, pod którą kryje się całe morze możliwości, wystarczy tylko zdobyć się na odwagę i rozpocząć eksplorację. Niżej zamieszczam odnośniki do przewodników, instrukcji i samouczków ułatwiających korzystanie z programu. Zwłaszcza samouczki są znakomitą sposobem osiągnięcia zamierzonych celów i obserwowania dokonywanych postępów.

Dodatkowe informacje o GIMP-ie

W internecie można znaleźć setki stron poświęconych programowi GIMP, oto kilka moich ulubionych witryn.

- ♦ www.gimp.org: strona domowa programu GIMP. Ponieważ jest to statek-matka programu, to jednocześnie najlepszy punkt startowy, aby sprawdzać aktualną dokumentację, upewnić się, że używa się najnowszej wersji aplikacji, wyszukać samouczki oraz znaleźć dalsze odnośniki do kolejnych stron.

- ♦ <http://emyrean.lib.ndsu.nodak.edu/~nem/gimp/tuts>: Nem W. Slecht zebrał na jednej stronie listę łączy do ogromnej kolekcji samouczków do GIMP-a poświęconych różnym zagadnieniom.
- ♦ <http://gimp-savvy.com/BOOK/index.html>: doskonała, choć już nieco wiekowa książka poświęcona programowi GIMP, dostępna w sieci. Choć nie znajdzie się w niej informacji o najnowszych funkcjach i możliwościach programu, wciąż jest znakomitą podręcznikiem, warto ją posiadać, jeżeli użytkownik planuje intensywnie wykorzystywać GIMP-a.
- ♦ <http://gimps.de/en/tutorials/gimp/picture-photo-image/index.htm>: świetny zbiór samouczków dla GIMP-a poświęconych retuszowi fotografii.



Miłośnicy fotografii cyfrowej powinni zapoznać się z pakietem Bibble (www.bibble.com). Bibble dostępny jest w dwóch wersjach: Lite i Professional (kosztują odpowiednio 70 i 130 dolarów). Ponieważ jest to komercyjny program, nie jest instalowany w Ubuntu. Ten pakiet to kolejny przykład rosnącego wyboru komercyjnego oprogramowania dostępnego dla Linuksa, dzięki czemu staje się on równorzędnym systemem operacyjnym dla użytkowników domowych.

Używanie programu OpenOffice.org Draw

OpenOffice.org (www.openoffice.org) to pakiet aplikacji biurowych, na który składają się: edytor tekstu Writer, arkusz kalkulacyjny Calc, program do tworzenia prezentacji Impress, baza danych dBase oraz program graficzny Draw. W poprzednich rozdziałach omówiłem inne elementy pakietu, ogólne informacje o samym pakiecie podałem w podrozdziale rozdziału 10., zatytułowanym „Edycja tekstu w programie OpenOffice.org Writer”, dalszą część tego rozdziału poświęcam programowi Draw.

Głównym przeznaczeniem programu Draw jest tworzenie rysunków w formacie pakietu OpenOffice.org (.odg). Podobnie jak Inkscape (również omówiony w tym rozdziale), w programie Draw grafiki tworzy się w formacie SVG, choć jego zdolność do importowania plików w tym formacie jest ograniczona.

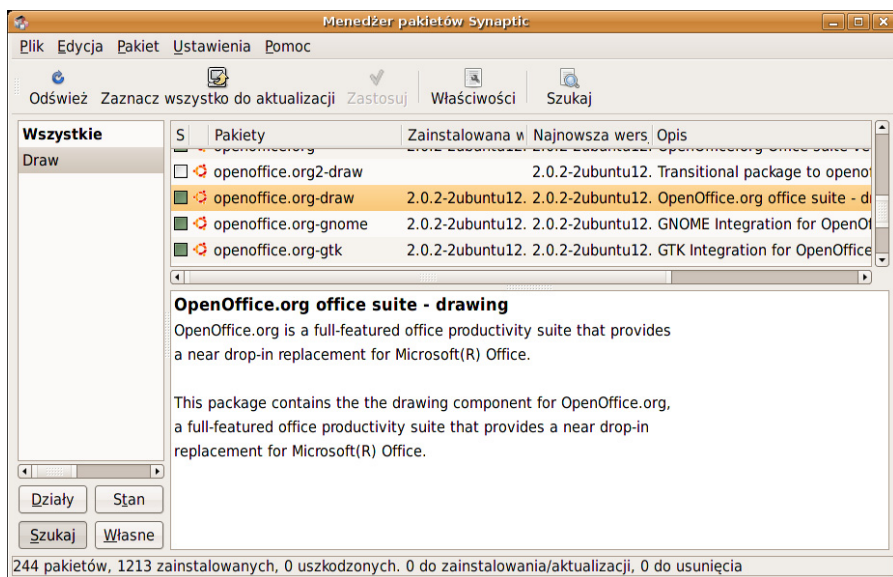
Aby nie powtarzać za każdym razem długiej nazwy OpenOffice.org Draw, w dalszej części używana będzie skrótowa, choć poprawna nazwa — Draw. Pojawienie się nazwy oodraw oznaczać będzie odwołanie do polecenia, za którego pomocą można uruchomić program z linii poleceń.

Instalowanie programu OpenOffice.org Draw

Draw jest instalowany domyślnie, jeżeli Ubuntu zostało zainstalowane z płyty Live CD, Alternate Install CD z wykorzystaniem trybu tekstowego lub trybu OEM. Program zostanie również zainstalowany automatycznie, jeżeli w systemie umieszczono program Impress.

Aby zainstalować program Draw, należy uruchomić program Synaptic (menu *System/Administracja*), następnie wybrać przycisk *Szukaj* i wpisać w odpowiednim polu Draw.

Kiedy wyszukiwanie zostanie zakończone, należy na wyświetlonej liście znaleźć pozycję *OpenOffice.org Draw*, następnie wcisnąć prawy przycisk myszy i z menu kontekstowego wybrać opcję *Instaluj* (patrz rysunek 12.11).



Rysunek 12.11. Instalowanie programu *OpenOffice.org Draw* za pomocą menedżera pakietów *Synaptic*

Zależnie od wcześniej zainstalowanego oprogramowania w systemie, może się okazać, że konieczne będzie zainstalowanie dodatkowych pakietów, które zostaną wskazane przez *Synaptic*.

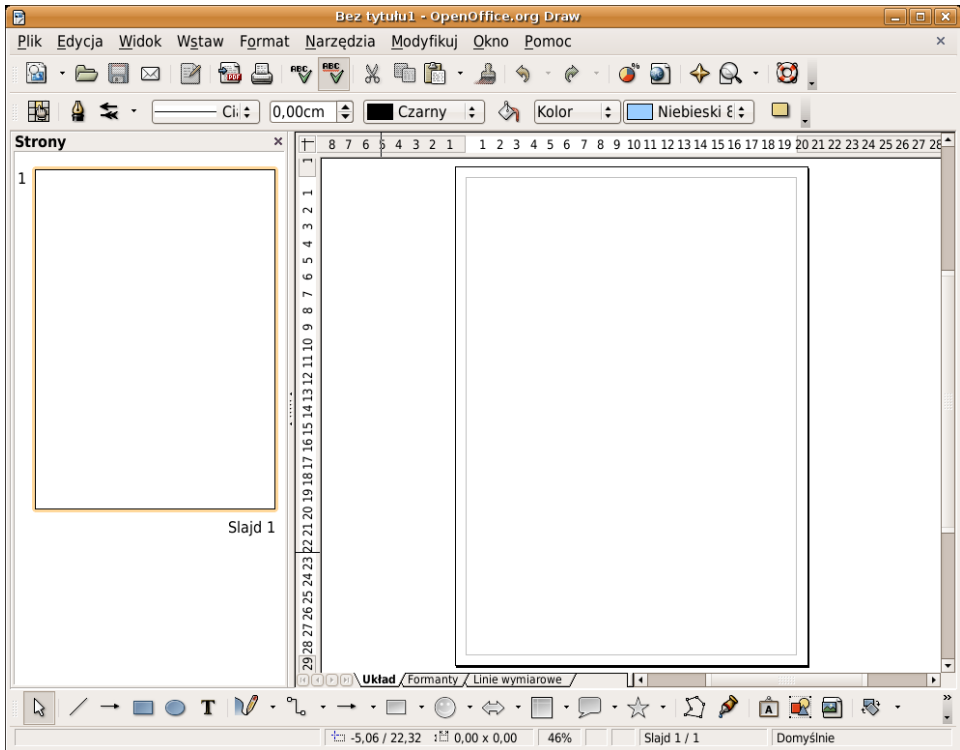
Po zaznaczeniu wszystkich niezbędnych pakietów należy wcisnąć przycisk *Zastosuj*, a program *Draw* zostanie zainstalowany w systemie. Program po zainstalowaniu jest gotowy do uruchomienia.

W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, zamieszczam więcej informacji na temat instalowania oprogramowania za pomocą menedżera pakietów *Synaptic* czy narzędzi do zarządzania oprogramowaniem *apt-get* i *aptitude*.

Uruchamianie programu *Draw*

W Ubuntu 6.06 LTS program *Draw* nie jest automatycznie dodawany do menu, dlatego aby go uruchomić, należy w terminalu wpisać polecenie *oodraw*, a następnie wcisnąć klawisz *Enter*. Można dodać nową pozycję do menu, korzystając z programu *Alacarte*, więcej informacji na temat jego używania zamieściłem w rozdziale 5.

Po uruchomieniu programu zostanie wyświetlone okno widoczne na rysunku 12.12.



Rysunek 12.12. Program OpenOffice.org Draw

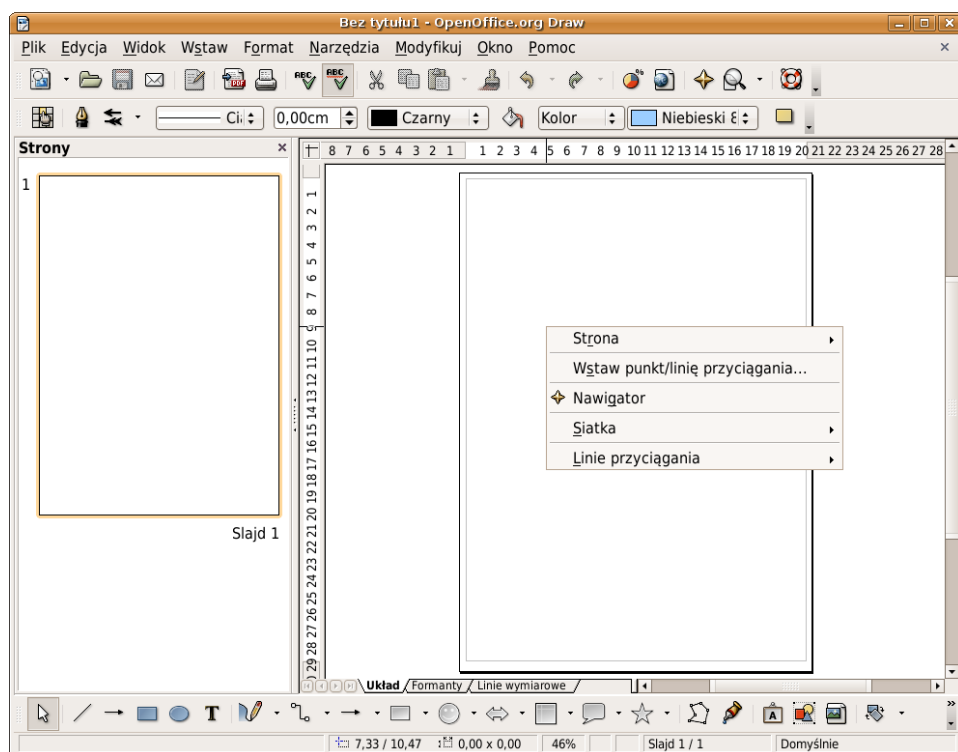
Przewodnik po programie Draw

Ponieważ program OpenOffice.org Draw wyposażony jest w doskonały system pomocy, nie będę zanudzać czytelnika opisem zawartości każdego menu. Zamiast tego zamieszczam tylko krótkie omówienie interfejsu programu widocznego na rysunku 12.12.

- ♦ **Menu:** rozwijane menu są zorganizowane tak, jak w większości programów tego typu, dzięki czemu użytkownicy nie powinni mieć kłopotów z korzystaniem z nich. Pozycje menu, przy których prawej krawędzi znajdują się strzałki, prowadzą do kolejnych podmenu.
- ♦ **Paski narzędzi:** domyślnie w programie są wyświetlane następujące paski narzędzi: *Standardowy*, *Linie i wypełnienia*, *Rysunek*, które za pośrednictwem znajdujących się na nich ikon umożliwiają wykonywanie szeregu operacji. Korzystając ze strzałek umieszczonych przy prawej krawędzi pasków narzędzi, użytkownik może dopasować ich zawartość do swoich potrzeb. Aby włączyć lub wyłączyć wybrany pasek narzędzi, należy użyć menu *Widok/Paski narzędzi*. W programie Draw dostępnych jest wiele pasków narzędzi ukierunkowanych na obsługę procesów graficznych; np. do obsługi obiektów trójwymiarowych, dzięki któremu można łatwo tworzyć i modyfikować trójwymiarowe obrazy.
- ♦ **Linijki:** za pomocą linijek można określić pozycję kursora w płaszczyznach poziomej i pionowej. Szare części każdej linijki odpowiadają obszarowi

zajmowanemu przez marginesy strony. Wyświetlanie linijek można wyłączyć, wystarczy z menu wybrać opcję *Widok/Linijka*.

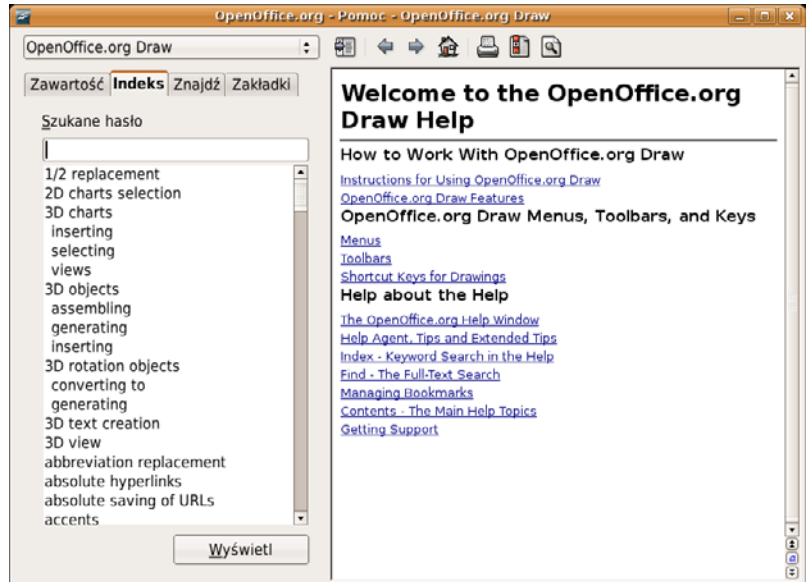
- ♦ **Okno Style i formatowanie:** chociaż okno *Style i formatowanie* nie jest wyświetlane domyślnie, można je przywołać w każdej chwili, wciskając klawisz *F11* (aby wyłączyć okno, należy powtórnie wcisnąć ten sam klawisz). Umożliwia ono łatwy dostęp do stylów dostępnych w edytowanym rysunku i jest szczególnie użyteczne, jeżeli nie jest wyświetlany pasek narzędziowy *Formatowanie*. Jak każde pływające okno, można je zakotwiczyć w głównym oknie, klikając dwukrotnie w szary obszar z nazwą. Aby okno odłączyć od głównego okna programu, należy dwukrotnie kliknąć nagłówek zadokowanego obszaru.
- ♦ **Menu kontekstowe:** po umieszczeniu kursora na tle dowolnego rysunku edytowanego w programie Draw wystarczy wcisnąć prawy przycisk myszy, zostanie wówczas wyświetlone menu kontekstowe widoczne na rysunku 12.13. Umożliwia ono łatwy dostęp do kilku podstawowych operacji związanych z grafiką, takich jak zmiana formatu strony, wyświetlanie siatki czy linii przyciągania.



Rysunek 12.13. Menu kontekstowe programu OpenOffice.org Draw

Program Draw jest też wyposażony w obszerny system pomocy, wystarczy wybrać z menu *Pomoc/OpenOffice.org* lub wcisnąć klawisz *F1*. Na rysunku 12.14 widać zawartość okna z pomocą dla programu Draw. Można w nim znaleźć wyszukiwarkę słów kluczowych, listę tematów oraz szczegółową zawartość wybranego zagadnienia (po prawej stronie okna).

Rysunek 12.14.
Pomoc programu
OpenOffice.org Draw



Dodatkowe informacje o programie OpenOffice.org Draw

W miarę, jak pakiet OpenOffice.org staje się coraz bardziej popularny, przybywa stron internetowych, na których można znaleźć darmową pomoc dla każdego z jego składników. Poniżej zamieszczam kilka odnośników do moich ulubionych stron poświęconych programowi Draw.

- ♦ <http://www.oauthors.org/en/members/tutorials/draw> — można tu znaleźć podręczniki oraz dział FAQ (ang. *Frequently Asked Questions*) w kilku językach.
- ♦ <http://openoffice-support.net/#draw> — można tu znaleźć kilka podręczników oraz bardzo przydatne arkusze CLUE. Część informacji pochodzi z witryny firmy Sun (dotyczą pakietu StarOffice), ale mogą być przydatne również dla użytkowników OpenOffice’a.
- ♦ www.openoffice.org — statek-baza pakietu OpenOffice.org, na stronie można znaleźć listy dyskusyjne, przykłady dokumentów, dział FAQ i znacznie, znacznie więcej.

Inkscape — program do edycji grafiki wektorowej

Nim komputery na stałe zagościły w domach, a internet dostarczył platformy do wymiany oraz prezentacji tekstu i ilustracji, ustandaryzowane formaty dokumentów i plików graficznych nie były tak ważne. W poprzednich rozdziałach, w których omawiałem przeglądarkę internetową Firefox oraz edytor tekstu Writer, podkreślałem znaczenie i wartość otwartych standardów zapisu plików HTML czy tekstowych. Sytuacja wygląda podobnie w przypadku formatów plików graficznych: starsze formaty, takie jak JPEG czy GIF, są coraz częściej zastępowane przez otwarte formaty, np. PNG (ang. *Portable Network*

Graphics) czy SVG (ang. *Scalable Vector Graphics*) opracowany przez organizację W3C dla grafiki wektorowej (więcej informacji na temat standardu SVG można znaleźć na stronie www.w3.org/Graphics/SVG/).

Inkscape to program o otwartym kodzie służący do tworzenia i edytowania grafiki wektorowej. Jest podobny do tradycyjnie używanych programów, takich jak Adobe Illustrator, Aldus (Adobe) Freehand czy CorelDraw. Inkscape może importować wiele różnych formatów plików zawierających grafikę rastrową i wykorzystywanych jako składniki tworzonego obrazu SVG. Program może nawet eksportować pliki PNG tak, aby mogły być wyświetlane przez starsze przeglądarki internetowe obsługujące grafikę rastrową. Oprócz standardowej obsługi skalowalnej grafiki wektorowej, program przetwarza też metadane (Creative Commons) i pliki XML.



Omówiony wcześniej program OpenOffice.org Draw może eksportować pliki do formatu SVG, posiada też pewne możliwości importowania plików, jednak to właśnie Inkscape został zaprojektowany do obsługi grafiki SVG. Draw doskonale sprawdza się podczas przygotowywania grafiki na potrzeby innych komponentów pakietu OpenOffice.org. Jeżeli jednak użytkownik chce tworzyć ilustracje przeznaczone na strony internetowe, wówczas sugeruję użycie Inkscape'a.

Instalowanie programu Inkscape

Ponieważ nie wszyscy zajmują się tworzeniem grafiki wektorowej (a wielu użytkowników wykorzystuje do tego celu program Draw), Inkscape nie jest domyślnie instalowany w systemie Ubuntu. Na szczęście, to niedopatrzenie łatwo naprawić.

Aby zainstalować program w systemie, należy uruchomić menedżera pakietów Synaptic (menu *System/Administracja*), kliknąć przycisk *Szukaj* i w wyświetlonym oknie wpisać Inkscape, a następnie wcisnąć przycisk *Szukaj*. Po zakończeniu wyszukiwania należy zaznaczyć pozycję *Inkscape* (będzie to jedyna odnaleziona), wcisnąć prawy przycisk myszy i za pomocą menu kontekstowego zainstalować program (patrz rysunek 12.15).

System wyświetli listę dodatkowych pakietów, które muszą zostać zainstalowane, aby Inkscape mógł działać poprawnie. Należy potwierdzić ich pobranie.

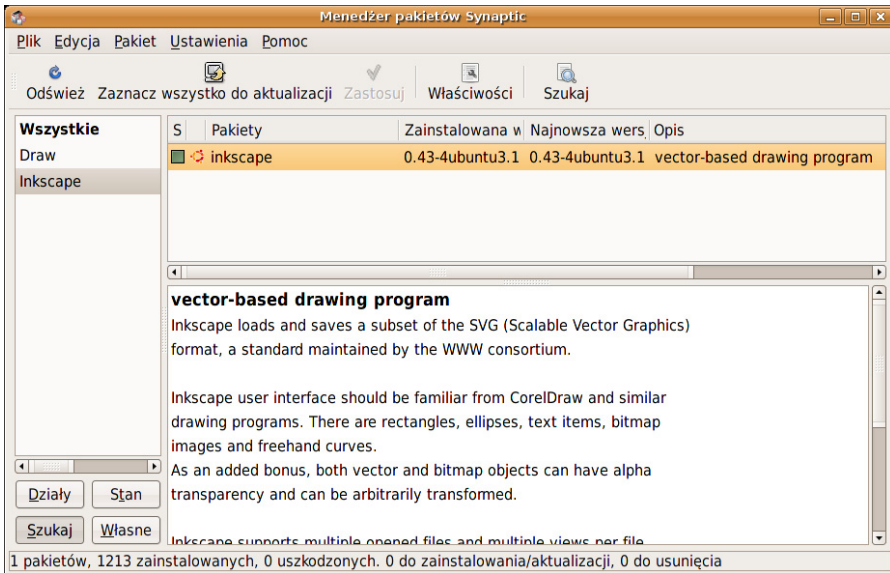
Po zakończeniu wybierania plików trzeba kliknąć przycisk *Zastosuj* znajdujący się na pasku narzędziowym Synaptica. Po zakończeniu procesu instalacji program jest gotowy do użytku. W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, zamieszczam dodatkowe informacje na temat korzystania z menedżera pakietów Synaptic.

Uruchamianie programu Inkscape

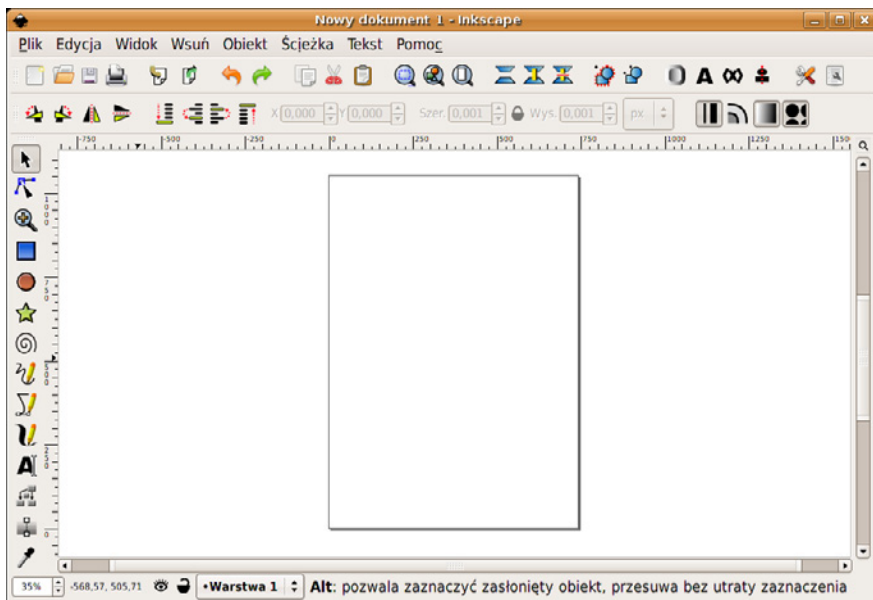
Program po zainstalowaniu można uruchomić na wiele różnych sposobów, dwa najpopularniejsze to:

- ♦ wybranie programu z menu *Aplikacje/Grafika*,
- ♦ wpisanie w terminalu polecenia `inkscape`.

Po uruchomieniu programu zostanie wyświetlone okno widoczne na rysunku 12.16.



Rysunek 12.15. Instalowanie programu Inkscape za pomocą menedżera pakietów Synaptic

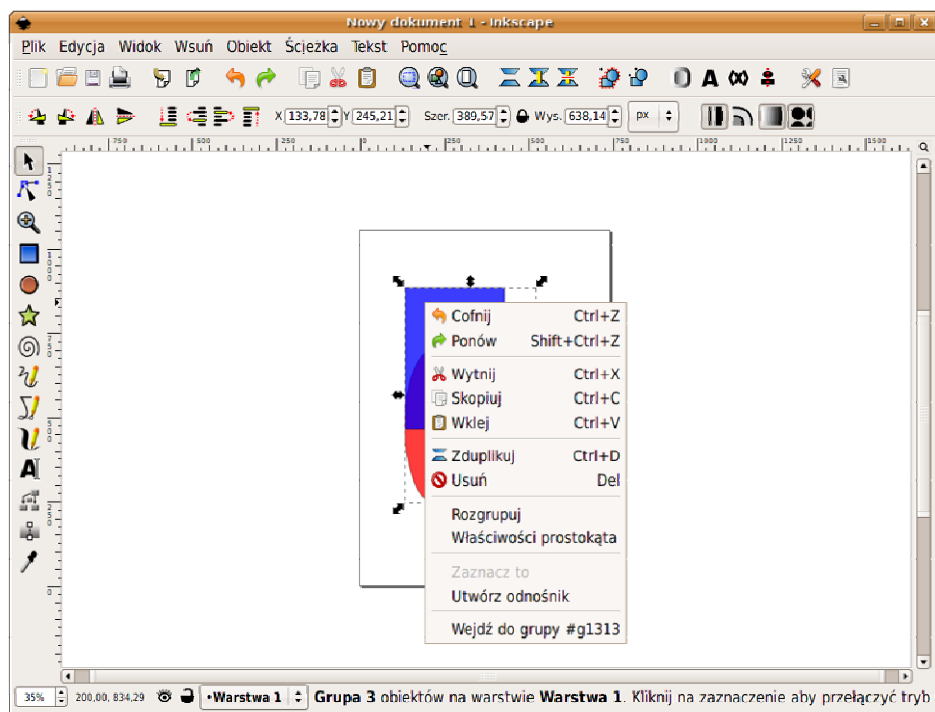


Rysunek 12.16. Główne okno programu Inkscape

Krótki przewodnik po programie Inkscape

Program Inkscape wyposażony jest w rozbudowany system pomocy, w którym można znaleźć nawet samouczki. Poniżej omawiam elementy interfejsu programu widoczne na rysunku 12.16.

- ♦ **Menu:** rozwijane menu programu Inkscape są zorganizowane tak, aby umożliwić jak najłatwiejszy dostęp do poleceń związanych z podstawowymi elementami grafik SVG, takimi jak warstwy, obiekty, ścieżki i tekst. Twórcy Inkscape wykonali znakomitą pracę, minimalizując ilość podmenu wykorzystywanych w programie — niemal każde polecenie jest dostępne z poziomu głównego menu.
- ♦ **Paski narzędzi:** w programie dostępne są dwa paski narzędziowe; górny umożliwia dostęp do standardowych poleceń służących do obsługi pliku czy zawartości schowka, a także do czynności, takich jak powiększanie, kopiowanie czy grupowanie wybranych obiektów. Dolny pasek ułatwia manipulowanie obrazem, obracanie, pracę z warstwami i skalowanie.
- ♦ **Linijki:** za pomocą linijek można określić pozycję kursora w płaszczyznach poziomej i pionowej.
- ♦ **Menu kontekstowe:** po umieszczeniu kursora na dowolnym rysunku edytowanym w programie Inkscape wystarczy wcisnąć prawy przycisk myszy, a zostanie wyświetlone menu kontekstowe widoczne na rysunku 12.17. Umożliwia ono łatwy dostęp do kilku podstawowych menu i operacji związanych z grafiką, takich jak kopiowanie zaznaczonych obiektów, grupowanie, cofanie wykonanej operacji i wyświetlanie właściwości zaznaczonego obiektu.



Rysunek 12.17. Menu kontekstowe programu Inkscape

Dodatkowe informacje o programie Inkscape

Najlepiej, kiedy o programie wypowiadają się zaawansowani graficy. Oto kilka moich ulubionych stron poświęconych programowi Inkscape, na których można znaleźć opinie ekspertów:

- ♦ www.inkscape.org — strona domowa projektu Inkscape, można tu znaleźć wiele różnorodnych zasobów i źródeł informacji o programie, takich jak listy dyskusyjne, artykuły, zbiory obrazów itp.,
- ♦ www.inkscape.org/mailling_lists.php — listy dyskusyjne poświęcone programowi Inkscape, jest to świetne miejsce do zadawania pytań i szukania odpowiedzi, niezależnie od stopnia zaawansowania i znajomości programu; ruch na listach nie jest szczególnie intensywny, ale warto je subskrybować, bo mamy dostęp do świeżych informacji na temat najnowszych wydań i poprawek do programu,
- ♦ <http://tavmjong.free.fr/INKSCAPE/MANUAL/html/> — Tavmjong Bah pisze książkę o programie Inkscape, która będzie dostępna za darmo w sieci; to wspaniała pozycja ze świetnie dobranym materiałem,
- ♦ www.angelfire.com/mi/kevincharles/inkscape/index.html — nieoficjalny (choć niezwykle przydatny) podręcznik do programu Inkscape napisany przez Kevina Wixsona,
- ♦ www.openclipart.org/ — strona zawierająca ogromny zbiór wolno dostępnych grafik. Większość z nich zapisana jest w formacie SVG, ale można też znaleźć pliki PDF i WMF.

Jak to zwykle bywa w przypadku popularnych programów o otwartych źródłach, kolejne strony poświęcone danej aplikacji pojawiają się zapewne w chwili, gdy piszę te słowa. Jeżeli czytelnicy odkryją lokalizacje, w których znajdują się interesujące informacje, proszę mnie o tym poinformować. Inkscape to naprawdę świetny program!

Podsumowanie

W niniejszym rozdziale omówiłem dostępne w Linuksie narzędzia do tworzenia i edytowania plików graficznych zapisanych w najróżniejszych formatach. W Linuksie można znaleźć doskonały wybór wydajnych programów graficznych, z których można skorzystać w codziennej pracy, jedynym ich ograniczeniem jest wyobraźnia użytkownika. GIMP jest doskonałym przykładem, jak potężne może być oprogramowanie o otwartych źródłach.

W rozdziale 13. zamieszczam informacje na temat obsługi dźwięku i obrazu w Ubuntu: poczynawszy od konfigurowania systemu dźwiękowego, na obsłudze płyt CD i DVD kończąc. Czytelnicy znajdą tam również podstawowe informacje na temat formatów plików audio i wideo, które można spotkać dziś w sieci.

Rozdział 13.

Multimedia

W tym rozdziale:

- ♦ Słownik terminologii
- ♦ Konfigurowanie dźwięku w Ubuntu
- ♦ Instalowanie kodeków
- ♦ Odtwarzanie, rzucanie i wypalanie płyt CD
- ♦ Praca z plikami audio i korzystanie z radia internetowego
- ♦ Odtwarzanie, rzucanie i wypalanie płyt DVD

MP3, WMA, OGG, FLAC, SHN, APE — brzmi to jak seria z karabinu maszynowego, ale naprawdę jest to lista najpopularniejszych obecnie formatów cyfrowych plików dźwiękowych. Podobnie w dziedzinie cyfrowego wideo królują pojęcia, takie jak MPG, MPEG czy VOB. Są też wyjątkowo brzydkie skróty, takie jak DRM, MPAA czy RIAA, służące do określania pewnych organizacji, które próbują ograniczyć, z korzyścią dla siebie, prawa konsumentów, ale tego zagadnienia nie będę szerzej omawiać w tym rozdziale.

Wspaniałym aspektem cyfrowych plików audio i wideo jest fakt, że już nie potrzeba specjalnych urządzeń, aby odsłuchać muzykę czy obejrzeć film. Płyty CD i DVD można odtwarzać wprost na komputerze. Po zakupie nośnika CD czy DVD można zupełnie legalnie przenieść jego zawartość na dysk komputera i odtwarzać ją już bez obecności płyty w napędzie. To eliminuje konieczność noszenia ze sobą oryginalnych nośników, które mogą zostać w domu i służyć jako kopia zapasowa w razie awarii komputera.

W tym rozdziale mam zamiar pomóc czytelnikowi w zorientowaniu się w gąszczu skrótów i terminów używanych do opisywania plików audio i wideo, a także wyjaśnić, jak używać oprogramowania dostępnego w Ubuntu do odtwarzania muzyki, filmów, a nawet tworzenia własnych płyt audio CD i DVD. Dużą wygodą jest relaks w pokoju hotelowym przy ukochanej muzyce odtwarzanej za pomocą ulubionego narzędzia multimedialnego, czyli komputera z systemem Ubuntu.

Przegląd terminologii audio i wideo

Jak wspominałem we wstępie do niniejszego rozdziału, cyfrowe audio i wideo to obszar, w którym istnieje mocna tendencja do używania żargonu i trzyliterowych akronimów, co sprawia, że rozmowy i przekazywane informacje są niemal niedostępne dla nowicjuszy. Od wielu lat kolekcjonuję nagrania, płyty CD i w ogóle muzykę, a wciąż natrafiam na terminy, których znaczenie muszę sprawdzać, a trzeba pamiętać, że nie zawsze udaje się szybko odnaleźć znaczenie nowych pojęć.

Poniżej zamieszczam podręczny słownik terminologii związanej z cyfrowym audio i wideo, z którą często można się zetknąć. Nie ma potrzeby czytania wszystkich pojęć, mają one raczej służyć pomocą w trakcie lektury niniejszego rozdziału.

- ♦ **AAC (ang. *Avancer Audio Coding*)** — stratne kodowanie zaprojektowane przez Dolby Laboratories. Pliki w nim tworzone mają rozmiar zbliżony do plików w formacie MP3, ale oferują nieco lepszą jakość. AAC jest obecnie często — błędnie — kojarzone z formatem Apple Audio Codec, ponieważ firma Apple używa zabezpieczonej wersji AAC w celu narzucenia użytkownikom iTunes korzystania z technologii DRM.
- ♦ **ABR (ang. *Average Bit Rate*)** — typ kodowania, w którym użytkownik określa przepustowość dla próbkowania i odtwarzania, następnie program kodujący oblicza właściwą VBR (ang. *Variable Bit Rate*) konieczną do osiągnięcia przyjętej średniej wielkości liczby bitów informacji w określonym przedziale czasu.
- ♦ **AIFF (ang. *Audio Interchange File Format*)** — stratne kodowanie audio zaprojektowane przez Apple Computer i powszechnie używane w systemach Macintosh.
- ♦ **ALAC (ang. *Apple Lossless Audio Codec*)** — własnościowy zamknięty kodek audio zaprojektowany przez Apple Computer. Ten format jest często mylony z MP4, który z kolei może być pojemnikiem (*container*) dla formatu ALAC oraz innych.
- ♦ **APE** — rozszerzenie plików audio kodowanych bezstratnym, własnościowym kodekiem Monkey Audio. Kodek oferuje nieco lepszy stopień kompresji (czyli mniejsze pliki) niż formaty FLAC czy SHN, ale odbywa się to kosztem szybkości kodowania. Pliki APE są przede wszystkim używane do bezstratnego kodowania na starszych platformach Windows, a to dlatego, że nie jest dostępny żaden oficjalny port dla systemów linuksowych. Więcej informacji o formacie Monkey Audio można znaleźć na stronie www.monkeysaudio.com.
- ♦ **Bit rate** — liczba bitów informacji w określonym przedziale czasu tworząca sygnał audio. Najczęściej używanym przedziałem czasu jest sekunda (BPS, bity na sekundę).
- ♦ **CBR (ang. *Constant Bit Rate*)** — kodek wykorzystujący — w odróżnieniu do ABR i VBR — stałą ilość bitów na sekundę.
- ♦ **CD (ang. *Compact Disc*)** — nośnik zdolny do przechowywania 650 MB (w przybliżeniu) danych w formacie cyfrowym.

- ♦ **CDDA (ang. *Compact Disc Digital Audio*)** — format używany w większości płyt niezabezpieczonych przed kopiowaniem. Czasem używa się też skrótu CDA, ze względu na niezdolność używania czteroliterowych skrótów przez Microsoft.
- ♦ **CDDB (ang. *CD Database*)** — komercyjna, internetowa baza danych dostarczająca informacje identyfikujące ścieżki na komercyjnych płytach CD.
- ♦ **CD-R (ang. *Compact Disc/Recordable*)** — nośnik zdolny do przechowywania 650 MB (w przybliżeniu) danych w formacie cyfrowym, który w każdej swojej części może być jednokrotnie zapisany na komputerach użytkowników za pomocą odpowiednich nagrywarek. Zapisywanie danych na takiej płycie może odbywać się w wielu sesjach, podczas których dane nagrywane są na innej części dysku. Na stronie Mike'a Richtersa (<http://www.mrichter.com/cdr/primer/primer.htm>) można znaleźć ogromną ilość informacji na tematy związane z płytami CD-R (i nie tylko). Podobnym źródłem na temat oprogramowania do nagrywania płyt jest strona Andy'ego McFaddensa (www.cdrfaq.org). Strony, takie jak te, są prawdziwymi dowodami potęgi internetu, solidności oraz ukierunkowania na użytkowników. Czapki z głów przed ich autorami.
- ♦ **CD-RW (ang. *Compact Disc/Rewritable*)** — nośnik zdolny do przechowywania 650 MB (w przybliżeniu) danych w formacie cyfrowym, który może być wielokrotnie zapisywany w komputerach użytkowników za pomocą odpowiednich nagrywarek. Tylko napędy CD-RW mogą odczytywać dane z płyt CD-RW.
- ♦ **DAE (ang. *Digital Audio Extraction*)** — proces klonowania nagranych mediów cyfrowych poprzez bezpośrednie kopiowanie plików, a nie ich odtwarzanie; później następuje konwertowanie danych wyjściowych do formatu analogowego, a następnie zapisywanie na cyfrowym nośniku.
- ♦ **DAT (ang. *Digital Audio Tape*)** — nośnik wykorzystujący technologię cyfrową do zapisywania danych, używany w studiach nagraniowych oraz przez specjalistów.
- ♦ **DMCA (ang. *Digital Millenium Copyright Act*)** — rozwiązanie prawne przyjęte w Stanach Zjednoczonych, którego celem była ochrona praw autorskich, a które w rzeczywistości doprowadziło do zduszenia inżynierii wstecznej, wyeliminowania wolnych dyskusji na temat technologii szyfrowania danych oraz pozbawiło użytkowników możliwości pełnego korzystania z legalnie nabytych plików audio i wideo. Dobry pomysł, fatalne wykonanie.
- ♦ **DRM (ang. *Digital Rights Management*)** — technologia uniemożliwiająca legalnym użytkownikom swobodne kopiowanie i użytkowanie plików audio i wideo.
- ♦ **DSP (ang. *Digital Signal Processing*)** — korzystanie z różnych algorytmów do konwertowania i modyfikowania sygnału audio i wideo. DSP może być realizowane programowo, sprzętowo lub z wykorzystaniem obu tych opcji.
- ♦ **DVD (ang. *Digital Video Disc*)** — nośnik zdolny do przechowywania 4,7 GB (w przybliżeniu) danych w formacie cyfrowym.
- ♦ **EQ (ang. *Equalize*)** — efekt nagrywania powodujący zwiększanie lub zmniejszanie głośności różnych zakresów częstotliwości sygnału audio. Wyrównywanie zakresu częstotliwości wygładza dźwięk, eliminując krańcowe różnice w głośności.

- ♦ **FFP (ang. *FLAC Footprint*)** — suma kontrolna umożliwiająca zweryfikowanie spójności plików kodowanych w formacie FLAC. Więcej informacji na temat sum kontrolnych można znaleźć przy objaśnieniu terminu MD5.
- ♦ **FLAC (ang. *Free Lossless Audio Codec*)** — bezstratny kodek umożliwiający uzyskanie kompresji na poziomie 30% – 50% bez utraty jakości oryginalnego nagrania.
- ♦ **Formaty bezstratne** — formaty plików, w których zachowana jest jakość oryginału; innymi słowy, zachowana jest pełna zawartość w zakresie dźwięku i obrazu, które zostały oryginalnie nagrane czy umieszczone na nośniku.
- ♦ **Formaty stratne** — formaty plików, w których jakość poświęca się dla zredukowania rozmiaru pliku. Termin stratny odnosi się do algorytmów kompresji używanych do zmniejszania plików, podczas których część oryginalnych informacji zostaje usunięta, dzięki czemu plik wyjściowy (skompresowany) ma mniejsze rozmiary. Z pliku tak skompresowanego nie uda się odtworzyć pierwotnej zawartości pliku poddanego temu zabiegowi. W przypadku plików audio zwykle redukcji poddaje się przepustowość (bit rate) oraz zakres częstotliwości, w jakim dźwięk został nagrany; teoria jest taka, że większość słuchaczy i tak nie jest w stanie usłyszeć usuniętej zawartości.
- ♦ **Freedb (ang. *Free DB*)** — wolna alternatywa dla CDDb umożliwiająca użytkownikom wymieniać się informacjami o płytach CD.
- ♦ **ID3, znaczniki** — 128 bajtowy nagłówek plików MP3 zawierający informacje o pliku dźwiękowym. ID3v1 zawierał informacje o albumie, wykonawcy, ogólny komentarz, gatunek i tytuł; ID3v2 umożliwia podawanie tych danych w o wiele swobodniejszej formie i o większej objętości, ID3v3 pozwala również na zamieszczanie takich danych jak teksty piosenek czy ilustracje.
- ♦ **ISO (ang. *International Standards Organization*)** — w kontekście płyt CD i DVD ta nazwa odnosi się do trzech liter będących rozszerzeniem plików obrazów zgodnych ze standardem ISO9660, który definiuje formaty CD i DVD. Użytkownicy często nazywają w ten sposób pliki zawierające kopie płyt CD i DVD.
- ♦ **Kodek (ang. *compressor/decompressor*)** — algorytm, za którego pomocą sygnał cyfrowy jest kompresowany tak, aby straty jakości były jak najmniejsze. Służy również do dekompresowania plików w celu ich odtwarzania.
- ♦ **M4A** — kontener pliku audio, zwykle umieszcza się w nim pliki AAC lub ALAC. Zabezpieczona wersja plików M4A została objęta DRM i obecnie występuje pod nazwą MP4.
- ♦ **MD5** — popularny mechanizm nazywany potocznie sumą kontrolną, wykorzystywany do weryfikacji spójności cyfrowych plików z danymi, zwłaszcza w przypadku bezstratnych formatów dźwiękowych. Pliki tekstowe z takim rozszerzeniem zawierają zwykle 32-cyfrowy identyfikator jednego lub większej liczby plików. Suma kontrolna danego pliku jest unikalna, dlatego często jest wykorzystywana do sprawdzania jego integralności. Po wygenerowaniu sumy kontrolnej dla danego pliku (np. za pomocą polecenia systemu Linux `md5sum`) należy porównać otrzymany wynik z podaną sumą. Jeżeli są identyczne, wówczas można mieć pewność, że plik nie został zmieniony.

- ♦ **MP3 (*MPEG-1 Audio Layer 3*)** — najpopularniejszy wykorzystywany obecnie format plików dźwiękowych. Jest to własnościowy, stratny format oferujący dźwięk relatywnie wysokiej jakości przy stosunkowo niewielkim pliku.
- ♦ **MP3 CD** — płyta kompaktowa zawierająca zamiast tradycyjnych plików CDDA pliki MP3. Ponieważ pliki w formacie MP3 są znacznie mniejsze od plików CDDA, na płycie można przechowywać znacznie więcej muzyki. Format zdobył sporą popularność, mimo że takich płyt nie da się odtwarzać w standardowych odtwarzaczach CD.
- ♦ **MPAA (*ang. Motion Picture Association of America*)** — filmowy odpowiednik stowarzyszenia RIAA; grupa przemysłowa, której celem jest maksymalizowanie dochodów przemysłu filmowego często kosztem końcowych odbiorców oraz artystów.
- ♦ **MPEG (*ang. Motion Picture Expert Group*)** — konsorcjum przemysłowe odpowiedzialne za definiowanie standardów dla plików audio i wideo.
- ♦ **MPEG-4** — w czasie pisania niniejszej książki była to najnowsza generacja standardu MPEG. Oferuje on zawartość audio, wideo oraz inne multimedia.
- ♦ **Normalizacja (*ang. normalize*)** — technika obróbki dźwięku polegająca na identyfikowaniu najgłośniejszych fragmentów pliku, do których następnie dostosowywana jest głośność odtwarzania pozostałych plików. Normalizacja nie powoduje zmiany parametrów głośności samego pliku, a tylko jego odtwarzania.
- ♦ **OGG (*Ogg Vorbis*)** — popularny, otwarty, stratny format plików dźwiękowych i jednocześnie pojemnik; doskonała alternatywa dla kodeka niskiej jakości, takiego jak MP3. Ogg Vorbis został zaprojektowany przez fundację Xiph.org, która zajmuje się promowaniem wolnych formatów multimedialnych. Ogg to format, Vorbis to kodek, choć pliki nim kodowane mają rozszerzenie .ogg. Więcej informacji można znaleźć na stronach www.xiph.org oraz www.vorbis.com. Jeżeli dany odtwarzacz (sprzętowy lub programowy) nie obsługuje tego formatu, należy skontaktować się z jego producentem.
- ♦ **P2P (*ang. peer to peer*)** — popularny mechanizm wymiany danych, w którym każdy klient identyfikuje inne klienty i wymienia z nimi dane bez pośrednictwa centralnego serwera. Programy używane do takiej wymiany to BitTorrent, Gnutella, LimeWire itd.
- ♦ **PCM (*ang. Pulse Code Modulation*)** — format danych audio i wideo dostarczający rzeczywistej reprezentacji nagranych sygnałów dzięki próbkowaniu go w regularnych odstępach czasu. Jest to standard używany w komputerach, płytach kompaktowych i cyfrowym wideo.
- ♦ **RealAudio** — popularny strumieniowy format audio często wykorzystywany na stronach internetowych. Więcej informacji można znaleźć na stronie jego producenta, firmy Real.com (www.real.com).
- ♦ **Redukcja szumów** — technika obróbki dźwięku polegająca na „czyszczeniu” plików audio poprzez usuwanie, filtrowanie lub wyrównywanie określonych zakresów częstotliwości.

- ♦ **RIAA (ang. *Recording Industry Association of America*)** — grupa przemysłowa, której celem jest maksymalizowanie dochodów przemysłu filmowego często kosztem końcowych odbiorców oraz artystów.
- ♦ **Ripowanie, zrzucanie** — termin używany na określenie procesu konwertowania zawartości płyt CD i DVD do plików, z których można korzystać na komputerze.
- ♦ **SHN (*kodek Shorten*)** — popularny, bezstratny kodek często używany do autoryzowanego rozpowszechniania nagrań w bezpośredniej transmisji internetowej. Oferuje on nieco większe pliki niż jego główny konkurent, format FLAC. Interesujące jest to, że format ten podaje też tryb stratny, choć nie bardzo wiadomo, dlaczego ktoś miałby go używać. Więcej informacji na temat tego kodeka oraz współpracującego z nim oprogramowania dla systemów linuksowych można znaleźć na stronie <http://research.umbc.edu/~hamilton/shnlinks.html>. Kodek Shorten to komercyjny produkt firmy SoftSound Ltd, ale wyposażony w przyjazną dla użytkownika licencję. Więcej informacji można znaleźć na stronie www.softsound.com.
- ♦ **Skórka** — alternatywny interfejs dla aplikacji graficznych. Nowe skórki mogą być dodawane do programów bez ponownej ich kompilacji czy innych poważnych zmian; wystarczy tylko zmiana ustawień. Wiele programów do odtwarzania (takich jak XMMS czy WinAmp, który był prekursorem tej technologii) oferuje tę opcję, dzięki czemu ich użytkownicy zyskują wspaniałe narzędzie dostosowywania aplikacji do swoich wymagań.
- ♦ **Strumieniowe przesyłanie danych** — metoda dostarczania danych cyfrowych poprzez internet w taki sposób, że aplikacja odbierająca (np. odtwarzacz audio) otrzymuje porcje danych na bieżąco, co eliminuje potrzebę ich zapisywania na dysku komputera. Najczęściej podawanymi przykładami odtwarzaczy pracujących w tej technologii są RealPlayer oraz RealOne produkowane przez firmę RealAudio. Więcej informacji można znaleźć na stronie www.real.com.
- ♦ **SVCD (ang. *Super Video CD*)** — następca formatu VCD, zawiera obraz w formacie MPEG-2 oraz dźwięk w formacie MPEG-1 lub MPEG-2. Płyty SVCD można odtwarzać za pomocą większości odtwarzaczy DVD, ponieważ obsługują one format kodowania MPEG-2.
- ♦ **Track-at-once** — mechanizm zapisu ścieżek na płytach CD i DVD, umożliwiający zapisywanie kolejnych ścieżek.
- ♦ **Track-at-once** — technologia zapisu na płytach CD i DVD w której zapisywana jest od razu cała powierzchnia dysku, a nie pojedyncze informacje, które użytkownik chce zachować.
- ♦ **Transkodowanie** — proces konwertowania pliku z jednego zakodowanego formatu do innego, zwykle odnoszący się do konwersji plików medialnych lub jednego obiektu do drugiego. Ma on miejsce np. podczas nagrywania płyt DVD z plików w formacie MPEG (lub w odwrotnym kierunku).
- ♦ **TTA (ang. *True Audio codec*)** — wolny, bezstratny kodek umożliwiający redukcję rozmiaru pliku średnio o około 30% – 40% bez jakiegokolwiek utraty jakości. Więcej informacji na temat kodeka True Audio można znaleźć na stronie www.true-audio.com.

- ♦ **VBR (ang. *Variable Bit Rate*)** — rodzaj kodowania pozwalający na zmniejszenie rozmiaru pliku bez poświęcania jakości przy pomocy dostosowywania przepustowości, z jaką nagrywane są dane audio. Złożone fragmenty nagrania rejestrowane są z większą przepustowością, podczas gdy proste z niższą.
- ♦ **VCD (ang. *Video CD*)** — nośnik zdolny do przechowywania 650 MB (w przybliżeniu) zapisanych wstępnie danych audio i wideo i zakodowanych w formacie MPEG-1. Większość odtwarzaczy CD i DVD potrafi odtwarzać płyty VCD, ponieważ obsługują one kodek MPEG-1, którym płyty VCD są kodowane.
- ♦ **WAV** — cyfrowy format audio zaprojektowany przez Microsoft i IBM, który stał się standardowym formatem dystrybuowania bezstratnych plików audio. Pliki zapisane w tym formacie składają się z porcji danych, które w zasadzie mogą być zakodowane przy użyciu prawie każdego kodeka, jednak w większości przypadków są surowymi danymi PCM. Dane zapisane w formacie PCM są bezstratne, ale niestety bardzo duże. Ten format plików używany jest często przez profesjonalistów zajmujących się dźwiękiem, ponieważ umożliwia zachowanie maksymalnej jakości i wierności w stosunku do oryginalnych nagrań.
- ♦ **Wavpack** — darmowy i bezstratny kodek audio używany początkowo w systemach Windows, ale z czasem rozpowszechnił się i na innych architekturach, również w Linuksie, dostępny jest np. w programie XMMS. Mimo to, poza środowiskami Windows, spotyka się go stosunkowo rzadko, chociaż oferuje dużą szybkość kompresji i dekompresji bez straty jakości. Więcej informacji można znaleźć na stronie www.wavpack.com.
- ♦ **WMA (ang. *Windows Media Audio*)** — popularny w systemach Microsoft Windows kodek audio, domyślnie używany przez program Windows Media Player. Dostępny jest w wersji stratnej, bezstratnej oraz zabezpieczonej DRM. Najważniejsze dla użytkowników Linuksa jest to, w jaki sposób konwertować zakodowane w nim pliki do bardziej przyjaznych formatów.
- ♦ **WMV (ang. *Windows Media Video*)** — popularny w systemach Microsoft Windows kodek wideo, domyślnie używany przez program Windows Media Player. Najważniejszą dla użytkowników Linuksa informacją jest ta, w jaki sposób konwertować zakodowane w nim pliki do bardziej przyjaznych formatów.
- ♦ **Wypalanie** — zapisywanie plików na nośnikach jedno- i wielokrotnego zapisu typu CD i DVD. Sam termin wziął się stąd, że do samego zapisu na płytach CD i DVD używa się promienia lasera, który umieszcza na specjalnej warstwie wewnątrz płyty specjalne wgłębienia, które odpowiadają danym cyfrowym.

Jeżeli czytnik posiada mnóstwo wolnych szarych komórek, z którymi nie wie, co zrobić, wówczas może zapamiętać wszystkie podane tu terminy, w przeciwnym razie może pozwolić spełnić książce swoje zadanie i sięgać do niej, kiedy trzeba. Teraz, kiedy czytelnicy poszerzyli już swoją wiedzę z zakresu terminologii audio i wideo, czas przejść do kolejnych zagadnień: odtwarzania audio i wideo w Ubuntu.

Konfigurowanie urządzeń dźwiękowych, poziomu dźwięku i dźwięków systemowych

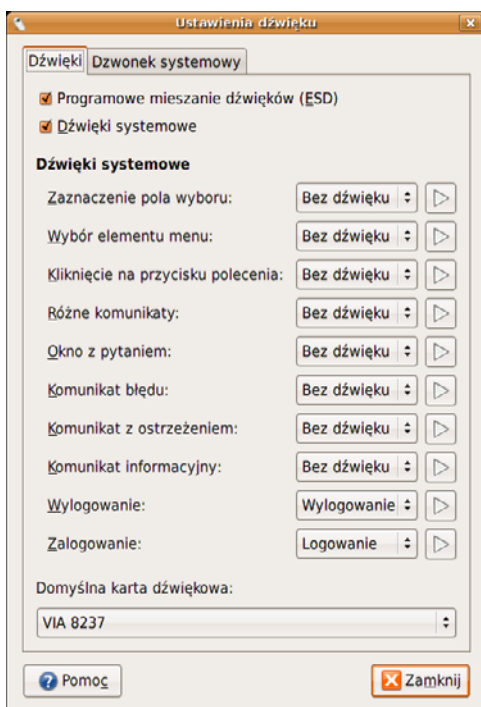
Większość ustawień konfiguracyjnych dotyczących systemu dźwiękowego w Ubuntu jest dokonywana automatycznie podczas instalowania systemu. Instalator Ubuntu bardzo dobrze radzi sobie z wykrywaniem sprzętu oraz instalowaniem i konfigurowaniem systemu ALSA (ang. *Advanced Linux Sound Architecture*), sterowników i bibliotek, dzięki którym system operacyjny oferuje doskonały dźwięk wysokiej jakości. Poniżej zamieszczam wskazówki, w jaki sposób przetestować możliwości systemu dźwiękowego komputera, dostosować dźwięki do różnych zdarzeń mających miejsce w systemie oraz ustawić poziom głośności.

Testowanie i dostosowywanie systemu dźwiękowego

Domyślnie system jest skonfigurowany tak, aby podczas uruchamiania odtworzyć dźwięki „powitalne”. Żeby na tej podstawie przekonać się, czy system dźwiękowy działa, należy najpierw upewnić się, czy głośniki zostały włączone i podpięte do portów karty dźwiękowej (zwykle jest to gniazdo koloru zielonego znajdujące się pomiędzy innymi). Następnie trzeba wybrać z menu kolejno *System/Preferencje/Dźwięk*, a zostanie wyświetlone okno widoczne na rysunku 13.1.

Rysunek 13.1.

Domyślne ustawienia dźwięków w systemie Ubuntu



W tym oknie można sprawdzić, przy których zdarzeniach systemowych będą odtwarzane dźwięki. Domyślnie są to logowanie do systemu i wylogowanie z niego, co widać na

rysunku 13.1. Aby odtworzyć dźwięk przypisany do danego zdarzenia, wystarczy kliknąć przycisk ze strzałką skierowaną w prawą stronę, znajdujący się na prawo od listy przypisanych dźwięków (jeżeli jest wybrana opcja *Bez dźwięku*, wówczas żaden dźwięk nie zostanie odtworzony). Jeżeli użytkownik usłyszy dźwięk — gratulacje! Oznacza to, że system dźwiękowy w Ubuntu jest poprawnie skonfigurowany.

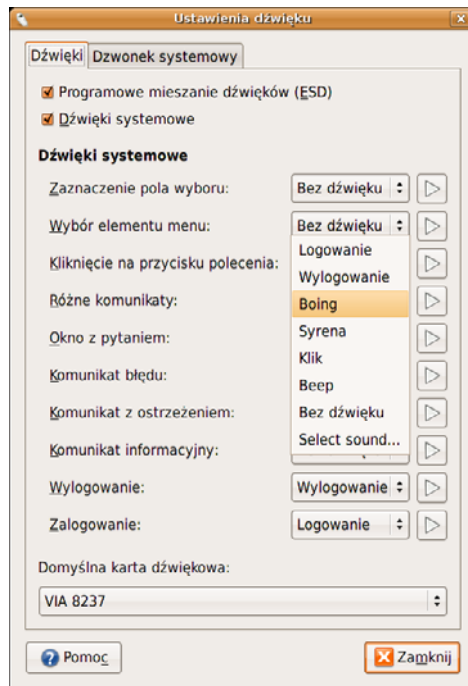


Jeżeli w komputerze jest więcej niż jedna karta dźwiękowa (np. zintegrowana z płytą główną oraz zamontowana oddzielnie), wówczas należy się upewnić, że na dole okna *Ustawienia dźwięku* wybrana jest ta karta, do której wpięte są głośniki. Jeżeli tak nie jest, wówczas należy kliknąć pole i wybrać odpowiednią kartę, po czym spróbować ponownie odtworzyć dźwięki.

Jeżeli użytkownik chciałby przypisać do wybranego zdarzenia systemowego jakiś dźwięk, wówczas należy kliknąć etykietę *Bez dźwięku* przy danym zdarzeniu, a następnie wybrać z listy rozwijanej dowolny dźwięk (patrz rysunek 13.2).

Rysunek 13.2.

Lista rozwijana z dźwiękami dostępnymi w systemie



Jeżeli użytkownik chciałby wykorzystać dźwięk spoza wyświetlonej listy, wówczas należy z listy rozwijanej wybrać opcję *Select sound*, a następnie, korzystając z okna widocznego na rysunku 13.3, wskazać odpowiedni plik.

Ustawianie poziomu głośności

W rozdziale 5., w podpunkcie „Dostosowywanie zawartości paneli”, napisałem, jak dodać do paneli GNOME różne aplety. Jednym z bardziej przydatnych, a instalowanych domyślnie w systemie, jest aplet regulacji głośności, który wygląda jak mały głośnik i jest wyświetlany na górnym panelu, na lewo od apletu daty.

Rysunek 13.3.

Wybieranie nowego
dźwięku dla zdarzenia
systemowego



Aby dostosować poziom głośności dźwięku w systemie, należy kliknąć prawym przyciskiem myszy ten aplet, zostanie wówczas wyświetlony suwak (patrz rysunek 13.4).

Rysunek 13.4.

Szybka zmiana
poziomu głośności
dźwięku

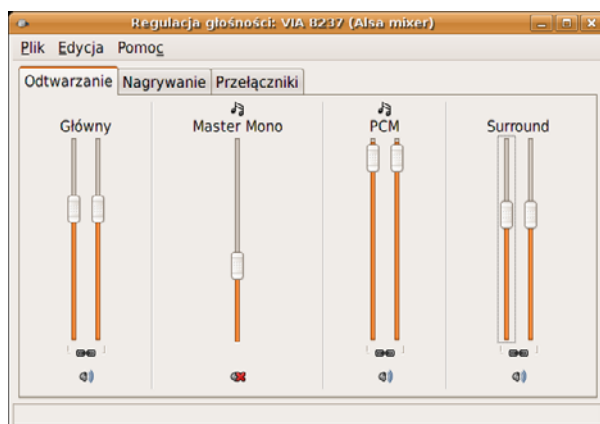


Aby zmniejszyć głośność dźwięków, należy przeciągnąć suwak w dół, aby zwiększyć — w górę.

By uzyskać dostęp do bardziej zaawansowanych ustawień, trzeba kliknąć aplet prawym przyciskiem myszy i wybrać z menu kontekstowego opcję *Otwórz regulację głośności*, a zostanie wyświetlone okno dialogowe widoczne na rysunku 13.5.

Rysunek 13.5.

Szczegółowa kontrola
poziomu głośności
dźwięku

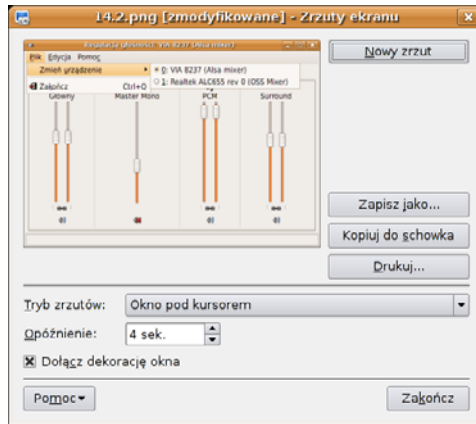




Regulacja dźwięku za pomocą narzędzi widocznych na rysunku 13.5 jest oparta na możliwościach karty dźwiękowej zainstalowanej w komputerze. Dlatego też zawartość okna dialogowego *Regulacja głośności* zależy od typu karty. Na rysunku 13.5 można zauważyć, że kartą dźwiękową używaną w systemie jest zintegrowany z płytą główną podzespół VIA.

Jeżeli w systemie znajduje się kilka kart dźwiękowych (np. karta zintegrowana z płytą główną oraz dodatkowa, zainstalowana oddzielnie), wówczas należy upewnić się, że wszelkie wprowadzane zmiany dotyczą tej karty, do której podpięte są głośniki, i tej, która wymieniona jest w belce tytułowej okna widocznego na rysunku 13.5. Aby zmienić urządzenie, dla którego modyfikowane są ustawienia, należy z menu wybrać kolejno *Plik/Zmień urządzenie* i zaznaczyć pole wyboru przy odpowiedniej pozycji (patrz rysunek 13.6).

Rysunek 13.6.
Zmiana domyślnej
karty dźwiękowej

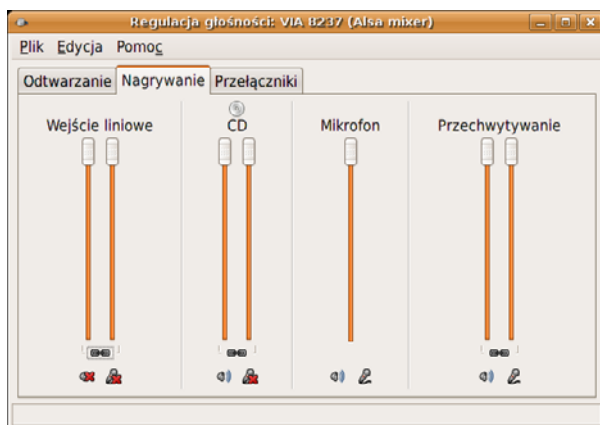


Oto podstawowe ustawienia, które użytkownik może dostosować w przypadku każdej karty dźwiękowej:

- ♦ **Główny** — głośność odtwarzania dla wszystkich aplikacji audio,
- ♦ **Headphone** — głośność odtwarzania w słuchawkach, jeżeli zostaną podłączone do odpowiedniego portu znajdującego się na karcie dźwiękowej,
- ♦ **PCM** — głośność odtwarzania plików WAVE, które zwykle mają format PCM,
- ♦ **PC Speaker** — głośność odtwarzania dźwięków za pomocą głośnika wbudowanego w komputerze, który zwykle jest bezużyteczny.
- ♦ Pod każdym z tych elementów mogą znajdować się ikony (jedna lub dwie). Za pomocą symbolu ogniwa łańcucha można zdecydować, czy suwaki dla obu wyjść (kanały lewy i prawy) mają przesuwac się jednocześnie, czy użytkownik może nimi kierować osobno. Jeżeli ogniwa łańcucha są połączone, oznacza to, że kanałami można sterować jednocześnie; jeżeli ogniwa są rozłączone, wówczas dla każdego wyjścia ustawia się poziom głośności osobno. Aby połączyć lub rozdzielić ogniwa łańcucha, należy po prostu kliknąć ikonę.
- ♦ Ikona głośnika informuje o tym, czy dane urządzenie jest wyciszone, czy też nie. Jeżeli na ikonie widoczny jest czerwony znak „X”, wówczas urządzenie jest wyciszone, aby zmienić ten stan, należy kliknąć ikonę.

Aby dostosować głośność urządzeń służących do nagrywania dźwięku, należy przejść do zakładki *Nagrywanie*, jej zawartość przedstawiam na rysunku 13.7.

Rysunek 13.7.
Dostosowywanie
poziomu głośności
nagrywanego dźwięku



Oto podstawowe ustawienia, które użytkownik może dostosować dla każdej karty dźwiękowej:

- ♦ **Wejście liniowe** — ustawienia głośności wszelkich zewnętrznych urządzeń podłączanych do portu (oznaczanego zwykle jako *line-in*) karty dźwiękowej,
- ♦ **CD** — głośność odtwarzania z napędu CD, jeżeli jest podłączony bezpośrednio do karty dźwiękowej,
- ♦ **Mikrofon** — ustawienia głośności mikrofonu podłączonego do odpowiedniego portu znajdującego się na karcie dźwiękowej,
- ♦ **Przechwytywanie** — ustawienia głośności dźwięków aktualnie odtwarzanych w systemie.

Oprócz wymienionych powyżej dwóch ikon znajdujących się na zakładkach *Odtwarzanie* i *Nagrywanie*, na tej drugiej może też być wyświetlona ikona z symbolem mikrofonu informująca, czy dane źródło dźwięku jest używane do jego przechwytywania. Jeżeli znajduje się na niej czerwony znak „X”, wówczas ta opcja jest wyłączona. Aby zmienić ten stan, należy kliknąć ikonę z mikrofonem.

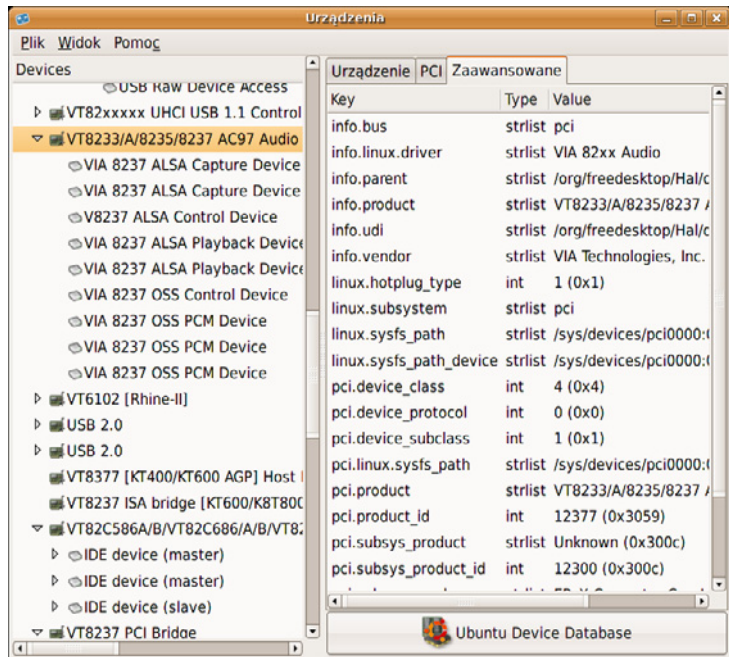
Kiedy już wszystkie zmiany zostaną wprowadzone, wówczas należy wybrać z menu kolejno *Plik/Zakończ*, aby zamknąć okno.

Uzyskiwanie dodatkowych informacji o karcie dźwiękowej

Jeżeli po wypróbowaniu metod opisanych powyżej głośniki w komputerze użytkownika nadal milczą, należy dokładniej przyrzeć się ustawieniom sprzętu audio zainstalowanego w systemie, aby zdobyć informacje, które będą potrzebne podczas szukania pomocy. By wyświetlić informacje o zainstalowanych w komputerze podzespołach, w tym również karcie dźwiękowej, należy kolejno wybrać z menu *System/Administracja/Urządzenia*, zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 13.8.

Rysunek 13.8.

Informacje o karcie
dźwiękowej dostępnej
w systemie Ubuntu



Wpisy widoczne w oknie dialogowym *Urządzenia* pomogą określić, czy Ubuntu poprawnie rozpoznał kartę dźwiękową, jaki jest jej model oraz czy istnieją odpowiednie wpisy w katalogu `/sys`, wirtualnym systemie plików umożliwiającym bezpośredni dostęp do różnych ustawień urządzeń zainstalowanych w komputerze.



Jeżeli system dźwiękowy działa poprawnie, ale nie odtwarza dźwięków z napędu CD, wówczas należy się upewnić, czy napęd jest poprawnie podłączony do karty dźwiękowej. Jest to kabel, który z jednej strony wpięty jest do tylnej części napędu CD, a z drugiej do odpowiedniego gniazda na karcie dźwiękowej lub — kiedy karta jest zintegrowana z płytą główną — w odpowiednim miejscu na samej płycie. Niektóre karty dźwiękowe posiadają kilka podobnych łącz, dlatego możliwe jest, że kabel został wpięty do niewłaściwego gniazda lub po prostu poluzował się.

Jeżeli wpisy widoczne w oknie dialogowym zdają się być zgodne z zainstalowanym sprzętem, wówczas należy ruszyć na poszukiwania w sieci, rozpoczynając od różnych zasobów Ubuntu wymienionych w rozdziale 1., w podrozdziale „Wsparcie techniczne dla Ubuntu”. Płatna pomoc też jest jakimś rozwiązaniem, ale najtańszą opcją jest przejrzenie różnych forów dyskusyjnych w poszukiwaniu informacji o podobnych problemach z takim sprzętem oraz ich rozwiązaniu. Jeżeli nie przyniesie to rezultatu, warto zamieścić pytanie na liście dyskusyjnej. W całej książce podkreślam fakt, że Ubuntu słynie z ogromnej i bardzo pomocnej społeczności użytkowników. Oczywiście, zawsze można też zapytać mnie, ale odpowiedź nadejdzie dużo później, niż podczas skorzystania z forów czy list dyskusyjnych.

Konfigurowanie napędów CD i DVD

Obecnie już bardzo niewiele osób korzysta ze stacji dyskietek, zastąpiły je napędy CD oraz DVD i dostępne dla nich nośniki wymienne. Nośnik wymienny oznacza nośnik do przechowywania danych w formie cyfrowej, który nie jest konieczną częścią systemu komputerowego. Do innej kategorii należą przenośne dyski, czyli pamięci USB, przenośne dyski twarde, a także urządzenia, które po podłączeniu do systemu udostępniane są właśnie jako dyski (np. iPody), te urządzenia zostały omówione w rozdziale 17., „Urządzenia elektroniczne w Ubuntu”, a także w rozdziale 23., „Dodawanie sprzętu i przyłączanie urządzeń peryferyjnych”. W niniejszym rozdziale omawiam metody konfigurowania systemu do pracy z płytami CD i DVD, zarówno nagranyymi, jak i pustymi.

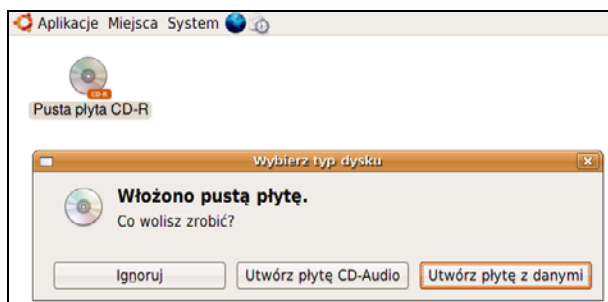
Po umieszczeniu w napędzie pustej płyty CD lub DVD system wykona dwie rzeczy:

- ♦ wyświetli na pulpicie ikonę odpowiednią do rodzaju włożonego nośnika,
- ♦ wyświetli okno dialogowe z pytaniem, co użytkownik zamierza zrobić dalej.

Na rysunku 13.9 przedstawiam ikonę i komunikat widoczne po umieszczeniu w napędzie pustej płyty CD-R lub CD-RW. Na rysunku 13.10 widać te same elementy po umieszczeniu w napędzie pustej płyty DVD-R.

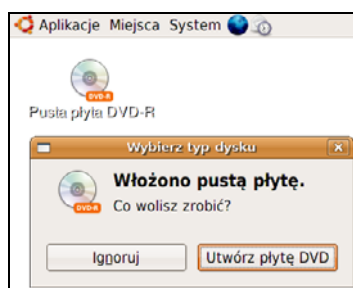
Rysunek 13.9.

Okno dialogowe i ikona wyświetlane po umieszczeniu w napędzie płyty CD-R/RW



Rysunek 13.10.

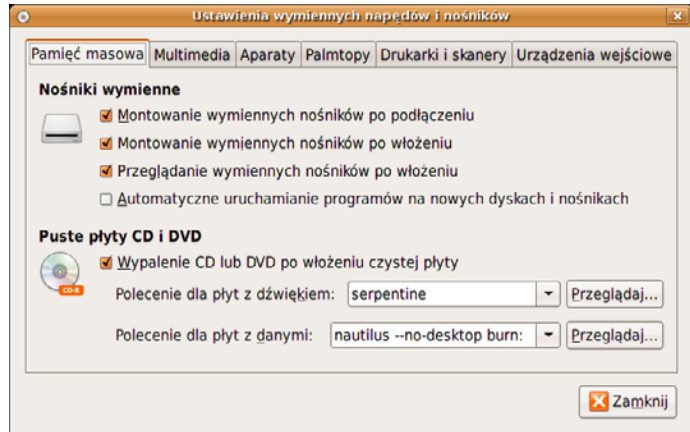
Okno dialogowe i ikona wyświetlane po umieszczeniu w napędzie pustej płyty DVD-R



Jeżeli użytkownik nie chce wykonywać żadnej czynności, należy kliknąć przycisk *Ignoruj*. Użytkownik może też skonfigurować, jakiego rodzaju działanie ma podjąć system operacyjny, a dokładniej, czy ma uruchomić wskazaną przez użytkownika aplikację służącą do nagrywania płyt. Do tego celu służy okno dialogowe widoczne na rysunku 13.11.

Rysunek 13.11.

Konfigurowanie działań podejmowanych przez system po umieszczeniu w napędzie pustej płyty CD lub DVD



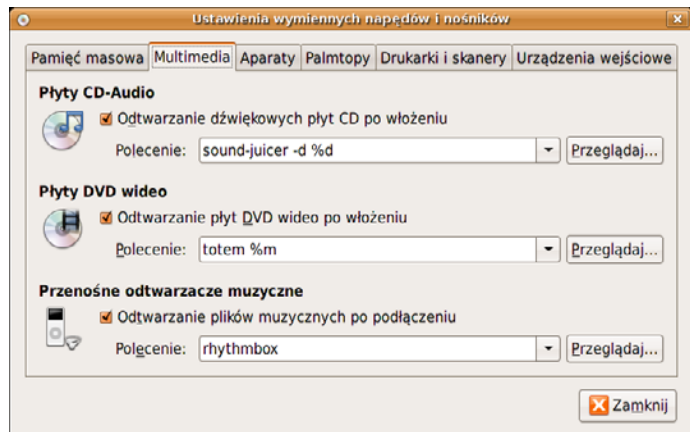
Aby wyświetlić to okno, należy z menu wybrać kolejno *System/Preferencje/Napędy i nośniki wymienne* (patrz rysunek 13.11). Pierwsza zakładka wyświetlonego okna służy do konfigurowania reakcji systemu na umieszczenie w napędzie pustych nośników CD i DVD.

Aby wyłączyć wyświetlanie przez system okna dialogowego widocznego na rysunkach 13.9 i 13.10, należy usunąć zaznaczenie z pola *Wypalenie CD lub DVD po włożeniu czystej płyty*. Żeby tylko zmienić program używany do obsługi procesu nagrywania płyt, należy pozostawić to pole zaznaczone, a następnie kliknąć przycisk *Przeglądaj* i wskazać inne aplikacje. Domyślne ustawienia dla Ubuntu przewidują użycie, do nagrywania płyt CD i DVD zawierających dane, menedżera plików Nautilus.

Reakcja systemu na umieszczenie w napędzie już nagranej płyty CD lub DVD zależy od ustawień wprowadzonych w zakładce *Multimedia* okna dialogowego *Ustawienia wymiennych napędów i nośników* (patrz rysunek 13.12).

Rysunek 13.12.

Konfigurowanie ustawień dla nagranych płyt CD i DVD



Aby wyłączyć automatyczne uruchamianie wybranej aplikacji, należy usunąć zaznaczenie z pola wyboru znajdującego się obok danego rodzaju nośnika. By wskazać inny program, który ma być włączany po wykryciu nośnika, należy kliknąć przycisk *Przeglądaj* i przejść

do wybranej aplikacji. W domyślnych ustawieniach Ubuntu do odtwarzania nagranych płyt CD i DVD używa tej samej aplikacji o nazwie Totem (więcej na jej temat w dalszej części rozdziału).

Instalowanie GStreamer i wtyczek

Większość aplikacji audio i wideo działających w GNOME korzysta ze szkieletu (ang. *framework*) o nazwie GStreamer (www.gstreamer.net). Ułatwia on korzystanie z kodeków poprzez system wtyczek, które z kolei są zintegrowane ze szkieletem. Większość aplikacji opisanych w niniejszym rozdziale to rozwojowe elementy środowiska GNOME, wykorzystujące kodeki dzięki strukturze GStreamer i systemowi wtyczek. Dlatego, zanim zostanie uruchomiony Rhythmbox czy Totem, należy skorzystać z menedżera pakietów Synaptic, aby zainstalować najnowszą wersję GStreamer oraz skojarzony z nim zestaw wtyczek. Inaczej użytkownik może być srodze zawiedziony możliwościami multimedialnymi Ubuntu.

Jak wyjaśniłem w rozdziale 1., programiści Ubuntu kładą szczególny nacisk na kwestię wymogów licencyjnych pakietów umieszczonych w systemie. Ze względu na prawa autorskie oraz opłaty licencyjne wiele powszechnie używanych na innych platformach i w różnych programach kodeków nie zostało domyślnie zainstalowanych w Ubuntu. Najlepszym przykładem są pliki w formacie MP3. Teoretycznie nie każdy musi odtwarzać muzykę w najpopularniejszym na świecie formacie... Właściwie to nie można winić programistów Ubuntu za taki punkt widzenia nawet wtedy, kiedy nie spełnia oczekiwań wszystkich użytkowników.



W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, szczegółowo wyjaśniam procedurę dodawania repozytoriów za pomocą Synaptica. Poniżej, bez wyjaśnienia zasad, zamieszczam tylko proste wskazówki, za których pomocą można dodać potrzebne repozytoria.

Aby w systemie Ubuntu odtwarzać i konwertować pliki MP3 i płyty DVD, należy zainstalować GStreamer oraz powiązane z nim wtyczki, ale by to uczynić, najpierw trzeba nieco zmienić rodzaj oprogramowania wyszukiwanego przez menedżera pakietów Synaptic. Żeby to zrobić, należy przeprowadzić edycję pliku `/etc/apt/sources.list`, najlepiej za pomocą ulubionego edytora tekstu, np. vi:

```
$ sudo vi /etc/apt/sources.list
```

Następnie trzeba usunąć znak `#` z początku linii zawierających repozytoria *universe*, następnie dodać słowo *multiverse* na końcu linii i zapisać plik.



Aby wszystko było zupełnie jasne: dodając do systemu repozytoria *universe* i *multiverse*, uzyskuje się możliwość korzystania z oprogramowania, które może być obwarowane ograniczeniami licencyjnymi nie do przyjęcia dla niektórych firm (np. ograniczenia dotyczące plików MP3); dla nich też nie jest gwarantowane wsparcie twórców Ubuntu (nawet w razie wykupienia płatnego wsparcia), a zatem mogą wystąpić trudności z uzyskaniem pomocy. Dlatego też, decydując się na korzystanie z tego oprogramowania, użytkownik podejmuje pewne ryzyko.

Następnie należy uruchomić pakiet Synaptic (menu *System/Administracja*) i kliknąć przycisk *Odśwież*, co spowoduje ponowne wczytanie listy wszystkich dostępnych pakietów. Kiedy ten proces zostanie zakończony, trzeba wcisnąć przycisk *Szukaj* i wpisać ciąg *gstre-amer*. Kiedy wyszukiwanie zostanie zakończone, należy na wyświetlonej liście zaznaczyć następujące pakiety:

- ♦ *gstreamer0.10-alsa* (powinien już być zaznaczony),
- ♦ *gstreamer0.10-esd* (powinien już być zaznaczony),
- ♦ *gstreamer0.10-ffmpeg*,
- ♦ *gstreamer0.10-gl*,
- ♦ *gstreamer0.10-gnomevfs* (powinien już być zaznaczony),
- ♦ *gstreamer0.10-pitfdll*,
- ♦ *gstreamer0.10-plugins-bad*,
- ♦ *gstreamer0.10-plugins-bad-multiverse*,
- ♦ *gstreamer0.10-plugins-base* (powinien już być zaznaczony),
- ♦ *gstreamer0.10-plugins-base-apps* (powinien już być zaznaczony),
- ♦ *gstreamer0.10-plugins-good* (powinien już być zaznaczony),
- ♦ *gstreamer0.10-plugins-ugly*,
- ♦ *gstreamer0.10-plugins-ugly-multiverse*,
- ♦ *gstreamer0.10-tolls* (powinien już być zaznaczony),
- ♦ *gstreamer0.10-x* (powinien już być zaznaczony),
- ♦ *gstreamer0.8-dv*,
- ♦ *gstreamer0.8-dvd*,
- ♦ *gstreamer0.8-faac*,
- ♦ *gstreamer0.8-faad*,
- ♦ *gstreamer0.8-flac*,
- ♦ *gstreamer0.8-lame*,
- ♦ *gstreamer0.8-mad*,
- ♦ *gstreamer0.8-misc*,
- ♦ *gstreamer0.8-mpeg2dec*,
- ♦ *gstreamer0.8-musepack*,
- ♦ *gstreamer0.8-speex*,
- ♦ *gstreamer0.8-swfdec*,
- ♦ *gstreamer0.8-vorbis*,
- ♦ *gstreamer0.8-xvid*,
- ♦ *totem-gstreamer* (powinien już być zaznaczony).

Aby zaznaczyć dany element, należy kliknąć go prawym przyciskiem myszy i z menu kontekstowego wybrać opcję *Zaznacz do instalacji*.



Jeżeli użytkownik nie planuje przeprowadzania konwersji dźwięku, wówczas nie wszystkie wymienione pakiety będą mu potrzebne. Jednak sugeruję zainstalowanie wszystkich dostępnych pakietów, jedynym kosztem jest tylko nieco przestrzeni dyskowej.

W zależności od wybranej wersji instalowanych pakietów, a także innego oprogramowania obecnego w systemie, Synaptic może wyświetlić okno z informacją o dodatkowych pakietach, które trzeba będzie dodać do systemu — należy je zatwierdzić.

Kiedy zostaną już wybrane wszystkie potrzebne pakiety, należy kliknąć przycisk *Zastosuj*, pakiety zostaną wówczas pobrane i zainstalowane w systemie. Jeżeli cała operacja zakończy się sukcesem, można zakończyć działanie menedżera pakietów Synaptic.

Instalowanie w systemie kodeków Microsoft Windows

W Microsoft Windows używanych jest wiele własnościowych kodeków audio i wideo. Aby odtwarzać wiele z nich w Linuksie, należy zainstalować wolno dostępny pakiet kodeków Windows i korzystać z nich poprzez wymieniony wcześniej pakiet *gststreamer0.10-pitfdll*. Można dodać go do systemu na dwa sposoby:

- ♦ pobrać odpowiedni pakiet Debiana zawierający te kodeki i zainstalować go, korzystając z polecenia `dpkg`,
- ♦ pobrać plik archiwum zawierający te kodeki i zainstalować go w systemie w `/usr/lib/win32`.

Aby pobrać i zainstalować pakiet Debiana zawierający te kodeki, należy kolejno:

1. Wpisać następujące polecenie (wszystko w jednej linii, bez spacji po `http`):

```
$ wget -c http://download.ubuntu.pl/Dapper_Drake/w32codecs/
w32codecs_20060611-1_i386.deb
```

2. Zainstalować pakiet, korzystając z następującego polecenia:

```
$ sudo dpkg -i w32codecs_20060611-1_i386.deb
```

Aby zdobyć kodeki ze strony internetowej i zainstalować je samodzielnie, należy kolejno:

1. Przejść na stronę www.mplayerhq.hu/design7/dload.html (można przełączyć język interfejsu na polski), a następnie pobrać pakiet *essential-2006-06-11.tar.bz2* (lub nowszą wersję) i pobrać ją z serwera lustrzanego znajdującego się najbliżej.

2. Rozpakować zawartość archiwum, używając polecenia

```
$ tar jxvf essential-2006-06-11.tar.bz2
```

3. Przenieść utworzony poprzednio katalog (*essential-2006-06-11*) i zmienić jego nazwę na `/usr/lib/win32` za pomocą polecenia:

```
$ mv -v essential-2006-06-11 /usr/lib/win32
```

Niezależnie od metody, której użyje czytelnik, jej efektem będzie możliwość odtwarzania znacznie większej ilości formatów plików multimedialnych np. za pomocą Rhythmboksa.

Instalowanie w systemie oprogramowania umożliwiającego odtwarzanie płyt DVD

Choć technicznie rzecz biorąc, zainstalowanie biblioteki umożliwiającej odtwarzanie zakodowanych płyt DVD nie ma nic wspólnego z GStreamerem, to jednak pozwoli wykorzystać kolejne możliwości oferowane przez ten szkielet, zatem do dzieła.



Moim zdaniem instalowanie w Ubuntu oprogramowania CSS (http://pl.wikipedia.org/wiki/Content_Scrambling_System) umożliwiającego odtwarzanie komercyjnych płyt DVD nie jest przywilejem ani też łamaniem prawa. Należy jednak pamiętać, że nie jestem prawnikiem. Zainstalowanie tego oprogramowania w systemie może spowodować, że pewnego ranka siły specjalne CBS wyważą drzwi w domu użytkownika i poddadzą go przymusowej reedukacji. Dlatego przed dodaniem tego oprogramowania do systemu należy się upewnić, że nie przyniesie to żadnych negatywnych skutków i na pewno nie instalować go na komputerach firmowych.

Aby zainstalować omawianą bibliotekę, należy użyć następującego polecenia:

```
$ sudo /usr/share/doc/libdvdrread3.examples/install-css.sh
```

Jak zawsze, podczas korzystania z polecenia `sudo`, trzeba będzie podać hasło, dopiero po tym poleceniu zostanie wykonane. Skrypt instalacyjny *install-css.sh* spowoduje pobranie biblioteki *decss* i zainstalowanie jej w systemie.

Gratulacje! Od tej chwili można już w domowym zaciszu odtwarzać zabezpieczone filmy DVD w systemie Ubuntu.



Najnowsze informacje na temat korzystania w Ubuntu z własnościowych kodeków audio i wideo można znaleźć na stronie <https://help.ubuntu.com/community/RestrictedFormats>.

Używanie płyt CD

Posiadam znajomych, którzy wciąż dobrowolnie używają Windows, argumentując, że system „po prostu działa”. Dzięki wysiłkowi wszystkich członków społeczności i programistów Ubuntu to samo można powiedzieć o odtwarzaniu muzyki w tym Linuksie. Nigdy nie napotkałem żadnych problemów z napędem CD w komputerach stacjonarnych i przenośnych, na których był zainstalowany system Ubuntu.

Po włożeniu do napędu nagranej płyty audio CD system wykona dwie operacje:

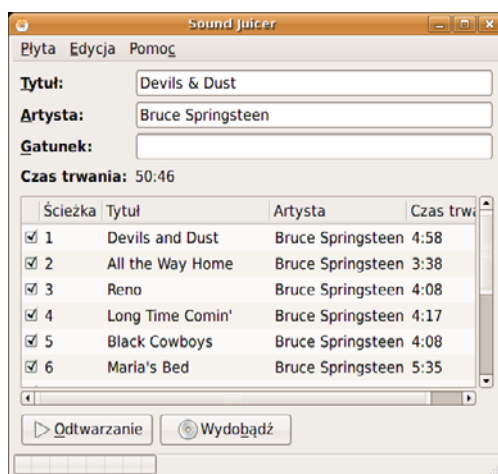
- ♦ wyświetli odpowiednią ikonę na pulpicie,
- ♦ uruchomi program Sound Juicer, za którego pomocą będzie można zrzucić zawartość płyty na dysk komputera.

Jeżeli program Sound Juicer nie zostanie uruchomiony automatycznie, można włączyć go samodzielnie, wybierając z menu kolejno *Aplikacje/Dźwięk i obraz/Zgrywarka CD-Audio Sound Juicer*.

Odtwarzanie płyt CD za pomocą programu Sound Juicer

Kiedy program zostanie już uruchomiony, spróbuje wyszukać informacje o znajdującej się w napędzie płycie CD w internetowej bazie danych MusicBrainz. Jeżeli je odnajdzie, wówczas lista ścieżek znajdujących się na płycie zostanie wyświetlona w oknie programu (patrz rysunek 13.13).

Rysunek 13.13.
Informacje o płycie CD
pobrane
z MusicBrainz



Jeżeli informacje o płycie nie zostaną znalezione, wówczas program użyje standardowych wpisów typu *Nieznany artysta*, *Nieznany tytuł*, a kolejnym ścieżkom zostaną nadane nazwy *Ścieżka1*, *Ścieżka2*.

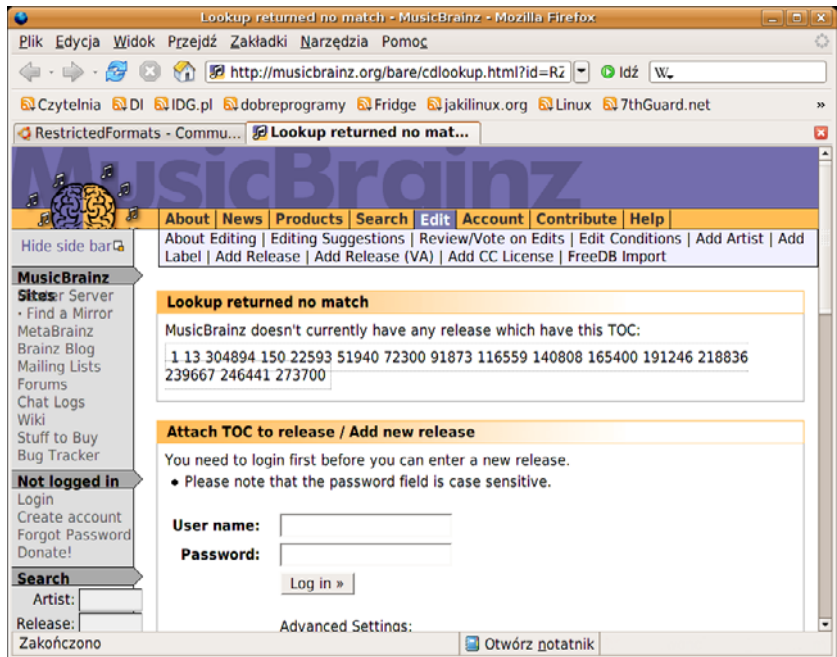
Jeżeli informacje o płycie, która została umieszczona w napędzie w celu odtworzenia lub zrzucenia (więcej na ten temat w dalszej części rozdziału), nie znajdują się jeszcze w bazie MusicBrainz (np. dlatego, że płyta właśnie ukazała się na rynku), a użytkownik chciałby być dobrym członkiem społeczności internetowej, może pokusić się o dodanie wpisu o danej płycie do bazy. W tym celu należy z menu wybrać *Płyta/Wyslij nazwy ścieżek*, co spowoduje połączenie z witryną MusicBrainz (patrz rysunek 13.14), na której będzie można założyć konto i następnie wprowadzić informacje o płycie. Jest to bardzo chwalebne zachowanie, jeżeli tylko dysponuje się czasem; następny użytkownik, który będzie szukał informacji o tej płycie, bez trudu będzie mógł je znaleźć.

Po zakończeniu wprowadzania informacji, aby odtworzyć płytę, wystarczy kliknąć przycisk *Odtwarzanie*. To po prostu działa!



Można odtworzyć tylko wybrane ścieżki, należy tylko odpowiednio dodać lub usunąć zaznaczenia w polu wyboru znajdującym się przy numerze każdego utworu.

Rysunek 13.14.
Internetowa baza
danych o płytach CD

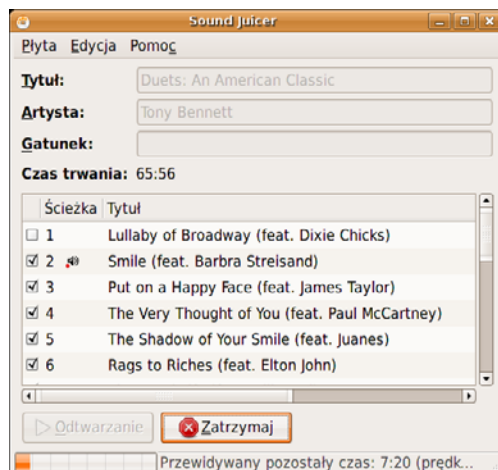


Po odtworzeniu całej płyty można wysunąć tackę napędu, w tym celu należy wybrać z menu *Płyta/Wysuń*, teraz można umieścić nową płytę w napędzie lub zakończyć działanie programu Sound Juicer.

Zgrywanie płyt CD za pomocą programu Sound Juicer

Zgrywanie płyt (używa się też terminu zrzucanie) przebiega niemal tak samo jak ich odtwarzanie, tyle tylko, że zamiast klikać przycisk *Odtwarzanie*, należy kliknąć przycisk *Wydobądź*. Podczas zgrywania zawartości płyty CD i konwertowania jej do postaci plików audio program będzie wyświetlał pasek postępu widoczny na rysunku 13.15.

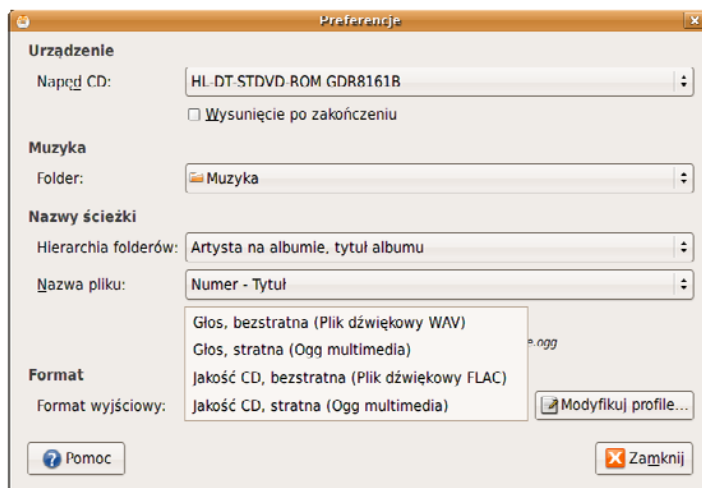
Rysunek 13.15.
Zgrywanie zawartości
płyty CD audio
za pomocą programu
Sound Juicer



Przed rozpoczęciem zgrywania warto sprawdzić konfigurację programu, czyli lokalizację, w której będą zapisywane pliki, format itd. Te ustawienia można wyświetlić, wybierając z menu kolejno *Edycja/Preferencje*, wówczas zostanie wyświetlone okno dialogowe widoczne na rysunku 13.16.

Rysunek 13.16.

Konfigurowanie programu Sound Juicer



Okno dialogowe widoczne na rysunku 13.16 umożliwia użytkownikowi wprowadzenie następujących zmian w ustawieniach programu.

- ♦ **Urządzenie** — jeżeli w komputerze zamontowano więcej niż jeden napęd CD/DVD, wówczas dzięki tej liście rozwijanej użytkownik może upewnić się, że wybrano odpowiednie urządzenie.
- ♦ **Muzyka** — tu można wskazać, do którego folderu mają zostać zapisane pliki. Zwykle tworzę w katalogu domowym folder o nazwie *Muzyka*, w którym przechowuję całą kolekcję muzyczną, dzięki czemu zachowany jest względny porządek. Takie rozwiązanie upraszcza również wykonywanie kopii zapasowych.
- ♦ **Nazwy ścieżek** — umożliwia określenie hierarchii katalogów tworzonych przez program podczas zgrywania muzyki. Domyślnie pliki zapisywane są w folderze o takiej samej nazwie, jak tytuł płyty CD, który z kolei zapisywany jest w folderze z nazwą artysty. Nazwa pliku odpowiadającego każdej ścieżce jest poprzedzona jej numerem, co ułatwia odtwarzanie utworów w takiej kolejności, w jakiej zostały umieszczone na płycie CD. Takie rozwiązanie wydaje się bardzo wygodne, ale użytkownik może zmienić te ustawienia, dopasowując je do własnych potrzeb.
- ♦ **Format** — umożliwia określenie formatu plików, w jakim mają być zapisane zgrywane utwory. Do wyboru są następujące opcje: *Głos, bezstratna (Plik dźwiękowy WAV)*, *Głos, stratna (Ogg multimedia)*, *Jakość CD, bezstratna (Plik dźwiękowy FLAC)*, *Jakość CD, stratna (Ogg multimedia)*. W przypadku nagrań na żywo, którymi użytkownik chce się podzielić z innymi, warto wybrać opcję *Jakość CD, bezstratna (Plik dźwiękowy FLAC)*.

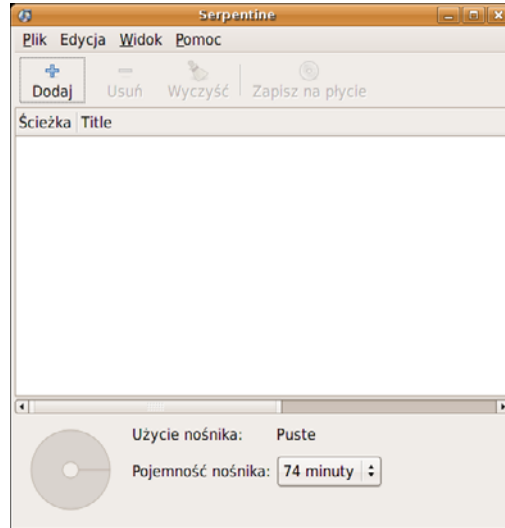
Użytkownik raczej niezbyt często będzie zmieniał powyższe ustawienia — może z wyjątkiem hierarchii folderów — ale warto wiedzieć, że do dyspozycji są odpowiednie możliwości.

Nagrywanie płyt z użyciem programu Serpentine

Dzięki programowi GNOME Serpentine tworzenie płyt audio CD na podstawie zgromadzonych w systemie plików dźwiękowych jest proste i wygodne. Program ten zostanie automatycznie uruchomiony przez system, kiedy tylko użytkownik umieści w napędzie pustą płytę CD i określi, że chce nagrać płytę audio (patrz rysunek 13.9). Ekran startowy programu przedstawiam na rysunku 13.17.

Rysunek 13.17.

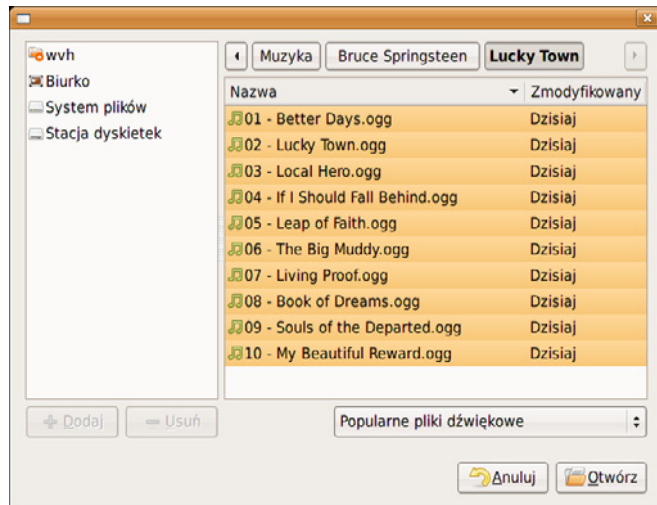
Pierwszy etap podczas nagrywania płyty audio CD



Kiedy okno programu zostanie wyświetlone, wówczas należy kliknąć przycisk *Dodaj*, co spowoduje pojawienie się okna, za którego pomocą będzie można wybrać utwory do nagrania na płytę (patrz rysunek 13.18).

Rysunek 13.18.

Wybieranie utworów, które mają zostać nagrane na płytę



Jak w większości aplikacji graficznych, poszczególne pliki można zaznaczyć kursorem myszy (wcisnięty klawisz *Ctrl* i lewy przycisk myszy); aby zaznaczyć grupę plików, należy wcisnąć klawisz *Shift* i wskazać je kursorem myszy. Po zakończeniu dodawania plików trzeba kliknąć klawisz *Otwórz*. Zostanie ponownie wyświetlone okno dialogowe widoczne na rysunku 13.17, tym razem jednak z listą wybranych utworów oraz łącznym czasem ich odtwarzania i informacją, czy zmieszczą się na płycie.



Jeżeli w komputerze znajduje się więcej niż jeden napęd CD/DVD, wówczas przed rozpoczęciem nagrywania warto się upewnić, że program *Serpentine* użyje właściwego urządzenia. Aby to zrobić, należy wybrać z menu programu *Edycja/Preferencje* i sprawdzić, czy wskazano właściwy napęd.

Jeżeli wszystko jest przygotowane, należy kliknąć przycisk *Zapisz na płycie*, a zostanie wyświetlone okno dialogowe widoczne na rysunku 13.19.

Rysunek 13.19.

Żądanie potwierdzenia rozpoczęcia nagrywania płyty CD



Po kliknięciu przycisku rozpocznie się proces nagrywania. W trakcie jego trwania program będzie wyświetlał okno z widocznym paskiem postępu (patrz rysunek 13.20).

Rysunek 13.20.

Okno dialogowego z paskiem postępu wyświetlane podczas nagrywania płyty



Kiedy nagrywanie płyty CD zostanie zakończone, program wysunie tackę z napędu i wyświetli komunikat informujący, czy nagrywanie zakończyło się sukcesem, czy nie. Należy kliknąć przycisk *Zamknij*. Użytkownik może zdecydować, czy chce nagrać kolejną płytę z tą samą zawartością, wystarczy wówczas umieścić w napędzie nową, pustą płytę i kliknąć odpowiedni przycisk. Jeżeli na kolejnej płycie mają znaleźć się nowe utwory, wówczas należy kliknąć przycisk *Wyczyść* znajdujący się na pasku narzędzi programu i wprowadzić nowe piosenki na listę. Jeżeli sesja nagraniowa została zakończona, trzeba zamknąć program, wybierając z menu *Plik/Zakończ*.

Inne źródła dźwięku

Obecnie, dzięki internetowi, przenośnym odtwarzaczom muzycznym, podcastom i ilości wolnego miejsca na przechowywanie plików dźwiękowych, słuchanie muzyki za pomocą komputera nie ogranicza się tylko do odtwarzania płyt CD. Dostępna jest ogromna różnorodność rozmaitych źródeł muzyki: transmisje strumieniowe w internecie, podcasty, które

można pobrać na dysk lokalny (są niczym innym, jak plikami MP3), czy pliki zgrane z płyt CD (które pełnią rolę kopii zapasowych). Poniżej wyjaśniam, jak korzystać z programu Rhythmbox do odtwarzania dźwięków ze wszystkich tych źródeł, a także omawiam doskonale program do konwertowania plików dźwiękowych z ich oryginalnego formatu do takiego, który jest przyjaźniejszy dla komputera.

Pliki dźwiękowe i internetowe radio w Rhythmboxie

Rhythmbox to aplikacja GNOME służąca do odtwarzania muzyki w wielu różnych formatach, takich jak internetowe stacje radiowe, pliki zapisane na dysku komputera itp.

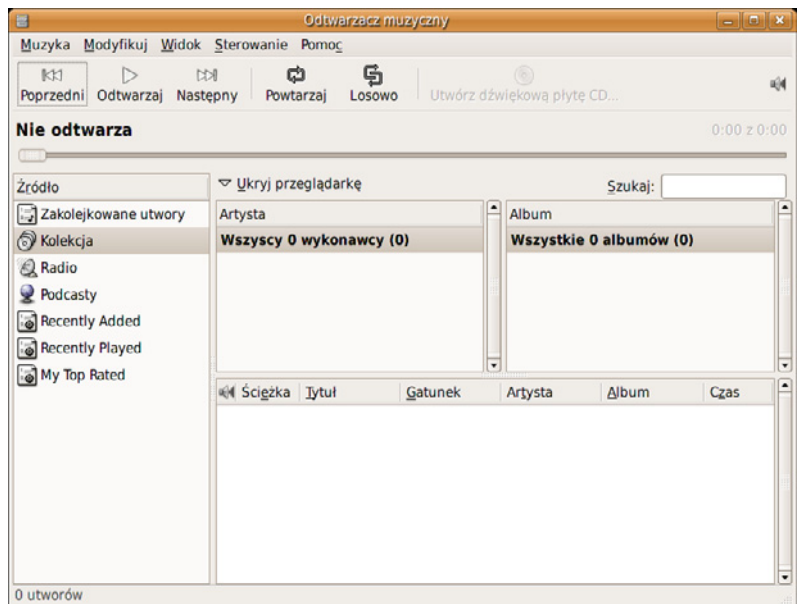
Aby uruchomić program, należy z menu wybrać kolejno *Aplikacje/Dźwięk i obraz/Odtwarzacz muzyki Rhythmbox*.



Podczas pierwszego uruchomienia programu użytkownik zostanie przeprowadzony przez proces konfiguracji, który ma na celu wskazanie folderów, w jakich przechowywane są pliki dźwiękowe. Ponieważ jest to proces jednorazowy, a ponadto można go uruchomić w późniejszym czasie już wprost z programu, sugeruję pominięcie tego etapu (należy kolejno kliknąć przyciski *Do przodu*, następnie zaznaczyć pole wyboru *Pomiń ten krok*, a na końcu *Zastosuj*). W dalszej części rozdziału zamieszczam informacje o konfigurowaniu biblioteki muzycznej.

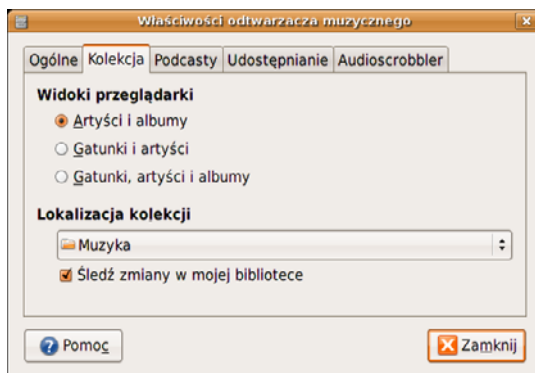
Na rysunku 13.21 przedstawiam interfejs programu z pustą biblioteką audio, bez pobranych podcastów i innych źródeł dźwięków.

Rysunek 13.21.
Rhythmbox przed importem plików dźwiękowych



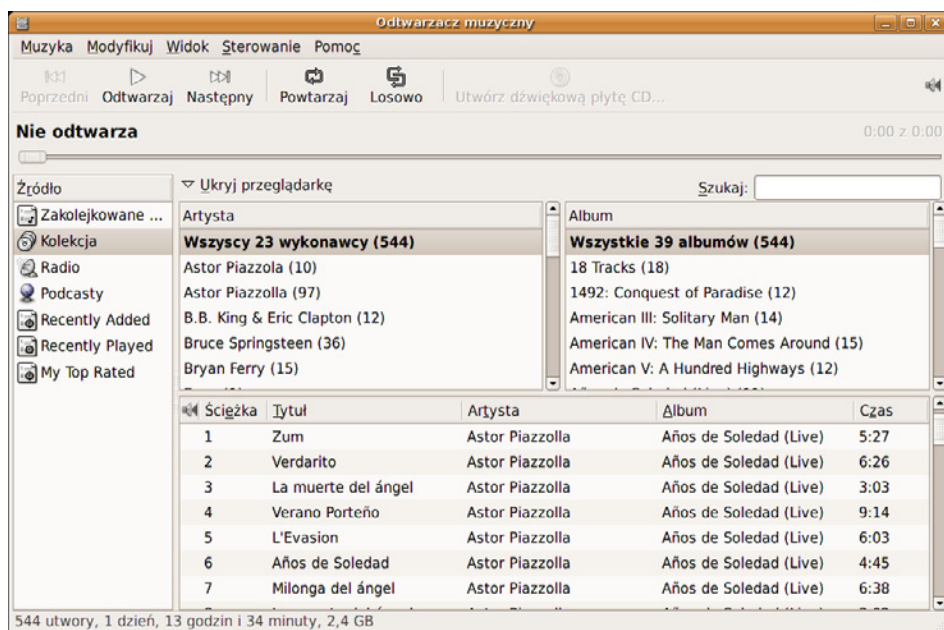
Aby wskazać programowi, gdzie są przechowywane pliki audio, należy wybrać z menu *Modyfikuj/Preferencje*; następnie trzeba przejść do zakładki *Kolekcja* (patrz rysunek 13.22).

Rysunek 13.22.
Określanie położenia
plików audio



Należy kliknąć listę rozwijaną *Lokalizacja kolekcji*, zostaną wówczas wyświetlone foldery, w których program wyszuka pliki; jeżeli znajdują się one w katalogu, który nie znajduje się na liście, wówczas należy wybrać pozycję *Inne* i wskazać odpowiedni folder.

Kiedy odpowiedni plik zostanie wskazany, trzeba zaznaczyć pole *Śledź zmiany w mojej bibliotece*, dzięki czemu wszystkie nowe pliki pojawiające się w wybranym folderze będą automatycznie dodawane do programu. Następnie należy kliknąć przycisk *Zamknij*. Rozpocznie się proces importowania plików do programu, a zaktualizowane informacje zostaną wyświetlone w jednym z paneli. W razie napotkania na jakiegokolwiek problemy, informacje o tym zostaną wyświetlone w osobnym oknie, a wtedy będzie można je sprawdzić. Zwykle problemy dotyczą uszkodzonych plików, w najgorszym przypadku będzie można je usunąć i zgrać raz jeszcze z płyt CD. Na rysunku 13.23 przedstawiam okno programu Rhythmbox po zakończeniu importowania plików.



Rysunek 13.23. Program po utworzeniu biblioteki utworów

Po imporcie zgromadzonych w komputerze plików muzycznych do biblioteki programu można zauważyć, że dla każdego utworzono wpis w kolekcji oraz odwzorowano hierarchię katalogów. Aby odtworzyć daną płytę (album), należy zaznaczyć ją w panelu *Album*, a następnie kliknąć przycisk *Odtwórz*. Aby wyświetlić wszystkie albumy danego wykonawcy, trzeba zaznaczyć go w panelu *Artysta*, wówczas zmieni się odpowiednio zawartość panelu *Albumy*, a wówczas można wybrać dowolną płytę i użyć przycisku *Odtwórz*. Można też wybrać określone utwory z danego albumu, wystarczy tylko zaznaczyć je w dolnym panelu i kliknąć *Odtwórz*.

Tworzenie list odtwarzania

Jeżeli czytelnicy, tak jak ja, posiadają obszerne kolekcje muzyczne, to zapewne na każdym albumie mają tylko kilka ulubionych utworów. Program Rhythmbbox umożliwia tworzenie własnych kolekcji „największych przebojów” za pomocą mechanizmu list odtwarzania, w których umieszcza się wskazane utwory z wybranych albumów.

Aby utworzyć nową listę odtwarzania i dodać do niej nowe piosenki, należy kolejno z menu programu wybrać *Muzyka/Lista odtwarzania/Nowa lista odtwarzania* (lub użyć skrótu klawiaturowego *Ctrl+N*). Zostanie wówczas wyświetlona nowa lista odtwarzania w panelu *Źródło*. Ponieważ nie ma ona żadnej nazwy, należy kliknąć ją prawym przyciskiem myszy i wpisać dowolną. Aby zmienić tę nazwę w późniejszym czasie, trzeba kliknąć daną listę lewym przyciskiem myszy i z menu kontekstowego wybrać opcję *Zmień nazwę*, a następnie wprowadzić nową i wcisnąć klawisz *Enter*.

Aby do nowej listy dodać piosenki, wystarczy podczas przeglądania zgromadzonej kolekcji przeciągać wybrane utwory do nowej listy. Na rysunku 13.24 przedstawiam listę odtwarzania zawierającą kilka wybranych utworów jednego artysty. Dzięki możliwości tworzenia list odtwarzania użytkownik może w prosty sposób przygotować zestawy utworów odpowiednie na najróżniejsze okazje.

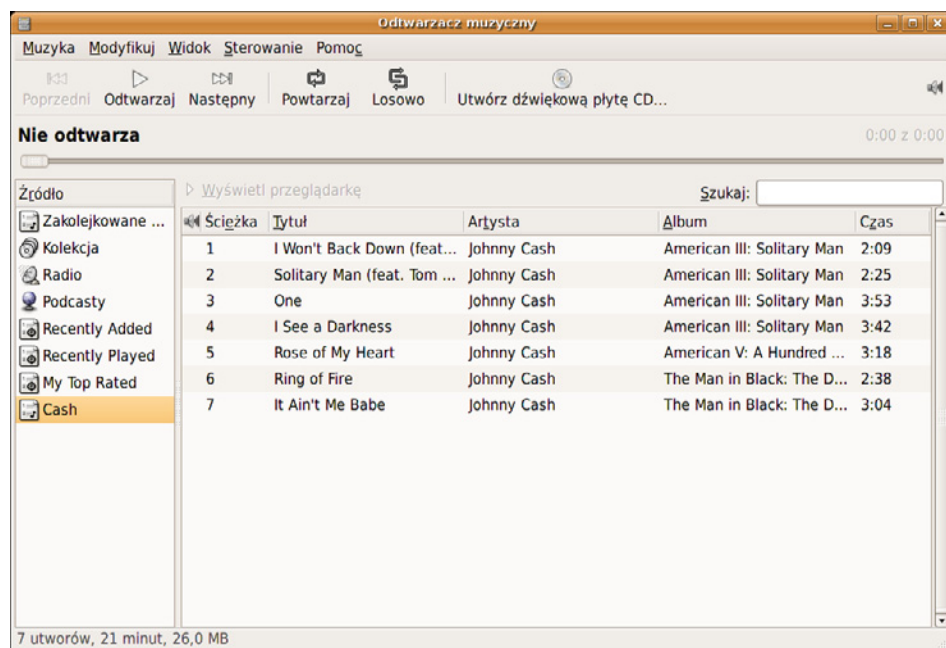
Aby usunąć dany utwór z listy, należy wywołać menu kontekstowe (prawym przyciskiem myszy) i wybrać z niego *Usuń*. To spowoduje usunięcie danego utworu tylko z listy odtwarzania, a nie z kolekcji, a więc utwór będzie można odtworzyć w dowolnym momencie.

Odtwarzanie internetowych stacji radiowych

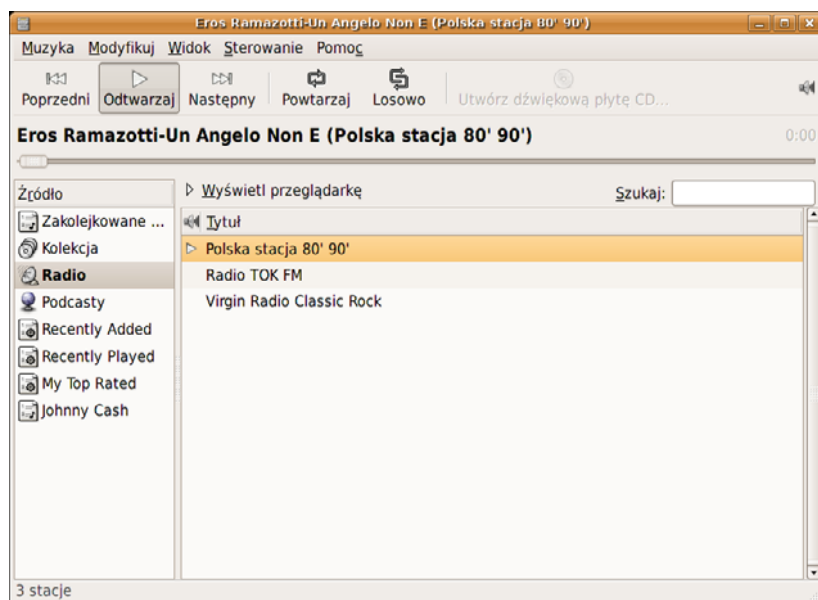
Internetowe radio to rewolucja z zakresie nadawania audycji, ponieważ można słuchać wybranej stacji, niezależnie od miejsca, w którym użytkownik się znajduje, i bez względu na to, czy jest w zasięgu nadawania danej rozgłośni, czy też nie.

Aby dodać nową stację do programu, należy wskazać pozycję *Radio* w panelu *Źródło*, a następnie wybrać z menu kontekstowego pozycję (zresztą jedyną) *Nowa stacja radia internetowego*. Zostanie wyświetlone okno dialogowe, w którym użytkownik może wprowadzić adres URL wybranej stacji. Następnie należy jeszcze kliknąć przycisk *Dodaj* (patrz rysunek 13.25).

Aby wyświetlić wszystkie dostępne w programie stacje radiowe, należy kliknąć pozycję *Radio* znajdującą się w panelu *Źródło*; dwukrotne kliknięcie wybranej rozgłośni spowoduje odtworzenie nadawanego przez nią programu.



Rysunek 13.24. Dostosowywanie list odtwarzania

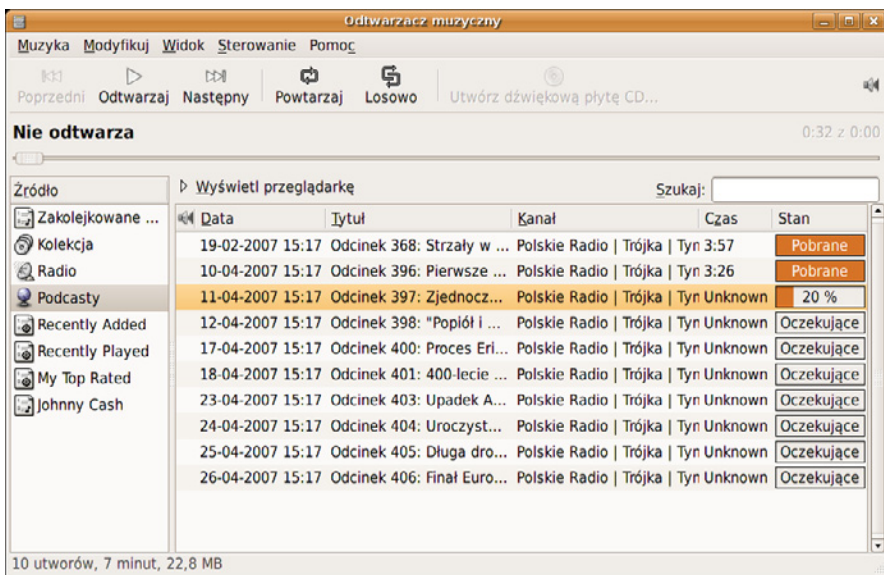


Rysunek 13.25. Odtwarzanie radia internetowego w programie Rhythmbox

Podcasty

Podcasty to kolejna technologia, która przyczyniła się do rewolucji w świecie mediów, ponieważ umożliwia każdemu dzielenie się własnymi opiniami z innymi. Podcasty można łatwo subskrybować i odtworzyć w dogodnym dla użytkownika momencie. Jest to możliwość szczególnie przydatna dla osób podróżujących, ponieważ uzyskują możliwość stałego napływu informacji, kościelnych kazań i innych wiadomości z rodzinnego miasta.

Aby subskrybować kanał podcastów, należy przejść do panelu *Źródło*, zaznaczyć pozycję *Podcasty*, kliknąć prawym przyciskiem myszy i z menu wybrać opcję *Nowy kanał podcastów*. Zostanie wyświetlone okno dialogowe, w którym użytkownik może wprowadzić adres URL wybranego kanału. Następnie należy jeszcze kliknąć przycisk *Dodaj* (patrz rysunek 13.26).



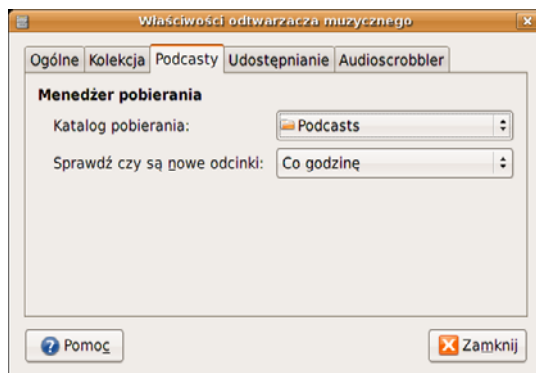
Rysunek 13.26. Dodawanie i odtwarzanie podcastów w programie Rhythmbox

Następnie można kliknąć dodany kanał i sprawdzić, czy są dostępne nowe podcasty; jeżeli tak, wówczas należy kliknąć prawym przyciskiem myszy wybrany plik i wybrać z menu *Ściągnij odcinek*. Kiedy plik zostanie już pobrany na dysk komputera, można go odtworzyć, wystarczy dwukrotnie kliknąć.

Lokalizację, w której zapisywane są pobrane pliki, oraz częstotliwość ich pobierania użytkownik może dostosować w oknie *Właściwości odtwarzacza muzycznego*. Aby je wyświetlić, należy wybrać z menu *Modyfikuj/Preferencje*, a następnie wskazać odpowiednie opcje z list rozwijanych. Na rysunku 13.27 przedstawiam opisywane okno.

Rysunek 13.27.

Konfigurowanie
ustawień podcastów
w programie
Rhythmbox



Konwertowanie plików audio

Zawsze może się zdarzyć, że kiedyś użytkownik pobierze z internetu pliki audio, czy też zgra je z płyty audio CD, po czym okaże się, że z jakiegoś powodu mają one „nie-właściwy” format. Najlepszymi przykładami są następujące sytuacje:

- ♦ po zgraniu płyt CD do formatu Ogg Vorbis okazuje się, że nie można ich odtworzyć na iPodzie, ponieważ nie korzysta z kodeków OGG;
- ♦ po pobraniu nagrania koncertu z internetu w formacie FLAC okazuje się, że obecnie ten format nie jest używany w iPodach (i prawdopodobnie ze względu na rozmiary plików nigdy nie będzie);
- ♦ podczas próby nagrania płyty audio CD do samochodu okazuje się, że można do tego wykorzystać tylko pliki w formacie MP3, OGG i FLAC.

Niezależnie od tego, która z wymienionych sytuacji jest najbardziej prawdopodobna, tak samo irytujące jest posiadanie muzyki, której nie można odsłuchać w odtwarzaczu czy nagrać na płytę CD. Problemem, który powyższe przykłady unaocznily, jest fakt, że różne urządzenia wykorzystują odmienne formaty plików dźwiękowych. Urządzenia produkowane przez firmę Apple odtwarzają pliki w formacie MP3 oraz własnym formacie Apple, na płytach CD audio pliki muszą być zapisane w formacie WAV itd. Na szczęście, w Ubuntu mamy znakomite narzędzie, które pomoże rozwiązać ten problem — jest to program o nazwie Sound Converter i, jak nazwa wskazuje, służy do konwersji plików audio z jednego formatu do innego.

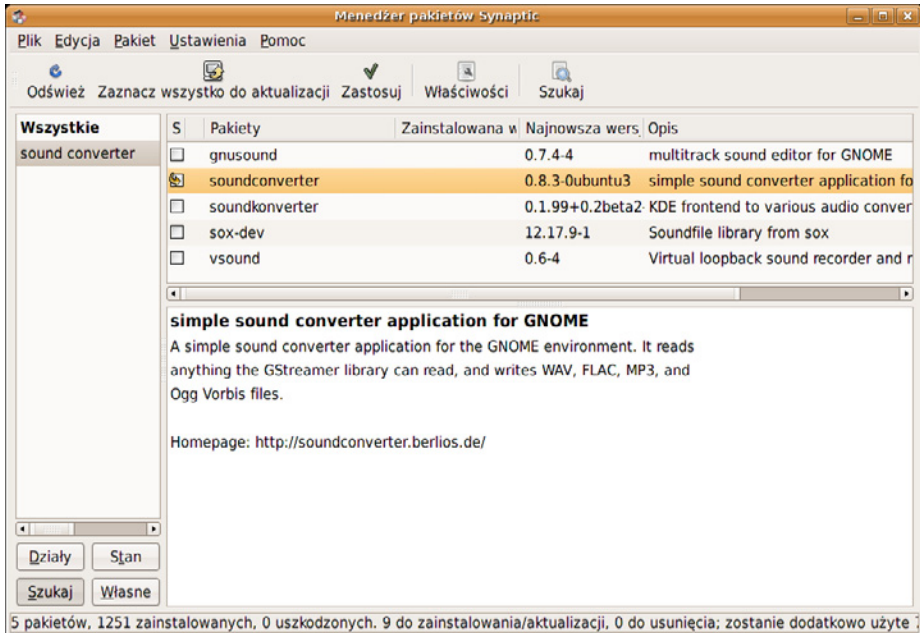


Program Sound Converter znajduje się w repozytoriach *universe* i do działania potrzebuje GStreamer (omówionego wcześniej w podrozdziale „Instalowanie GStreamer i wtyczek”. Przed zainstalowaniem Sound Convertera należy wykonać opisane tam czynności.

Instalowanie programu Sound Converter

Ponieważ nie wszyscy stosują Ubuntu do pracy z dźwiękiem, dlatego też Sound Converter nie jest instalowany domyślnie w systemie. Aby zainstalować program, należy najpierw uruchomić menedżer pakietów Synaptic (menu *System/Administracja*), następnie wybrać sekcję *Multimedia (universe)* i przewinąć (lub od razu wyszukać) do pakietu o nazwie

soundconverter. Trzeba zaznaczyć pakiet i z menu kontekstowego wybrać opcję instalacyjną (patrz rysunek 13.28).



Rysunek 13.28. Instalowanie programu Sound Converter za pomocą menedżera pakietów Synaptic

W zależności od oprogramowania już zainstalowanego w systemie, a także wersji pakietów, program może wskazać dodatkowe pliki, które muszą zostać dodane do Ubuntu. Zostanie wyświetlona ich lista, a użytkownik poproszony o zgodę na ich zainstalowanie. Należy kliknąć przycisk *Zaznacz wszystkie*.

Następnie trzeba kliknąć przycisk *Zastosuj* znajdujący się na pasku narzędzi Synaptica; po zakończeniu procesu instalacji pliki audio można konwertować na dowolne formaty.

W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, zamieszczam dodatkowe informacje dotyczące użytkowania menedżera pakietów Synaptic.

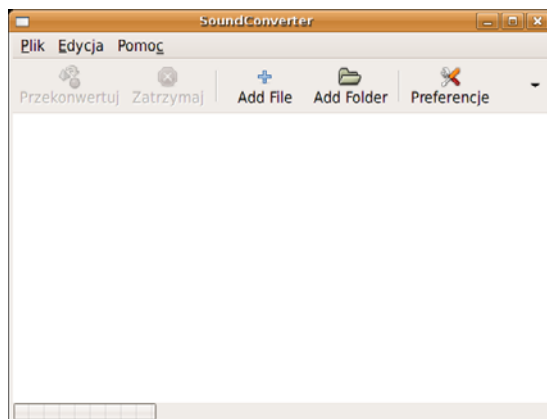
Korzystanie z programu Sound Converter

Kiedy program zostanie już zainstalowany, można go uruchomić, wybierając z menu kolejno *Aplikacje/Dźwięk i obraz/Sound Converter*. Główne okno dialogowe widoczne jest na rysunku 13.29.

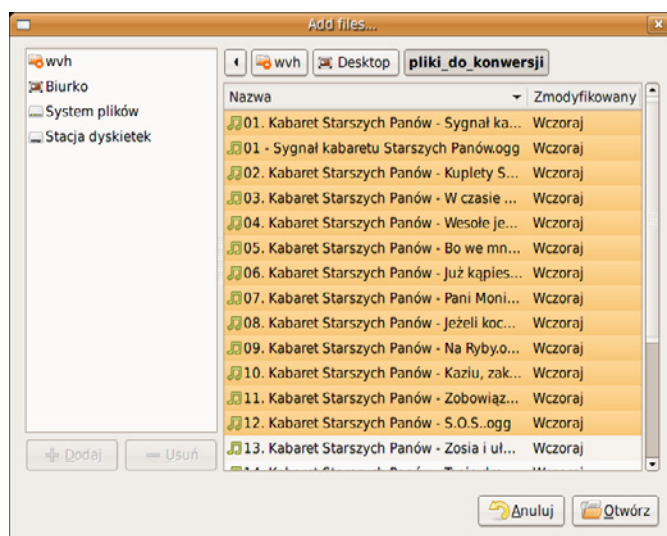
Aby wybrać pliki, które mają zostać poddane konwersji, należy kliknąć przycisk *Add File*, znajdujący się na pasku narzędzi programu; by dodać cały folder, należy skorzystać z przycisku *Add Folder* (patrz rysunek 13.30).

Rysunek 13.29.

Uruchamianie
programu Sound
Converter

**Rysunek 13.30.**

Wskazywanie plików,
które mają zostać
poddane konwersji



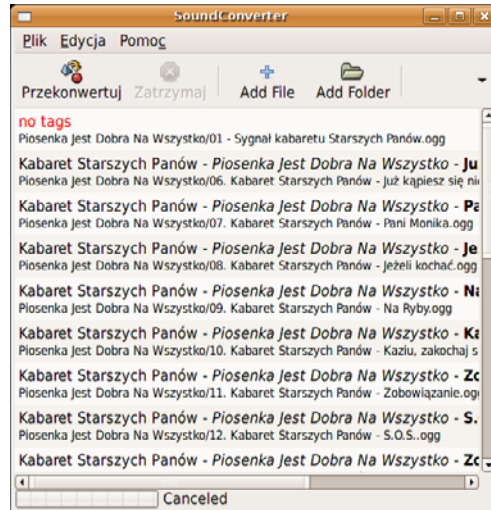
W tym oknie można przejść i zaznaczyć odpowiednie pliki, które mają zostać poddane konwersji do innego formatu. Po zaznaczeniu odpowiednich plików należy kliknąć przycisk *Otwórz*, okno nawigacyjne zostanie wówczas zamknięte. Wybrane uprzednio pliki zostaną wyświetlone w głównym oknie programu (patrz rysunek 13.31).

Teraz trzeba określić ustawienia procesu konwertowania plików. Na rysunku 13.31 widać, że wszystkie wybrane pliki mają format Ogg, natomiast potrzebne są w formacie MP3. To żaden problem — wystarczy określić format docelowy; aby to zrobić, należy kliknąć przycisk *Preferencje*, zostanie wówczas wyświetlone okno widoczne na rysunku 13.32.

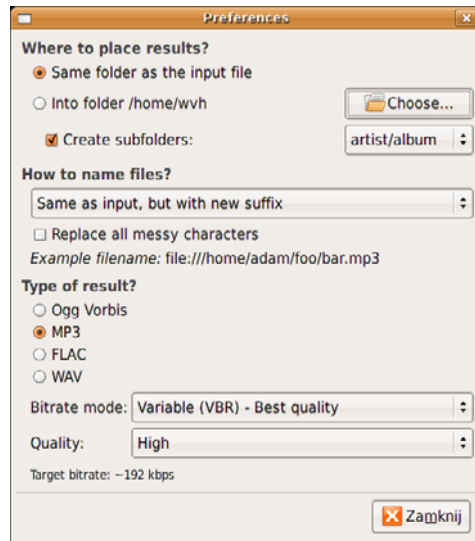
W części okna *Where to place results?* (widocznej na rysunku 13.32) można określić, gdzie będą zapisane utworzone pliki (zwykle pozostawiam je w tym samym folderze, dzięki czemu łatwo je później odnaleźć). W części *Type of result?* umożliwiono wybranie formatu docelowego plików (w przykładzie wybrano format MP3), a także jakości plików. Po wprowadzeniu wszystkich ustawień należy kliknąć przycisk *Zamknij*; okno zostanie zamknięte, a proces konwertowania plików rozpoczęty.

Rysunek 13.31.

*Plik przeznaczone
do zmiany formatu*

**Rysunek 13.32.**

*Konfigurowanie
konwersji plików*



Po jego zakończeniu utworzone pliki będzie można np. przenieść do iPod'a, wypalić na płycie itp.



Jednym z kilku nieobsługiwanych w Ubuntu formatów plików audio jest Shorten (SHN), wolny, bezstratny kodek stosunkowo popularny w kręgach osób zajmujących się nagraniami na żywo. Aby z niego korzystać, należy zainstalować wtyczkę do programu xmms (sam program też). Innym rozwiązaniem jest instalacja za pomocą menedżera pakietów Synaptic narzędzia linii poleceń o nazwie `shntool`, które umożliwi przekonwertowanie ich do formatu WAV, z którym z kolei poradzi sobie program Sound Converter.

Płyty DVD

Jedyną rzeczą, która jest w stanie wywołać nostalgię za kasetami wideo jest fakt, że obecnie koncerny medialne bardzo źle traktują użytkowników DVD. Kontrola regionalna, licencjonowane kodeki, szyfrowanie danych. To tak, jakby zakazać naukowcom mówić, ponieważ są mądrzejsi od personelu administracyjnego.

Wszystkie te powody (a także inne) sprawiają, że odtwarzanie płyt DVD w komputerach nie jest rzeczą łatwą. Na szczęście, kłopotom tym można zaradzić. Dzięki oferowanemu w Ubuntu programowi Totem, wspartemu omówionym wcześniej GStreamerem, możliwe jest odtwarzanie płyt DVD. Poniżej opisuję, jak skonfigurować i używać programu Totem, który cały ten proces upraszcza.

Zamieszczam też instrukcje, w jaki sposób zgrywać płyty DVD na dysk komputera; jest to nawet prostsze niż ich odtwarzanie, ponieważ do tego celu wystarczy popularne narzędzie linii poleceń `dd`, dostępne w niemal każdym Linuksie. Tworzenie obrazów ISO istniejących płyt DVD (np. w celu wykonania kopii zapasowej) za pomocą polecenia `dd` jest tylko nieco wolniejsze, a na pewno tańsze od kupowania drugiego egzemplarza.

Na końcu zamieszczono wskazówki dotyczące kopiowania i nagrywania płyt DVD z obrazów ISO oraz umieszczania plików na dyskach DVD. Do każdego z tych zadań zdecydowanie polecam wykorzystanie programu K3b. Już niemal sływać okrzyki sprzeciwu fanatyków GNOME, ponieważ K3b to aplikacja przeznaczona dla środowiska KDE. No i co z tego? Również jestem fanem GNOME i przedkładam jego prostotę oraz łatwość użytkowania nad KDE, ale jednocześnie jestem jeszcze większym fanem używania odpowiednich narzędzi do odpowiednich zadań. Fakt jest następujący: K3b jest lepszy od jakiegokolwiek innego narzędzia używanego do pracy z płytami DVD w GNOME. K3b pozwala też na nagrywanie płyt CD, ale ich odpowiedniki w GNOME są w tym przypadku wystarczające. Tyle o K3b, teraz pora na odtworzenie płyt DVD.

Odtwarzanie płyt DVD

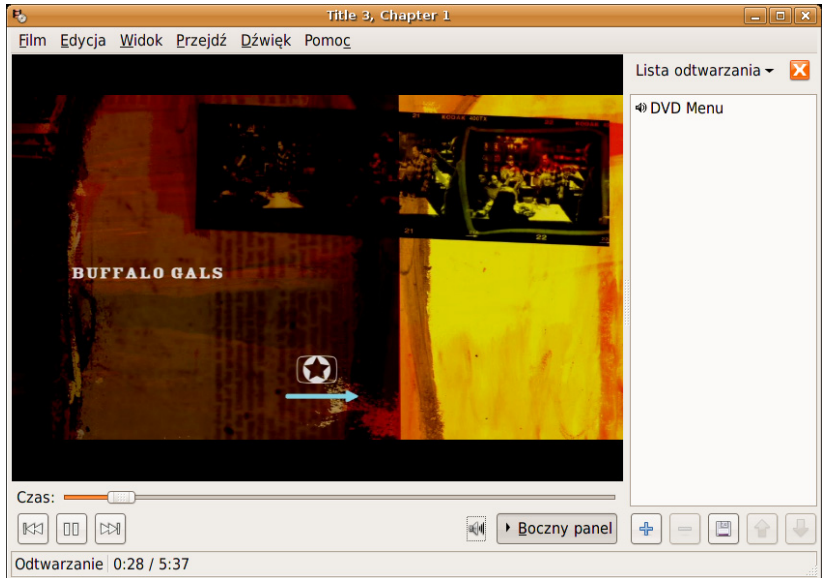
Totem (<http://www.gnome.org/projects/totem/>) to aplikacja GNOME w pełni przygotowana do odtwarzania DVD i wideo, korzystająca z technologii, takich jak GStreamer (domyślnie) lub Xine (kolejny odtwarzacz oraz kodek). Totem jest domyślnie instalowany w Ubuntu, podobnie zresztą jak we wszystkich dystrybucjach wykorzystujących środowisko GNOME.

Odtwarzanie DVD za pomocą Totema jest proste. Ponieważ w większości sprzedawanych płyt DVD z nagraniami umieszczono jakiś mechanizm szyfrowania danych, przed próbą ich odtworzenia należy zainstalować GStreamera oraz powiązane z nim biblioteki. Więcej informacji na ten temat zamieszczam w podrozdziale „Instalowanie GStreamera i wtyczek”.

Jak wyjaśniałem wcześniej, środowisko GNOME jest w Ubuntu skonfigurowane tak, że po umieszczeniu nagranej płyty w napędzie automatycznie uruchamia program do odtwarzania multimedialnych — w tym przypadku jest to Totem. Dlatego też, jeżeli użytkownik nie zmienił domyślnych ustawień, aby odtworzyć płytę DVD, wystarczy umieścić ją w napędzie, a zostanie wyświetlone okno widoczne na rysunku 13.33 (oczywiście, zawartość będzie zależeć od umieszczonej płyty).

Rysunek 13.33.

Odtwarzanie płyty
DVD w Ubuntu



Program Totem można też uruchomić samodzielnie; wystarczy tylko wybrać go z menu *Aplikacje/Dźwięk i obraz/Odtwarzacz filmów Totem* lub wpisać polecenie *totem* w terminalu GNOME lub xterm.

Podobnie jak ma to miejsce w większości graficznych aplikacji GNOME, również odtwarzacz Totem można obsługiwać za pomocą skrótów klawiszowych, oto kilka najczęściej wykorzystywanych:

- ♦ **B** — powrót do poprzedniego rozdziału lub filmu na płycie;
- ♦ **F** — przełączenie do trybu pełnoekranowego; w trybie pełnoekranowym w górnym prawym rogu wyświetlana jest ikona, która umożliwia powrót do trybu okienkowego widocznego na rysunku 13.33;
- ♦ **H** — ukrywane są wszelkie elementy sterujące; aby je wyświetlić, należy ponownie wcisnąć klawisz **H**;
- ♦ **N** — przejście do następnego rozdziału lub filmu na płycie;
- ♦ **F9** — ukrywanie i przywoływanie bocznego panelu;
- ♦ **Page-Up** — zwiększanie głośności;
- ♦ **Page-Down** — zmniejszanie głośności.

Po zakończeniu seansu należy wybrać z menu *Film/Wysuń płytę* (lub użyć skrótu klawiszowego **Ctrl+E**), dzięki czemu będzie można wyjąć nośnik z napędu. Jeżeli użytkownik będzie chciał obejrzeć kolejny film, wówczas po umieszczeniu kolejnej płyty w napędzie należy użyć opcji *Film/Otwórz*. Aby zamknąć program, należy wybrać z menu *Film/Zakończ*.



Odtwarzacz Totem jest wciąż w fazie rozwoju, dlatego niektóre wersje mogą mieć problemy z odtwarzaniem płyt DVD za pomocą menu. Jeżeli program wyświetli komunikat informujący, że nie jest w stanie odnaleźć odpowiedniego kodeku potrzebnego do odtworzenia filmu (po użyciu polecenia *Film/Otwórz*), należy go wyłączyć, wysunąć tackę z płytą z napędu, a następnie wsunąć ją ponownie i pozwolić, aby system automatycznie uruchomił odtwarzacz raz jeszcze.

Zgrywanie płyt DVD

W repozytoriach Ubuntu i w internecie można znaleźć wiele aplikacji służących do zgrywania płyt DVD w GNOME, ale większość z nich korzysta z tego samego narzędzia. Jako geek z długim doświadczeniem w Uniksie i Linuksie po otrzymaniu płyty DVD moim pierwszym pragnieniem jest wykonaniem jej kopii, tak aby dysponować jeszcze jednym egzemplarzem kiedy nasz rottweiler zakończy korzystanie z oryginału. Moim ulubionym narzędziem do zgrywania płyt DVD jest:

```
$ dd if=/dev/dvd of=nazwa_pliku_wyjsciowego.iso
```

Polecenie jest proste, działa w każdym Linuksie, a w efekcie jego pracy uzyskuje się plik obrazu identyczny z oryginalnym DVD. Plik wyjściowy można później zamontować, tak jak napęd DVD; w tym celu należy użyć następujących poleceń:

```
$ sudo mkdir /mnt/dvd
$ sudo mount -o loop iso9660 nazwa_pliku_wyjsciowego.iso /mnt/dvd
```

Pierwsze polecenie tworzy katalog, w którym będzie zamontowany plik ISO; drugie montuje obraz pliku w systemie i czyni go dostępnym.

Kiedy już za pomocą polecenia *dd* zostanie utworzony obraz płyty, wówczas można go wykorzystać w programie *K3b* i nagrać na płytę DVD-R. Więcej informacji na ten temat w dalszej części rozdziału.



Jeżeli użytkownik chce koniecznie zgrać płytę w inny sposób, niż oferuje to narzędzie *dd*, wówczas powinien skorzystać z narzędzia *dvd::rip*, które dostępne jest w postaci pakietu w repozytoriach Ubuntu (pod nazwą *dvdrip*). Zostało ono napisane w języku Perl, a zatem można się co nieco po nim spodziewać. Podczas instalacji do systemu zostanie dodanych od 10 do 20 dodatkowych pakietów, koniecznych do działania programu, ale to nie koniec. Użytkownik będzie musiał samodzielnie zainstalować kolejne pakiety, takie jak *xcdroast*, *mplayer* itd. Przed pobraniem i zainstalowaniem programu warto dowiedzieć się więcej o jego wymaganiach i działaniu, te informacje można znaleźć na stronie <http://www.exit1.org/dvdrip/>.

Używanie programu K3b do kopiowania i nagrywania płyt DVD

Jak już wspominałem we wstępie, przeznaczony dla środowiska KDE program *K3b* jest — moim zdaniem — najlepszym i najprostszym narzędziem linuksowym do kopiowania płyt DVD, nagrywania obrazów ISO na te płyty i do nagrywania plików audio czy wideo. Zainstalowanie go w systemie wymaga dodania wielu bibliotek, ale możliwości oferowane przez ten program są warte poświęcenia na to czasu i przestrzeni dyskowej. Poza tym, biorąc pod uwagę rozmiary dzisiejszych dysków, te sto megabajtów to naprawdę niewiele.

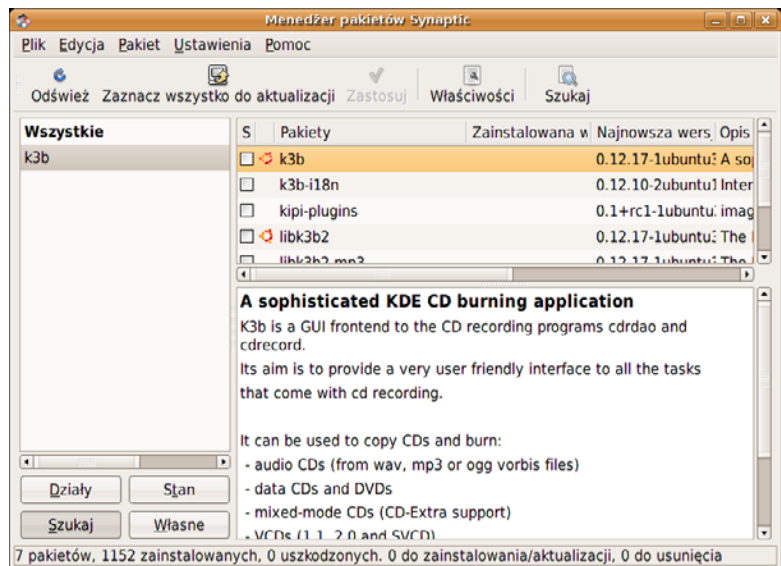
W pozostałej części rozdziału napisałem, jak zainstalować program i wykonać za jego pomocą podstawowe operacje (takie jak kopiowanie i nagrywanie płyt DVD). Program, oczywiście, może o wiele więcej, ale ograniczymy się do wyjaśnienia podstawowych zagadnień. K3b posiada dobrą dokumentację, można też znaleźć wiele stron internetowych prowadzonych przez jego miłośników (kilka z nich wskazuję na końcu rozdziału), które też mogą być doskonałym źródłem informacji.

Instalowanie programu K3b

Aby zainstalować program K3b, należy uruchomić menedżera pakietów Synaptic (menu *System/Administracja*), a następnie kliknąć przycisk *Szukaj* i wpisać tam ciąg k3b. Kiedy wyszukiwanie zostanie zakończone, wyniki powinny wyglądać tak, jak przedstawione na rysunku 13.34.

Rysunek 13.34.

Wybór pakietów
do instalacji
programu K3b



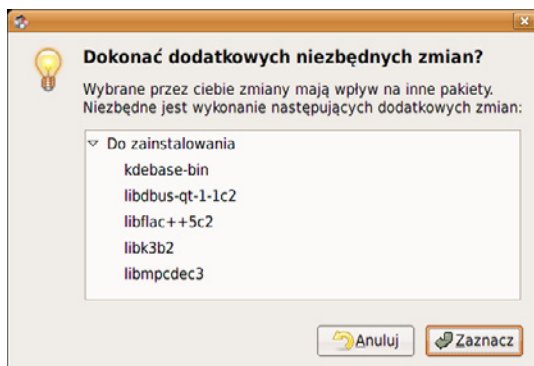
Po zaznaczeniu pakietu program wyświetli okno z informacją o dodatkowych pakietach potrzebnych w systemie i poprosi użytkownika o zgodę na ich pobranie. Nie należy dziwić się ich liczbie, w końcu to aplikacja środowiska KDE, dlatego aby K3b mogło szczęśliwie działać, potrzebny jest pokaźny zestaw bibliotek KDE i programów pomocniczych. Aby zaakceptować instalację, należy kliknąć przycisk *Zaznacz* (patrz rysunek 13.35).

Po wybraniu i zaznaczeniu wszystkich niezbędnych pakietów należy kliknąć przycisk *Zastosuj*. Zostanie wyświetlone okno z informacjami podsumowującymi zmiany, które będą wprowadzone, należy kliknąć *Zastosuj*. Po pobraniu i zainstalowaniu wszystkich zaznaczonych pakietów można zamknąć program Synaptic.

W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, zamieszczono dodatkowe informacje dotyczące korzystania z menedżera pakietów Synaptic.

Rysunek 13.35.

*Dodatkowe pakiety
niezbędne
do zainstalowania K3b*

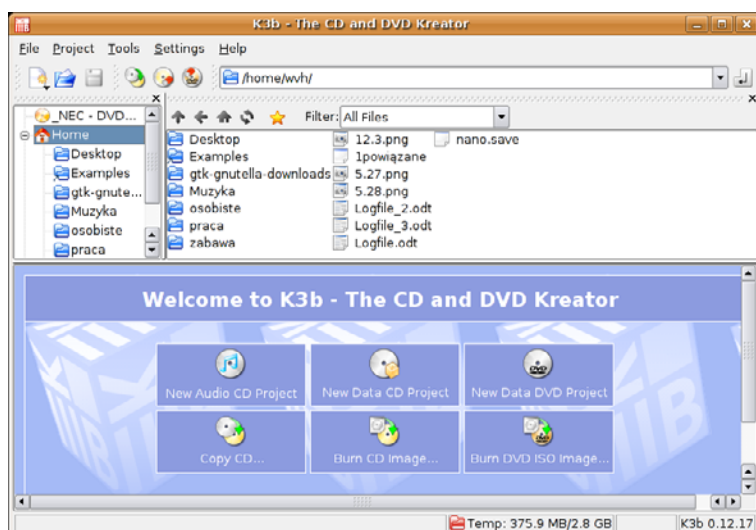


Nagrywanie obrazów ISO na płyty DVD w programie K3b

Po zainstalowaniu programu, najprostszym sposobem jego uruchomienia jest wybranie kolejno z menu *Aplikacje/Dźwięk i obraz/K3b*. Zostanie wówczas wyświetlony ekran startowy widoczny na rysunku 13.36.

Rysunek 13.36.

*Ekran startowy
programu K3b*

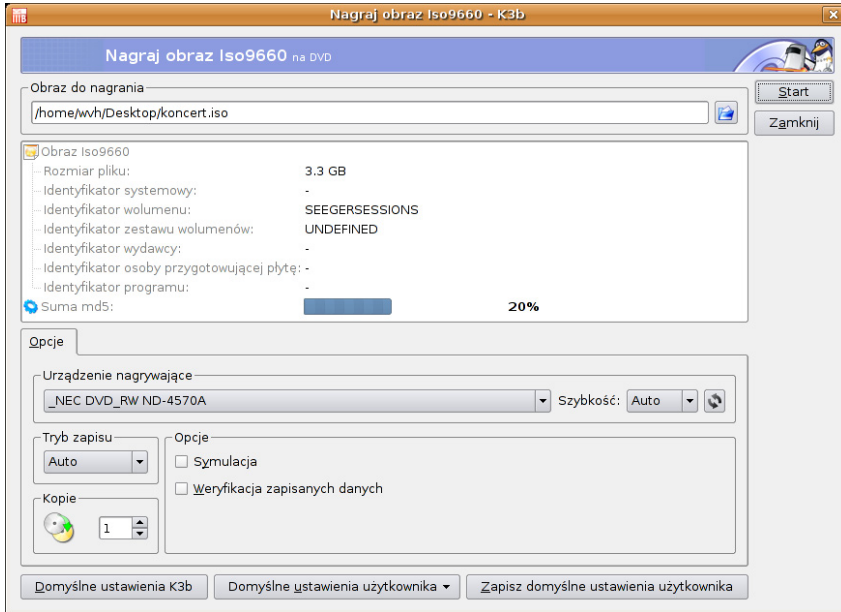


Aby nagrać plik ISO na płytę DVD, należy z menu programu wybrać kolejno *Narzędzia/Nagraj DVD z pliku ISO*, a zostanie wyświetlone okno widoczne na rysunku 13.37.

Aby przejść do folderu, gdzie umieszczony jest plik ISO, należy kliknąć ikonę katalogu umieszczoną na prawo od pola tekstowego *Obraz do nagrania*, zostanie wówczas wyświetlone okno widoczne na rysunku 13.38.

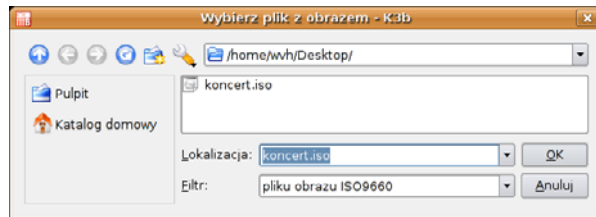
Kiedy odpowiedni plik zostanie już wskazany, trzeba kliknąć *OK*, aby zamknąć okno. Program obliczy sumę kontrolną pliku; za jej pomocą można sprawdzić integralność.

W przypadku posiadania więcej niż jednej nagrywarki w systemie, należy się upewnić, że w polu *Urządzenie nagrywające* wybrany jest odpowiedni napęd. Domyślnie program



Rysunek 13.37. Nagrywanie pliku ISO na dysk DVD — okno dialogowe programu

Rysunek 13.38.
Wskazanie pliku ISO
do nagrania



wykrywa wszystkie dostępne nagrywarki CD oraz DVD i wyświetla okno z możliwością wyboru jednej z nich.

Aby rozpocząć nagrywanie obrazu na płytę, należy umieścić nośnik w odpowiednim napędzie i kliknąć przycisk *Start*. Można też przeprowadzić symulację zapisu (wystarczy zaznaczyć pole wyboru *Symulacja*), dzięki czemu program wykryje potencjalne problemy (patrz rysunek 13.39).

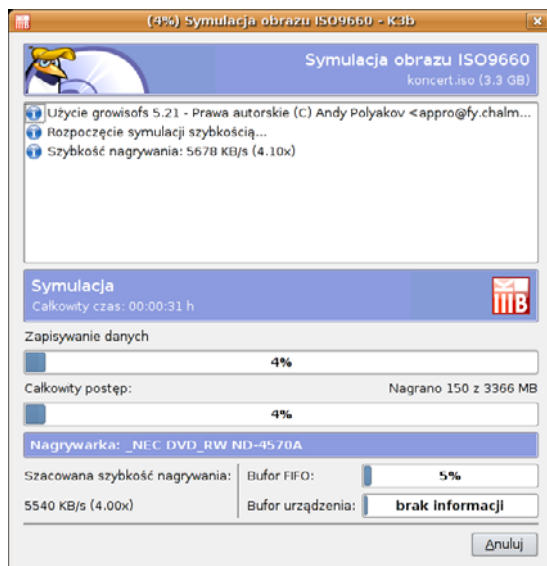
Po zakończeniu procesu nagrywania program wyświetli okno dialogowe, w którym użytkownik będzie mógł sprawdzić komunikaty systemowe, co jest przydatne podczas analizowania przyczyn błędów. Zwykle jednak wystarczy użyć przycisku *Zamknij*. Ponownie zostanie wyświetlone okno widoczne na rysunku 13.37; aby powrócić do głównego okna programu, należy kliknąć przycisk *Zamknij*.

Kopiowanie płyt DVD za pomocą programu K3b

Proces kopiowania płyt w programie K3b składa się z kilku etapów: najpierw program tworzy plik ISO, wysuwa oryginalną płytę i prosi o umieszczenie w napędzie pustego nośnika, a następnie zapisuje plik ISO na płycie DVD.

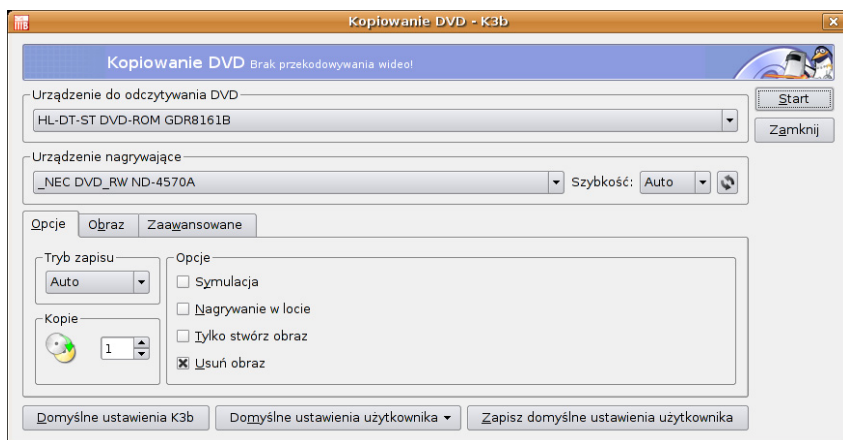
Rysunek 13.39.

*Symulacja zapisu
w programie K3b*



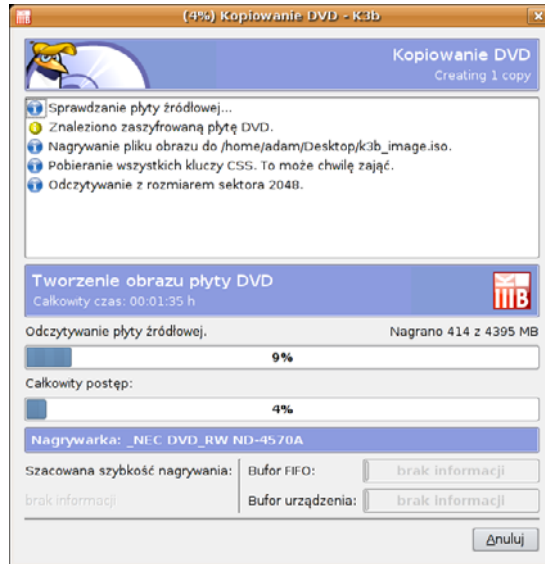
Podczas kopiowania płyt DVD użytkownik może wyłączyć automatyczne uruchamianie programu Totem. Aby to zrobić, należy wybrać z menu *System/Preferencje/Nośniki i napędy wymienne*, a następnie zgodnie z informacjami podanym w niniejszym rozdziale zmienić ustawienia w oknie *Ustawienia wymiennych nośników i napędów*. To wyeliminuje potencjalne konflikty pomiędzy programami K3b i Totem. Po zakończeniu korzystania z K3b wszystkie wprowadzone zmiany można cofnąć.

Aby skopiować płytę DVD za pomocą programu K3b, należy z menu wybrać *Narzędzia/Kopij DVD*, zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 13.40.

**Rysunek 13.40. Okno dialogowe Kopiowanie DVD**

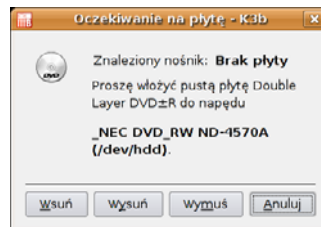
Po sprawdzeniu poprawności ustawień napędu źródłowego i docelowego należy kliknąć przycisk *Start*, a proces kopiowania zostanie rozpoczęty (patrz rysunek 13.41).

Rysunek 13.41.
Informacje o postępie
procesu kopiowania



Program odczyta zawartość dysku i utworzy plik ISO. Jeżeli napęd źródłowy i docelowy będzie taki sam, wówczas po utworzeniu pliku ISO program poprosi o umieszczenie w napędzie pustej płyty DVD (patrz rysunek 13.42).

Rysunek 13.42.
Żądanie
umieszczenia
nośnika w napędzie



Należy umieścić odpowiednią płytę i kliknąć przycisk *Wsuń*. Okno zostanie zamknięte, a program nagra plik ISO na dysku. I mamy gotową kopię.

Po zakończeniu procesu nagrywania program wyświetli okno dialogowe, w którym użytkownik będzie mógł sprawdzić komunikaty systemowe, co jest przydatne podczas analizy błędów. Zwykle jednak wystarczy użyć przycisku *Zamknij*. Ponownie zostanie wyświetlone okno widoczne na rysunku 13.40; aby powrócić do głównego okna programu, należy kliknąć przycisk *Zamknij*.

Dodatkowe informacje o programie K3b

Nie jestem jedynym zwolennikiem programu K3b; ma on wielu miłośników, nie tylko wśród fanów KDE. Oto kilka ciekawszych stron, które udało mi się znaleźć w internecie.

- ♦ www.k3b.org — strona domowa programu, zawiera FAQ oraz udostępnia najnowsze wydania aplikacji, a także repozytorium subwersji, jeżeli ktoś nie może doczekać się oficjalnego wydania.

- ♦ www.novell.com/coolsolutions/feature/2746.html — ciekawy artykuł Jasona Jonesa pt. „CD Burning in Linux Made Easy with k3b”.
- ♦ <http://sourceforge.net/projects/k3b/> — strona domowa projektu w witrynie SourceForge.
- ♦ <http://www.newsforge.com/software/03/10/20/1340254.shtml?tid=82> — interesujący artykuł o programie K3b autorstwa Joe Barra.
- ♦ www.pclinuxonline.com/wiki/K3b — szczegółowe, ale przydatne informacje o korzystaniu z programu K3b.
- ♦ <http://pl.wikipedia.org/wiki/K3b> — hasło w internetowej encyklopedii.

Podsumowanie

W niniejszym rozdziale wyjaśniłem wiele podstawowych zagadnień dotyczących obsługi plików audio i wideo w Ubuntu. Na początku zamieściłem słownik podstawowych pojęć związanych cyfrowymi multimediami, płytami CD oraz DVD i plikami przechowywanymi w komputerze. Następnie objaśniłem, jak skonfigurować Ubuntu, aby mogło wydawać różne dźwięki, a to dzięki zainstalowaniu szkieletu GStreamer, który umożliwia obsługę wielu różnorodnych formatów plików multimedialnych używanych w dzisiejszych komputerach. Omówiłem też kopiowanie i zgrywanie zawartości płyt audio CD, łączenie się z internetowymi stacjami radiowymi, odtwarzanie i konwertowanie plików audio. Obecnie ze względu na rozmiary dysków twardych można pozwolić sobie na wykonywanie kopii oryginalnych płyt CD czy DVD, co znacznie upraszcza i ułatwia ich późniejsze odtwarzanie, nie wspominając już o kwestiach bezpieczeństwa, jakie daje posiadanie kopii zapasowych.

W rozdziale 14. zdradzę, że moim ulubionym sposobem na spędzanie czasu przed monitorem są gry komputerowe. To przykre, ale większość producentów oprogramowania tego typu nie udostępnia swoich aplikacji w wersji dla Linuksa. Na szczęście, nie oznacza to, że użytkownicy Ubuntu nie będą mogli grać w gry — może poza najnowszymi strzelankami, wersjami Grand Theft Auto i innymi, których hiszpańska inkwizycja chętnie by zakazała. W Ubuntu mamy duży wybór klasycznych gier planszowych i karcianych, a także kilka ciekawych gier wideo. Setki innych dostępne są w internecie i repozytoriach menedżera pakietów Synaptic.

Rozdział 14.

Zagrasz?

W tym rozdziale:

- ♦ Domyślne gry dla GNOME w Ubuntu
- ♦ Gry karciane
- ♦ Gry planszowe
- ♦ Gry wideo
- ♦ Inne popularne gry dla Ubuntu

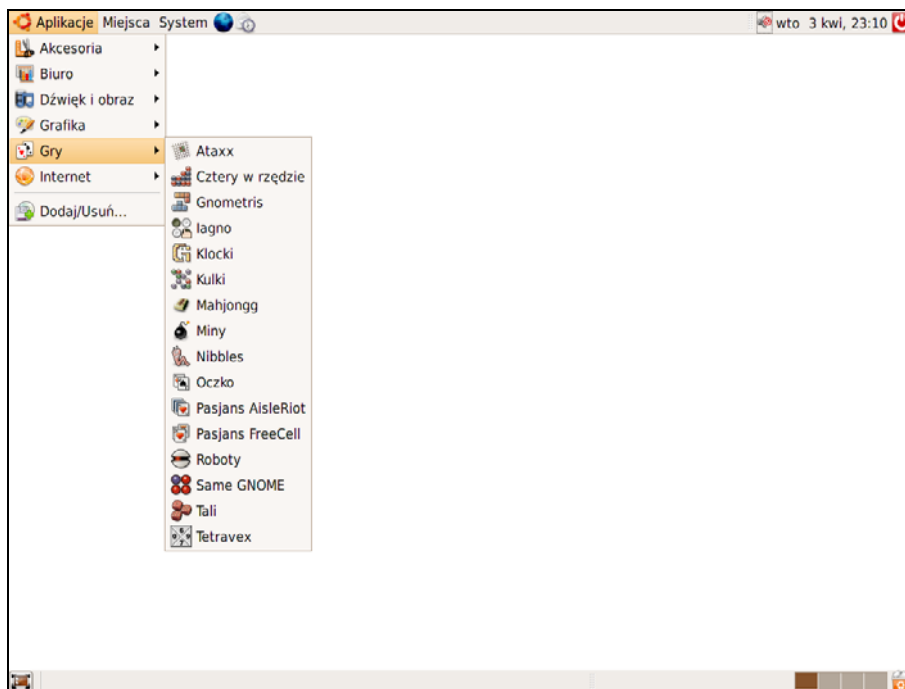
Gry w systemach uniksopodobnych przebyły długą drogę od czasu „Polowanie na Wumpusa”, tekstowej gry przygodowej, nad którą spędziłem długie godziny w 1982 roku, siedząc przed monitorem PDP-11. Dzisiaj, zgodnie z nadziejami, gry to skomplikowane, wydajne aplikacje graficzne z mnóstwem wodotrysków. Rzeczą, na którą czytelnicy zwrócą uwagę, zapoznając się z opisanymi w tym rozdziale grami przeznaczonymi dla Linuksa, jest to, że można w nich znaleźć dużą dawkę humoru, co zawsze działa odświeżająco. Śmiech i przegrywanie są prawie tak samo dobre jak wygrywanie (często sobie to powtarzam...).

Podkreślam w całej książce, że Linux jest dla ludzi, a im należy się trochę relaksu i rozrywki. Gry to ten aspekt systemów komputerowych, który oferuje możliwość oderwania się i rozrywki bez odchodzenia od biurka (choć nie jestem przekonany, czy to właściwe miejsce, kiedy mowa o relaksie, ale tak już jest).

W Ubuntu dostępny jest całkiem pokaźny zbiór gier. Kolejną ogromną ich porcję można znaleźć w repozytoriach, źródłach i pakietach na wielu różnych stronach rozsypanych w internecie. W niniejszym rozdziale omówię gry standardowo dostarczane wraz z Ubuntu, podkreślając fakt, że za pomocą tego samego komputera można wykonać daną pracę oraz nieco się zabawić. W pierwszym podrozdziale omówię gry instalowane domyślnie w systemie; w drugim podam informacje na temat innych popularnych gier i ich typów; zamieszcze też wskazówki gdzie je znaleźć i w jaki sposób zainstalować.

Domyślne gry dla GNOME

Jak widać na rysunku 14.1, w domyślnej instalacji Ubuntu dostępne jest szerokie spektrum różnorodnych gier, które można podzielić na planszowe, karciane i wideo.



Rysunek 14.1. Standardowe menu Gry dostępne tuż po instalacji

Następny podrozdział poświęcony jest różnym grom dostępnym po standardowej instalacji Ubuntu. Dla każdej zamieszczam opis oraz zrzut ekranu. Ponieważ są dostępne domyślnie i bez ponoszenia żadnych kosztów, a gusta mamy różne, użytkownik będzie mógł samodzielnie zdecydować, które go interesują.

Gry karciane

Chyba wszyscy spotkali się z sytuacją, gdy podczas przechodzenia między biurkami użytkowników komputerów dało się zauważyć szybkie minimalizowanie okna z pasjansem Solitaire. Dzisiaj ten pożeracz czasu jest już grą międzyplatformową i — oczywiście — dostępny jest również w Ubuntu. Wiele innych pasjansów można znaleźć w repozytoriach Ubuntu. Poniżej opisuję gry karciane dostępne w domyślnej instalacji systemu Ubuntu.

Pasjans AisleRiot

Pasjanse układałem od dziecka (a właściwie pasjansa) i jakoś tak się złożyło, że tym, którego się nauczyłem, był właśnie Klondike. Kolejną porcję wiedzy na ten temat zdobyłem, kiedy żona zapytała, czy są dostępne inne wersje pasjansa. Chwila googlowania

i już było wiadomo, że istnieją całe biliony gier tego typu, pozostało już tylko kupić analogowe urządzenie do przechowywania danych na ten temat — w tym przypadku była to książka.

Kolejne oświecenie przyszło po zapoznaniu się z systemem pomocy dla pasjansa AisleRiot, w którym można znaleźć informacje, że Klondike (patrz rysunek 14.2) to tylko jeden z wielu dostępnych pasjansów, a sam program zawiera ich 81.

Rysunek 14.2.
*Pasjans Klondike
w programie AisleRiot*



Fani układania pasjansów będą z pewnością zadowoleni, kiedy dowiedzą się, że w programie, oprócz Klondike, można zagrać jeszcze w takie gry jak: Atena, Aunt Mary, Blokuj dziesiątkę, Bristol, Camelot, Canfield, Cruel, Cygan, Czterdzieści, Czternaście, Do góry, Dubletony, Dywan, Dziesięć wszereż, Easthaven, Forteca, Fortuny, Feeceł, Gaps, Glenwood, Gnomy, Golf, Helsinki, Isabel, Jamestown, Jukon, Jumbo, Kansas, Kciuk i fałda, Kings Audience, Klasy, Kopalnia diamentów, Król Albert, Królewski wschód, Labirynt, Lady Jane, Monte Carlo, Oblężony zamek, Odessa, Osmoza, Ósemka odpada, Pająk, Pajęczycza, Piekarska gra, Piekarski tuzin, Pierwsze prawo, Plac Unii, Poker, Potrójny szczyt, Przesada, Przykrywka, Saratoga, Sąsiad, Schody ruchome, Seahaven, Sir Tomasz, Skorpion, Skrzydło orła, Stare dobre czasy, Starzec, Szachownica, Szkielet, Szkocka polka, Tasowanie, Trzydzieści, Trzyńaście, Uliczki i aleje, Walentynka, Warkocz, Westhaven, Wierzchołek, Winda, Zamęt, Zebra, Zegar, Zestaw pajak dwa, Zestaw pajak jeden, Złodzieje, Żłudne nadzieje, Zysk. Uff!

Jeżeli, tak jak ja, lubicie Klondike'a, również w Ubuntu znajdziecie implementację tej wspaniałej gry.

Oczko, czyli Blackjack

Jeżeli czytelnicy są miłośnikami gry w oczko i chcieliby rozegrać kilka tysięcy partii, wygrywając lub przegrywając pieniądze, to przed wakacjami w Vegas z pewnością przyda im się nieco ćwiczeń w liczeniu kart. Ubuntu oferuje taką możliwość, ponieważ zawiera całkiem przyjemną wersję tej gry (patrz rysunek 14.3).

Rysunek 14.3.
Oczko Vegas Strip



To wersja gry w oczko wieloma taliami, jak w kasynie. Program umożliwia także ustawienie bardziej zaawansowanych opcji, takich jak szybkie rozdania czy wyświetlanie prawdopodobieństw. Gra toczy się według zasad Vegas Strip, ale można też skonfigurować grę Atlantic City, Ameristar czy Vegas Downtown, dzięki czemu można zaplanować wakacyjną podróż po największych kasynach amerykańskich.

Pasjans FreeCell

Pasjans FreeCell stał się bardzo popularną grą karcianą dzięki obecności w systemach operacyjnych Microsoftu. Na rysunku 14.4 przedstawiam wersję tej gry dla GNOME; da się zauważyć, że na pasku narzędzi znajduje się przycisk *Podpowieź*, z którego zawsze można skorzystać, gdy sytuacja na planszy stanie się zbyt skomplikowana.

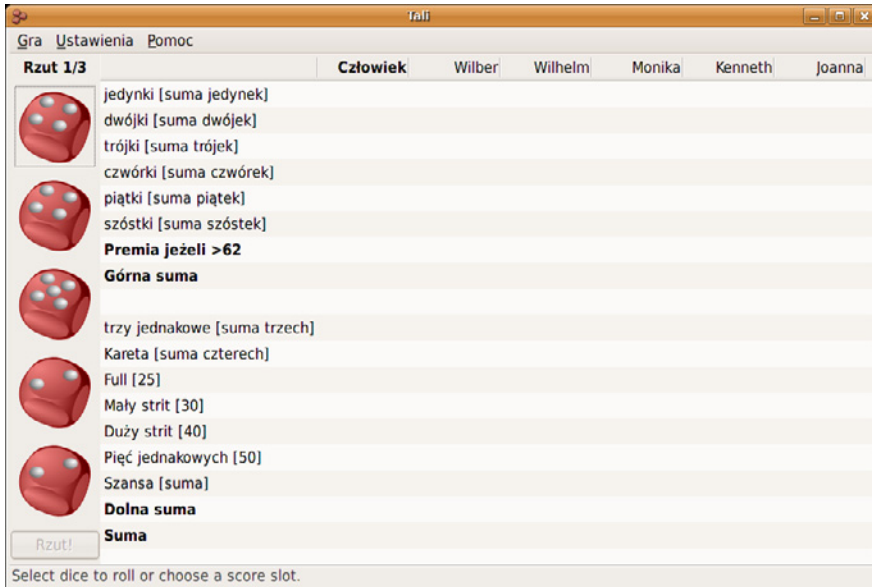
Rysunek 14.4.
FreeCell



Sposób, w jaki zaimplementowano tę grę w systemie, może być nieco mylący: przy wybieraniu jej z menu *Gry* wygląda podobnie jak pasjans AisleRiot, ale po uruchomieniu okazuje się, że mamy do czynienia z pojedynczą grą.

Tali

To internetowa wersja gry w kości, która w istocie jest emulacją gry karcianej — pokera. Na rysunku 14.5 przedstawiam ekran początkowy tej właśnie gry.



Rysunek 14.5. *Tali*

Gracze zaznajomieni z grami, takim jak Yahtzee firmy Hasbro, zapewne nie będą zaskoczeni Tali, poczują się raczej tak, jakby znaleźli dawnego znajomego. Domyślnie Tali skonfigurowana jest dla jednego gracza, który ma czterech — kierowanych przez komputer — przeciwników. Ale łatwo można zmienić te ustawienia tak, aby zamiast z komputerem zagrać z ludźmi. Szczególnie zabawny jest pojawiający się co jakiś czas komunikat: „Człowiek — twój ruch!”. Tak jest, panie Borg!

Gry planszowe

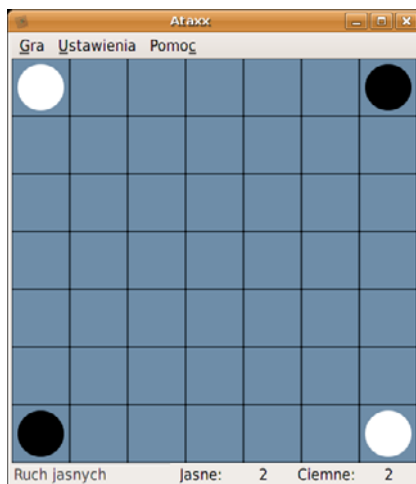
Gry planszowe zawsze były popularne, a w dodatku nie mają tych grzesznych konotacji wiążących się z grami karcianymi. Chociaż niektóre z gier, które się tu znalazły, to linuksowe implementacje innych gier komputerowych; dla ich zdefiniowania przyjęto, że są to te, które można rozgrywać na planszy lub z użyciem siatki.

Ataxx

Ataxx to gra polegająca na manipulowaniu żetonami, dość podobna do gry Iagno, która również wchodzi w skład podstawowego zestawu gier zainstalowanych w GNOME. Na rysunku 14.6 przedstawiam planszę startową Ataxxa.

Rysunek 14.6.

Ataxx



Jak można się zorientować, Ataxx (podobnie jak Iagno) wywodzi się z tego samego pomysłu, co istniejące gry planszowe, takie jak Othello czy Reversi. Jednak inaczej aniżeli w przypadku Iagno czy analogowych gier z żetonami, gra rozpoczyna się z żetonami umieszczonymi w narożnikach planszy i wymaga nieco innej strategii, chociaż główne zadanie, czyli zdobywanie żetonów przeciwnika, jest — oczywiście — takie samo.

Kulki

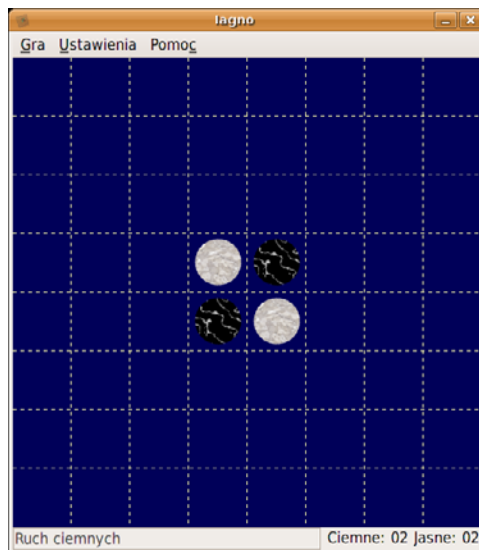
Kulki to linuksowa wersja gry dla Windows znanej pod tą samą nazwą. Na rysunku 14.7 przedstawiam planszę gry chwilę po rozpoczęciu.

Jak wiele dobrych gier, kulki są bardzo eleganckie w swojej prostocie. W każdej turze na planszy w losowo wybranych miejscach pojawiają się trzy kolorowe kulki. Zadaniem gracza jest ułożenie rzędu składającego się z co najmniej pięciu żetonów tego samego koloru, które dzięki temu znikną z planszy. Coś jak wolniejsza i nieliniowa wersja Tetris.

Niestety, w tej wersji brakuje — bardzo przydatnego — przycisku umożliwiającego ukrycie gry, gdyby na horyzoncie pojawił się szef. Po kliknięciu tego przycisku (obecnego w wersji dla Windows) na monitorze wyświetlana była inna zawartość ekranu, dzięki czemu kiedy tylko szef pojawiał się w zasięgu monitora, można było udawać, że poświęcamy się pracy. A ponieważ gra jest tak samo wciągająca, radzimy ukrywać ją przed przełożonymi.

Rysunek 14.7.*Kulki***lagno**

lagno to gra polegająca na manipulowaniu żetonami, dość podobna do gry Ataxx, która również wchodzi w skład podstawowego zestawu gier zainstalowanych w GNOME. Na rysunku 14.8 przedstawiam planszę startową lagno.

Rysunek 14.8.*lagno*

Jak można się zorientować, lagno przypomina popularne, istniejące gry planszowe, takie jak Othello czy Reversi.

Klocki

Klocki to strategiczna gra polegająca na manewrowaniu dużymi obiektami w zamkniętej przestrzeni (patrz rysunek 14.9). Na elementach wyświetlane są strzałki, za których pomocą można nimi manewrować. Gra może sprawić wiele przyjemności i posiada wiele plansz. Wyświetlana jest też informacja o ilości ruchów potrzebnych do osiągnięcia celu.

Rysunek 14.9.

Klocki

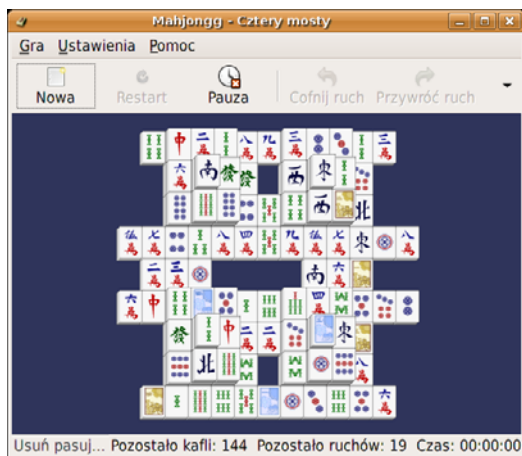


Mahjongg

GNOME Mahjongg to wersja dla jednego gracza klasycznej chińskiej gry Mahjongg, która z kolei powstała w skutek połączenia kilku innych, wcześniejszych gier. Na rysunku 14.10 przedstawiam wersję dla GNOME.

Rysunek 14.10.

Mahjongg



Mahjongg była bardzo popularna w Stanach Zjednoczonych i w Wielkiej Brytanii na początku XX wieku i nadal cieszy się powodzeniem. Celem gry jest usunięcie klocków poprzez dopasowanie do siebie odpowiednich wzorów. Wzory na klockach są bardzo zawile i mogą być też bardzo piękne. A kiedy użytkownik zmęczy się już wpatrywaniem w bardzo ładne, ale też bardzo podobne do siebie puzzle, wówczas może skorzystać z odpowiedzi oferowanych przez program.

Miny

Miny to wersja dla GNOME popularnej w systemach Microsoft Windows gry Saper. Na rysunku 14.11 przedstawiam ekran gry.

Rysunek 14.11.

Miny



Kliknięcie danego kwadratu powoduje wyświetlenie informacji o ilości min znajdujących się dookoła odkrytego pola. Kiedy pole z miną zostanie zidentyfikowane, należy oznaczyć je specjalną flagą. Aby wygrać, należy odkryć wszystkie pola na planszy, nie wpadając na minę. Ilość zdobytych punktów zależy od czasu, w jakim udało się oczyścić całą planszę.

Nigdy nie mogłem zrozumieć, dlaczego tak wielu ludzi uważa tę grę za interesującą, ale ponieważ dotyczy to zarówno wersji dla Windows, jak i dla GNOME, można przyjąć, że ta druga całkiem wiernie odpowiada oryginałowi.

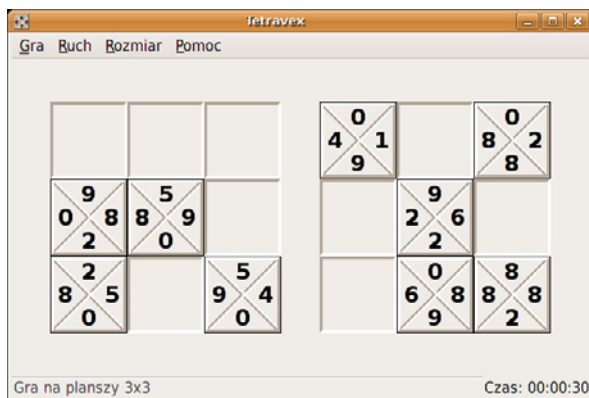
Tetravex

Tetravex to kolejna interesująca gra; polega na takim umieszczeniu na drugim polu kwadratów z cyframi, aby stykały się one ze sobą bokami, na których są takie same wartości liczbowe. Na rysunku 14.12 przedstawiam ekran z gry.

Tetravex wydaje się być całkiem interesująca. Podobnie jak w większości gier dla GNOME, również w tym przypadku dostępny jest system pomocy oferujący odpowiedzi.

Gry wideo

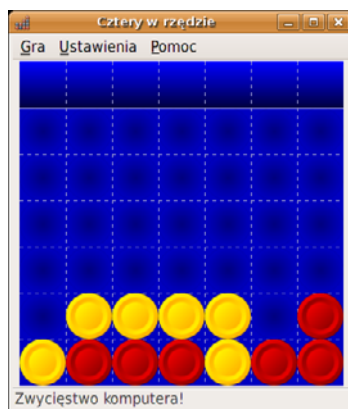
Gry wideo są prawdopodobnie najlepszym czynnikiem motywującym dla dzieci, żeby spędzały więcej czasu przed komputerami. I wcale nie zaskakują myśli, które przychodzą do głowy podczas uruchamiania kolejnej gry: „Chciałbym umieć zrobić coś takiego”. Zresztą jest to dużo prawdopodobniejszy scenariusz, niż nastolatek stojący przed teczką z dokumentami i myślący: „Chciałbym napisać program, który tworzyłby takie raporty”.

Rysunek 14.12.*Tetravex*

W pakiecie gier wideo zainstalowanych w Ubuntu można znaleźć implementacje kilku znanych, klasycznych już dziś gier, a wiele innych, takich jak np. Quake, również można uruchomić pod Linuksem. Więcej informacji na temat niektórych gier zręcznościowych podaję w ramce „MAME, MESS i przyjaciele”.

Cztery w rzędzie

Cztery w rzędzie to właściwie coś pomiędzy tryktrakiem a Tetrisem, co czyni tę grę dość interesującą i pozwala oderwać się od prawdziwej pracy na wiele godzin. Na rysunku 14.13 przedstawiam ekran z gry.

Rysunek 14.13.*Cztery w rzędzie*

Zadaniem gracza jest ułożenie rzędu żetonów w odpowiednim kolorze składającego się z czterech sztuk, zanim zrobi to komputer. Kolejne spadające żetony należy układać tak, aby stworzyć sobie możliwość zwycięstwa i jednocześnie uniemożliwić to przeciwnikowi.

Gnometris

Gnometris to implementacja dla GNOME gry Teris, prawdopodobnie najpopularniejszej gry komputerowej wszechczasów. Wymyślona w Związku Radzieckim, posiada wszelkie elementy dobrej gry: jest prosta, łatwa, a kiedy gracz przegrywa, myśli: „Byłem tak blisko, następnym razem się uda!”. Na rysunku 14.14 przedstawiam ekran z gry.

MAME, MESS i przyjaciele

MAME to akronim *Multiple Arcade Machine Emulator*, jest to też nazwa pakietu oprogramowania umożliwiającego uruchomienie starych gier wideo dzięki wykorzystaniu skompresowanych obrazów ROM. Jest to naprawdę niesamowite oprogramowanie i obecnie tylko przy jego użyciu możliwe jest zagranie w jedną z tych archiwalnych gier (chyba że ktoś posiada maszynę do podróży w czasie lub znajomości w muzeum starych komputerów). Ponieważ według licencji GPL, MAME nie jest otwartym oprogramowaniem, dlatego też nie jest domyślnie instalowane w Ubuntu. Kod źródłowy programu oraz binaria są wolno dostępne, ale komercyjne wykorzystanie i redystrybucja są zakazane. System graficzny *xmame* niezbędny do uruchomienia MAME znajduje się w repozytoriach Ubuntu. Za pomocą MAME można uruchomić ponad 6000 starych gier wideo. Wykonano naprawdę wspaniałą pracę, a wrażenie po uruchomieniu jednej z takich gier, która wydaje w dodatku autentyczne dźwięki, jest niesamowite. MAME jest dostępne dla większości współczesnych platform: Linuksa, Mac OS X (MacMame) i dla Microsoft Windows.

Jak można się spodziewać, jedyna trudność w korzystaniu z MAME wynika z konieczności zdobycia płyty CD ze starymi grami. W teorii (a często i w praktyce) użytkownik powinien posiadać własne, legalne kopie gier, niezależnie od skompresowanych obrazów ROM. Posiadam własną wersję Centipede (ulubionej gry) oraz Millipede w kolekcji oprogramowania komputerowego, ale nie jest to powszechna sytuacja. Szkoda, że firmy, kierując się chciwością, nie chcą uwolnić gier, na których kiedyś zarobiły miliony, ale tak to zwykle bywa. Z powodu takiego postępowania cała rzesza użytkowników pozbawiona jest możliwości legalnego korzystania ze starych gier.

Więcej informacji można znaleźć na stronach:

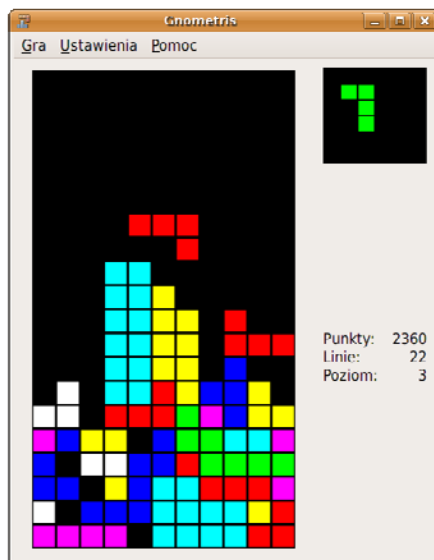
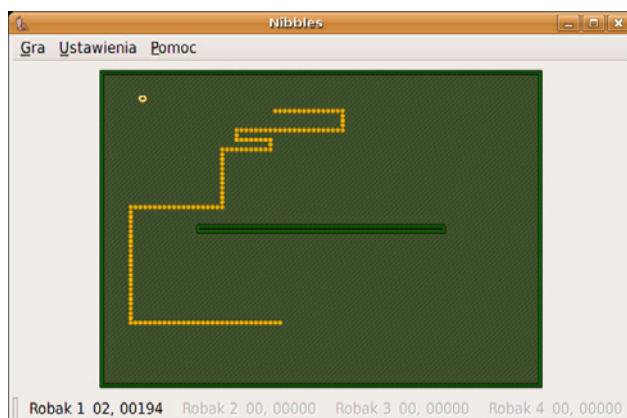
- ♦ <http://en.wikipedia.org/wiki/MAME> — ogólne informacje na temat działania MAME, statusu prawnego i obrazów ROM,
- ♦ <http://mameromlinks.com/> — wygodna strona, na której użytkownicy starych automatów do gier lub samych gier mogą bezpłatnie pobierać obrazy ROM z ogromnej kolekcji w formacie MAME, bez konieczności posiadania własnego czytnika PROM,
- ♦ <http://www.mamedev.org/> — podstawowa strona programistów MAME, można tam znaleźć najnowszą wersję oprogramowania MAME, a także narzędzia umożliwiające zmodyfikowanie programu do własnych potrzeb,
- ♦ Wyszukiwanie w Google hasła „MAME ROMs” przyniesie kilka milionów trafień.

Jeżeli czytelnicy trwonią swój czas na gry wideo, warto wówczas przetestować MAME. MESS z kolei to podobny emulator, tyle że ukierunkowany na stare gry komputerowe i konsolowe. Tak samo imponujący, sprawia tyle samo przyjemności, a jego używanie prowadzi do tej samej myśli: „Jak to możliwe, że jest już czwarta nad ranem?”.

Informacja dla czytelników, którzy wychowali się w jaskini: w grach typu Tetris chodzi o to, aby układać spadające elementy tak, by utworzyć z nich kompletne poziome rzędy, co spowoduje ich zniknięcie. Z czasem elementy zaczynają spadać coraz szybciej, a kiedy dotkną górnej krawędzi — gra jest skończona. Proste, przyjemne, szybkie i wciągające.

Nibbles

Nibbles to prosta, ale klasyczna już gra polegająca na kontrolowaniu ruchów węża i przemieszczaniu się po kolejnych planszach oraz pożeraniu kolejno pojawiających się na ekranie elementów, co powoduje zwiększanie się długości węża. Na rysunku 14.15 widać jedną planszę z gry.

Rysunek 14.14.*Gnometris***Rysunek 14.15.***Nibbles*

W czasie każdej gry wąż ma trzy życia, liczba ta będzie się zmniejszać za każdym razem, kiedy nasz gad uderzy w ścianę lub w siebie samego. Kiedy zostanie zużyte ostatnie życie, gra się kończy.

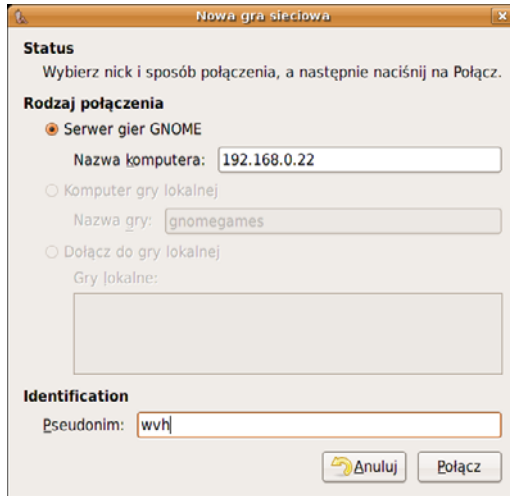
Nibbles jest także interesujący dlatego, że umożliwia grę w sieci (patrz rysunek 14.16).

Korzystając z tego okna dialogowego, można określić komputer, który będzie pełnił rolę serwera, a następnie rozpocząć rozgrywkę z innymi użytkownikami, mającymi dostęp do serwera. Wiele klasycznych gier planszowych, takich jak szachy czy go, posiada również tryb gry sieciowej. Nibbles jest prostym i przyjaznym przykładem gry, w którą można grać z innymi za pośrednictwem sieci.

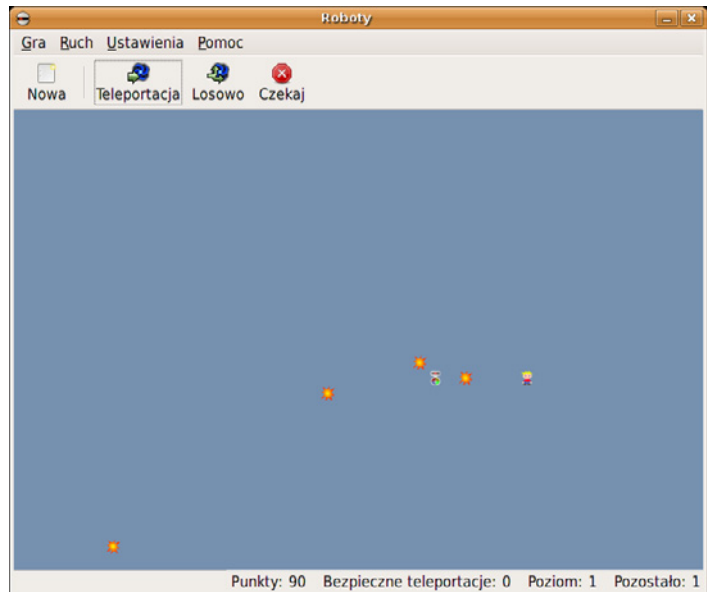
Roboty

Roboty to graficzna wersja tekstowych gier typu „poluj i zabij” dostępnych dla starszych systemów Unix. Na rysunku 14.17 przedstawiam ekran z gry Roboty.

Rysunek 14.16.
*Konfiguracja Nibbles
do gry w sieci*



Rysunek 14.17.
*Roboty polujące
na bohatera gry*



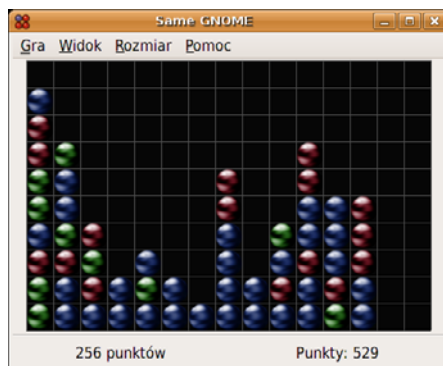
Wszystkie roboty występujące w grze chcą tylko jednego — zabić bohatera, dlatego jego cel jest prosty: po pierwsze, przeżyć, po drugie, zmusić roboty, aby zniszczyły się wzajemnie, wpadając na siebie lub na wraki zniszczonych wcześniej. Ta gra potrafi być zaskakująco uzależniająca.

Same GNOME

Same GNOME to gra łącząca w sobie różne aspekty Tetris, kulek i innych podobnych, polegająca na usuwaniu kulek w podobnych kolorach. Wskutek ich usuwania kolejne kulki zajmują zwolnione w ten sposób miejsca itd. Na rysunku 14.18 przedstawiam zrzut ekranu gry Same.

Rysunek 14.18.

Same GNOME



To naprawdę przyjemna gra i potrafi wciągnąć. Łatwo wpaść w pułapkę myślenia: „Pobiję ostatni rekord” i siedzieć przed monitorem komputera długo po tym, jak ostatnia żywa istota w domu już dawno zapadnie w sen.

Inne popularne gry znajdujące się w repozytoriach Ubuntu

Pakiet gier dla GNOME zainstalowanych w Ubuntu to wierzchołek góry lodowej, które można znaleźć w Linuksie. W repozytoriach *main*, *universe* i *multiverse* znajduje się mnóstwo innych gier; wystarczy parę kliknięć w menedżerze pakietów Synaptic i mogą się znaleźć na dysku komputera. Gdybym chciał opisać je wszystkie, wówczas nie starczyłoby miejsca na omówienie innych aspektów korzystania z Ubuntu, dlatego w tym podrozdziale skupię się tylko na wskazówkach, gdzie można znaleźć odpowiedniki dobrze znanych gier, z których na pewno korzystałoby wielu użytkowników Ubuntu, gdyby tylko wiedzieli, jak one się nazywają i gdzie ich szukać. Informacje na temat menedżera pakietów Synaptic można znaleźć w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”. Poniżej zamieszczam informacje, co powinien zrobić użytkownik, jeżeli jest zainteresowany daną grą.

Miliony wersji szachów

W świecie otwartego oprogramowania szachy posiadają dwie ważne cechy: ogromną popularność oraz fakt, że nikt nie ma ich na własność. Okoliczność, że szachy cieszą się popularnością od kilku tysięcy lat i są strategiczną grą zmuszającą do myślenia, spowodowała to, że są one obecne w komputerach od czasów, kiedy te zaczęły mieć wystarczającą ilość pamięci do obsługi tak skomplikowanej gry. Ponieważ nikt nie ma wyłącznych praw do nazwy „szachy”, nie trzeba ponosić żadnych kosztów czy płacić tantiem za jej używanie i dlatego każda gra komputerowa może nazywać się po prostu „szachy” (ang. *chess*), bez konieczności kamuflowania nazwy czy dokonywania zmian, aby umknąć przed chciwością korporacji.

Proste wyszukiwanie pakietu o nazwie *chess* w Synapticu przyniesie wynik w postaci 26 pozycji. Obejmuje on zarówno wersje do gry samodzielnej, sieciowej, silniki do gry, z którymi można się zmierzyć, grę, w którą można grać w edytorze emacs, specjalne

czcionki do zapisywania ruchów szachowych, a nawet szachownice, które można formatować w programach TeX i LaTeX (opisanych szerzej w rozdziale 10.) i wiele, wiele więcej.

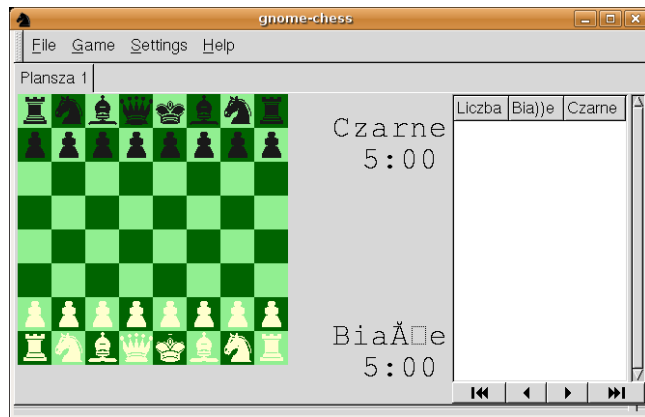


Jeżeli czytelnik jest prawdziwym fanem szachów, to dzięki możliwościom oferowanym przez internet nie musi się już ograniczać do gry tylko z przyjaciółmi, rodziną czy sąsiadami. Wystarczy wyszukać jeden z wielu serwerów szachowych dostępnych w sieci, które umożliwiają grę z przeciwnikami z różnych stron świata bez odchodzenia od klawiatury komputera. Więcej informacji można znaleźć na stronach, takich jak www.freechess.net czy www.tim-mann.org/ics.html, które są świetnym źródłem informacji na temat szachów w internecie. Dostępnych jest też wiele serwerów, za których wykorzystanie trzeba płacić (np. wykupuje się subskrypcję), można wówczas grać z innymi osobami, które również wykupiły takie subskrypcje.

Na rysunku 14.19 widoczne jest *gnome-chess*: interfejs graficzny do gry w szachy.

Rysunek 14.19.

Interfejs
gnome-chess



Gnome-chess to po prostu interfejs wyświetlający szachownicę oraz rejestrujący ruchy figur. Aby można było zrobić coś interesującego, należy wskazać silnik, który będzie przeciwnikiem, a także serwer internetowy, gdyby użytkownik chciał grać z innymi. Aby wprowadzić te ustawienia, należy wybrać z menu *Settings/Preferences*, zostanie wyświetlone okno dialogowe (patrz rysunek 14.20), w którym będzie można wprowadzić odpowiednie ustawienia. W przykładzie użyto programu *gnuchess*, który jest dostępny w repozytoriach Ubuntu i jednocześnie jest najlepiej znanym programem do gry w szachy w Linuksie (i większości systemów, ponieważ jest to oprogramowanie GNU i w związku z tym zupełnie darmowe). Inne popularne programy dostępne w repozytoriach to np. *fruit* i *phalanx*.

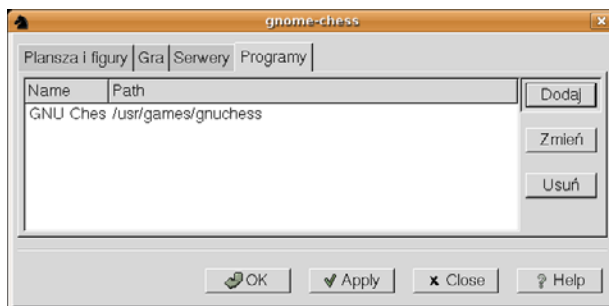
Kiedy program będzie już wiedział, gdzie szukać silnika do gry lub serwera internetowego, wówczas wystarczy wybrać odpowiednią opcję z menu *File/Programs* lub *File/Servers*, a zostanie wyświetlona nowa plansza do gry.



Z nieznanых bliżej powodów wersja *gnome-chess* dostępna podczas pisania niniejszej książki nie umieszczała odpowiedniego wpisu (*gnome-chess*) w programie, dlatego użytkownik powinien wprowadzić go samodzielnie. Należy uruchomić program terminala GNOME lub xterm (zgodnie ze wskazówkami zawartymi w rozdziale 6.), następnie dodać wpis `/usr/bin/gnome-chess` w ustawieniach programu i na końcu jeszcze dopisać program do menu (zgodnie ze wskazówkami zawartymi w rozdziale 5.).

Rysunek 14.20.

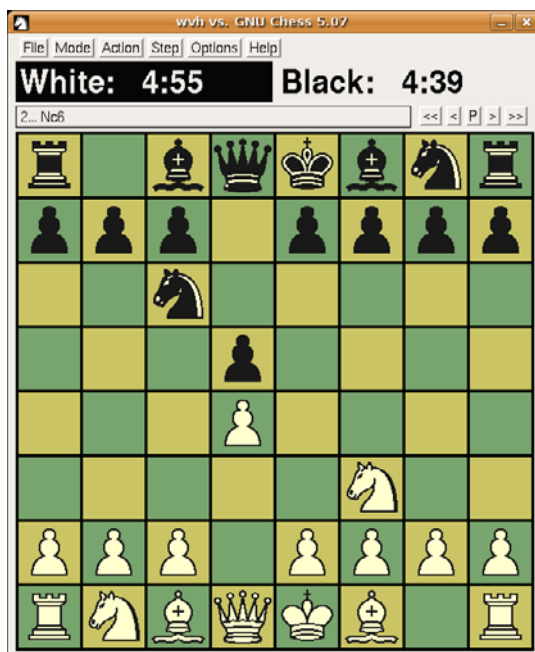
Wskazywanie
programu
do gry w szachy



Jak wspominałem wcześniej, repozytoria Ubuntu udostępniają wiele różnorodnych aplikacji do gry w szachy. Na rysunku 14.21 przedstawiam *xboard*, kolejny interfejs graficzny, za którego pośrednictwem można zmierzyć się z *gnuchess* lub z przeciwnikami w internecie. Po zainstalowaniu programu *xboard* odpowiedni wpis zostanie automatycznie umieszczony w menu *Aplikacje/Gry*.

Rysunek 14.21.

Program *xboard*
również korzystający
z silnika *gnuchess*



Kolejnym popularnym programem jest *eboard*, który z kolei jest chyba programem o największych możliwościach dostosowania; również jest dostępny w repozytoriach Ubuntu. Fani KDE powinni sprawdzić program *knights*, mój ulubiony program szachowy działający w tym środowisku.

Oczywiście, z czasem może się zdarzyć, że użytkownik pokona samego Bobby'ego Fishera czy Big Blue lub też po prostu poczuje się znudzony zwykłymi szachami. Wówczas powinien wypróbować pakiet *3d chess* (patrz rysunek 14.22), który wyświetla jednocześnie trzy szachownice i potrafi grać przeciwko użytkownikowi. Aby wybrać kolor, którym ma grać komputer, należy dodać opcję `-play white`.

Rysunek 14.22.
3D chess w działaniu



Po zainstalowaniu tego pakietu w menu *Aplikacje/Gry* nie pojawi się odpowiedni wpis, dlatego aby uruchomić program, należy wydać polecenie `3Dc`, jako opcję można wskazać kolor figur, którymi komputer ma grać. Użytkownicy, którzy wcześniej nie mieli do czynienia z tym programem, być może będą chcieli zapoznać się z dokumentacją, w której omówiono cechy aplikacji oraz sposób, w jaki działa. Dokumentację można znaleźć w `/usr/share/doc/3dchess/3dc-rules.html`.



Miłośnicy innych wersji szachów, takich jak np. japońskie Shogi, będą zapewne zadowoleni z faktu, że mogą zainstalować program *xshogi* (graficzny interfejs) oraz *gnushogi* (czyli sam silnik do gry) wprost z repozytoriów Ubuntu.

Go

Inną starą (liczącą sobie kilka tysięcy lat) grą strategiczną jest go, również powszechnie znana na świecie. Podobnie jak szachy, go jest popularna i wolna, dzięki czemu można znaleźć wiele interesujących implementacji w Linuksie, np. *gnugo*, czyli silnik gry. W internecie można też znaleźć wiele serwerów, na których dzień i noc czekają przeciwnicy gotowi stoczyć pojedynek. Wystarczy w menedżerze pakietów Synaptic przeprowadzić wyszukiwanie ciągu „go”, a wyświetlonych zostanie wiele różnych pakietów, np. wspomniany wcześniej *gnugo* czy *cgoban* (interfejs graficzny).

Na rysunku 14.23 przedstawiam ekran startowy interfejsu graficznego *cgoban* dla gry go.

W tym oknie dialogowym użytkownik ma do wyboru kilka opcji, takich jak IGS (ang. *Internet Go server*, czyli internetowy serwer go), NNGS (ang. *No Name Go Server*, czyli serwer bez nazwy), może też skorzystać z połączenia modemowego czy podczas przerwy

Rysunek 14.23.

Konfigurowanie
parametrów gry go



w dostawie sygnału internetowego zagrać z którymś z domowników. Na rysunku 14.24 wyświetlone jest okno dialogowe umożliwiające określenie parametrów połączenia modemowego oraz ustawienie programu tak, aby człowiek mógł zmierzyć się z maszyną.

Rysunek 14.24.

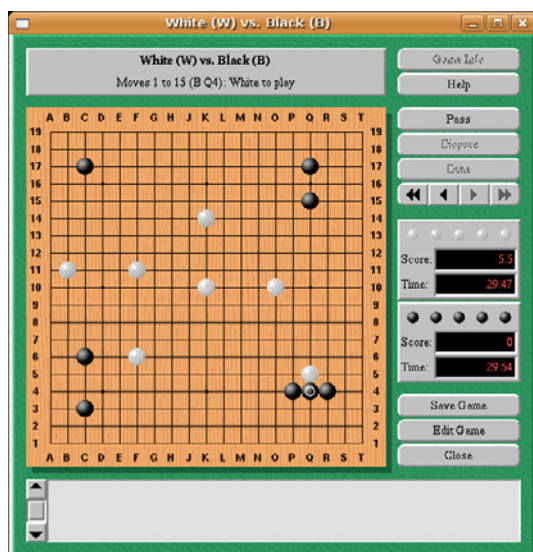
Konfigurowanie
graczy w programie
cgoban



Na rysunku 14.25 przedstawiona jest rozgrywka, ekran ten jest wyświetlany po uzupełnieniu kilku dodatkowych informacji dotyczących zasad, rozmiaru planszy itp.

Rysunek 14.25.

Człowiek kontra
maszyna
(czyli program gnugo)
— partia go



Informacja dla osób zainteresowanych go; najlepsza strona, jaką udało mi się znaleźć, należy do Brytyjskiego Stowarzyszenia go (BGA — ang. *British Go Association*), można tam znaleźć historię gry, informacje o programach umożliwiających rozgrywkę, o organizowanych zawodach oraz serwerach dostępnych w internecie. Warto też odwiedzić

witrynę Międzynarodowej Federacji go (www.nihonkiin.or.jp/igf), która — co prawda — nie jest tak przyjazna jak strona BGA, ale zawiera wiele przydatnych łączy do stron poświęconych go rozrzuconych na całym świecie. Obszerną listę internetowych serwerów go korzystających z różnych protokołów można natomiast znaleźć na stronie <http://senseis.xmp.net/?GoServers>.

Klony gry Monopol

Podobnie jak szachy i go, Monopol to również gra planszowa ciesząca się ogromną popularnością. Niestety, gra jako taka należy do firmy Hasbro (która z kolei wykupiła Parker Brothers Inc., czyli firmę, gdzie wymyślono Monopol, więcej informacji można znaleźć na stronie www.hasbro.com/monopoly), a ta — jak na razie — nie zdecydowała się na przygotowanie wersji dla Linuksa. Ale nie ma się czym martwić, bo od czego jest społeczność *open source*. W repozytoriach Ubuntu dostępna jest gra Atlantik, która bardzo przypomina Monopol i umożliwia grę z innymi graczami za pośrednictwem internetu. Aby odnaleźć grę w czeluściach repozytoriów, należy wyszukać w menedżerze Synaptic pakiet o nazwie *Monopoly*. Użytkownicy GNOME powinni zainstalować pakiet o nazwie *gkatlantic*; gra oryginalnie została zaprojektowana dla środowiska KDE i pakiet *atlantik* właśnie dla niego jest przeznaczony. Niezależnie od używanego środowiska, jeżeli użytkownik planuje używać swojego komputera jako hosta dla internetowej rozgrywki, należy wówczas zainstalować jeszcze pakiet *monopd*.



Atlantik jest grą zaprojektowaną do rozgrywki sieciowej, która może działać jako serwer, ale nie oferuje trybu dla pojedynczego gracza. Oznacza to, że aby rozegrać partię, należy skontaktować się z już działającym serwerem Atlantika lub uruchomić własny i zaprosić do skorzystania z niego innych.

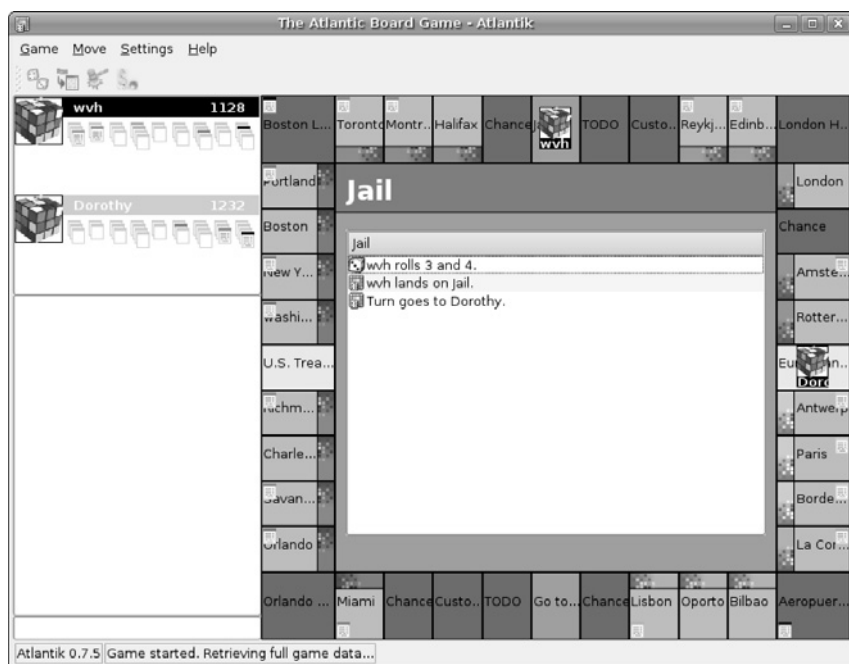
Po zainstalowaniu odpowiednich pakietów grę można uruchomić, wybierając kolejno z menu *Aplikacje/Gry/Atlantik*. Na rysunku 14.26 przedstawiam ekran z gry z użyciem interfejsu *gkatlantic*.

Z kolei na rysunku 14.27 można sprawdzić, jak wygląda gra uruchomiona z interfejsem *atlantik*. Istnieje kilka różnic między interfejsem widocznym na rysunku 14.26 a przedstawionym na rysunku 14.27. Podstawowa to — oczywiście — wygląd planszy; obowiązujące zasady w obu przypadkach są takie same, choć można je nieco modyfikować (o tym później).

Dotarcie za pierwszym razem do momentu, w którym pojawia się wreszcie właściwy ekran gry, może być nieco kłopotliwe. Po uruchomieniu programu *gkatlantic* zostanie wyświetlony ekran startowy; aby przejść dalej, należy wcisnąć klawisz *Enter*. Zostanie wówczas wyświetlone okno dialogowe przedstawione na rysunku 14.28.

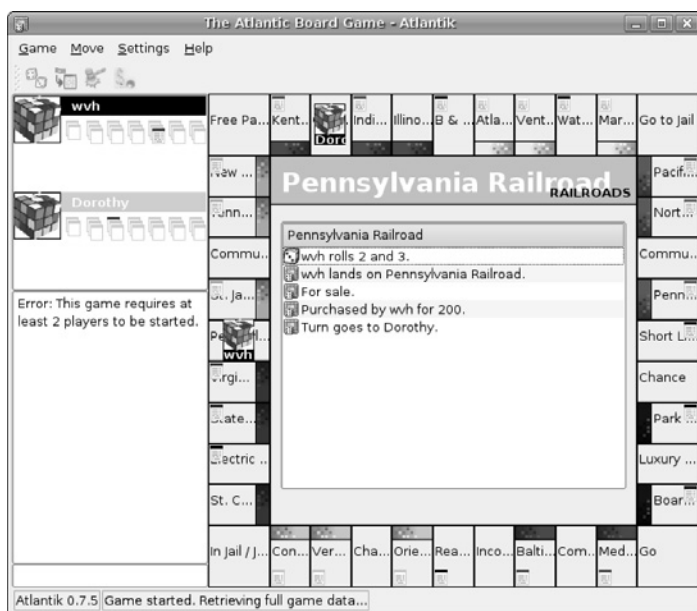
W polu *Another server* trzeba podać nazwę hosta lub adres IP serwera, z którym użytkownik chce się połączyć. Następnie należy kliknąć przycisk *Get games*, co spowoduje wyświetlenie listy dostępnych serwerów. Teraz trzeba zaznaczyć odpowiedni serwer, a następnie kliknąć przycisk *CREATE/JOIN GAME* (widoczny na rysunku 14.28). Gry, które zostaną wyświetlone, zależą od tego, jaki typ serwera został wybrany.

Jeżeli żadna gra nie jest aktualnie dostępna, a użytkownik uruchomił program *monopd*, wówczas nie pozostaje nic innego, jak uruchomić własną grę i poczekać, aż ktoś się do



Rysunek 14.26. Gra Atlantik

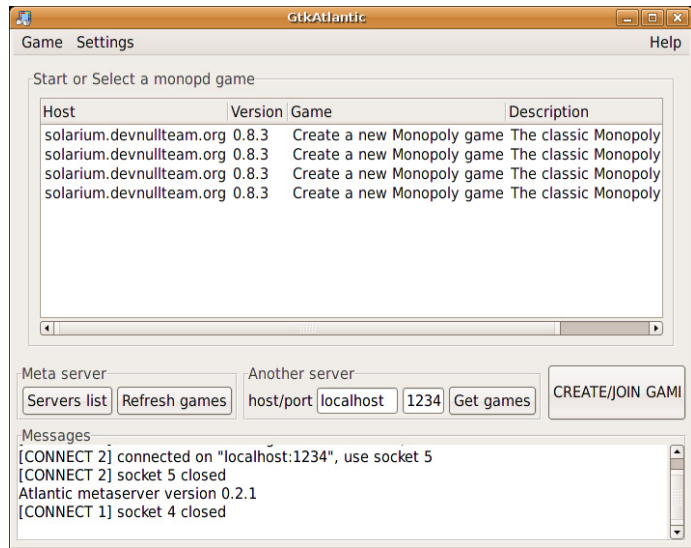
Rysunek 14.27.
Atlantik — klon gry
Monopol



niej włączy. Proces przebiega niemal identycznie jak podłączanie się do już istniejącej gry, tyle tylko, że z listy serwerów należy wybrać którąś z opcji *Create a new...* zamiast *Join...*, wówczas zostanie wyświetlone okno konfiguracji programu; dopiero w momencie, gdy ktoś zechce włączyć się do gry, zostanie wyświetlona właściwa plansza programu.

Rysunek 14.28.

Atlantik — łączenie się z serwerem gry gdzieś w sieci



Jeżeli pakiet *monopd* został zainstalowany na komputerze, wówczas możliwe będzie uruchomienie serwera o tej samej nazwie. Jeżeli użytkownik nie może połączyć się ze zdalnym serwerem lub po prostu chciałby uruchomić swój własny, może sprawdzić jego status, wykonując polecenie: `ps alxww | grep monopd`. Jeśli okaże się, że nie działa żaden proces o nazwie *monopd*, wówczas można go uruchomić poleceniem `sudo -u nobody /usr/sbin/monopd`.

Jako osobny pakiet dostępny jest edytor plansz o nazwie Atlantik Designer. Na rysunku 14.29 przedstawiam zrzut ekranu edytora, na którym widoczne są puste pola. Użytkownik może je swobodnie dostosowywać, następnie zapisać tak przygotowaną planszę i używać w kolejnych grach.

Rysunek 14.29.

Edytor Atlantik Designer umożliwia projektowanie własnych plansz do gry

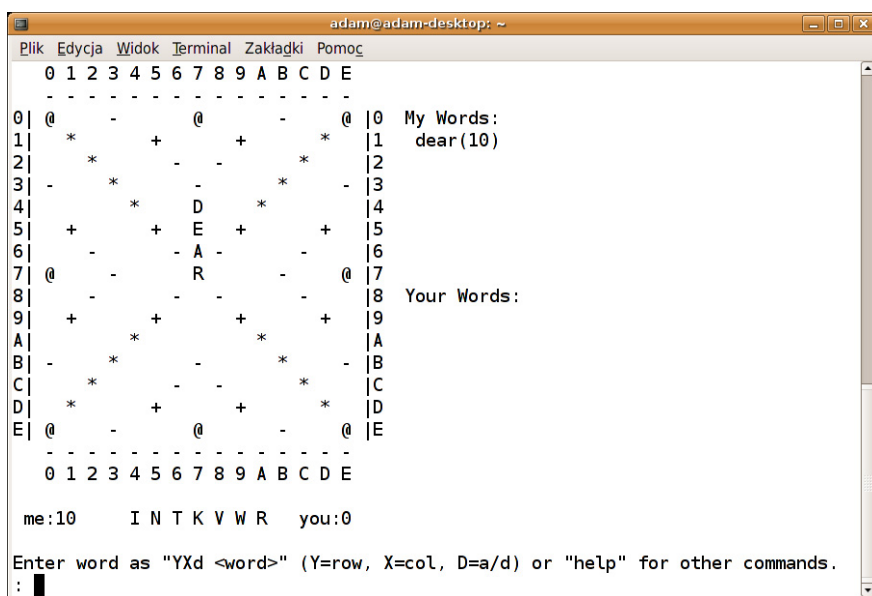


Korzystanie z edytora plansz może znacznie urozmaicić rozgrywkę, wystarczy tylko na planszy zamienić standardowe wpisy na firmy występujące w danym regionie. Któż nie chciałby rządzić w sąsiedztwie (a przynajmniej stać się właścicielem okolicznych firm)?

Atlantik to sposób na zostanie kapitalistą bez potrzeby robienia komukolwiek krzywdy, a sama gra przysparza tyle samo przyjemności, ile jej papierowy odpowiednik — Monopol. Miło byłoby, gdyby firma Hasbro zdecydowała się opublikować wersję gry przeznaczoną dla Linuksa, ale dopóki to się nie stanie, Atlantik jest w zupełności wystarczający.

Klony gry Scrabble

Scrabble to kolejna niezwykle popularna gra planszowa, która cierpi na tę samą chorobę własnościową jak Monopol. Scrabble są własnością Milton Bradley, która również w całości należy do Hasbro. Dostępne są wersje tej gry dla systemów Microsoft Windows oraz Macintosh, ale dopóki firma Hasbro nie zdecyduje się wydać wersji dla Linuksa, użytkownikom Ubuntu pozostaje korzystanie z wersji nieoficjalnej. Aby ją odnaleźć, wystarczy uruchomić menedżera Synaptic i wyszukać pakietu o nazwie *scrabble*. Po jego zainstalowaniu grę można uruchomić, korzystając z terminala GNOME lub okna xterm, wystarczy wpisać polecenie *scrabble*. Na rysunku 14.30 przedstawiam dostępną w Ubuntu wersję Scrabble.



Rysunek 14.30. Podstawowa wersja Scrabble w Ubuntu

I tak oto standardowy terminal Ubuntu został przekształcony w planszę do gry w Scrabble. Po pierwszym szoku łatwo się przyzwyczaić; kiedy już użytkownik przywyknie do podawania pozycji wprowadzanych słów, Scrabble dla Ubuntu okazują się wyjątkowo grywalne. Gra jest wyposażona w całkiem inteligentny silnik, który okazuje się sprytniejszy od części użytkowników (czytaj: mnie), co tylko podnosi atrakcyjność rozgrywki.



Dla Linuksa dostępne są również inne wersje Scrabble, oferujące interfejs graficzny, choć aktualnie nie są one oferowane w postaci pakietów, które łatwiej zainstalować w Ubuntu. Można pobrać źródła *xscrabble*, wersji korzystającej z systemu X11 (dostępne pod adresem ftp://ac-grenoble.fr/ge/educational_games). Można też pobrać grę *Jduplicate*, klon Scrabble działający w środowisku Java, dostępną pod adresem <http://jduplicate.sourceforge.net>.

Podsumowanie

Dla Ubuntu dostępny jest szeroki wybór różnorodnych gier komputerowych, począwszy od klasycznych gier karcianych i planszowych, aż do współczesnych gier wideo. W repozytoriach Ubuntu można znaleźć wiele różnorodnych gier, również tych zaawansowanych graficznie np. różnego rodzaju symulatorów. Niezależnie od tego, czy użytkownik chce zagrać w grę, która ma kilka tysięcy lat tradycji, jak go i szachy, czy też po prostu postrzelać do kosmitów, dzięki Ubuntu może spędzić niezliczone godziny na zabawie, przynajmniej dopóki nie zaśnie nad klawiaturą albo będzie musiał udać się do pracy.

W rozdziale 15. przedstawiam metody łączenia się z innymi komputerami pracującymi w sieci. Niezależnie od tego, czy użytkownik musi skorzystać z linii poleceń, czy użyć trybu graficznego, aby wykonać na zdalnym komputerze kilka poleceń lub dokonać transferu plików pomiędzy kilkoma komputerami — większość potrzebnych do tego narzędzi już znajduje się w systemie Ubuntu. Pozostałe potrzebne narzędzia oddalone są tylko o kilka kliknięć w najbliższym repozytorium Ubuntu.

Rozdział 15.

Łączenie się z innymi systemami

W tym rozdziale:

- ♦ Ustawianie bezpiecznego połączenia
- ♦ Umożliwianie bezpiecznego połączenia z własnym komputerem
- ♦ Zdalny dostęp do Mac OS X (konsola i graficzny)
- ♦ Korzystanie z sieci wirtualnych

Nie tak dawno temu, kiedy ktoś mówił o sieci, zwykle miał na myśli rozmawianie ze znajomymi za pomocą komunikatora, wysyłanie poczty elektronicznej czy też surfowanie po internecie z komputera domowego. Dziś komputery działające w domach są znacznie bardziej skomplikowane, często umożliwiają tworzenie sieci domowych czy przeznaczonych dla małych biur, w których mogą wymieniać pliki, często są centralnie administrowane, a łączyć się z nimi zwykle chce (lub musi) wielu użytkowników. Chociaż sieciowy dostęp do współdzielonych zasobów, takich jak serwery plików czy wydruku jest realizowany przez SMB od dawna w dużych przedsiębiorstwach, to w dzisiejszych czasach jego obsługa w niewielkich sieciach domowych jest konieczna.

W niniejszym rozdziale omówię różnego rodzaju narzędzia dostępne w Ubuntu zarówno z linii poleceń, jak i te z interfejsem graficznym, za których pomocą można łączyć się ze zdalnymi komputerami, logując się na nie bezpośrednio lub uzyskując zdalny, graficzny dostęp do uruchomionych na nich sesji. W rozdziale tym nie będzie mowy o łączeniu się ze zdalnymi serwerami plików oraz o udostępnianiu plików, te zagadnienia opiszę w rozdziale 16.

Ustawianie bezpiecznego połączenia z innymi systemami

Tradycyjne narzędzie uniksowe używane do nawiązywania zdalnego połączenia z innym terminalem poprzez sieć to *telnet*. Program ten łączy się z demonem telnet działającym na zdalnym komputerze i po wymianie danych identyfikacyjnych tworzy połączenie.

Niestety, wszystkie informacje przesyłane tym protokołem pomiędzy klientem a serwerem są jawne, co oznacza, że są dostępne dla każdego, kto ma dostęp do sieci, wystarczającą ciekawość lub inną motywację. Oczywiście, ten problem można rozwiązać, korzystając np. z szyfrowanych wirtualnych sieci prywatnych (VPN), tuneli, można też zainstalować wersję telnetu obsługującą protokół Kerberos. Ale po co uparczywie trwać przy starym programie i protokole, skoro dostępne są nowsze i bezpieczniejsze rozwiązania?



W Ubuntu dostępny jest klient programu telnet (instalowany domyślnie w systemie), ponieważ użytkownik może potrzebować go do połączenia się ze starszymi systemami lub takimi, które nie obsługują bardziej zaawansowanych połączeń zdalnych, natomiast serwer programu telnet nie jest oficjalnie wspierany. Jeżeli użytkownik musi z niego skorzystać, wówczas zawsze może zainstalować taki serwer (program o nazwie *telnetd*), korzystając z repozytoriów *universe* i menedżera pakietów Synaptic.

Starej daty miłośnicy Uniksa zapewne pamiętają zbiór poleceń *r-* z Berkeley, takich jak *rsh* czy *rcp*. Miały one takie same problemy z bezpieczeństwem jak dawny telnet, przysyłały jawnie nawet hasło i inne dane służące do uwierzytelniania. Chociaż te polecenia pojawiły się w Ubuntu, zostały przystosowane do pracy w SSH (są to odpowiednio *ssh* i *scp*), więcej informacji na ten temat można znaleźć w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Logowanie się na zdalnych systemach za pomocą ssh

Nowszą, nowocześniejszą i bezpieczniejszą alternatywą dla telnetu jest program *ssh*, wykorzystujący bezpieczny protokół SSH. Program *ssh* jest instalowany domyślnie w Ubuntu wraz z innymi aplikacjami korzystającymi z tego samego bezpiecznego protokołu oraz danych uwierzytelniających, a także podobnej składni (polecenie *sftp* — *secure FTP* oraz *scp* — *secure copy*). Za pomocą *ssh* można ustanowić zdalną sesję z dowolnym zdalnym komputerem, na którym działa demon *ssh*. Jeżeli podczas próby połączenia się ze zdalnym komputerem zostanie wyświetlony poniższy komunikat:

```
$ ssh ulaptop
ssh: connect to host ulaptop port 22: Connection refused
```

oznacza to, że demon *ssh* nie działa lub jest uruchomiony na innym, niż 22 porcie.

W tej części powiem, jak połączyć się komputerem, na którym działa demon *ssh*. Informacje, jak uruchomić ten demon w Ubuntu można znaleźć w dalszej części rozdziału.

Skorzystanie z możliwości nawiązywania połączenia jest bardzo proste, wystarczy wpisać polecenie *ssh nazwa_hosta*, gdzie *nazwa_hosta* to nazwa lub adres IP komputera, z którym użytkownik chce się połączyć. Podczas pierwszego łączenia się z danym systemem zostanie wyświetlony komunikat podobnej treści:

```
$ ssh ulaptop
The authenticity of host 'ulaptop (192.168.6.90)' can't be established.
RSA key fingerprint is 07:e6:3a:50:4b:6d:e6d8:f1:80:c6:b2:da:02:a3:da
Are you sure you want to continue connecting (yes/no)?
```

Aby kontynuować proces łączenia się, należy wpisać *yes* i wcisnąć klawisz *Enter*. Narzędzie *ssh* doda klucz RSA do listy znanych hostów znajdującej się na komputerze zdalnym (co zapobiega podawaniu się w przyszłości innych komputerów za kogoś innego), następnie zostanie wyświetlony komunikat z prośbą o podanie hasła:

Warning: Permanently added 'ulaptop' (RSA) to the list of known hosts.
wvh@ulaptop's password:



Informacja dla zainteresowanych: skrót RSA pochodzi od nazwisk programistów (Rivest, Shanir, Adleman), którzy zaprojektowali technologię szyfrowania opartą na kluczu publicznym, używaną w OpenSSH, czyli w implementacji protokołu SSH. Więcej informacji o infrastrukturze klucza publicznego (PKI — ang. *Public Key Infrastructure*) oraz powiązanych metodach szyfrowania można znaleźć na stronach Wikipedii (<http://en.wikipedia.org/wiki/Pki> oraz <http://pl.wikipedia.org/wiki/PKI>).

Teraz należy wprowadzić hasło dostępu do zdalnego systemu i wcisnąć klawisz *Enter*. Przy założeniu, że użytkownik wprowadzi poprawne hasło dostępu, system wyświetli komunikat potwierdzający zalogowanie się w zdalnym systemie, tak jak w poniższym przykładzie:

```
Linux Ubuntu32 2.6.15-21-386 #1 PREEMPT Fri Apr 21 16:43:33 UTC 2007
```

```
The program included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the individual files in  
/usr/share/doc/*/copyright.
```

```
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.  
Last login: Wed May 1 20:15:16 2007  
wvh@ubuntu32:~$
```

Po ustanowieniu za pomocą `ssh` połączenia ze zdalnym komputerem i wyświetleniu powyższego znaku zachęty można właściwie zrobić wszystko to, co zrobimy, siedząc bezpośrednio przed komputerem, łącznie z uruchamianiem aplikacji graficznych w systemie lokalnym (jeżeli tylko zmienna środowiskowa `DISPLAY` jest odpowiednio ustawiona; więcej informacji na ten temat zamieszczam w rozdziale 5., „Środowisko GNOME”). Po zakończeniu pracy można się wylogować, używając skrótu klawiszowego `Ctrl+D` lub wpisując polecenie `exit`.

Wszystko działa znakomicie przy założeniu, że we wszystkich systemach, w których loguje się użytkownik, używana jest ta sama nazwa użytkownika lub użytkownik loguje się zawsze na swoje konto. Na szczęście, narzędzie `ssh` umożliwia również logowanie się na zdalnych systemach na konta innych użytkowników za pomocą polecenia `ssh użytkownik@host`, gdzie *użytkownik* to odpowiednia nazwa użytkownika, a *host* to nazwa komputera, z którym ma zostać nawiązane połączenie. I tak np. polecenie `ssh jan_kowalski@ulaptop` spowoduje połączenie się za pośrednictwem `ssh` na komputerze *ulaptop* oraz umożliwi zalogowanie się na koncie użytkownika *jan_kowalski*. Znając nazwę użytkownika oraz hasło do danego konta, można zalogować się na wybranym komputerze i przeprowadzić potrzebne operacje.

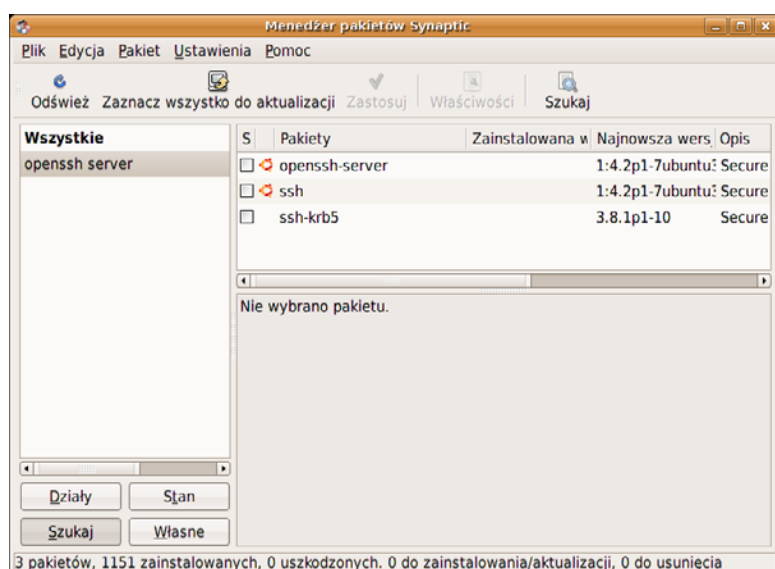
Polecenie `ssh` ma wiele opcji, użytkownik zapewne będzie potrzebował tylko kilku z nich. Jeżeli potrzebne są dodatkowe informacje, wówczas warto skorzystać z pomocy dostępnej w systemie, wystarczy w terminalu wykonać polecenie `man ssh`.

Uruchamianie demona SSH w systemie

Jak wspominałem wcześniej, demon SSH (`sshd`) musi działać w systemie, aby się z nim połączyć zdalnie z wykorzystaniem polecenia `ssh` i protokołu SSH. Wersja SSH dostępna w Ubuntu to OpenSSH (www.openssh.org), opiekują się nią programiści odpowiedzialni też za projekt OpenBSD (www.openbsd.org), a także inne otwarte systemy oparte na oryginalnym Berkeley Unix, znanym z priorytetowego traktowania kwestii bezpieczeństwa. Serwer OpenSSH nie jest domyślnie instalowany, ale (jak wszystko inne) można go łatwo dodać do naszego systemu, korzystając z menedżera pakietów Synaptic.

Aby zainstalować serwer OpenSSH, należy najpierw uruchomić Synaptic (wybierając z menu *System/Administracja*), następnie należy uruchomić wyszukiwanie ciągu *openssh*, po jego zakończeniu system wyświetli wynik widoczny na rysunku 15.1.

Rysunek 15.1.
Wyszukiwanie
serwera OpenSSH
w Synapticu



Po zaznaczeniu do zainstalowania pakietu *openssh-server* trzeba kliknąć przycisk *Zastosuj*. Następnie należy potwierdzić chęć jego zainstalowania, a po pobraniu programu i jego pomyślnym zainstalowaniu można zamknąć program Synaptic.

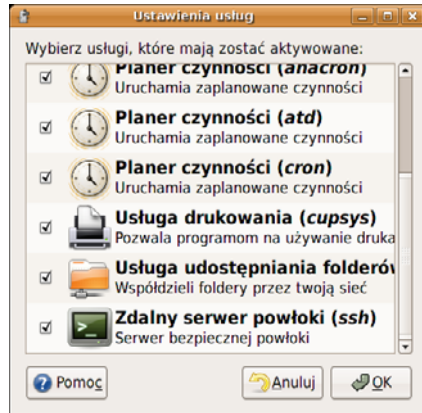
W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, można znaleźć więcej informacji na temat korzystania z menedżera pakietów Synaptic.

Podczas instalowania serwera OpenSSH w systemie zostanie także umieszczony wpis dla nowej usługi – demona `ssh`. Aby go wyświetlić, wystarczy wybrać z menu *System/Administracja/Usługi*. Jak widać na rysunku 15.2, po zainstalowaniu serwer SSH jest automatycznie włączany podczas startu systemu.

Jeżeli użytkownik chciałby wyłączyć możliwość logowania się zdalnych użytkowników na swoim komputerze, wystarczy usunąć zaznaczenie przy pozycji *Zdalny serwer powłoki (ssh)* i kliknąć *OK*, demon `ssh` zostanie wyłączony.

Rysunek 15.2.

*Sprawdzanie statusu
serwera SSH*



Łączenie się z innymi systemami za pomocą VNC

Podczas korzystania z sieci, w której znajduje się wiele komputerów, wstawianie i podchodzenie do każdego, aby wykonać jakieś zmiany w konsoli, szybko może stać się męczące. Czyż nie lepiej byłoby łączyć się z innymi komputerami i wyświetlać uruchamianie na nich aplikacje graficzne na własnym monitorze? Taka właśnie możliwość jest udostępniania w domyślnej instalacji systemu Ubuntu.

Virtual Network Computing (VNC), ułatwia pracę na wielu komputerach korzystających ze środowiska graficznego, ponieważ umożliwia wyeksportowanie zawartości ekranu jednego komputer na ekran innego. VNC to technologia międzyplatformowa, oryginalnie zaprojektowana przez laboratorium badawcze Olivetti w Cambridge, które zostało później przejęte przez AT&T. Serwer VNC jest uruchamiany na jednym komputerze i eksportuje obraz pulpitu, który z kolei może być wyświetlany na innej maszynie. Serwery VNC są zwykle chronione hasłem, co sprawia, że to rozwiązanie jest doskonałym sposobem na graficzne, zdalne administrowanie innymi komputerami. Eksport danych przez serwery VNC ma miejsce przez port 5900, do tej liczby dodawana jest ilość eksportowanych obrazów środowisk. I tak np. serwer VNC uruchomiony w środowisku X Windows na wyświetlaczu :1 będzie korzystał z portu o numerze 5901.

Technologia VNC jest dostępna na licencji GPL, a wielu programistów VNC pracuje obecnie dla firmy RealVNC (www.realvnc.com), dystrybuującej i wspierającej komercyjnie implementacje VNC, ale też udostępniającej kod źródłowy kolejnych, nowych wersji, zgodnie z wymogami licencji GPL. Program kliencki nazywa się *vncviewer*, jego używanie w celu podłączenia się do innych komputerów omówię w tym rozdziale, w punkcie „Korzystanie z programu vncviewer”. Inną bardzo popularną dystrybucją jest TightVNC, oferująca serwer i program kliencki, które — chociaż nie są instalowane domyślnie w systemie — również dostępne są dla Ubuntu.

Pobieranie programu VNC w wersji serwerowej oraz klienckiej dla odpowiedniej platformy

Nim użytkownik spróbuje użyć programu vncviewer do połączenia się z wybranym komputerem w sieci, należy na nim uruchomić serwer VNC. Poniżej objaśniam, w jaki sposób uruchomić serwer VNC w Ubuntu, a także jak uzyskać klienta i serwer VNC przeznaczony dla innych, popularnych platform, takich jak Mac OS X i Microsoft Windows.



Niezależnie od używanej platformy, należy się upewnić, czy porty, z których będzie korzystał program VNC, nie są blokowane przez inne aplikacje. Jeżeli np. na komputerze używane jest oprogramowanie do filtrowania pakietów przychodzących z sieci, należy sprawdzić, czy nie blokuje ono portów 590x (używanych do eksportowania danych przez serwer VNC), 6000 (używanego do komunikowania się z systemem X Window w Linuksie i Uniksie), czy też 580x (wykorzystywanych do łączenia się z serwerem VNC poprzez internet).

Serwer vino VNC dla Ubuntu

Połączenie się z innym systemem Ubuntu znajdującym się w sieci to nic trudnego. Oprogramowanie potrzebne do połączenia z konsolą graficzną jest już zainstalowane. To przeznaczony dla GNOME serwer VNC o nazwie vino (plik wykonywalny nosi nazwę *vino-server*) zaprojektowany przez Marka McLoughlina.

Podstawowe informacje o programie vino można znaleźć w dokumencie dostępnym pod adresem <http://cvs.gnome.org/bonsai/cvsblame.cgi?file=vino/docs/remote-desktop.txt>, a także w dokumentacji znajdującej się w systemie: `/usr/share/doc/vino`. Jednak z mojego doświadczenia wynika, że program po prostu działa i nie ma konieczności wyszukiwania kolejnych informacji.



Vino działa inaczej niż większość serwerów VNC; tuż po zalogowaniu się do sesji GNOME program jest od razu uruchamiany. Może wówczas eksportować tylko zawartość ekranu wyświetlaną po zalogowaniu się użytkownika, czyli odmiennie niż większość serwerów VNC, które umożliwiają również zdalne wyświetlenie ekranu logowania, dzięki czemu można uruchomić sesję dla dowolnego użytkownika dostępnego w systemie. Zalety działających bardziej tradycyjnie serwerów VNC zostały omówione w ramce „Jak działają inne serwery VNC?”.

Aby upewnić się, czy program vino VNC jest włączony, należy wybrać z menu *System/Preferencje/Zdalny pulpit*, zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 15.3. Aby umożliwić zdalny dostęp do pulpitu innym użytkownikom, należy zaznaczyć w części *Udostępnianie* opcję *Zezwolenie innym użytkownikom na podgląd pulpitu*. To pozwoli innym tylko na podgląd ekranu; aby udostępnić im również możliwość korzystania z myszy i klawiatury, należy też zaznaczyć opcję *Zezwolenie innym użytkownikom na kontrolę pulpitu*.

Podczas konfigurowania serwera vino zawsze należy ustawić hasło w części *Bezpieczeństwo* (patrz rysunek 15.3). Aby je wprowadzić, należy zaznaczyć opcję *Wymaganie podania poniższego hasła*, a następnie wprowadzić odpowiednie hasło w polu tekstowym.

Jak działają inne serwery VNC?

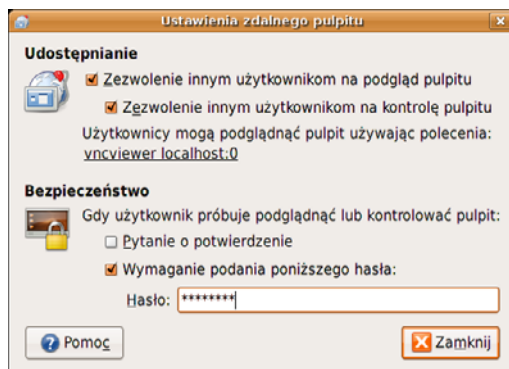
Większość serwerów VNC jest uruchamiana przez skrypt powłoki lub napisany w języku Perl o nazwie *vncserver* i eksportują raczej własny pulpit niż ten, do którego zalogował się użytkownik. Skrypt *vncserver* oferuje nieco bardziej elastyczny mechanizm przesyłania argumentów na serwer, wyświetlając informacje o jego statusie już po uruchomieniu, a także umożliwiając wykorzystanie skryptu startowego do identyfikacji menedżera okien i innych aplikacji, które powinny zostać uruchomione przez serwer VNC. Skrypt startowy serwera VNC zwykle znajduje się w pliku *~/vnc/startup*. Jeżeli podczas pierwszego uruchomienia serwera katalog i plik nie powstaną, wówczas katalog i skrypt zostaną utworzone na podstawie pliku startowego systemu X Window */etc/X11/xinit/xinitrc*.

Wykorzystanie skryptu startowego do eksportowania konsoli graficznej, która jest niezależna od konsoli fizycznej, umożliwia tradycyjnym serwerom VNC podążanie zawiązaną ścieżką łańcucha normalnie używanych plików startowych systemu X Window: *~/xinitrc*, *~/Xclients*, */etc/X11/xinit/xinitrc*, *etc/X11/Xsession* itd. Dzięki zastosowaniu różnych zmiennych systemowych pliki te umożliwiają uruchomienie różnorodnych środowisk graficznych i menedżerów okien.

W Ubuntu dostępne są serwery VNC korzystające z bardziej tradycyjnego modelu działania, takie jak *vnc4server* czy *tightvncserver*, które można zainstalować za pomocą menedżera pakietów Synaptic. Jeżeli użytkownik zdecyduje się na zainstalowanie któregoś z wymienionych, powinien wcześniej usunąć z systemu serwer *vino*, co pozwoli uniknąć potencjalnych konfliktów.

Rysunek 15.3.

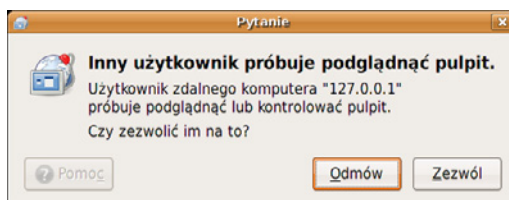
Konfigurowanie
serwera vino VNC
w Ubuntu



Można też zaznaczyć pole *Pytanie o potwierdzenie*, co spowoduje, że za każdym razem, kiedy ktoś będzie próbował połączyć się zdalnie z komputerem, będzie wyświetlane okno komunikatu widoczne na rysunku 15.4.

Rysunek 15.4.

Informacja o żądaniu
umożliwienia
zdalnego dostępu
do pulpitu Ubuntu



Włączanie tej opcji nie jest zalecane, ponieważ poważnie zmniejsza możliwość obsługi zdalnych połączeń VNC, a także dlatego, że serwer VNC zawsze powinien być chroniony hasłem.

Po włączeniu usługi zdalnego dostępu za pomocą okna dialogowego *Ustawienia zdalnego pulpitu* GNOME uruchomi proces *vino-server*, który od tej pory będzie automatycznie uruchamiany, gdy użytkownik będzie logował się w Ubuntu.

Serwer i klient VNC dla Mac OS X

Dla systemów Mac OS X dostępnych jest wiele różnorodnych programów klienckich i serwerów VNC. Ulubionym programem autora jest OSX VNC dostępny na stronie <http://sourceforge.net/projects/osxvnc>. Można pobrać go na dysk komputera jako plik obrazu dysku, który zostanie zamontowany, a instalator automatycznie uruchomiony. Po zainstalowaniu programu zostanie wyświetlone okno dialogowe, w którym można skonfigurować działanie programu (patrz rysunek 15.5).



Rysunek 15.5. Konfigurowanie narzędzia OSX VNC w systemie Mac OS X

Konfigurowanie tego narzędzia przypomina konfigurowanie innych serwerów VNC; użytkownik może ustawić hasło (zalecane) oraz ustalić inne szczegóły działania OSX VNC (niezalecane).

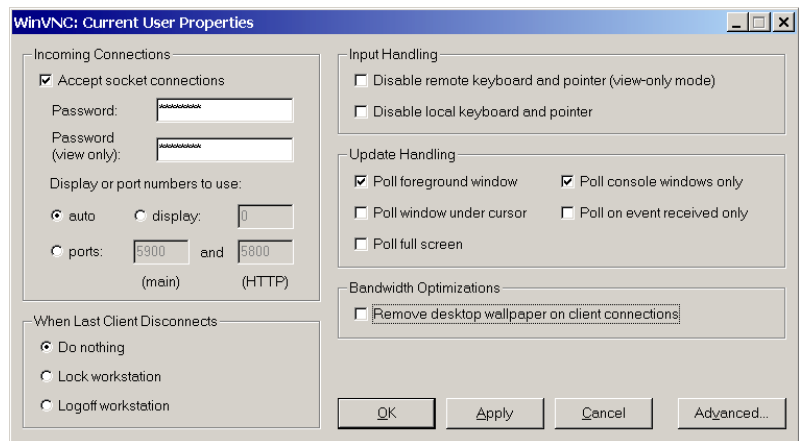
Ulubiony klient VNC dla systemów Mac OS X to — moim zdaniem — *Chicken of the VNC*. To niewielki, szybki program, który w dodatku ma zabawną nazwę. Czy można przebić takie połączenie? Najnowsza wersja programu jest do pobrania na stronie projektu <http://sourceforge.net/projects/cotvnc>.

Serwer i klient VNC dla Windows

TightVNC to mały, doskonale zoptymalizowany klient i serwer VNC. Dzięki kilku sprytnym sztuczkom z wykorzystaniem kompresji JPEG służącym wyświetlaniu kursora w różnych pozycjach znacznie zmniejszono zapotrzebowanie na przepustowość łącza podczas przesyłania danych. Tight VNC oferuje również zabezpieczenia, których nie ma w RealVNC łącznie z automatycznym tunelowaniem połączenia SSH. Chociaż dla Windows dostępne są aplikacje RealVNC, to jednak rekomenduję korzystanie z TightVNC. Stabilną oraz rozwojową wersję programu można pobrać ze stron www.tightvnc.com/download.html oraz ze strony SourceForge <http://sourceforge.net/projects/vnc-tight>.

Wystarczy pobrać plik z rozszerzeniem EXE z dowolnej strony, zawiera on aplikację serwerową oraz kliencką. Po pobraniu i zainstalowaniu, program można uruchomić wybierając odpowiednią pozycję z menu. Po jego uruchomieniu zostanie wyświetlone okno konfiguracji programu widoczne na rysunku 15.6, w którym użytkownik może ustawić hasło (zalecane) oraz ustalić inne szczegóły działania TightVNC (niezalecane).

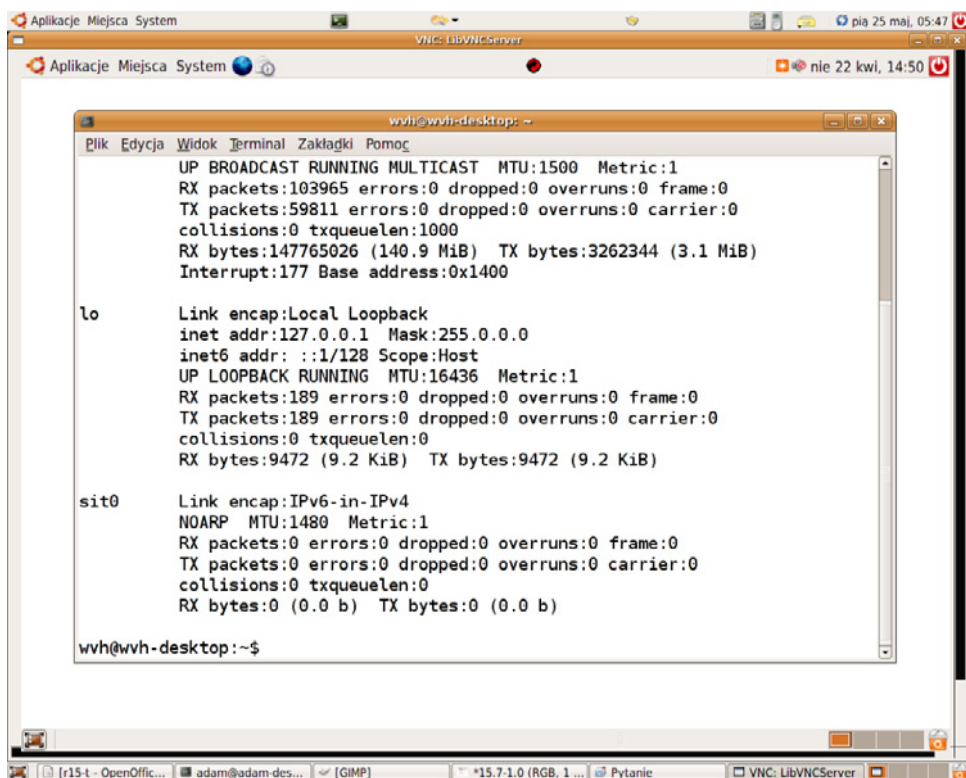
Rysunek 15.6.
Konfigurowanie programu Tight VNC w wersji serwerowej w systemie Windows



Jedną z dostępnych do skonfigurowania opcji jest to, czy program podczas działania ma wyświetlać swoją ikonę w zasobniku systemowym. Jest to bardzo wygodne rozwiązanie, ponieważ użytkownik może na bieżąco monitorować status programu, a także uzyskać łatwy dostęp do okna konfiguracji widocznego na rysunku 15.6.

Korzystanie z programu vncviewer

Kiedy program VNC serwer działa już na jednym z komputerów, można połączyć się z nim z jednego z systemów zdalnych, wykonując polecenie `vncviewer host:display`, gdzie `host` to nazwa komputera, na którym działa VNC w wersji serwerowej, a `display` to liczba wyświetlanych systemów. Podczas łączenia się z systemem Windows lub Mac OS X zwykle nie ma potrzeby podawania parametru `display`, chyba że serwer działający na tym komputerze został skonfigurowany tak, aby wyświetlać niestandardową liczbę pulpitów. Na rysunku 15.7 przedstawiam połączenie uzyskane za pomocą VNC pomiędzy dwoma komputerami.



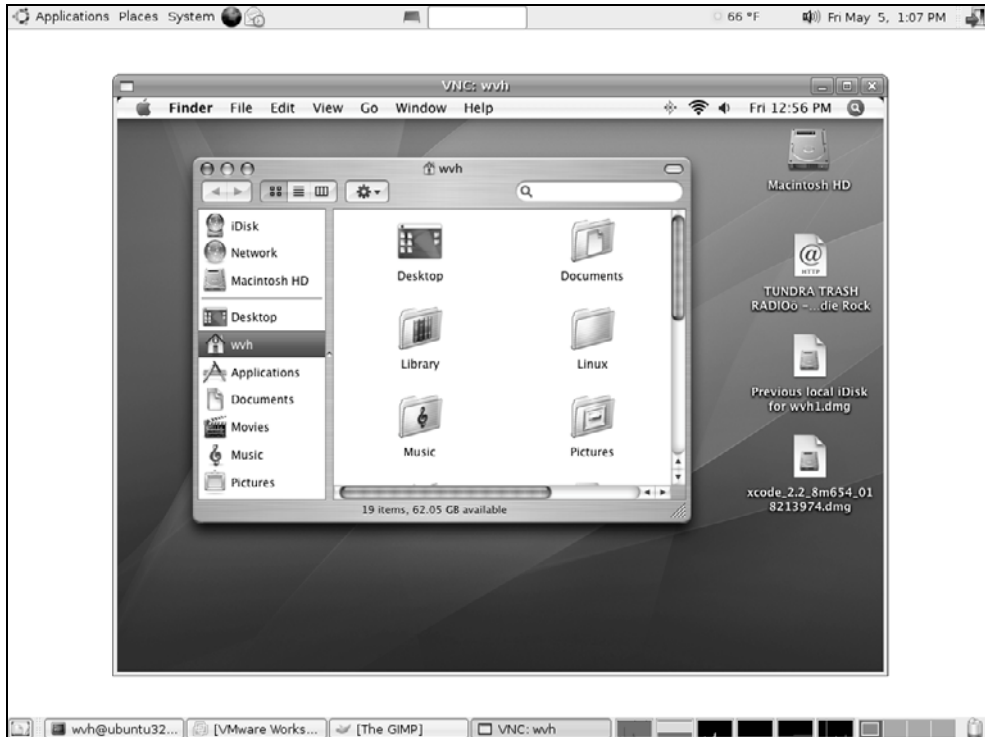
Rysunek 15.7. vncviewer i połączenie pomiędzy dwoma systemami Ubuntu

Jak wspominałem wcześniej, tuż po uruchomieniu serwera VNC na dowolnym komputerze można łączyć się z nim z dowolnego komputera z Ubuntu, co znacznie ułatwia zadania administracyjne, rozwiązywanie problemów i wprowadza w zdumienie dzieci, kiedy zdalnie zalogujemy się na ich komputerze. Na rysunku 15.8 przedstawiam połączenie ze zdalnym komputerem, na którym działa system Mac Os X.

Na rysunku 15.9 przedstawiam połączenie za pomocą VNC z systemem Microsoft.

Łączenie ze zdalnym serwerem usług terminalowych Windows

Dla większości z nas konieczność korzystania z systemów Windows to codzienny fakt, zwykle związany z pracą zawodową, a czasem też i z sieciami domowymi. W zastosowaniach prywatnych doskonale sprawdza się omówione wcześniej rozwiązanie oparte na VNC. Ale w przypadku, kiedy trzeba łączyć się zdalnie z systemem Windows i dysponuje się nieco wyższym budżetem, warto przyjrzeć się bliżej usłudze o nazwie Windows Remote Terminal.



Rysunek 15.8. vncviewer i połączenie z systemem Mac OS X

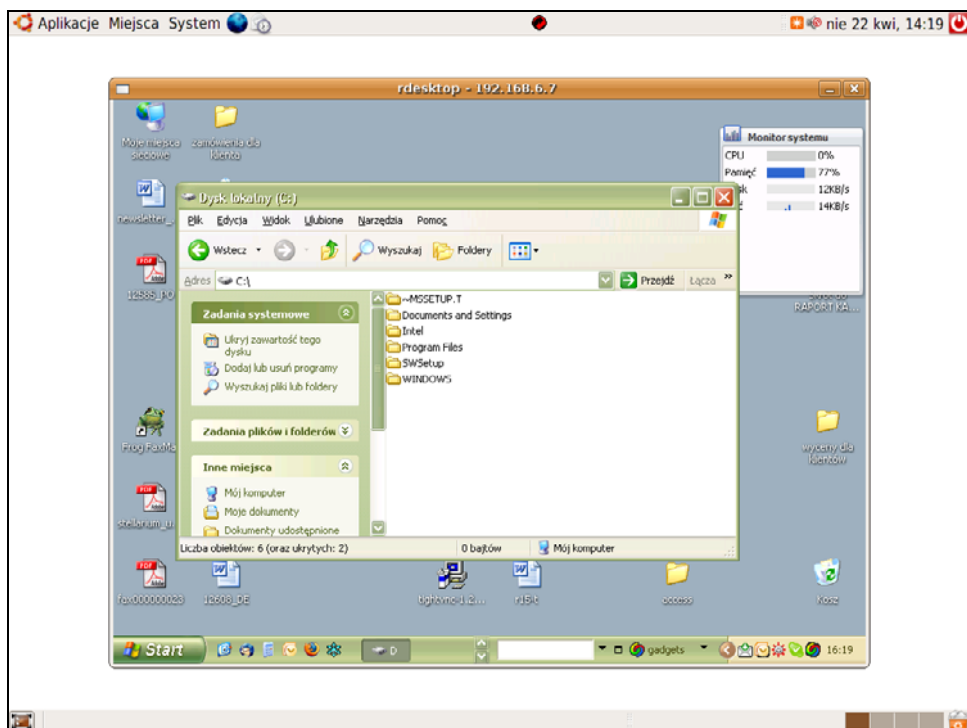
Jeśli Twoi użytkownicy potrzebują dostępu do systemu Windows tylko raz na jakiś czas, a Ty wolisz minimalizować wydatki, niż wyposażać każdego w osobny komputer, to dobrym rozwiązaniem może okazać się wykorzystanie serwera usług terminalowych Windows. Wystarczy jeden "wypasiony" komputer i zakup kilku licencji klienckich. Od tego momentu będzie można korzystać ze zdalnego dostępu. Na szczęście, ta metoda nie wymaga posiadania systemów Windows na komputerach, które będą łączyć się z serwerem usług terminalowych. Linux oferuje łatwy dostęp do tej usługi za pomocą narzędzi *rdesktop* lub *tsclient*, dwóch pakietów open source komunikujących się przy pomocy *Remote Desktop Protocol* (RDP) używanym w Windows Terminal Services.



Podczas korzystania z Terminal Services w Microsoft Windows Domain Controller lub serwera należy się upewnić, że użytkownicy, którzy mają uzyskać do niego dostęp, mogą logować się lokalnie lub należą do grupy, która posiada takie uprawnienia. W przeciwnym razie będzie wyświetlany komunikat o tym, że lokalne zasady bezpieczeństwa uniemożliwiają logowanie do systemu i trzeba będzie każdemu dać osobny komputer.

Korzystanie z programu rdesktop

Pakiet *rdesktop* (www.rdesktop.org) umożliwia zdalne łączenie się z pulpitem i jest przeznaczony dla systemów linuksowych i uniksowych, na których uruchomiony jest X Window. Pakiet jest instalowany domyślnie w Ubuntu i oferuje doskonałą, wydajną aplikację kliencką z wieloma przydatnymi funkcjami opisanymi poniżej. Program dostępny jest



Rysunek 15.9. vncviewer i połączenie z systemem Microsoft

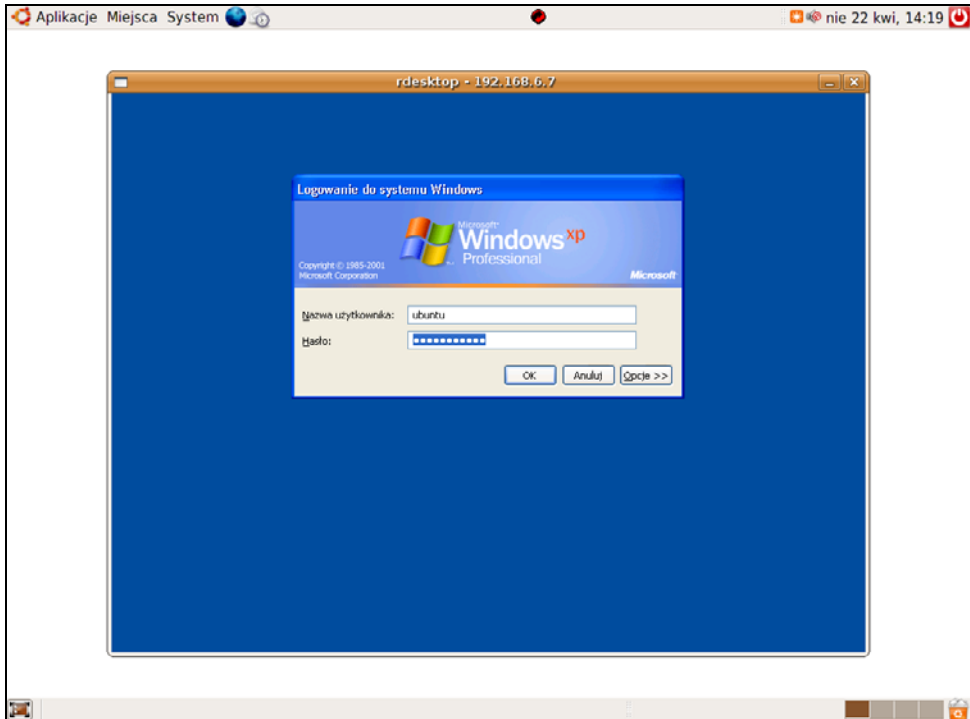
w repozytoriach Ubuntu, ale ze strony <http://sourceforge.net/projects/rdesktop> można pobrać najnowsze pliki (źródła i binaria) oraz wersje przeznaczone dla innych systemów operacyjnych.

Program uruchamia się za pomocą linii poleceń lub xterm, można też dodać odpowiednią pozycję do menu GNOME (zgodnie z informacjami zamieszczonymi w rozdziale 5.). Najprostszym poleceniem używanym do łączenia się z działającym Windows Terminal Services jest `rdesktop host`, gdzie `host` to nazwa lub numer IP komputera, z którym ma zostać nawiązane połączenie. Na rysunku 15.10 przedstawiam ekran logowania się do systemu Windows wyświetlony w Ubuntu.

Po zalogowaniu się i — w razie potrzeby — podaniu nazwy domeny program `rdesktop` wyświetli pulpit systemu Windows (patrz rysunek 15.11). Ten przypadek jest szczególnie interesujący, ponieważ widać w nim edytowany w OpenOffice.org rozdział niniejszej książki, który znajduje się w systemie Ubuntu, a Windows ma do niego dostęp, dzięki połączeniu z Smbą (więcej informacji na ten temat zamieszczam w rozdziale 32., „Konfiguracja serwera Samba”).

Program `rdesktop` oferuje kilka opcji, które ułatwiają dostęp do usług terminala Windows. Oto te, które najczęściej wykorzystuję:

- ♦ `-d` — nazwa domeny, do której użytkownik chce się zalogować,
- ♦ `-f` — tryb pełnoekranowy; zdalny pulpit, który będzie wyświetlany, zajmie cały ekran; aby wyłączyć tryb pełnoekranowy, należy użyć kombinacji klawiszy `Ctrl+Alt+Enter`,



Rysunek 15.10. Ekran logowania się do Windows wyświetlony za pomocą programu rdesktop

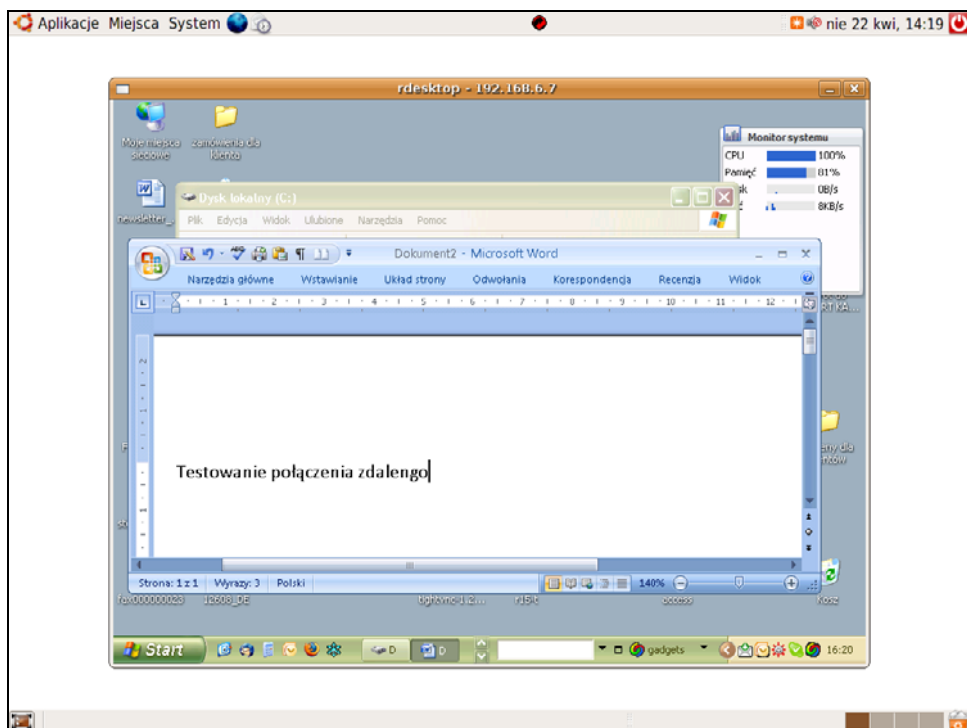
- ♦ -p — hasło do zdalnej domeny,
- ♦ -u — nazwa użytkownika, na którego konto ma nastąpić zalogowanie.

Kiedy na komputerze zdalnym działa Windows XP lub jeszcze nowsza wersja Windows, bardzo przydatną opcją, która nie została wymieniona powyżej, jest -r; umożliwia ona udostępnienie zasobów Linuksa pod Windows. Jest to wygodne zwłaszcza wówczas, gdy trzeba np. zamapować lokalną kolejkę wydruku do drukarki znajdującej się w systemie, z którym się łączymy. Składnia tego polecenia wygląda następująco: -r printer: nazwa_kolejki_wydruku oraz -r disk:nazwa_udziału=/urządzenie/ścieżka. Aby np. wydrukować dokument zakolejkowany jako *SuperScript-660i*, należy użyć następującego polecenia: -r printer: SuperScript-660i. Na rysunku 15.12 przedstawiam lokalną kolejkę wydruku w oknie dialogowym Windows.

Podobnie sprawa wygląda z mapowaniem udostępnionego napędu CD-ROM; do polecenia rdesktop należy dodać opcję -r disk:cdrom=/dev/cdrom, jeżeli użytkownik chciałby użyć stacji dyskiek, wówczas dodana opcja powinna wyglądać następująco: -r disk: floppy=/dev/fd0. Nazwa udziału nie może być dłuższa niż 8 znaków.

Korzystanie z programu tsclient

Jeżeli z pewnych powodów użytkownik nie lubi programu rdesktop, Ubuntu oferuje inną możliwość łączenia się z Windows Terminal Server. Jest to program o nazwie tsclient (*Terminal Services Client*), więcej informacji o nim można znaleźć na stronie <http://>



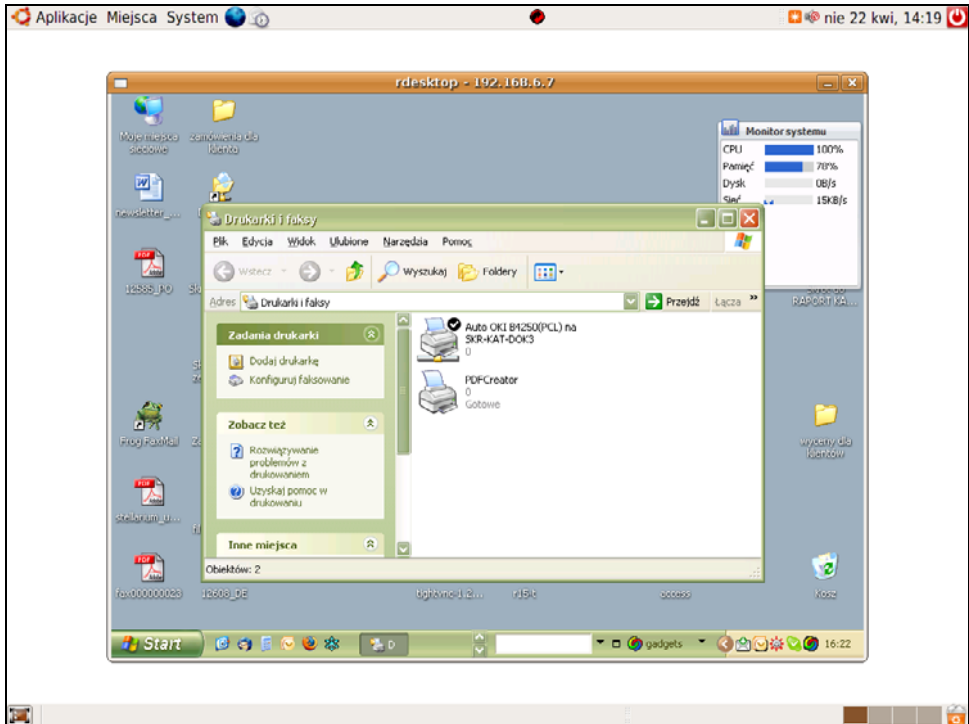
Rysunek 15.11. Windows XP w programie rdesktop

gnomepro.com/tsclient. Jest to aplikacja zoptymalizowana do działania w środowisku GNOME. Umożliwia zapisywanie informacji na temat wykonanych połączeń, dzięki czemu użytkownicy mogą szybko uzyskać dostęp do odpowiednich danych oraz je edytować. Chociaż ostatnia wersja *tsclient* jest zawsze dostępna w repozytoriach Ubuntu, można też pobrać kod źródłowy i paczki dla innych systemów linuksowych korzystających z GNOME, wystarczy odwiedzić stronę <http://sourceforge.net/projects/ts-client>.

Aby uruchomić program *tsclient*, należy kolejno wybrać z menu *Aplikacje/Internet/Terminal Server Client*, zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 15.13.

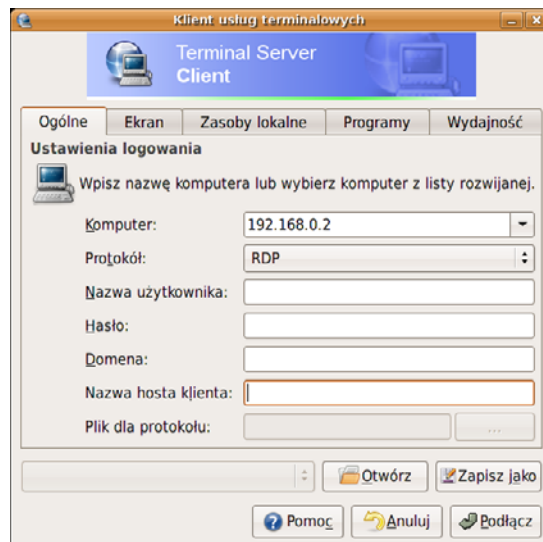
Jeżeli zostanie wprowadzona tylko nazwa terminala, z którym użytkownik chce się połączyć, zostanie wyświetlony standardowy ekran logowania (patrz rysunek 15.14), a użytkownik będzie musiał podać login i hasło.

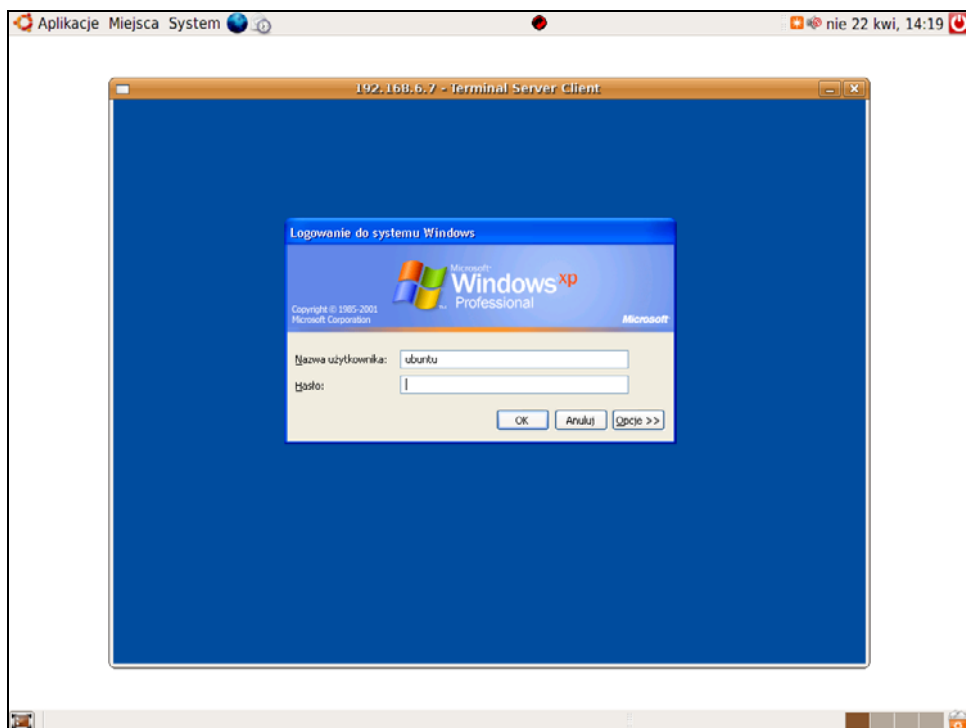
Aby w przyszłości ułatwić łączenie z często wykorzystywanymi serwerami, należy w oknie widocznym na rysunku 15.13 wypełnić wszystkie wymagane pola i zapisać profil danego połączenia pod odpowiednią nazwą (używając przycisku *Zapisz jako*). Program zapisze wszystkie dane (z wyjątkiem hasła) w pliku z rozszerzeniem *.rdp*, który będzie przechowywany w katalogu *~/.tsclient*, w katalogu domowym użytkownika. Następnym razem, kiedy użytkownik będzie chciał nawiązać jedno z zapisanych połączeń, wystarczy wybrać plik o odpowiedniej nazwie i wprowadzić hasło, reszta danych zostanie pobrana z zapisanego wcześniej pliku.



Rysunek 15.12. Drukowanie za pomocą drukarki podłączonej przez rdesktop

Rysunek 15.13.
Ekran startowy
programu tsclient





Rysunek 15.14. Logowanie się w systemie za pomocą programu *tsclient*

Korzystanie z innych narzędzi do zdalnego łączenia się z komputerem

Dla Ubuntu dostępne są również inne graficzne narzędzia służące do zdalnego logowania się na komputerach, choć nie są oficjalnie wspierane przez programistów Ubuntu, np. darmowe wersje NoMachine NX, znane jako FreeNX. Ten pakiet (*freenx*) nie jest dostępny za pośrednictwem repozytoriów Ubuntu i nie jest też standardowo instalowany w systemie, przede wszystkim dlatego, że przeznaczony jest dla środowiska KDE, co wymaga zainstalowania wielu dodatkowych bibliotek. Darmowa wersja FreeNX dostępna jest w innych repozytoriach wymienionych w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”; w tym rozdziale można znaleźć dodatkowe informacje na ten temat.

FreeNX to wolna (GPL) implementacja programu NoMachine NX Server (www.nomachine.com), w którym zastosowano technologię kompresji danych, co pozwoliło znacznie zmniejszyć ilość informacji systemu X Window, wprowadzono też inne usprawnienia, takie jak pamięć podręczna czy zoptymalizowanie protokołu. W pakiecie *freenx* znajduje się aplikacja kliencka NX dla Linuksa, ze strony producenta można pobrać też wersje dla systemów Microsoft Windows, Mac OS X czy Solaris. Podstawowa różnica pomiędzy wersjami płatnymi i bezpłatnymi leży głównie w warunkach użytkowania i — oczywiście — kosztach. FreeNX oferuje wszystkie główne możliwości dostępne w wersji komercyjnej, z wyjątkiem obsługi protokołów SMB i CUPS.

Jeżeli nawet czytelnicy są fanami technologii VNC, warto zapoznać się NX chociażby ze względu na technologię kompresji danych i wprowadzone usprawnienia, a także dlatego, że do komunikowania się używany jest bezpieczny protokół SSH. FreeNX to efektywny zamiennik VNC, obsługuje też technologie Linux Terminal Server (www.ltsp.org), Windows Terminal Server, Citrix Metaframe i standardowy dla systemu X Window XDMCP (ang. *X Display Manager Control Protocol*). Ponieważ jednak program nie jest dostępny w oficjalnych repozytoriach Ubuntu, w razie kłopotów nie należy liczyć na pomoc ze strony Canonical.

Podsumowanie

Dziś sieci komputerowe są obecne na uczelniach, w przedsiębiorstwach i w domach. Wraz z ich rozwojem pojawia się coraz większa potrzeba dostępu do różnorodnych systemów. W niniejszym rozdziale omówiłem bezpieczne narzędzia linii poleceń, umożliwiające łączenie się ze zdalnymi systemami Linux i Unix oraz opisałem instalowanie serwera SSH, za którego pomocą z danym komputerem w bezpieczny sposób mogą łączyć się inni użytkownicy. Następnie omówiłem znakomity program klienta i wersję serwerową programu VNC (ang. *Virtual Network Computing*), przy których użyciu można zdalnie łączyć się z systemami Microsoft Windows, Mac OS X, Linux i Unix. Na końcu rozdziału zamieściłem informacje na temat mechanizmów łączenia się z Windows Terminal Server za pomocą standardowych narzędzi dostępnych w Ubuntu; opisałem także programy NX i FreeNX, które z kolei umożliwiają bezpieczne połączenie niemal z każdą graficzną zdalną konsolą.

W rozdziale 16. omówię łączenie się ze zdalnymi serwerami plików udostępniającymi znajdujące się na nich pliki. Napiszę też, w jaki sposób współdzielić pliki znajdujące się w systemie Ubuntu, dzięki czemu inne systemy operacyjne znajdujące się w sieci będą mogły łatwo uzyskać do nich dostęp.

Rozdział 16.

Przesyłanie i udostępnianie plików w Ubuntu

W tym rozdziale:

- ◆ Przesyłanie plików między różnymi systemami
- ◆ Udostępnianie plików
- ◆ Udostępnianie plików w systemach Linux i Unix
- ◆ Udostępnianie plików w sieciach P2P (peer-to-peer)

Wcześniej czy później nadejdzie taki moment, kiedy użytkownik będzie chciał przekopiować jakiś plik z jednego komputera na inny. W czasach przedsiéciowych i przedmałowych (których pewnie większość czytelników nie pamięta) oznaczało to zwykle, że dany plik należało nagrać na zewnętrzny nośnik pamięci (najczęściej dyskietkę), a następnie przenieść do drugiego komputera i skopiować na dysk twardy. Ta procedura była znana pod nazwą sieci dyskietkowej lub pieszej (ang. *sneaker net*) — poczucie humoru komputerowych dziwaków jest niesamowite.

Dzisiaj, dzięki temu, że sieci są niemal wszechobecne w większości przedsiębiorstw, ale też i w wielu domach, elektroniczny przesył danych stał się ważnym ich aspektem. Jak można się spodziewać, istnieje wiele metod przesyłania lub kopiowania danych pomiędzy komputerami, począwszy od *point-to-point* do *peer-to-peer* (P2P), tak „lubianego” przez koncerty fonograficzne i wytwórnie filmowe. W pierwszym przypadku pliki są kopiowane pomiędzy dwoma komputerami i zwykle użytkownik posiada dane identyfikujące obie maszyny. Podczas przesyłania plików w systemach P2P grupy komputerów biorące udział w procesie udostępniania (zarówno pobierające, jak i wysyłające) nie muszą mieć specjalnych danych o innych komputerach uczestniczących w tym samym procesie. Oczywiście, podczas wymiany plików w sieciach P2P ktoś musi wiedzieć, które komputery udostępniają jakie pliki, ale ta funkcja zwykle wbudowana jest w oprogramowanie kontrolujące cały proces.

W pierwszej części rozdziału omówię wymianę plików za pomocą popularnych technologii *point-to-point* dostępnych w systemie Ubuntu. Drugą część poświęcę rozwiązaniom działającym w sieciach P2P, które również można łatwo i szybko zainstalować w Ubuntu.



Jak zapewne czytelnik zdążył zauważyć, nie jestem prawnikiem i mam w głębokiej pogardzie grupy przemysłowe starające się ograniczać rozwój technologii tylko dlatego, że mogą one posłużyć do zrobienia czegoś nielegalnego. Można sobie wyobrazić użycie komputera przenośnego jako maczugi w celu odebrania komuś życia, ale czy jest to powód do wyjęcia spod prawa wszystkich laptopów? W idei udostępniania plików w sieciach P2P nie ma niczego nielegalnego. Jest to znakomita metoda na efektywne współdzielenie i wymianę plików między wieloma systemami, które udostępniają innym pobrane pliki. Do czego dokładnie posłuży ta technologia, to już indywidualna sprawa każdego użytkownika. Do mnie należy tylko wyjaśnienie, jak działa.

Przesyłanie plików do innych systemów

Jak wspominałem we wprowadzeniu do niniejszego rozdziału, ta jego część poświęcona jest omówieniu metod kopiowania plików pomiędzy komputerami w sytuacji, kiedy użytkownik zna tożsamość komputerów biorących udział w tym procesie. Oto najpowszechniejsze przyczyny przesyłania plików: „Muszę skopiować ten plik z mojego laptopa na komputer stojący na moim biurku”, „Chcę umieścić ten plik na naszym domowym serwerze, dzięki czemu będzie można go odtwarzać na każdym komputerze wpiętym do naszej sieci” albo też „Muszę skopiować ten arkusz kalkulacyjny na serwer firmowy, tak aby szef widział, nad czym pracuję w domu”. We wszystkich przypadkach użytkownik posiada autoryzowany dostęp do obu komputerów uczestniczących w przesyłaniu danych, zna ich nazwy (niezależnie od tego, czy są to pełne nazwy hostów typu *fip.vonhagen.org*, czy też adresy IP, takie jak np. *207.44.142.34*, czy też nazwy używane w sieci lokalnej, np. *laptop*, *serwer1* itd.).

Oprócz autoryzowanego dostępu do systemu, na który kopiowane są pliki, musi być na nim uruchomiona odpowiednia usługa, rodzaj elektronicznego chwytaka, który złapie wysyłane pliki. W niniejszym rozdziale skupię się komputerach-klientach, łączących się z serwerami, na których umieszcza się dane, przedstawię też informacje niezbędne do odpowiedniego skonfigurowania komputera tak, aby mógł on połączyć się z komputerem zdalnym. Nie zamieszczę tu wskazówek, w jaki sposób skonfigurować serwer FTP czy udostępniać katalogi w systemach Windows czy Linux. Informacje o tym, jak można udostępniać w Linuksie katalogi, aby były osiągalne dla komputerów pracujących pod kontrolą Microsoft Windows, Mac OS X czy Linuksa, można znaleźć w rozdziale 32., „Konfiguracja serwera Samba”. Informacje o tym, jak udostępniać w Linuksie katalogi, aby były one osiągalne dla komputerów pracujących pod kontrolą Mac OS X czy Linuksa można znaleźć w rozdziale 31., „Konfiguracja serwera NFS”. Z serwerów NFS można też korzystać w systemach Microsoft Windows, ale pod warunkiem zainstalowania dodatkowego oprogramowania, które nie znajduje się w nich domyślnie. Informacje na temat konfigurowania serwera FTP można znaleźć w rozdziale 31.



Inną powszechną przyczyną kopiowania plików jest prośba: „Mój brat bardzo chciałby mieć ten plik”. Zwykle w takich przypadkach doradzam dołączenie danego pliku do wiadomości elektronicznej, z którą może być wysłany. Użytkownik zwykle nie zna nazwy komputera odbiorcy, jego adresu IP i nie ma innego sposobu bezpośredniego dostępu.

Korzystanie z FTP

Skrót FTP zapisany dużymi literami oznacza *File Transfer Protocol*, jest to najstarszy sposób przesyłania plików w sieciach z wykorzystaniem protokołu TCP/IP, który jest podstawowym budulcem internetu. Jeżeli użyte są małe litery (ftp), oznacza to nazwę polecenia, które można wykonać w celu nawiązania połączenia ze zdalnym serwerem, co umożliwia z kolei dwukierunkowy przesył danych. Za pomocą Ubuntu można, zgodnie z zamieszczonymi poniżej informacjami, w łatwy sposób korzystać z serwerów FTP — w środowisko GNOME wbudowana jest obsługa FTP, z kolei w linii poleceń można skorzystać z wydajnego narzędzia ftp, dostępne jest też dodatkowe narzędzie graficzne do obsługi tego protokołu.

FTP to stary protokół, a polecenia konieczne do jego obsługi są proste, nawet podczas korzystania z linii poleceń. Ponieważ jednak istnieje wiele narzędzi linii poleceń, takich jak `wget` czy `curl`, które do pobierania plików wykorzystują nowsze protokoły (np. HTTP), może się okazać, że użytkownik nigdy nie będzie musiał korzystać z narzędzia ftp. Zdecydowałem się jednak na zamieszczenie odpowiednich informacji — jeżeli nie będą przydatne, zawsze można je pominąć. Jeżeli natomiast okażą się pomocne, wówczas obie strony, ja i czytelnik, będą zadowolone.

Łączenie się z serwerami FTP za pomocą narzędzi środowiska GNOME

W menu GNOME *Miejsca* znajduje się polecenie *Połącz z serwerem*, za którego pomocą można utworzyć na pulpicie ikonę skrótu do serwera FTP, który z kolei będzie można łatwo eksplorować przy użyciu menedżera plików Nautilus. Po wybraniu kolejno z menu *Miejsca/Połącz z serwerem* zostanie wyświetlone okno dialogowe widoczne na rysunku 16.1.

Rysunek 16.1.

Konfigurowanie połączenia z publicznym serwerem FTP



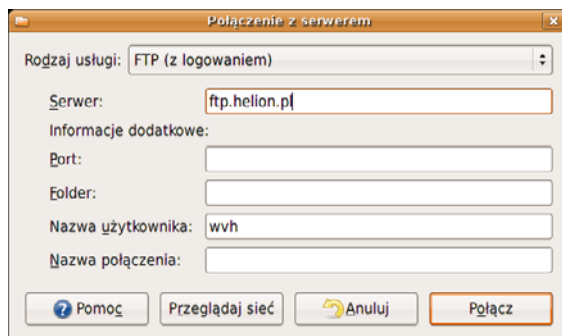
Zwykle serwer, z którym chce się połączyć użytkownik, wymaga uwierzytelnienia (czyli podania nazwy użytkownika oraz hasła; więcej informacji na ten temat zamieszczam w dalszej części rozdziału), dlatego też wyświetlana domyślnie możliwość łączenia się z publicznym serwerem FTP nie jest najlepszym rozwiązaniem. Jak można się domyślać, publiczne serwery FTP nie wymagają uwierzytelniania, dlatego też często określa się je nazwą **anonimowych serwerów FTP**. Jednak nawet podczas łączenia się z anonimowym serwerem FTP należy podać przynajmniej jego nazwę lub adres IP. Jeżeli użytkownik chciałby, aby wyświetlana na pulpicie ikona miała określoną nazwę, można ją wpisać w polu *Nazwa połączenia*; rzadko korzystam z tej możliwości, pozostawiając oryginalną nazwę

połączenia, która zwykle jest też nazwą danego serwera. Pole *Port* należy uzupełnić tylko wówczas, gdy zdalny serwer działa na niestandardowym porcie TCP/IP. Nazwę folderu można podać wówczas, gdy użytkownik chce od razu przejść do określonego katalogu, innego niż ten, którego zawartość wyświetlana jest po zalogowaniu się na serwer. Po kliknięciu przycisku *Połącz* na pulpicie zostanie utworzona odpowiednia ikona.

Jeżeli użytkownik chce nawiązać połączenie z serwerem FTP wymagającym uwierzytelniania, wówczas należy rozwinąć listę *Rodzaj usługi* i wybrać opcję *FTP z logowaniem*. Zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 16.2.

Rysunek 16.2.

Konfigurowanie
połączenia
z serwerem FTP
wymagającym
uwierzytelnienia



Podobnie jak w przypadku anonimowego serwera FTP, użytkownik musi podać przynajmniej jego nazwę lub adres IP. Analogicznie można wprowadzić własną nazwę dla wyświetlanej na pulpicie ikony, ale zwykle pozostawia się oryginalną nazwę połączenia, która z reguły jest nazwą danego serwera. Można też podać nazwę użytkownika, który będzie logował się na serwerze, jeżeli nie zostanie ona wpisana teraz, trzeba będzie uzupełnić ją w kolejnym oknie dialogowym. Tak jak w przypadku anonimowego serwera FTP, pole *Port* należy uzupełnić tylko wówczas, gdy zdalny serwer działa na niestandardowym porcie TCP/IP. Nazwę folderu można podać wówczas, gdy użytkownik chce od razu przejść do określonego katalogu, innego niż ten, którego zawartość wyświetlana jest po zalogowaniu się. Po kliknięciu przycisku *Połącz* na pulpicie zostanie utworzona odpowiednia ikona.



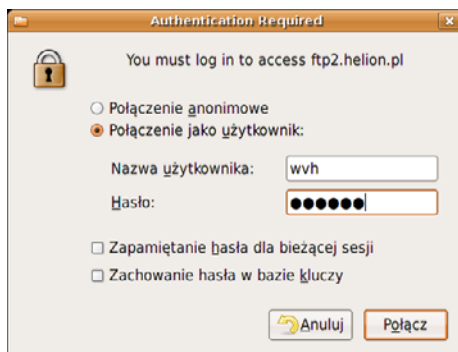
Jeżeli przyjrzeć się bliżej ikonom tworzonej podczas konfigurowania połączenia z serwerami FTP zarówno anonimowymi, jak i wymagającymi uwierzytelniania, da się zauważyć, że znajduje się na nich symbol połączenia sieciowego, a także czerwony znacznik z umieszczonymi wewnątrz literami FTP. Osoby przechowujące dużo ikon na pulpicie i posiadające przy okazji dobry wzrok mogą łatwo zidentyfikować odpowiednie protokoły. Jest to szczególnie przydatne podczas równoczesnego korzystania z udostępnianych katalogów Microsoft Windows, o czym powiem w dalszej części rozdziału.

Samo utworzenie na pulpicie ikony symbolizującej połączenie z serwerem anonimowym lub wymagającym uwierzytelnienia nie oznacza jeszcze jego nawiązania. Aby to zrobić, należy kliknąć dwukrotnie daną ikonę. Jeżeli połączenie będzie nawiązywane z serwerem wymagającym uwierzytelnienia, wówczas zostanie wyświetlone okno dialogowe widoczne na rysunku 16.3, w którym należy wprowadzić odpowiednie informacje.

Jeżeli połączenie będzie następowało z serwerem wymagającym uwierzytelniania i nie zostanie podana nazwa użytkownika, wówczas okno widoczne na rysunku 16.3 da jeszcze

Rysunek 16.3.

Kompletowanie informacji potrzebnych do zalogowania się na serwerze wymagającym uwierzytelniania

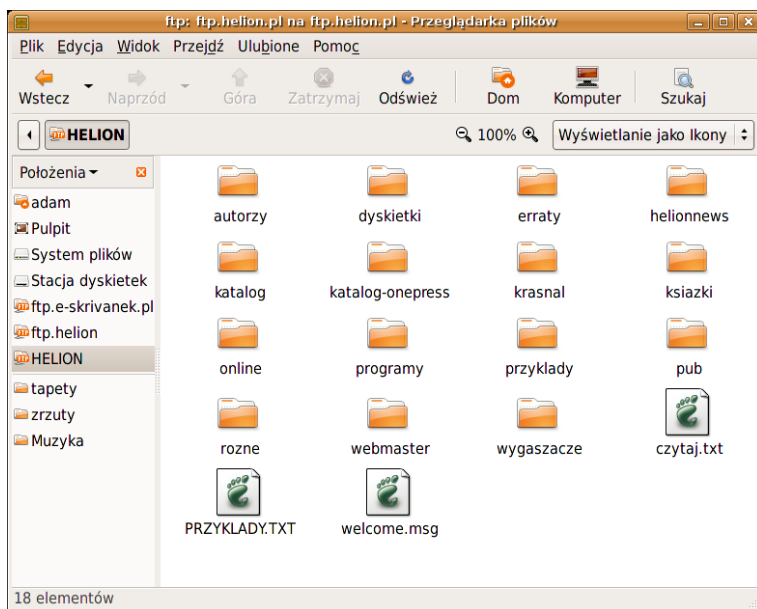


szansę zalogowania się anonimowo. W innym przypadku, aby zalogować się na serwerze, należy najpierw zaznaczyć pole *Połączenie jako użytkownik*, a następnie wpisać odpowiednią nazwę użytkownika oraz hasło.

Po wprowadzeniu odpowiedniej nazwy użytkownika i hasła (lub połączeniu się z serwerem anonimowo) menedżer plików Nautilus wyświetli zawartość domyślnego katalogu przypisanego dla logującego się użytkownika (patrz rysunek 16.4).

Rysunek 16.4.

Połączenie z serwerem FTP za pomocą menedżera plików Nautilus



Kiedy w Nautilusie zostanie wyświetlona zawartość danego serwera FTP, pliki można kopiować i przenosić, korzystając ze standardowych metod stosowanych w środowisku GNOME, opisanych w rozdziale 5.

Kiedy wszystkie potrzebne operacje zostaną wykonane, należy zamknąć okno Nautilusa. Aby usunąć ikonę danego połączenia z pulpitu, trzeba ją zaznaczyć, następnie za pomocą prawego przycisku myszy wyświetlić menu kontekstowe i wybrać z niego opcję *Odmontuj wolumin*. Jeżeli użytkownik planuje nawiązywać połączenia z danym wybranym serwerem

w przyszłości, wówczas należy pozostawić ikonę na pulpicie; następnym razem będzie można zalogować się na serwerze, podając tylko dane uwierzytelniające, bez konieczności ponownego konfigurowania połączenia.

Łączenie się z serwerami FTP za pomocą narzędzi linii poleceń

Z serwerami FTP można też łączyć się, wpisując w terminalu GNOME lub xterm polecenie `ftp` oraz podając nazwę komputera lub jego adres IP. W poniższym przykładzie nawiązano połączenie z serwerem FTP o nazwie *ftp.helion.pl*:

```
$ ftp ftp.helion.pl
Connected to 192.168.6.24
220-Welcome to Pure_FTPd.
220-You are user number 2 of 10 allowed
220-This is a private system - No anonymous login
220-IPv6 connection are also welcome on this server.
220- You will be disconnected after 15 minutes of inactivity:
Name (192.168.6.64:wvh):
```

Podczas łączenia się z serwerem FTP za pomocą narzędzi linii poleceń zostaną wyświetlone pewne informacje o samym serwerze oraz systemie, na którym działa. Użytkownik może zostać poproszony o podanie odpowiedniej nazwy i hasła. To, ile informacji zostanie wyświetlonych, zależy od użytego na serwerze oprogramowania oraz szczegółów jego konfiguracji.

Jak widać w ostatniej linii przykładu, serwer próbuje pozyskać do zalogowania użytkownika pewne dostępne dane. Ponieważ użytkownik zalogowany w systemie, z którego następuje łączenie z serwerem, ma nazwę *wvh*, program zaproponował to jako nazwę potrzebną do zalogowania się na serwerze FTP. Jeżeli nazwa jest poprawna, wówczas wystarczy wcisnąć klawisz *Enter*, aby ją potwierdzić. Jeżeli użytkownik chciałby do zalogowania się użyć innej nazwy, wówczas należy ją wprowadzić właśnie w tym miejscu; w poniższym przykładzie użyto nazwy *vonhagen*:

```
Name (192.168.6.64:wvh): vonhagen
331 User vonhagen OK. Password required
Password:
```



Podczas anonimowego logowania się na serwery FTP zwykle jako nazwy użytkownika używa się wpisu *anonymous*, natomiast jako hasło podaje się swój adres poczty elektronicznej.

Po wprowadzeniu nazwy użytkownika zdalny serwer poprosi o wprowadzenie hasła, po czym należy wcisnąć klawisz *Enter*. Jeżeli wprowadzone nazwa użytkownika oraz hasło będą poprawne, zostanie wyświetlony komunikat potwierdzający ich zaakceptowanie, a także zobaczymy dodatkowe informacje na temat konfiguracji serwera. W kolejnej linii zostanie wyświetlony znak zachęty (`ftp>`) informujący, że system jest gotowy na przyjęcie poleceń od użytkownika (patrz poniższy przykład).

```
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Korzystać z serwera można za pomocą poleceń, których lista wyświetli się po wpisaniu za znakiem zachęty znaku zapytania (?):

```
ftp> ?
Commands may be abbreviated.  Commands are:
!          debug          mdir       qc             send
$          dir            mget       sendport      site
account   disconnect      mkdir      put           size
append    exit              mls        pwd           status
ascii     form              mode       quit          struct
bell      get               modtime    quote         system
binary    glob             mput      recv          sunique
bye       hash             newer      reget         tenex
case      help            nmap      rstatus       tick
cd        idle           nlist     rhelp         trace
cdup      image          ntrans    rename        type
chmod     lcd            open      reset         user
close     ls             prompt    restart       umask
cr        macdef         passive   rmdir         verbose
delete    mdelete        proxy     runique       ?
```

Ponieważ szanse na to, że użytkownik nigdy nie będzie miał okazji skorzystania z większości tych poleceń, są całkiem spore, a także dlatego, że objaśnienie każdego z nich można znaleźć w podręczniku systemowym uruchamianym w terminalu poleceniem `man ftp`, poniżej omówię tylko te, które są najczęściej używane do wyszukiwania plików i pobierania ich na dysk lokalny komputera.

Serwery FTP eksportują hierarchię katalogów dokładnie tak, jak ma to miejsce w systemach Linux. Dzięki temu można skorzystać ze znanych poleceń, takich jak np. `pwd`, aby sprawdzić nazwę bieżącego katalogu, `cd` do zmiany katalogu itd. Jeżeli podczas logowania użyto nazwy użytkownika oraz hasła, wówczas bieżącym katalogiem będzie prawdopodobnie standardowy katalog do transferu danych (o nazwie `/pub`) lub przypisany danemu użytkownikowi katalog domowy.

Kiedy użytkownik przejdzie do katalogu, w którym chce umieścić pliki lub z którego chce je skopiować, wówczas należy użyć odpowiednio polecenia `put`, aby wysłać pliki na serwer, lub `get`, aby je z niego pobrać. Przed rozpoczęciem przesyłania pliku należy się upewnić, czy jest ustawiony odpowiedni tryb. Większość serwerów automatycznie ustawia tryb transferu plików tak, aby użytkownik mógł pobierać pliki binarne, takie jak grafikę, archiwa (w formatach `.zip`, `.tgz`, `.bz2`, `.tar`, `.pax`, itp.), pliki muzyczne itp. Zawsze można się upewnić, czy serwer pracuje w odpowiednim trybie i wykonać polecenie `bin`:

```
ftp> bin
200 Switching to Binary mode.
ftp>
```

Jeżeli pliki nie będą przesyłane w trybie binarnym, będą przesyłane jako pliki ASCII, co spowoduje ich uszkodzenie, a w konsekwencji uniemożliwi ich uruchomienie. Jeżeli po pobraniu, danego pliku nie uda się go wyświetlić, odtworzyć czy rozpakować, przyczyną może być pominięcie powyższego kroku.

W ustawieniach domyślnych podczas przesyłania danych (w obie strony) nie są wyświetlane żadne informacje, ma to miejsce dopiero po zakończeniu transferu. Jeżeli użytkownik

chciałby wiedzieć, co aktualnie dzieje się na serwerze, powinien wprowadzić polecenie `hash`, które spowoduje, że po pobraniu każdego kilobajta danych będzie wyświetlany znak `#`. Pierwsze wpisanie polecenia `hash` powoduje włączenie opcji wyświetlania informacji o rozmiarze pobranych danych, drugie wpisanie wyłącza tę funkcję. Oto przykład użycia tej opcji:

```
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp>
```

Teraz właściwie wszystko jest przygotowane do rozpoczęcia wysyłania lub pobierania danych. Aby wysłać pliki na serwer, wystarczy użyć polecenia `put`, po którym należy podać nazwę pliku, jaki ma zostać tam skopiowany, zgodnie z poniższym przykładem:

```
ftp> put JDSC-0.5.9a.zip
local: JDSC-0.5.9a.zip remote: JDSC-0.5.9a.zip
200 PORT command successful
150 Connecting to port 39972
#####
#####
#####
#####
#####
#####
226-File successfully transferred
226 0.202 seconds (measured here), 2.36 Mbytes per second
499668 bytes sent in 0.19 secs (2526.9 kb/s)
ftp>
```

Pobieranie plików z serwera FTP jest równie łatwe; należy użyć polecenia `get` i podać nazwę pliku, który ma zostać pobrany z serwera, jak w poniższym przykładzie:

```
ftp> get JDSC-0.5.9a.zip
local: JDSC-0.5.9a.zip remote: JDSC-0.5.9a.zip
200 PORT command successful
150-Connecting to port 39972
150 488.0 kbytes to download
#####
#####
#####
#####
#####
#####
226-File successfully transferred
226 0.105 seconds (measured here), 4.53 Mbytes per second
499668 bytes received in 0.12 secs (4217.0 kb/s)
ftp>
```



Podczas wysyłania plików na serwer FTP lub pobierania plików z niego można nadać im inną nazwę (zarówno tym, które mają znaleźć się na serwerze, jak i pobieranym na dysk lokalny komputera), w tym celu należy po nazwie pliku oryginalnego podać nową nazwę.

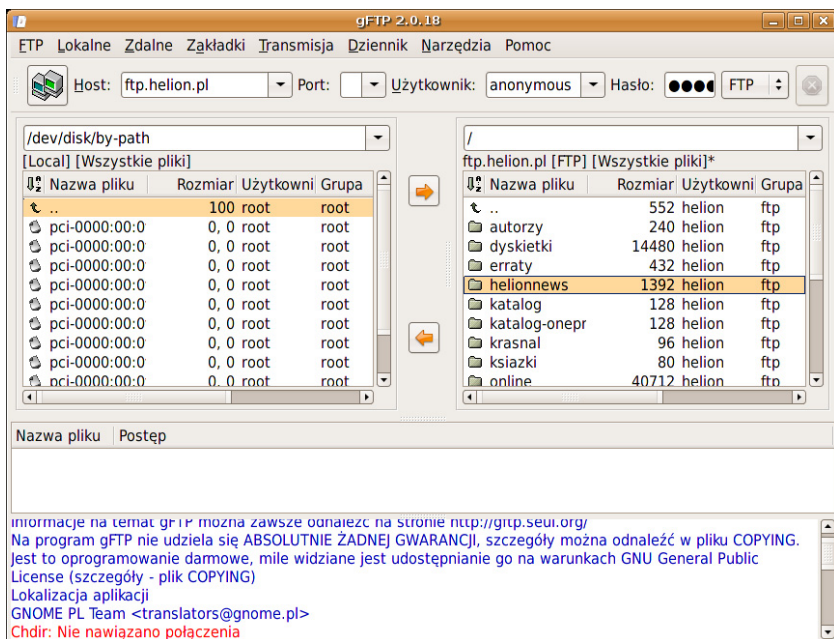
Po udanym przesyłaniu danych można zamknąć połączenie z serwerem FTP i zakończyć pracę programu `ftp`, wpisując polecenie `quit`, jak w poniższym przykładzie:

```
ftp> quit
```

Prościej już chyba nie można. Przyjemnym aspektem pracy z programem linii poleceń ftp jest fakt, że to narzędzie jest dostępne w większości systemów (nawet w Microsoft Windows), w dodatku używane polecenia są wszędzie takie same. Jak mawiają: „Raz się nauczyć i ciągle używać”.

Łączenie się z serwerami FTP za pomocą narzędzi graficznych

Jeżeli użytkownik do łączenia się z serwerami FTP preferuje narzędzia z interfejsem graficznym, może skorzystać z dostępnego w repozytoriach Ubuntu programu gtfp-gtk, czyli klienta GNOME FTP używającego graficznej biblioteki GTK. Aplikacja nie jest instalowana standardowo w Ubuntu, ale za pomocą menedżera pakietów Synaptic łatwo ją dodać do systemu. Po jej zainstalowaniu należy wybrać z menu kolejno *Aplikacje/Internet/gFTP*. Program umożliwia łatwe przysyłanie plików w obie strony, tworzenie zakładki dla wybranych serwerów, zapisywanie nazw hostów, użytkowników, haseł itd., tak jak ma to miejsce w przeglądarkach internetowych. Na rysunku 16.5 przedstawiam okno programu gtfp-gtk po nawiązaniu połączenia z serwerem FTP.



Rysunek 16.5. Graficzny klient FTP — gtfp-gtk

Dostęp do zasobów systemów Windows

Ubuntu potrafi łatwo uzyskać dostęp do współdzielonych folderów systemów Microsoft Windows, które zwykle nazywane są udziałami, za pomocą jednego z najpopularniejszych i najbardziej użytecznych pakietów wolnego oprogramowania, jakie do tej pory napisano, a mianowicie Samby (www.samba.org). Samba została napisana przez Andrew Tridgella, który zresztą za ten wyczyn zebrał wiele nagród oraz wyrazów uznania. Tridgell

pracuje nad Sambą do dziś, choć obecnie swój wkład do tej aplikacji wniosły też tysiące innych programistów i jest ona już efektem zbiorowego wysiłku. W rozdziale 32., „Konfiguracja serwera Samba”, zamieszczam więcej informacji na temat samego pakietu.

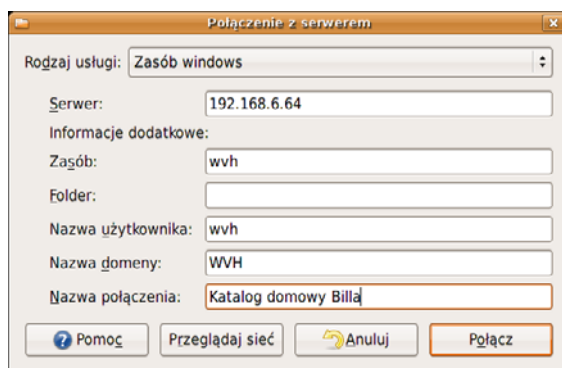
Samba swoją nazwę wzięła od protokołu SMB (ang. *Server Message Protocol*), który jest wykorzystywany w systemach Microsoft Windows do sieciowego współdzielenia plików. Tridgell, tworząc Sambę, musiał się uciec do technik inżynierii wstecznej (ponieważ — co za niespodzianka — Microsoft nie udostępnił publicznie specyfikacji SMB), co pozwoliło mu zapoznać się z technologią działania tego protokołu. Obecnie specyfikacja SMB jest już otwarta (w dużej mierze dzięki wysiłkom samego Tridgella oraz Sambie) i stanowi część specyfikacji CIFS (ang. *Common Internet File System*).

Jak wspominałem we wstępie do tej części rozdziału, Sambę można również wykorzystać do udostępniania plików w systemach linuksowych, które dzięki temu będą wyglądały jak udziały Microsoft Windows, a użytkownicy tych systemów będą mogli z nich korzystać. Szczegółową konfigurację serwera Samby opisałem w rozdziale 32. Poniżej skupię się na omówieniu metod uzyskiwania dostępu do udziałów Windows z Ubuntu oraz przesyłania plików z lub do systemów Windows.

W menu GNOME *Miejsca* znajduje się polecenie *Połącz z serwerem*, za którego pomocą można utworzyć na pulpicie ikonę skrótu do zasobów Windows, które z kolei będzie można łatwo eksplorować przy użyciu menedżera plików Nautilus. Po wybraniu kolejno z menu *Miejsca/Połącz z serwerem* zostanie wyświetlone okno dialogowe widoczne na rysunku 16.1. Aby określić typ serwera, z którym ma zostać nawiązane połączenie, należy rozwinąć znajdującą się u góry listę i wybrać z niej opcję *Zasób windows* (patrz rysunek 16.6).

Rysunek 16.6.

Identyfikowanie zasobów Windows, z którymi ma zostać nawiązane połączenie



Trzeba podać przynajmniej nazwę lub adres IP serwera Windows udostępniającego zasoby, z którym użytkownik chce nawiązać połączenie. Można też wprowadzić własną nazwę dla wyświetlanej na pulpicie ikony, co może być praktyczne, ponieważ proponowana domyślnie jest zwykle nazwą danego serwera, co w przypadku korzystania z wielu zasobów na jednym serwerze może być mylące. Można też podać nazwę użytkownika, który będzie logował się na serwerze oraz nazwę domeny lub grupy roboczej Windows, a także nazwę udziału, ale jest to opcjonalne. Jeżeli będą potrzebne dodatkowe informacje, użytkownik zostanie poproszony o ich wprowadzenie w kolejnym oknie dialogowym. Na rysunku 16.6 widać, że preferuję wprowadzanie jak największej ilości informacji, ale

zależy to tylko od preferencji użytkownika. Po kliknięciu przycisku *Połącz* na pulpicie zostanie utworzona odpowiednia ikona.



Kiedy przyjrzymy się bliżej ikonom tworzącym podczas konfigurowania połączenia z zasobami Windows, da się zauważyć, że znajduje się na nich symbol połączenia sieciowego, a także czerwony znacznik z umieszczonymi wewnątrz literami SMB. Dzięki temu osoby przechowujące dużo ikon na pulpicie mogą łatwo zidentyfikować odpowiednie protokoły używane do różnych połączeń. Jest to szczególnie przydatne podczas korzystania również z mechanizmów łączenia się z serwerami FTP, co omówiłem wcześniej w tym rozdziale.

Samo utworzenie na pulpicie ikony symbolizującej połączenie z zasobami Windows nie oznacza jeszcze jego nawiązania. Aby to zrobić, należy kliknąć dwukrotnie odpowiednią ikonę. Zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 16.7, w którym należy wprowadzić brakujące informacje.

Rysunek 16.7.

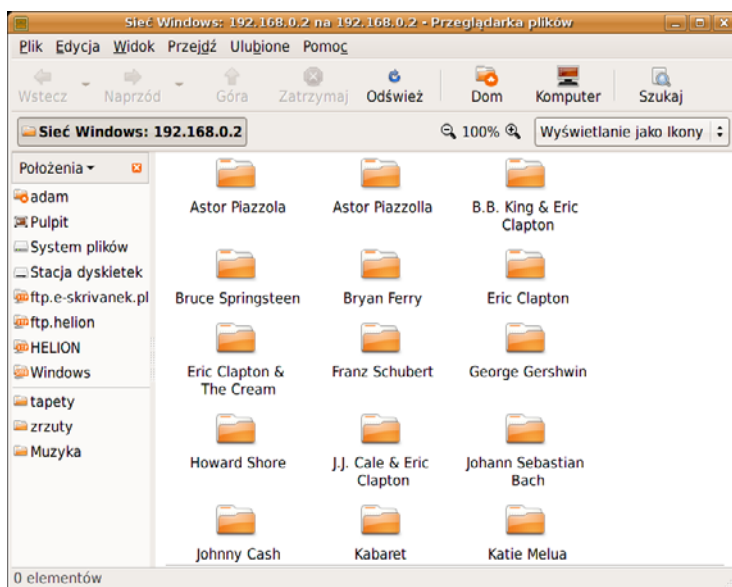
Kompletowanie informacji potrzebnych do uzyskania dostępu do zasobów Windows



Po wprowadzeniu brakujących informacji, które są niezbędne do nawiązania połączenia z zasobami Windows, zostanie wyświetlone okno menadżera plików Nautilus wraz z zawartością domyślnego folderu przypisanego logującemu się użytkownikowi (rysunek 16.8).

Rysunek 16.8.

Nawiązane połączenie z zasobem Windows widoczne w oknie Nautilus



Kiedy już w Nautilusie zostanie wyświetlona zawartość danego serwera FTP, pliki można kopiować i przenosić, korzystając ze standardowych metod stosowanych w środowisku GNOME, które zostały opisane w rozdziale 5.

Kiedy już wszystkie potrzebne operacje zostaną wykonane, należy zamknąć okno Nautilusa. Aby usunąć ikonę danego połączenia z pulpitu, trzeba ją zaznaczyć, następnie za pomocą prawego przycisku myszy wyświetlić menu kontekstowe i wybrać z niego opcję *Odmontuj wolumin*. Jeżeli jednak użytkownik planuje jeszcze nawiązywać połączenia z danym zasobem Windows, wówczas należy pozostawić ikonę na pulpicie; następnym razem będzie można zalogować się na serwerze, choć trzeba będzie ponownie przejść przez proces uwierzytelniania.

Dostęp do katalogów NFS z systemów Linux

Na długo przed pojawieniem się Microsoft Windows i Apple Macintosh pliki systemów uniksowych były udostępniane w sieciach przy użyciu sieciowego systemu plików NFS, zaprojektowanego we wczesnych latach 80. przez firmę Sun Microsystems. Sun był na tyle mądry, że udostępnił publicznie specyfikację tej technologii, dzięki czemu znalazła ona zastosowanie w każdym tworzonym systemie Unix, a także systemach na nim opartych. Choć dzisiaj większość dostawców znanych z tamtych czasów nie istnieje, a nawet przyszłość samego Suna jest niepewna, to NFS pozostał domyślnym sieciowym systemem plików i mechanizmem ich udostępniania we wszystkich systemach uniksowych czy uniksopodobnych, takich jak Mac OS X.

Niestety, obecnie środowisko GNOME nie oferuje żadnego graficznego narzędzia do montowania NFS, dlatego należy w tym celu skorzystać z narzędzi dostępnych w linii poleceń. Być może czytelnik pomyślał nawet, że uda mu się to osiągnąć bez czytania rozdziału 6. Na szczęście, montowanie systemu plików z poziomu linii poleceń jest bardzo proste. Sam mechanizm służący do eksportowania lokalnego systemu plików przez NFS jest również prosty, więcej informacji na ten temat zamieszczam w rozdziale 31., „Konfiguracja serwera NFS”.

System NFS jest identyfikowany na podstawie nazwy lub adresu IP komputera, na którym się znajduje, następnie umieszcza się dwukropek, a po nim nazwę katalogu, który jest eksportowany. I tak np. zapis `192.168.6.64:/home/wvh` oznacza katalog `/home/wvh` znajdujący się na komputerze o adresie `192.168.6.64`. System plików NFS jest montowany w katalogu lokalnego systemu Ubuntu, tak jak każdy inny system plików (więcej informacji na ten temat zamieszczam w rozdziale 4.).

Aby zamontować system NFS, najpierw należy utworzyć dla niego punkt montowania. Zwykle tworzę wszelkie punkty montowania w katalogu `/mnt`, a zatem mogę je łatwo zidentyfikować jako tymczasowo zamontowane urządzenia. Oto przykład tworzenia punktu montowania o nazwie `/mnt/nfs`:

```
$ sudo mkdir /mnt/nfs
Password:
```

Po podaniu hasła punkt montowania zostanie utworzony i teraz można zamontować w nim katalog NFS, korzystając ze standardowego narzędzia `mount`; a tak trzeba to zrobić:


```
$ sudo mount -t nfs 192.168.6.64:/home/wvh /mnt/nfs
```

Teraz można już sprawdzić, czy katalog został zamontowany pomyślnie; w tym celu wystarczy wyświetlić jego zawartość lub użyć polecenia `df`, tak jak w poniższym przykładzie:

```
$ df
Filesystem            1K-blocks    Used   Available   Use%    Mounted on
/dev/hda1             77299808 40359464 33013660    56%     /
varrun                257968      136    257832      1%     /var/run
varlock               257968        4    257964      1%     /var/lock
udev                 257968       96    257872      1%     /dev
devshm               257968        0    257968      0%     /dev/shm
192.168.6.64:/home/wvh
160010496 151044928 8965568    95%     /mnt/nfs
```

Zamontowany katalog jest teraz dostępny, a użytkownik może tworzyć w nim nowe pliki. Ponieważ NFS jest systemem linuksowym (czy uniksowym), użytkownik musi mieć odpowiednie uprawnienia, aby odczytywać i zapisywać w nim pliki. W większości środowisk korzystających z systemu plików NFS tożsamość użytkownika jest synchronizowana na komputerze lokalnym i zdalnym. Jeżeli tak się nie stanie, wówczas użytkownik może samodzielnie — wspomagając się poleceniem `sudo` — skopiować pliki i katalogi do systemu NFS, jak w poniższym przykładzie:

```
$ sudo cp resume.xml /mnt/nfs
```

Synchronizowanie tożsamości użytkowników w wielu systemach oraz ogólne zasady identyfikowania użytkowników w systemach Linux szerzej omówiłem w rozdziale 21., „Zarządzanie użytkownikami, grupami i zaawansowane uprawnienia”.

Aby odmontować dany system NFS, należy użyć standardowego polecenia systemu Linux `umount`; oto przykład:

```
$ sudo umount /mnt/nfs
```

Warto zwrócić uwagę, że to polecenie to nie *umount*, ale *umount*; po co wpisywać literę *n*, kiedy nie jest to potrzebne?

Udostępnianie plików w sieciach P2P

Załóżmy, że czytelnik stał się posiadaczem archiwów tar lub plików ISO zawierających najlepsze programy *open source*, jakie do tej pory napisano, i każdy, kto o nich usłyszy, chciałby mieć kopię. Co więcej, czytelnik chciałby udostępnić je wszystkim zainteresowanym. Klasycznym rozwiązaniem tego problemu jest umieszczenie plików na serwerze FTP czy — częściej — serwerze WWW, a następnie poinformowanie o tym fakcie całego świata. Niezależnie od tego, jak szybkimi serwerami dysponuje użytkownik, niezależnie od ich liczby oraz przepustowości łącza internetowego, łatwo napotkać problem, że zbyt wielu użytkowników rozpocznie pobieranie plików, co pochłonie wszystkie zasoby sprzętowe i techniczne. Kolejni użytkownicy, którzy będą próbowali pobrać pliki, nie będą mogli dostać się na serwer, a ci którym się to udało zauważą znaczne zwolnienie prędkości przesyłu, aż w końcu na skutek przeciążenia padną serwery. Takie są — niestety — skutki umieszczenia informacji o danej stronie w serwisie SlashDot (www slashdot.com).

Przyczyną opisanego problemu jest fakt, że dane pliki znajdują się tylko w jednym miejscu, do którego zgłaszają się wszyscy chcący je pobrać. I właśnie dlatego na tej jednej stronie, która staje się obiektem zainteresowania wszystkich, tak szybko tworzy się korek. Nawet jeżeli użytkownik dysponuje wieloma replikami serwera oraz używa najlepszego oprogramowania do zarządzania ruchem w sieci, to i tak te zasoby są ograniczone, podczas gdy popyt na dane pliki może być nieograniczony.

Imponującym rozwiązaniem tego problemu jest wymiana plików za pomocą sieci P2P. Jak wspominałem we wprowadzeniu do niniejszego rozdziału, różnica pomiędzy tradycyjnym modelem przesyłania danych w sieciach (*point-to-point*) a P2P polega na tym, że w tym drugim przypadku transfer plików jest efektem zbiorowego wysiłku i współpracy, w trakcie której wymieniający między sobą pliki użytkownicy nie muszą mieć wiedzy na temat tego, kto znajduje się po drugiej stronie. Oczywiście, ktoś musi wiedzieć, na których komputerach znajdują się dane pliki, ponieważ w przeciwnym razie nie byłoby wiadomo, skąd pobierać dane. Na szczęście, istnieje doskonałe rozwiązanie tego problemu. Niestety, stało się ono przyczyną upadku pioniera technologii P2P, Napstera, który korzystał z centralnej bazy zawierającej informacje na temat tego, co oferuje dany komputer innym użytkownikom. Niektórzy idioci zajmujący się prawem stwierdzili, że to oznacza, iż Napster akceptuje fakt, że pomiędzy wymienianymi w sieci plikami znajdują się takie, za które użytkownicy powinni płacić. To zmusiło serwis do zamknięcia centralnego serwera, co z czasem spowodowało zatrzymanie ruchu w sieci Napster.

Na szczęście, przeprowadzenie analizy „legalnych” przyczyn upadku Napstera wydatnie pomogło drugiej fali serwisów P2P zaproponować nowe rozwiązania. Podstawową zmianą było uruchomienie dynamicznego monitorowania i tworzenia zapytań w celu ustalenia, na którym z komputerów dostępnych w danej sieci znajdują się poszukiwane przez użytkownika pliki. Kluczem do sukcesu odniesionego przez serwisy P2P drugiej generacji jest — i będzie — anonimowość komputerów udostępniających i pobierających dane z sieci. Bez skierowania do każdego komputera z osobną zapytania o zawartość udostępnianych zasobów nie sposób uzyskać kompletnego obrazu współczesnych, anonimowych sieci P2P. I bardzo dobrze.

Obecnie istnieje wiele popularnych protokołów P2P, sieci udostępniających pliki oraz programów do korzystania z nich. Oto kilka najbardziej znanych:

- ♦ **BitTorrent** — www.bittorrent.com,
- ♦ **eDonkey2000** — www.edonkey2000.com,
- ♦ **FastTrack** (właściwie **Kazaa**) — www.kazaa.com,
- ♦ **Freenet** — <http://freenet.sourceforge.net>,
- ♦ **Gnutella** — www.gnutella.com.

W następnej części rozdziału wyjaśnię, jak stać się częścią rewolucji, instalując i wykorzystując gtk-gnutella, popularnego linuksowego klienta sieci Gnutella; a także gnome-bt-download, który z kolei jest kompletną aplikacją służącą do pobierania plików BitTorrent. Podobnie jak w całym rozdziale, omawiając te programy, główny nacisk położę na pobieranie plików, a nie na ich udostępnianie. Oczywiście, dzieje się to automatycznie, ale ten proces nie zostanie tu omówiony.

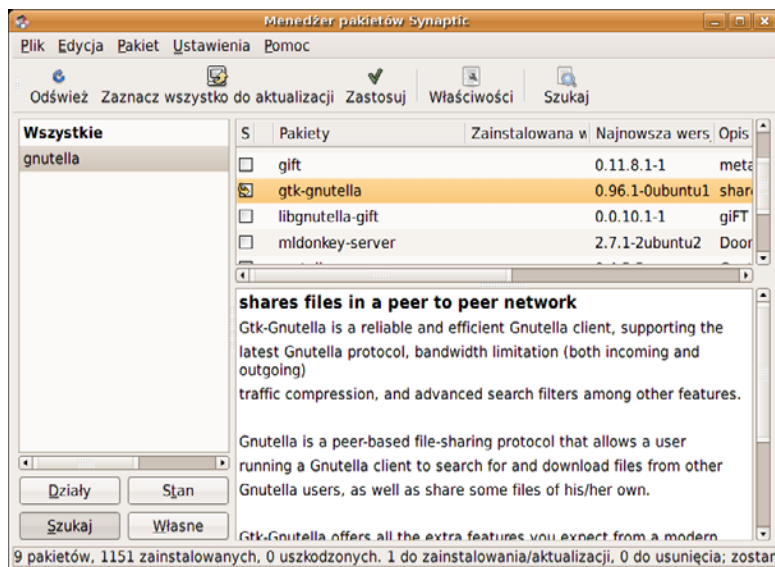
Instalowanie i korzystanie z programu gtk-gnutella

Sieć Gnutella (nie mylić z Nutellą, producentem pysznych czekoladek) jest rozległą i niezmiernie popularną siecią P2P. Wielu z nas nie zdaje sobie sprawy, że powszechnie używane programy, takie jak LimeWire czy BearShare, to w rzeczywistości programy klienckie tej właśnie sieci. Jedną z najfajniejszych opcji dostępnych w programach klienckich przeznaczonych dla sieci Gnutella jest możliwość wyszukiwania, którą omówię w dalszej części rozdziału. Jak wspominałem wcześniej, wyszukiwanie nie oznacza kontaktowania się z centralnym serwerem zbierającym informacje. W zamian oferowany jest bardzo wygodny mechanizm do zapytań typu „przyjaciel do przyjaciela”. Wysłane zapytanie trafia do najbliższego węzła sieci Gnutella, który następnie przesyła je do kolejnego znanego sobie węzła itd., do czasu aż większa część sieci nie będzie rozbrzmiewała zapytaniem wysłanym przez użytkownika.

Instalowanie programu gtk-gnutella

Ponieważ nie wszyscy chcą lub potrzebują korzystać z sieci P2P, dlatego też gtk-gnutella nie jest instalowana standardowo w systemie. Aby to zrobić, należy uruchomić menedżer pakietów Synaptic (menu *Aplikacje/System*), wywołać okno wyszukiwania, wprowadzić do niego ciąg `gnutella`, a następnie kliknąć przycisk *Szukaj*. Następnie w wynikach wyszukiwania należy odnaleźć odpowiednią pozycję, zaznaczyć ją, a z menu kontekstowego, które zostanie wyświetlone, wybrać opcję *Zaznacz do instalacji* (patrz rysunek 16.9).

Rysunek 16.9.
Instalowanie programu gtk-gnutella za pomocą menedżera pakietów Synaptic



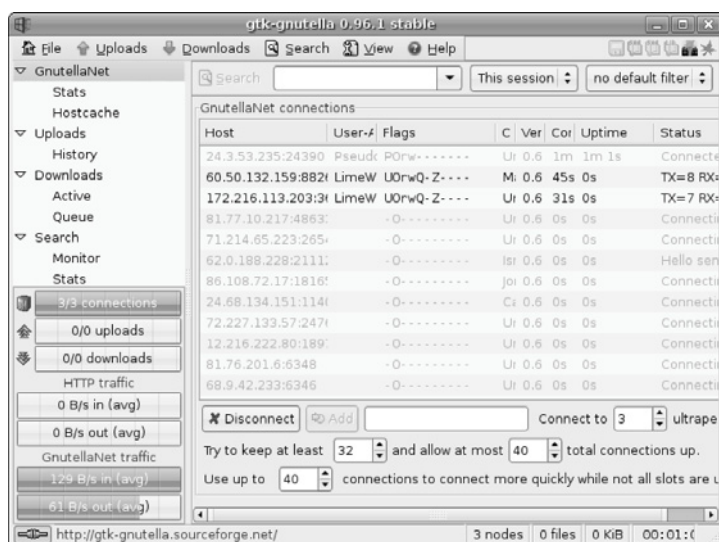
W zależności od tego, jakie oprogramowanie użytkownik zainstalował uprzednio w systemie, może zostać wyświetlony komunikat o dodatkowych pakietach, które również należy dodać do systemu. Jeżeli zobaczymy ów komunikat, należy zaakceptować proponowane zmiany.

Kiedy odpowiednie pakiety zostaną zaznaczone, należy kliknąć przycisk *Zastosuj* znajdujący się na pasku narzędzi Synaptica. Po zakończeniu instalacji użytkownik może włączyć się do sieci Gnutella! W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, można znaleźć dodatkowe informacje na temat korzystania z programu Synaptic.

Korzystanie z programu gtk-gnutella

Po zainstalowaniu programu, co opisałem w poprzedniej części, można go uruchomić, wybierając z menu kolejno *Aplikacje/Internet/Gtk-Gnutella*, po chwili zostanie wyświetlone okno programu widoczne na rysunku 16.10.

Rysunek 16.10.
Okno startowe
programu *gtk-gnutella*



Podczas pierwszego uruchomienia programu zostanie utworzony katalog *gtk-gnutella-downloads* w katalogu domowym użytkownika; w nim będą zapisywane wszystkie pobrane pliki. W katalogu tym znajdują się trzy podkatalogi: *complete* — w którym przechowywane są w pełni pobrane pliki, *incomplete* — pliki, które są w trakcie pobierania, oraz *corrupt* — pobrane pliki, których suma kontrolna nie zgadza się z oficjalnymi danymi. Pliki znajdujące się w tym ostatnim są zapewne uszkodzone, ale na wszelki wypadek można je sprawdzić.

Oprócz standardowego paska menu, w programie można znaleźć jeszcze inne interesujące elementy widoczne na rysunku 16.10, oto one:

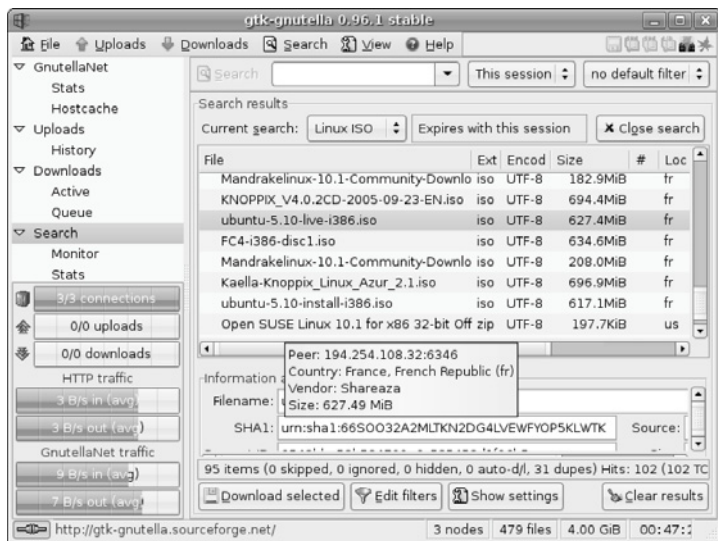
- ♦ **obszar graficznego podsumowania** — znajduje się w dolnym lewym rogu; można tam znaleźć informacje o ilości podłączonych klientów, liczbie trwających pobrań i wysyłanych plików oraz aktualnego ruchu w sieci,
- ♦ **panel ze szczegółowymi informacjami** — znajduje się powyżej poprzedniego obszaru; znajdują się na nim rozwijane elementy, z których można dowiedzieć się o statusie sieci Gnutella, aktywnym i zakończonym pobieraniu oraz wysyłaniu plików, a także statusie wyszukiwania,

- ♦ **pasek wyszukiwania** — znajduje się w środkowej części okna, tuż pod menu; umożliwia inicjowanie wyszukiwania określonych plików, których nazwy odpowiadają wpisanym ciągom znaków,
- ♦ **okno statusu połączenia** — znajduje się w środku ekranu; wyświetla informacje o bieżącym połączeniu, rezultatach wyszukiwania, stanie aktywnych procesów pobierania plików oraz tych, które są zakolejkowane; typ wyświetlanych tu informacji zależy od zawartości określonej w oknie kontrolnym,
- ♦ **okno kontroli** — umożliwia kontrolowanie ilości połączeń obsługiwanych aktualnie przez komputer.

Interesującą możliwością programu jest opcja wyszukiwania, wyświetlania dopasowanych trafień oraz pobierania plików. Na początek wyszukiwanie. Aby zademonstrować, jak wiele legalnych plików można uzyskać dzięki Gnutelli, należy umieścić kursor w polu wyszukiwania, wprowadzić wyrażenie `Linux ISO` i wcisnąć klawisz `Enter`. Rozpocznie się wyszukiwanie, a wszystkie pliki odpowiadające wyszukiwanemu ciągowi będą wyświetlane w oknie statusu połączenia. Kiedy już wyniki wyszukiwania zostaną wyświetlone, można zobaczyć informacje o każdym z nich, umieszczając kursor nad wybraną pozycją (patrz rysunek 16.11).

Rysunek 16.11.

Dodatkowe informacje o wyszukanych plikach

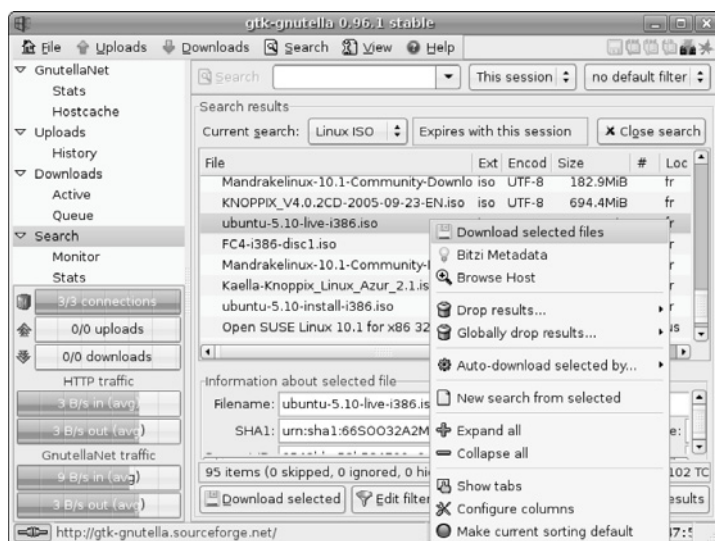


Kiedy użytkownik wybierze plik, który chce pobrać, musi go zaznaczyć, a następnie za pomocą prawego przycisku myszy wywołać menu kontekstowe, z którego należy wybrać opcję *Download selected files* (patrz rysunek 16.12).

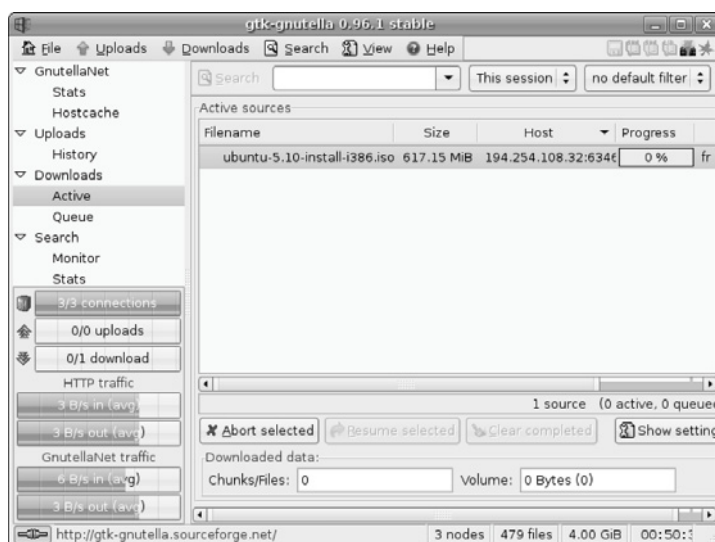
Kiedy pobieranie zostanie rozpoczęte, można śledzić jego status, wybierając w panelu ze szczegółowymi informacjami opcję *Downloads/Active*. Jeżeli nie będzie tu informacji o danym pobieraniu pliku, oznacza to, że oczekuje ono w kolejce, której zawartość można sprawdzić, wybierając w tym samym panelu opcję *Downloads/Queue*. Na rysunku 16.13 przedstawiam właśnie rozpoczęty proces pobierania pliku.

Rysunek 16.12.

Inicjowanie pobierania plików w programie *gtk-gnutella*

**Rysunek 16.13.**

Pobieranie pliku w programie *gtk-gnutella*



Jeżeli pojawią się problemy z poprawnym pobieraniem plików przez program, należy się upewnić, czy standardowe porty używane przez *gtk-gnutellę*, czyli 5634, 6346, 6347, 6348, 6349 i 6355 nie są zablokowane. Jeżeli użytkownik korzysta z bramy lub routera i zauważy problemy z szybkością połączenia, wówczas może trzeba będzie przekierować te porty wprost na host, na którym uruchamiany jest klient Gnutelli. Więcej informacji na ten temat można znaleźć na stronie www.portforward.com/routers.htm.

W sieci Gnutella wiele różnych komputerów oferuje mnóstwo różnorodnych plików, ale może się okazać, że ktoś inny wcześniej zaczął je pobierać i chwilowo wyczerpany jest limit połączeń ustalony przez ich użytkownika, dlatego też trzeba będzie poczekać jakiś czas, zanim pobieranie pliku się rozpocznie. Moją metodą na skrócenie czasu oczekiwania jest wskazanie kilku źródeł do pobierania tego samego pliku, dzięki czemu można ocenić,

które rokuje najlepsze rezultaty. Kiedy już właściwe pobieranie zostanie rozpoczęte, należy anulować pozostałe, tak aby nie marnować przepustowości sieci Gnutella.

Korzystanie z BitTorrenta w Ubuntu

BitTorrent zaprojektowany przez Brama Cohena obecny jest w sieci od kilku lat i w tym czasie zdobył sobie ogromną popularność jako protokół i narzędzie do przesyłania plików w sieciach P2P. Oryginalny BitTorrent Brama to oprogramowanie wydane na licencji GPL i napisane w Pythonie. Jednak sam wciąż w głównej mierze używam wersji programu przeznaczonej dla linii poleceń (*btdownloadheadless.py*), ponieważ ona „po prostu działa”. Czytelnicy nie powinni się jednak obawiać, że i w tym przypadku będę ich nakłaniał do pójścia w moje ślady, chociaż program ten jest domyślnie instalowany w systemie Ubuntu. Podobnie rzecz ma się z prostym narzędziem o nazwie *gnome-btdownload*, które omówię w dalszej części rozdziału.

Cała tajemnica korzystania z sieci P2P BitTorrent kryje się w plikach torrent zawierających informacje o plikach, które mają być dystrybuowane w sieci (to kolejna zaleta tego protokołu: pliki mogą być przesyłane w częściach), oraz w komputerach (tzw. trackerach), które przysyłają informacje o innych komputerach pobierających pliki. Odmienne niż w przypadku starego Napstera, bittorrentowy tracker nie wie nic na temat zawartości torrenta, ma tylko informacje, którzy użytkownicy uczestniczą w procesie pobierania, wysyłania i udostępniania danego pliku torrent. Komputer „tracker” organizuje komunikację pomiędzy użytkownikami zainteresowanymi określonym torrentem; kiedy już użytkownik nawiąże komunikację z drugim, tracker przestaje być potrzebny. Jak można się spodziewać, potrzebny jest przynajmniej jeden komputer posiadający kompletny plik, który ma być udostępniany; takiego użytkownika określa się nazwą „ziarno” (ang. *seed*).

Kolejnym interesującym usprawnieniem wprowadzonym w tej technologii jest to, że pobieranie i wysyłanie części pliku następuje w losowej kolejności. Oznacza to, że nawet gdy wielu użytkowników pobiera jednocześnie ten sam plik, szansa na to, że będą pobierać ten sam fragment, jest bardzo niewielka. To sprawia, że protokół jest odporny i radzi sobie nawet gdy od sieci zostaną odłączone tracker i ziarno. Sprawia to, że istnieje spora szansa na to, iż na pozostałych komputerach biorących udział w wymianie znajdują się wszystkie fragmenty pliku torrent, a wtedy zakończenie pobierania pozostaje tylko kwestią czasu. Taka sytuacja nosi nazwę **dystrybuowania kopii**.

BitTorrent to znakomite oprogramowanie i doskonały mechanizm dystrybucyjny, jest to też znakomity przykład, jak sieci P2P mogą pomóc zagwarantować wolną wymianę dowolnych danych w sieci. Jest też bezpłatny. Jeżeli czytelnicy chcieliby wspomóc jego istnienie i dalszy rozwój, mile widziane są wszelkie dotacje. Tak właśnie zrobiłem. Swoją wkład w przyszłość swobody komunikacji można wnieść za pośrednictwem PayPal na stronie <http://bittorrent.com/donate.myt>.

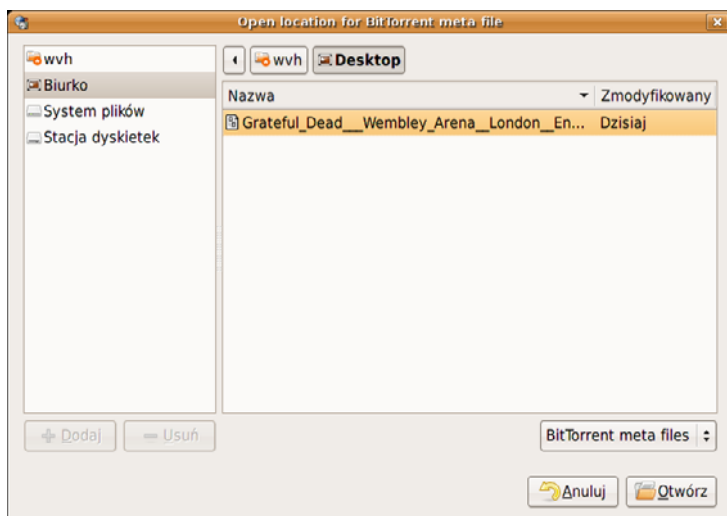
Korzystanie z programu *gnome-btdownload*

Ponieważ program *gnome-btdownload* oraz Python są instalowane wraz z podstawowym systemem Ubuntu, aby je uruchomić, nie trzeba żadnych specjalnych czynności konfiguracyjnych. Niestety, nie ma żadnego wpisu w menu, dlatego program należy uruchomić

z terminala GNOME lub okna xterm. Po włączeniu `gnome-btdownload` zostanie wyświetlony ekran startowy widoczny na rysunku 16.14.

Rysunek 16.14.

Ekran startowy programu `gnome-btdownload`

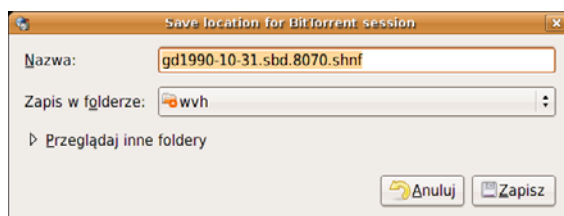


Jeżeli okaże się, że użytkownik często korzysta z programu `gnome-btdownload`, wówczas warto dodać go do menu *Aplikacje*, zgodnie z informacjami zamieszczonymi w rozdziale 5.

Za pomocą okna widocznego na rysunku 16.14 należy przejść do miejsca, w którym zapisano plik torrent, a następnie zaznaczyć go i kliknąć przycisk *Otwórz*, zostanie wówczas wyświetlone kolejne okno dialogowe widoczne na rysunku 16.15.

Rysunek 16.15.

Określanie miejsca zapisania pliku

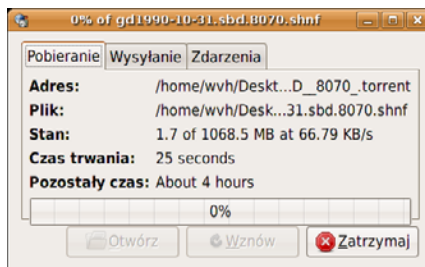


Plik można zapisać w bieżącym katalogu (czyli tym, który był otwarty w momencie uruchamiania programu `gnome-btdownload`) lub wskazać inną lokalizację, korzystając z opcji *Przeglądaj inne foldery*. Po wybraniu odpowiedniego miejsca należy kliknąć przycisk *Zapisz*, co spowoduje rozpoczęcie pobierania pliku (patrz rysunek 16.16).

W oknie dialogowym widocznym na rysunku 16.16 użytkownik może śledzić postęp pobierania pliku (zakładka *Pobieranie*); określić, ilu innym użytkownikom program ma udostępnić plik (zakładka *Wydarzenia*) oraz monitorować działania programu (zakładka *Zdarzenia*). Najbardziej interesująca informacja wyświetlana jest na pierwszej zakładce w polu *Stan*, jest to mianowicie ilość danych, która już została pobrana z danego pliku. Warto też sprawdzać, jak długo potrwa pobieranie pliku (pole *Pozostały czas*), ta wartość będzie się zmieniać wraz z upływem czasu i ilością już pobranych danych, ilością dostępnych plików w sieci oraz przepustowością łącza.

Rysunek 16.16.

Pobieranie pliku
za pomocą programu
gnome-btdownload



Jeżeli pojawią się problemy z poprawnym pobieraniem przez program plików, należy się upewnić, czy używana zaporą sieciowa nie blokuje portów od 6881 do 6999. Jeżeli użytkownik korzysta z bramy lub routera i zauważy problemy z wydajnością połączenia, może trzeba będzie przekierować te porty wprost na host, na którym uruchamiany jest klient BitTorrenta. Więcej informacji na ten temat można znaleźć na stronie www.portforward.com/routers.htm oraz na stronie Michaela Ingrama www.slyck.com/news.php?story=493.

Udanego pobierania! Jeżeli z czasem okaże się, że minimalistyczny interfejs programu *gnome-btdownload* nie zaspokaja wszystkich potrzeb, można poszukać w repozytoriach Ubuntu innych klientów sieci BitTorrent. Niektóre z nich to dobrze znane, np.: *azureus*, oparty na Javie program zawierający również tracker, *bittornado-gui*, szybki, wydajny i mający niewielkie wymagania klient, czy *qtorrent* — graficzny klient korzystający z biblioteki QT. Fani KDE mogą wypróbować interesujący *ktorrent*, program przeznaczony właśnie dla środowiska KDE.

Wyszukiwanie torrentów

Zapewne w tym miejscu czytelnik myśli: „Wygląda na to, że bittorrent to najlepszy pomysł od czasów wynalezienia młotka, ale skąd wziąć te torrenty?”. Ponieważ BitTorrent nie oferuje wyszukiwania plików znanego z sieci Gnutella, użytkownik musi sam odnaleźć potrzebne pliki. Jak zapewne czytelnicy zauważyli, w poprzednim przykładzie użyto pliku torrent z nagraniem sesji live zespołu Grateful Dead, który znany jest z tego, że udostępnia nieodpłatnie swoje nagrania koncertowe. Szybkie wyszukiwanie w sieci haseł „Grateful Dead” oraz „torrent” przyniesie mnóstwo rezultatów, ale trudno określić to mianem najbardziej efektywnego mechanizmu wyszukiwania koncertów muzycznych, w dodatku rozpowszechnianych bezpłatnie.

Na szczęście, w internecie można znaleźć wiele stron zaprojektowanych specjalnie do wyszukiwania torrentów, oto kilka z nich:

- ♦ **BtBot** — www.btbot.com,
- ♦ **isoHunt** — <http://isohunt.com>,
- ♦ **Torrent Search** — www.torrentsearch.com,
- ♦ **Torrent Spy** — www.torrentspy.com.

Ponieważ BitTorrent okazał się bardzo efektywnym mechanizmem P2P, wiele dystrybucji Linuksa (łącznie z Ubuntu) oferuje, oprócz standardowych plików ISO, również taką metodę pobierania plików, która znakomicie wspomaga powolne pobieranie ze stron

internetowych. Czytelnicy mogą wypróbować ten sposób i pobrać torrent do ulubionej dystrybucji Linuksa (mam nadzieję, że jest nią Ubuntu, ale można też wypróbować dystrybucje pochodne, takie jak Kubuntu, Xubuntu czy Ubuntu w wersji serwerowej; łączy do stron zawierających torrenty najnowszych wydań Ubuntu można znaleźć na stronie www.ubuntu.com/download).

Więcej informacji o BitTorrentcie

Ponieważ książka jest poświęcona Ubuntu, a nie aplikacji BitTorrent, mogłem przekazać jedynie garść podstawowych informacji o tej sieci P2P. Kiedy czytelnicy zaczną korzystać z tej sieci, będą pojawiały się dalsze pytania; oto kilka źródeł informacji, z których można dowiedzieć się nieco więcej o BitTorrentcie:

- ♦ **Oficjalny FAQ BitTorrenta** — www.bittorrent.com/faq.myt, lista najczęściej zadawanych pytań znajdująca się na stronie domowej BitTorrenta,
- ♦ **Przewodnik rozpowszechniania plików za pośrednictwem BitTorrenta** — www.bittorrent.com/guide.myt, przydatny dla wszystkich, którzy chcieliby tworzyć i dystrybuować swoje własne torrenty,
- ♦ **FAQ Briana Dessenta** — www.dessent.net/btfaq; alternatywna lista najczęściej zadawanych pytań dostarczająca wielu przydatnych informacji.

Podsumowanie

W środowiskach biznesowych i akademickich zwykle dostępne są serwery, na których przechowywane są dokumenty, katalogi z projektami, a czasem nawet działające oprogramowanie. Coraz częściej takie rozwiązania można też spotkać w domach, wówczas na takim serwerze zwykle przechowywane są rodzinne zdjęcia, muzyka czy dokumenty. W niniejszym rozdziale objaśniłem, jak uzyskać dostęp do zdalnych zasobów udostępnianych na serwerach działających pod kontrolą systemów Microsoft Windows, Mac OS X i Linux za pomocą oprogramowania, takiego jak Samba czy system plików NFS. Ostatnia część rozdziału poświęcona została sieciom P2P (obecnie najpopularniejszej metodzie wymiany plików w sieci) i możliwościom oferowanym w tym zakresie przez oprogramowanie dostępne w Ubuntu.

W rozdziale 17. omówię metody podłączania urządzeń elektronicznych, takich jak iPody i inne odtwarzacze muzyczne, aparaty cyfrowe i komputery kieszonkowe. Wymienione urządzenia dla wielu ludzi stały się obecnie przedmiotami codziennego użytku, ale nie będą specjalnie użyteczne, jeżeli nie można aktualizować przechowywanych na nich danych czy skopiować zdjęć z aparatu, aby zrobić w nim miejsce na następną serię ujęć z wakacji.

Rozdział 17.

Urządzenia elektroniczne w Ubuntu

W tym rozdziale:

- ◆ Rozpoznawanie urządzeń elektronicznych
- ◆ Podłączanie aparatu cyfrowego
- ◆ Podłączanie i synchronizacja urządzeń PDA
- ◆ Korzystanie z iPodów i innych odtwarzaczy MP3
- ◆ Stosowanie kart pamięci CompactFlash i SD

Linux to system operacyjny wbudowany w wiele współczesnych urządzeń. System wbudowany to określenie takiego systemu, który steruje i zarządza różnymi urządzeniami. Dwadzieścia lat temu takimi urządzeniami były elektrownie, fabryki, linie produkcyjne i statki kosmiczne. Dzisiaj te obiekty również znajdują się na liście, ale została ona poszerzona o samochód, bramę garażową, nagrywarke wideo, palmtop, odtwarzacz MP3 itd. Obecnie wiele urządzeń elektronicznych zawiera pełnowymiarowe systemy elektroniczne do ich konfigurowania i komunikowania.

Chociaż fakt, że Linux jako system operacyjny jest wybierany do wielu współczesnych urządzeń, może nie dotyczyć czytelnika osobiście, ale jest to dobry znak dla konsumentów i użytkowników tych systemów. Istnieje kilka przyczyn popularności Linuksa w tej dziedzinie. Fakt, że jest bezpłatny i nie wymaga wnoszenia opłat licencyjnych czy tantiem jest szczególnie ważny dla producentów planujących wytworzenie kilku milionów sztuk danego urządzenia. Po pierwsze, jeżeli producenci musieliby płacić jakiś rodzaj tantiem za system operacyjny, jakiego używają, ten koszt byłby po prostu przerzucony na konsumentów. Po drugie, elastyczność Linuksa sprawia, że łatwo go zaadaptować do olbrzymiej liczby urządzeń, co z kolei oznacza, że wsparcie dla sterowników urządzeń i ich peryferiów będzie się poszerzać. I wreszcie, otwartość Linuksa ułatwia dostęp do plików przechowywanych na większości współczesnych systemów operacyjnych i komunikowanie się z tymi systemami. Jest to dobra wiadomość np. dla firmy projektującej nowy odtwarzacz audio, który ma oferować wolność i wydajność Linuksa, ale jego użytkownicy muszą mieć też zapewnioną możliwość synchronizacji plików muzycznych ze znajdującymi się w systemie Mac OS X lub Microsoft Windows. Zapis i odczyt w popularnych w Windows

systemach plików? Już tam jest. Obsługa USB, połączeń bezprzewodowych czy portów szeregowych? Bezpłatna. Myślę, że rozumiecie, o co chodzi.

Fakt, że czytelnicy i ja używamy Linuksa na swoich komputerach, powoduje, że dysponujemy tą samą mocą i elastycznością. Jak będzie się można przekonać z lektury niniejszego rozdziału, Linux może komunikować się z niemal każdym urządzeniem. Użytkownicy innych systemów operacyjnych wymagają od producentów sprzętu elektronicznego dodatkowego oprogramowania, które umożliwi im korzystanie z tego sprzętu w ich komputerach. Użytkownicy Microsoft Windows nie mogliby docenić dobrze zaprojektowanych i lekkich iPodów, gdyby Apple nie dostarczył programu iTunes dla ich systemu operacyjnego. Użytkownicy urządzeń PDA korzystający z systemów Mac i Windows nie mogliby synchronizować swoich organizatorów, gdyby Apple i Palm nie dostarczyli odpowiedniego oprogramowania służącego do tego celu. Czasem to oprogramowanie jest bezpłatne, czasem trzeba na nie wydać pieniądze, dla użytkowników Linuksa jest to bez znaczenia.

Jak czytelnik będzie mógł się przekonać w niniejszym rozdziale, Ubuntu może komunikować się, synchronizować i wymieniać pliki oraz dane niemal ze wszystkim. Aby w świecie Linuksa podłączyć zupełnie nowe urządzenie, potrzebny jest tylko uzdolniony haker i już po niedługim czasie w internecie dostępny będzie pakiet, za którego pomocą to urządzenie będzie można połączyć z Ubuntu. Być może będzie to wymagało pewnych zmian konfiguracji, ale za to nic nie przebijie ceny takiego rozwiązania.

Konfigurowanie systemu do obsługi urządzeń elektronicznych

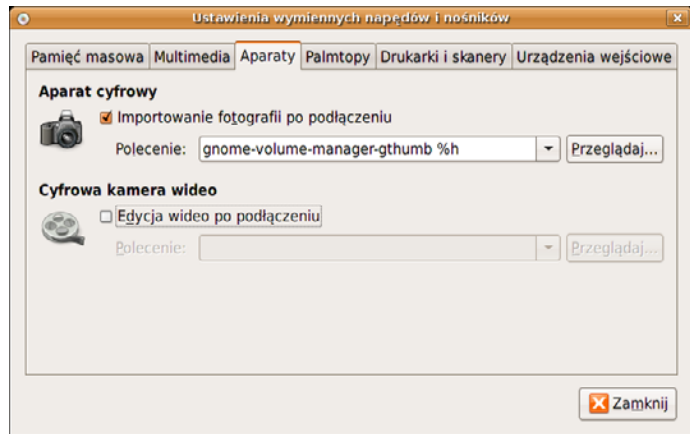
W rozdziale 13. wyjaśniłem, jak za pomocą okna dialogowego *Ustawienie wymiennych napędów i nośników* skonfigurować system, aby podejmował konkretne działania po włożeniu do napędu pustej płyty CD lub DVD. Płyty CD lub DVD to prawdopodobnie najczęściej wykorzystywane nośniki pamięci, ale urządzenia elektroniczne plasują się tuż za nimi. W tym samym oknie dialogowym można określić, co ma zrobić system operacyjny, kiedy wykryje aparat cyfrowy, organizator elektroniczny, cyfrowy odtwarzacz audio itd. W rozdziale 23., „Dodawanie sprzętu i przyłączanie urządzeń peryferyjnych”, napisałem o współpracy systemu z przenośnymi mediami. W dalszej części tego rozdziału skupię się na skonfigurowaniu systemu do obsługi różnego rodzaju urządzeń elektronicznych podłączanych do Ubuntu.

Konfigurowanie obsługi aparatów cyfrowych i cyfrowych kamer wideo

Aby skonfigurować system do obsługi aparatów cyfrowych i cyfrowych kamer wideo, należy wybrać z menu *System/Preferencje/Napędy i nośniki wymienne*, a następnie kliknąć zakładkę *Aparaty* (patrz rysunek 17.1).

W tym oknie dialogowym można ustawić opcje ułatwiające pracę z opisanymi niżej urządzeniami.

Rysunek. 17.1.
Konfigurowanie
obsługi aparatów
i kamer cyfrowych



- ♦ **Aparat cyfrowy** — jeżeli zostanie zaznaczona opcja *Importowanie fotografii po podłączeniu*, wówczas system operacyjny po wykryciu, że do komputera został podłączony poprzez łącze USB lub FireWire aparat cyfrowy, uruchomi wskazaną aplikację. Domyślnie jest to program gthumb — uruchamiany przez skrypt powłoki o nazwie *gnome-volume-manager-gthumb* — do pobierania i wyświetlania zdjęć cyfrowych. Użytkownik może wskazać inny program, w tym celu należy kliknąć przycisk *Przeglądaj* i wybrać dowolną aplikację.
- ♦ **Cyfrowa kamera wideo** — jeżeli zostanie zaznaczona opcja *Edycja wideo po podłączeniu*, wówczas system operacyjny po wykryciu, że do komputera została podłączona poprzez łącze USB lub FireWire kamera cyfrowa, uruchomi wskazaną aplikację. W standardowej instalacji Ubuntu nie znajduje się aplikacja do edycji plików wideo. Najpopularniejsze programy służące do tego celu to dvgrab i kino, które można zainstalować z repozytoriów Ubuntu za pomocą programu *apt-get* lub menedżera pakietów Synaptic, zgodnie ze wskazówkami zawartymi w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

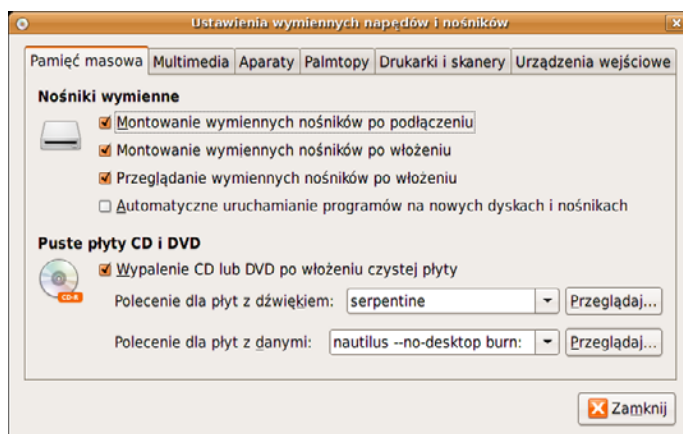
Korzystanie ze skryptu *gnome-volume-manager-gthumb* oraz skojarzonego z nim programu gthumb opisałem w podrozdziale „Aparaty cyfrowe i Ubuntu”, w którym zamieściłem również informację, co zrobić w sytuacji, kiedy system operacyjny nie rozpoznał aparatu.

Konfigurowanie obsługi iPodów i innych odtwarzaczy muzycznych

Zależnie od typu pamięci masowej używanej w odtwarzaczach, są one rozpoznawane przez system jako nośniki podłączane przez USB (lub FireWire) lub właśnie odtwarzacze audio. Aby skonfigurować system do ich obsługi, należy wybrać z menu *System/Preferencje/Napędy i nośniki wymienne*, a następnie przejść do zakładki *Pamięć masowa* (patrz rysunek 17.2).

Aby umożliwić rozpoznanie jak największej liczby typów różnorodnych odtwarzaczy audio, należy zaznaczyć pola wyboru *Montowanie wymiennych nośników po podłączeniu* i *Montowanie wymiennych nośników po włożeniu*. Jeżeli użytkownik korzysta z odtwarzacza

Rysunek 17.2.
Konfigurowanie obsługi nośników pamięci



innego niż iPod (a rozpoznawanego przez Ubuntu), wówczas warto zaznaczyć również pole wyboru *Przeglądanie wymiennych nośników po włożeniu*. Nie jest to konieczne, ale upraszcza korzystanie z wielu odtwarzaczy, a z pewnością im nie szkodzi.

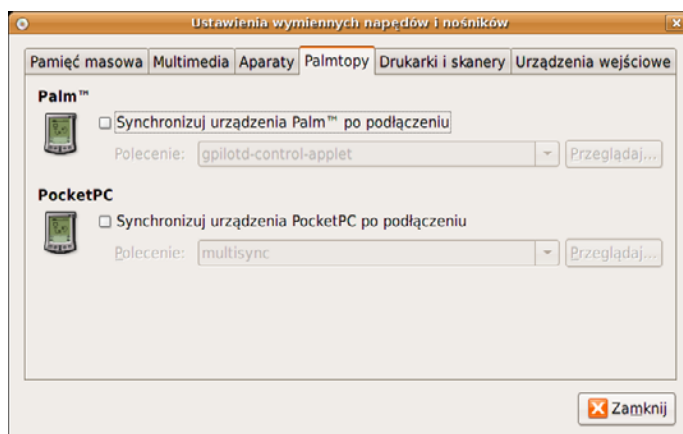


Jeżeli użytkownik chce, aby po podłączeniu do komputera automatycznie była uruchamiana wybrana aplikacja, wówczas należy sprawdzić, czy w zakładce *Multimedia* w sekcji *Przenośne odtwarzacze muzyczne* podany jest odpowiedni program. Domyślnie jest to Rhythmbox, ale użytkownik może zmienić ten wpis na np. gtkpod lub gtkpod-aac; o tych aplikacjach napiszę w dalszej części rozdziału.

Konfigurowanie obsługi palmtopów i smartfonów

GNOME rozpoznaje palmtopy z systemami Palm, Microsoft Windows CE, Microsoft Windows Pocket Edition i większość typów smartfonów korzystających z wymienionych systemów. GNOME można skonfigurować tak, aby po podłączeniu urządzenia z uruchomionym systemem Palm OS czy odchudzoną wersją Windows i podaniu polecenia synchronizacji włączał określoną aplikację. Aby skonfigurować system do obsługi tych urządzeń, należy wybrać z menu *System/Preferencje/Napędy i nośniki wymienne*, a następnie przejść do zakładki *Palmtopy* (patrz rysunek 17.3).

Rysunek 17.3.
Konfigurowanie urządzeń PDA



Dostępne są tam dwie opcje:

- ♦ **Urządzenia działające pod kontrolą systemu Palm OS** — aby po podłączeniu takiego urządzenia do komputera włączany był określony program, należy zaznaczyć opcję *Synchronizuj urządzenia Palm™ po podłączeniu*. Jeżeli pole zostanie zaznaczone, wówczas Ubuntu po podłączeniu takiego urządzenia będzie uruchamiał aplet *gpilotd-control-aplet*. Aby wskazać inny program lub skrypt, który ma być uruchamiany, należy kliknąć przycisk *Przeglądaj* i w oknie nawigacyjnym wskazać odpowiedni plik.
- ♦ **Urządzenia działające pod kontrolą systemu Microsoft Windows** — aby po podłączeniu takiego urządzenia do komputera włączany był określony program, należy zaznaczyć opcję *Synchronizuj urządzenia PocketPC po podłączeniu*. Jeżeli pole zostanie zaznaczone, wówczas Ubuntu po podłączeniu takiego urządzenia będzie uruchamiał *multisync*. Aby wskazać inny program lub skrypt, który ma być uruchamiany, należy kliknąć przycisk *Przeglądaj* i w oknie nawigacyjnym wskazać odpowiedni plik.

Zależnie od systemu operacyjnego uruchomionego na urządzeniu PDA, przed pierwszą synchronizacją trzeba skonfigurować *gpilotd-control-aplet* lub *multisync*. Więcej informacji na ten temat podaję w podrozdziale „Palmtopy, smartfony i Ubuntu”.

Konfigurowanie obsługi kart pamięci i innych urządzeń elektronicznych

Niektóre aparaty cyfrowe i odtwarzacze audio nie są wyposażone w port USB czy Fire-Wire, za których pomocą można by podłączyć je bezpośrednio do komputera w celu aktualizacji danych. Na szczęście, urządzenia te zwykle korzystają z przenośnych nośników pamięci, takich jak karty CompactFlash czy SD, które można połączyć z komputerem za pośrednictwem specjalnego czytnika. Aby skonfigurować system do ich obsługi, należy wybrać z menu *System/Preferencje/Napędy i nośniki wymienne*, a następnie przejść do zakładki *Pamięć masowa* (patrz rysunek 17.2).

Aby system automatycznie montował te typy nośników, trzeba zaznaczyć opcję *Montowanie wymiennych nośników po włożeniu*. Jeżeli zostanie zaznaczona opcja *Przeglądanie wymiennych nośników po włożeniu*, wówczas po podłączeniu urządzenia system automatycznie uruchomi przeglądarkę plików Nautilus i będzie można obejrzeć zawartość danego nośnika, a także w razie potrzeby przenieść lub przekopiować pliki. Więcej informacji na ten temat można znaleźć w podrozdziale „Obsługa kart CompactFlash i SD”.

Aparaty cyfrowe i Ubuntu

Jeżeli użytkownik postąpił zgodnie ze wskazówkami zamieszczonymi w punkcie „Konfigurowanie obsługi aparatów cyfrowych i cyfrowych kamer wideo”, system jest skonfigurowany tak, że podłączeniu aparatu cyfrowego do komputera zostanie wykonany skrypt *gnome-volume-thumb-manager*, a do importu, porządkowania i wyświetlania zdjęć zostanie wykorzystana przeglądarka *gthumb*. Kiedy aparat zostanie połączony z komputerem poprzez port USB i włączony, wówczas system wyświetli komunikat widoczny na rysunku 17.4.

Rysunek 17.4.

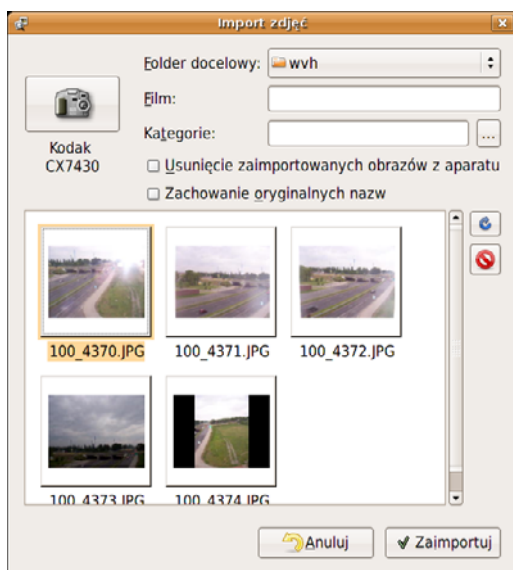
Ubuntu wykrył
aparat podłączony
do komputera



Aby skopiować zdjęcia z aparatu do komputera, należy kliknąć przycisk *Importuj fotografie*, innym rozwiązaniem może być użycie przycisku *Ignoruj*. Jeżeli dana akcja ma być zawsze wykonywana przez system, wówczas przed kliknięciem jednego z dwóch przycisków należy zaznaczyć pole *Zawsze podejmuj tę akcję*. Jeżeli użytkownik wybierze opcję *Importuj fotografie*, wówczas system załaduje sterowniki dla danego aparatu, a następnie uruchomi program *gthumb*, w którym dostępnych będzie kilka opcji importu. Okno dialogowe widoczne jest na rysunku 17.5.

Rysunek 17.5.

Program *gthumb*
przed rozpoczęciem
importu zdjęć
z aparatu



Oto opis opcji dostępnych w oknie programu *gthumb*, z którymi warto się zapoznać przed rozpoczęciem importowania zdjęć do komputera.

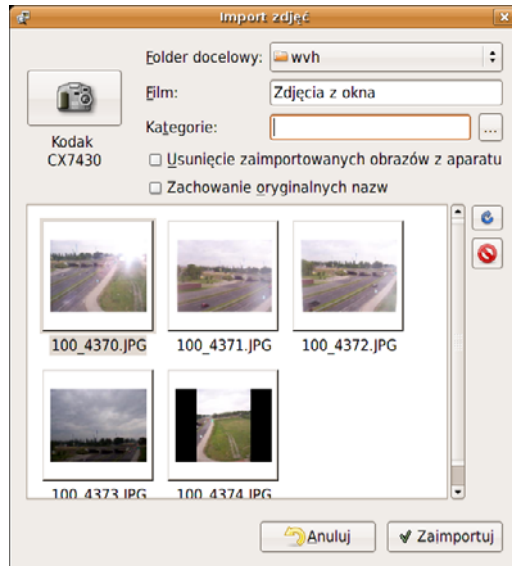
- ♦ **Folder docelowy** — tu można ustawić folder, w którym przechowywane są importowane zdjęcia. Domyślnie program *gthumb* importuje zdjęcia do katalogu znajdującego się w katalogu domowym użytkownika, a którego nazwa to data importu zdjęć (i tak folder o nazwie *2006-05-18 – 12.31.43* zawiera zdjęcia zaimportowane 18 maja 2006 roku o godzinie 12:31). Użytkownik może zmienić miejsce, w którym ten folder będzie tworzony; wystarczy rozwinąć listę i wskazać nową lokalizację. Przyjąłem metodę przechowywania zdjęć w specjalnym folderze o nazwie *Zdjęcia*, ale użytkownik może dowolnie zmodyfikować tę opcję.
- ♦ **Film** — to nazwa grupy importowanych zdjęć; jak wspominałem w powyższym akapicie, program *gphoto* używa daty importu jako nazwy dla katalogu, w którym będą przechowywane zdjęcia. W tym polu użytkownik może określić własną nazwę,

która zostanie nadana tworzonemu folderowi. Katalog ten powstanie w lokalizacji wskazanej w polu *Folder docelowy*.

- ♦ **Kategorie** — umożliwia przypisanie importowanych zdjęć do danej kategorii, program oferuje kilka przykładowych kategorii, które można wykorzystać, dzięki czemu wyszukiwanie podobnych zdjęć będzie w przyszłości łatwiejsze. Po kliknięciu przycisku zostanie rozwinięta lista z dostępnymi kategoriami. Użytkownik może również określić własne kategorie (patrz rysunek 17.6).

Rysunek 17.6.

Przypisywanie kategorii importowanym fotografiom



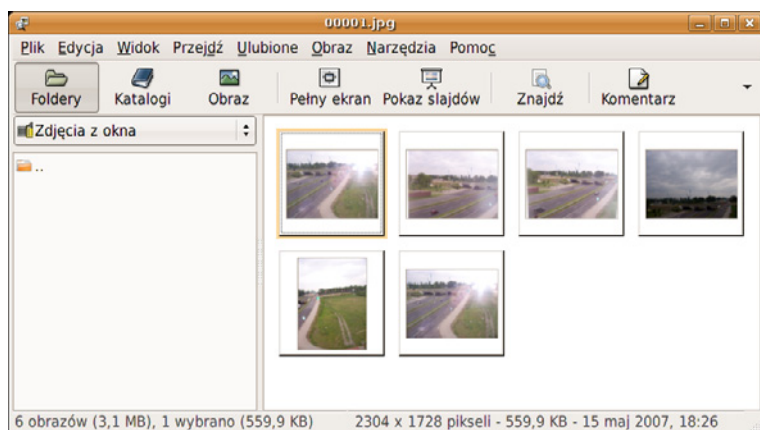
- ♦ **Usunięcie zaimportowanych obrazów z aparatu** — po zaznaczeniu tej opcji zdjęcia — po pomyślnie zakończonym imporcie — zostaną usunięte z aparatu i zwolni się miejsce na następne.
- ♦ **Zachowanie oryginalnych nazw** — domyślnie program nadaje importowanym zdjęciom kolejne numery, rozpoczynając numerowanie od początku dla każdego procesu importu, dzięki czemu można je odróżnić. Jeżeli użytkownik chciałby zachować oryginalne nazwy zdjęć nadane im przez aparat fotograficzny (z których każde jest unikalne dla każdej używanej karty pamięci), wówczas powinien zaznaczyć to pole.

Po wprowadzeniu zmian w opisanych polach można rozpocząć import zdjęć (klikając przycisk *Zaimportuj*) lub anulować cały proces (przycisk *Anuluj*). Jeżeli użytkownik zdecyduje się przeprowadzić import, wówczas po zakończeniu tego procesu zostanie wyświetlone okno dialogowe widoczne na rysunku 17.7.

Jeżeli użytkownik zaznaczy opcję usunięcia zdjęć po imporcie, należy zachować ostrożność. Zwykle importuję zdjęcia z wyłączoną powyższą opcją. Dopiero kiedy po skopiowaniu zdjęć upewnię się, że wszystko się zgadza, usuwam zdjęcia z aparatu. Skrypt *gnome-volume-thumb-manager* jest stosunkowo prosty i wykrywa tylko te zdjęcia, które są przechowywane w określonym miejscu. Jeżeli skrypt ten nie zadziała odpowiednio w przypadku aparatu użytkownika, wówczas można wypróbować inny skrypt, autorstwa Petera

Rysunek 17.7.

Okno ze zdjęciami
po zakończonym
procesie importu



Parkkalisa, dostępny na stronie <http://pfp.iki.fi/gnome/import-photos>. Jeżeli natomiast dany aparat nie jest rozpoznawany przez program gthumb, zdjęcia można skopiować samodzielnie, korzystając ze wskazówek zamieszczonych w podrozdziale „Obsługa kart CompactFlash i SD”.



Wiele aparatów cyfrowych wyłącza się samoczynnie po pewnym okresie bezczynności. To świetne rozwiązanie w zakresie oszczędzania energii, ale może doprowadzić do przerwania importu. Jeżeli po zaimportowaniu zdjęć program gthumb utworzy pliki, które nie będą zdjęciami, należy powtórzyć cały proces.

Palmtopy, smartfony i Ubuntu

Osobiści cyfrowi asystenci, czyli po prostu palmtopy, to znakomite narzędzia do zarządzania czasem czy sporządzania krótkich notatek. Jeżeli są wyposażone w łącze sieciowe, a użytkownik dysponuje wystarczająco dobrym wzrokiem, to można za ich pomocą przeglądać internet, wysyłać i odbierać pocztę elektroniczną, a nawet pracować z dokumentami elektronicznymi w popularnych formatach — wszystko za pomocą urządzenia mieszczącego się w kieszeni koszuli. Oprogramowanie potrzebne do synchronizacji palmtopa działającego pod kontrolą systemu Palm OS jest już zainstalowane w Ubuntu, a jeżeli urządzenie działa pod kontrolą Microsoft Windows Pocket PC lub Microsoft Windows Pocket CE PDA, wówczas wystarczy tylko sięgnąć do repozytoriów. Poniżej napisałem, jak zainstalować, skonfigurować i wykorzystywać oprogramowanie niezbędne do synchronizowania danych przechowywanych w Ubuntu z niemal każdym dostępnym palmtopem.



Jeżeli użytkownik chce zsynchronizować palmtop poprzez port szeregowy, musi wówczas należeć do grupy *dialout*. Aby wyświetlić listę grup, do których użytkownik należy, wystarczy użyć polecenia `groups` w konsoli GNOME. Więcej informacji o użytkownikach i grupach w Ubuntu można znaleźć w rozdziale 21., „Zarządzanie użytkownikami, grupami i zaawansowane uprawnienia”.

Konfigurowanie i synchronizowanie urządzeń Palm OS

Jak wspominałem wcześniej w punkcie „Konfigurowanie obsługi palmtopów i smartfonów”, standardowo w Ubuntu instalowany jest aplet o nazwie *gpilot* (czasem dla tej aplikacji używa się nazwy pakietu ją zawierającego, czyli *gnome-pilot* lub *pilot-link*). W rozdziale 5., w punkcie „Dostosowywanie paneli” wyjaśniłem, jak można dodać do panelu kolejny aplet; teraz powtórzę w skrócie: należy kliknąć prawym przyciskiem myszy wybrany panel, z wyświetlonego okna wybrać odpowiedni aplet i przeciągnąć go na panel. Aplet służący do synchronizowania palmtopów nosi nazwę *Pilot Applet* i znajduje się w sekcji o nazwie *Narzędzia*.

Po dodaniu apletu *gpilot* do panelu należy jeszcze skonfigurować sposób, w jaki będzie on komunikował się z palmtopem.



Podczas pierwszego uruchomienia apletu zostanie wyświetlone okno kreatora, który przeprowadzi użytkownika przez proces konfiguracji. Przebiega on tak, jak napisałem poniżej, w dodatku w tej samej kolejności. Poniżej omówię samodzielną konfigurację, ponieważ kreator jest uruchamiany tylko raz. Jeżeli użytkownik będzie chciał wprowadzić jakieś zmiany w późniejszym czasie, wówczas musi to zrobić sam, zgodnie z zamieszczonymi poniżej wskazówkami.

Aby skonfigurować ustawienia synchronizacji, należy kliknąć ikonę apletu znajdującą się na panelu, a zostanie wyświetlone okno widoczne na rysunku 17.8.

Rysunek 17.8.

Konfigurowanie
ustawień
synchronizacji
z Palm OS

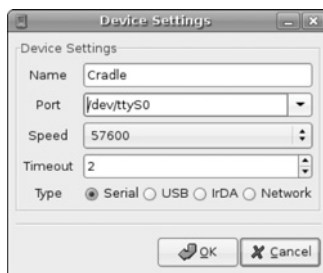


Przed przejściem do kolejnego etapu należy zweryfikować lub w ogóle zdefiniować połączenie z palmtopem. Możliwe połączenia to szeregowe, poprzez port USB, port IrDA (podczerwień) lub sieć. Jeżeli wykorzystywane jest połączenie poprzez USB lub szeregowe, należy połączyć palmtop z komputerem, a następnie — niezależnie od typu połączenia — uruchomić urządzenie i odczekać kilka chwil, aż Ubuntu utworzy niezbędne węzły. Na końcu trzeba przejść do zakładki *Devices*, kliknąć przycisk *Add* i zdefiniować nowy wpis dla połączenia *gpilota* z palmtopem (patrz rysunek 17.9).

Następnie należy wybrać odpowiedni rodzaj połączenia (pole wyboru *Type*) oraz uzupełnić pole *Port* do komunikacji z palmtopem. Jeżeli nie jest wykorzystywane połączenie szeregowe, można wpisać po prostu `/dev/pilot`, w przypadku portu szeregowego wpis zwykle ma postać `/dev/ttyS0`.

Rysunek 17.9.

Konfigurowanie
ustawień urządzenia
do komunikowania
się palmtopem



Domyślna nazwa urządzenia podana w polu *Name* to *Cradle*, jeżeli użytkownik zamierza synchronizować więcej niż jeden palmtop, może rozważyć podanie osobnych nazw dla każdego urządzenia.

Aby zapisać wprowadzone ustawienia, należy kliknąć przycisk *OK* widoczny na rysunku 17.8. Następnie przejść do zakładki *Pilots* i kliknąć *Add*, aby dodać informację o użytkowniku korzystającym z danego połączenia; zostanie wyświetlone okno dialogowe widoczne na rysunku 17.10.

Rysunek 17.10.

Wprowadzanie
informacji
o użytkowniku



Aby pobrać dane o użytkowniku znajdujące się w palmtopie, należy użyć przycisku *Get from pilot*. Kiedy gpilot wyświetli odpowiedni komunikat, trzeba uruchomić program HotSync oraz wcisnąć przycisk *Sync*. Palmtop wymieni wówczas odpowiednie informacje z Ubuntu (patrz rysunek 17.11).

Rysunek 17.11.

Informacje
o użytkowniku
pobrane z palmtopa



Ostatnim etapem konfiguracji jest zdefiniowanie, które dane mają być synchronizowane z palmtopem. Każdy typ informacji, które mają być synchronizowane, to tzw. **conduit**. W programie gpilot zdefiniowano całkiem sporo różnych typów danych, które można

wymieniać z Ubuntu, ale większość z nich jest wyłączona. Aby je aktywować, należy przejść do zakładki *Conduits*, widocznej na rysunku 17.12.

Rysunek 17.12.

Określanie informacji,
które mają być
synchronizowane



Oto dostępne w programie gpilot typy danych.

- ♦ **Backup** — wykonywana jest kopia zapasowa wszystkich baz danych znajdujących się w palmtopie, kopie umieszczane są w lokalizacji określonej w zakładce *Pilots*.
- ♦ **EAddress** — synchronizowanie książek adresowych pomiędzy palmtopem a programem Evolution (szerzej omówionym w rozdziale 8.).
- ♦ **ECalendar** — synchronizowanie informacji w kalendarzach pomiędzy palmtopem a programem Evolution.
- ♦ **EMemos** — synchronizowanie notatek pomiędzy palmtopem a programem Evolution.
- ♦ **EToDo** — synchronizowanie listy zadań pomiędzy palmtopem a programem.
- ♦ **Expense** — pobieranie danych w różnych formatach z baz znajdujących się w palmtopie.
- ♦ **File** — instalowanie określonych plików na palmtopie.
- ♦ **MAL** — synchronizacja palmtopa z informacjami z AvantGo (<http://www.avantgo.com/frontdoor/index.html>).
- ♦ **MemoFile** — synchronizacja notatek znajdujących się palmtopie z umieszczonymi w plikach lokalnych systemu Ubuntu.
- ♦ **Sendmail** — wysyłanie za pomocą aplikacji Ubuntu wiadomości pocztowych zakolejkowanych w PDA.
- ♦ **Test** — testowanie połączenia pomiędzy palmtopem a systemem Ubuntu; używany do diagnostyki problemów występujących podczas synchronizacji.
- ♦ **Time** — synchronizacja czasu wyświetlanego w palmtopie z używanym w Ubuntu. Działa tylko z systemami Palm OS 3.3 lub nowszymi.

Wszystkie typy danych mogą być wyłączone (ustawienie domyślne) lub włączone. Aby je aktywować, należy wybrać dany typ, kliknąć przycisk *Settings*, a zostanie wyświetlone okno dialogowe widoczne na rysunku 17.13.

Rysunek 17.13.

Szczegółowa
konfiguracja
typów informacji



W oknie tym można ustawić następujące dane.

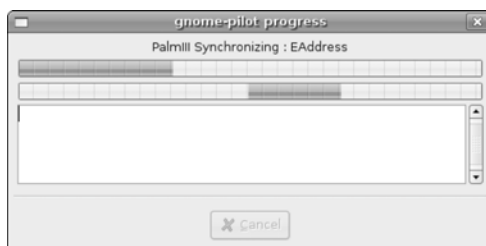
- ♦ **Action** — aktualny status typu danych, kiedy jest wyłączony (*Disable*), można go włączyć (*Enable*); zsynchronizować (*Synchronize*), co działa w obie strony, od i do palmtopa, oraz skopiować dane z komputera do palmtopa (*Copy*).
- ♦ **One Time Action** — pozwala na określenie działania, które zostanie podjęte podczas następnego połączenia palmtopa z komputerem. Dostępne możliwości to *None* — nie zostanie podjęta żadna akcja, *Synchronize* — dwukierunkowe synchronizowanie danych oraz *Copy*, czyli kopiowanie danych z komputera do palmtopa.

Opcje dostępne dla innych typów danych zależą od ich rodzaju. I tak np. dane EAddress, których okno konfiguracyjne jest widoczne na rysunku 17.13, umożliwiają wskazanie, która książka adresowa ma zostać zsynchronizowana, czy mają być wymieniane informacje prywatne i publiczne oraz która z książek dostępnych w palmtopie ma zostać zaktualizowana, czy w ogóle użyta.

Po skonfigurowaniu programu gpilot sama synchronizacja jest już prosta. Należy połączyć urządzenie z komputerem, włączyć je, uruchomić HotStart (lub inną aplikację do synchronizowania danych) i wcisnąć klawisz *Sync*. Zostanie wyświetlone okno widoczne na rysunku 17.14, co będzie oznaczało, że proces wymiany danych został rozpoczęty.

Rysunek 17.14.

Synchronizowanie
danych



I to właściwie wszystko! Jeżeli użytkownik napotka na komplikacje w użytkowaniu programu gpilot lub po prostu będzie chciał sprawdzić inne możliwości, wówczas powinien wypróbować aplikację multisync opisaną poniżej. Pakiet *multisync* to bardzo wydajny program i elastyczne rozwiązanie umożliwiające (prawie) synchronizację Ubuntu z kuchenką i lodówką.

Użytkownicy chcący wypróbować inne otwarte oprogramowanie przeznaczone dla urządzeń działających pod kontrolą systemu Palm OS mogą sprawdzić dwa inne programy.

Pierwszy z nich to jpilot, aplikacja oparta na Javie i alternatywa wobec Palm Dekstop dostarczanego zwykle przez producenta palmtopa; druga to kpilot, aplikacja przeznaczona dla środowiska KDE. Program jpilot wydaje się być lepszym rozwiązaniem, ponieważ działa niezależnie od tego, czy użytkownik korzysta z GNOME, czy KDE, a co więcej, w obu działa tak samo. Jednak moją ulubioną aplikacją jest kpilot, mimo że do działania wymaga zainstalowania bibliotek środowiska KDE. Jest to doskonały program, przyjazny dla użytkownika i prosty w codziennej obsłudze (czego zresztą można się było spodziewać po programistach KDE). Jego podstawową wadą jest fakt, że został zaprojektowany do pracy z aplikacjami pakietu KDE Office, co z kolei powoduje, że nie jest w stanie wymienić wszystkich informacji z programami środowiska GNOME, takimi jak np. Evolution, a także to, że trwa właśnie proces zastępowania tego programu przez aplikację KitchenSync, tworzoną przez zespół OpenSync (www.opensync.org).

Konfigurowanie i synchronizowanie urządzeń z systemem Microsoft Windows

Poniżej opowiem, jak synchronizować z Ubuntu palmtopy i smartfony pracujące pod kontrolą systemów Microsoft Windows Pocket Edition czy Microsoft Windows CE. Niezbędny do tego program, multisync, to niesamowicie elastyczna i wydajna aplikacja korzystająca z technologii wykorzystującej wtyczki, dzięki czemu może obsługiwać wiele różnorodnych urządzeń mobilnych z Windows na pokładzie, ale też korzystających z systemu Palm OS Linuksa (takie jak Sharp Zaurus) i w ogóle większość współczesnych palmtopów. Strona domowa programu multisync znajduje się pod adresem <http://multisync.sourceforge.net/news.php>.



Z mojego doświadczenia wynika, że wsparcie synchronizacji dla palmtopów z systemami Windows jest bardzo kruche. Poza tym dużo lepsze rezultaty można osiągnąć, korzystając z 9-pinowej podstawki szeregowej, a nie z USB. Różne wersje multisynca i wykorzystywane w nich wtyczki potrafią mieć swoje dziwactwa. Niniejszy opis powstał podczas testowania palmtopa iPaq 3650 z uruchomionym systemem Windows CE 3.0.11171, który jest moim ulubionym urządzeniem, ponieważ łatwo na nim zainstalować Linuksa. Zapewne w innych modelach mogą pojawić się inne trudności. Warto pamiętać, że w takich sytuacjach najlepszymi przyjaciółmi użytkownika będą wyszukiwarki internetowe, takie jak Google.

Pakiet *multisync* nie jest domyślnie instalowany w Ubuntu głównie z powodu jego elastyczności, co wiąże się z ogromną liczbą dostępnych wtyczek. Z tego powodu trudno przewidzieć, który zestaw wtyczek będzie przydatny dla palmtopa, z którego korzysta użytkownik. W związku z tym, to właśnie użytkownikowi pozostawiono decyzję o tym, czy instalacja multisynca jest potrzebna. Na szczęście, w Ubuntu jest to całkiem proste zadanie. Tak więc — do dzieła!

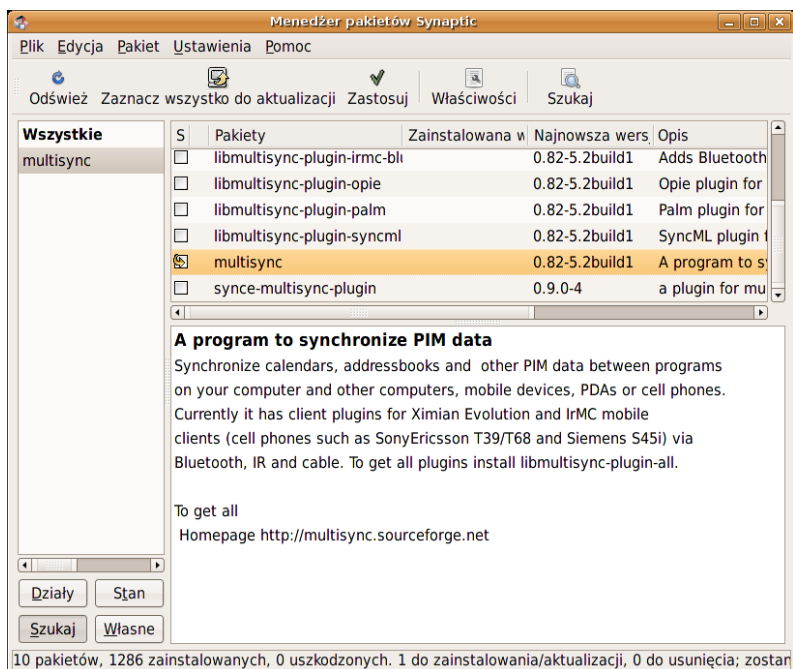


Następcą multisynca jest program OpenSync; w którym wykorzystano nowocześniejszą architekturę z łatwiejszym w użytkowaniu systemem wtyczek potrzebnych do synchronizacji z wieloma urządzeniami, aplikacjami i protokołami. Więcej informacji na temat tego projektu można znaleźć na stronie <http://www.opensync.org/>. Tam też umieszczono pakiety dla Debiana, nowe wersje multisynca (które są konieczne do działania) oraz nowe wtyczki. Należy jednak pamiętać, że sam program znajduje się nadal w fazie testów i nie został jeszcze oficjalnie opublikowany. Jeżeli użytkownik chciałby zacząć korzystać z OpenSynca, zapewne będzie też chciał skompilować z kodu najnowsza i „najlepsza” wersja programu; odpowiednie pliki można znaleźć na stronie <http://www.opensync.org/wiki/download>.

Instalowanie programu multisync, wtyczek, bibliotek oraz powiązanych narzędzi

Aby zainstalować program multisync, wtyczki, biblioteki oraz powiązane narzędzia, należy uruchomić menedżer pakietów Synaptic (z menu *System/Administracja*), kliknąć przycisk *Szukaj* i wpisać ciąg *multisync*, a następnie rozpocząć wyszukiwanie. Na wyświetlonej liście pakietów trzeba odszukać i zaznaczyć do instalacji pakiet o nazwie *multisync* (patrz rysunek 17.15).

Rysunek 17.15.
Instalowanie pakietu *multisync*



Aby zainstalować program multisync, muszą być aktywne repozytoria *universe*. W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, znajdują się informacje na temat dodawania i aktywowania dodatkowych repozytoriów.

Oprócz samego programu, należy też zainstalować wtyczki, które umożliwią synchronizowanie danych z palmtopem. Niezbędny z pewnością jest pakiet o nazwie *libmultisync-plugin-all*, zawierający wszystkie standardowe wtyczki używane przez program multisync (zawiera też wtyczkę o nazwie *libmultisync-plugin-evolution*, która umożliwia synchronizowanie danych z programem pocztowym), a także pakiet *synce-multisync-plugin*, który z kolei pozwala na synchronizację urządzeń z systemem Windows CE. Należy też przejrzeć całą listę pod kątem pakietów przeznaczonych do obsługi danego modelu palmtopa posiadanego przez użytkownika, używanego na nim systemu operacyjnego oraz specjalnych mechanizmów synchronizacji, takich jak np. SyncML.

Zależnie od zainstalowanego już w systemie oprogramowania oraz tego, które zostało teraz zaznaczone w Synapticu, system może wyświetlić listę dodatkowych pakietów koniecznych do dodania do systemu i poprosić o jej zatwierdzenie. Jeżeli taki komunikat zostanie wyświetlony, należy zaakceptować wskazane pakiety.

Następnie trzeba wyszukać w Synapticu ciąg „synce” i zaznaczyć do instalacji pakiety: *librra0-tools*, *synce-dccm* i *synce-serial*.

Po zaznaczeniu wszystkich potrzebnych pakietów należy kliknąć znajdujący się na pasku narzędzi programu Synaptic przycisk *Zastosuj*.



Po zainstalowaniu pakietów Synaptic wyświetli okno dialogowe, w którym użytkownik będzie mógł przeprowadzić konfigurację programów. Domyślnie w polu *Serial Interface* wyświetlane jest urządzenie połączone za pomocą podstawki USB. Jeżeli użytkownik korzysta z połączenia szeregowego (jedyne działające połączenie z palmtopem iPaq będącym w moim posiadaniu), wówczas należy zmienić wpis na */dev/ttyS0*. Nie trzeba modyfikować żadnych innych ustawień; jeżeli jednak użytkownik chciałby je zmienić, powinien użyć polecenia *synce-serial-config* (wraz z poleceniem *sudo*), a uzyskać dostęp do odpowiednich opcji.

Kiedy instalacja zostanie zakończona, wszystko jest już właściwie gotowe do rozpoczęcia procesu synchronizacji (oczywiście, trzeba jeszcze przeprowadzić niezbędne czynności konfiguracyjne).

Więcej informacji o instalowaniu oprogramowania za pomocą menedżera pakietów Synaptic można znaleźć w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

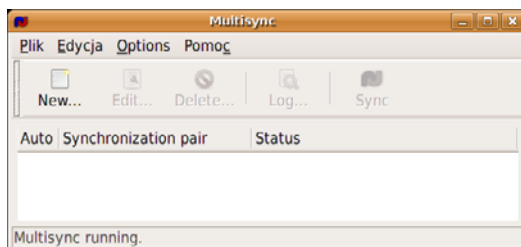
Definiowanie par do synchronizacji

Pary synchronizacyjne (ang. *synchronization pairs*) występujące w środowisku Palm pod nazwą *conduits*, to zestawy wtyczek wejściowych i wyjściowych szczegółowo definiujących typ synchronizacji, jaką użytkownik chce przeprowadzić. Choć takie rozwiązanie oferuje dużą elastyczność synchronizacji, oznacza również konieczność przeprowadzenia nieco bardziej złożonej konfiguracji, niż ma to miejsce w opisanym wcześniej przypadku synchronizacji palmtopów z systemem Palm OS. Poniżej opiszę sposób utworzenia par synchronizacyjnych, dzięki którym będzie można wymienić dane między książką adresową Pocket PC czy Windows CE a programem Evolution dostępnym w systemie Ubuntu. Użytkownik z pewnością będzie chciał utworzyć własne pary, ale zestaw oraz liczba możliwości zależą bezpośrednio od typu używanego palmtopa oraz oprogramowania do synchronizacji plików.

Po zainstalowaniu programu należy go uruchomić, wybierając kolejno z menu *Aplikacje/Akcesoria/Multisync*. Po włączeniu programu zostanie wyświetlone okno dialogowe widoczne na rysunku 17.16.

Rysunek 17.16.

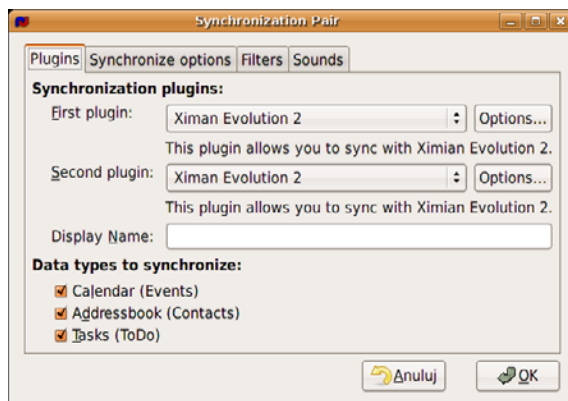
Ekran startowy programu multisync



Aby skonfigurować nową parę, należy kliknąć przycisk *Sync*, a zostanie wyświetlone kolejne okno dialogowe widoczne na rysunku 17.17.

Rysunek 17.17.

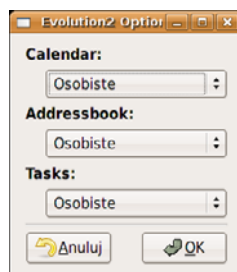
Definiowanie
wtyczek dla danych
wejściowych
i wyjściowych



Jeżeli zsynchronizowane mają być informacje w Evolution i palmtopie, należy wybrać z listy rozwijanej *Second plugin* wtyczkę SyncCE. Teraz trzeba określić informacje, które mają być wymienione. Chociaż w dolnej części okna dialogowego widocznego na rysunku 17.17 zaznaczone są rodzaje danych, które będą synchronizowane (kalendarz — *Calendar*, książka adresowa — *Addressbook* i lista zadań — *Tasks*), to należy wskazać konkretne nazwy tych elementów. W tym celu trzeba kliknąć przycisk *Options* umieszczony z prawej strony pierwszej wtyczki, zostanie wówczas wyświetlone okno dialogowe widoczne na rysunku 17.18.

Rysunek 17.18.

Wskazywanie
elementów
do zsynchronizowania



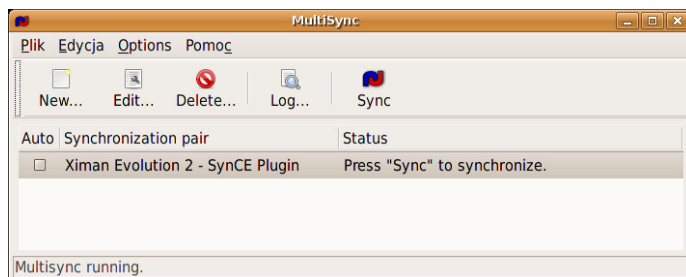
Podczas synchronizowania danych z Evolution nazwy alternatywnego kalendarza, książki adresowej czy listy zadań będą wyświetlane tylko wówczas, jeżeli wcześniej został skonfigurowany program pocztowy. Więcej informacji na ten temat można znaleźć w rozdziale 8.

Kiedy zobaczymy to okno dialogowe, początkowo nie będzie w nim żadnych pozycji dla kalendarza, książki adresowej czy listy zadań. Trzeba dopiero rozwinąć odpowiednie listy i z nich wybrać właściwe pozycje. Mają one zwykle oznaczenie *Osobiste* (patrz rysunek 17.18). Aby zapisać wprowadzone ustawienia, należy kliknąć przycisk *OK*, okno zostanie zamknięte, a pojawi się znowu okno widoczne na rysunku 17.17. Aby je zamknąć, również należy kliknąć przycisk *OK*, zostanie wówczas wyświetlone główne okno, ale już z aktualną treścią (patrz rysunek 17.19).

Dane konfiguracyjne zostały wprowadzone, pora więc na rozpoczęcie synchronizacji!

Rysunek 17.19.

Główne okno
dialogowe programu
multisync
po skonfigurowaniu
ustawień



Synchronizacja danych za pomocą programu Multisync

Po zainstalowaniu programu Multisync oraz skonfigurowaniu ustawień synchronizacja danych pomiędzy palmtopem z systemem Windows Pocket PC czy CE PDA a Ubuntu jest już zupełnie prosta.

Najpierw należy użyć polecenia `sudo`, aby uruchomić narzędzie `synce-serial-config`, które zdefiniuje połączenie między Ubuntu a iPaq-iem:

```
$ sudo synce-serial-config /dev/ttyS0
You can now run synce-serial-start to start a serial connection
```

W tym przykładzie palmtop jest podłączony do pierwszego portu szeregowego, jeżeli użytkownik skorzysta z portu USB, nazwa urządzenia powinna wyglądać mniej więcej tak: `/dev/ttyUSB0`. Aby zidentyfikować port szeregowy, do którego podłączony jest palmtop, można przejrzeć wpisy w plikach `/var/log/message`. Jeżeli zostanie wyświetlony inny komunikat niż zamieszczony powyżej, oznacza to, że nastąpił problem z ustanowieniem połączenia. Warto sprawdzić, czy poprawnie zidentyfikowano urządzenie, a następnie ponownie je podłączyć.

Teraz trzeba uruchomić demon `dccm`, który obsługuje połączenie z palmtopem, uruchamia demon PPP do komunikowania się z palmtopem, a także utrzymuje połączenie w stanie aktywnym tak długo, jak jest to potrzebne. Aby wykonać te wszystkie operacje, wystarczy w terminalu GNOME zrealizować polecenie `dccm` — bez konieczności posiłkowania się poleceniem `sudo`. Program `dccm` zostanie uruchomiony w trybie demona. Aby sprawdzić, czy program działa, można skorzystać z polecenia `ps alxww | grep dccm`, jak w poniższym przykładzie:

```
$ ps alxww | grep dccm
1 1000 16371 1 16 0 1928 780 - Ss ? 0:00 dccm
0 1000 16771 16431 16 0 2876 800 pipe_w S+ pts/2 0:00 grep dccm
```

Następnie należy użyć polecenia `sudo` i wykonać komendę `synce-serial-start`, oto przykład:

```
$ sudo synce-serial-start
synce-serial-start is waiting for your device to connect.
```

Jeżeli system dźwiękowy palmtopa nie został wyłączony lub też nie zmieniono jego konfiguracji, wówczas po połączeniu użytkownik powinien usłyszeć dźwięk. Jeżeli nie zmieniono ustawień i żaden dźwięk nie był słyszalny, należy podłączyć urządzenie ponownie.



Po podłączeniu można użyć polecenia `synce-pstatus` do pobrania informacji o palmtopie, takich jak zainstalowana wersja systemu Windows itp. Poprawne wykonanie tego polecenia sprawdzi również stan połączenia między urządzeniem a komputerem. Oto nieco skrócone dane wyjściowe uzyskane po zastosowaniu takiego polecenia:

```
$ synce pstatus
Version
=====
Version:      3.0.11171 (Merlin: Pocket PC 2002)
Platform:     3 (Windows CE)
Details:      " "

System
=====
Processor architecture: 5 (ARM)
Processor type:      2577 (StrongARM)
Page size:           0x10000
```

Teraz należy uruchomić znajdującą się w palmtopie aplikację *ActivSync* (menu *Start/ActiveSync*), która wyświetli informację o połączeniu.

Następnie trzeba uruchomić program *Multisync* (menu *Aplikacje/Akcesoria/Multisync*), upewnić się, że para synchronizacyjna jest zaznaczona, i wcisnąć przycisk *Sync*. Przez chwilę w polu *Status* wyświetlane będą różne komunikaty dotyczące połączenia i synchronizacji, a po chwili proces zostanie zakończony. Gratulacje, dane zostały zaktualizowane.



Jeżeli pojawią się problemy z korzystaniem z programu *Multisync*, należy uruchomić go z linii poleceń, używając komendy `/usr/bin/multisync`, zamiast korzystać z menu. Zostanie wyświetlona spora porcja danych wyjściowych i łatwiej będzie zdiagnozować problem.

iPody, inne odtwarzacze audio i Ubuntu

Cyfrowe odtwarzacze audio różnych rodzajów w większości już zastąpiły inne przenośne odtwarzacze muzyczne, zarówno kasetowe, jak i odtwarzające płyty CD. Cyfrowe odtwarzacze nie przeskakują po utworach, są bardzo małe, zużywają niewiele energii i są odporne na uszkodzenia, ponieważ zawierają niewiele ruchomych części. Z drugiej strony, kiedy taka nieruchoma część, np. dysk, ulegnie awarii, zwykle ma to katastrofalne skutki. Dlatego ważne jest posiadanie kopii zapasowych kolekcji muzycznych na dyskach komputerów, bo są tam bezpieczniejsze.

W niniejszej części wyjaśnię, jak korzystać z różnego rodzaju odtwarzaczy muzycznych, które ze względu na ogromną ilość różnych typów podzielono na iPody i „nieiPody”. Poniżej opiszę, jak zainstalować i wykorzystywać program o nazwie *gtkpod*, który umożliwia kopiowanie plików (również audio), tworzenie list odtwarzania i inne czynności, jakie użytkownik chciałby robić ze swoim iPodem. Następnie powiem, jak używać w Ubuntu odtwarzaczy innych typów.

Edytowanie znaczników ID3 w Ubuntu

Jak wspominałem w rozdziale 13., większość używanych obecnie, popularnych formatów plików dźwiękowych, takich jak MP3, umożliwia stosowanie znaczników ID3, zawierających informacje o plikach, np. tytuł albumu, wykonawcę, nazwę utworu, datę nagrania itp. Wszystkie odtwarzacze potrafią odczytać i wyświetlić te dane, a w większości z nich można ich użyć do wyszukiwania i sortowania biblioteki muzycznej. Dlatego przed wyeksportowaniem plików do odtwarzacza należy sprawdzić poprawność znaczników ID3, w przeciwnym razie można mieć trudności z odgadnięciem, kto wykonuje utwór *ścieżka03.mp3*. Taką weryfikację można przeprowadzić, po prostu odtwarzając pliki w programie, takim jak np. Rhythmbox czy xmms, ponieważ one również automatycznie odczytują znaczniki ID3 zamieszczone w plikach.

Większość aplikacji używanych do zgrywania zawartości płyt CD audio może automatycznie wprowadzić odpowiednie informacje do plików podczas ich rzucania na dysk komputera, ale od czasu do czasu zdarza się, że dana płyta nie zostanie znaleziona w internetowych bazach i wówczas dane należy wprowadzić samodzielnie. W repozytoriach Ubuntu znajduje się wiele narzędzi przeznaczonych do aktualizowania i wprowadzania danych w standardzie ID3 i ID3v2, są to m.in. *id3*, *id3v2* czy *id3tool* (wszystkie są narzędziami linii poleceń). Dostępne są też *mp3info* — aplikacja terminalowa — oraz *kid3* — graficzny edytor przeznaczony dla środowiska KDE. Jeżeli użytkownik zdecyduje się na korzystanie z programu *kid3*, będzie musiał zainstalować biblioteki środowiska KDE, ale w zamian otrzyma wygodny program z graficznym interfejsem.

Korzystanie z iPod'a w Ubuntu

O komputerach Apple w niniejszej książce wystarczyłoby tylko przypis, gdyby nie system operacyjny Mac OS X oraz — co szczególnie ważne w kontekście niniejszego rozdziału — ogromny sukces cyfrowego odtwarzacza iPod oraz towarzyszącej mu aplikacji iTunes, czyli internetowego sklepu z cyfrową muzyką. iPod to świetny przykład tego, co może się stać, kiedy spotkają się ludzie rozumiejący projektowanie przemysłowe z tymi, którzy znają się na oprogramowaniu i użyteczności. W wielu domach iPod zastąpił samochód, czyli stał się sprzętem, o który się nieustannie dba, poleruje i wyposaża w coraz nowsze akcesoria.

Trzeba jednak dodać, że iPod ma też pewne minusy. Po pierwsze, wspiera wstrętne formaty muzyczne z systemem kontroli dostępu (DRM), zaprojektowane po to, aby uniemożliwić użytkownikom w pełni wolne korzystanie z muzyki, którą kupili¹. Po drugie, dysk tego odtwarzacza może być formatowany tylko za pomocą systemów plików HFS Macintosh lub Microsoft Windows. Systemy operacyjne Mac OS i Ubuntu mogą odczytywać oba te systemy plików, ale Microsoft Windows tylko ten drugi. Choć takie założenie z pewnością przyczynia się do problemów u części użytkowników, to jednak w ogólnym rozrachunku trudno polemizować z sukcesem odniesionym przez iPod'a. Kolejny powód jest taki, że program iTunes, czyli oprogramowanie z Apple używane do obsługi iPod'a, nie jest dostępne dla Linuksa. Jedna z ważnych w środowisku linuksowym osób powiedziała kiedyś: „Nie potrzebujemy śmierdzącego iTunes”. Jak użytkownik będzie mógł się przekonać, istnieje doskonała — i wolna — alternatywa dla iTunes, jest to program gtkpod, dzięki któremu z odtwarzaczem można zrobić prawie wszystko, na co pozwala program iTunes.

¹ Trzeba nadmienić, że w trakcie tłumaczenia tej książki pojawiły się informacje, które być może wieszczą koniec ery DRM; sam Steve Jobs, właściciel Apple, zaczął publicznie zastanawiać się nad rezygnacją z takich ograniczeń — *przyjp. thum*.



Chociaż gtkpod jest zamiennikiem iTunes w kwestiach synchronizowania urządzenia oraz tworzenia list odtwarzania, to jednak — co jest oczywiste — nie umożliwia dokonywania zakupów w internetowym sklepie muzycznym Apple. Muzyka oferowana w iTunes jest zabezpieczona mechanizmami DRM, co jest niezgodne z filozofią Ubuntu. Eksperymentalnie nabyłem kilka utworów w iTunes, ale ostatecznie pozostałem przy zgrzaniu płyt CD na dysk komputera, a później konwertowaniu ich z formatu OGG do MP3, tak aby można było je odtwarzać na iPodzie. W tej części rozdziału opowiem, jak przenosić muzykę, listy odtwarzania (także jak je tworzyć) i inne dane z iPod'a i z powrotem. Więcej informacji na temat korzystania z zabezpieczonych DRM plików mp4 w Linuksie i gtkpod można znaleźć na stronie <http://all-streaming-media.com/faq/recording-media-stream/> oraz w internecie, szukając haseł „Linux iTunes” czy „Linux gtkpod DRM”.

Być może użytkownik zastanawia się, dlaczego trzeba instalować osobny program gtkpod, kiedy to Rhythmbox pełni rolę głównego programu do zarządzania kolekcją muzyczną. Faktycznie Rhythmbox potrafi wykryć podłączony iPod oraz wyświetlić jego zawartość, natomiast wersja znajdująca się w Ubuntu 6.06 jeszcze nie obsługuje zarządzania plikami w iPodzie. W niedługim czasie Rhythmbox będzie umożliwiał zarówno odczytywanie z iPod'a, jak i zapisywanie na iPodzie, ale na razie najlepszym rozwiązaniem — moim zdaniem — jest używanie programu gtkpod. To stabilna aplikacja, która może też wystąpić w roli zamiennika Rhythmboksa, z całą pewnością warto poświęcić trochę czasu na jej naukę zwłaszcza wtedy, kiedy korzysta się też z innych dystrybucji Linuksa, w których nie ma Rhythmboksa.

Instalowanie programu gtkpod

Aby zainstalować gtkpod — program do obsługi iPod'a — należy uruchomić menedżer pakietów Synaptic (z menu *System/Administracja*), następnie kliknąć przycisk *Szukaj* i w polu tekstowym wyświetlonego okna wpisać ciąg *iPod*. Kiedy wyszukiwanie zostanie zakończone, trzeba odnaleźć na liście pakiet o nazwie *gtkpod* i zaznaczyć (wybrać odpowiednią pozycję z menu kontekstowego) go do instalacji. Na rysunku 17.20 przedstawiam ten etap instalacji.

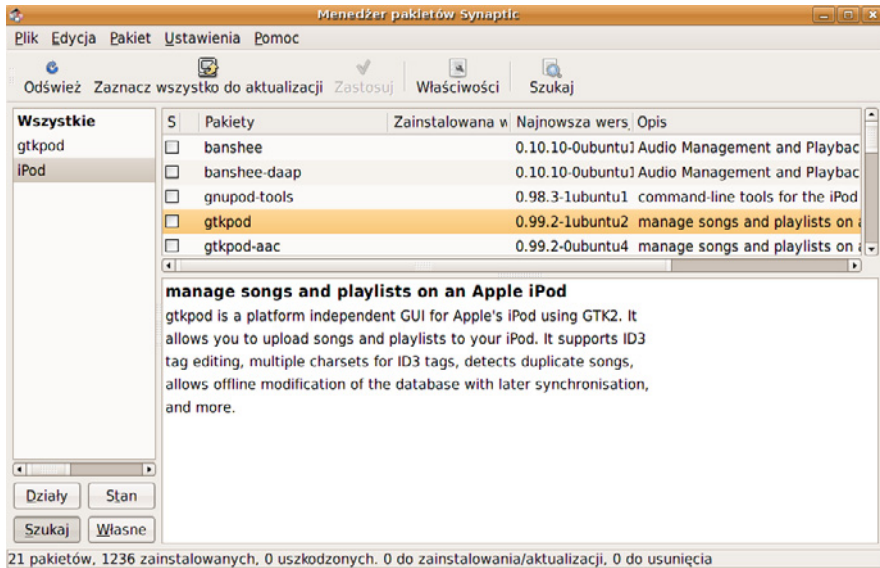


Użytkownicy, którzy zakupili w internetowym sklepie iTunes muzykę zabezpieczoną mechanizmami DRM, powinni zainstalować pakiet *gtkpod-aac* zamiast *gtkpod*. Pakiet ten umożliwia korzystanie z plików AAC zabezpieczonych DRM.

Zależnie od tego, jakie oprogramowanie zostało wcześniej zainstalowane w Ubuntu oraz wybrane w Synapticu, może zostać wyświetlone okno z dodatkowymi pakietami niezbędnymi do dodania do systemu. Jeżeli taki komunikat zostanie wyświetlony, użytkownik powinien zaakceptować proponowane zmiany.

Po wybraniu i zaznaczeniu wszystkich niezbędnych pakietów należy kliknąć przycisk *Zastosuj* znajdujący się na pasku narzędzi Synaptic. Po zakończeniu procesu instalacji system jest w pełni przygotowany na przyjęcie iPod'a.

W rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”, zamieszczam więcej informacji na temat korzystania z menedżera pakietów Synaptic.



Rysunek 17.20. Instalowanie programu *gtkpod*

Podłączanie iPod'a

Przed podłączeniem iPod'a poprzez port USB lub FireWire należy się upewnić, czy zostały zastosowane wszystkie wskazówki omówione we wcześniejszym punkcie rozdziału, „Konfigurowanie obsługi iPodów i innych odtwarzaczy muzycznych”. iPod po podłączeniu do Ubuntu zostanie rozpoznany jako urządzenie pamięci masowej, automatycznie zamontowany w katalogu `/media/ipod`, a na pulpicie zobaczymy odpowiednią ikonę.



Po podłączeniu iPod'a do Ubuntu na ekranie odtwarzacza zostanie wyświetlona ikona i komunikat, aby nie odłączać urządzenia. Należy zaufać tym ostrzeżeniom. W podpunkcie rozdziału pt. „Bezpieczne odłączanie iPod'a” podaję wskazówki, jak odłączyć odtwarzacz od komputera, nie narażając się na ryzyko uszkodzenia lub utraty danych.

Jeżeli iPod zostanie wykryty przez system, uruchomi się program Rhythmbox, umożliwiając odtworzenie muzyki znajdującej się w odtwarzaczu i — być może — również zapis plików. Jeżeli użytkownik zamierza do tworzenia list odtwarzania czy przenoszenia plików audio używać programu *gtkpod*, może wyłączyć program Rhythmbox. Jeżeli jednak chce tylko odtworzyć muzykę znajdującą się w iPodzie, wówczas może z powodzeniem wykorzystać Rhythmbox.

Jeżeli użytkownik zastosował się do wskazówek zamieszczonych w punkcie „Konfigurowanie obsługi iPodów i innych odtwarzaczy muzycznych”, a mimo to po podłączeniu odtwarzacza na pulpicie nie została wyświetlona odpowiednia ikona, wówczas warto sprawdzić rady dotyczące rozwiązywania problemów umieszczone w rozdziale 22., „Wykonywanie i przywracanie kopii zapasowych”. Jeżeli ikona została wyświetlona, ale nadal występują problemy z zapisywaniem plików w odtwarzaczu, wówczas, korzystając z konsoli, należy skontrolować utworzone węzły urządzenia. Najpierw za pomocą polecenia `df` i `ls -l` trzeba sprawdzić, gdzie urządzenie zostało zamontowane, tak jak w poniższym przykładzie:

```
$ df | grep ipod
/dev/sda2          1862560    546080    1316480    30% /media/ipod

$ ls -al /dev/sda2*
brw-rw---- 1 root plugdev 8, 0 2007-05-14 19:11 /dev/sda
brw-rw---- 1 root plugdev 8, 1 2007-05-14 19:11 /dev/sda1
brw-rw---- 1 root plugdev 8, 2 2007-05-14 19:11 /dev/sda2
```

Następnie należy sprawdzić, czy użytkownik należy do grupy *plugdev*, można do tego celu wykorzystać polecenie `groups`:

```
$ groups
wvh adm dialout cdrom floppy audio dip video plugdev lpadmin scanner admin
```

Jeżeli iPod został zamontowany w katalogu */media/ipod*, wówczas urządzenie należy do grupy *plugdev*. Jeżeli jej członkiem jest użytkownik i w ogóle wszystko działa, wówczas czytelnik może pominąć następne trzy akapity.

Jeżeli jednak iPod został zamontowany w jakimś innym punkcie lub należy do grupy (np. *disk*), której członkiem nie jest czytelnik, rozwiązaniem jest dodanie użytkownika. Członkowie grupy *disk* mają uprawnienia do odczytu i zapisu na standardowych urządzeniach dyskowych. Aby dodać wybranego użytkownika do tej grupy, należy użyć następującego polecenia (zapis *nazwa_użytkownika* należy zastąpić odpowiednią nazwą):

```
$ sudo adduser nazwa_użytkownika disk
```

Ten krok jest jednak potrzebny stosunkowo rzadko, a informacja o nim pojawiła się na skutek problemów, które kiedyś — jeden raz — napotkałem. Ta metoda rozwiąże problem, ale jest to rozwiązanie niemal ostateczne. Ponieważ umożliwia ono zapis i odczyt na wszystkich urządzeniach dyskowych, należy go używać ostrożnie. Więcej informacji o użytkownikach i grupach w Ubuntu można znaleźć w rozdziale 21., „Zarządzanie użytkownikami, grupami i zaawansowane uprawnienia”.

Korzystanie z programu gtkpod

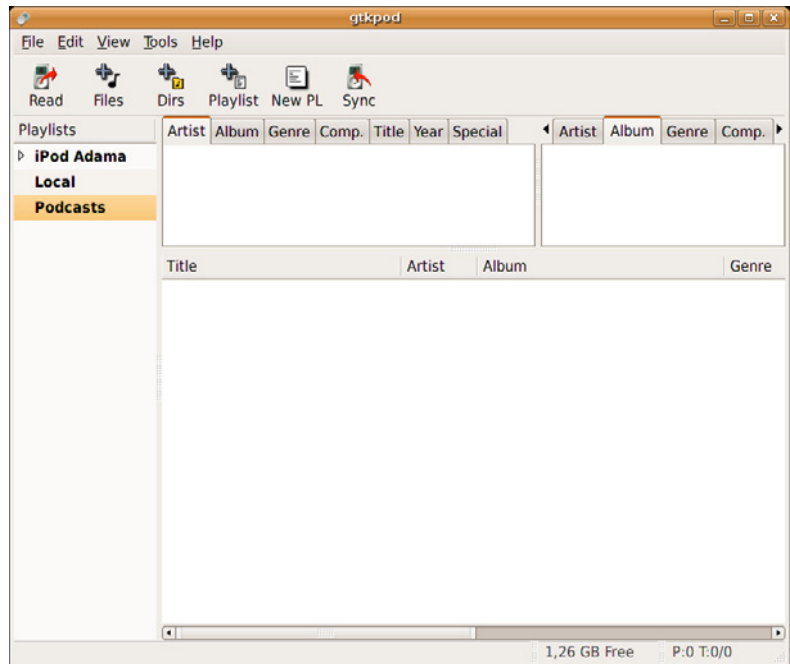
Po zainstalowaniu programu *gtkpod* w systemie do menu (*Aplikacje/Dźwięk i obraz*) zostanie dodana nowa pozycja umożliwiająca uruchomienie programu bez konieczności zapamiętywania jego nieco zagadkowej nazwy. Po uruchomieniu zostanie wyświetlone jego okno startowe (patrz rysunek 17.21).

Jeżeli iPod udało się podłączyć bez specjalnych kłopotów, nazwa urządzenia powinna być wyświetlana u góry panelu znajdującego się w lewej części okna. Aby przejrzeć zawartość urządzenia i zacząć z niego korzystać, należy zaznaczyć jego nazwę w lewym panelu, w innym oknie zostanie wówczas wyświetlona jego zawartość (patrz rysunek 17.22).

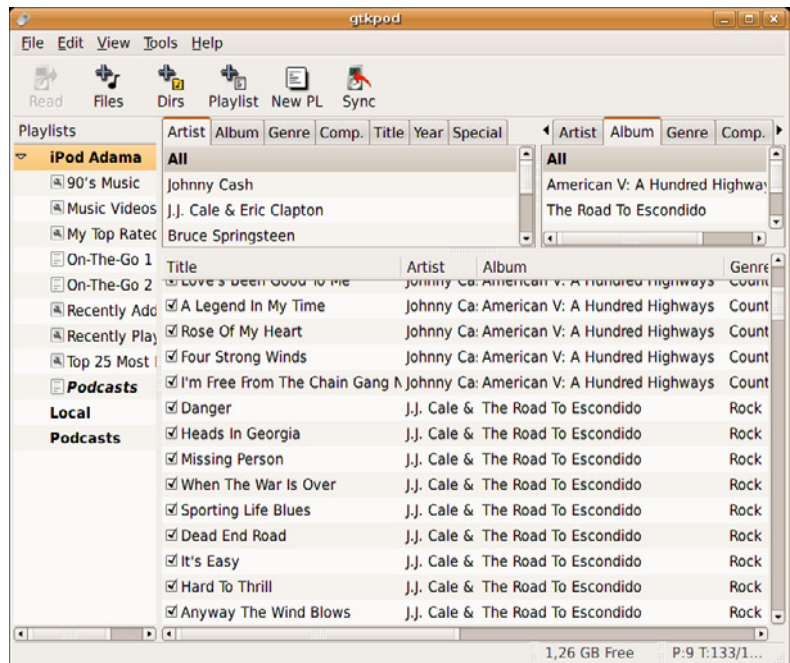
Jeżeli zawartość urządzenia nie zostanie wyświetlona w programie, należy wybrać z menu *Edit/Edit preferences*, a zobaczymy okno, takie jak na rysunku 17.23. Należy sprawdzić, czy wpis w polu *iPod Mount Point* (punkt montowania iPod'a) jest zgodny ze stanem faktyczny i w razie potrzeby skorygować wpis. Jeżeli zmiana zostanie wprowadzona, konieczne będzie ponowne uruchomienie programu. W celu odczytania zawartości urządzenia można też użyć przycisku *Read* dostępnego na pasku narzędzi programu.

Rysunek 17.21.

Ekran startowy programu gtkpod

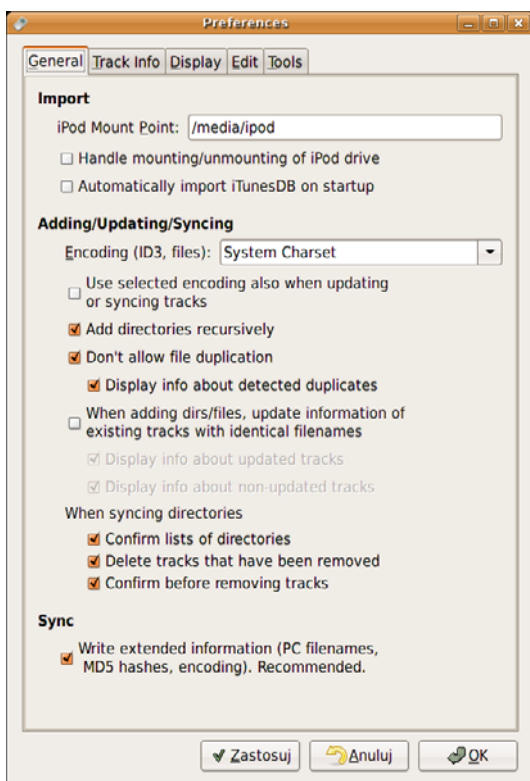
**Rysunek 17.22.**

Zawartość iPoda widoczna w programie gtkpod



W dalszej części rozdziału napiszę, jak wykonać podstawowe czynności związane z obsługą iPoda. Oczywiście, nie wyczerpują one tematu, o iTunes napisano wiele książek. Zamieszczone wskazówki mają jedynie pomóc czytelnikom rozpocząć przygodę z iPodem w Linuksie i objaśnić zasady działania programu gtkpod.

Rysunek 17.23.
*Sprawdzanie punktu
 montowania iPoda
 za pomocą programu
 gtkpod*



Dodawanie plików audio do iPoda

Pliki MP3 można dodawać do iPoda na dwa różne sposoby: jako pojedyncze pliki lub też jako foldery zawierające pliki MP3. Aby dodać pojedynczy plik, należy kliknąć przycisk *Files* znajdujący się na pasku narzędzi programu, a pojawi się okno dialogowe widoczne na rysunku 17.24.

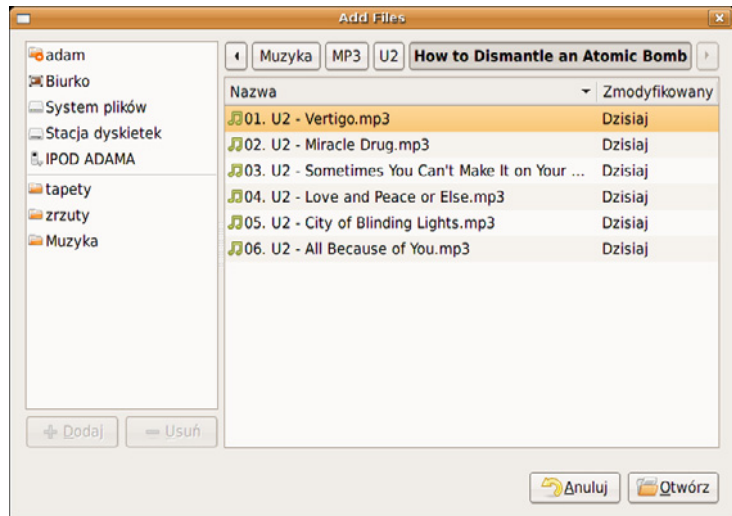
Kiedy okno zostanie wyświetlone, należy wskazać wybrany plik i kliknąć przycisk *Open*. Plik zostanie wówczas wgrany do odtwarzacza, a jeżeli znaczniki ID3 zawierają informacje o tytule i wykonawcy, one również będą dostępne (patrz rysunek 17.25).

Częściej jednak zdarza się, że użytkownik będzie chciał dodać wszystkie pliki z danego albumu, którego zawartość została zgrana na dysk komputera po ostatniej wizycie w sklepie muzycznym. Aby to zrobić, należy kliknąć przycisk *Dirs* znajdujący się na pasku narzędzi programu, zostanie wówczas wyświetlone okno widoczne na rysunku 17.26.

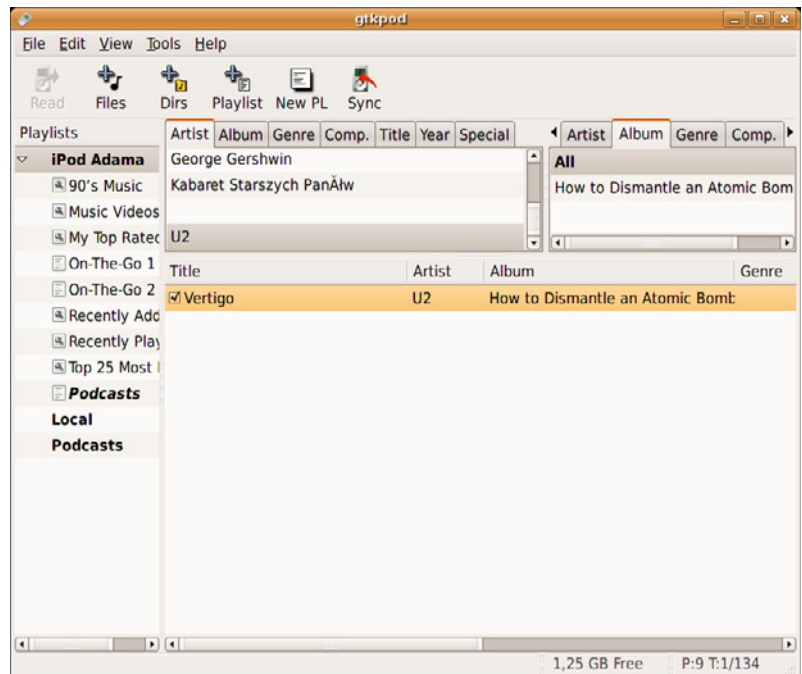
W nowo otwartym oknie trzeba przejść do odpowiedniego folderu, zaznaczyć go i kliknąć przycisk *OK*, a cała zawartość zostanie wgrana do odtwarzacza. Po zakończeniu tego procesu zawartość głównego okna programu gtkpod zostanie zaktualizowana (patrz rysunek 17.27).

Rysunek 17.24.

Dodawanie pojedynczych plików do iPod'a

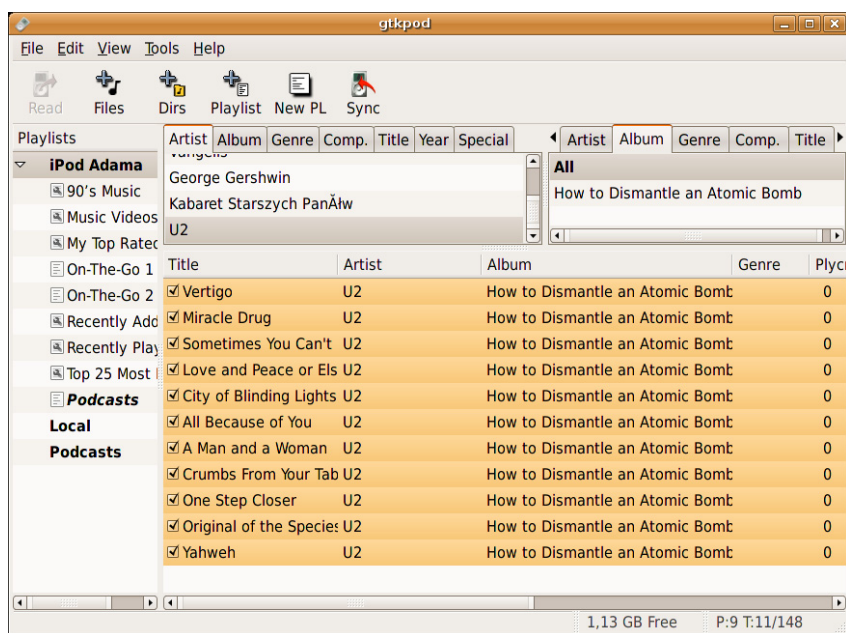
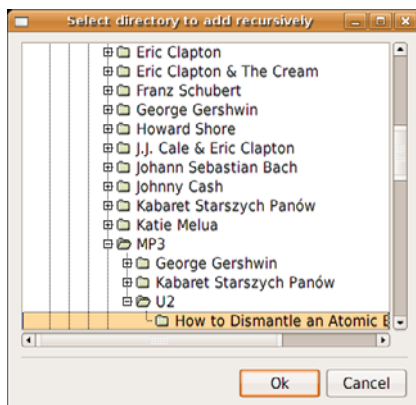
**Rysunek 17.25.**

Zaktualizowana baza po dodaniu nowego pliku



Chociaż w głównym oknie programu obok nazw plików wyświetlane są również nazwy katalogów, to do odtwarzacza zostały skopiowane tylko pliki. Baza danych używana w iPodzie, w której przechowywane są informacje na temat utworów i wykonawców albumów, zostanie zaktualizowana dopiero wówczas, kiedy użytkownik kliknie przycisk *Sync*, dla którego wyświetlana jest etykieta *Write Changes to iPod/Disk* (więcej informacji podam w dalszej części rozdziału).

Rysunek 17.26.
Dodawanie katalogów
z plikami MP3



Rysunek 17.27. Zaktualizowana zawartość odtwarzacza

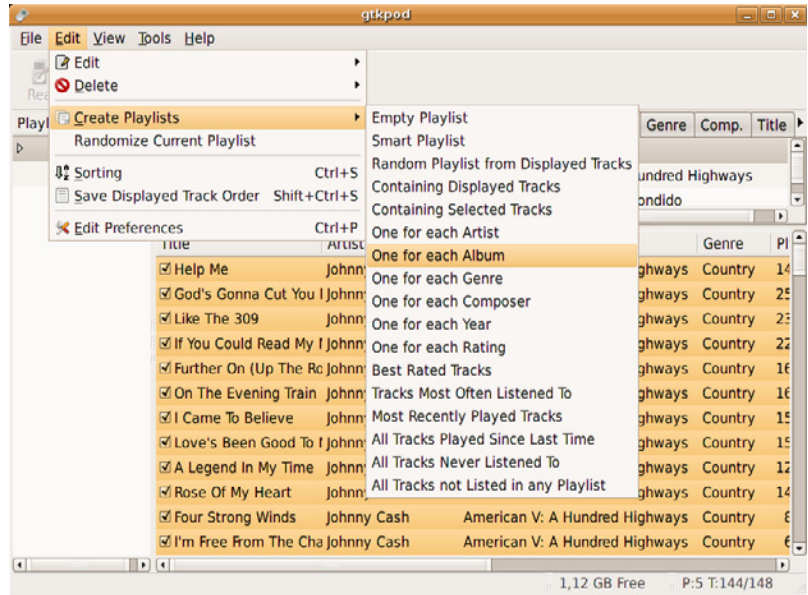
Tworzenie i edytowanie list odtwarzania

Jedną z kolejnych przyjemności płynących z posiadania iPod'a jest możliwość tworzenia własnych list odtwarzania zawierających utwory dobrane wg określonego klucza. Najprostsze jest zastosowanie ich dla utworów z danego albumu, dzięki czemu będzie można wysłuchać ich w układzie zaprojektowanym przez wykonawcę.

Program gtkpod umożliwia proste tworzenie list odtwarzania. Istnieje wiele różnych możliwości do utworzenia typów list odtwarzania, wszystkie je można sprawdzić, korzystając z poleceń menu *Edit/Create Playlists* (patrz rysunek 17.28).

Rysunek 17.28.

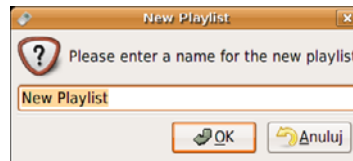
Typy list
odtworzenia
dostępne
w programie gtkpod



Aby np. utworzyć listę odtwarzania zawierającą wszystkie utwory z danego albumu, należy najpierw je zaznaczyć (można do tego celu użyć standardowego skrótu klawiszowego dostępnego w GNOME: *Ctrl+A*). Następnie trzeba wybrać z menu *Edit/Create Playlists/Containing Selected Tracks*. Zostanie wyświetlone okno dialogowe widoczne na rysunku 17.29, umożliwiające wprowadzenie własnej nazwy dla nowo tworzonej listy (domyślną proponowaną nazwą jest *New Playlist*, co jednak nie jest najłatwiejsze do zapamiętania).

Rysunek 17.29.

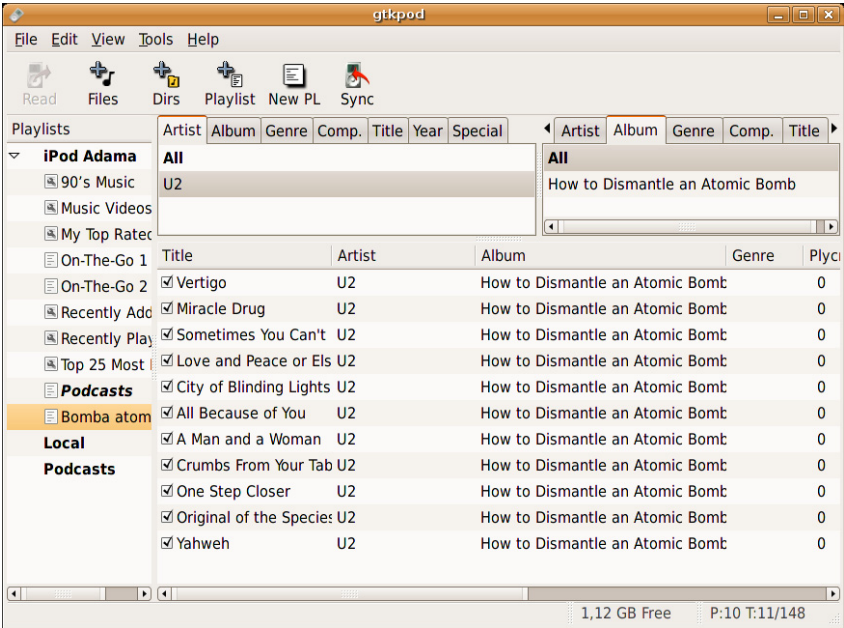
Wprowadzanie nazwy
dla nowej listy
odtworzenia



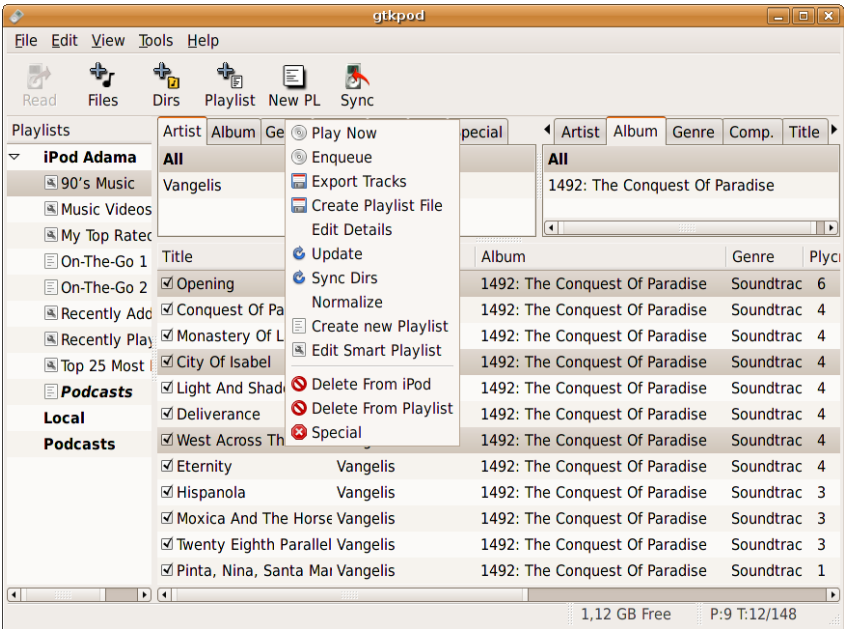
Jak widać na rysunku 17.28, w programie gtkpod można utworzyć naprawdę wiele różnorodnych list odtwarzania. Po utworzeniu nowej listy jej nazwa zostanie wyświetlona w lewym panelu programu. Po jej zaznaczeniu w oknie programu zostanie wyświetlona jej zawartość (patrz rysunek 17.30); w tym przypadku jest to lista o wdzięcznej nazwie *Bomba atomowa*.

W każdej chwili można zmienić nazwę listy odtwarzania, wystarczy tylko ją zaznaczyć i raz jeszcze kliknąć, a następnie wprowadzić nową nazwę. Aby usunąć daną listę, należy ją zaznaczyć, a następnie z menu wybrać kolejno *Edit/Delete/Selected Playlist*. Częściej jednak będzie się zdarzać, że użytkownik będzie chciał po prostu zmodyfikować zawartość listy. Aby usunąć wybrane utwory z listy lub wykonać inne działanie, należy zaznaczyć dany utwór, a następnie wywołać menu kontekstowe (patrz rysunek 17.31).

Aby usunąć dany utwór z listy odtwarzania bez usuwania pliku z iPod'a, należy wybrać polecenie *Delete from Playlist* (patrz rysunek 17.31). Aby dany utwór usunąć zarówno z listy, jak i z odtwarzacza, należy wybrać polecenie *Delete from iPod*.



Rysunek 17.30. Nowo utworzona lista odtwarzania



Rysunek 17.31. Menu kontekstowe w programie gtkpod

Istniejącymi listami odtwarzania również można zarządzać na kilka sposobów. Można np. sortować je, klikając dowolny z nagłówków widocznych w panelach programu gtkpod. Można też dodawać nowe utwory do list, po prostu przeciągając je z jednego panelu na

wybraną listę. Program oferuje zdumiewające możliwości zarządzania listami odtwarzania, tworzenia ich, tak jak chciałby tego użytkownik, kiedy by chciał, i w układzie, jakim chce.

Zapisywanie zmian w iPodzie

Kiedy już użytkownik doda nowe utwory do iPod'a, ułoży nowe listy odtwarzania itd., z pewnością będzie chciał zapisać wszystkie zmiany. Chociaż pliki, katalogi z plikami i inne elementy zostały wgrane do odtwarzacza natychmiast po wykonaniu tej operacji w iPodzie, to jednak baza danych, z której urządzenie korzysta podczas pracy, nie została jeszcze zaktualizowana o informacje umieszczone w programie gtkpod. Dlatego należy zsynchronizować urządzenie po wprowadzeniu jakichkolwiek zmian, co pozwoli uniknąć ryzyka utraty lub uszkodzenia informacji. Z nowych piosenek wgranych do iPod'a nie będzie wiele pożytku, jeżeli nie będzie można ich odtwarzać.

Aby wykonać synchronizację danych, trzeba użyć przycisku *Sync*, znajdującego się na pasku narzędzi programu gtkpod. Po zakończeniu procesu zostanie wyświetlone okno z informacją o jego przebiegu. Po zamknięciu tego okna można bezpiecznie odłączyć iPod od Ubuntu, co opisuję poniżej.

Bezpieczne odłączanie iPod'a

Zapisanie zmian w odtwarzaczu i zakończenie działania programu gtkpod nie oznacza jeszcze, że iPod został odłączony z Ubuntu. Podobnie jak wszystkie inne urządzenia pamięci, iPod jest zamontowany jako część systemu plików, co umożliwia dostęp do niego, tak jak do każdej innej części standardowej struktury katalogów. Po podłączeniu iPod domyślnie jest montowany w katalogu */media/ipod*. Choć urządzenie jest montowane jako część systemu plików Ubuntu, to na jego wyświetlaczu pojawia się ikona ze znakiem zakazu i komunikat *Nie odłączaj*. Aby bezpiecznie odłączyć iPod'a, należy najpierw go odmontować, co zagwarantuje, że dane na nim zostaną zaktualizowane, podobnie jak system plików.

W Linuksie dostępne jest polecenie *eject* służące do bezpiecznego odmontowania partycji dysków lub zamontowanych urządzeń. Dlatego przed odłączeniem iPod'a od komputera należy wykonać następujące polecenie:

```
$ sudo eject /media/ipod
```

Po wprowadzeniu hasła polecenie odmontuje system plików iPod'a, zostanie też usunięta ikona urządzenia z pulpitu, a z wyświetlacza odtwarzacza zniknie ostrzeżenie przed rozłączeniem. Po bezpiecznym odmontowaniu na iPodzie pojawi się standardowe menu, co oznacza, że teraz można odłączyć go od komputera.

Korzystanie z innych cyfrowych odtwarzaczy audio

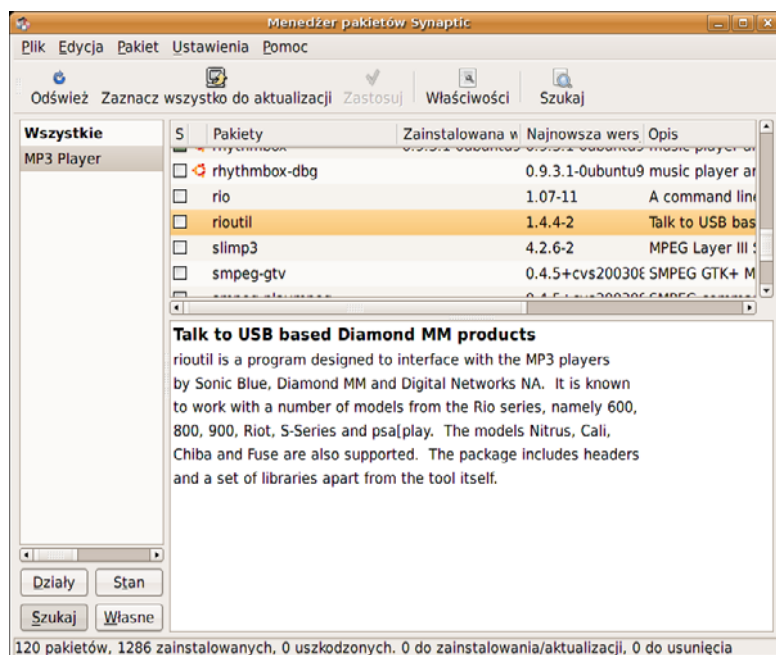
Podłączenie innych odtwarzaczy audio do Ubuntu przebiega tak samo jak podłączenie iPod'a. Po pierwsze, należy się upewnić, że wykonano wszystkie zalecenia znajdujące się w punkcie „Konfigurowanie obsługi iPodów i innych odtwarzaczy muzycznych”.

Podłączanie jakiegokolwiek cyfrowego odtwarzacza audio poprzez port USB lub FireWire spowoduje reakcję wywoływaną przez przenośne urządzenia pamięci masowej, a na pulpicie

zostanie automatycznie utworzona ikona. Jeżeli nie zostanie uruchomiona przeglądarka plików Nautilus z zawartością urządzenia, wówczas użytkownik może ją włączyć samodzielnie — wystarczy kliknąć dwukrotnie ikonę urządzenia.

W repozytoriach Ubuntu znajduje się wiele pakietów, których zadaniem jest uproszczenie pracy z różnymi rodzajami odtwarzaczy muzycznych. Jeżeli np. użytkownik korzysta z urządzeń firm SonicBlue, Diamond MultiMedia czy Digital Networks, wówczas może zainstalować i wykorzystywać narzędzia o nazwie *rio* lub *rioutil*. Można je szybko i łatwo zainstalować przy użyciu menedżera pakietów Synaptic (patrz rysunek 17.32).

Rysunek 17.32.
Narzędzia do obsługi
odtwarzaczy MP3
dostępne w Synapticu



Większość współczesnych odtwarzaczy audio umożliwia dodawanie nowych plików za pomocą operacji „przeciągnij i upuść”. Ale wykonanie krótkich poszukiwań w repozytoriach Ubuntu oraz użycie wyszukiwarki Google to z pewnością dobry pomysł. Niektóre odtwarzacze — tak jak iPod — stosują własną bazę danych o przechowywanych plikach i samo przeciągnięcie plików może nie wystarczyć do jej zaktualizowania.

Po zakończeniu pracy z danym odtwarzaczem należy go bezpiecznie odłączyć, czyli kliknąć prawym przyciskiem ikonę urządzenia znajdującą się na pulpicie i z menu kontekstowego wybrać opcję *Wysunąć*. Wtedy mamy pewność, że wszelkie wprowadzone zmiany zostaną zapisane, a urządzenie możemy bezpiecznie odłączyć od Ubuntu.

Obsługa kart CompactFlash i SD

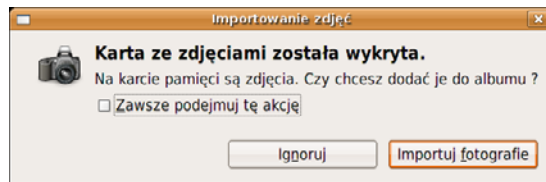
Chociaż Ubuntu potrafi rozpoznawać ogromną ilość różnorodnych aparatów cyfrowych, palmtopów i innych urządzeń, może przydarzyć się taka sytuacja, w której system nie będzie w stanie rozpoznać danego urządzenia lub nie będzie można go połączyć z kom-

puterem za pomocą portu USB lub FireWire. Na szczęście, w większości takich urządzeń dostępne są przenośne nośniki pamięci, takie jak karty CompactFlash czy SD, do których dostęp w Ubuntu można uzyskać przy użyciu specjalnego czytnika takich kart. Urządzenia te są obecnie dość powszechne i bez trudu można je nabyć na aukcjach internetowych czy w sklepach z akcesoriami fotograficznymi. Stosowanie ich w Ubuntu jest całkiem proste, tak więc można wykonać kopię zapasową cennych fotografii wakacyjnych czy innych plików, które użytkownik chciałby przechowywać na dysku komputera.

Po pierwsze, należy się upewnić, że zostały wprowadzone wszystkie wskazówki zamieszczone w punkcie „Konfigurowanie obsługi kart pamięci i innych urządzeń elektronicznych”, w tym rozdziale. Następnie nośnik pamięci trzeba wyjąć z danego urządzenia (aparatu cyfrowego, palmtopa itp) i umieścić w czytniku, który z kolei należy podłączyć do komputera. Jeżeli Ubuntu będzie w stanie rozpoznać kartę pamięci jako taką, która pochodzi z aparatu cyfrowego (innymi słowy, jako kartę pamięci zawierającą katalog o nazwie *DCIM*), zostanie wyświetlone okno dialogowe widoczne na rysunku 17.33.

Rysunek 17.33.

Obsługa kart pamięci
zawierających zdjęcia
cyfrowe



Jeżeli okno zostanie wyświetlone, a użytkownik kliknie przycisk *Importuj fotografie*, wówczas Ubuntu uruchomi program *gthumb*, który przeprowadzi proces importowania zdjęć opisany już wcześniej w podrozdziale „Aparaty cyfrowe i Ubuntu”. Można tam znaleźć więcej informacji na temat samego procesu. W niektórych przypadkach import może się nie powieść, ponieważ program nie wykryje danego typu aparatu, a tylko samą kartę pamięci. Wtedy można po prostu skopiować pliki za pomocą Nautilusa, *xterm*a czy konsoli GNOME.

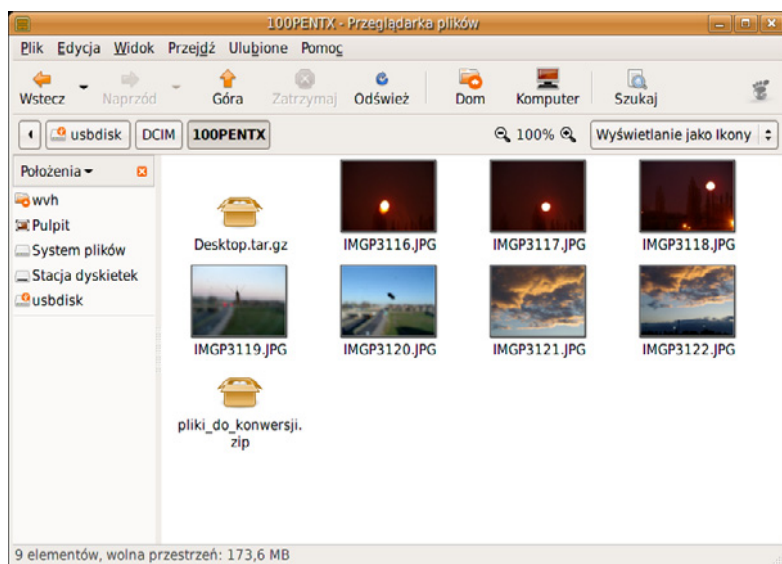
Niezależnie od zawartości karty pamięci, jeżeli użytkownik zastosował się do wskazówek zawartych w tym rozdziale, w punkcie „Konfigurowanie obsługi kart pamięci i innych urządzeń elektronicznych”, na pulpicie zostanie wyświetlona ikona symbolizująca podłączone urządzenie i zostanie uruchomiona przeglądarka plików Nautilus wyświetlająca zawartość danej karty (patrz rysunek 17.34).

Przy użyciu Nautilusa można łatwo i szybko skopiować wybrane pliki na dysk komputera (wskazówki jak to zrobić, zamieściłem w rozdziale 5.). Jeżeli Nautilus nie zostanie automatycznie uruchomiony, należy dwukrotnie kliknąć ikonę urządzenia wyświetlaną na pulpicie. Jeżeli przeglądana karta pamięci CompactFlash lub SD pochodzi z aparatu cyfrowego, wszystkie fotografie będą umieszczone w katalogu o nazwie *DCIM*.

Po zakończeniu pracy z kartą pamięci należy ją bezpiecznie usunąć z systemu, w tym celu trzeba zaznaczyć ikonę danego urządzenia i wybrać z menu kontekstowego opcję *Wysuń*. Spowoduje to odmontowanie karty pamięci z systemu plików Linuksa, chociaż sama karta nadal pozostanie w czytniku. Używając tego polecenia, można mieć pewność, że wszelkie dokonane zmiany zostaną zapisane na karcie pamięci, a ona sama zostanie bezpiecznie wyłączona i będzie można odłączyć urządzenie od komputera.

Rysunek 17.34.

Przeglądanie zawartości karty pamięci za pomocą przeglądarki plików Nautilus



Podsumowanie

Przenośne urządzenia elektroniczne, takie jak palmtopy, cyfrowe odtwarzacze muzyczne, aparaty cyfrowe i smartfony, to prawdziwe perełki współczesnej techniki. Niemal każdy posiada przynajmniej jedno takie z nich. W niniejszym rozdziale zamieściłem wiele informacji na ich temat; napisałem, jak synchronizować palmtopy z Ubuntu, jak pracować z iPodem w Ubuntu oraz jak obsługiwać aparaty cyfrowe wyposażone w karty pamięci CompactFlash i SD. Chociaż korzystam z Linuksa już od ponad dekady, to jakoś, wydajność i łatwość użytkowania tego systemu wciąż mnie zaskakuje. Mam nadzieję, że po przeczytaniu tego rozdziału czytelnicy będą podzielać moje zdanie.

W rozdziale 18. wejdę na tradycyjne terytorium Linuksa — kompilowanie programów w Ubuntu. Domyślnie w systemie nie ma narzędzi niezbędnych do tego celu, ponieważ nie wszyscy ich potrzebują, ale w repozytoriach znajduje się spory ich wybór. Możliwość tworzenia własnych wersji oprogramowania w Linuksie często jest podawana przez niechętnych mu użytkowników jako jego wada decydująca o tym, że jest to system tylko dla dziwaków. To nie jest prawda. Obecnie użytkownik może nie tylko tworzyć własne oprogramowanie, ale — co opisałem w rozdziale 18. — dzięki aplikacjom do zarządzania pakietami, takim jak apt-get, aptitude czy Synaptic, może też zainstalować narzędzia i wydajne zintegrowane środowiska programistyczne do tworzenia programów — jeśli ich potrzebuje.

Rozdział 18.

Tworzenie oprogramowania w Ubuntu

W tym rozdziale:

- ♦ Instalowanie i używanie GCC
- ♦ Wykorzystanie polecenie make
- ♦ Instalowanie i używanie zintegrowanego środowiska programistycznego Eclipse
- ♦ Instalowanie, konfigurowanie i używanie Subversion

Jak można przeczytać na stronie projektu, a także w niniejszej książce, Ubuntu to dystrybucja dla wszystkich. Dotyczy to też programistów! Fakt, że Ubuntu jest łatwe w użytkowaniu, nie oznacza jeszcze, że prawdziwi, zaawansowani programiści nie mogą używać go do pracy — jest wręcz przeciwnie. Najlepszy dowód na to jest taki, że dystrybucja Ubuntu została stworzona przez programistów Ubuntu w systemie Ubuntu. Łatwość użytkowania tego systemu oznacza również, że czas, który w innym przypadku należałoby poświęcić na konfigurowanie systemu i instalowanie oprogramowania, można przeznaczyć na bardziej produktywne działanie, np. pracę nad kodem tworzonych programów.

Wśród wielu użytkowników systemów, takich jak Microsoft Windows czy Mac OS X, Linux ma złą reputację, postrzegany jest jako system dla mieszkańców jaskiń korzystających z terminali VT100 i żyjących w linii poleceń. Chociaż można by wiele powiedzieć o linii poleceń, to jednak współczesne dystrybucje Linuksa, takie jak Ubuntu, zachowują doskonałą równowagę, oferując z jednej strony graficzne narzędzia, za których pomocą można zrobić prawie wszystko, a z drugiej dostęp do konsoli, kiedy tylko użytkownik tego zapragnie. Współczesny model Linuksa jest uderzająco podobny do modelu użytkownika oferowanego przez Mac OS X, a nowe wersje Microsoft Windows oferują „nowe” narzędzia wspomagające pracę w linii poleceń czy obsługę wyrafinowanych skryptów. Czy to czegoś nie przypomina?

W tym rozdziale opowiem, jak zainstalować pakiety niezbędne do tworzenia oprogramowania w Linuksie, rozpocznę od podstaw, czyli zainstalowania najlepszego na świecie kompilatora (gcc, oferowanego jako składnik *Free Software Foundation GNU Compiler Collection*, czyli GCC) i narzędzi z nim powiązanych, a niezbędnych do tworzenia oprogramowania w Ubuntu. Następnie omówię inne aplikacje cieszące się popularnością wśród wielu programistów. Należą do nich zintegrowane środowiska programistyczne (ang. *Integrated Development Enviroments*, IDE) oferujące łatwy w użyciu interfejs, narzędzia kontroli kodu oraz zarządzania oprogramowaniem, dzięki którym można łatwo śledzić zmiany wprowadzane w kodzie.

Instalowanie i używanie narzędzi programistycznych w Ubuntu

W Ubuntu domyślnie nie jest instalowane żadne środowisko programistyczne, ponieważ większości użytkowników nie jest on potrzebny. Poza tym, w Ubuntu dostępna jest tak duża ilość gotowego oprogramowania, że większość użytkowników nigdy nie odczuje potrzeby tworzenia aplikacji dla własnych potrzeb. Na pierwszym miejscu w tej dystrybucji stawia się oferowanie gotowego, prostego w instalacji i zintegrowanego z systemem oprogramowania.

Dla mnie w słowie *software* (oprogramowanie) najważniejsza jest pierwsza część, czyli *soft*. Niesie ona ze sobą wiele znaczeń, jedno z nich wskazuje, że oprogramowanie jest elastyczne. Różni użytkownicy potrzebują odmiennego oprogramowania do zaspokajania własnych potrzeb. Kiedy jednak użytkownik tworzy program mający wykonywać specyficzne zadania, których inni zdają się nie dostrzegać lub chce wykorzystać specjalne właściwości danego pakietu, być może będzie musiał samodzielnie skompilować oprogramowanie.

Tradycyjne środowisko deweloperskie w Linuksie składa się z kilku różnych pakietów, dostarczających kompilatory, linker, assembler, narzędzia do tworzenia i używania bibliotek, narzędzie upraszczające budowanie skomplikowanych pakietów i inne. Jak można się spodziewać, w Ubuntu wszystkie te komponenty można zainstalować przy użyciu jednego, głównego nadpakietu. Jak pokażę poniżej, zainstalowanie jednego pakietu pociąga za sobą instalację wszystkich podstawowych składników środowiska, które możesz następnie powiększyć o dowolne, ulubione narzędzia które poznałeś i pokochałeś tworząc oprogramowanie pod Linuksem.

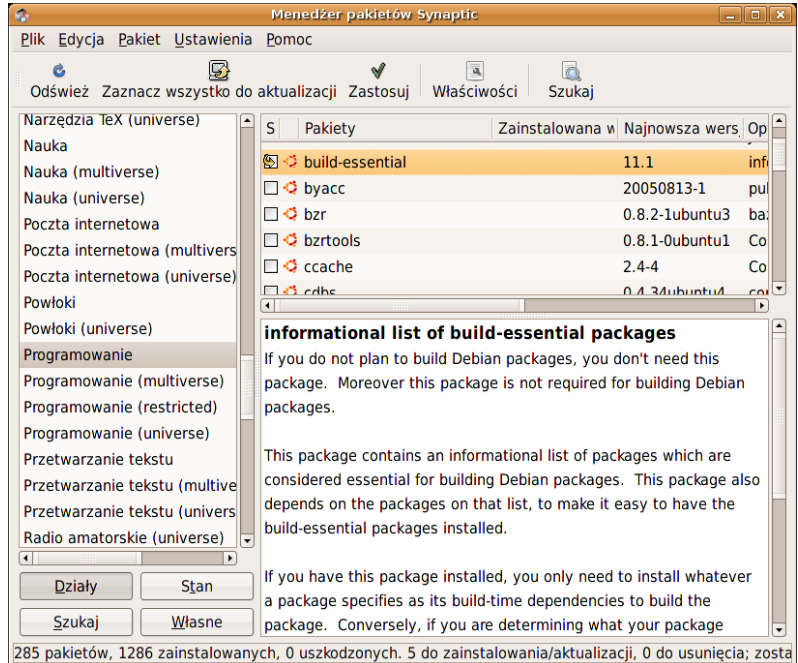
Instalowanie GCC, make i innych narzędzi

W Ubuntu zdecydowanie ułatwiono instalację kluczowych elementów niezbędnych do tworzenia programów, a to dzięki pakietowi *build-essential*, który dodaje do systemu kompilatory GCC C, C++, odpowiednie biblioteki i zestaw nagłówków jądra Linuksa. Aby zainstalować ten pakiet bez korzystania z narzędzi graficznych, należy użyć polecenia `apt-get install build-essential` lub `aptitude -r install build-essential`. Jeżeli użytkownik zdecyduje się wykorzystać powyższe polecenia, wówczas nie musi zapoznawać się z dalszymi wskazówkami na temat instalacji. Więcej informacji na temat używania

wymienionych poleceń można znaleźć w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Aby zainstalować opisany metapaket, używając narzędzi z interfejsem graficznym, należy uruchomić z menu *System/Administracja* menedżera pakietów Synaptic, następnie przejść do grupy *Programowanie* znajdującej się w lewym panelu, z kolei w prawym odszukać pakiet *build-essential*, zaznaczyć go prawym przyciskiem myszy i z menu kontekstowego wybrać opcję *Zaznacz do instalacji*. Na rysunku 18.1 przedstawiam okno Synaptica z zaznaczonym do instalacji pakietem *build-essential*.

Rysunek 18.1.
*Instalowanie pakietu
build-essential*



Zależnie od zainstalowanego wcześniej w systemie oprogramowania oraz pakietów wybranych w Synapticu, menedżer może wyświetlić kolejne okno dialogowe z dodatkowymi pakietami, które muszą być zainstalowane w systemie; należy zaakceptować tę propozycję i kliknąć przycisk *Zaznacz*.

Po zaznaczeniu wszystkich potrzebnych pakietów należy kliknąć znajdujący się na pasku narzędzi przycisk *Zastosuj*, Synaptic pobierze i zainstaluje w systemie narzędzia *gcc*, *g++*, *make* i inne. Kiedy instalacja zostanie zakończona, można rozpocząć samodzielne kompilowanie programów.

W zależności od wymagań oprogramowania czy pakietów, które chce użytkownik kompilować, konieczna może być instalacja również innych, dodatkowych pakietów. I tak np. narzędzia konfiguracyjne, takie jak *autoconf* czy *automake*, nie znajdują się w pakiecie *build-essential*, który zawiera tylko kompilatory *GCC C* i *C++*, dlatego aby kompilować programy napisane w innych językach, należy osobno zainstalować, za pomocą Synaptica czy innych menedżerów, odpowiednie pakiety i niezbędne biblioteki.

Pakiety źródłowe

W niniejszej książce informacje na temat oprogramowania podawane są przy założeniu, że użytkownik będzie korzystał z pakietów binarnych, które zawierają programy wykonywalne, niezbędne i skompilowane już biblioteki oraz inne dane konfiguracyjne zarówno w postaci binarnej, jak i tekstowej. Ponieważ w tym rozdziale omawiam zagadnienia związane z tworzeniem oprogramowania, wydaje się rozsądnym umieszczenie wzmianki o tworzeniu oprogramowania z pakietów źródłowych. Krótko mówiąc, istnieją dwa główne etapy, oba znacząco ułatwione dzięki zastosowaniu narzędzia linii poleceń `apt-get`:

1. pozyskiwanie kodu dla pakietów, które użytkownik chce skompilować,
2. spełnianie zależności dla tego oprogramowania, bibliotek itp.

Aby pobrać kod źródłowy dla jakiegokolwiek pakietu Ubuntu, należy użyć polecenia `apt-get source pakiet`, gdzie pakiet to nazwa pakietu, który ma zostać zbudowany ze źródeł. Aby zainstalować inne narzędzia, biblioteki, nagłówki i inne elementy niezbędne do kompilowania programów ze źródeł, należy użyć polecenia `apt-get build-dep pakiet`, gdzie pakiet to nazwa danego pakietu, który chcemy zbudować.

Więcej informacji na temat spełniania zależności oraz pobierania kodów źródłowych za pomocą narzędzia `apt-get` można znaleźć w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Więcej informacji na temat instalowania oprogramowania za pomocą menedżera pakietów Synaptic i innych narzędzi dostępnych w Ubuntu można znaleźć w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Kompilator GCC — podstawowe informacje

Skrót GCC zapisany dużymi literami oznacza *GNU Compiler Collection*, znany też jako *GNU Compiler Suite*. Jeżeli natomiast zostaną zastosowane małe litery, wówczas skrót `gcc` odnosi się do konkretnego kompilatora z powyższego pakietu, mianowicie GNU C. GCC to kolekcja kompilatorów dla różnych języków, obejmuje ona: C (`gcc`), C++ (`g++`), obiektowy C (`gcc` z dodatkowymi bibliotekami), Fortran (`gfortran`) i Javę (`gcj`). Tam też znajdują się kompilatory typu *front-end* dla języków Pascal, Modula-3 i Ada-9x. Z kolei kompilator języka C potrafi obsłużyć kilka jego odmian, w tym tradycyjną oraz ANSI C. Kompilator C++ jest natywnym kompilatorem, co oznacza, że nie konwertuje kodu C++ do pośredniej postaci C przed właściwą kompilacją, jak miało to miejsce w starszych kompilatorach.

GCC to projekt GNU kierowany bezpośrednio przez *Free Software Foundation* (FSF). Fundacja sprawuje pieczę nad prawami autorskimi oraz licencją GPL. Inne komponenty, takie jak biblioteki uruchomieniowe czy środowiska testowe, publikowane są na wielu różnych licencjach. FSF zajmuje się też prawnymi aspektami projektu GCC.

Używanie kompilatora GCC C

Do kontrolowania działania kompilatora `gcc` służy ogromna ilość opcji linii poleceń. Można stosować zarówno opcje składające się z jednej litery (np. `-o`), jak również wieloliterowe ciągi (np. `-ansi`). Ze względu na to nie można łączyć opcji jednoliterowych,

jak ma to miejsce w wielu innych narzędziach linii poleceń (w przypadku poleceń takich jak `ps` czy `ls`). I tak opcja `-pg` nie jest tożsama z opcjami `-p` i `-g`. Opcja `-pg` tworzy dodatkowy kod w pliku wynikowym, który dostarcza danych potrzebnych do profilowania przy pomocy narzędzia GNU `gprof`. Z drugiej strony opcje `-p` i `-g` powodują umieszczenie w pliku binarnym dodatkowego kodu zawierającego informacje użyteczne dla programu `prof` (`-p`) oraz dodają informacje wymagane do debugowania zapisane w formacie systemowym (`-g`).

Mimo że zwracam tu uwagę na używanie opcji wielo- i jednoliterowych, użytkownik może swobodnie zmieniać ich układ. Oto przykład dwóch poleceń, których efekt działania będzie taki sam:

```
$ gcc -pg -fno-strength-reduce -g halo.c -o halo
$ gcc halo.c -o halo -g -fno-strength-reduce -pg
```

Zwykle kolejność argumentów i opcji nie jest ważna. W niektórych sytuacjach — dotyczy to zwłaszcza użycia opcji tego samego rodzaju — ma to znaczenie. Tak jest w przypadku opcji `-I`, określającej katalog lub katalogi w których kompilator będzie szukał plików nagłówkowych. Jeżeli zostanie ona użyta kilka razy, wówczas kompilator `gcc` przeszuka wymienione katalogi w określonym porządku.

Skompilowanie pojedynczego pliku źródłowego (w tym przykładzie jest to *halo.c*) przy pomocy `gcc` wymaga podania jako argument jego nazwy:

```
$ gcc halo.c
```

Domyślnie plik wyjściowy będzie miał nazwę *a.out* i zostanie zapisany w bieżącym katalogu. Aby go wykonać, należy użyć polecenia `./a.out`. Żeby określić nazwę pliku wyjściowego, trzeba skorzystać z opcji `-o`, jak w poniższym przykładzie:

```
$ gcc halo.c -o uruchomnie
```

Jeżeli użytkownik kompiluje wiele plików źródłowych, może je wszystkie podać w linii poleceń, efektem będzie jeden plik o wskazanej przez użytkownika nazwie (w tym przykładzie jest to plik *pokazdate*):

```
$ gcc pokazdate.c pomoc.c -o pokazdate
```

Jeśli użytkownik chce kompilować pliki przyrostowo i połączyć je w plik binarny, może użyć opcji `-c`, aby zatrzymać kompilację po utworzeniu pliku obiektowego. Następnie można skorzystać z polecenia kompilatora `gcc` powodującego połączenie plików obiektowych w jeden wynikowy, o określonej nazwie:

```
$ gcc -c pokazdate.c
$ gcc -c pomoc.c
$ gcc pokazdate.o pomoc.o -o pokazdate
```

Kompilowanie w linii poleceń projektu składającego się z większej liczby plików z kodem źródłowym szybko stanie się niezmiernie żmudne zwłaszcza, kiedy trzeba będzie łączyć go z zewnętrznymi bibliotekami, dodawać polecenia optymalizujące lub inne opcje kompilatora `gcc`. W uniknięciu nudy linii poleceń pomaga narzędzie `make`, o którym piszę poniżej.

Tworzenie prostych plików makefile

Od samych początków Uniksa większość aplikacji dla tego systemu była tworzona przy pomocy narzędzia `make`, które wywoływało kompilatory i inne narzędzia niezbędne w procesie tworzenia programu z plików źródłowych. Program `make` jest podwaliną współczesnego stylu tworzenia i budowania aplikacji dla Linuksa i Uniksa napisanych w językach C i C++. Oto jego główne zadania:

- ♦ umożliwia programistom identyfikowanie związków i zależności pomiędzy modułami źródłowymi, plikami nagłówkowymi i bibliotekami niezbędnymi do udanej kompilacji,
- ♦ określa kolejność kompilowania plików, dzięki czemu cały proces może zakończyć się pomyślnie,
- ♦ pozwala uniknąć rekompilacji ograniczając kompilację do fragmentów aplikacji, w których doszło do zmiany kodu źródłowego lub statycznie linkowanych bibliotek.

Program `make` został napisany przez Stu Feldmana w firmie AT&T dla wczesnej wersji Uniksa. Jako część Uniksa AT&T program nie był wolno dostępny, dlatego z czasem został zastąpiony przez program GNU `make`. GNU `make` posiada wszystkie opcje oryginalnego programu `make` oraz wiele dodatkowych. W dalszej części rozdziału program `make` oznaczać będzie właśnie wersję GNU `make`.

Reguły stosowane przez program `make` są przechowywane w pliku tekstowym o nazwie *Makefile*, który zawiera informację na temat zależności oraz poleceń niezbędnych do ich spełnienia. Jak można się spodziewać w przypadku złożonych aplikacji, plik *Makefile* może być bardzo skomplikowany, ponieważ musi zawierać odwołania do bibliotek i narzędzi dla danej platformy, które z kolei mogą być umieszczone w różnych katalogach i mieć różne nazwy zależnie od systemu. Najprostszym możliwym plikiem *Makefile* jest taki, który dotyczy prostej aplikacji kompilowanej z jednego pliku źródłowego, jak ma to miejsce w legendarnym programie „Halo świecie” (ang. „Hello! World”). Plik *Makefile* dla takiej aplikacji wygląda następująco:

```
halo: halo.c
gcc halo.c -o halo
```

W pierwszej linii wskazuje się binarny plik wyjściowy (*halo*), który ma zostać utworzony, a także fakt, że plik musi być skompilowany ponownie, jeżeli plik źródłowy (*halo.c*) został zmieniony. Druga linia (rozpoczynająca się od tabulatora) określa polecenie niezbędne do utworzenia pliku binarnego. Aby przywołać plik *Makefile*, należy po prostu wykonać polecenie `make` w katalogu, w którym znajduje się plik *Makefile*. Program `make` automatycznie odszuka plik *Makefile* i zastosuje się do reguł w nim zawartych.



Fakt, że wszystkie linie zawierające reguły muszą rozpoczynać się od tabulatora, jest jednym z najczęstszych problemów podczas samodzielnego tworzenia pliku *Makefile*.

W powyższym przykładzie widać tylko podstawowe możliwości oferowane przez plik *Makefile*, ponieważ ze względu na jego trywialność nie sposób pokazać rzeczywistych korzyści płynących z używania tej metody do upraszczania i automatyzowania procesu

kompilacji i rekompilacji. Oto kolejny przykład. Powyżej omówiłem sytuację, w której kompilowano plik binarny z dwóch plików:

```
$ gcc -c pokazdate.c
$ gcc -c pomoc.c
$ gcc pokazdate.o pomoc.o -o pokazdate
```

Plik *Makefile* dla tej aplikacji wyglądałby następująco:

```
pokazdate: pokazdate.o pomoc.o
    gcc pokazdate.o pomoc.o -o pokazdate

pokazdate.o: pokazdate.c
    gcc -c pokazdate.c
pomoc.o: pomoc.c
    gcc -c pomoc.c
```

Powyższy przykład ilustruje, jak plik *Makefile* może zredukować ilość rekompilacji, które trzeba wykonać po dokonaniu zmian w części plików związanych z projektem. W przykładzie widzimy, że plik binarny *pokazdate* zależy od dwóch plików obiektowych: *pokazdate.o* i *pomoc.o*, a także zawiera regułę tworzenia pliku binarnego poprzez zlinkowanie obu tych plików. W dalszej części można znaleźć informację, że każdy z tych plików jest zależny od plików źródłowych napisanych w języku C oraz regułę ponownego tworzenia każdego z plików.

Wraz ze wzrostem stopnia skomplikowania projektów użytkownik będzie chciał wykorzystać zalety pliku *Makefile*, takie jak definiowanie i używanie zmiennych, korzystanie z symboli do upraszczania reguł itp. Oto kilka przydatnych źródeł informacji na temat tworzenia i korzystania z plików *Makefile*:

- ♦ http://linuxdevcenter.com/pub/a/linux/2002/01/31/make_intro.html — artykuł na temat używania polecenia *make* zamieszczony na stronie O'Reilly & Associates Linux Development Center,
- ♦ <http://tldp.org/HOTO/Software-Building-HOTO-3.html> — artykuł zawierający omówienie polecenia *make* zamieszczony na stronach projektu Linux Documentation,
- ♦ <http://www.gnu.org/software/make/> — strona domowa programu GNU *make*, znajdują się na niej m.in. łącza do oficjalnej dokumentacji,
- ♦ <http://www.gnu.org/software/make/manual/> — oficjalna strona z dokumentacją programu GNU *make* dostępną w różnych formatach.

Korzystanie ze zintegrowanych środowisk programistycznych

Zintegrowane środowisko programistyczne (IDE) to ujednolicony zestaw narzędzi zawierający zwykle edytor, kompilator, środowisko do budowania aplikacji oraz debugger działające razem w ramach jednego pakietu. IDE oferuje zwykle też infrastrukturę do tworzenia projektów programistycznych tworzonych z wielu plików źródłowych, innych plików oraz bibliotek i zarządzania nimi. Większość współczesnych IDE to graficzne

narzędzia działające w systemach, takich jak X Window, KDE, GNOME, Microsoft Windows czy Mac OS X. Z tego powodu wiele używanych obecnie IDE wyposażonych jest w narzędzia do tworzenia interfejsu użytkownika i innych graficznych elementów programów.

W zintegrowanych środowiskach programistycznych kluczowym elementem jest integracja ich elementów składowych. Większość IDE sprawia wrażenie jakby były jednym, dużym narzędziem, bez względu na ilość oferowanych funkcji.

Zgodnie z informacjami zamieszczonymi w dalszej części niniejszego rozdziału, zintegrowane środowiska programistyczne zwykle ukierunkowane są na programistów posługujących się określonym językiem czy zestawem języków programowania. W ostatnim czasie coraz większą popularność zdobywają środowiska, takie jak Eclipse, ponieważ oferują ustandaryzowaną aplikację umożliwiającą programowanie niezależnie od języka, którym posługuje się programista, a także pozwalającą na poszerzanie możliwości IDE przy pomocy systemu wtyczek.

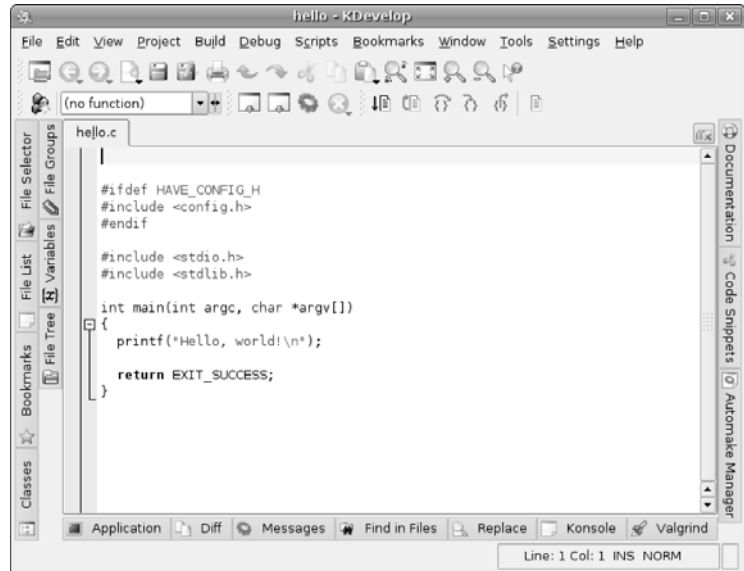
Popularne IDE dla Linuksa

Jak można się spodziewać, Linux charakteryzuje się różnorodnością rozwiązań, a to dlatego, że jest systemem operacyjnym używanym przez programistów aplikacji tworzonych w ogromnej ilości języków, a każdy programista posiada własne poglądy na temat tego, jak powinno wyglądać pisanie programów. Ubuntu oferuje spory wybór różnorodnych IDE przeznaczonych dla różnych języków programowania. Oto kilka bardziej znanych aplikacji dostępnych w repozytoriach.

- ♦ **Eclipse** — to napisane w Javie IDE posiadające wiele zastosowań. Początkowo Eclipse było napisane jako rama dla tworzenia aplikacji w Javie, ale wkrótce jego możliwości zostały rozszerzone o wsparcie dla innych języków programowania. Ponieważ Eclipse zostało napisane w Javie, można z niego korzystać w wielu różnych środowiskach graficznych i systemach operacyjnych. Środowisko to omówię w dalszej części rozdziału, dodatkowe informacje można znaleźć na stronie domowej projektu www.eclipse.org.
- ♦ **KDevelop3** — graficzne IDE dla języków C i C++ przeznaczone dla środowiska KDE. KDevelop jest bardzo popularny i łatwy w użyciu, a to za sprawą licznych kreatorów upraszczających tworzenie standardowych programów linii poleceń i aplikacji graficznych korzystających z bibliotek Qt, KDE i GNOME dla Linuksa. Więcej informacji na temat KDevelop można znaleźć na stronie internetowej projektu www.kdevelop.org. Na rysunku 18.2 przedstawiam program „Witaj świecie!” napisany w języku C w środowisku KDevelop.
- ♦ **Motor** — to tekstowe środowisko programistyczne przeznaczone dla języków C, C++ i Java, dobrze sprawujące się w aplikacjach terminala, takich jak xterm czy GNOME Terminal. Środowisko to oferuje obsługę różnych kompilatorów GCC oraz debuggera GNU — gdb. Więcej informacji na temat tej aplikacji można znaleźć na stronie domowej projektu <http://thekonst.net/motor>.
- ♦ **Pida** — czyli *Python Integrated Development Application* — to oparte na GTK środowisko programistyczne do tworzenia graficznych lub tekstowych aplikacji

Rysunek 18.2.

Przykładowa aplikacja napisana w języku C w środowisku KDevelop



za pomocą skryptowego języka Python. Pida cieszy się sporą popularnością i pozwala na łatwą integrację popularnych narzędzi programistycznych. Więcej informacji można znaleźć na stronie domowej projektu <http://pida.co.uk/>.

- ♦ **Sbcl** — czyli *Steel Bank Common Lisp* — to graficzne środowisko programistyczne do tworzenia aplikacji w języku Lisp. Początkowo Sbcl bazowało na środowisku Common Lisp napisanym w Carnegie Mellon University CMU Common Lisp, jednak obecnie jest aktywnie rozwijane i ulepszone. Sbcl oferuje natywny kompilator kodu języka Lisp i wyrafinowany debugger. Więcej informacji można znaleźć na stronie <http://sbcl.sourceforge.net/>.
- ♦ **Free-java-sdk** — to IDE stworzone przez połączenie najlepszych dostępnych narzędzi jako alternatywa dla środowiska Java Development Kit oferowanego przez firmę Sun Microsystems. Pakiet *free-java-sdk* został skompletowany z różnych projektów otwartego oprogramowania przez programistów Debiana, a następnie włączony do Ubuntu.

Powyższa lista to tylko fragment zawartości repozytoriów, ale ilustruje różnorodność dostępnych rozwiązań oraz fakt, że wiele IDE skupia się na określonych językach programowania. Ulubionym IDE autora jest Eclipse, pierwotnie zaprojektowane przez IBM jako łatwy w rozbudowie szkielet dla programistów Javy. Po uwolnieniu oprogramowania przez IBM zanotowano gwałtowny wzrost jego popularności. Obecnie Eclipse jest wydawane i rozwijane przez Fundację Eclipse (www.eclipse.org), organizację pożytku publicznego zajmującą się promowaniem tego środowiska oraz niesieniem pomocy wszystkim, którzy chcą z niego korzystać. Zastosowanie modelu opartego na wtyczkach pozwoliło programistom na swobodę wyboru i integracji wtyczek pochodzących z różnych projektów i dostawców komercyjnych, dzięki czemu mogą dokładnie dopasowywać które robi to, co robić powinno. Eclipse jest obecnie dostępne dla Linuksa, Uniksa, HP-UX, AIX, Solarisa, QNX, Mac OS X i Microsoft Windows.

Kolejną część rozdziału poświęcę zagadnieniom związanym z instalowaniem i używaniem środowiska programistycznego Eclipse.

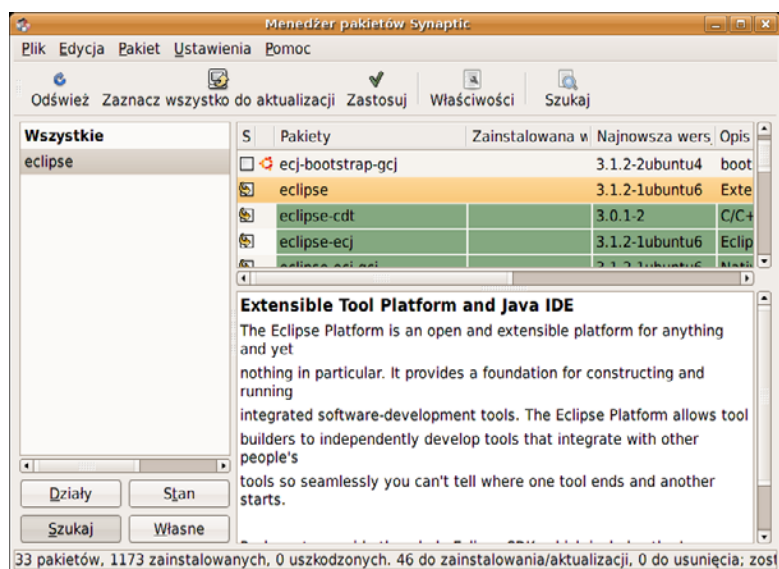
Instalowanie środowiska Eclipse

Aby zainstalować Eclipse i powiązane z nim pakiety, należy uruchomić menedżera pakietów Synaptic z menu *System/Administracja*, wybrać *Szukaj*, następnie wpisać ciąg `eclipse` i raz jeszcze kliknąć przycisk *Szukaj*. Kiedy wyszukiwanie zostanie zakończone, trzeba przewinąć listę znalezionych pakietów i zaznaczyć odpowiednie. Nie wszystkie z nich są konieczne, ale warto je zainstalować, oto wybór, jaki sugeruję.

- ♦ *eclipse* — (wymagane) to główny pakiet środowiska, dostarcza wszystkiego, co jest niezbędne, aby używać standardowego środowiska Eclipse do programowania aplikacji w Javie.
- ♦ *eclipse-cdt* — (*C Developers Toolkit*, opcjonalne) zbiór narzędzi przeznaczonych dla języka C. Pakiet jest konieczny, jeżeli za pomocą Eclipse mają być tworzone aplikacje w C lub C++. W pakiecie zawarto wtyczki, kreatory i inne elementy ułatwiające programowanie i debugowanie aplikacji napisanych w C i C++ w środowisku Eclipse.
- ♦ *eclipse-ecj-gcj* — (opcjonalne), kompilator GNU gcj Javy dla Eclipse, który jest częścią zbioru kompilatorów GCC.
- ♦ *eclipse-pydev* — wtyczka dla języka Python, pakiet jest niezbędny tylko wówczas, gdy użytkownik będzie korzystał z tego języka.

Należy zaznaczyć odpowiednie pakiety i wybrać z menu kontekstowego opcję *Zaznacz do instalacji*. Na rysunku 18.3 przedstawiam ten etap instalacji.

Rysunek 18.3.
Instalowanie Eclipse i powiązanych pakietów



Zależnie od zainstalowanego wcześniej w systemie oprogramowania oraz pakietów wybranych w Synapticu, menedżer może wyświetlić kolejne okno dialogowe z dodatkowymi pakietami niezbędnymi do umieszczenia w systemie; należy zaakceptować tę propozycję i kliknąć przycisk *Zaznacz*.

Po zaznaczeniu wszystkich potrzebnych pakietów trzeba kliknąć przycisk *Zastosuj* znajdujący się na pasku narzędzi Synaptic, a Eclipse i pakiety towarzyszące zostaną zainstalowane w systemie. Po zakończeniu procesu instalacji można rozpocząć tworzenie aplikacji, korzystając z najpopularniejszej platformy IDE.

Więcej informacji na temat instalowania oprogramowania za pomocą menedżera pakietów Synaptic i innych narzędzi dostępnych w Ubuntu można znaleźć w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Używanie Eclipse do tworzenia aplikacji w języku C

Poniżej zamieszczam dwa przykłady użycia Eclipse: w pierwszym pokazuję, jak utworzyć prostą aplikację od podstaw, w drugim natomiast — jak zaimportować kod źródłowy istniejącej aplikacji do środowiska Eclipse, dzięki czemu będzie można skorzystać z oferowanych przez to IDE możliwości kompilowania, debugowania i edycji. Dostępne są całe książki omawiające korzystanie z Eclipse. Trudno więc traktować niniejszy wstęp jako ich ekwiwalent, jednak po jego przeczytaniu użytkownik powinien zorientować się w możliwościach i łatwości obsługi środowiska Eclipse.



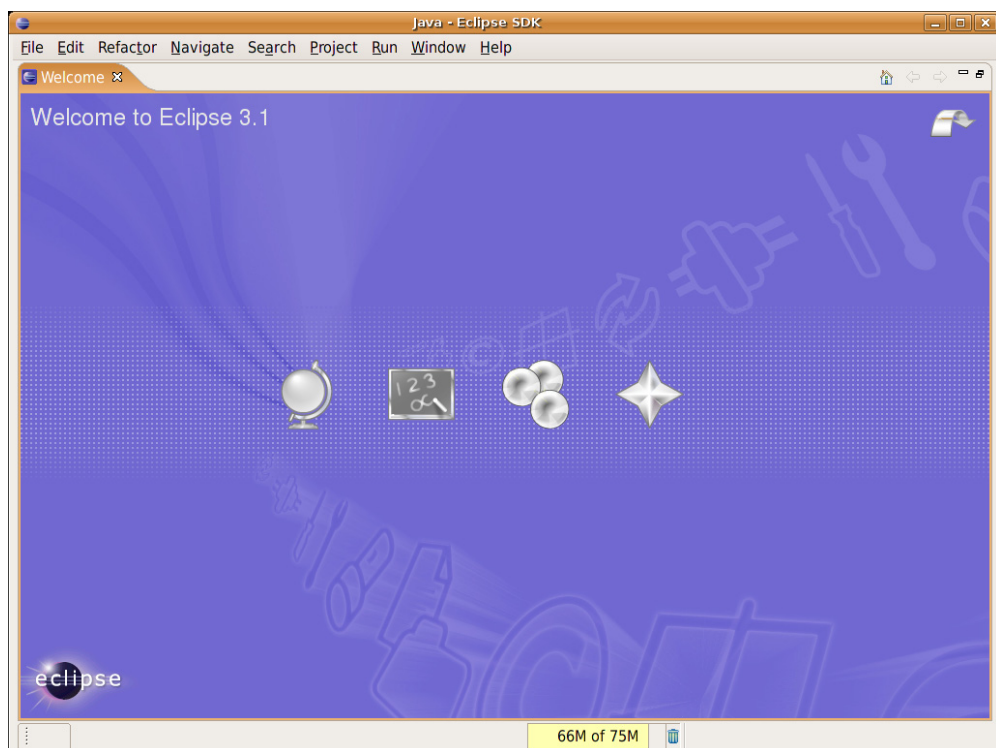
Zamieszczone poniżej przykłady korzystają z języka C, dlatego przyjąłem, że czytelnicy skorzystali z mojej rady i zainstalowali Eclipse wraz z pakietem *eclipse-cdt*, ponieważ dostarcza on wsparcie dla tego właśnie języka. Do pracy z przykładami niezbędne są też pakiety: *build-essential*, instalujący kompilator gcc, oraz *make*, niezbędny do tworzenia programu w języku C, oba zostały omówione w punkcie „Instalowanie GCC, make i innych narzędzi”.

Pierwsze uruchomienie Eclipse

W trakcie instalacji programu do menu *Aplikacje* została dodana nowa pozycja: *Programowanie*, w której znajduje się kolejne menu, właśnie *Eclipse*, za jego pomocą można uruchomić omawiane IDE. Przy pierwszym uruchomieniu Eclipse zostanie wyświetlony ekran powitalny widoczny na rysunku 18.4.

Podczas pierwszego uruchomienia program poprosi użytkownika o wskazanie miejsca, gdzie zostanie utworzony katalog roboczy o nazwie *workspace*, w którym będą przechowywane tworzone projekty. Domyślnie jest on tworzony w katalogu domowym użytkownika. Inny katalog, o nazwie *.eclipse*, używany jest do przechowywania danych konfiguracyjnych programu, organizacji interfejsu użytkownika itp. Można uruchomić Eclipse tak, aby został użyty inny katalog roboczy, należy wówczas w linii poleceń podać argument *-data*, po którym trzeba wpisać pełną ścieżkę nowego katalogu roboczego.

Na ekranie powitalnym znajdują się łącza do informacji dotyczących pracy w Eclipse, takich jak samouczki, przykładowe kody źródłowe itd. Aby zamknąć ekran powitalny, należy po prostu kliknąć znak *x* znajdujący się przy nazwie zakładki *Welcome*. By ponownie wyświetlić ekran powitalny, trzeba wybrać z menu programu *Help/Welcome*.



Rysunek 18.4. Ekran powitalny w Eclipse

Po zamknięciu ekranu powitalnego zostanie wyświetlone okno widoczne na rysunku 18.5.

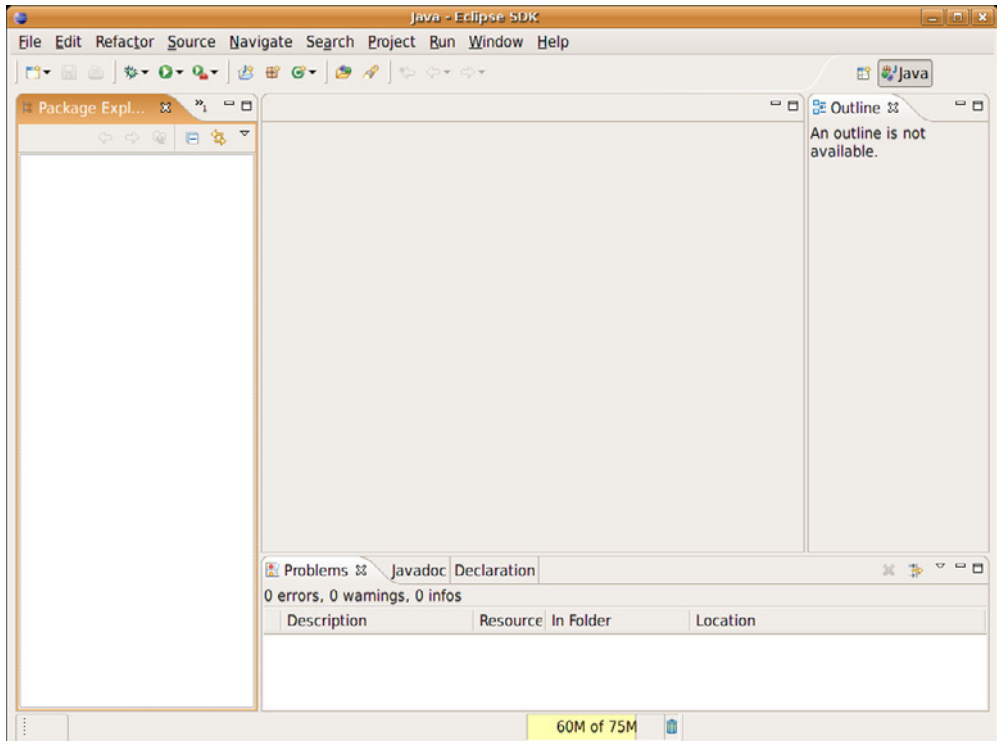
W tym momencie wszystkie panele są puste, ponieważ nie został jeszcze utworzony żaden projekt. W następnych dwóch częściach postaram się usunąć to niedopatrzenie.

Tworzenie nowego projektu w Eclipse

Po uruchomieniu Eclipse nowy projekt można utworzyć, wybierając z menu *File/New*, a następnie odpowiedni element. Aby utworzyć aplikację i pracować z nią w Eclipse, najpierw należy utworzyć projekt, w którym będzie przechowywany kod źródłowy. W Eclipse dostępne są kreatory, które znacznie upraszczają tworzenie projektów. Kreator to seria następujących po sobie okien programu, które pomagają użytkownikowi przejść przez proces tworzenia określonego projektu, dzięki czemu można mieć pewność, że żaden z wymaganych etapów nie zostanie pominięty.

Po wybraniu z menu *File/New/Project* zostanie uruchomiony kreator, którego pierwsze okno widoczne jest na rysunku 18.6.

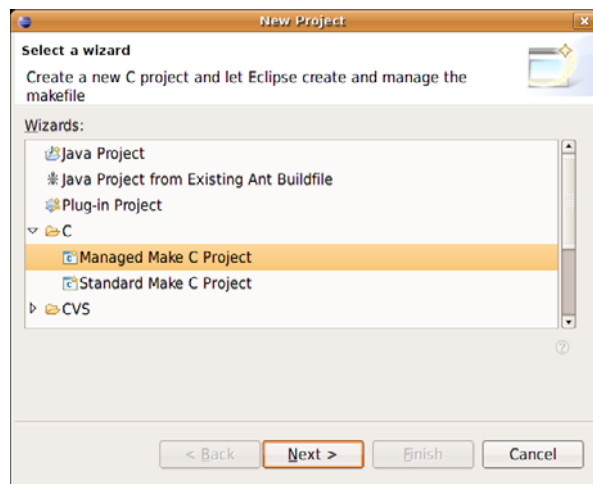
Ponieważ w poniższym przykładzie do tworzenia programu zostanie wykorzystany język C, dlatego należy w oknie kreatora najpierw rozwinąć folder oznaczony literą *C*, a następnie zaznaczyć pozycję *Managed Make C Project*. Dostępne są dwie opcje, wymieniona właśnie *Managed Make C Project*, dla której Eclipse automatycznie wygeneruje plik *Makefile*,



Rysunek 18.5. Standardowy interfejs użytkownika w programie Eclipse

Rysunek 18.6.

Wybór kreatora projektu w programie Eclipse



oraz *Standard Make C Project*, tu ten obowiązek spoczywa na użytkowniku. W pierwszym przykładzie zostanie wykorzystana opcja *Managed Make C Project*, w drugim, prezentującym możliwości importu i pracy w Eclipse nad już istniejącym projektem, zastosowana zostanie opcja *Standard Make C Project*.

Terminologia

Eclipse zostało zaprojektowane w firmie IBM, która — jak się zdaje — opracowała własny język do opisu elementów interfejsu użytkownika i w ogóle całego środowiska. Osoby, które korzystały już wcześniej z produktów IBM, powinny być zaznajomione z tą nomenklaturą. Poniżej zamieszczam kilka terminów używanych w programie. Jak ma to miejsce w większości języków, po ich bliższym poznaniu stają się one jasne i logiczne, ale przy pierwszym spotkaniu mogą wprawić w pewne zakłopotanie.

- ♦ **editor** (edytor) — okno umożliwiające edycję określonego pliku lub innego zasobu. Zawartość menu oraz pasków narzędzi Eclipse jest uzależniona od edytora używanego w danym momencie.
- ♦ **feature** (funkcjonalność) — jedna lub kilka wtyczek składających się na produkt.
- ♦ **perspective** (perspektywa) — zestaw okien zorganizowanych w określony sposób i wyświetlanych w graficznym interfejsie Eclipse, powiązanych ze sobą wyświetlaną treścią czy użytą wtyczką.
- ♦ **plugin** (wtyczka) — kod wykonujący pewną funkcję lub zestaw funkcji, który może być zintegrowany ze środowiskiem Eclipse.
- ♦ **view** (widok) — okno interfejsu graficznego Eclipse, które może obsługiwać edytor lub umożliwiać dostęp do określonego typu zasobów. Każdy widok (*view*) posiada pasek tytułu który udostępnia zakładki służące do wyboru różnych zasobów otwieranych w widoku; często posiada własny zestaw ikon i menu dostępnych po kliknięciu prawym przyciskiem myszy lub zaznaczeniu ikony znajdującej się na pasku tytułowym.
- ♦ **workbench** (środowisko robocze) — pierwsza z dwóch kluczowych wtyczek programu Eclipse, oferuje podstawowy interfejs użytkownika, struktury menu, mechanizm powiadomień oraz mechanizm tworzenia nowych okien i widoków. Umożliwia również deweloperom tworzenie menu, zdarzeń, edytorów i widoków w tworzonych przez nich wtyczkach. Z punktu widzenia użytkownika środowisko robocze (*workbench*) to po prostu interfejs programu.
- ♦ **workspace** (przestrzeń robocza) — druga z dwóch kluczowych wtyczek dostępnych w IDE Eclipse, umożliwia interakcję z zewnętrznymi zasobami, takimi jak katalogi projektów czy pliki. Termin ten może być również użyty do określenia katalogu, w którym przechowywane są projekty tworzone w środowisku Eclipse.

Jest to — oczywiście — moja interpretacja „języka Eclipse”. W pomocy znajdującej się na stronach projektu, w części *Workbench User Guide*, można znaleźć oficjalne definicje powyższych terminów.

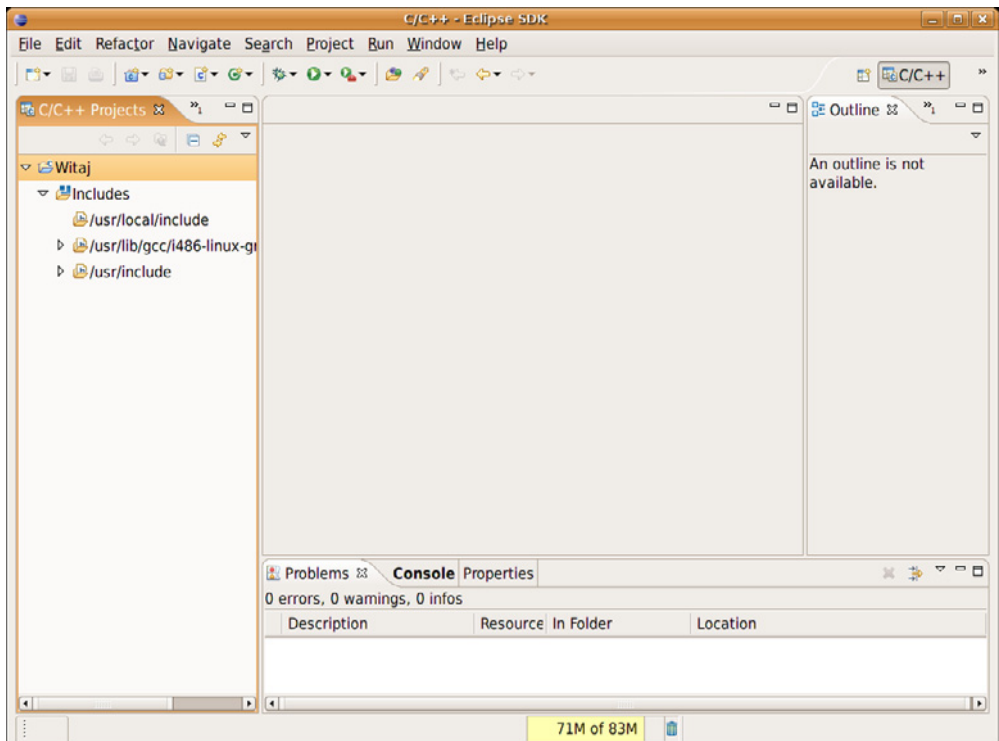
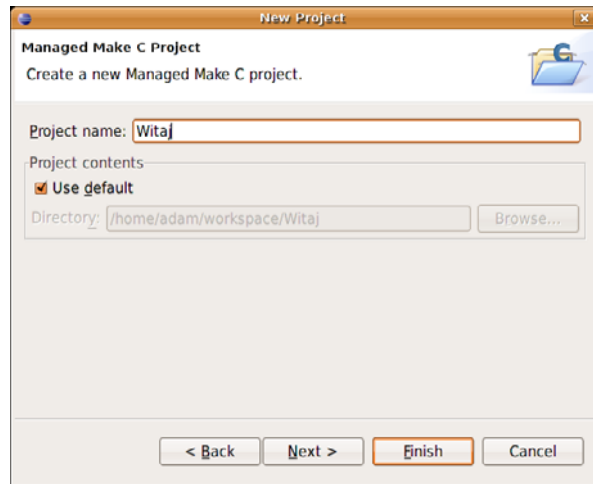
Po wybraniu odpowiedniej pozycji (*Managed Make C Project*) należy kliknąć przycisk *Next*, zostanie wyświetlone kolejne okno kreatora (patrz rysunek 18.7).

Teraz wprowadzamy nazwę dla nowego projektu (np. *halo*). Następnie można kliknąć przycisk *Finish*, wtedy projekt zostanie utworzony z domyślnymi ustawieniami (skonfigurowanymi do utworzenia samodzielnej aplikacji dla Linuksa) lub *Next* jeżeli użytkownik chciałby zbudować kod obiektowy innego typu, np. współdzielonej lub statycznej biblioteki linuksowej. Program pokaże kolejne okno z pytaniem, czy ma wyświetlić nowo utworzony projekt w perspektywie (*perspective*) C/C++, która jest przypisana do projektów tworzonych w języku C. Po kliknięciu przycisku *Yes* pojawi się struktura oraz zawartość projektu (patrz rysunek 18.8).

Oto opis zawartości domyślnego okna po uruchomieniu nowego projektu w perspektywie C/C++.

Rysunek 18.7.

Określanie nazwy nowego projektu w Eclipse



Rysunek 18.8. Nowy projekt w Eclipse

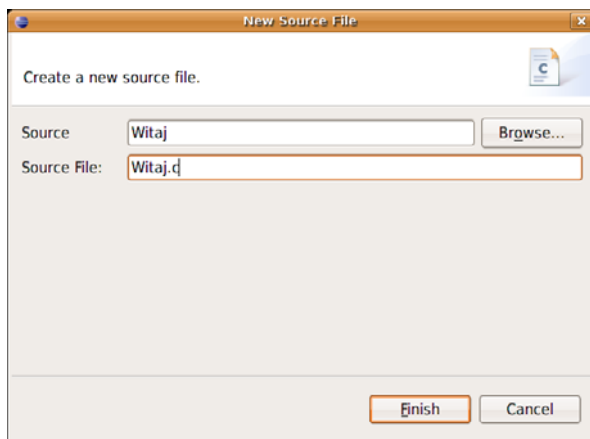
- ♦ **Navigator** — panel widoczny po lewej stronie służący do nawigowania po projekcie oraz plikach i innych zasobach które tworzą projekt.
- ♦ **Editor** — panel widoczny w centralnej części okna, który służy do edycji zasobów zaznaczonych w panelu *Navigator*. Dostosowuje się do typu edytowanego zasobu.

- ♦ **Console** — panel widoczny w dolnej części okna, w którym wyświetlane są komunikaty dotyczące działania programu. W zakładce *Problems* znajdują się informacje na temat znalezionych błędów i potencjalnych problemów w danym projekcie, natomiast w zakładce *Properties* można sprawdzić właściwości projektu.
- ♦ **Outline** — panel widoczny w prawej części okna, służy do wyświetlania struktury i schematu plików oraz funkcji użytych w danym projekcie.

Jedyną rzeczą, której na razie brakuje w tym projekcie, jest jakikolwiek plik napisany w języku C. Aby to zmienić, należy wybrać z menu *File/New/Source File*, zostanie wyświetlone okno widoczne na rysunku 18.9. Można też utworzyć nowe zasoby dla projektu, klikając prawym przyciskiem myszy jego nazwę, a następnie z menu kontekstowego wybierając element, który będzie potrzebny.

Rysunek 18.9.

Tworzenie nowego
pliku źródłowego
w Eclipse

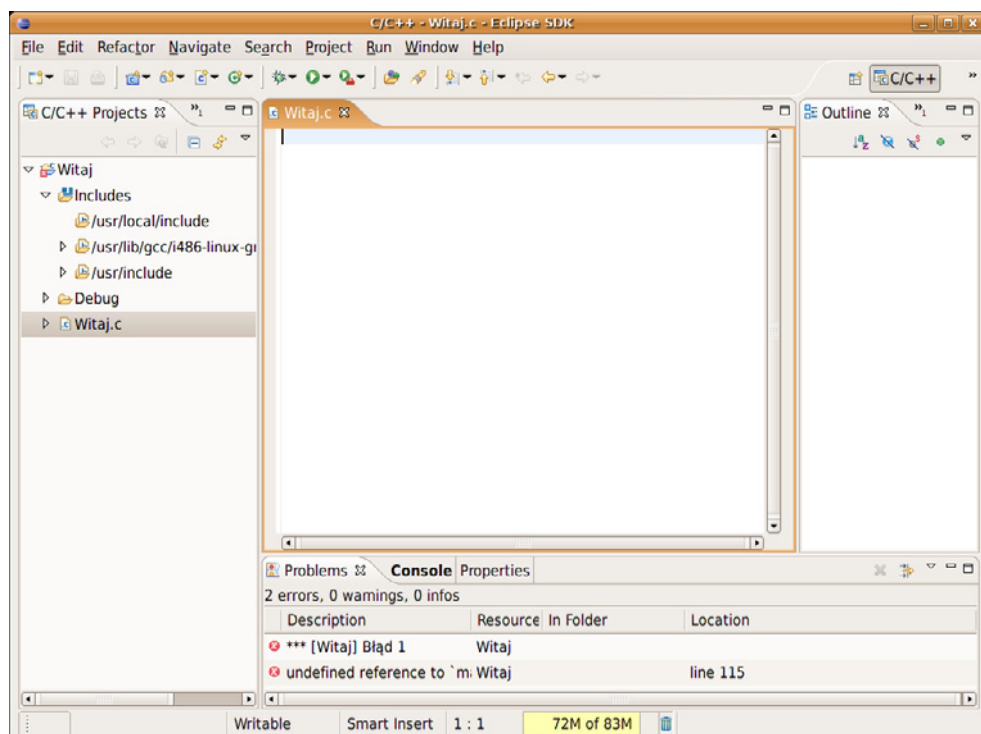


Zawartość nowego pliku po utworzeniu zostanie wyświetlona w głównym oknie programu (patrz rysunek 18.10).

Po wykreowaniu pierwszego pliku projektu w trybie *Managed Make C Project* można zauważyć, że w panelu *Console* została wyświetlona zakładka *Problems*, w której zwraca się uwagę użytkownika na fakt, że projekt nie ma głównej procedury (ponieważ plik jest pusty). Jest to pomocny komunikat, ponieważ za każdym razem, gdy coś zostanie zmienione, program próbuje automatycznie zbudować projekt. Można wyłączyć funkcję automatycznego budowania, wystarczy usunąć zaznaczenie z pozycji menu *Project/Build Automatically*. Od tej chwili program będzie tworzył projekt tylko na polecenie użytkownika.

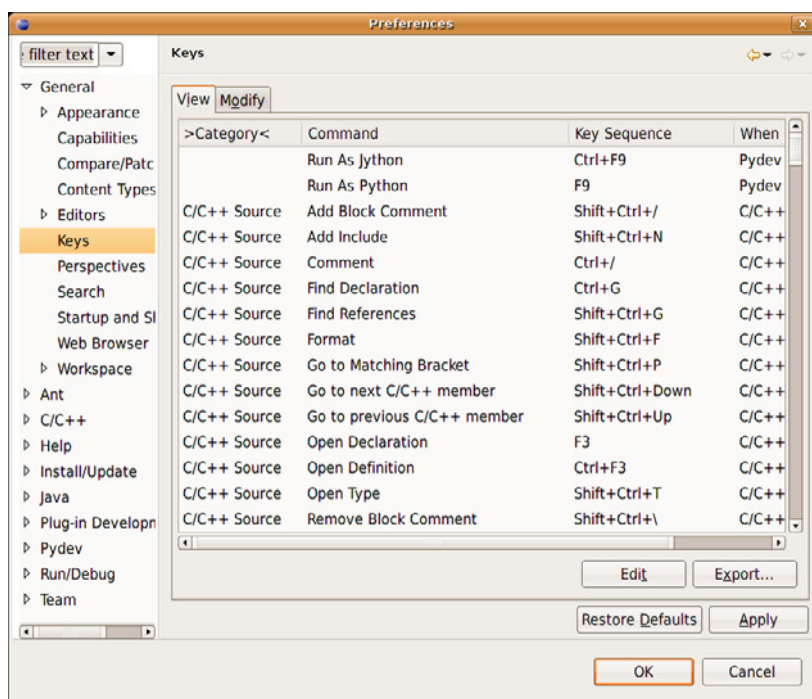
Następnie w panelu edytora należy wprowadzić kod źródłowy. Eclipse rozpoznaje wiele skrótów klawiszowych ułatwiających pracę z programem. Listę aktualnie używanych można wyświetlić, wybierając z programu *Window/Preferences*, a następnie *General/Keys* (patrz rysunek 18.11).

Jeżeli użytkownik jest przyzwyczajony do używania określonych skrótów (np. takich samych jak w edytorze emacs), może dostosować używane w Eclipse do swoich przyzwyczajeń. W tym celu należy przejść do zakładki *Modify* (patrz rysunek 18.11). i wyświetlić okno służące do zmiany ustawień klawiszy. W programie dostępne są gotowe schematy znane z innych programów (np. właśnie z emacsa), z których można skorzystać bez

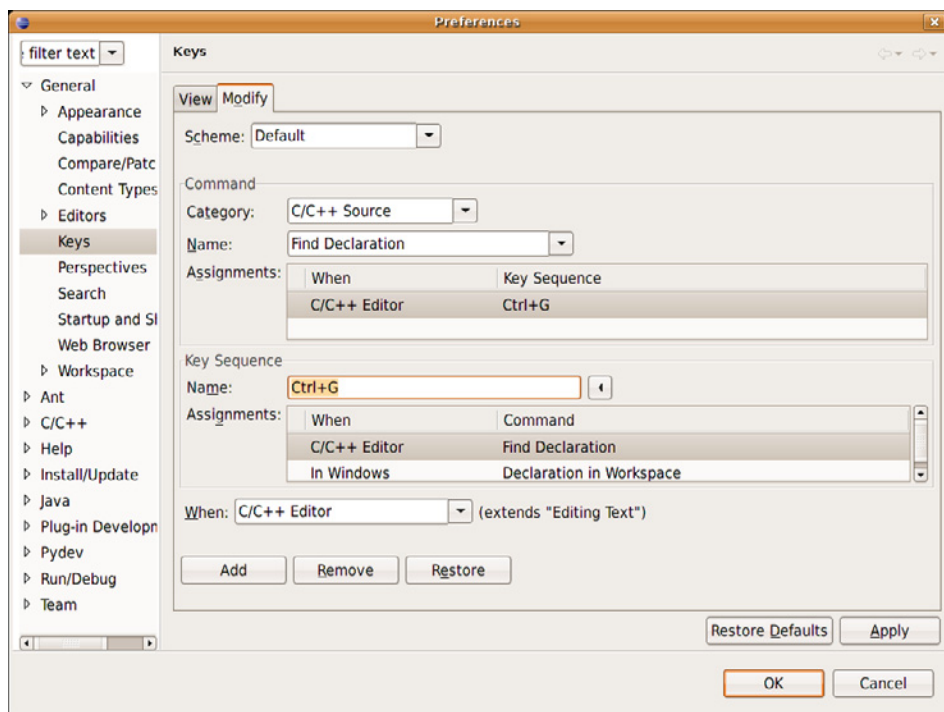


Rysunek 18.10. Nowy plik źródłowy w edytorze

Rysunek 18.11.
Aktualnie używane
skrótów klawiszowych
w programie Eclipse



konieczności zmieniania wszystkiego samodzielnie. Na rysunku 18.12 przedstawiam okno, w którym można wprowadzić te zmiany. Aby zamknąć okno, należy kliknąć przycisk *OK*.

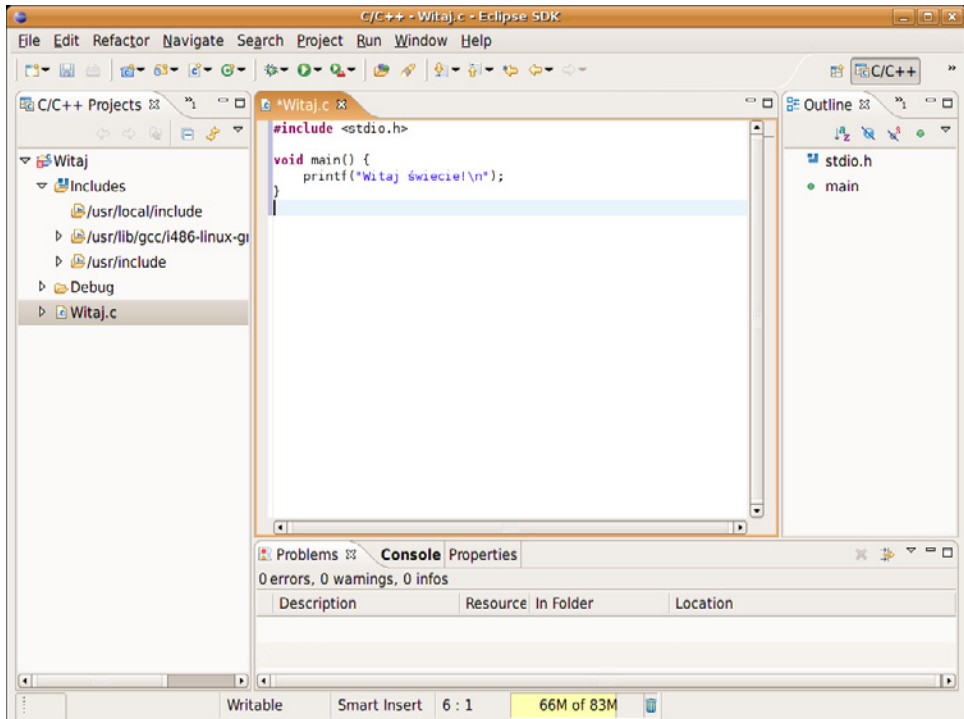


Rysunek 18.12. Dostosowywanie skrótów klawiszowych w programie Eclipse

Kiedy już użytkownik wie, jak sprawnie poruszać się w oknie edytora, pora wprowadzić kod programu. W przykładzie użyto sztandarowego programu „Halo świecie”. Podczas wpisywania kodu użytkownik szybko zauważy, że Eclipse samodzielnie rozpoczyna nowe linie, zaznacza nawiasy i zaznacza kod tak, jak można tego oczekiwać od współczesnego środowiska programistycznego. W trakcie wprowadzania kodu można też zauważyć, że w panelu Outline na bieżąco aktualizowane są odniesienia do kodu, definiowane funkcje (łącznie z *main*) itd. Na rysunku 18.13 przedstawiam Eclipse po wprowadzeniu kompletnego kodu programu „Witaj świecie”.

Po wprowadzeniu kodu należy go zapisać, w tym celu trzeba wybrać z menu *File/Save*. Częścią procesu zapisywania jest analiza kodu, podczas której wykrywane są potencjalne problemy i błędy, najczęściej są to brakujące linie, błędne wartości itp.

Teraz pora na uruchomienie zbudowanego programu. Jeżeli w menu nie została wyłączona opcja *Build Automatically*, wówczas Eclipse wykona to automatycznie po zapisaniu projektu. Jeśli opcja została wyłączona, należy wykonać to samodzielnie. W tym celu trzeba wybrać z menu *Project/Build Project* (lub kliknąć dany projekt prawym przyciskiem myszy i z menu wybrać *Build Project*). Przy odrobinie szczęścia w konsoli nie pojawią się żadne błędy.



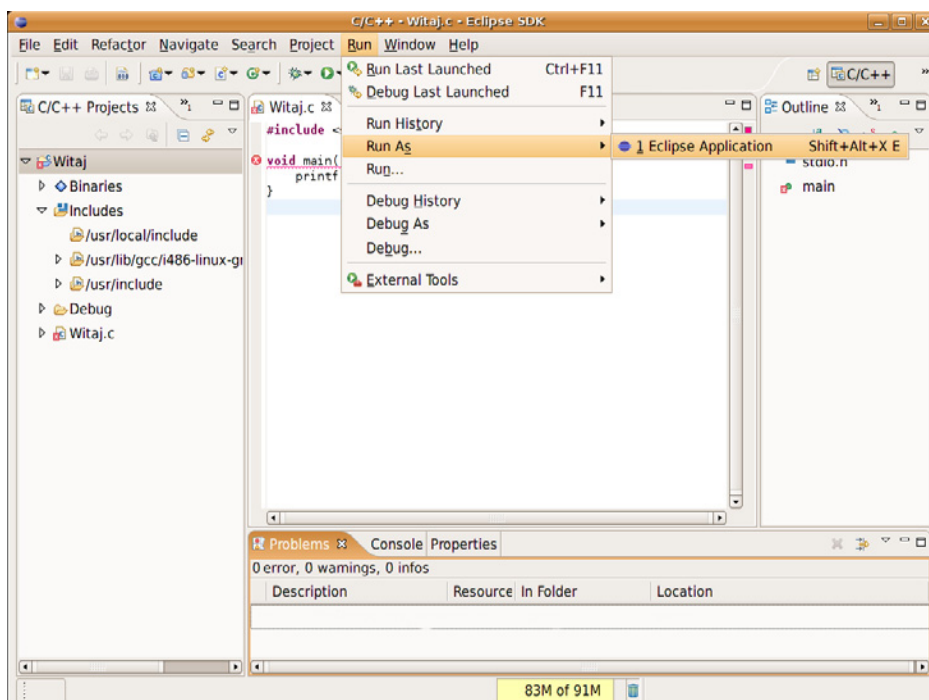
Rysunek 18.13. Przykładowa aplikacja w Eclipse

Ostatnim krokiem jest uruchomienie programu i sprawdzenie efektu końcowego. Aby to zrobić, należy wybrać z menu polecenie *Run*. Eclipse pozwala na zdefiniowanie wielu konfiguracji zarówno w przypadku budowania jak i uruchamiania zbudowanych programów. Dla powyższego przykładu wybrałem standardowy sposób uruchomienia samodzielnej aplikacji C/C++, w tym celu należy kolejno z menu wybrać *Run/Run As/Run Local C/C++ Application* (patrz rysunek 18.14).

Po wybraniu tego polecenia z menu program wyświetli okno dialogowe widoczne na rysunku 18.15, w którym można zdecydować, jak ma zostać uruchomiony dany program. Ponieważ w ustawieniach domyślnych podczas tworzenia programu używany jest debugger, w oknie tym można zdecydować, czy wykorzystany zostanie lokalny GNU debugger (gdb), czy nastąpi połączenie ze zdalnym serwerem gdb.

Należy wybrać opcję GDB Debugger i kliknąć *OK*. Eclipse uruchomi aplikację i zaktualizuje zawartość wyświetlaną w konsoli, w której pojawi się wynik działania programu, taki jak na rysunku 18.16.

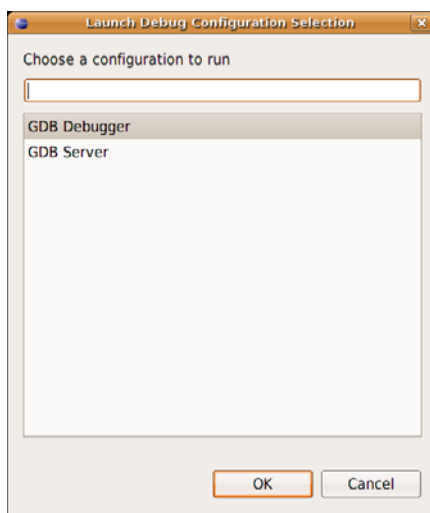
Aby uprościć uruchamianie przykładowego programu, można utworzyć specjalną konfigurację polecenia *Run*, która spowoduje wykonywanie programów z innymi ustawieniami polecenia *Build*. Nosi ona nazwę *Release* i umożliwi tworzenie aplikacji np. bez symboli debuggera, ale jej omówienie wykracza poza ramy niniejszego rozdziału. W powyższym przykładzie użytkownik utworzył projekt, kod źródłowy, a nawet uruchomił program. Całkiem nieźle, jak na dziesięć minut pracy.



Rysunek 18.14. Uruchamianie projektu

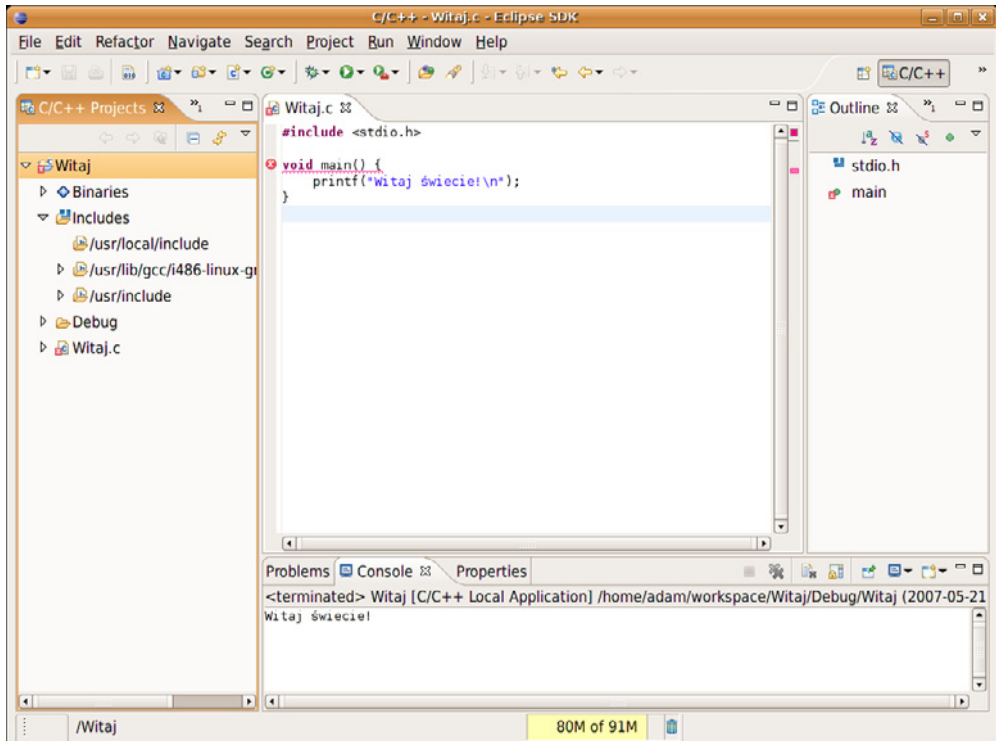
Rysunek 18.15.

Określanie parametrów debugowania w Eclipse



Importowanie istniejących projektów do środowiska Eclipse

Tworzenie nowych projektów to jedno, ale inną funkcją udostępnianą przez Eclipse jest możliwość importowania już istniejących projektów. W niniejszym przykładzie użyty zostanie jeden z moich ulubionych otwartych programów, czyli pop3check, który służy



Rysunek 18.16. Działająca przykładowa aplikacja w Eclipse

do wyświetlania listy nagłówków wiadomości pocztowych znajdujących się na koncie zdalnego serwera POP. Warto stosować ten program jako narzędzie linii poleceń do sprawdzania, czy nie nadeszły nowe wiadomości. Program `pop3check` został napisany przez Stevena Radacka i Simon Liddington, można go pobrać ze strony <http://sourceforge.net/projects/pop3check/>.

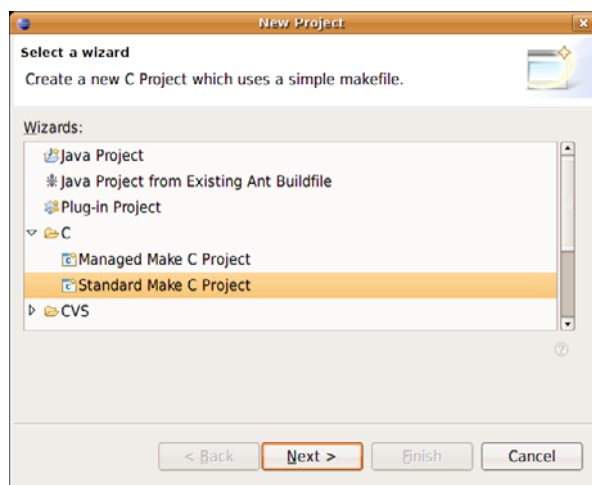
Moim zdaniem, najprostszym sposobem importu istniejącego projektu do środowiska Eclipse jest użycie archiwum tar zawierającego źródła. Pozwala to na import „dziewiczego” kodu a dodatkowo pozwala zachować stan źródeł w momencie ich importowania. Przygotowując się do importu, należy przejść do katalogu w którym znajdują się źródła programu a następnie usunąć pliki obiektowe, kopie plików oraz inne elementy które nie mają trafić do nowego projektu. Chcemy też, aby struktura plików i katalogów w naszym projekcie odpowiadała oryginałowi, dlatego archiwum będziemy generowali w katalogu głównym projektu. Większość archiwów tar zbudowana jest w taki sposób aby po rozpakowaniu wszystkie pliki projektu znajdowały się w katalogu o nazwie, z reguły, zbliżonej do nazwy programu, co w przypadku zaimportowania takiego archiwum do Eclipse spowoduje umieszczenie plików projektu w dodatkowym, zbędnym podkatalogu. Nasz plik możemy przygotować w następujący sposób:

```
$ cd pop3check-0.100.3
$ make clean
$ tar cfz ../pop3check.tar *
```


Plik w formacie tar przygotowany, pora uruchomić Eclipse (*Aplikacje/Programowanie/Eclipse*) i utworzyć nowy projekt, wybierając kolejno z menu *File/New/Project*, zostanie wówczas uruchomiony kreator. Ponieważ tym razem będzie importowany gotowy projekt, w którym jest już plik *Makefile*, w oknie widocznym na rysunku 18.17 należy wybrać opcję *Standard Make C Project*.

Rysunek 18.17.

Nowy projekt
w Eclipse



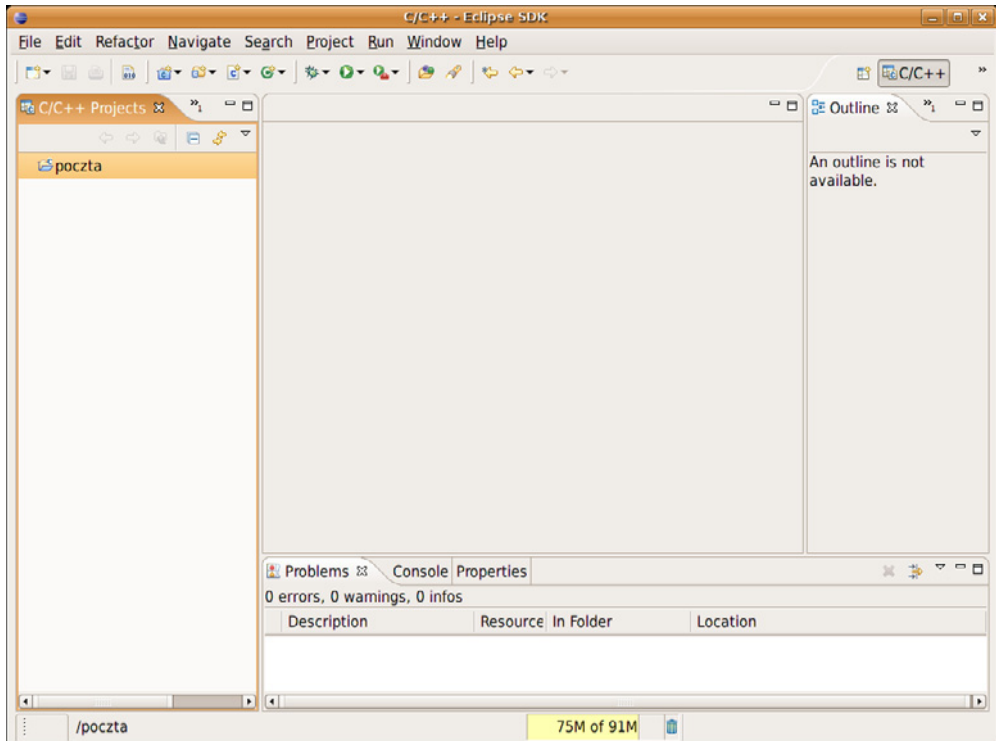
Po wprowadzeniu nazwy nowego projektu trzeba kliknąć przycisk *Finish*, zostanie wyświetlone okno programu Eclipse z nowym — jeszcze pustym — projektem (patrz rysunek 18.18).

Jak można się spodziewać, nowo utworzony projekt jest zupełnie pusty. Aby zaimportować plik w formacie tar przygotowany wcześniej, należy kliknąć prawym przyciskiem nazwę projektu i z menu kontekstowego wybrać opcję *Import*. Zostanie wyświetlone okno dialogowe widoczne na rysunku 18.19.

Ponieważ importowany będzie plik tar, należy wybrać pozycję *Archive file* i kliknąć *Next*. Zostanie wyświetlone kolejne okno dialogowe (patrz rysunek 18.20), w którym będzie można wskazać plik do importu przygotowany wcześniej.

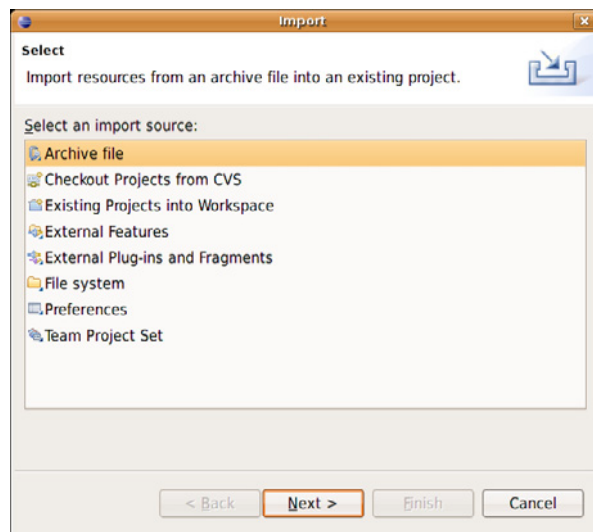
W tym oknie można przejrzeć zawartość pliku tar i sprawdzić, czy zawiera wszystkie potrzebne pliki. Aby rozpocząć import, należy kliknąć przycisk *Finish*. Kiedy operacja zostanie zakończona, zawartość pliku tar zostanie dodana do folderu projektu (patrz rysunek 18.21).

Po zaimportowaniu pliku do Eclipse dobrze jest zweryfikować, czy cele zdefiniowane w pliku *Makefile* odpowiadają tym, których oczekuje Eclipse. Można poinformować Eclipse o innych celach klikając plik *Makefile* prawym przyciskiem i wybierając z menu polecenie *Create Make target*, ale najłatwiej rozpocząć od standardowych. Aby sprawdzić plik *Makefile*, należy kliknąć dwukrotnie jego nazwę w panelu nawigatora (patrz rysunek 18.22).



Rysunek 18.18. Standardowe okno programu Eclipse

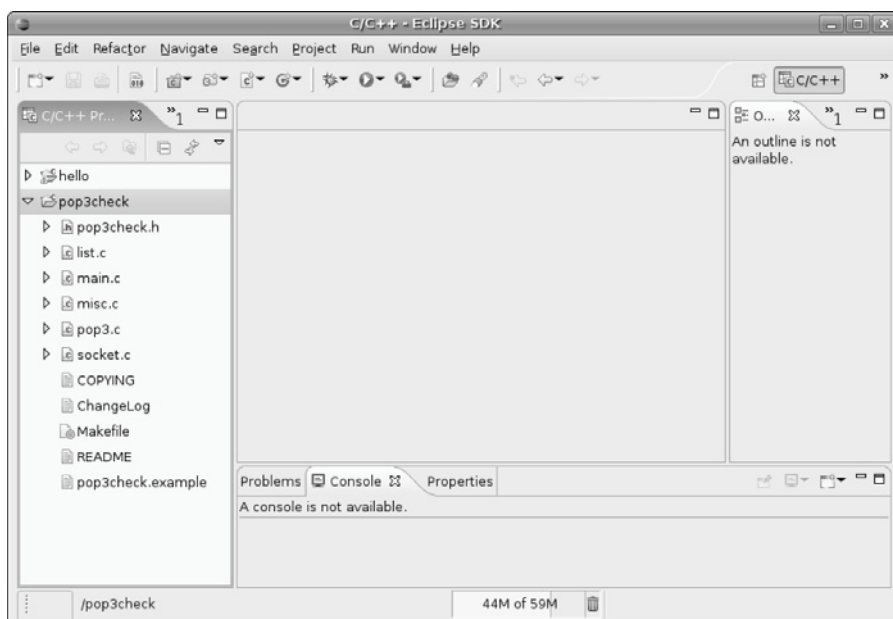
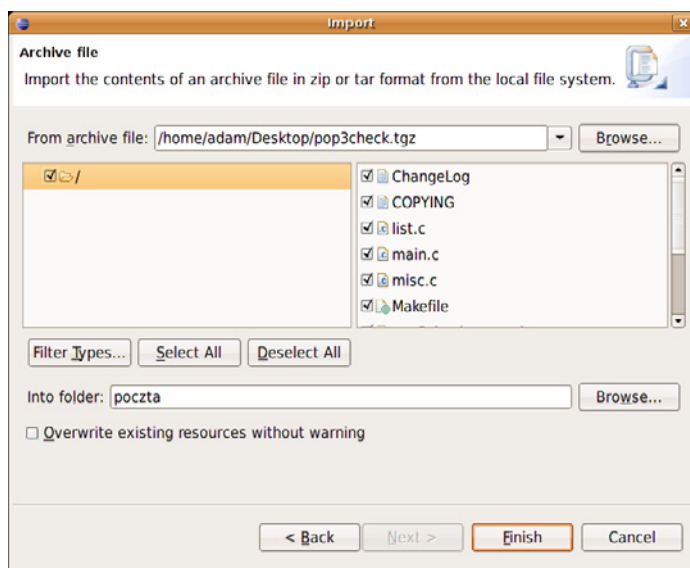
Rysunek 18.19.
Określanie rodzaju
importu istniejącego
projektu do programu
Eclipse



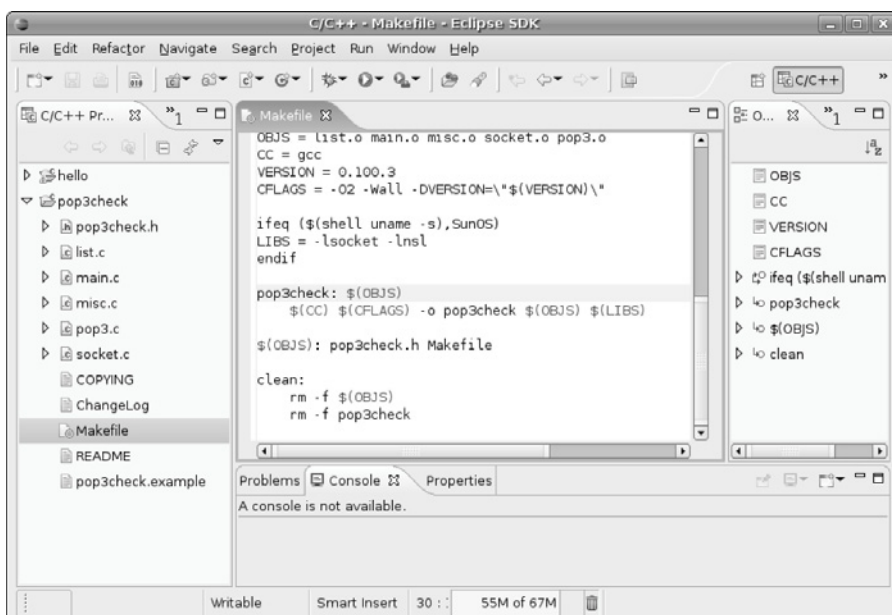
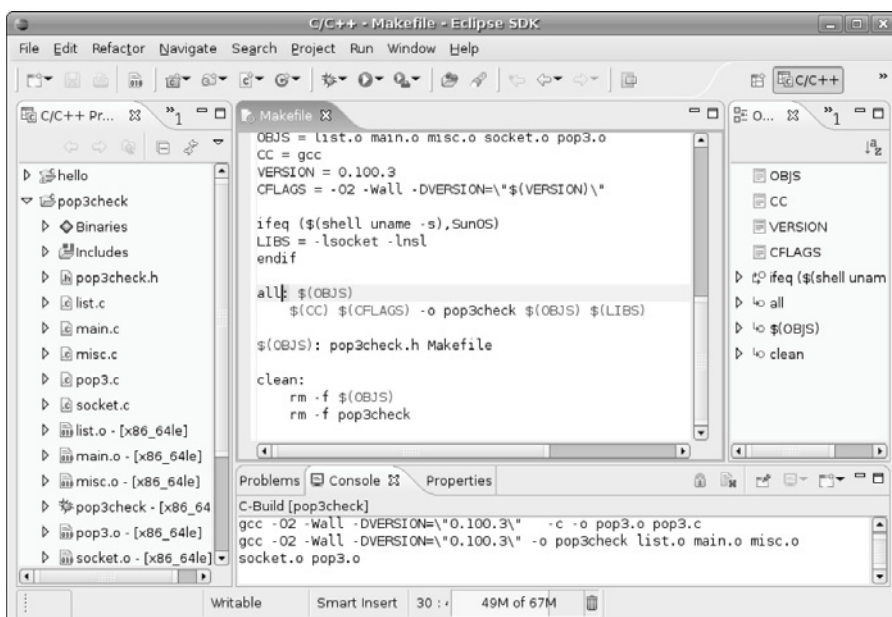
W tym przypadku domyślnym celem jest pop3check. Po wprowadzeniu ewentualnych zmian i ich zapisaniu (menu *File/Save*) można zbudować projekt zgodnie z informacjami zamieszczonymi powyżej (*Project/Build All*). W naszym przykładzie, zaimportowany projekt zbudował się poprawnie (patrz rysunek 18.23).

Rysunek 18.20.

*Wybór pliku,
który ma zostać
zaimportowany
do programu Eclipse*

**Rysunek 18.21. Zaimportowany projekt**

Oczywiście, jeżeli użytkownik będzie importował inne projekty, wówczas procedura może się nieco różnić. Łatwość, z jaką można to robić, powoduje, że warto przenosić istniejące projekty i wykorzystywać możliwości oferowane przez te nowoczesne i łatwe w użyciu zintegrowane środowiska programistyczne, do których zalicza się Eclipse.

**Rysunek 18.22.** Edytowanie pliku Makefile**Rysunek 18.23.** Dane wyjściowe w konsoli po utworzeniu zaimportowanego wcześniej projektu

Dodatkowe informacje o Eclipse

Eclipse to potężne i posiadające wiele funkcji środowisko programistyczne; dwa powyższe przykłady przedstawiły mały wycinek oferowanych przez nie możliwości. Aby uzyskać więcej informacji na jego temat, warto zajrzeć do jednej z książek lub odwiedzić jedną z poniższych stron internetowych:

- ♦ www.eclipse.org — strona domowa projektu,
- ♦ www.eclipse.org/documentation/ — strona z dokumentacją programu,
- ♦ <http://planet.eclipse.org/planet/> — planeta Eclipse często odwiedzana przez programistów, uczestników projektu, hakerów i użytkowników Eclipse,
- ♦ <http://eclipseplugincentral.com/> — centrala, z której można pobrać wiele darmowych wtyczek do Eclipse.

W następnej części niniejszego rozdziału omówię systemy kontroli wersji oprogramowania ze szczególnym uwzględnieniem Subversion SCCS. Przykładem różnorodności wtyczek dostępnych dla Eclipse są dwie bezpłatne wtyczki Subversion:

- ♦ Subversion Plug-in, www.polarion.org,
- ♦ Subclipse Plug-in, <http://subclipse.tigris.org>.

Wtyczki te ułatwiają pracę z Subversion w Eclipse, są również doskonałym przykładem na to, jak wolne oprogramowanie może ze sobą współpracować. A cena — oczywiście — nadal bezkonkurencyjna...

System kontroli wersji

Jeżeli czytelnikowi zdarzyło się kiedyś skasować przypadkiem kod programu, nad którym długo pracował, utracić na skutek awarii dysku powieść lub pracować nad projektem z kimś, kto bezmyślnie nadpisał kod bez zachowania kopii, zapewne zastanawiał się nad tym, że korzystanie z systemu kontroli wersji zapobiegłoby takim wypadkom. System kontroli wersji, nazywany czasem też SCC (ang. *source code control*), to oprogramowanie służące do zarządzania kodem źródłowym, śledzenia wprowadzanych do niego zmian, zarządzania różnymi wersjami kodu (zwanymi **gałęziami**) oraz oferujące mechanizmy odzyskiwania dowolnej wersji kodu z dowolnego pliku i dowolnego okresu.

Gdybym miał zliczyć wszystkie sytuacje w których SCCS uratował mi skórę, na pewno zabrakłoby mi palców u rąk i stóp. Nie znaczy to że jestem niechlujny czy niestaranny. Gdy wielu deweloperów pracuje nad tymi samymi plikami, konflikty pomiędzy wprowadzanymi przez nich zmianami są nieuniknione. SCCS dostarcza mechanizmy identyfikujące i usuwające konflikty między zmianami (tzw. *merging*). System kontroli wersji ułatwia eksperymentowanie z kodem umożliwiając bezpieczny powrót do działającej wersji kodu w przypadku gdyby błyskotliwy pomysł w praktyce nie był już tak olśniewający.

W dalszej części rozdziału omówię dwa najpopularniejsze dla Linuksa systemy kontroli wersji kodu: CVS (ang. *Concurrent Versioning System*) i Subversion. Używam CVS od lat, ale poniżej na temat tego programu zamieszczę tylko kilka zdań. Nie bez przyczyny

o Subversion mówi się, że jest to CVS++. To faktycznie rozwinięcie koncepcji tamtego systemu, z większą ilością nowoczesnych funkcji, obszerniejszym zestawem poleceń, nowszymi protokołami i większym stopniem integracji z przeglądarkami internetowymi, zintegrowanymi środowiskami programistycznymi, graficznymi interfejsami, a w dodatku system jest stale rozwijany. Biedny CVS...



Powszechnym nieporozumieniem jest przyjęcie założenia, że system kontroli wersji może zastąpić system wykonywania kopii zapasowych. To — niestety — nie jest prawdą. Może on, co prawda, posłużyć użytkownikowi jako kopia, z której może odzyskać swoje pliki, gdyby jego własny komputer stanął w płomieniach. Jednak centralne repozytorium systemu przechowujące wszystkie dane musi być regularnie archiwizowane, ponieważ jest to najsłabszy, a zarazem mający krytyczne znaczenie punkt systemu.

CVS — Concurrent Versioning System

Moje humorystyczne komentarze pod adresem CVS to — oczywiście — żarty, które wcale nie oznaczają, że CVS nie jest solidnie przetestowanym czy stabilnym systemem. Ten system jest używany przez programistów od lat i nadal często wykorzystywany w projektach otwartego oprogramowania. Jednak większość projektów, które do tej pory stosowały CVS, powoli migruje do Subversion i ta tendencja się utrzyma, a nawet przybierze na sile. Ponieważ żaden z omawianych systemów nie jest domyślną częścią instalacji Ubuntu, zalecam, aby użytkownicy instalowali Subversion zamiast CVS.

Jeżeli czytelnik jest zdeklarowanym zwolennikiem CVS lub uczestniczy w projekcie, w którym właśnie ten system jest wykorzystywany, może go zainstalować, korzystając z repozytoriów Ubuntu oraz menedżera pakietów Synaptic. W dalszej kolejności warto odwiedzić wymienione poniżej strony, ponieważ można na nich znaleźć informacje o konfigurowaniu CVS oraz uruchamianiu sieciowych repozytoriów CVS:

- ♦ http://www.faqs.org/docs/ldev/0130091154_198.htm — część Linux Development FAQ poświęcona systemowi CVS,
- ♦ <http://freeos.com/articles/4608> — ciekawy (i krótki!) artykuł autorstwa Mahantesha M. Vantmuriego opisujący konfigurowanie systemu CVS oraz synchronizowanie kopii repozytoriów; choć artykuł liczy już sobie dobrych parę lat, to jednak niewiele się zmieniło — wciąż mowa o tym samym CVS,
- ♦ <http://flexwiki.com/default.aspx/FlexWiki/ConfiguringCVS.html> — zbiór przydatnych informacji pomagających zrozumieć CVS i korzystać z niego.

CVS to dobry system, a sam korzystałem z niego w wielu firmach. Jeżeli jednak czytelnik planuje uruchomić system kontroli wersji na Ubuntu, zaczynając od podstaw, warto rozpocząć od nowszego i wydajniejszego systemu, jakim jest Subversion.

Instalowanie i korzystanie z Subversion

Subversion (<http://subversion.tigris.org/>) to stosunkowo nowy, ale już niesłychanie popularny system kontroli wersji. Oprócz jego wydajności oraz obszernego zestawu poleceń, swą popularność zawdzięcza również temu, że wykorzystano w nim WebDAV (ang. *WebDistributed Authoring and Versioning*), sieciowy protokół umożliwiający przeglądanie i dostęp do repozytoriów z poziomu przeglądarki internetowej. Obejmuje to również

możliwość przeciągania wprost z przeglądarki plików na pulpit i w kierunku odwrotnym. Skonfigurowanie programu i repozytoriów oraz zintegrowanie go z przeglądarką jest proste, podobnie zresztą jak korzystanie z samego systemu, czy to za pomocą pulpitu, czy linii poleceń.

Instalowanie Subversion

Aby zainstalować Subversion oraz powiązane pakiety, należy uruchomić menedżer pakietów Synaptic (*System/Administracja*), kliknąć znajdujący się na pasku narzędzi przycisk *Szukaj*, wpisać ciąg *subversion* i znowu kliknąć przycisk *Szukaj*. Kiedy wyszukiwanie zostanie zakończone, trzeba na wyświetlonej liście odszukać i zaznaczyć odpowiednie pakiety. Nie wszystkie wymienione poniżej są obowiązkowe, raczej zalecane z powodów, które umieszczono przy każdym z nich.

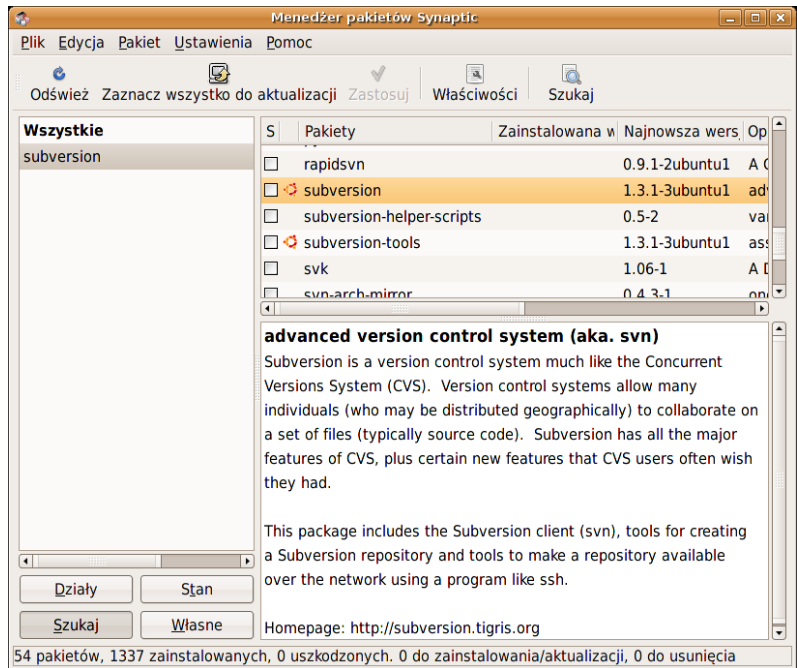
- ♦ *libapache2-svn* — (opcjonalne), ten pakiet jest niezbędny tylko wówczas, gdy użytkownik chce udostępniać repozytoria Subversion w sieci za pośrednictwem serwera WWW. Zaznaczenie tego pakietu spowoduje zainstalowanie w systemie również serwera Apache 2 i powiązanych z nim plików. Udostępnianie repozytoriów w sieci to bardzo przydatna opcja, łatwo ją skonfigurować, a dodatkowa ilość miejsca na dysku, którą trzeba przeznaczyć na serwer Apache, z pewnością jest tego warta. Oczywiście, ten pakiet zawsze można dodać później. W rozdziale 26., „Konfiguracja serwera WWW”, zamieszczam dodatkowe informacje na temat samodzielnej instalacji różnych wersji Apache.
- ♦ *rapidsvn*, *esvn* lub *kdesvn* — (opcjonalne), klienty graficzne dla systemu Subversion. Są niezbędne tylko wówczas, gdy użytkownik chce korzystać z interfejsu graficznego. Popularnym klientem dla GNOME jest *rapidsvn*; z kolei program *esvn* korzysta z biblioteki graficznej Qt, dzięki czemu nie zabiera wiele miejsca zarówno w GNOME, jak i KDE. Program *kdesvn* jest przeznaczony dla systemu KDE i do działania wymaga bibliotek KDE, ale jest to jedyny klient Subversion akceptujący argumenty linii poleceń. Może on być również przydatny, jeżeli użytkownik planuje zintegrowanie klienta Subversion z przeglądarką internetową Firefox do obsługi adresów typu *svn*: (więcej informacji na ten temat zamieszczono w podpunkcie „Obsługa adresów typu svn: w Firefoksie” znajdującym się w dalszej części rozdziału).
- ♦ *subversion* — (wymagane), główny pakiet systemu Subversion, zawiera wszystko, co potrzebne, aby używać systemu kontroli wersji na lokalnym komputerze za pomocą konsoli.
- ♦ *subversion-helper-scripts* — (opcjonalne), zestaw skryptów upraszczających pracę z repozytoriami Subversion. To mały, ale bardzo przydatny pakiet, zdecydowanie warto poświęcić na niego miejsce (ok. 200 kilobajtów).



Podczas instalacji warto też zastanowić się nad dodaniem pakietu *subversion-tools*. Oferuje on kilka narzędzi do udostępniania repozytoriów między użytkownikami oraz wykonywania nieco bardziej skomplikowanych importów istniejącego kodu do katalogów systemu Subversion. Jeżeli użytkownik nie posiada zainstalowanego w systemie serwera pocztowego, to po zaznaczeniu do instalacji tego pakietu zostanie on również dodany wraz z powiązаныmi pakietami. Ponieważ i tak używam serwera pocztowego, nie jest to dla mnie żadnym problemem, ale w przypadku części czytelników może być inaczej. W rozdziale 27., „Konfiguracja serwera poczty”, zamieszczam informacje na temat różnych serwerów pocztowych dostępnych w Ubuntu oraz wyjaśniam, dlaczego jest to dobre rozwiązanie.

Odpowiednie pakiety należy kliknąć prawym przyciskiem myszy i z menu kontekstowego wybrać opcję *Zaznacz do instalacji* (patrz rysunek 18.24).

Rysunek 18.24.
Instalowanie systemu Subversion oraz powiązanych pakietów



Zależnie od zainstalowanego wcześniej w systemie oprogramowania oraz pakietów wybranych w Synapticu, menedżer może wyświetlić kolejne okno dialogowe z dodatkowymi pakietami niezbędnymi do umieszczenia w systemie; należy zaakceptować tę propozycję i kliknąć przycisk *Zaznacz*.

Po zaznaczeniu wszystkich potrzebnych pakietów trzeba kliknąć znajdujący się na pasku narzędzi przycisk *Zastosuj*, Synaptic pobierze i zainstaluje system Subversion oraz inne, wybrane pakiety. Kiedy instalacja zostanie zakończona, można rozpocząć konfigurowanie repozytoriów, a jeżeli użytkownik zdecydował się zainstalować pakiet *libapache2-svn*, również i Apache 2 wspomagający system kontroli wersji.

Więcej informacji na temat instalowania oprogramowania za pomocą menedżera pakietów Synaptic i innych narzędzi dostępnych w Ubuntu można znaleźć w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Konfigurowanie repozytoriów systemu Subversion

Do tworzenia repozytoriów Subversion oraz ich struktury używane jest polecenie `svnadmin create`. Dla tego polecenia potrzebny jest jeden argument, mianowicie nazwa katalogu, w którym będą przechowywane repozytoria.

Ponieważ Subversion korzysta z WebDAV, które z kolei zależne jest od protokołu HTTP, dlatego należy pamiętać, że właścicielem Subversion musi być użytkownik i grupa

posiadająca uprawnienia do komunikowania się za pomocą tego protokołu. W Ubuntu są to odpowiednio użytkownik i grupa *www-data*.



Jeżeli użytkownik tworzy repozytorium Subversion o nazwie *svn* i zamierza eksportować je przy pomocy serwera WWW korzystając z jego nazwy, wówczas katalog z repozytorium nie może znajdować się w katalogu *DocRoot* serwera WWW. Zgodnie ze wskazówkami zamieszczonymi w dalszej części rozdziału, pełną ścieżkę do katalogów Subversion podaje się w pliku konfiguracyjnym Apache Subversion. Próba użycia repozytorium o nazwie *svn*, zlokalizowanego w katalogu *DocRoot* serwera WWW, spowoduje błąd 301 Moved Permanently (http://nazwa_serwera).

Aby np. utworzyć repozytorium w katalogu */var/svn*, należy wykonać następujące polecenie:

```
$ sudo -u www-data svnadmin create /var/svn
```

Po wprowadzeniu hasła powstanie repozytorium wraz z odpowiednimi podkatalogami, których właścicielem będzie właściwy użytkownik i grupa. Aby sprawdzić efekt działania, można wyświetlić zawartość danego katalogu, tak jak w poniższym przykładzie:

```
$ ls -l /var/svn
total 28
drwxr-xr-x 2 www-data www-data 4096 2007-05-20 15:23 conf
drwxr-xr-x 2 www-data www-data 4096 2007-05-20 15:23 dav
drwxr-sr-x 5 www-data www-data 4096 2007-05-20 15:23 db
-r--r--r-- 1 www-data www-data 2 2007-05-20 15:23 format
drwxr-xr-x 2 www-data www-data 4096 2007-05-20 15:23 hooks
drwxr-xr-x 2 www-data www-data 4096 2007-05-20 15:23 locks
-rw-r--r-- 1 www-data www-data 229 2007-05-20 15:23 README.txt
```

Oczywiście, puste repozytorium nie jest specjalnie ciekawe. Poniżej napisałem, jak dodać do niego już istniejący projekt, tak aby można było wreszcie mieć jakąś kontrolę nad kodem.

Importowanie projektu do repozytorium Subversion

Repozytorium Subversion to pojemnik na pliki i katalogi składające się na dany projekt. Można po prostu sprawdzać katalogi wprost w repozytorium, ale lepszym rozwiązaniem będzie opracowanie oddzielnej struktury katalogów dla każdego projektu.

Aby sprawdzić katalog projektu, najpierw należy usunąć z niego wszystkie niepotrzebne pliki niezawierające kodu (takie jak pliki obiektowe, tymczasowe, binaria itd.). Następnie trzeba zaimportować dany projekt do Subversion, używając w tym celu polecenia *svn import*. W poniższym przykładzie zostanie zaimportowany projekt dla języka C „Witaj świecie!”.

```
$ svn import witaj file:///var/svn/witaj -m "tworzenie projektu witaj"
Authentication realm: <http://192.168.6.90:80> Subversion Repository Password for 'wvh':
Adding witaj/witaj.c
```

Argumentem dla polecenia *svn import* jest nazwa projektu, który ma zostać zaimportowany; adres URL wskazuje, gdzie projekt ma zostać umieszczony, a opcja *-m* (ang. *message* — wiadomość) umożliwia umieszczenie treści komunikatu zamkniętego w cudzysłowach. Opcja ta nie jest konieczna, ale pozwoli zaoszczędzić trochę czasu, ponieważ

w innym przypadku użytkownik będzie i tak musiał wprowadzić tekst wiadomości w edytorze, czego wymaga svn.

Po utworzeniu projektu i sprawdzeniu plików za pomocą polecenia `svn list` można wyświetlić zawartość danego repozytorium, tak jak w poniższym przykładzie, w którym dodatkowo zastosowano jeszcze opcję `-R`, powodującą rekursywne wyświetlenie zawartości:

```
$ svn list -R file:///var/svn
witaj/
witaj/witaj.c
```

Można zauważyć, że adres URL rozpoczyna się od zapisu *file:///*, co oznacza, że repozytorium znajduje się w lokalnym systemie plików. Jest to o tyle konieczne, że do tej chwili nie został uruchomiony serwer Subversion, dlatego wszystkie adresy muszą wskazywać bieżący system plików. Próba uzyskania dostępu do repozytoriów poprzez adres sieciowy przyniesie komunikat błędu o treści `svn: PROPFIND of '/svn': could not connect to server` w przypadku, kiedy serwer sieciowy nie będzie uruchomiony, lub `svn: PROPFIND of '/svn': 405 MethodNotAllowed`, jeżeli serwer będzie działał, ale nie będzie na nim żadnego repozytorium. Poniżej omawiam dwa rodzaje serwerów dostępnych w Ubuntu, a także metody ich konfigurowania i udostępniania za ich pomocą repozytoriów innym komputerom.

Konfigurowanie serwera Subversion

Chociaż Subversion oparte jest na WebDAV i do podstawowej komunikacji pomiędzy klientem a serwerem używany jest protokół HTTP, to do lokalnego repozytorium (czyli takiego, które znajduje się fizycznie na tym samym komputerze, z którego korzysta użytkownik) dostęp jest możliwy bez uruchamiania serwera Subversion. Jego uruchomienie będzie jednak konieczne, jeżeli repozytorium ma być udostępniane innym komputerom. Dla Subversion dostępne są dwa podstawowe serwery.

- ♦ **Apache2** — ten serwer wyposażony jest w zintegrowane wsparcie dla Subversion. Obsługę WebDAV łatwo dodać do Apache czy Apache2, co z kolei umożliwi dostęp do danego repozytorium systemu Subversion poprzez sieć za pomocą adresów typu *http://nazwa-hosta/repozytorium* czy *https://nazwa-hosta/repozytorium* (drugi w przypadku, jeżeli w Subversion i Apache 2 zostały skonfigurowane z obsługą protokołu SSL). Ten przykład konfiguracji i integracji Apache 2 (który jest nowszą i wydajniejszą wersją Apache) omówię poniżej.
- ♦ **Svnserve** — samodzielny serwer umożliwiający dostęp do repozytoriów Subversion poprzez sieć z wykorzystaniem adresów typu *svn://nazwa-hosta/repozytorium*. Binarna wersja tego serwera jest częścią pakietu *subversion*.

Te dwa serwery nie wykluczają się wzajemnie i mogą działać jednocześnie na tym samym komputerze, umożliwiając dostęp do poprawnie skonfigurowanego repozytorium. Poniżej napiszę, w jaki sposób skonfigurować oba te serwery tak, aby udostępniały proste repozytorium, które zostało utworzone w zamieszczonym powyżej przykładzie.

Konfigurowanie serwera Apache 2 do obsługi Subversion

Konfigurowanie serwera Apache 2 do obsługi Subversion jest proste, wymaga jedynie zainstalowania i skonfigurowania odpowiedniego modułu oraz utworzenia pliku uwierzytelniania, który będzie kontrolował prawa użytkowników do zapisu i modyfikacji danych w repozytorium.

Instalowanie Apache 2 z obsługą Subversion omówiłem wcześniej w podpunkcie zatytułowanym „Instalowanie Subversion”; wystarczy dodać do systemu za pomocą menedżera pakietów Synaptic lub narzędzia `apt-get` pakiet *libapache2-svn*. Obsługa Subversion przez Apache 2 jest możliwa dzięki wczytaniu pliku */etc/apache2/modules-available/dav_svn.load*. Plik konfiguracyjny dla tego modułu można znaleźć w */etc/apache2/modules-available/dav_svn.conf*.

Po zainstalowaniu obsługi Subversion w serwerze Apache należy przeprowadzić edycję wymienionego pliku, do tego celu potrzebne będą narzędzie `sudo` oraz ulubiony edytor tekstu. Opcje konfiguracyjne są następujące.

1. Jeżeli repozytorium ma być dostępne pod adresem innym niż `http://serwer-sieciowy/svn`, należy zaktualizować wpis `<Location /svn>`.
2. Aby aktywować repozytorium, należy usunąć znak komentarza w linii `DAV svn`.
3. Należy poprawić wpis `SVNPath` tak, aby wskazywał pełną ścieżkę dostępu do repozytorium.
4. Aby umożliwić podstawowe uwierzytelnienie HTTP, należy usunąć znak komentarza z następujących trzech linii:

```
AuthType Basic
AuthName "Subversion Repository"
AuthUserFile /etc/apache2/dav_svn.passwd nazwa-uzytkownika
```

5. Aby weryfikować hasła podczas uwierzytelniania HTTP, należy zmodyfikować nazwę ścieżki we wpisie `AuthUserFile`.
6. Trzeba usunąć istniejący plik uwierzytelnienia i w jego miejsce utworzyć nowy, używając poniższego polecenia, gdzie *nazwa-uzytkownika* to nazwa pierwszego użytkownika, dla którego zostanie utworzone hasło:

```
$ sudo rm -f /etc/apache2/dav_svn.passwd
$ sudo htpasswd2 -c /etc/apache2/dav_svn.passwd nazwa-uzytkownika
New password:
Re-type new password:
```

7. Należy też dodać kolejnych, nowych użytkowników, używając poniższego polecenia:

```
$ sudo htpasswd2 /etc/apache2/dav_svn.passwd nazwa-uzytkownika
```

Po zakończeniu edycji plik będzie wyglądał mniej więcej tak jak na poniższym przykładzie, z którego usunięto komentarze, aby stał się czytelniejszy:

```
<Location /svn>
DAV svn
SVNPath /var/svn
AuthType Basic
AuthName "Subversion Repository"
```

```
AuthUserFile /etc/apache2/dav_svn.passwd
<LimitExcept GET PROPFIND OPTIONS REPORT>
    Require valid-user
</LimitExcept>
</Location>
```

Teraz wystarczy już tylko uruchomić ponownie serwer Apache (co można zrobić za pomocą polecenia `sudo /etc/init.d/apache2 restart`), po tym zabiegu repozytorium Subversion będzie dostępne w sieci pod adresem typu `http://adres-serwera/svn`. Więcej informacji na temat aktualizowania plików za pomocą repozytorium zamieszczam w tym rozdziale, w podpunkcie „Pobieranie plików z repozytoriów Subversion”.

Konfigurowanie serwera svnserve do obsługi Subversion

Jak wspominałem wcześniej, serwer svnserve umożliwia klientom Subversion łączenie się za pomocą protokołu svn zamiast HTTP. Serwer svnserve jest częścią pakietu Subversion instalowaną wraz z głównym programem.

Serwer svnserve można uruchomić samodzielnie lub jako demona (inetd czy xinetd). Program nie wymaga żadnej specjalnej konfiguracji, ponieważ to użytkownik wskazuje główne repozytorium w linii poleceń czy pliku konfiguracyjnym dla każdego demona usług internetowych, który ma być uruchomiony. Aby uruchomić demona svnserve z linii poleceń i umożliwić dostęp do repozytorium utworzonego zgodnie z zamieszczonymi wcześniej wskazówkami, wystarczy wykonać następujące polecenie:

```
$ sudo -u www-data /usr/bin/svnserve -r /var -d
```

Powoduje ono uruchomienie serwera dla użytkownika `www-data`, opcja `-r` została użyta do zidentyfikowania głównego repozytorium w katalogu `/var`, natomiast opcja `-d` powoduje uruchomienie demona w tle.

Jak wspominałem wcześniej, program umożliwia też skonfigurowanie demona usług internetowych do obsługi nadchodzących zapytań dla adresów typu `svn`., nie trzeba go wtedy każdorazowo uruchamiać samodzielnie czy dodawać do usług startowych. W Ubuntu dostępne są dwa demony internetowe: `inetd` i `xinetd`. Demon `inetd` jest rozwiązaniem starszym, podczas gdy `xinetd` jest nowszy i bardziej elastyczny. Żaden z nich nie jest domyślnie instalowany w Ubuntu, więc decydując się na któryś z nich (sugeruję `xinetd`), trzeba będzie go zainstalować za pomocą menedżera pakietów Synaptic lub narzędzia `apt-get`.

Jeżeli użytkownik zainstaluje `xinetd`, trzeba będzie również utworzyć plik konfiguracyjny `/etc/xinetd.d/svn`, dzięki któremu demon będzie podejmował odpowiednie działania, kiedy na porcie 3690 pojawi się zapytanie, zgodnie z tym, co zdefiniowano w pliku `/etc/services/`. A tak wygląda zawartość pliku `/etc/xinetd.d/svn`, przy którego użyciu użytkownicy z zewnątrz mają dostęp do repozytorium utworzonego wcześniej:

```
service svn
{
    disable = no
    socket_type = stream
    protocol = tcp
    wait = no
    user = www-data
```

Obsługa adresów typu svn: w Firefoksie

Podczas próby przejścia w Firefoksie pod adres rozpoczynający się od `svn` użytkownik zostanie uruchomiony następującym komunikatem: *Firefox nie jest w stanie otworzyć tego adresu, ponieważ protokół „svn” nie jest przypisany do żadnego programu.* Aby ominąć tę przeszkodę, należy utworzyć niewielki skrypt, dzięki któremu adresy URL będą traktowane w programach klienckich jako argumenty. Używam poniższego skryptu do automatycznego uruchamiania klienta `kdesvn` (jako jedyny akceptuje argumenty z linii poleceń) wówczas, gdy zostanie wywołany adres typu `svn:`:

```
#!/bin/bash
/usr/bin/kdesvn "$1"
```

Po zapisaniu skryptu jako pliku i nadaniu mu atrybutu wykonywalności należy jeszcze zmodyfikować Firefoksa tak, aby uruchamiał skrypt po wpisaniu adresu typu `svn:`.

1. W pasku adresu przeglądarki należy wpisać `about:config` i wcisnąć klawisz *Enter*.
2. Trzeba kliknąć prawym przyciskiem myszy w wolnym polu i z menu kontekstowego wybrać kolejno: *Dodaj ustawienie typu/Łańcuch (string)*.
3. Jako nazwę łańcucha należy wpisać `network.protocol-handler.app.svn`, a następnie kliknąć *OK*.
4. Jako wartość łańcucha trzeba podać pełną ścieżkę dostępu do pliku z zapisanym uprzednio skryptem.

Nie jest to specjalnie ekscytujące rozwiązanie, ale pozwala zapobiec zdenerwowaniu w razie próby skorzystania z adresu typu `svn:`. Byłoby dobrze, gdyby programy klienckie `esvn` lub `rapidsvn` obsługiwały argumenty linii poleceń, ponieważ program `kdesvn` do działania wymaga bibliotek KDE. Może któregoś dnia...

```
server = /usr/bin/svnserve
server_args = -i -r /var
}
```

Po utworzeniu tego pliku należy uruchomić ponownie demona `xinetd`, używając poniższego polecenia:

```
$ sudo /etc/init.d/xinetd restart
```

Pobieranie plików z repozytoriów Subversion

Teraz, kiedy są już skonfigurowane serwery dla Subversion, można uzyskać dostęp do przykładowego repozytorium z dowolnego komputera, mogącego połączyć się z maszyną, na której jest ono umieszczone. Aby pobrać z niego projekt *witaj*, można użyć następujących poleceń.

- ♦ `svn co file:///var/svn/witaj` — standardowe polecenie służące do pobierania projektu. Można go użyć tylko na tym komputerze, na którym znajduje się repozytorium Subversion.
- ♦ `svn co http://192.168.6.90/svn/witaj` — polecenie służące do pobrania projektu za pośrednictwem protokołu HTTP; zadziała tylko wówczas, jeżeli na komputerze, na którym znajduje się repozytorium, jest uruchomiony serwer sieciowy (zgodnie z informacjami zawartymi w podpunkcie „Konfigurowanie serwera Apache 2 do obsługi Subversion”).

- ♦ `svn co svn://192.168.6.90/svn/witaj` — polecenie służące do pobrania projektu za pośrednictwem protokołu SVN; zadziała tylko wówczas, jeżeli na komputerze, na którym znajduje się repozytorium, jest uruchomiony serwer Subversion (zgodnie z informacjami zawartymi w podpunkcie „Konfigurowanie serwera svnserve do obsługi Subversion”).

Wykonanie któregośkolwiek z powyższych poleceń spowoduje utworzenie w bieżącym katalogu podkatalogu o nazwie *witaj*, w którym zostanie umieszczony plik *witaj.c*. Gratulacje — Subversion właśnie zaczęło działać!

Dokonywanie zmian w repozytoriach Subversion

Po pobraniu plików z repozytorium i dokonaniu w nich zmian użytkownik zapewne będzie chciał je umieścić z powrotem tam, skąd zostały pobrane. Repozytorium to odpowiednie miejsce do przechowywania najnowszej i najlepszej wersji kodu źródłowego, a w dodatku umożliwia śledzenie wprowadzanych zmian — bo przecież właśnie do tego służy system kontroli wersji.

Za pomocą systemu łatwo śledzić wprowadzane zmiany, ponieważ przechowuje on informacje o takich działaniach. Jeżeli w katalogu Subversion, w którym znajduje się katalog projektu, zostanie użyte polecenie `ls -al`, wówczas będzie można zauważyć pliki z rozszerzeniem `.svn`, w których te zmiany są przechowywane.

Ponieważ Subversion zarządza tymi informacjami, wyręczając użytkowników, muszą oni tylko umieścić pliki z wprowadzonymi zmianami z powrotem w repozytorium za pomocą polecenia `ci` (ang. *check in*). W poniższym przykładzie z powrotem w repozytorium zostanie umieszczona poprawiona wersja pliku *witaj.c*, zostaną też wyświetlone dane wyjściowe:

```
$ svn ci witaj.c -m "Dodano komentarz"
Sending          witaj.c
Transmitting file data .
Committed revision 7.
```

Opcja `-m` umożliwia umieszczenie własnego komunikatu. Jeżeli nie zostanie użyta, Subversion automatycznie uruchomi edytor tekstu, w którym użytkownik będzie mógł wprowadzić komentarz.

Dodatkowe informacje o Subversion

Subversion to użyteczny system kontroli wersji kodu wyposażony w wiele różnych funkcji omówionych pokrótce powyżej. Oto kilka przydatnych stron zawierających informacje na temat Subversion:

- ♦ <http://svnbook.red-bean.com>, książka — internetowa wersja książki wydawnictwa O'Reilly poświęcona systemom kontroli wersji i Subversion; choć w sieci dostępna jest jej kopia, wygodnie też posiadać jej wersję papierową i podziękować w ten sposób jej autorom za trud wniesiony w jej napisanie,
- ♦ <http://subversion.tigris.org/faq.html> — lista najczęściej zadawanych pytań (i odpowiedzi) na temat Subversion; znakomite źródło informacji o omawianym systemie kontroli wersji,

- ♦ <http://subversion.tigris.org> — strona domowa projektu zawierająca najnowsze informacje na temat Subversion, których można spodziewać się na macierzystej witrynie,
- ♦ irc.freenode.net, kanał #svn — kanał IRC poświęcony Subversion, tu można znaleźć odpowiedzi na wszelkie nurtujące użytkowników pytania,
- ♦ lista dyskusyjna poświęcona systemowi Subversion — istnieje kilka różnych list dyskusyjnych; announce@subversion.tigris.org to lista o niewielkim natężeniu ruchu, w której zamieszczane są informacje o nowych wydaniach lub uaktualnieniach. Lista dev@subversion.tigris.org przeznaczona jest dla programistów Subversion. Lista users@subversion.tigris.org to z kolei miejsce, gdzie wiele się dzieje, ponieważ przeznaczona jest dla użytkowników systemu, mogą oni tu zadawać pytania i szukać pomocy u innych. Aby subskrybować którąkolwiek z list, należy wysłać wiadomość na adres nazwalisty-subscribe@subversion.tigris.org, a następnie postępować zgodnie z instrukcjami, które zostaną przesłane w wiadomości zwrotnej. Wymienione listy są moderowane, dlatego może nastąpić pewne opóźnienie między wysłaniem wiadomości, a jej pojawieniem się na liście.

W sieci można znaleźć wiele stron dotyczących Subversion; proste wyszukiwanie za pomocą Google czy innej wyszukiwarki dostarczy materiału na długie tygodnie. Indyjski magazyn „Linux For You” (www.linuxforu.com) opublikował ciekawy artykuł o Subversion i chociaż nie udało mi się znaleźć go w archiwach serwisu, to jest dostępny na stronie autora: http://mia.ece.uic.edu/~papers/publications/subversion_final.pdf.

Podsumowanie

Oczywiście, użytkownik nie musi tworzyć samodzielnie oprogramowania przeznaczonego dla Linuksa, do swojej dyspozycji ma kompletny zestaw aplikacji, bibliotek oraz jądro dla całej dystrybucji. Ale od czasu do czasu może się zdarzyć, że jednak będzie chciał skompilować własny, doskonały program i wyposażać go w najnowsze dostępne funkcje; być może czytelnik jest programistą i chciałby napisać własny, otwarty (mam nadzieję) program, ponieważ to jest właśnie jego praca. W niniejszym rozdziale napisałem, jak skompilować własne programy napisane w języku C i C++ w systemie Ubuntu i wykorzystać w tym celu odpowiednie narzędzia (make) oraz biblioteki. W rozdziale omówiłem też zintegrowane środowiska programistyczne dostępne dla systemów linuksowych, skupiając się zwłaszcza na popularnym, potężnym i otwartym Eclipse. Na końcu rozdziału zamieściłem omówienie systemów kontroli wersji kodu; konfigurowania własnego repozytorium oraz serwera Subversion. Opisałem też, jak szybko i łatwo udostępnić takie repozytorium w sieci dzięki serwerowi Apache.

W rozdziale 19. napiszę, co się dzieje w trakcie procesu uruchamiania i zamykania systemu Linux. Zrozumienie mechanizmu działania procesu startowego jest ważne, aby implementować nowe usługi oraz funkcje do systemu. Wiedza na temat tego, gdzie i jak są uruchamiane nowe usługi, jest ważna również do optymalizowania systemu, ponieważ można wyłączyć odpowiednie usługi i zaoszczędzić nieco mocy obliczeniowej procesora. Natomiast zrozumienie procesu wyłączania systemu pozwoli w przyszłości bezpiecznie wyłączać różne usługi w odpowiednim porządku, zatem będzie można uniknąć uruchamiania usług do porządkowania systemu za każdym razem, kiedy włączany jest komputer.

Część III

Ubuntu dla administratorów systemu

W tej części:

Rozdział 19. Proces rozruchu i zamykania systemu Ubuntu

Rozdział 20. Dodawanie, usuwanie i aktualizacja oprogramowania

Rozdział 21. Zarządzanie użytkownikami, grupami i zaawansowane uprawnienia

Rozdział 22. Wykonywanie i przywracanie kopii zapasowych

Rozdział 23. Dodawanie sprzętu i przyłączanie urządzeń peryferyjnych

Rozdział 24. Konfiguracja sieci i bezpieczeństwo

Rozdział 25. Sieć bez kabli

Rozdział 19.

Proces rozruchu i zamykania systemu Ubuntu

W tym rozdziale:

- ♦ Proces rozruchowy Ubuntu
- ♦ Monitorowanie procesu rozruchowego Linuksa
- ♦ Optymalizacja procesu rozruchowego Linuksa
- ♦ Proces zamykania Linuksa

Obserwując uruchamiający się system Ubuntu Linux widzimy przyjazny graficzny ekran wyświetlający różne komunikaty i paski postępu prezentujące poszczególne etapy procesu rozruchowego, inicjacji i uruchamiania systemu. To bardzo sympatyczne, ale czasem warto wiedzieć, co się dzieje pod maską. Interesujące jest również to, jakie działania Ubuntu podejmuje w standardowym procesie zamykania systemu, czyli w wyniku wywołania polecenia *System/Zakończ* i kliknięciu ikony *Wyłącz komputer* („proces” nieporządnego zamykania systemu, polegający na wyjęciu z gniazda wtyczki zasilania jest znacznie prostszy: być! i system jest zamknięty, ale za to proces rozruchowy przy następnym uruchomieniu komputera będzie nieco bardziej skomplikowany, niż zwykle, ponieważ system musi sprawdzić, czy nie wystąpiły problemy w systemie plików i naprawić to, co uda się naprawić w sposób automatyczny).

W tym rozdziale opisuję ogólne etapy procesu rozruchowego Linuksa oraz specyficzne mechanizmy zastosowane w dystrybucji Ubuntu Linux. W dalszej części podaję metody zdobycia szczegółowych informacji o procesach uruchamianych przy starcie systemu oraz usprawnienia planowane w nowszych wersjach dystrybucji, które pozwolą przyspieszyć proces rozruchowy. Na końcu tego rozdziału omawiam proces zamykania systemu. Ten fragment jest bardzo krótki, bo w rzeczywistości proces zamykania systemu stanowi w zasadzie odwrócenie procesu jego uruchamiania.

Proces rozruchowy dystrybucji Ubuntu Linux

W tym podrozdziale opiszę proces uruchamiania systemu Ubuntu Linux, od włączenia komputera do momentu, gdy system jest uruchomiony i czeka na działania użytkownika. Starałem się opisać ten proces w sposób ogólny, dla różnych typów sprzętu obsługiwanego przez Ubuntu, zatem niektóre z punktów mogą nie dotyczyć systemu użytkowanego przez czytelnika. Wszystkie poniższe punkty dotyczą przede wszystkim systemów zgodnych z architekturą x86.

Boot Monitor lub BIOS

Pierwszym etapem po włączeniu komputera jest uruchomienie tzw. boot monitora lub systemu *Basic Input/Output System* (BIOS), zainstalowanego bezpośrednio w układach elektronicznych płyty głównej. To oprogramowanie jest z reguły zapisane w pamięci flash znajdującej się na płycie głównej. BIOS lub boot monitor to niskopoziomowe oprogramowanie dostosowane do obsługi sprzętu płyty głównej, którego zadaniem jest pierwotna inicjalizacja, wstępne testy i weryfikacja sprawności sprzętu (często opcjonalna). Na tym etapie komputer pozwala użytkownikowi na przerwanie procedury rozruchowej (przez określony czas), po czym przechodzi do załadowania do pamięci innego programu. Ten „inny program” to najczęściej kod z głównego rekordu rozruchowego (ang. *Master Boot Record*, MBR) dysku twardego, pamięci flash lub napędu zewnętrznego np. CD-ROM, który po załadowaniu przejmuje kontrolę nad procesem ładowania systemu.

W systemach o ograniczonych zasobach systemowych, takich jak systemy embedded, programem ładowanym do pamięci przez boot monitor jest samo jądro Linuksa, co zostało opisane w kolejnym punkcie. W przypadku większości systemów biurkowych i serwerowych proces rozruchowy jest nieco bardziej skomplikowany. W tego typu systemach, w pierwszym etapie rozruchu, do pamięci ładowany jest program pośredni zapisany w MBR, zawierający dodatkową konfigurację i opcje dostosowujące proces rozruchowy. W nowoczesnych komputerach PC i SPARC program ładowany na tym etapie nosi nazwę boot loadera lub programu rozruchowego, który jest konfigurowalną aplikacją stanowiącą mechanizm wspomagający ładowanie właściwego systemu operacyjnego. W Linuksie najczęściej stosowanym programem rozruchowym dla systemów z rodziny x86 (również 64-bitowych) jest *Grand Unified Boot Loaders* (GRUB), choć niektóre dystrybucje nadal wykorzystują starszy program rozruchowy o nazwie *Linux Loader* (LiLo). Biurkowe komputery w architekturze PowerPC i SPARC z zainstalowanym Linuksem nie wykorzystują tych programów rozruchowych, lecz własne, specyficzne dla platformy programy rozruchowe ładowane przez boot monitor.

Program rozruchowy

Program rozruchowy swoje zadanie realizuje z reguły w dwóch etapach, dlatego składa się z dwóch osobnych programów ładujących: pierwszy z nich jest minimalistyczny, natomiast drugi realizuje bardziej zaawansowane funkcje. Dzieje się tak dlatego, że rozmiar pierwszego programu ładującego jest ograniczony rozmiarem MBR, który jest równy

rozmiarowi fizycznego sektora na dysku (czyli wynosi 512 bajtów). Program ładujący pierwszego etapu zawiera z reguły wystarczająco dużo informacji, aby załadować program ładujący drugiego etapu, który obsługuje właściwe zadanie zainicjowania rozruchu systemu operacyjnego.

Jak wspomniałem wcześniej, Linux oferuje dwa programy rozruchowe: LiLo i GRUB. Program LiLo był rozwijany na długo przed powstaniem projektu GRUB i nie jest w stanie obsłużyć formatów systemów plików stosowanych w Linuksie. Z tego powodu program ładujący drugiego etapu (*/boot/boot.b*) jest w LiLo identyfikowany na podstawie fizycznego adresu na urządzeniu dyskowym. Program rozruchowy GRUB zastosowany w Ubuntu również wykorzystuje dwa programy ładujące, ale dodatkowo stosuje etap pośredni. GRUB w pierwszym etapie rozruchu ładuje program ładujący etapu 1.5 (stage 1.5), który zawiera procedury odczytu z określonego systemu plików, dzięki czemu możliwe jest wczytanie tekstowego programu konfiguracyjnego programu GRUB bezpośrednio z głównego systemu plików Linuksa (*root filesystem*). GRUB obsługuje wiele różnych typów programów ładujących etapu 1.5:

- ♦ *e2fs_stage1_5*, obsługa systemu plików ext2 lub ext3;
- ♦ *fat_stage1_5*, obsługa systemu plików VFAT znanego z DOS i Microsoft Windows;
- ♦ *ffs_stage1_5*, obsługa systemu plików Berkeley Fast;
- ♦ *iso9660_stage1_5*, obsługa systemu plików ISO9660, wykorzystywanego na dyskach CD-ROM;
- ♦ *jfs_stage1_5*, obsługa systemu plików JFS;
- ♦ *minix_stage1*, obsługa systemu plików Minix;
- ♦ *reiserfs_stage1_5*, obsługa systemu plików ReiserFS;
- ♦ *ufs2_stage1_5*, obsługa tradycyjnego systemu plików Unix;
- ♦ *vstafs_stage1_5*, obsługa systemu plików Plan 9 VSTa;
- ♦ *xfs_stage1_5*, obsługa systemu plików XFS.

W domyślnej instalacji systemy Ubuntu zawierają programy ładujące fazy 1.5 obsługujące systemy plików ext2/ext3, JFS, Minix, ReiserFS, VFAT i XFS.

Program ładujący fazy 1.5 ładuje bezpośrednio z systemu plików właściwy program ładujący drugiej fazy oraz jego pliki konfiguracyjne.



Zasadnicza różnica między programami rozruchowymi LiLo i GRUB polega na tym, że ten pierwszy wymaga, aby jego konfiguracja była zapisana bezpośrednio w programie rozruchowym, podczas gdy w drugim może być zapisana w pliku na dysku, skąd jest bez problemu wczytywana. Oba te programy rozruchowe wykorzystują pliki konfiguracyjne, ale w przypadku LiLo po każdej modyfikacji pliku konfiguracyjnego (*/etc/lilo.conf*) należy wywołać program */sbin/lilo* zapisujący parametry opisujące opcje menu rozruchowego i parametry jądra w pliku *boot.b*, którego fizyczne położenie na dysku jest następnie zapisywane w MBR. Programy ładujące fazy 1.5 programu GRUB pozwalają uniknąć tego etapu przez bezpośredni odczyt konfiguracji z pliku zapisanego w systemie plików. Dzięki temu wystarczy jedynie zmodyfikować plik konfiguracyjny, aby GRUB przy następnym rozruchu systemu zastosował się do tych zmian, nie ma potrzeby wykonywać jakichkolwiek dodatkowych konwersji.

Program ładujący etap drugiego wyświetla na ekranie menu pozwalające użytkownikowi na wybór jednej z kilku opcji rozruchowych, np. może wybrać uruchomienie jednego z kilku systemów operacyjnych zainstalowanych w komputerze. Opcja wyświetlania menu rozruchowego jest obsługiwana przez GRUB i LiLo, istnieje możliwość ukrycia tego menu z opcją ustalenia czasu oczekiwania na wymuszenie jego wyświetlenia (np. przez naciśnięcie klawisza *Esc*). Po uruchomieniu Linuksa można zmienić parametry wyświetlania menu rozruchowego przez odpowiednią modyfikację pliku konfiguracyjnego programu rozruchowego, czyli */etc/lilo.conf* w przypadku LiLo, lub */boot/grub/menu.lst* dla programu GRUB.

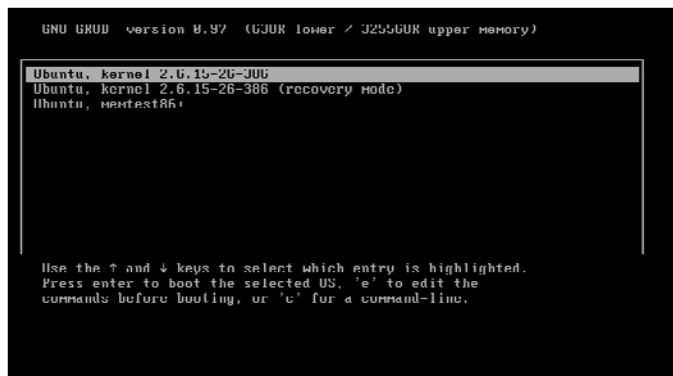


Ubuntu Linux wykorzystuje program rozruchowy GRUB, zatem na nim skupimy się w tym podrozdziale. Więcej informacji na temat LiLo (nadal wykorzystywanym w niektórych dystrybucjach, takich jak Xandros Linux) można znaleźć na stronie <http://www.freshmeat.net/projects/lilo>, która jest stroną poświęconą projektowi LiLo w ramach serwisu FreshMeat.

W domyślnej konfiguracji GRUB Ubuntu nie wyświetla menu rozruchowego, ale można je zobaczyć, naciskając klawisz *Esc* (GRUB daje na to pięć sekund). Przykładowe menu rozruchowe jednego z moich systemów Ubuntu przedstawiam na rysunku 19.1. Jeśli użytkownik nie naciśnie klawisza *Esc* w odpowiednim momencie procesu rozruchowego, GRUB automatycznie uruchomi system z użyciem domyślnej konfiguracji.

Rysunek 19.1.

Przykładowe menu rozruchowe programu GRUB



Niektóre dystrybucje Linuksa, takie jak Fedora Core, instalują menu programu GRUB znacznie bardziej estetyczne od przedstawionego na rysunku 19.1. Moja opinia na ten temat jest prosta: „A co to za różnica, skoro i tak nie chcę tego oglądać”. Menu programu GRUB jest za to przydatne w przypadku konfiguracji wielosystemowej, gdy istnieje powód, aby w jednym komputerze posiadać kilka systemów operacyjnych, uruchamianych na zmianę w miarę potrzeb. Oprócz tego przypadku, nie ma powodu, aby oglądać menu rozruchowe.



Za każdym razem, gdy w systemie instalowana jest nowa wersja jądra Linuksa, np. w wyniku działania aplikacji *Menedżer aktualizacji*, proces instalacji pakietu automatycznie aktualizuje plik konfiguracyjny menu rozruchowego i związane z nim pliki. Pozycja menu rozruchowego dotycząca poprzedniej wersji jądra jest nadal dostępna, dzięki czemu w przypadku problemów z nową wersją jądra system można uruchomić, wykorzystując starszą, sprawdzoną wersję jądra. Wystarczy nacisnąć klawisz *Esc* i z menu rozruchowego wybrać odpowiednią pozycję.

Pliki konfiguracyjne programu GRUB z reguły zawierają kilka pozycji definiujących różne sposoby uruchomienia systemu. Każdy zestaw reguł jest zapisany z użyciem określonego szablonu:

```
title          Ubuntu, kernel 2.6.15-23-386
root           (hd0,0)
kernel         /boot/vmlinuz-2.6.15-23-386 root=/dev/hda1 ro quiet splash
initrd         /boot/initrd.img-2.6.15-23-386
savedefault
boot
```

Każdy wiersz szablonu rozpoczyna się od słowa kluczowego identyfikującego jeden z elementów procesu rozruchowego. W powyższym przykładzie wyróżnimy następujące elementy:

- ♦ **title**: etykieta elementu wyświetlana w menu programu GRUB;
- ♦ **root**: identyfikuje bezwzględną lokalizację głównego systemu plików; definicja ta jest zapisywana niestandardowo, tzn. z użyciem wewnętrznej składni programu GRUB;
- ♦ **kernel**: określa plik jądra oraz parametry jego uruchomienia;
- ♦ **initrd**: wskazuje RAM-dysk lub systemu plików RAM, który powinien być załadowany przez kernel podczas rozruchu;
- ♦ **savedefault**: informuje, że tę pozycję GRUB potraktuje jako domyślną;
- ♦ **boot**: wskazuje programowi GRUB, że ma uruchomić system z użyciem powyższej konfiguracji.

Te słowa kluczowe to zaledwie podzbiór słów kluczowych obsługiwanych przez program GRUB. Pełną listę instrukcji można znaleźć w dokumentacji programu GRUB dostępnej w ramach pakietu *grub-doc*, który należy samodzielnie zainstalować z użyciem programu Synaptic lub `apt-get`. Po zainstalowaniu pakietu mamy dostęp do pełnej dokumentacji programu GRUB, wystarczy wywołać polecenie `info grub`. Jeszcze więcej informacji można znaleźć na stronie <http://www.gnu.org/software/grub>. Na tej stronie można również odnaleźć informacje o nowszej, udoskonalonej wersji programu GRUB, nazwanej GRUB2, która jest obecnie na etapie rozwojowym.

Ładowanie jądra

Jądro zdefiniowane w konfiguracji programu GRUB to typowy skompresowany obraz jądra, ładowany do pamięci. Uruchomienie tego jądra jest poprzedzone wywołaniem szeregu instrukcji realizujących wstępną inicjację sprzętu i konfigurację środowiska (takich jak definicja stosu, inicjacja tablicy stron, uruchomienie obsługi przestrzeni wymiany itp.), rozkompresowanie obrazu jądra do pamięci, inicjację ramdysku w pamięci oraz uruchomienie załadowanego jądra.

Jądro ustawia przerwania, dokonuje dalszej inicjacji sprzętu, po czym rozkompresowuje i montuje inicjujący ramdysk (*initial RAM disk*) lub inne systemy plików zdefiniowane w jądrze lub w parametrach przekazanych za pomocą słowa kluczowego `initrd` programu GRUB. Jeśli został odnaleziony lub wyspecyfikowany obraz dysku RAM lub systemu

plików RAM, system wykona sekwencję rozruchu opisaną w punkcie „Ładowanie i wykorzystanie inicjującego dysku RAM lub systemu plików RAM”. Jeśli nie został wskazany dysk RAM lub system plików RAM, jądro montuje główny system plików, zgodnie z parametrem `root`, po czym rozpoczyna standardowy proces inicjacyjny, zgodnie z opisem podanym w dalszej części rozdziału, w punkcie „Proces init”.

Ładowanie i wykorzystanie inicjującego dysku RAM lub systemu plików RAM

Wiele dystrybucji Linuksa w procesie rozruchowym przed zamontowaniem rzeczywistego systemu plików wykorzystuje specjalny typ systemu plików. Tego typu inicjujące systemy plików (*initial filesystem*) stosuje się opcjonalnie i są dostępne w dwóch odmianach: *initrd* (*initial RAM disk*, czyli inicjujący dysk RAM) oraz *initramfs* (*initial RAM filesystem*, czyli inicjujący system plików). W większości systemów biurkowych i serwerowych te inicjujące systemy plików są wykorzystywane do ładowania modułów jądra, takich jak specjalizowane sterowniki urządzeń, które są potrzebne na pierwszych etapach rozruchu systemu, ale nie są statycznie wkompiłowane w jądro. Tego typu specjalizowane funkcje inicjacyjne dla sprzętu i systemu są z reguły zapisywane w modułach dostępnych w inicjującym systemie plików. W Ubuntu, jak i w większości innych dystrybucji Linuksa, inicjujący dysk RAM i inicjujący system plików RAM są wykorzystywane w przejściowym etapie procesu rozruchowego. W konfiguracjach typu osadzonego (ang. *embedded*) mechanizmy *initramfs* i *initrd* mogą stanowić właściwy system plików w całym czasie działania systemu Linux.

Inicjujące dyski RAM są tradycyjnym mechanizmem wykorzystywanym w wielu instalacjach jądra 2.6 oraz wcześniejszych i umożliwiają ładowanie modułów jądra niezbędnych do rozruchu. Inicjujące dyski RAM to skompresowane pliki zawierające system plików, na którym zapisane są pliki niezbędne Linuksowi do rozruchu. System plików zapisany na takim wirtualnym dysku może być różnego typu: `ext2`, `romfs` itp. Dysk RAM jest rozpakowywany do wstępnie zaalokowanego obszaru w pamięci RAM, po czym jego system plików jest montowany przez jądro. Podstawowy problem tego rozwiązania wiąże się ze zużyciem dużej ilości pamięci oraz z faktem, że stanowi urządzenie blokowe, przez co w jądrze musi być wkompiłowana obsługa wykorzystywanego w nim systemu plików.

W jądrze 2.6 wprowadzono koncepcję inicjującego systemu plików RAM (*initial ram filesystem*), zapisywanego bezpośrednio w pliku jądra i korzystającego z pamięci cache katalogów. Inicjujący system plików RAM może być wczytywany przez jądro z pliku, można go też wkompiłować bezpośrednio w plik jądra, podobnie jak dyski RAM. Różnica polega na tym, że inicjujące systemy plików RAM są skompresowane w formacie zgodnym z wynikiem polecenia `cpio`. Takie archiwa zawierają wszystkie pliki i katalogi niezbędne do realizacji funkcji inicjującego systemu plików RAM i są rozpakowywane bezpośrednio do pamięci cache plików i katalogów, skąd są montowane jako system plików. Wykorzystanie pamięci cache plików i katalogów w charakterze systemu plików pozwala na znaczną redukcję rozmiaru pamięci niezbędnej do obsługi inicjującego systemu plików RAM. Pamięć cache katalogów (*directory entry*, w skrócie *dentry*), to specjalny obszar jądra, do którego tymczasowo kopiowane są pliki i katalogi zapisane w systemie plików.

W przypadku inicjującego systemu plików RAM pliki i katalogi istnieją wyłącznie w tej pamięci cache (dokąd i tak musiałyby zostać skopiowane gdyby były zapisane na dysku w sposób „klasyczny”). Oszczędność pamięci bierze się stąd, że jądro nie musi obsługiwać żadnych dodatkowych struktur danych, oprócz standardowych mechanizmów wykorzystywanych do obsługi wszystkich systemów plików. Dla odróżnienia: w przypadku inicjujących dysków RAM jądro musi dodatkowo obsługiwać mechanizm obsługi systemu plików zapisanych w obrazie dysku. Inicjujący system plików RAM jest zatem przypadkiem zastosowania systemu plików Linux *tmpfs*.

Systemy Linux stosujące inicjujący dysk RAM po rozkompresowaniu i zamontowaniu tego systemu plików uruchamiają plik */linuxrc*. To najczęściej plik tekstowy zawierający polecenia powłoki, może to być także odmiana programu */sbin/init*, którego działanie jest zgodne z procesem inicjacji systemu opisanym w punkcie „Proces init”. Ten drugi przypadek występuje z reguły wówczas, gdy inicjujący dysk RAM jest wykorzystywany jako jedyny system plików systemu, np. w systemach osadzonych. W przypadku użycia inicjującego systemu plików RAM po załadowaniu systemu plików do pamięci uruchamiany jest plik */init*. Podobnie jak plik */linuxrc*, plik */init* jest najczęściej tekstowym plikiem zawierającym polecenia lub odmianą programu */sbin/init*.

W systemach biurkowych i serwerowych korzystających z głównego systemu plików na nośniku masowym, ostatnim etapem wykonywanym przez plik */linuxrc* lub */init* jest zamontowanie tego systemu plików (identyfikowanego przez parametr rozruchowy *root=*), oraz uczynienie go bieżącym głównym systemem plików (z wykorzystaniem polecenia *pivot_root* lub *chroot*). Następnie rozpoczyna się proces inicjacji systemu opisany w następnym punkcie.

Proces init

Po załadowaniu jądra i zamontowaniu systemu plików tradycyjne systemy Unix i Linux uruchamiają aplikację systemową o nazwie *init* (proces inicjujący), zapisaną w systemach Linux w */sbin/init*. Procesowi temu jest nadawany identyfikator 1, co można sprawdzić korzystając z polecenia *ps*. Program *init* często nazywa się procesem pierwotnym, ponieważ to on uruchamia wszystkie inne procesy w systemie. Proces *init* wczytuje konfigurację zapisaną w pliku */etc/inittab*, która definiuje sposób uruchamiania systemu.



Program *init* jest częścią pakietu aplikacji znanych pod nazwą *SysVinit*. Pakiet *SysVinit* wykorzystuje pakiet *initscripts*, zawierający skrypty wykorzystywane do uruchamiania systemu na różnych poziomach (ang. *runlevel*) oraz definiowania kolejności uruchamiania aplikacji w poszczególnych poziomach pracy systemu.

Większość biurkowych i serwerowych systemów Linux wykorzystuje koncepcję poziomów pracy (ang. *runlevel*). Dla każdego poziomu pracy systemu można zdefiniować osobny zestaw aplikacji oraz kolejność ich uruchamiania. Poszczególne poziomy pracy to nic innego jak profile systemowe pozwalające zdefiniować środowisko pracy w postaci domyślnego zestawu aplikacji i usług. Dzięki temu system można uruchomić na kilka różnych sposobów, w zależności od pożądanego trybu jego użycia. Przykładowo poziom 1. (ang. *runlevel 1*) jest z reguły znany jako tryb pojedynczego użytkownika i wykorzystuje się go do zadań administracyjnych. Na tym poziomie pracy uruchamiany jest tylko

Porównanie mechanizmów inicjującego dysku RAM i inicjującego systemu plików RAM

Oto najważniejsze różnice między inicjującymi dyskami RAM a inicjującymi systemami plików RAM.

- ♦ Do zbudowania inicjującego systemu plików RAM nie są potrzebne specjalne uprawnienia. Zbudowanie inicjującego dysku RAM wymaga z reguły uprawnień konta root, chyba że wykorzysta się specjalną aplikację, taką jak *genext2fs*.
- ♦ Inicjujące dyski RAM (*initrd*) występują w postaci skompresowanych plików zawierających obraz systemu plików w określonym formacie, takich jak ext2, romfs, cramfs, squashfs itp. Inicjujące systemy plików RAM (*initramfs*) występują w postaci skompresowanych archiwów w formacie cpio.
- ♦ Inicjujące dyski RAM muszą być przygotowane oprócz jądra i muszą być zidentyfikowane w trakcie procesu budowania jądra, jeśli ma być możliwe ich włączenie do jądra. Inicjujące systemy plików RAM mogą być wykorzystane w ten sam sposób, można je jednak konstruować dynamicznie podczas kompilacji jądra. W jądrach 2.6 i nowszych można podać katalog zawierający pliki, które mają tworzyć inicjujący system plików RAM, lub plik konfiguracyjny opisujący jego zawartość.
- ♦ Inicjujące dyski RAM alokują i wykorzystują rozmiar pamięci określony z góry na etapie konfiguracji i kompilacji jądra, niezależnie od rozmiaru plików i katalogów zapisanych w dysku RAM. Niewykorzystana przestrzeń pamięci jest zupełnie marnotrawiona. Inicjujące systemy plików RAM są alokowane bezpośrednio w przestrzeni adresowej jądra i cache katalogów (*dentry*), z tego powodu wymagają zaalokowania pamięci wyłącznie w takiej ilości, jaka jest niezbędna do wczytania zapisanych w nich plików i katalogów.
- ♦ Do zapisanego na dysku (nieskonsolidowanego z jądrem) pliku zawierającego inicjujący system plików RAM można z łatwością dodać skrypt lub inny plik. Wystarczy rozpakować plik za pomocą programu *gunzip*, użyć polecenia *cpio --append*, aby dodać plik do archiwum, po czym skompresować je ponownie z użyciem polecenia *gzip*. Jednak tego typu operacje należy wykorzystywać wyłącznie do celów testowych, ponieważ takie modyfikacje zostaną nadpisane w przypadku ponownej kompilacji jądra lub instalacji aktualizacji pakietów jądra. Plik można dodać na stałe do inicjującego systemu plików RAM, dopisując informacje o nim do katalogu lub skryptu, który konfiguracji jądra jest wskazywany dyrektywą `CONFIG_INITRAMFS_SOURCE`.
- ♦ Inicjujące dyski RAM automatycznie wykonują zapisany na nich plik */linuxrc*. Inicjujące systemy plików RAM automatycznie wykonują zapisany na nich plik */init*. Plik */init* jest wykonywany z ID procesu równym 1, tak samo jak proces *init* w systemie Linux.

Zawartości inicjujących dysków RAM lub inicjujących systemów plików RAM mogą się różnić między różnymi systemami, w zależności od ich konfiguracji sprzętowej. Biurkowe systemy Linux z reguły wykorzystują te systemy plików do ładowania modułów jądra niezbędnych na pierwszych etapach procesu rozruchowego. Osadzone systemy Linux często wykorzystują inicjujące dyski RAM lub inicjujące systemy plików RAM w charakterze rzeczywistych systemów plików pod nieobecność dysku twardego czy pamięci flash. Zarówno inicjujące dyski RAM, jak i systemy plików RAM stanowią implementację tymczasowego zasobu pamięci masowej, który nie daje możliwości trwałego zapisu zmian, tzn. wszelkie zmiany będą obowiązywać tylko do ponownego uruchomienia komputera. Inicjujące systemy plików RAM są znacznie efektywniejsze pod względem wykorzystania pamięci, dzięki czemu są mniej wymagające z punktu widzenia jądra.

podstawowy zestaw aplikacji niezbędnych administratorowi do wykonania zadań administracyjnych, np. związanych z konserwacją systemu. W Ubuntu wykorzystywane są następujące poziomy pracy systemu:

- ♦ 0: zatrzymanie;
- ♦ 1: tryb pojedynczego użytkownika;

- ♦ 2: domyślny poziom wieloużytkownikowy systemów Ubuntu/Debian;
- ♦ 3 – 5: poziomy niestandardowe, do wykorzystywania przez użytkownika;
- ♦ 6: ponowne uruchomienie.



Większość pozostałych dystrybucji Linuksa i inne dystrybucje oparte na Debianie wykorzystują jako domyślny poziom 3., czyli poziom pracy bez trybu graficznego. W tych dystrybucjach do uruchamiania trybu graficznego wykorzystuje się poziom 5.

W systemach stosujących *SysVInit* i *initcripts* używana jest następująca sekwencja działań: jądro wykonuje proces */sbin/init*, który wczytuje plik */etc/inittab*, w nim zapisana jest konfiguracja podstawowych procesów, jakie mają być uruchomione w systemie.

W pliku */etc/inittab* podstawowy poziom uruchamiania jest zapisany w wierszu zawierającym słowo kluczowe *initdefault*, np.:

```
id:2:initdefault:
```

W tym przypadku poziom pracy 2. jest zdefiniowany jako domyślny. W wierszu zawierającym słowo kluczowe *sysinit* znajduje się definicja skryptu systemowego odpowiedzialnego za procedurę inicjacji systemu, np.:

```
si::sysinit:/etc/init.d/rcS
```



W systemach Ubuntu polecenie */etc/init.d/rcS* uruchamia polecenie */etc/init.d/rc* z argumentem *S*. Skrypt ten z kolei uruchamia kolejno wszystkie skrypty z katalogu */etc/rcS.d*, których nazwy rozpoczynają się od litery *S*. W niektórych dystrybucjach Linuksa jest wykorzystywany tylko jeden skrypt inicjacyjny, do którego administrator dopisuje własne wywołania w celu uwzględnienia ich w procesie rozruchowym.

Po uruchomieniu skryptu inicjacyjnego dystrybucje Linuksa wykorzystujące pakiety *SysVInit* i *initcripts* mają zdefiniowane zachowanie systemu dla poszczególnych poziomów pracy. Kilka wierszy dalej w pliku */etc/inittab* znajdziemy np. informację na temat poziomu 2.:

```
l2:2:wait:/etc/init.d/rc 2
```

W tym wierszu zdefiniowane jest wywołanie skryptu */etc/init.d/rc* z parametrem 2 w przypadku uruchomienia systemu na poziomie 2. To wywołanie spowoduje, że skrypt *rc* wywoła w kolejności alfabetycznej odpowiednie skrypty z katalogu */etc/rc2.d*. Pliki w tym katalogu mają nazwy postaci *SNNazwa* i *KNNazwa* i są z reguły plikami tekstowymi (a właściwie dowiązaniem symbolicznymi do skryptów) zawierającymi polecenia powłoki. Pliki o nazwach rozpoczynających się na literę *S* są wywoływane podczas uruchamiania określonego poziomu działania systemu. Pliki o nazwach rozpoczynających się na literę *K* są wywoływane podczas zakończenia pracy na określonym poziomie działania. Polecenia są wywoływane w kolejności alfabetycznej wymuszonej liczbami *NN*. Część nazwy występująca po tych liczbach jest dowolna, ale najczęściej pochodzi od nazwy programu lub podsystemu, który jest obsługiwany przez dany skrypt.

Dla skryptów startowych w Ubuntu można przyjąć następujące dwa założenia.

- ♦ Po prawidłowym wywołaniu skryptów rozpoczynających się od ciągu znaków S40* powinny zostać zainicjowane wszystkie sterowniki urządzeń i zamontowane wszystkie lokalne systemy plików.
- ♦ Po prawidłowym wywołaniu skryptów rozpoczynających się od ciągu znaków S60* powinien być ustawiony zegar systemowy, zamontowane systemy plików NFS (zdefiniowane w pliku */etc/fstab*, w przypadku wykorzystania mechanizmu automatycznego montowania zostaną zamontowane w późniejszym etapie), a wszystkie systemy plików powinny być sprawdzone pod względem spójności i poprawności.

Te reguły należy wziąć pod uwagę, definiując własne skrypty startowe dla dowolnych poziomów działania systemu.

Systemy Linux uruchamiają te same polecenia na różnych poziomach, pliki definiujące środowisko na poszczególnych poziomach to z reguły dowiązania symboliczne do skryptów z podstawowego zbioru startowych skryptów systemowych, umieszczonego w katalogu */etc/init.d*. Nazwy dowiązań symbolicznych na poszczególnych poziomach wykorzystują konwencję *SNNnazwa* i *KNNnazwa*. Kolejność wykonywania skryptów jest zdefiniowana kolejnością alfabetyczną plików w katalogu. Nazwy właściwych plików w katalogu */etc/init.d* z reguły nie stosują się już do tej konwencji. Zastosowanie dowiązań symbolicznych, w miejsce właściwych plików, umożliwia w prosty sposób dodawanie i usuwanie wywołań dla poszczególnych poziomów działania systemu bez konieczności tworzenia duplikatów skryptów, pozwalając jednocześnie utrzymać jednolite, spójne repozytorium skryptów w katalogu */etc/init.d*. Modyfikacja odpowiedniego skryptu w tym katalogu powoduje, że ta zmiana będzie automatycznie zastosowana we wszystkich poziomach działania systemu, dla których zostało utworzone odpowiednie dowiązanie symboliczne.

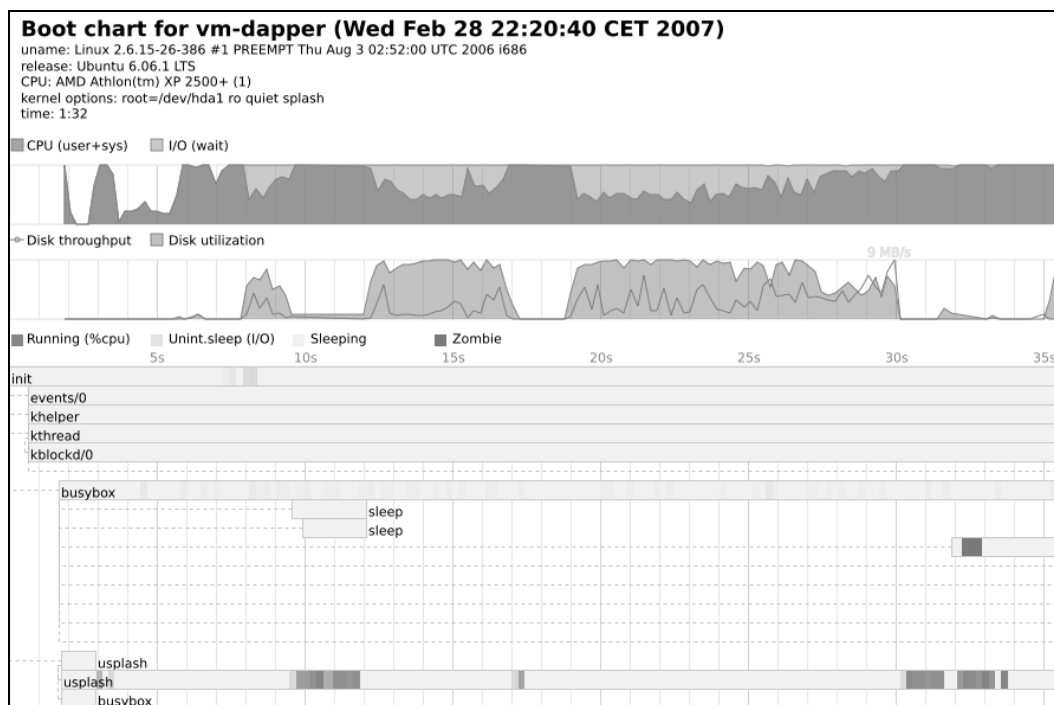
W ostatnim etapie procesu uruchamiania systemu, w dowolnym poziomie wieloużytkownikowym, proces *init* uruchamia skrypt */etc/rc.local*. Pozwala on użytkownikowi na zdefiniowania własnych wywołań bez konieczności ingerowania w skrypty startowe i dowiązania symboliczne.

W podrozdziale „Optymalizacja procesu rozruchowego dystrybucji Ubuntu”, w dalszej części tego rozdziału, można znaleźć informacje na temat aplikacji do zarządzania skryptami startowymi i zamykającymi oraz dowiązaniami definiującymi środowisko systemowe na poszczególnych poziomach działania systemu Ubuntu.

Analiza procesu startowego z użyciem programu Boot Chart

Program Boot Chart (<http://www.bootchart.org>), napisany przez Zigę Mahkoveca, jest imponującym i użytecznym narzędziem do monitorowania procesu rozruchowego i tworzenia jego graficznej reprezentacji. Fragment wygenerowanego przez ten program wykresu, obrazującego proces rozruchowy, przedstawiam na rysunku 19.2.

Boot chart nie jest domyślnie instalowany w Ubuntu, ale można zainstalować go samodzielnie, wykorzystując menedżer pakietów Synaptic lub program *apt-get*. Po zainstalowaniu



Rysunek 19.2. Fragment wyniku działania programu *Boot chart*

w Ubuntu, Boot Chart dopisuje się do inicjującego systemu plików RAM i konfiguruje go w taki sposób, żeby podczas procesu rozruchowego mógł zapisywać swoje statystyki. Boot chart dodaje do systemu również własny skrypt rozruchowy (*/etc/init.d/stop-bootchart* (oraz dowiązanie symboliczne do niego pod nazwą */etc/rc2.d/S99stop-bootchart*), który przetwarza zgromadzone dane i generuje graficzną reprezentację procesu rozruchowego.



W dystrybucjach Linuksa, które nie wykorzystują inicjującego systemu plików RAM (ale zawierających inicjujący dysk RAM) również istnieje możliwość wykorzystania aplikacji Boot Chart. W tym celu po instalacji programu należy zmodyfikować konfigurację programu GRUB w taki sposób, aby zamiast procesu *init* uruchamiał program *bootchartd*. Aby tak się stało, należy dopisać parametr jądra w postaci *init=/sbin/bootchartd*.

Boot Chart generuje plik graficzny w formacie *Portable Network Graphics* (PNG), zapisywany w katalogu */var/log/bootchart*. Nazwa pliku zawiera datę i czas jego utworzenia, dzięki czemu można porównywać wykresy różnych procesów rozruchowych. Program Boot Chart można skonfigurować tak, by zapisywał wykresy w formatach SVG i EPS (można też zablokować opcję usuwania wykorzystywanych przez program plików z danymi). W tym celu należy odpowiednio zmodyfikować skrypt */etc/init.d/stop-bootchart*.

Graficzna reprezentacja procesu rozruchowego to nie tylko ciekawy gadżet, ale przede wszystkim bardzo cenne narzędzie pomagające zidentyfikować skrypty rozruchowe, których nie potrzebujemy, zbędne wywołania procesów X Window itp. elementy środowiska ładowane podczas uruchamiania się systemu. Boot Chart nie zwiększa obciążenia systemu w znaczący sposób, a korzyści z generowanej przez niego statystyki mogą okazać się bardzo pomocne.

Optymalizacja procesu rozruchowego dystrybucji Ubuntu

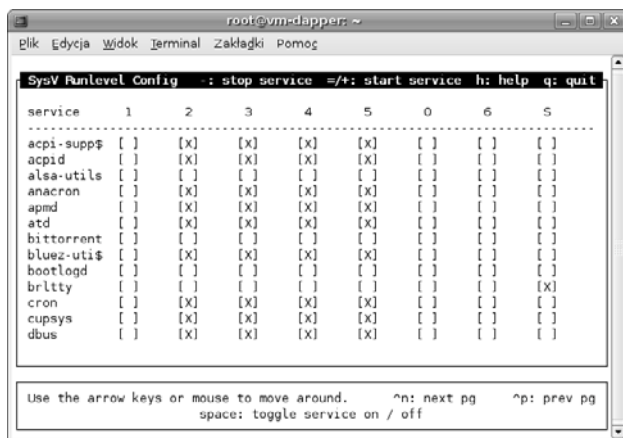
W punkcie „Proces init” wspomniałem, że proces `/sbin/init` do określenia domyślnego poziomu działania systemu wykorzystuje plik `/etc/inittab` i na podstawie tego liczbowego poziomu działania systemu określa jakie skrypty mają być uruchamiane podczas rozruchu systemu. Poziomy pracy systemu można modyfikować ręcznie, usuwając lub dodając odpowiednie dowiązania symboliczne, ale ten sposób jest bardzo żmudny. Jak można się domyślić, w repozytoriach Ubuntu znajdują się narzędzia ułatwiające zadanie konfiguracji skryptów uruchamianych w poszczególnych poziomach działania systemu. Standardowo w Ubuntu instalowany jest program (*System/Administracja/Usługi*), który umożliwia użytkownikowi kontrolę nad usługami uruchamianymi przez system. Narzędzie to nie udostępnia jednak precyzyjnej kontroli według poziomów uruchomienia, nie pozwala też ustalać kolejności w jakiej uruchamiane są usługi (Najdziwniejsze jest to, że starsza wersja tego narzędzia dawała taką możliwość. Jak widać postęp nie zawsze idzie *do przodu*).

Osobiście preferuję inne narzędzia do kontroli usług systemowych w Ubuntu. Oto one.

- ♦ **`sysv-rc-conf`**. Jest to aplikacja pracująca w trybie tekstowym, czyli można ją uruchomić w terminalu xterm lub terminalu GNOME, co przedstawiam na rysunku 19.3. Aplikacja *SysV Runlevel Config* posiada prosty w użyciu interfejs pozwalający aktywować i dezaktywować usługi na poszczególnych poziomach. Aplikacja oferuje też pomoc dotyczącą obsługi jej interfejsu.

Rysunek 19.3.

Aplikacja `sysv-rc-conf`
uruchomiona
w terminalu Gnome



- ♦ **Zarządca programów startowych**. Graficzna aplikacja oferująca doskonały interfejs do aktywowania i dezaktywowania skryptów startowych na poszczególnych poziomach. Na rysunku 19.4 przedstawiam przykładowy ekran programu *Zarządca programów startowych*.

Programy `sysv-rc-conf` i *Zarządca programów startowych* nie są instalowane automatycznie w ramach domyślnej instalacji Ubuntu, ale można je z łatwością zainstalować z użyciem menedżera pakietów Synaptic lub programu `apt-get`. Szczegółowe informacje na

Rysunek 19.4.
Aplikacja Zarządca programów startowych



temat instalowania pakietów z użyciem tych programów można znaleźć w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Aplikacja *Zarządca programów startowych* jest z pewnością ciekawsza pod warunkiem, że wykorzystuje się środowisko X Window. Po zainstalowaniu tej aplikacji w menu systemowym pojawi się pozycja *System/Administracja/Zarządca programów startowych*. Program do uruchomienia wymaga nieco czasu, ponieważ musi odczytać i przetworzyć każdy skrypt startowy zapisany w katalogu */etc/init.d* oraz sprawdzić każde dowiązanie symboliczne z katalogów */etc/rc* odpowiadających poszczególnym poziomom pracy systemu. Ten dodatkowy czas oczekiwania to niewiele, biorąc pod uwagę możliwości aplikacji.

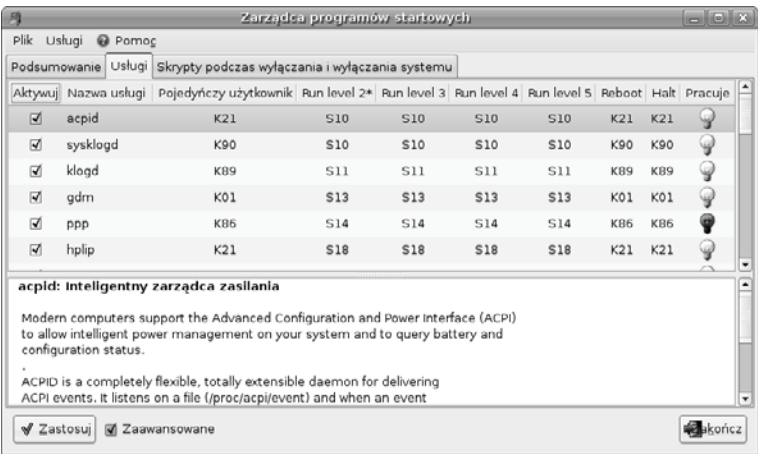
Po uruchomieniu programu *Zarządca programów startowych* mamy możliwość zapoznania się z podstawowymi informacjami na temat każdego ze skryptów startowych zainstalowanych w systemie, co przedstawia rysunek 19.4. Dzięki tym informacjom użytkownik może podejmować decyzje na temat tego, czy dana usługa jest mu potrzebna i czy w związku z tym powinien ją uaktywnić, czy może ją bezpiecznie wyłączyć. Po zaznaczeniu opcji *Zaawansowane* pojawią się dwie dodatkowe karty, które stanowią prawdziwą siłę programu. Na każdej z nich można znaleźć szczegółowe informacje, a do standardowej postaci z rysunku 19.4 łatwo powrócić, przechodząc na kartę *Podsumowanie*.

Na rysunku 19.5 przedstawiam kartę *Usługi*. Tu można znaleźć szczegółowe opisy usług obsługiwanych przez poszczególne skrypty startowe oraz informację o tym, na których poziomach pracy systemu dana usługa jest uruchamiana.

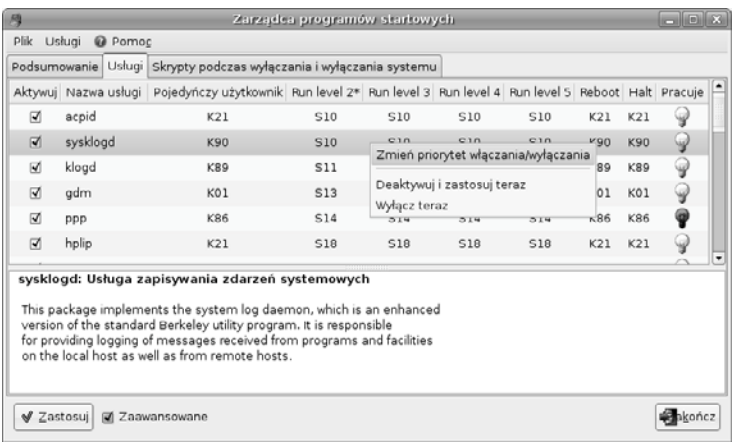
Każdą z usług można uruchomić lub zatrzymać, można je też usuwać z sekwencji startowej lub kończącej pracę na danym poziomie, można też zmodyfikować kolejność, w jakiej dana usługa będzie uruchamiana w stosunku do pozostałych usług systemowych. W tym celu należy kliknąć wybraną pozycję prawym przyciskiem myszy, co spowoduje wyświetlenie menu kontekstowego usługi, tak jak na rysunku 19.6.

Aby zmienić sekwencję, w jakiej dany skrypt jest wykonywany w trakcie procedury rozruchowej, należy z menu rozwijanego wybrać funkcję *Zmień priorytet włączania/wyłączania* (swoją drogą, nazywanie kolejności skryptu w sekwencji jego priorytetem jest dość

Rysunek 19.5.
Strona Usługi programu Zarządca programów startowych

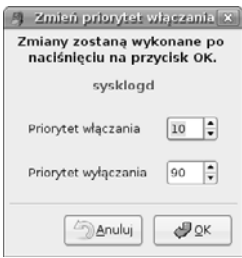


Rysunek 19.6.
Menu kontekstowe do modyfikacji usług w programie Zarządca programów startowych



mylące). Okno dialogowe przedstawione na rysunku 19.7 pozwala wprowadzić liczbową wartość sekwencji dla danego skryptu podczas uruchamiania i zamykania systemu.

Rysunek 19.7.
Modyfikacja sekwencji uruchamiania skryptów usług w programie Zarządca programów startowych

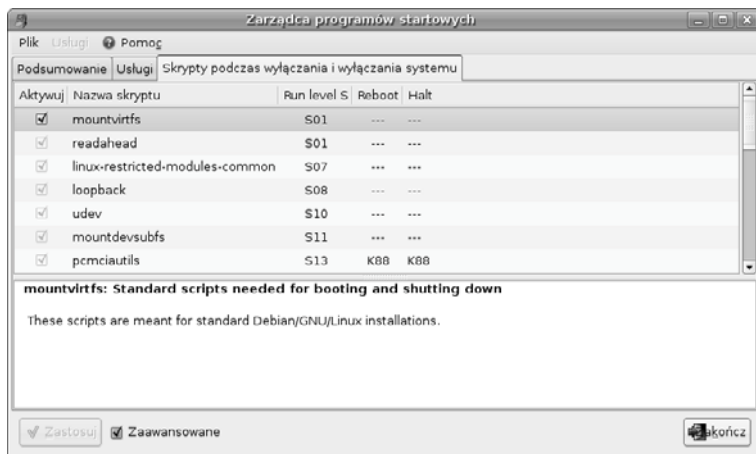


Po zmodyfikowaniu tych wartości należy kliknąć *OK* w celu zapisania zmian, co spowoduje powrót do głównego okna programu. Wszelkie zmiany wprowadzone w programie mają natychmiastowe zastosowanie w systemie (jednak aktualna wersja programu *Zarządca programów startowych* nie odświeża swojej listy usług i nie podaje nowego numeru sekwencji, co z pewnością może wprowadzić użytkownika w błąd).

Zarządca programów startowych, oprócz możliwości modyfikowania kolejności uruchamiania skryptów podczas startu i zamykania systemu, w trybie zaawansowanym oferuje stronę, na której można przejrzeć skrypty uruchamiane w procesie rozruchu i zamykania systemu w ramach skryptu zdefiniowanego wpisem `sysinit` w pliku `/etc/inittab`. Dostęp do tej strony odbywa się poprzez zakładkę *Skrypty podczas wyłączania i włączania systemu*. Znajdziemy tam listę skryptów ze zdefiniowaną sekwencją ich uruchamiania, co przedstawiam na rysunku 19.8.

Rysunek 19.8.

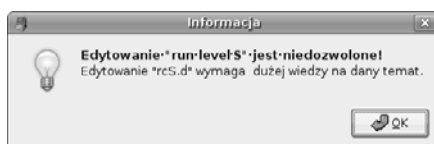
Zakładka skryptów startowych i końcowych programu Zarządca programów startowych



Te skrypty mają ścisły związek z systemowym (niskopoziomowym) procesem inicjacyjnym, w związku z tym na tej stronie nie ma możliwości wprowadzania zmian. Próba wyłączenia któregośkolwiek z tych skryptów zakończy się nieco irytującym komunikatem przedstawionym na rysunku 19.9 (na tej zakładce nieaktywny jest również przycisk *Zastosuj*).

Rysunek 19.9.

Ostrzeżenie dotyczące skryptów inicjacji systemu w programie Zarządca programów startowych



Ten komunikat nazwałem „lekko irytującym” jedynie przez grzeczność, ponieważ modyfikacja skryptów rozruchowych i zamykających system to zadanie wymagające sporej wiedzy na temat tego, co się robi. Jestem jednak pewny, że modyfikacja ustawień tych skryptów zostanie udostępniona w przyszłych wersjach narzędzia *Zarządca programów startowych*. W końcu to jest open source!

Aby zaakceptować i na stałe zatwierdzić wszelkie zmiany wprowadzone w ustawieniach usług, należy przed zakończeniem pracy w programie *Zarządca programów startowych* kliknąć przycisk *Zastosuj*, a następnie przycisk *OK*.

Init, następna generacja

Procedura znana pod nazwą SysVinit jest znanym i sprawdzonym sposobem kontroli rozruchu w systemach Unix, choć jest używana bez większych modyfikacji od wielu lat. Największą wadą tego mechanizmu wynika z tego, że wszystkie skrypty są tu uruchamiane sekwencyjnie, co oznacza, że proces rozruchu systemu może trwać znacznie dłużej niż jest to konieczne, przy założeniu, że skrypty rozruchowe nie zależą wzajemnie od siebie. Jimmy Wennlund opracował program *initng* (*Init, Next Generation*), który jest alternatywą dla procesu `/sbin/init` i stanowi krok we właściwym kierunku. Program ten obsługuje bowiem możliwość równoległego uruchamiania skryptów rozruchowych wszędzie tam, gdzie to tylko możliwe. Program *initng* umożliwia gromadzenie dodatkowych statystyk, które mogą być użyteczne przy analizie wydajności procesu rozruchowego. Więcej szczegółów na temat tego interesującego projektu (oraz, oczywiście, kod źródłowy) można znaleźć na jego stronie WWW: <http://initng.thinktux.net>.

Projekt *initng* jest w fazie aktywnego rozwoju i nadal dojrzewa, ale można go już zainstalować w swoim systemie i poddać eksperymentom. Po instalacji programu należy zmodyfikować parametry uruchamiania jądra, dopisując do nich `init=/sbin/initng`. Dzięki temu można swobodnie eksperymentować, a w przypadku problemów bez kłopotu wrócić do klasycznego mechanizmu rozruchowego.

Proces zamykania systemu Ubuntu Linux

Aby zamknąć Ubuntu Linux, należy z menu systemowego wybrać funkcję *System/Administracja/Zakończ*, co spowoduje wyświetlenie okna dialogowego przedstawionego na rysunku 19.10.

Kliknięcie ikony *Wyłącz komputer* w tym oknie spowoduje rozpoczęcie procedury zamykania systemu, która w gruncie rzeczy stanowi odwrócenie procedury rozruchowej. Po wybraniu tej opcji system wylogowuje użytkownika, a proces `/sbin/init` rozpoczyna uruchamianie skryptów zamykających system, z tą różnicą, że tym razem uruchamiane są skrypty o nazwach rozpoczynających się na literę *K* z katalogu `/etc/rcNN.d`, gdzie *NN* to bieżący poziom działania systemu. Następnie system uruchamia skrypty zamykające zapisane w katalogu systemowym, po czym zatrzymuje komputer.

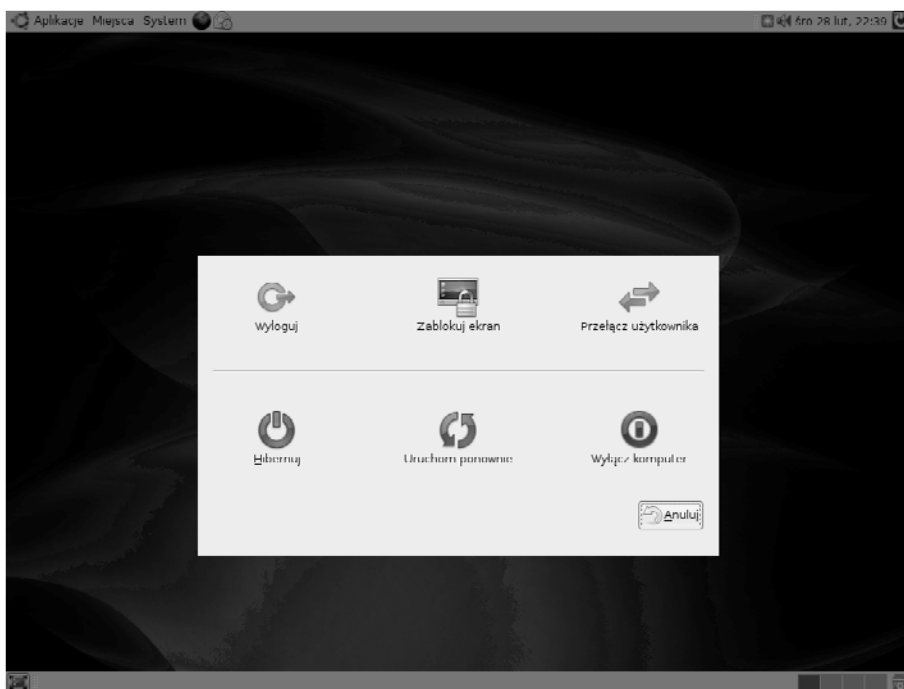
Jako swoisty hołd oddany tradycji komputerów PC opartych na architekturze x86 (a raczej w charakterze srebrnej kuli, w uniwersalny sposób rozwiązującej wszelkie problemy) proces *init* ma zaimplementowaną obsługę kombinacji klawiszy *Control+Alt+Delete*, której można przypisać wykonanie określonych działań w systemie. Ta kombinacja klawiszy jest również określana jako „trzypalcowy salut”.

Definicja zachowania systemu w efekcie naciśnięcia kombinacji klawiszy *Control+Alt+Delete* znajduje się w pliku `/etc/inittab`. Służy do tego słowo kluczowe `ctrlaltdel`, a w systemie Ubuntu znajdziemy następującą definicję:

```
ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -r now
```

Nakazuje ona systemowi wykonanie polecenia `/sbin/shutdown` z opcjami:

- ♦ oczekiwania jednej sekundy między komunikatem o zamykaniu systemu a rozpoczęciem samej procedury (`-t1`);
- ♦ wykorzystania pliku `/etc/shutdown.allow` do określenia użytkowników upoważnionych do wykonania tej operacji (`-a`);



Rysunek 19.10. Okno dialogowe zamykania systemu w Ubuntu

- ♦ po zamknięciu system ma uruchomić się ponownie (`-r`);
- ♦ operacja ma odbyć się natychmiast (`now`).

Bardziej szczegółowe informacje na temat polecenia `/sbin/shutdown` można znaleźć w podręczniku systemowym tego polecenia (wywołanie `man shutdown` w programie `xterm` albo terminalu GNOME). Warto zapoznać się z tą dokumentacją przed zamknięciem systemu: jedna sekunda oczekiwania to z pewnością za mało, aby przeczytać ten podręcznik.

Podsumowanie

W tym rozdziale zawarłem informacje na temat mechanizmu rozruchu systemu Linux oraz szczególnych cech tego procesu w dystrybucji Ubuntu. Proces rozruchowy Linuksa jest dość interesującą kombinacją elementów zaszytych w systemie na stałe oraz elementów kontrolowanych przez użytkownika, dzięki czemu jest mocno modyfikowalny w zakresie wyboru uruchamianych usług oraz kolejności ich uruchamiania.

W rozdziale 20. omówiłem mechanizmy pozwalające dodawać, usuwać i aktualizować oprogramowanie w systemie Ubuntu. Wszystkie instalacje Ubuntu oferują ogromny wybór aplikacji, ale jak się będziemy mogli wkrótce przekonać, w opcjonalnych repozytoriach Ubuntu czeka na nas jeszcze większy wybór bardzo użytecznego oprogramowania. Następny rozdział zawiera opis sposobów pobierania oprogramowania, przeglądania zasobów oraz rozpoznawania, który pakiet należy zainstalować, aby uzupełnić system o określoną funkcję lub niezbędny plik.

Rozdział 20.

Dodawanie, usuwanie i aktualizacja oprogramowania

W tym rozdziale:

- ♦ Przegląd mechanizmów zarządzania pakietami
- ♦ Wykorzystanie repozytoriów Ubuntu
- ♦ Wyszukiwanie elementów systemu
- ♦ Wykorzystanie programów apt-get, aptitude i Synaptic
- ♦ Wykorzystanie menedżera aktualizacji
- ♦ Porządkowanie systemu

Każdy, kto przed zainstalowaniem Ubuntu rozejrzył się nieco w dostępnej ofercie Linuksów, z pewnością ma świadomość, że istnieje mnóstwo dystrybucji, każda z nich ma własny instalator, preferowane środowisko biurkowe oraz menedżer okien systemu X Window, zestaw podstawowych aplikacji i jakiś sposób na aktualizowanie oprogramowania dostarczanego w ramach dystrybucji, rozszerzanie go i zarządzanie nim. Gdy się nad tym zastanowić głębiej, ostatnie z wymienionych zagadnień jest stanowczo najważniejszym aspektem dystrybucji Linuksa, obok rozmiaru i poziomu zaangażowania społeczności użytkowników i deweloperów. Istnieje interesująca zależność między łatwością użytkowania i utrzymania a rozmiarem społeczności użytkowników. Dystrybucja Linuksa, która ułatwia utrzymanie oprogramowania w jak najbardziej aktualnych wersjach, instalację nowego oprogramowania i zarządzanie oprogramowaniem zainstalowanym w systemie, to dystrybucja, która ma szansę na znalezienie dużej liczby użytkowników. Ta uwaga wymaga przyjęcia jednego podstawowego założenia: ktoś musi aktualizować oprogramowanie w dystrybucji.

Jak wspominałem w rozdziale 1., uważam, że społeczność dystrybucji Ubuntu Linux jest najciekawszą, dynamiczną i energiczną społecznością linuksową z tych, które miałem okazję poznać. Społeczność Ubuntu jest oddana i dotyczy to zarówno użytkowników, jak

i jego twórców. Aktualizacje programów i poprawki błędów są częste, co (wbrew pozorom) jest dobrą wiadomością. Na pobranie i instalację czeka szeroki wybór oprogramowania.

Dostęp do nowego oprogramowania i jego aktualizacji jest bardzo łatwy. Menedżer aktualizacji Ubuntu automatycznie informuje użytkownika o dostępności aktualizacji oprogramowania zainstalowanego w systemie i utrzymanie systemu w świeżości wymaga jedynie kilku kliknięć. Nie najgorzej! Programy `dpkg`, `apt-get`, `aptitude` i `synaptic` umożliwiają instalację oprogramowania w równie prosty sposób, a menedżer pakietów Synaptic daje użytkownikom możliwość wyszukiwania oprogramowania wśród dostępnych w repozytoriach. Czy można chcieć czegoś więcej?

Kluczem do tej prostoty jest doskonały i łatwy w użyciu format pakietów DEB (od Debiana), opracowany na potrzeby dystrybucji Debian Linux i wykorzystany w Ubuntu. Można powiedzieć, że format pakietów DEB i narzędzia do jego obsługi mogą zawstydzić pozostałe formaty, w tym RPM. Osobiście nie miałbym nic przeciwko terapii elektrowstrząsami, jeśli pomogłaby mi zapomnieć o godzinach spędzonych na pokonywaniu pokręconego labiryntu wzajemnych zależności między pakietami RPM. Jeśli ktoś nie wie, co to jest RPM, ma szczęście. I niech tak zostanie.



Czytelnicy dobrze zaznajomieni z dystrybucjami opartymi na menedżerze pakietów RPM, takim jak Fedora Core, Mandriva, Red Hat czy Yellow Dog, z pewnością zakrzykną: „Głupcze, przecież `yum` rozwiązuje wszelkie problemy z pakietami RPM”. To prawdopodobnie twierdzenie, które da się obronić w sądzie, ale nie uważam, żeby dodawanie warstw abstrakcji rozwiązujących problemy „wbudowane” w format RPM było właściwym rozwiązaniem problemu. Wolę wykorzystywać inteligentniejszy format pakietów, jakim jest DEB, a wraz z nim stosować inteligentne narzędzia, takie jak te, które omówię w tym rozdziale, niż wiecznie udowadniać, że żaba nie ma sierści, uprzednio goląc ją dokładnie.



Jeśli zdarzy się potrzeba użycia pakietu przygotowanego dla innej dystrybucji, tzn. zapisanego w innym formacie, warto zapoznać się z punktem „Konwersja pakietów z innych formatów” w dalszej części tego rozdziału.

Narzędzia `dpkg`, `apt-get` i `aptitude` posiadają szczegółowe podręczniki online i wyczerpujące podręczniki systemowe objaśniające wszelkie kombinacje opcji wywołania w wierszu poleceń, nie będę więc zanudzał czytelnika powielaniem tych dokumentów. W tym rozdziale skupię się natomiast na realizacji za pomocą tych narzędzi często wykonywanych zadań, czyli pokażę, w jaki sposób wykonać konkretne zadania, zamiast omawiać narzędzia w sposób ogólny. Pierwsza część tego rozdziału zawiera przegląd narzędzi do zarządzania pakietami wykorzystywanych w Ubuntu (i omówionych szczegółowo w dalszej części rozdziału). Dzięki temu czytelnik będzie miał możliwość zorientowania się w roli każdego z tych programów, co przyda się w dalszej części rozdziału. W następnym podrozdziale wyjaśnię, w jaki sposób zorganizowane jest oprogramowanie, które można pobrać bezpośrednio z repozytoriów Ubuntu, przekażę też kilka wskazówek dotyczących innych, nieoficjalnych repozytoriów, które mogą zawierać użyteczne pakiety. Po tym podrozdziale czytelnik zapozna się z częścią praktyczną, wyjaśniającą sposoby wyszukiwania elementów i informacji o pakietach zainstalowanych w systemie, określania nazwy pakietu, w którym znajdował się wskazany plik zainstalowany w systemie oraz to, jaki pakiet należy zainstalować, aby uzyskać w systemie określony plik. Te operacje wykonuje się najczęściej w wierszu poleceń i te podrozdziały będą wykorzystywały głównie narzędzia trybu tekstowego.

Kolejne podrozdziały będą omawiały użycie programów: `apt-get` (tekstowy), `aptitude` (terminalowy) i `Synaptic` (wyposażony w interfejs graficzny). Są to narzędzia służące do instalowania nowych pakietów w systemie. W dwóch ostatnich podrozdziałach omówię użycie menedżera aktualizacji w celu utrzymania systemu w świeżości oraz zagadnienia oczyszczania systemu z pakietów, które nie są już potrzebne, ponieważ zostały zainstalowane wcześniej do obsłużenia zależności pakietów już usuniętych.

Przegląd oprogramowania do zarządzania pakietami w Ubuntu

Standardowa instalacja dystrybucji Ubuntu Linux zawiera kilka różnych narzędzi do zarządzania i odczytu informacji o pakietach zainstalowanych i dostępnych do instalacji. Narzędzia te są bardzo różnorodne: od niskopoziomowych, obsługiwanych z wiersza poleceń, po graficzne, wyposażone w wygodny interfejs użytkownika. Oto podstawowe narzędzia zainstalowane standardowo w każdym systemie Ubuntu.

- ♦ `apt-get`: narzędzie obsługiwane z wiersza poleceń, za którego pomocą można instalować pakiety w systemie, usuwać je i nimi zarządzać. Można wykonywać operacje na poszczególnych pakietach, jak i na całych komponentach dystrybucji.
- ♦ `aptitude`: narzędzie terminalowe, czyli wyposażone w tekstowy interfejs użytkownika, stanowiące w gruncie rzeczy nakładkę na niskopoziomowe narzędzia, takie jak `apt-get` i `dpkg`. Program `aptitude` umożliwia instalację pakietów, ich usuwanie i wyszukiwanie oraz zarządzanie nimi w pseudograficznym interfejsie tekstowym wyposażonym w menu.
- ♦ `dpkg`: podstawowe narzędzie wykorzystywane z wiersza poleceń, służące do instalowania pakietów, ich usuwania i wyszukiwania oraz zarządzania nimi. Wykorzystuje ono jeszcze bardziej niskopoziomowe narzędzia, takie jak `dpkg-deb`, do instalowania pakietów, ich usuwania i manipulowania nimi, oraz `dpkg-query` do wyszukiwania pakietów. Tych niskopoziomowych narzędzi nie będziemy omawiać w tym rozdziale. Są one bardzo użyteczne do bezpośredniego użytku, średnio w 0,0001% przypadków, ale uważam, że o wiele wygodniej jest je wykorzystywać za pośrednictwem `dpkg`.
- ♦ `dselect`: interfejs do programu `dpkg` obsługiwany terminalowo, oferujący pseudograficzne menu i interaktywne ekrany z wykorzystaniem środowiska tekstowego: może to być `xterm`, `GNOME Terminal` lub tym podobne. Program `dselect` można również wykorzystywać z wiersza poleceń, dzięki czemu nie jest konieczne wybieranie operacji z menu programu. Tej aplikacji nie będę omawiał w tym rozdziale, ponieważ pozostałe narzędzia posiadają te same możliwości.
- ♦ `synaptic`: graficzne narzędzie do instalowania pakietów, usuwania ich i wyszukiwania oraz zarządzania nimi w Ubuntu. To moje ulubione narzędzie do codziennych zadań związanych z instalowaniem pakietów.
- ♦ `update-manager`: graficzne narzędzie do identyfikowania i instalowania zaktualizowanych wersji pakietów zainstalowanych w systemie Ubuntu Linux.



Programy `dpkg`, `apt-get` i `aptitude` wykorzystują różne bazy, w których zapisują informacje o zainstalowanych pakietach, z tego powodu stosują odmienne mechanizmy do identyfikacji oraz rozstrzygania konfliktów i zależności między pakietami. Program `dpkg` zapisuje informacje o zainstalowanych pakietach w plikach i podkatalogach katalogu `/var/lib/dpkg`. Program `apt-get` informacje te zapisuje w katalogu `/var/lib/apt`, a `aptitude` analogicznie w katalogu `/var/lib/aptitude`. Wykorzystując `apt-get` lub `aptitude`, należy w pierwszej kolejności uruchomić polecenia `apt-get update` lub odpowiednio `aptitude update`, dzięki czemu zostaną zaktualizowane dane tych aplikacji dotyczące wersji pakietów i inne informacje na ich temat. Podobnie, jeśli po użytkowaniu programów `apt-get`, `aptitude` czy też `Synaptic` zdecydujemy się na wykorzystanie programu `dselect`, należy uruchomić polecenie `dselect update`, aby program `dselect` (a w konsekwencji `dpkg`) odświeżył sobie informacje o zainstalowanych i dostępnych pakietach.

Programy `apt-get`, `aptitude` i `Synaptic` swoje działanie uzależniają od informacji o dostępnych w repozytoriach i powiązaniach między pakietami. Informacje te są zapisane w zasobie, określanym często „buforem programu apt” (*apt cache*). Bufor ten jest bazą informacji o pakietach, zapisaną domyślnie w katalogu `/var/cache/apt` systemu Ubuntu Linux.

W tym rozdziale skupię się na omówieniu programów, takich jak `apt-get`, `aptitude`, `dpkg`, `Synaptic` i *Menedżer aktualizacji*, w kontekście wykonywania zadań, do których te narzędzia nadają się najlepiej; oprę się o moje doświadczenia z ich wykorzystaniem. Jestem fanem narzędzi graficznych, które „robą to, co powinny”, większość przykładów z tego rozdziału będę ilustrował zrzutami ekranu z programu *Menedżer pakietów Synaptic*, który uważam za bardzo użyteczny, a jednocześnie prosty w obsłudze. Jednakże w tym rozdziale staram się być obiektywny i zawsze wskazywać najlepsze narzędzie do realizacji konkretnego, omawianego zadania.



Innym narzędziem do zarządzania pakietami w systemach pokrewnych Ubuntu jest `adept`, graficzny interfejs dostępny w Kubuntu. Aplikacja `adept` jest doskonałym, prostym w użyciu narzędziem, ale nie omawiam go w tym rozdziale, ponieważ nie wchodzi w skład standardowej dystrybucji Ubuntu, na której skupia uwagę ta książka. Staram się unikać poruszania zagadnień specyficznych dla krewniaków Ubuntu, takich jak `Edubuntu`, `Kubuntu` czy `Xubuntu`.

Wszystkie zagadnienia związane z automatyzacją użycia narzędzi do zarządzania pakietami w Ubuntu opierają się na pobieraniu pakietów z repozytoriów Ubuntu. W następnym podrozdziale wyjaśnię, czym są repozytoria, w jaki sposób są zorganizowane, czym jest plik konfiguracyjny służący do identyfikacji repozytoriów oraz omówię sposób modyfikacji tego pliku po to, by uzyskać dostęp do nowych repozytoriów z najświeższymi wersjami oprogramowania.

Repozytoria i ich komponenty w systemie Ubuntu

Repozytorium jest dokładnie tym, co sugeruje jego nazwa, czyli magazynem obiektów określonego typu. Obiekty zapisane w repozytorium Ubuntu to pakiety binarne i źródłowe, z których składa się dystrybucja Ubuntu. Po pobraniu z serwera obrazu ISO płyty

instalacyjnej (czyli obrazu płyty CD lub DVD w formacie zgodnym ze standardem 9660 opracowanym przez International Standards Organization, czyli formatem systemu plików wykorzystywanym na płytach CD-ROM) i wypaleniu z niego dysku CD z instalatorem Ubuntu otrzymujemy środowisko rozruchowe Ubuntu (znane pod nazwą Isolinux), instalator i aplikacje niezbędne mu do pracy, podstawowy zestaw pakietów oraz nieco dokumentacji. W ramach tego procesu większość pakietów instalowanych w systemie może być pobierana bezpośrednio z internetu.

Udostępnianie obrazów ISO do pobrania z internetu nie jest nową koncepcją — w rzeczywistości to najpopularniejszy sposób dystrybuowania instalatorów Linuksa. Udostępnianie repozytoriów Ubuntu w sieci ma dla użytkowników dodatkowe zalety:

- ♦ nośnik instalacyjny Ubuntu może mieć mniejsze rozmiary, a zatem mieści się na pojedynczym dysku CD;
- ♦ dowolny element Ubuntu Linux jest dostępny w najbardziej aktualnej wersji dla każdego komputera przyłączonego do internetu.

Pakiety w Ubuntu są uporządkowane w taki sposób, żeby można było dokonać wyboru grup pakietów w zależności od potrzeb, zarówno z punktu widzenia poszczególnych osób, jak i korporacji. W tym celu wyróżniono cztery grupy, klasyfikujące oprogramowanie linuksowe względem typu obsługi oraz zagadnień licencyjnych. Oto one.

- ♦ **main**: główny element biblioteki pakietów binarnych (i większości źródłowych) oprogramowania dla systemu Linux oficjalnie wspieranego przez firmę Canonical, Ltd. i dopuszczonego do wolnej dystrybucji. Nie oznacza to, że wszystkie elementy repozytorium *main* są udostępnione na licencji GPL, mogą tu bowiem występować pakiety zawierające obiekty wyłącznie binarne, jak firmware sprzętu czy fonty. Część repozytorium *main* Ubuntu została zaprojektowana w taki sposób, aby zapewnić wszystkie elementy niezbędne do zbudowania w pełni funkcjonalnego środowiska biurowego opartego na systemie Linux, obsługiwanego w całości przez firmę Canonical, Ltd.
- ♦ **restricted**: ta część to pakiety źródłowe i binarne zawierające często wykorzystywane oprogramowanie udostępniane na licencjach nie do końca wolnych. Pakiety w części *restricted* repozytorium nie mają gwarancji pełnego wsparcia ze strony Canonical, Ltd., ale są udostępniane, ponieważ mogą być niezbędne do działania Linuksa na niektórych typach sprzętu. Z tego powodu niektóre elementy z repozytorium *restricted* są dostarczane z dyskiem instalacyjnym CD w podkatalogach */dist/s/dist-name/restricted* i */pool/restricted*. W wydaniu 6.06 (Dapper) w repozytorium *restricted* znajdują się sterowniki dla specyficznych kart sieciowych i kart graficznych.
- ♦ **universe**: ten element repozytorium to pakiety binarne i źródłowe zawierające inne oprogramowanie na licencjach free i open source, często wykorzystywane z systemem Linux, ale dostarczane na licencjach innych niż sam system. Oprogramowanie to nie ma gwarancji obsługi przez Canonical, Ltd. Po prostu w repozytorium *universe* jest zbyt wiele oprogramowania, aby firma Canonical mogła pozwolić sobie na udzielanie takiego wsparcia.
- ♦ **multiverse**: ta część repozytorium to binarne i źródłowe pakiety oprogramowania, które są udostępniane na licencjach niezgodnych ze standardami przyjętymi dla

projektu Ubuntu (<http://www.ubuntu.com/ubuntu/licensing>). To oprogramowanie nie jest wspierane przez Canonical, Ltd. i obowiązkiem użytkownika jest autoweryfikacja, czy spełnia wymogi narzucone przez twórców danego oprogramowania.



Oficjalny opis repozytoriów Ubuntu znajduje się pod adresem <http://www.ubuntu.com/ubuntu/components>. Starałem się zawrzeć tu esencję ich znaczenia, ale jeśli ktoś jest fanem licencji, znajdzie tam mnóstwo ciekawego materiału.

Istnieje jeszcze jeden element repozytorium, pod nazwą *backports*. To repozytorium zawiera zaktualizowane lub nowsze wersje oprogramowania dla Linuksa, w tym również oprogramowanie, które posiada swoje oficjalne (często starsze) wersje w jednej z pozostałych części repozytorium. Oprogramowanie z tego repozytorium nie jest wspierane przez Canonical, Ltd. ale jest udostępniane, ponieważ nowsze wersje oprogramowania często zawierają poprawki i ulepszenia, oczekiwane przez użytkowników.



Użytkownicy zainteresowani, jakie pakiety są zainstalowane w poszczególnych wydaniach Ubuntu, szczegółów mogą szukać na stronie <http://packages.ubuntu.com>. Na tej stronie można znaleźć listę pakietów dla poszczególnych wydań dystrybucji, zorganizowaną według różnych, logicznych kategorii. Można tam również skorzystać z opcji wyszukiwania pakietów.

Uaktywnianie poszczególnych elementów repozytorium

Najczęściej spotykaną sytuacją, w której koniecznie musimy zastosować część repozytorium inną niż *main* czy *restricted*, jest chęć wykorzystania aplikacji audio i wideo. Sze-roki wybór kodeków (kompresorów i dekompresorów) wykorzystywanych do zakodowania cyfrowych materiałów audio i wideo, powiązania systemowe tych kodeków oraz utrudnienia, jakie wielu dostawców mediów narzuca użytkownikom w celu utrzymania kontroli nad treściami audio i wideo, powodują, że użytkownik zainteresowany odtwarzaniem (i tworzeniem) tego typu materiałów jest zmuszony do samodzielnego rozważenia implikacji wykorzystania tych narzędzi z uwzględnieniem niuansów licencjonowania.

W domyślnej konfiguracji, po zainstalowaniu systemu Ubuntu Linux, uaktywnione są tylko repozytoria *main* i *restricted*. Uaktywnione, tzn. że w pliku `/etc/apt/sources.list` znajdują się definicje tylko tych części repozytorium. Ten plik jest wykorzystywany przez poszczególne aplikacje służące do instalowania pakietów omówionych w tym rozdziale i zarządzania nimi. To jest plik tekstowy zawierający wpisy definiujące lokalizację i nazwę poszczególnych repozytoriów internetowych oraz dostępnych w nich elementów. Ogólny format wpisu w `/etc/apt/sources.list` jest następujący:

```
deb URI dystrybucja komponent-lub-pakiety
```

Pole `deb` identyfikuje format pakietów, `URI` to adres lokalizacji, w której można znaleźć repozytorium i zwykle występuje w postaci adresu FTP lub HTTP. Pole `dystrybucja` jest najczęściej nazwą standardowej dystrybucji Ubuntu, ale może stanowić dowolny ciąg znaków identyfikujący katalog, w których znajdują się pakiety na serwerze, `komponent-lub-pakiety` to jedna lub więcej nazw komponentów repozytorium, co również określa ścieżki, w których program będzie poszukiwał pakietów. Jeśli w ramach pola `komponent-lub-`

-*pakiety* znajdzie się większa liczba pozycji, każda z nich wskazuje inny podkatalog w ścieżce *URI*.

Przykładem mogą być tu standardowe wpisy w pliku */etc/apt/sources.list*, definiujące repozytoria *main* i *restricted* dla wydania Ubuntu Linux 6.06 (Dapper):

```
deb http://archive.ubuntu.com/ubuntu/ dapper main restricted
deb-src http://archive.ubuntu.com/ubuntu/ dapper main restricted
```

Pierwszy z tych wpisów identyfikuje część repozytorium zawierającą pakiety binarne, drugi natomiast identyfikuje pakiety źródłowe. Analogiczna para wpisów definiujących pakiety aktualizacyjne będzie następująca:

```
deb http://archive.ubuntu.com/ubuntu/ dapper-updates main restricted
deb-src http://archive.ubuntu.com/ubuntu/ dapper-updates main restricted
```



Przykłady wpisów w pliku */etc/apt/sources.list* z tego podrozdziału dotyczą wydania Ubuntu 6.06 (Dapper). Użytkownicy innych wersji Ubuntu powinni słowo kluczowe *dapper* zastąpić odpowiednią nazwą wydania Ubuntu, np. *breezy* (wydanie 5.10, poprzedzające Dappera) lub *edgy* (wydanie 6.10, następca Dappera) itp. Nie należy mieszać w pliku */etc/apt/sources.list* wpisów repozytorium dla różnych wydań Ubuntu, ponieważ repozytoria dla nowszych wydań z reguły będą zawierały nowsze wersje programów dostępnych dla starszych wydań, ale skompilowane pod kątem innej wersji bibliotek systemowych, dostępnych w nowszym wydaniu. Informacje na ten temat można znaleźć w podrozdziale „Mieszanie repozytoriów Ubuntu i Debiana”, w dalszej części rozdziału, ponieważ czasem zachodzą sytuacje zmuszające użytkownika do wykorzystania wpisów pliku */etc/apt/sources.list*, które z założenia są niekompatybilne z używanym wydaniem Ubuntu.

W repozytorium Ubuntu znajdują się również pakiety z poprawkami bezpieczeństwa, ich wpisy mają następującą postać:

```
deb http://security.ubuntu.com/ubuntu dapper-security main restricted
deb-src http://security.ubuntu.com/ubuntu dapper-security main restricted
```

Wiersze rozpoczynające się od znaku *#* w pliku */etc/apt/sources.list* stanowią komentarze. Przeglądając standardowe wpisy w pliku */etc/apt/sources.list*, można zauważyć, że znajdują się tam już gotowe wpisy do części *universe*, ale są zablokowane właśnie tym znakiem komentarza. Uaktywnienie ich jest proste, wystarczy otworzyć ten plik w edytorze tekstu (z prawami użytkownika *root*), np.:

```
$ sudo emacs /etc/apt/sources.list
```

Usunięcie znaków *#* z początku tych wpisów uaktywni repozytoria *universe*, warto też dopisać *multiverse* na końcu wiersza, po czym zapisać plik. Zaktualizowane wpisy będą miały następującą postać:

```
deb http://archive.ubuntu.com/ubuntu/ dapper universe multiverse
deb-src http://archive.ubuntu.com/ubuntu/ dapper universe multiverse
```

Można, oczywiście, po prostu dopisać te dwie pozycje (*universe multiverse*) na końcu wpisów *main restricted* standardowych repozytoriów, ale wolę trzymać je osobno, po prostu dla czytelności związanej z licencjami, no i dlatego, żeby odróżnić wpisy domyślne od własnych modyfikacji w pliku *sources.list*.



Żeby wszystko było zupełnie jasne: sugerując, by dopisać repozytoria *universe* i *multiverse* do pliku konfiguracyjnego ze źródłami pakietów, nakłaniam czytelnika do instalowania pakietów, które mogą w różnych krajach lub instytucjach budzić wątpliwości dotyczące licencji (np. odnoszące się do opatentowanego formatu MP3 lub niektórych kodeków wideo), dla których nie ma żadnej gwarancji wsparcia twórców Ubuntu (nawet płatnego i nie ma żadnych planów, aby takie wsparcie świadczyć), co może w konsekwencji doprowadzić do problemu ze znalezieniem jakiegokolwiek wsparcia. Konfiguruje systemy z myślą o wykorzystaniu w korporacji, należy upewnić się, że użyte oprogramowanie będzie dostępne na licencjach niestanowiących zagrożenia dla firmy. Osobiście wykorzystuję wszystkie programy opisane w tej książce i jestem sam dla siebie testerem w zakresie wszystkiego, co opisuję, ale potrzeby czytelników (i ich zakres odpowiedzialności) mogą być zupełnie inne.

Uaktywnianie dodatkowych repozytoriów z użyciem edytora tekstu

Jak widać w przykładach z poprzedniego podrozdziału, standardowe repozytoria Ubuntu znajdują się na serwerach o adresach z domeny *ubuntu.com*. To ma sens, ponieważ jest to jedyny sposób, aby pracownicy firmy Canonical byli w stanie zagwarantować integralność swoich repozytoriów i zgodność elementów wchodzących w skład repozytoriów z przyjętymi zasadami dotyczącymi licencji. Jednak w internecie istnieją też inne repozytoria pakietów dla systemu Ubuntu, obsługiwane przez różne osoby i grupy, również w ogóle niezwiązane z firmą Canonical. Te alternatywne repozytoria mogą po prostu stanowić serwery lustrzane oficjalnych repozytoriów Ubuntu, co pozwala na zmniejszenie obciążenia oficjalnych serwerów. Inne z repozytoriów oferują oprogramowanie nieuwzględnione w oficjalnych repozytoriach Ubuntu. W tym przypadku oprogramowanie może również występować w wersjach nowszych niż te, które zawiera repozytorium *backports*, repozytoria mogą też być przedstawiane w ramach projektu rozwijającego dany program i oferować jego aktualne wersje dla różnych wydań dystrybucji. Jak wspominałem wcześniej, użytkownicy zainteresowani oprogramowaniem do obsługi audio i wideo mogą znaleźć w alternatywnych repozytoriach najświeższe wersje potrzebnych pakietów.

Oto niektóre znane alternatywne repozytoria oprogramowania dla Ubuntu, dostępne w okresie tworzenia tej książki.

- ♦ **FreeNX:** repozytorium zawierające najświeższą wersję klienta i serwera FreeNX. Serwer FreeNX to rozwiązanie alternatywne w stosunku do pakietów *vnc-server*, *vino* i *vnc-viewer*. W pliku */etc/apt/sources.list* dodajemy następujące wpisy:

```
deb http://mirror3.ubuntulinux.nl/ dapper-seveas freenx
deb-src http://mirror3.ubuntulinux.nl/ dapper-seveas freenx
```

- ♦ **Obsługa multimediów:** repozytorium udostępniające kodeki systemu Windows oraz inne oprogramowanie do odtwarzania cyfrowych plików audio i wideo. W pliku */etc/apt/sources.list* dopisujemy następujące wpisy:

```
deb ftp://cipherfunk.org/pub/packages/ubuntu/ dapper main
deb-src ftp://cipherfunk.org/pub/packages/ubuntu/ dapper main
```



Dla tego repozytorium należy zaimportować klucz GPG, pozwalający zweryfikować udostępniane pakiety. Klucz można pobrać z adresu <http://subkeys.gpg.net>. „Odcisk palca” tego klucza (ang. *fingerprint*) to 33BAC1B3.

- ♦ Penguin Liberation Front: repozytorium, w którym można znaleźć pakiety podlegające patentom lub innym mechanizmom, które potencjalnie blokują możliwość ich wolnego wykorzystania. W pliku `/etc/apt/sources.list` dopisujemy następujące wpisy:

```
deb http://packages.freecontrib.org/ubuntu/plf dapper free non-free
deb-src http://packages.freecontrib.org/ubuntu/plf dapper free non-free
```

- ♦ Wine: repozytorium zawierające najświeższą wersję WINE, czyli środowiska emulującego Windows API, umożliwiającego instalację i uruchamianie w Linuksie oprogramowania dla Windows. W pliku `/etc/apt/sources.list` dopisujemy następujące wpisy:

```
deb http://wine.budgetdedicated.com/apt dapper main
deb-src http://wine.budgetdedicated.com/apt dapper main
```

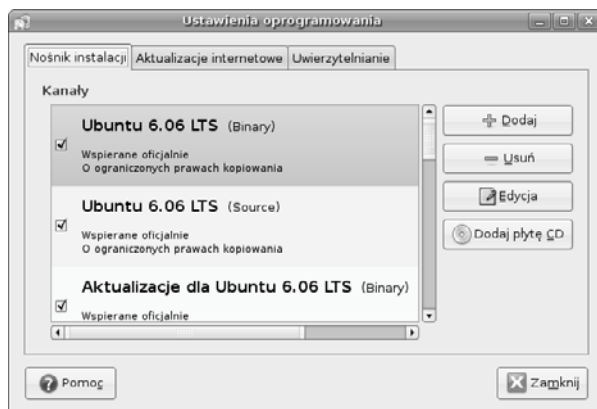
Użytkownicy zainteresowani eksploracją innych repozytoriów, którzy przy okazji chcą poznać naprawdę ciekawą aplikację, powinni przejrzeć stronę <http://www.ubuntu-linux.nl/source-o-matic>. Na niej można znaleźć aplikację WWW generującą plik `sources.list` w oparciu o wybrany zbiór dostępnych repozytoriów.

Uaktywnianie dodatkowych repozytoriów z wykorzystaniem narzędzia Ustawienia oprogramowania

Edycja tekstowego pliku konfiguracyjnego to „stara szkoła” zarządzania Linuksem i dla niektórych może stanowić krępujący temat w rozmowie ze znajomymi użytkownikami Windows lub Mac OS X (jestem pewny, że będziecie z tej możliwości korzystać, bo tak jest łatwiej i szybciej, ale uznajmy to za nasz mały, linuksowy sekret, OK?). Dla fanów graficznych interfejsów użytkownika Ubuntu zawiera narzędzie pod nazwą *Ustawienia oprogramowania*, które pozwala na przeglądanie wpisów z pliku `/etc/apt/sources.list` oraz łatwe uaktywnianie wpisów dotyczących repozytoriów i ich dezaktywację.

Narzędzie *Ustawienia oprogramowania* uruchamia się z menu *System/Administracja/Właściwości oprogramowania*. Po podaniu hasła użytkownika w celu uaktywnienia trybu administratorskiego pojawia się okno Ustawienia oprogramowania, przedstawione na rysunku 20.1.

Rysunek 20.1.
Główne okno
narzędzia Ustawienia
oprogramowania

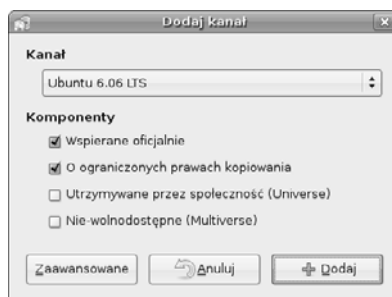


Jak widać na rysunku 20.1., główne okno narzędzia *Ustawienia oprogramowania* zawiera wszystkie wpisy z pliku */etc/apt/sources.list*, niezależnie od tego, czy są aktywne, czy zablokowane znakiem komentarza. Każda pozycja jest reprezentowana przez tzw. kanał. Pozycje nieaktywne są symbolizowane brakiem zaznaczenia w polu wyboru przy ich nazwie. To daje możliwość przeglądu wpisów repozytoriów z pliku */etc/apt/sources.list*.

Aby dodać nowe repozytorium lub element do listy repozytoriów, należy kliknąć przycisk *Dodaj*. To spowoduje wyświetlenie okna dialogowego, przedstawionego na rysunku 20.2.

Rysunek 20.2.

*Dodawanie
repozytorium
w programie
Ustawienia
oprogramowania*



Menu rozwijane *Kanał* z rysunku 20.2 pozwala wybrać dowolne ze standardowych repozytoriów Ubuntu i związanych z nimi elementów. Domyślnie aktywne są tylko elementy *main* i *restricted*, ale można dodać elementy *universe* i *multiverse* do dowolnego repozytorium, w którym są dostępne; wystarczy zaznaczyć pola wyboru *Utrzymywane przez społeczność (Universe)* i *Nie-wolnodostępne (Multiverse)*.

Okno dialogowe z rysunku 20.3 pozwala również dopisać pozycje dla repozytoriów niestandardowych. W tym celu należy kliknąć przycisk *Zaawansowane*, co spowoduje wyświetlenie okna przedstawionego na rysunku 20.3.

Rysunek 20.3.

*Dodawanie
niestandardowego
repozytorium
w narzędziu
Ustawienia
oprogramowania*



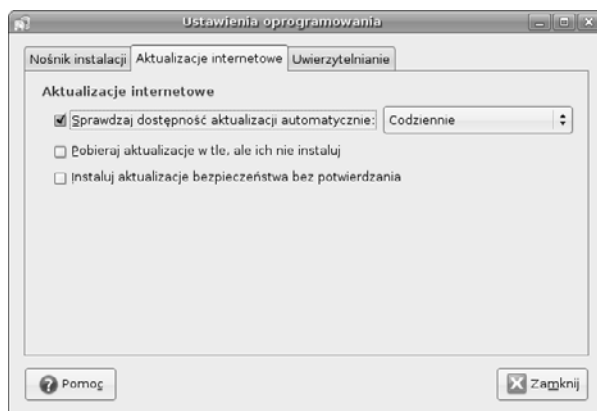
To okno dialogowe nie jest szczególnie eleganckie, ale pozwala z łatwością wkopiować gotową reprezentację repozytorium z wiadomości e-mail lub strony WWW. Tak wstawiona pozycja zostanie dopisana do pliku */etc/apt/sources.list*. Do okna z rysunku 20.2 można wrócić bez dodawania pozycji, wystarczy kliknąć przycisk *Anuluj*. Aby dopisać nową pozycję, należy wpisać ją zgodnie ze standardem konstrukcji wpisu w pliku *sources.list*, po czym kliknąć przycisk *Dodaj kanał*. To spowoduje dopisanie pozycji i powrót do okna z rysunku 20.1.

Narzędzie *Ustawienia oprogramowania* zawiera trzy zakładki, w których zarządza się różnymi aspektami dotyczącymi repozytoriów i procesu aktualizacji. Pierwsza strona, znajdująca się pod zakładką *Nośnik instalacji*, pojawia się domyślnie po uruchomieniu

narzędzia *Ustawienia oprogramowania*. Pozostałe z nich, *Aktualizacje internetowe* i *Uwierzytelnianie*, służą do zarządzania zagadnieniami związanymi z automatycznym pobieraniem aktualizacji z internetu oraz do konfiguracji mechanizmów weryfikacji poprawności repozytoriów.

Na rysunku 20.4 przedstawiam stronę *Aktualizacje internetowe* narzędzia *Ustawienia oprogramowania*, na której można skonfigurować parametry automatycznego sprawdzania aktualizacji oraz reakcji na wykrycie dostępności takiej aktualizacji.

Rysunek 20.4.
Strona Aktualizacje internetowe narzędzia Ustawienia oprogramowania



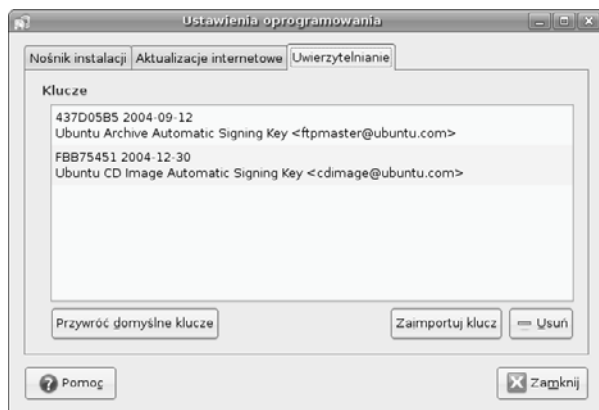
Domyślnie system automatycznie sprawdza dostępność aktualizacji raz dziennie, wyświetlając na panelu GNOME ikonę *Dostępność aktualizacji* z informacją o liczbie dostępnych nowych wersji pakietów (patrz rysunek 20.28). Użytkownicy laptopów i innych systemów niesystematycznie podłączanych do internetu mogą zmniejszyć częstotliwość sprawdzania aktualizacji do odpowiedniej wartości z menu rozwijanego *Sprawdzaj dostępność aktualizacji* automatycznie lub zupełnie zablokować tę opcję. W przypadku zablokowania tej opcji można ręcznie uruchamiać program *Menedżer aktualizacji*, omówiony w dalszej części rozdziału, w podrozdziale „Wykorzystanie menedżera aktualizacji”. Jeśli jednak decydujemy się regularnie sprawdzać dostępność aktualizacji, a system jest często (lub zawsze) podłączony do internetu, warto zaznaczyć opcję *Pobieraj aktualizacje w tle, ale ich nie instaluj*, dzięki czemu aktualizacje zostaną pobrane, zanim użytkownik zdecyduje się je zainstalować. Gdy mamy stały dostęp do internetu, warto zastanowić się nad uaktywnieniem opcji *Instaluj aktualizacje bezpieczeństwa bez potwierdzenia*, co spowoduje, że system automatycznie zainstaluje dostępne instalacje bezpieczeństwa, dzięki czemu zawsze będzie w pełni zabezpieczony przed znanymi atakami i dziurami w oprogramowaniu.

Na rysunku 20.5 przedstawiam stronę *Uwierzytelnianie* narzędzia *Ustawienia oprogramowania*, która usprawnia procedurę importowania kluczy elektronicznych służących uwierzytelnianiu repozytoriów.

Każdy pakiet w dowolnym repozytorium jest podpisany kluczem służącym do weryfikacji autentyczności i integralności. Narzędzia do zarządzania pakietami w Ubuntu nie będą pobierały pakietów z nieautoryzowanych i niezidentyfikowanych repozytoriów. Dodanie klucza uwierzytelniającego dla niestandardowego repozytorium wymaga wydania w wierszu poleceń kilku komend wykorzystujących trudne do zapamiętania opcje programu `gpg`

Rysunek 20.5.

Strona
Uwierzytelnianie
narzędzia Ustawienia
oprogramowania

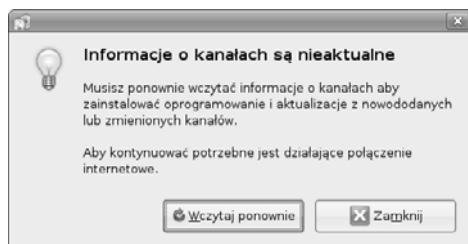


w połączeniu z narzędziem `apt-key`. Kliknięcie klawisza *Zaimportuj plik klucza* pozwala zaimportować klucz PGP zapisany w pliku ASCII i wykorzystać go do potrzeb narzędzi zarządzania pakietami.

Po wprowadzeniu niezbędnych zmian w liście repozytoriów dostępnych dla systemu, sposobie realizacji funkcji aktualizacji pakietów w systemie oraz na liście kluczy uwierzytelniających wykorzystywane repozytoria można zamknąć okno *Ustawienia oprogramowania*, klikając przycisk *Zamknij*. Jeśli zostały wprowadzone zmiany, program wyświetli komunikat informujący o tym, że informacja o kanałach (repozytoriach) jest nieaktualna, co przedstawiam na rysunku 20.6.

Rysunek 20.6.

Informacja
o zmianach
w narzędziu
Ustawienia
oprogramowania



Po kliknięciu przycisku *Wczytaj ponownie* program odświeży informacje bufora informacji o pakietach systemu Ubuntu. Można też opuścić narzędzie bez odświeżania informacji o pakietach, klikając przycisk *Zamknij*. W takim przypadku należy uruchomić w terminalu polecenie `apt-get update` lub kliknąć przycisk *Sprawdź* narzędzia *Menedżer aktualizacji* lub *Menedżer pakietów Synaptic*, inaczej system nie będzie miał aktualnych informacji o dostępnych pakietach we wszystkich repozytoriach.

Problemy z dodawaniem i dostępem do niestandardowych repozytoriów

Po zmodyfikowaniu pliku `/etc/apt/sources.list` może okazać się, że pakiety z dodanego właśnie repozytorium nie pojawiają się w liście dostępnych pakietów. Na początek należy sprawdzić, czy zostały odświeżone informacje o dostępnych pakietach (przycisk *Odśwież* programu Synaptic). Jeśli niektóre repozytoria są niedostępne lub w pliku znalazł się

błąd literowy czy składniowy, ta operacja wyświetli stosowny komunikat, który pozwoli zorientować się w sytuacji.

Jeśli to nie pomoże, należy uruchomić polecenie `apt-get update`, które wypisuje bardziej szczegółowe informacje na temat tego, co dzieje się z repozytoriami. Przykładowo dla komunikatów o zignorowaniu niektórych wpisów mamy sygnał, że nie zostały załadowane stosowne klucze uwierzytelniające:

```
Ign http://ftp.us.debian.org sarge Release
Ign http://ftp.us.debian.org unstable Release
```

W takim przypadku po zakończeniu operacji `apt-get update` pojawiają się dodatkowe komunikaty. Tym razem będą mniej subtelne i jednoznacznie wskażą, że powodem błędów jest brak odpowiednich kluczy:

```
W: GPG error: http://ftp.us.debian.org testing Release: \
The following signatures couldn't be verified because the public
key is not available: \
NO_PUBKEY 010908312D230C5F
W: GPG error: http://ftp.us.debian.org unstable Release: \
The following signatures couldn't be verified because the public
key is not available: \
NO_PUBKEY 010908312D230C5F
```

Otrzymaliśmy jednoznaczny sygnał, że w pliku `/etc/apt/sources.list` znajdują się wpisy, dla których brakuje kluczy potrzebnych do uwierzytelniania repozytoriów systemu Ubuntu. Informacje na temat pobierania i instalowania kluczy repozytoriów można znaleźć w podręczniku systemowym polecenia `apt-key` (`man apt-key`). Przykład procedury pobierania i instalowania kluczy repozytoriów do pęku kluczy narzędzi zarządzania pakietami systemu Ubuntu został opisany w kolejnym punkcie rozdziału.

Mieszanie repozytoriów Ubuntu i Debiana

Ubuntu Linux swoją koncepcję zarządzania pakietami zawdzięcza projektowi Debian, co powoduje, że zarówno Debian, jak i Ubuntu wykorzystują plik `/etc/apt/sources.list` do identyfikacji źródeł pakietów, oba też wykorzystują format pakietów DEB do rozpowszechniania pakietów, ich instalowania i zarządzania nimi. To wszystko powoduje, że kusząca wydaje się możliwość dodania repozytoriów Debiana do pliku `/etc/apt/sources.list` w systemie Ubuntu, co mogłoby dodatkowo zwiększyć ilość już dostępnego oprogramowania. Najlepsza reguła dotycząca mieszania w jednym systemie repozytoriów Debiana i Ubuntu jest następująca: NIE RÓB TEGO!

Niemniej jednak, w dalszej części tego punktu omówię zasady użytkowania jednocześnie repozytoriów Ubuntu i Debiana dla wszystkich tych, którzy zdecydują się zignorować moje ostrzeżenie, bo wiedzą, co robią (a w każdym razie tak sądzą). Na wszelki wypadek jednak przedstawię kilka dalszych ostrzeżeń. Mieszanie repozytoriów Ubuntu i Debiana może doprowadzić Ubuntu do awarii, zarytować znajomych użytkowników zarówno Ubuntu, jak i Debiana, doprowadzić do powstania systemu, w którym pochodzenie poszczególnych pakietów będzie zupełnie niejasne. Nie chcę ponosić odpowiedzialności za to, że niegdyś śliczny system Ubuntu zmieni się w dymiący krater po systemie, którego nie uda się już uruchomić ani przywrócić do życia. Ostrzegalem (i to nie pierwszy raz)!

Jak wspominałem w poprzednim podrozdziale, definicje nowych repozytoriów można bez przeszkód dopisać w pliku `/etc/apt/sources.list`, wykorzystując program Synaptic lub bezpośrednio modyfikując pliki konfiguracyjne systemu wykorzystywane przez programy Synaptic, aptitude i apt-get. W tym podrozdziale omówię sposób ręcznej modyfikacji tych plików tak, aby wymienione programy zaakceptowały repozytoria Debiana.

Dodanie repozytorium Debiana do listy repozytoriów wymaga następujących działań.

1. Do pliku `/etc/apt/sources.list` należy dopisać odpowiednie repozytoria. Repozytoria Debiana wykorzystują wydanie Debiana lub nazwy odgałęzień dystrybucji, nazwy komponentów również różnią się od analogicznych nazw w standardowym repozytorium Ubuntu: *main*, *non-free* i *contrib*. Dodając repozytorium Debiana do systemu Ubuntu, prawdopodobnie warto dodać odgałęzienia *testing* i *unstable*. W skrócie, w pliku `/etc/apt/sources.list` znajdują się następujące wpisy:

```
deb http://ftp.us.debian.org/debian testing main contrib non-free
deb-src http://ftp.us.debian.org/debian testing main contrib non-free
deb http://ftp.us.debian.org/debian unstable main contrib non-free
deb-src http://ftp.us.debian.org/debian unstable main contrib non-free
```

2. Dodając niestandardowe repozytoria, warto jawnie nadać im priorytety, które będą wykorzystywać apt-get i Synaptic do decydowania, w jakiej kolejności przeszukiwać dostępne repozytoria. Tę informację zapisuje się w pliku `/etc/apt/preferences`, w standardowej instalacji Ubuntu plik ten nie występuje. Należy utworzyć go za pomocą programu tekstowego (i polecenia `sudo`) i wpisać w nim priorytety dotyczące repozytoriów określonych w pliku `/etc/apt/sources.list`. W moim pliku zapisałem następującą konfigurację:

```
Package: *
Pin: release a=dapper
Pin-Priority: 900
Package: *
Pin: release o=debian a=testing
Pin-Priority: 500
Package: *
Pin: release o=debian a=unstable
Pin-Priority: 400
```

Priorytety o wartościach poniżej 1000 mają specjalne znaczenie dla narzędzi do zarządzania pakietami w Ubuntu. Wartości z powyższego przykładu nadają najwyższy priorytet pakietom z wydania *dapper*, niższy priorytet — pakietom z wydania *testing* dystrybucji Debian, a najniższy — pakietom z wydania *unstable* Debiana. Różne zakresy priorytetów mają odmiennie znaczenia w pliku *preferences*. Nie chciałbym pisać od nowa podręcznika systemowego dla pliku *preferences* (zainteresowanych odsyłam do polecenia `man apt_preferences`), dość stwierdzić, że wartości dla gałęzi *testing* i *unstable* Debiana z powyższego przykładu spowodują, że pakiety z tych wydań nie zostaną zainstalowane, o ile w wydaniu Dapper występuje pakiet o tej samej nazwie albo zainstalowany obecnie pakiet jest nowszy.

3. Należy zmodyfikować plik `/etc/apt/apt.conf`, zwiększając rozmiar pamięci dla odwzorowanego w pamięci pliku, w którym zapisywane są struktury danych przechowujące informacje o pakietach. Dodanie całego repozytorium Debiana do standardowych repozytoriów Ubuntu wymaga dużej ilości zasobów. Domyślny

rozmiar tego pliku to 4 MB (4194304 bajtów). Sugeruję zwiększyć ten rozmiar do 16 MB. W tym celu należy w pliku `/etc/apt/apt.conf` wpisać następujące ustawienie:

```
APT::Cache-Limit "16777216";
```

4. Należy również pobrać klucze uwierzytelniające repozytoria Debiana i zaimportować je do pęku kluczy narzędzi do zarządzania pakietami w Ubuntu. Jak pamiętamy, narzędzia w Ubuntu nie będą pobierały pakietów z nieautoryzowanych repozytoriów. Dodanie kluczy identyfikujących niestandardowe repozytoria jako prawidłowe wymaga wykonania kilku działań. Na początek, klucze repozytoriów Debiana są dostępne na publicznym serwerze, z którego można je zaimportować za pomocą polecenia `gpg` z opcją `--list-keys`. Odpowiednie do naszego celu polecenie będzie miało następującą składnię:

```
$ gpg -v --keyserver keyring.debian.org --list-keys
gpg: using classic trust model
-----
[existing key output deleted]
pub 1024D/2D230C5F 2006-01-03 [expires: 2007-02-07]
uid Debian Archive Automatic Signing Key (2006) <ftpmaster@debian.org>
```



Aktualny klucz repozytoriów Debiana w formacie ASCII można pobrać bezpośrednio z adresu http://ftp-master.debian.org/ziyi_key_rok.asc (gdzie **rok** jest zapisany w postaci czterocyfrowej, np. 2007), po czym można go zaimportować do pęku kluczy. Ta metoda jest jednak mniej bezpieczna w porównaniu z pobraniem klucza bezpośrednio z serwera kluczy.

5. Teraz trzeba pobrać klucz w oparciu o jego „odcisk palca” (w powyższym przykładzie to wartość 2D230C5F) do lokalnego pęku kluczy z użyciem opcji `--recv-keys` polecenia `gpg`. Odpowiednie polecenie dla naszego przykładu będzie miało następującą postać:

```
$ gpg -v --keyserver keyring.debian.org --recv-keys 2D230C5F
gpg: requesting key 2D230C5F from hkp server keyring.debian.org
gpg: armor header: Version: GnuPG v1.4.1 (GNU/Linux)
gpg: armor header: Comment: Key ID: 0x2D230C5F
gpg: pub 1024D/2D230C5F 2006-01-03 Debian Archive Automatic Signing Key (2006)
<ftpmaster@debian.org>
gpg: using classic trust model
gpg: key 2D230C5F: public key "Debian Archive Automatic Signing Key (2006)
<ftpmaster@debian.org>" imported
gpg: 2 keys cached (7 signatures)
gpg: 1 keys processed (1 validity counts cleared)
gpg: 3 marginal(s) needed, 1 complete(s) needed, classic trust model
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: Total number processed: 1
gpg: imported: 1
```

6. Po pobraniu klucza należy wyeksportować go do pęku kluczy narzędzi do zarządzania pakietami systemu Ubuntu, wykorzystując potok programów `gpg`, `sudo` i `apt-key`, np.:

```
$ gpg --armor --export 2D230C5F | sudo apt-key add -
OK
```

Opcja `--armor` opakowuje klucz w standardowe deskryptory ASCII, natomiast opcja `--export` eksportuje klucz o określonym „odcisku palca”.



Polecenie `gpg --armor --export` można wykorzystać do eksportu w formacie ASCII dowolnego klucza z pęku kluczy, np. w celu zapisania go w pliku testowym (przez zwykłe przekierowanie). Taki plik można wykorzystać z aplikacją, która wymaga klucza w formacie ASCII, taką jak narzędzie *Ustawienia oprogramowania* opisane wcześniej w podrozdziale „Uaktywnianie dodatkowych repozytoriów z użyciem narzędzia Ustawienia oprogramowania”.

W tym momencie można wywołać polecenie `apt-get update` w celu zaktualizowania informacji o repozytoriach i pakietach, wykorzystywanej przez narzędzia do zarządzania pakietami w Ubuntu. Następnie można instalować pakiety udostępniane przez te repozytoria. Powodzenia!



W przypadku problemów po wprowadzeniu zmian w ramach pliku *preferences* można posłużyć się poleceniem `apt-cache policy nazwa-pakietu`, które wypisze informację na temat priorytetu przypisanego danemu pakietowi w ramach poszczególnych źródeł. Więcej informacji na temat programu `apt-cache` można uzyskać w podręczniku systemowym (`man apt-cache`).

Eksploracja systemu z użyciem programu `dpkg` i pokrewnych

Oprogramowanie do zarządzania pakietami nie tylko upraszcza instalację, usuwanie i aktualizację oprogramowania, lecz również daje możliwość zadawania zapytań na temat zainstalowanego oprogramowania lub możliwego do instalacji w systemie. Przykładowo programy do zarządzania pakietami obsługiwane z wiersza poleceń pozwalają dokładnie sprawdzić, jakie oprogramowanie jest zainstalowane w systemie, co dokładnie zawiera określony pakiet, które pakiety zawierają określony plik itp. W kilku kolejnych punktach wyjaśnię, w jaki sposób zadawać zapytania na temat pakietów zainstalowanych i dostępnych do instalacji w systemie Ubuntu.

Sporządzanie listy pakietów zainstalowanych w systemie

Najprostszy sposób na sporządzenie listy pakietów zainstalowanych w systemie jest polecenie `dpkg -l` (istnieje też rozwinięta wersja tej opcji: `--long`). Po wywołaniu tego polecenia bez żadnych argumentów uzyskamy listing pakietów zainstalowanych w systemie:

```
$ dpkg -l
Wybór=U=Nieznany/I=Instalacja/R=Usunięcie/P=Wyczyszczenie/H=Zatrzymanie
| Stan=N=Brak/I=Zainst./C=Skonfig./U=Rozpakowany/F=Nieskonfig./H=Wpół-zainst.
|/ Błędy?=(brak)/H=Wstrzym./R=Do przeinst./X=Obydwa (Stan,Błędy:wielk.lit.=żle)
||/ Nazwa Wersja Opis
+++-----
```

ii	acpi	0.09-1	displays information on ACPI devices
ii	acpi-support	0.85	a collection of useful events for acpi
ii	acpid	1.0.4-1ubuntu1	Utilities for using ACPI power management

```
ii adduser      3.80ubuntu2    Add and remove users and groups
ii alacarte     0.8-0ubuntu12  easy menu editing
[...]
```

Kilka pierwszych wierszy listingu stanowi legendę, objaśniającą znaczenie poszczególnych pól wyniku i występujących w nich symboli. Informacje są dość mało czytelne, co jest podyktowane ograniczeniami tekstowego formatu wyników. Pozycje dotyczące nazwy pakietu, wersji i opisu są dość oczywiste, natomiast nieco więcej uwagi wymaga pierwsze pole. Znaki w tym polu mają następujące znaczenie (i potencjalne wartości).

- ♦ Pierwszy znak w pierwszym polu określa pożądany status pakietu, tzn. status, do którego w systemie pakietów należy sprowadzić pakiet. W tym polu mogą wystąpić znaki:
 - ♦ h — pakiet jest zatrzymany (ang. *on hold*) i nie może być zaktualizowany ani usunięty;
 - ♦ i — pakiet jest zainstalowany (ang. *installed*);
 - ♦ p — pakiet i cała jego konfiguracja mają być odinstalowane i wyczyszczone (ang. *purged*);
 - ♦ r — pakiet jest przeznaczony do usunięcia (ang. *remove*), ale jego pliki konfiguracyjne pozostaną nienaruszone;
 - ♦ u — pakiet nigdy nie był zainstalowany w systemie, zatem jego status jest nieznan (ang. *unknown*).
- ♦ Drugi znak w pierwszym polu oznacza rzeczywisty status pakietu w systemie. Dostępne znaki w tym polu:
 - ♦ c — pliki konfiguracyjne (ang. *configuration files*) pakietu są zainstalowane, ale sam pakiet nie;
 - ♦ f — skrypt wykorzystywany do konfiguracji pakietu po instalacji (znany jako skrypt poinstalacyjny) zakończył się błędem (ang. *failed*), co spowodowało, że pakiet nie jest prawidłowo zainstalowany;
 - ♦ h — pakiet jest zainstalowany częściowo (ang. *half-installed*), ponieważ proces instalacyjny został przerwany;
 - ♦ i — pakiet jest zainstalowany prawidłowo (ang. *installed*);
 - ♦ n — pakiet nie jest zainstalowany (ang. *not installed*);
 - ♦ u — pakiet został pobrany, rozpakowany (ang. *unpacked*) i częściowo zainstalowany, ale skrypt poinstalacyjny nie został uruchomiony.
- ♦ Trzeci znak określa błędy związane z danym pakietem. Dostępne znaki w tym polu to:
 - ♦ spacja — brak błędów, co jest szczególnie rzucające się w oczy, bo na tej pozycji nie występuje żaden widoczny znak;
 - ♦ H — pakiet jest oznaczony jako wstrzymany (ang. *on Hold*) przez sam system zarządzania pakietami, co z reguły oznacza, że nie zostały zainstalowane inne pakiety wymagane przez dany pakiet;

- ♦ R — wymagana jest reinstalacja pakietu (ang. *Reinstallation*);
- ♦ X — pakiet wymaga reinstalacji i został wstrzymany przez sam system zarządzania pakietami.

Po poznaniu znaczenia znaków sygnalizujących stan pakietu łatwo zrozumieć wynik działania polecenia `dpkg -l`. Nie jest to jednak coś, co można by nazwać „intuicyjną obsługą”, lecz trudno liczyć na taki komfort w środowisku znakowym, ograniczonym do kilkudziesięciu liter, kilku cyfr i kilku znaków przestankowych.

Sporządzanie listy pakietów dostępnych do instalacji

W poprzednim punkcie objaśniłem, w jaki sposób sprawdzić pakiety zainstalowane w systemie. To interesujące, ale zapewne ciekawsze jest pytanie: „Jakie pakiety są dostępne dla mojego systemu, których jeszcze nie zainstalowałem?”. Jak wspominałem w poprzednim punkcie, opcja `-l` programu `dpkg` bez argumentów dodatkowych spowoduje wypisanie wszystkich zainstalowanych pakietów. Jeśli programowi `dpkg` z tą opcją zada się argument, program wypisze listę dopasowaną do argumentu, niezależnie od tego, czy pakiet jest zainstalowany, czy też nie. Przykładowo polecenie `dpkg -l emacs` powinno wypisać wszystkie pakiety o nazwie `emacs`. Spróbujmy:

```
$ dpkg -l emacs
No packages found matching emacs.
```

To dość dziwne, ponieważ ten tekst piszę właśnie w programie `emacs` w systemie Ubuntu. Problem polega jednak na tym, że argument dostarczony z poleceniem nie jest dopasowywany w sposób dowolny, ale jest traktowany jako wzorzec dopasowania, a jak się wydaje, w systemie nie są zainstalowane pakiety o nazwie `emacs`. Spróbujmy ponownie, tym razem wykorzystując standardowy w Linuksie znak dopasowania, czyli gwiazdkę:

```
$ dpkg -l 'emacs*'
Wybór=U=Nieznany/I=Instalacja/R=Usunięcie/P=Wyczyszczenie/H=Zatrzymanie
| Stan=N=Brak/I=Zainst./C=Skonfig./U=Rozpakowany/F=Nieskonfig./H=Wpół-zainst.
|/ Błędy?=(brak)/H=Wstrzym./R=Do przeinst./X=Obydwa (Stan,Błędy:wielk.lit.=źle)
||/ Nazwa          Wersja          Opis
+++-----+-----+-----+
ii  emacs-chess     2.0b5-1         a client and library for playing Chess from
ii  emacs-chess-pi  2.0b5-1         XPM images of chess pieces for emacs-chess
ii  emacs21         21.4a-3ubuntu2  The GNU Emacs editor
ii  emacs21-bin-co  21.4a-3ubuntu2  The GNU Emacs editor's shared, architecture
ii  emacs21-common  21.4a-3ubuntu2  The GNU Emacs editor's shared, architecture
un  emacs21-el      <brak>          (brak dostępnego opisu)
un  emacs21-nox     <brak>          (brak dostępnego opisu)
un  emacs21-nox     <brak>          (brak dostępnego opisu)
un  emacs21-nox     <brak>          (brak dostępnego opisu)
ii  emacs21-nox     1.4.17         Common facilities for all emacs21
```

Tym razem dużo lepiej! Warto zwrócić uwagę, że stosując gwiazdkę w poleceniu tego typu, należy zabezpieczyć ją przed przedwczesnym rozwinięciem jej w poleceniu (co może spowodować dopasowanie do plików w bieżącym katalogu), a więc wystarczy ująć tekst w cudzysłowy.

instalacji w ramach repozytoriów zdefiniowanych w pliku */etc/apt/sources.list*. Oto przykładowy wynik wyszukiwania pakietów zawierających słowo kluczowe *emacs*:

```
$ aptitude search emacs
p  acl2-emacs          - A Computational Logic for Applicative Comm
p  aleph-emacs         - The Aleph programming language - emacs mod
p  cxref-emacs         - Generates latex and HTML documentation for
p  emacs               - The GNU Emacs editor
i  emacs-chess         - a client and library for playing Chess fro
i  emacs-chess-pieces  - XPM images of chess pieces for emacs-chess
p  emacs-color-themes  - Color themes for Emacs
p  emacs-extra         - emacs configuration
i  emacs-goodies-el    - Miscellaneous add-ons for Emacs
i  emacs21             - The GNU Emacs editor
i  emacs21-bin-common  - The GNU Emacs editor's shared, architectur
i  emacs21-common      - The GNU Emacs editor's shared, architectur
[...]
```

Jak można zauważyć, program *aptitude* znajduje więcej pakietów niż opisane wcześniej polecenie *dpkg -l*.

Inną, bardzo wygodną w użyciu aplikacją do wyszukiwania pakietów jest *apt-cache*, program wywoływany z wiersza poleceń i obsługujący polecenia do wyszukiwania i zarządzania pamięcią *cache* pakietów zainstalowanych lub dostępnych do instalacji w systemie oraz metadanymi związanymi z tymi pakietami. Aby zainstalować program *apt-cache*, należy uaktywnić repozytorium *universe*. Programu *apt-cache* można również użyć do wyszukiwania pakietów w dowolnych repozytoriach zdefiniowanych w pliku */etc/apt/sources.list*, w wyszukiwaniu jest nawet skuteczniejszy od programu *aptitude*, ponieważ szuka w nazwie pakietu oraz w jego opisie (skrótowym i pełnym), niezależnie od tego, czy jest zainstalowany w systemie. Oto przykład wywołania polecenia *apt-cache search* w celu wyszukania naszego ulubionego słowa kluczowego *emacs*:

```
$ apt-cache search emacs
acl2-emacs - A Computational Logic for Applicative Common Lisp: emacs interface
ada-mode - Ada mode for GNU Emacs and XEmacs
af - An Emacs-like mail reader and composer
aleph-emacs - The Aleph programming language - emacs mode
anjsp - A major mode to edit JSP and PSP code with Emacs
anthy-el - A Japanese input method (elisp fronted)
apel - portable library for emacsen
aplust-fsf-el - XEmacs lisp for A+ development
artist - Emacs Lisp drawing package
asn1-mode - Emacs mode for editing ASN.1 specification files
bhl - Emacs mode for converting brut text to HTML and LaTeX
bigloo-ude - Bigloo Unified Development Environment for Emacs
biomode - [Biology] An Emacs mode to edit genetic data
[...]
```

Jak widać, polecenia *apt-cache* i *aptitude* są najwygodniejsze w użyciu przy wyszukiwaniu pakietów i zwracają najbardziej szczegółowe wyniki.

Odczyt informacji o pakiecie

Po odszukaniu nazwy pakietu, którym jesteśmy zainteresowani, możemy uzyskać o nim więcej informacji oraz poznać jego zawartość; istnieje na to kilka sposobów. Jak mieliśmy okazję się przekonać, polecenia `apt-cache` i `aptitude` zapewniają najlepsze wyniki wyszukiwania, jak się zatem wydaje, sens ma również wykorzystanie ich do zasięgnięcia bardziej szczegółowych informacji o konkretnym pakiecie. Przykładowo w celu wyszukiwania szczegółowych informacji o pakiecie *xemacs21* znalezionym w poprzednim punkcie zastosujemy polecenia `apt-cache show xemacs21` lub `aptitude show xemacs21`. Oba zwracają prawie identyczne informacje. Poniższy przykład przedstawia wynik polecenia `aptitude show xemacs21` (który jest nieco bardziej szczegółowy w porównaniu z wynikiem polecenia `apt-cache show xemacs21`):

```
$ aptitude show xemacs21
Package: xemacs21
New: yes
State: installed
Automatically installed: no
Version: 21.4.18-1ubuntu1
Priority: optional
Section: universe/editors
Maintainer: OHURA Makoto <ohura@debian.org>
Uncompressed Size: 49.2k
Depends: xemacs21-mule (= 21.4.18-1ubuntu1) | xemacs21-mule-canna-wnn
        (= 21.4.18-1ubuntu1) | xemacs21-nomule (= 21.4.18-1ubuntu1) |
        xemacs21-gnome-mule (= 21.4.18-1ubuntu1) |
        xemacs21-gnome-mule-canna-wnn (= 21.4.18-1ubuntu1) |
        xemacs21-gnome-nomule (= 21.4.18-1ubuntu1)
Conflicts: xemacs, xemacs-widget
Replaces: xemacs, xemacs-widget
Provided by: xemacs21-nomule, xemacs21-mule-canna-wnn, xemacs21-mule,
            xemacs21-gnome-nomule, xemacs21-gnome-mule-canna-wnn,
            xemacs21-gnome-mule
Description: highly customizable text editor
 XEmacs is a full fledged programming language with a mail reader, news
 reader, info browser, Web browser, calendar, specialized editor for more
 programming languages and other formats than most people encounter in a
 lifetime, and much much more.
 This package exists to cause the installation of the real XEmacs packages.
```

Ten wynik prawdopodobnie zawiera więcej informacji, niż chcielibyśmy uzyskać, ale — jak wiadomo — od przybytku głowa nie boli.

Odczyt informacji o zawartości pakietu

W niektórych przypadkach pomocna może okazać się znajomość zawartości pakietu zainstalowanego w systemie. Do tego celu służy opcja `-L` polecenia `dpkg`, po której należy podać nazwę pakietu. Przykładowy wynik takiego polecenia przedstawia poniższy listing:

```
$ dpkg -L emacs21
/.
/usr
/usr/bin
/usr/bin/emacs21-x
```

```

/usr/share
/usr/share/emacs
/usr/share/emacs/21.4
/usr/share/emacs/21.4/etc
/usr/share/emacs/21.4/etc/DOC-21.4.1
/usr/share/applications
/usr/share/applications/emacs21.desktop
/usr/share/doc
/usr/share/doc/emacs21
/usr/share/doc/emacs21/README.Debian.gz
/usr/share/doc/emacs21/copyright
/usr/share/doc/emacs21/changelog.Debian.gz
/usr/lib
/usr/lib/emacs
/usr/lib/emacs/21.4
/usr/lib/emacs/21.4/i486-linux-gnu
/usr/lib/emacs/21.4/i486-linux-gnu/fns-21.4.1-x.el
/usr/lib/menu

```

Możliwość zidentyfikowania plików dostarczanych z pakietem zainstalowanym w systemie może być przydatna podczas rozważania możliwości usunięcia pakietu, gdy obawiamy się, czy nie usuniemy plików, które jednak są potrzebne. W następnym punkcie opiszę sposoby wyszukiwania pakietów, w których znajduje się określony plik.



Pliki wchodzące w skład niezainstalowanego pakietu można wylistować za pomocą programu `apt-file list nazwa_pakietu`. Więcej informacji na temat programu `apt-file` można znaleźć w punkcie, „Określenie pakietu, w którym znajduje się brakujący plik”, w dalszej części rozdziału.

Określanie pakietu, do którego należy istniejący plik

Po pewnym okresie pracy z Ubuntu, instalowaniu, aktualizowaniu i usuwaniu pakietów użytkownik może zainteresować się tym, w którym pakiecie znajduje się określony plik zainstalowany w systemie. Dość łatwo go znaleźć, wykorzystując polecenie `dpkg --search`. Poszukajmy np. pakietu, w którym znajdował się program `/usr/bin/ar`:

```

$ dpkg --search /usr/bin/ar
binutils: /usr/bin/ar

```

Jeśli wiadomo, że program, dla którego chcemy sprawdzić pakiet, znajduje się w ścieżce, ale nie znamy jej dokładnie, można posłużyć się przydatną sztuczką pozwalającą przekazać wynik jednego polecenia w charakterze parametru innego. W tym przypadku połączymy wynik polecenia `which`, zwracającego pełną ścieżkę do polecenia, z poleceniem `dpkg --search`, co w wyniku da dokładnie ten sam wynik, co powyższy przykład:

```

$ dpkg --search `which ar`
binutils: /usr/bin/ar

```

Przy wyszukiwaniu plików polecenie `dpkg --search` nie znajduje dowiązań symbolicznych stworzonych przez skrypty poinstalacyjne (*post-install*). Próbując bezskutecznie znaleźć pakiet zawierający istniejący plik, należy sprawdzić czy ów plik jest w istocie plikiem, a nie dowiązaniem symbolicznym. W tym drugim przypadku należy znaleźć rzeczywisty plik, na który wskazuje dane dowiązanie. Taka operacja czasem wymaga zastosowania kilku prób, np.:


```
$ dpkg --search /usr/bin/emacs
dpkg: /usr/bin/emacs not found.

$ ls -l /usr/bin/emacs
/usr/bin/emacs -> /etc/alternatives/emacs

$ dpkg --search /etc/alternatives/emacs
dpkg: /etc/alternatives/emacs not found.

$ ls -l /etc/alternatives/emacs
/etc/alternatives/emacs -> /usr/bin/emacs21-x

$ dpkg --search /usr/bin/emacs21-x
emacs21: /usr/bin/emacs21-x
```



Innym interesującym narzędziem służącym do wyszukiwania pakietów zawierających wybrany plik jest `dlocate`. Ten program nie jest domyślnie instalowany w Ubuntu, ale można go z łatwością zainstalować za pomocą programu `apt-get`, `aptitude` lub `Synaptic`. Pakiety można wyszukiwać dzięki opcji `-S` programu `dlocate`, np.:

```
$ dlocate -S /usr/bin/emacs
emacs21-bin-common: /usr/bin/emacscclient.emacs21
emacs21: /usr/bin/emacs21-x
emacs21: /usr/bin/emacs21
```

Opcja `-S` służy zdefiniowaniu ciągu znaków, który będzie wyszukiwany w charakterze podciągu, co jest dość wygodne przy wyszukiwaniu źródeł plików zainstalowanych w systemie.

Określenie pakietu, w którym znajduje się brakujący plik

Identyfikacja pakietu, w którego skład wchodzi plik nieistniejący w systemie, to dość trudne zadanie, ale dość powszechne, kiedy użytkownik np. kompiluje programy i okazuje się, że do poprawnej kompilacji brakuje określonego pliku nagłówkowego lub biblioteki. Niestety, program pomocny w tego typu sytuacjach nie jest instalowany domyślnie w Ubuntu, ale można go bez problemów zainstalować za pomocą programu `apt-get`, `aptitude` czy `Synaptic`. Po zainstalowaniu programu należy zaktualizować listę dostępnych pakietów, wykonując polecenie `sudo apt-file update`.

Po zakończeniu działania tego polecenia można podjąć próbę wyszukania pakietu zawierającego określony plik nawet wtedy, gdy ten pakiet nie jest zainstalowany w systemie. Służy do tego polecenie `apt-file search`. Spróbujmy np. wyszukać pakiet zawierający bibliotekę `libpowersave`:

```
$ ls -l /usr/lib/*power*
ls: /usr/lib/*power*: No such file or directory
$ apt-file search libpowersave.so
libpowersave-dev: usr/lib/libpowersave.so
libpowersave10: usr/lib/libpowersave.so.10
libpowersave10: usr/lib/libpowersave.so.10.0.3
```

Instalacja pakietu `apt-file` pozwoli oszczędzić wielu stresów i frustracji związanych z próbami zlokalizowania pakietu dostarczającego brakujący plik.



Za pomocą programu `apt-file` można również sprawdzić zawartość niezainstalowanego pakietu, służy do tego polecenie `apt-file list pakiet`, gdzie *pakiet* to nazwa konkretnego pakietu.

Usuwanie oprogramowania z użyciem programu apt-get

Polecenie `apt-get` jest jednym z podstawowych poleceń należących do zbioru narzędzi zarządzania pakietami dostępnych dla Ubuntu. Program `apt-get` jest wykorzystywany z wiersza poleceń, używa się go szybko i jest prosty w obsłudze.



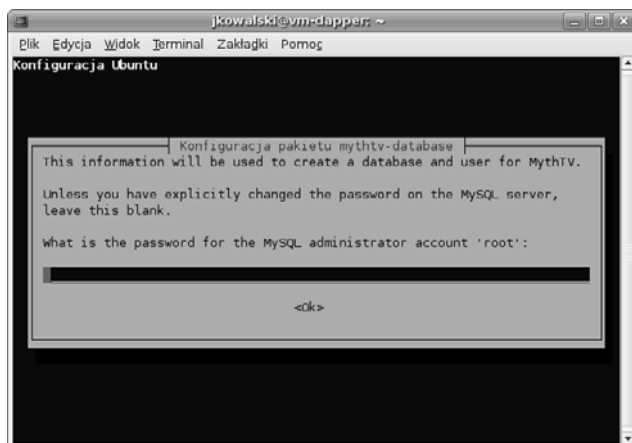
Podobnie jak w przypadku wielu innych narzędzi administracyjnych w Ubuntu, wykonywanie zadań z użyciem `apt-get` wymaga zastosowania polecenia `sudo` lub jego odpowiednika, np. przez uruchomienie powłoki w trybie administratora. Aby uprościć przykłady i czytelność tekstu, w dalszej części podrödziału przyjmę założenie, że polecenia są wykonywane z prawami administratora (a więc, że polecenie `sudo` jest wykonywane w sposób bezpośredni lub pośredni).

Większość zadań realizowanych przez `apt-get` wykonuje się w dość intuicyjny sposób. Przykładowo za pomocą polecenia `apt-get install pakiet` można zainstalować pakiet oraz wszystkie jego pakiety zależne. Można również usunąć pakiet (`apt-get remove pakiet`), a z opcją `--purge` zostaną też usunięte pliki konfiguracyjne i inne pliki związane z usuwanym pakietem. Przed użyciem polecenia `apt-get` w celu instalacji oprogramowania należy uruchomić polecenie `apt-get update`, które spowoduje odświeżenie baz informacji o dostępnych pakietach, wykorzystywanych przez program `apt-get`. Dzięki temu program będzie zawsze instalował najświeższe oprogramowanie z repozytoriów zdefiniowanych w pliku `/etc/apt/sources.list`.

Przy instalacji niektórych pakietów i wykorzystaniu polecenia `apt-get` do instalowania oprogramowania program w trakcie procesu instalacji może zadać użytkownikowi dodatkowe pytania. W takim przypadku zostanie wyświetlony pseudograficzny interfejs trybu tekstowego, przykład takiego interfejsu można zobaczyć na rysunku 20.7.

Rysunek 20.7.

Uzupełnianie informacji konfiguracyjnych podczas instalacji pakietu



Gdy pojawi się ekran tego typu, użytkownik musi uzupełnić niezbędne informacje, po czym instalacja pakietu będzie kontynuowana.



Gdy używamy programu `apt-get` do instalowania oprogramowania, może pojawić się komunikat, że zalecane lub sugerowane są dodatkowo inne pakiety. W takim wypadku warto wziąć pod uwagę instalację takiego pakietu z użyciem wywołania programu `aptitude` zamiast programu `apt-get`. Szczegóły omówię w punkcie „Instalacja i usuwanie oprogramowania z użyciem programu `aptitude`”, dość wspomnieć, że `aptitude` obsługuje specjalne opcje pozwalające zautomatyzować instalację pakietów wraz z ich zalecanymi pakietami.

Wielokrotnie wspominałem w tej książce, że moim ulubionym narzędziem do wyszukiwania, instalacji i usuwania oprogramowania jest graficzny program Synaptic. Program `apt-get` oferuje takie same możliwości, a oprócz tego wiele innych, bardzo wartościowych, których nie posiada Synaptic, oraz sporo funkcji zbliżonych, ale o wiele wygodniejszych w użyciu z wiersza poleceń niż z graficznego interfejsu użytkownika. Nawet zwolennicy programu Synaptic powinni zapoznać się z możliwościami programu `apt-get` w zakresie wykorzystania tych zaawansowanych funkcji.

Aktualizacja systemu z użyciem programu `apt-get`

Polecenie `apt-get upgrade` przeszukuje wszystkie repozytoria zdefiniowane w pliku `/etc/apt/sources.list` w poszukiwaniu nowych wersji pakietów zainstalowanych aktualnie w systemie, po czym pobiera i instaluje nowsze wersje pakietów. Przed wywołaniem tego polecenia należy uruchomić polecenie `apt-get update`, które aktualizuje bazę informacji o pakietach dostępnych w repozytoriach zdefiniowanych w pliku `/etc/apt/sources.list`.

Podczas procesu aktualizacji polecenie `apt-get upgrade` nie modyfikuje statusu pakietów znajdujących się w systemie. Chodzi tu jednak o status instalacji, nie o wersję pakietu. Jeśli bowiem w repozytoriach znajdzie się nowsza wersja pakietu, która wymaga instalacji nowej wersji innego pakietu, polecenie `apt-get upgrade` pobierze i zaktualizuje również wersję tego drugiego pakietu. Jeśli jednak nowsza wersja pakietu wymaga zainstalowania pakietu, który wcześniej nie był zainstalowany, lub wymaga usunięcia zainstalowanego pakietu, nowsza wersja pakietu nie zostanie zainstalowana. Aby tego dokonać, należy wywołać polecenie `apt-get dist-upgrade`, co zostało szczegółowo omówione w kolejnym punkcie.

Inteligentna aktualizacja systemu z użyciem programu `apt-get`

Różnica między poleceniem `apt-get upgrade` a poleceniem `apt-get dist-upgrade` polega na tym, że to drugie wykonuje procedurę tzw. inteligentnej aktualizacji. Oznacza to, że program postara się spełnić wszelkie wymagania nowych wersji pakietów, również związane z instalacją pakietów niezbędnych dla nowych wersji lub usunięciem pakietów będących z nimi w konflikcie. Jeśli np. nowa wersja pakietu wymaga zainstalowania nowego pakietu w systemie, polecenie `apt-get dist-upgrade` zainstaluje oba, natomiast polecenie `apt-get upgrade` nie zaktualizuje tego pakietu. Podobnie dzieje się, jeśli instalacja aktualizacji pakietu wymaga usunięcia innego pakietu zainstalowanego w systemie; polecenie `apt-get dist-upgrade` usunie problematyczny pakiet i zainstaluje aktualizację.

Nawet zagorzali zwolennicy programu Synaptic powinni co jakiś czas wywoływać polecenie `apt-get dist-upgrade`, dzięki czemu upewnią się, że wszystkie pakiety zainstalowane w systemie zostaną zaktualizowane. Co prawda, Synaptic posiada opcję „inteligentnej aktualizacji”, która służy temu samemu celowi, jednak, jak się okazuje w praktyce, uruchamianie co jakiś czas polecenia `apt-get dist-upgrade` jest nadal zalecane. Przy wykorzystaniu programu Synaptic symptomem sytuacji wymagającej użycia programu `apt-get` jest pojawianie się pakietów w stanie wstrzymanym (nie są aktualizowane). Co więcej, co jakiś czas znajdujemy też informację o tym, że zalecane jest wykonanie polecenia `apt-get dist-upgrade`.

Pobieranie kodów źródłowych pakietów z użyciem programu apt-get

Nawet w cudownym świecie stale aktualizowanych repozytoriów zawierających bogaty wybór oprogramowania może okazać się koniecznym samodzielna kompilacja pakietu oprogramowania. Najbardziej powszechną sytuacją, jest potrzeba zainstalowania poprawek usuwających poważne problemy z oprogramowaniem, gdy w oficjalnych repozytoriach (np. *backports*) nie pojawiły się jeszcze pakiety uwzględniające te poprawki. Może się również zdarzyć, że trzeba wprowadzić własne modyfikacje w kodzie źródłowym. Linux i jego ekosystem narzędzi i innych aplikacji opiera się w końcu na idei open source!

Pobieranie kodów źródłowych dzięki programowi `apt-get` jest bardzo prostym zadaniem. Jedynym warunkiem jest uaktywnienie pozycji rozpoczynających się od słowa kluczowego `deb-src` w pliku `/etc/apt/sources.list` dla tych repozytoriów, które zawierają binarną wersję pakietu, jaki ma zostać skompilowany ponownie. Kiedy np. binarny pakiet znajduje się w repozytorium *multiverse*, należy plik `/etc/apt/sources.list` skonfigurować w następujący sposób:

```
deb http://archive.ubuntu.com/ubuntu/ dapper multiverse
deb-src http://archive.ubuntu.com/ubuntu/ dapper multiverse
```

Gdy w pliku `/etc/apt/sources.list` znajdują się już odpowiednie definicje repozytoriów z pakietami źródłowymi, można wykorzystać polecenie `apt-get source`. Poniższy przykład demonstruje procedurę pobierania kodów źródłowych mojego ulubionego pakietu *MythTV*, czyli programowego magnetowidu:

```
$ apt-get source mythtv
Reading package lists... Done
Building dependency tree... Done
Need to get 9836kB of source archives.
Get: 1 http://.../multiverse mythtv 0.18.1-5ubuntu3 (dsc) [1084B]
Get: 2 http://... dapper/multiverse mythtv 0.18.1-5ubuntu3 (tar) [9817kB]
Get: 3 http://... dapper/multiverse mythtv 0.18.1-5ubuntu3 (diff) [17.3kB]
Fetched 9836kB in 1m31s (107kB/s)
dpkg-source: extracting mythtv in mythtv-0.18.1
dpkg-source: unpacking mythtv_0.18.1.orig.tar.gz
dpkg-source: applying ./mythtv_0.18.1-5ubuntu3.diff.gz
```

Przy pobieraniu kodu źródłowego pakietu polecenie `apt-get source` w rzeczywistości pobiera kilka plików, takich jak:

- ♦ plik z rozszerzeniem *diff.gz*: skompresowany plik zawierający dostępne łąty na standardowe pakiety źródłowe;
- ♦ plik z rozszerzeniem *DSC*: plik deskryptora pakietu Ubuntu identyfikujący zawartość pakietu, związane z nim pakiety binarne, zależności tego pakietu itp.
- ♦ plik z rozszerzeniem *tar.gz*: zdekompresowane archiwum zawierające oficjalny kod źródłowy pakietu.

Po pobraniu tych plików polecenie `apt-get source` rozpakowuje archiwum kodów źródłowych, tworząc katalog roboczy zawierający źródła, po czym automatycznie nakłada łąty dostępne (opcjonalnie) w skompresowanym pliku w formacie *diff*.

W rozdziale 18. omówiliśmy zagadnienie instalacji podstawowych pakietów wymaganych do kompilacji programów w Ubuntu. To może wystarczyć, gdy ktoś pisze własne programy lub po prostu samodzielnie kompiluje programy w Ubuntu, ale kiedy ktoś kompiluje skomplikowane programy wykorzystujące interfejs graficzny lub buduje własne pakiety DEB może pojawić się konieczność zainstalowania kilku dodatkowych pakietów. Wkrótce pojawi się frustracja spowodowana wzajemnymi zależnościami bibliotek, plików nagłówkowych i narzędzi, o których użytkownik nigdy wcześniej nie słyszał. Przy budowaniu pakietów DEB również potrzebne są liczne narzędzia, ale nie należy się przejmować, ponieważ program `apt-get` zawiera funkcję przydatną i w takich sytuacjach, co omówiłem w kolejnym punkcie.

Obsługa zależności kompilacyjnych przy użyciu programu `apt-get`

Wiele pakietów instalowanych w Linuksie jest zależna od innych pakietów, co ma sens, bo po co wynajdować koło od nowa, jeśli wystarczy dołączyć już istniejący kod. Jeśli jednak ktoś chce skompilować istniejący kod pakietu, może pogubić się w zależnościach, których nie jest w stanie przewidzieć. Rzadko bywają udokumentowane, pojawiają się natomiast jako iteracyjne sekwencje wzajemnych powiązań na etapie konfiguracji i kompilacji kodu. Tego typu kłopoty dodatkowo potęgują się podczas pracy z pakietami w innych formatach systemów zarządzania pakietami. Pakiety DEB wykorzystywane w Ubuntu i Debianie (i ich pochodnych) pozwalają w znacznym stopniu zredukować poziom komplikacji typowy dla innych formatów pakietów (jak RPM), ale do prawidłowego działania wymagają instalacji całego zestawu narzędzi. Początek pracy z systemem budowania pakietów może być dość trudny, jeśli nie wiadomo, jakie elementy są potrzebne w tym procesie.

Program `apt-get` zawiera doskonałą funkcję automatyzującą rozwiązywanie problemów tego typu. Jest to opcja `build-dep`, której użycie razem z nazwą pakietu spowoduje identyfikację zależności i zainstalowanie ze źródeł pakietów zawierających biblioteki i narzędzia niezbędne do kompilacji pakietu. Oto przykładowe wyjście z takiego wywołania, instalującego zależności kompilacyjne pakietu *MythTV*:

```
$ sudo apt-get build-dep mythtv
Reading package lists... Done
Building dependency tree... Done
The following NEW packages will be installed
  build-essential g++ g++-3.4 g++-4.0 libartsc0-dev libasound2-dev
```

```

libaudio-dev libdvb-dev libgl1-mesa-dev libglu1-mesa-dev
libjpeg62-dev liblame-dev liblcms1-dev liblircclient-dev
libmng-dev libmysqlclient14-dev libogg-dev libqt3-headers
libqt3-mt-dev libstdc++6-4.0-dev libstdc++6-dev libvorbis-dev
libxmu-dev libxmu-headers libxt-dev libxv-dev libxvmc-dev
libxxf86vm-dev mesa-common-dev qt3-dev-tools x11proto-video-dev
x11proto-xf86vidmode-dev
0 upgraded, 32 newly installed, 0 to remove and 0 not upgraded.
Need to get 14.2MB/17.9MB of archives.
After unpacking 63.4MB of additional disk space will be used.
Do you want to continue [Y/n]? Y
[...]
```

Jak widać w tym przykładzie, ręczne pobranie wszystkich pakietów niezbędnych do samodzielnej kompilacji pakietu *MythTV* wymaga wielu sekwencji kompilacyjnych. Identyfikacja pakietów zapewniających brakujące pliki i biblioteki może zająć dużo czasu, nawet za pomocą programu `apt-file` (patrz punkt „Określenie pakietu, w którym znajduje się brakujący plik”). Dzięki opcji `build-dep` programu `apt-get` całą tę operację można zautomatyzować, pozbywając się problemu; wtedy można się skupić na oprogramowaniu, a nie na infrastrukturze potrzebnej do jego instalacji.

Instalacja i usuwanie oprogramowania z użyciem programu aptitude

Jak wspomniałem wcześniej w tym rozdziale, program `dpkg` obsługuje koncepcyjny fundament zarządzania pakietami w systemach Ubuntu (i oczywiście Debian). To narzędzie, choć bardzo skuteczne, jest obsługiwane z wiersza poleceń, zatem jego wadą jest brak graficznego interfejsu użytkownika. Aby uzupełnić tę lukę, powstał program `dselect`, który pozwala realizować funkcję programu `dpkg` z poziomu pseudograficznego interfejsu (w trybie terminalowym) zrealizowanego z użyciem biblioteki *curses* (a właściwie *ncurses*). Program `dselect` nie uwolnił się jednak od swoich korzeni i zależności od wiersza poleceń, można go zatem obsługiwać i w ten sposób.

Program `apt-get` omówiony wcześniej w tym rozdziale został wyposażony w prostszy interfejs, na wyższym poziomie abstrakcji od oferowanego przez program `dpkg`. Jednak `apt-get` to nadal kolejny, typowy program obsługiwany z wiersza poleceń, a przez to nie oferuje interfejsu graficznego. Aby uzupełnić tę lukę, powstał program `aptitude`, lecz i on nie uwolnił się zupełnie od wiersza poleceń. Dzięki temu `aptitude` (jak jego poprzednik `dselect`) można obsługiwać z wiersza poleceń i za pomocą pseudograficznego (terminalowego) interfejsu użytkownika.

Osobiście uważam terminalowe interfejsy użytkownika, delikatnie mówiąc, za toporne. Z reguły unikam ich, o ile to możliwe, szczególnie w przypadkach, gdy mogę znaleźć ich odpowiedniki wyposażone w prawdziwy graficzny interfejs użytkownika, jakim w przypadku `aptitude` jest menedżer pakietów *Synaptic*.

Oprogramowanie można instalować, wykorzystując wiersz poleceń i wywołanie `aptitude install`, natomiast pakiety usuwa się za pomocą polecenia `aptitude remove`. W dalszej części rozdziału, w punkcie „Instalacja zalecanych pakietów z użyciem `aptitude`” opisuję

przydatne mechanizmy programu aptitude również w wywołaniach z wiersza poleceń. Program ten ma wielu zwolenników, przede wszystkim należą do nich osoby, które dobrze poznały jego możliwości. W kolejnym punkcie omówię absolutne podstawy użycia interfejsu użytkownika programu aptitude, skupiając się na wyjaśnieniu jego elementów.



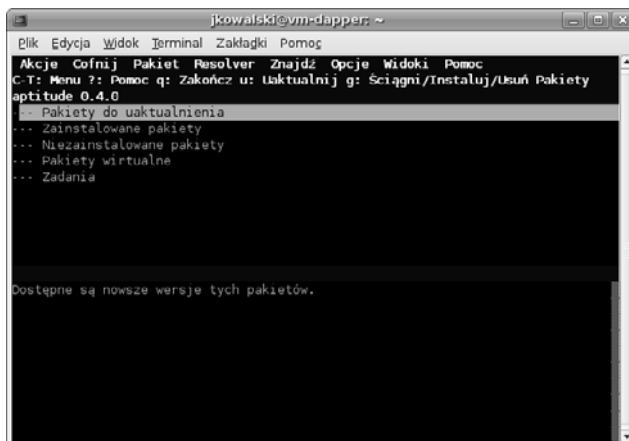
Podczas użytkowania programu aptitude może okazać się konieczne uzupełnianie informacji niezbędnych do kontynuacji procesu instalacyjnego. W takich przypadkach program aptitude wyświetla okna konfiguracyjne zbliżone do zaprezentowanego na rysunku 20.7 okna konfiguracyjnego programu apt-get. Użytkownik musi odpowiedzieć na pytania i uzupełnić niezbędne informacje, po czym instalacja może być kontynuowana.

Wskazówki dotyczące wykorzystania interfejsu użytkownika programu aptitude

Na rysunku 20.8 przedstawiam interfejs programu aptitude uruchomionego z terminala GNOME bez parametrów.

Rysunek 20.8.

Domyślny interfejs programu aptitude uruchomionego w terminalu GNOME



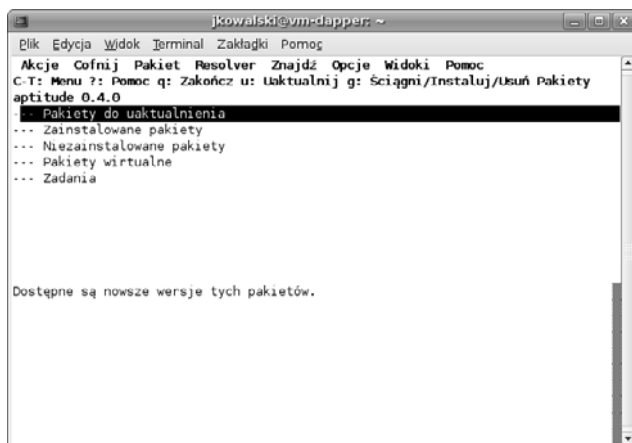
Ten interfejs wygląda tak samo w systemowym terminalu xterm i w dowolnej innej aplikacji terminalowej obsługującej kolor (w aplikacji xterm wygląd jest nieco mniej estetyczny, nie będę jednak męczył czytelnika zrzutami ekranu).

Jeśli zdecydowano się wykorzystywać interfejs graficzny programu aptitude, ale używany terminal ma problemy z wyświetleniem kolorów i pokazuje niezrozumiałe czcionki, to mamy pewien problem, ponieważ program aptitude nie obsługuje parametrów wywołania pozwalających zmodyfikować opcje kolorów i wyświetlania. Jednak nie wszystko jest stracone, czytelność interfejsu można poprawić, zmieniając bibliotekę obsługi interfejsu terminalowego, na której oparte jest działanie aptitude. W tym celu należy „zasugerować” tej bibliotece inny typ wykorzystywanego terminalu. Dobrym wyborem jest *vt100*, który jest klasycznym terminalem firmy Digital Equipment Corporation, obsługującym tylko czerń i biel. Aby biblioteka dostosowała się do wyświetlania w wybranym modelu terminalu, należy zamknąć aptitude (klawisz *Q*, przejście klawiszem *Tab* do pola potwierdzającego zamknięcie, *Enter*). Następnie trzeba wywołać następujące polecenie:

```
$ export TERM=vt100
```

Po jego wywołaniu można uruchomić aptitude, tym razem jego interfejs będzie bardziej ubogi kolorystycznie, co prezentuje rysunek 20.9.

Rysunek 20.9.
Interfejs programu
aptitude w czerni
i bieli



Zmiana schematu kolorystycznego nie powoduje, że interfejs aptitude stanie się bardziej atrakcyjny wizualnie, ale (przynajmniej dla mnie) zwiększa nieco jego użyteczność.

Program aptitude obsługuje się za pomocą poleceń wywoływanych naciśnięciem poszczególnych klawiszy, np.:

- ♦ *Q* — wyjście z aptitude;
- ♦ *?* — wyświetlenie panelu zawierającego informacje na temat użycia interfejsu programu;
- ♦ *Ctrl+H* — wyświetlenie menu programu aptitude, w których można nawigować za pomocą klawiszy strzałek lub naciskania klawiszy liter wyodrębnionych (wytluszczonych).

Na rysunku 20.10 przedstawiam menu programu aptitude w domyślnym schemacie kolorów tego programu. Czy trzeba to dalej komentować? Właśnie interfejs tego typu spowodował, że aplikacje linuksowe mają złą opinię wśród użytkowników innych systemów.

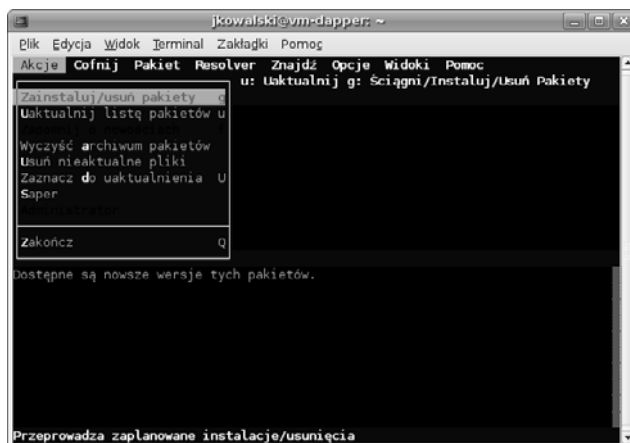
Bardziej szczegółowe (i mniej subiektywne) informacje na temat interfejsu programu aptitude można znaleźć w podręczniku systemowym (`man aptitude`) oraz na stronach i w forach WWW.

Instalacja zalecanych pakietów z użyciem aptitude

Jeśli do instalacji oprogramowania użyliśmy programu `apt-get` i program wyświetlił informację o tym, że zalecana jest instalacja dodatkowych pakietów, do instalacji takiego pakietu warto posłużyć się parametrem wywołania `aptitude` z wiersza poleceń, rezygnując w tym przypadku z użycia programu `apt-get`. Pakiety DEB używane w Ubuntu posiadają szereg metainformacji na temat zależności z innymi pakietami. Informacje te są podzielone na następujące kategorie.

Rysunek 20.10.

Menu w domyślnym interfejsie programu aptitude



- ♦ **Conflicts:** pakiety (lub wersje pakietów), z którymi dany pakiet jest w konflikcie i nie mogą równolegle z nim być zainstalowane w systemie.
- ♦ **Depends:** pakiety niezbędne danemu pakietowi do działania, które z tego powodu muszą być zainstalowane razem z pakietem.
- ♦ **Enhances:** pakiety, które stanowią opcjonalne rozszerzenie usprawniające lub upraszczające pracę instalowanego pakietu, ale nie są bezpośrednio zależne.
- ♦ **Pre-Depends:** pakiety, które muszą być zainstalowane w całości, zanim rozpocznie się instalacja danego pakietu.
- ♦ **Recommends:** pakiety, które powinny być zainstalowane równolegle z danym pakietem, ale nie są wymagane do jego instalacji.
- ♦ **Suggests:** pakiety, które mogą być użyteczne i z reguły bywają instalowane w systemie razem z danym pakietem.

Zależności zdefiniowane jako *Pre-Depends* (które są dość rzadko stosowane) muszą być spełnione, zanim rozpocznie się instalacja pakietu. Zależności zdefiniowane jako *Depends* są instalowane automatycznie z danym pakietem. Program aptitude pozwala dodatkowo wymusić automatyczną instalację pakietów zalecanych. W tym celu polecenie aptitude należy wywołać z wiersza poleceń z opcją `--with-recommends`.

Program aptitude jako jedyny z aplikacji obsługiwanych z wiersza poleceń pozwala na automatyczną instalację pakietów zalecanych razem z wymaganymi. Opcja `--with-recommends` zastosowana przy instalacji może stanowić znaczące usprawnienie instalacji. Co prawda, od dłuższego czasu sugeruje się, aby program apt-get uzupełnić o opcję `--recommends` do obsługi pakietów zalecanych oraz `--suggests` do automatycznej instalacji pakietów sugerowanych, ale nie znalazłem ich w żadnej z testowanych przeze mnie wersji programu apt-get. W punkcie poświęconym programowi Synaptic wspomnę szerzej o opcji pozwalającej wymusić traktowanie zalecanych pakietów jako wymagane, co wymusza ich bezwarunkową instalację. Można również skorzystać z opcji programu Synaptic pozwalającej w procesie zaznaczania pakietów do instalacji ręcznie wybrać te, które wydają się użyteczne.

Zalety użycia programu aptitude do instalacji i usuwania oprogramowania

Jak można się było przekonać, nie jestem wielkim zwolennikiem interfejsu użytkownika programu aptitude, ale muszę przyznać, że oferuje on użyteczne opcje. Jego funkcje wyszukiwania pakietów, omówione wcześniej w tym rozdziale (w punkcie „Sporządzanie listy pakietów dostępnych do instalacji”), są użyteczne i proste w użyciu. Aktualna wersja programu aptitude posiada też rzadziej używane funkcje instalacji i usuwania oprogramowania. Najważniejszą jest ta, że aptitude zapamiętuje dane o zależnościach pakietu podczas jego instalacji. Każde narzędzie do instalacji pakietów pod Ubuntu rozumie zależności i stosuje się do nich na etapie instalacji, ale aptitude zapamiętuje te zależności i wykorzystuje je przy usuwaniu zainstalowanych pakietów.

Jako przykład niech posłuży próba usunięcia pakietu bazy danych PostgreSQL z jednego z moich systemów Ubuntu. Najpierw spróbujemy polecenia apt-get:

```
$ sudo apt-get remove postgresql
Password:
Reading package lists... Done
Building dependency tree... Done
The following packages will be REMOVED
  postgresql
0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
Need to get 0B of archives.
After unpacking 45.1kB disk space will be freed.
Do you want to continue [Y/n]? n
Abort.
```

Jak widać, nie udało się. Natomiast aptitude poradzi sobie z tym zadaniem, ponieważ to narzędzie oferuje opcję usuwania pakietów, które były zainstalowane wspólnie z pakietem *postgresql*, ale nie będą już potrzebne, skoro ten pakiet ma zniknąć z systemu. Mamy więc:

```
$ sudo aptitude remove postgresql
Reading package lists... Done
Building dependency tree... Done
Reading extended state information
Initializing package states... Done
Building tag database... Done
The following packages are unused and will be REMOVED:
  postgresql-plperl-7.4 postgresql-plpython-7.4 postgresql-pltcl-7.4
The following packages will be REMOVED:
  postgresql
0 packages upgraded, 0 newly installed, 4 to remove and 0 not upgraded.
Need to get 0B of archives. After unpacking 651kB will be freed.
Do you want to continue? [Y/n/?] Y
[...]
```

Z mojego punktu widzenia, aptitude jest znacznie wygodniejszym narzędziem niż apt-get, na pewno do usuwania pakietów. Należy zwrócić uwagę, że aptitude pamięta powiązania między pakietami, które zostały zainstalowane przez ten program, dlatego używam go zarówno do instalowania, jak i do usuwania pakietów. Jeśli ktoś często instaluje i usuwa pakiety, a nie wykorzystuje opcji usuwania powiązań, w systemie gromadzą się liczne

zbędne pakiety, o których często nie pamiętamy. Informacje na temat zachowania systemu w porządku bez zbędnych pakietów można znaleźć w ostatnim podrozdziale tego rozdziału, „Utrzymanie porządku w systemie”.

Dodawanie i usuwanie pakietów z użyciem programu Synaptic

Menedżer pakietów Synaptic jest aplikacją graficzną środowiska GNOME służącą do instalowania pakietów oprogramowania dla Ubuntu, ich usuwania i zarządzania nimi. Wielokrotnie w tej książce zwracałem uwagę na to, że osobiście preferuję program Synaptic do realizacji większości funkcji tego typu. Synaptic to narzędzie, o którym użytkownicy innych dystrybucji Linuksa oraz innych systemów operacyjnych mogą jedynie marzyć.

Synaptic powstał w brazylijskiej firmie Conectiva, tworzącej własną dystrybucję Conectiva Linux. Firma została wchłonięta przez firmę Mandrakesoft, twórcę Mandrake Linux, która to z kolei zmieniła nazwę na Mandriva, a jej dystrybucja nazywa się obecnie Mandriva Linux dla zaznaczenia, że nie ma już wiele wspólnego ze starym Mandrake Linux. Synaptic został napisany z użyciem biblioteki *WINGS*, wykorzystywanej przede wszystkim przez menedżer okien Window Maker (<http://www.windowmaker.info>), kopiującej wygląd i zachowanie znane użytkownikom systemów operacyjnych NeXT. Obecne wersje programu Synaptic są tworzone z użyciem biblioteki graficznej GTK+ wykorzystywanej w GNOME. Co interesujące, Conectiva była dystrybucją opartą na pakietach RPM, a Synaptic jest obecnie kojarzony z formatem DEB i stosuje się go w dystrybucjach Ubuntu i Debian.

Program Synaptic jest domyślnie zainstalowany w Ubuntu. Uruchamia się go z menu systemowego *System/Administracja/Menedżer pakietów Synaptic*. Podobnie jak w większości narzędzi administracyjnych w Ubuntu, w celu uruchomienia programu Synaptic system poprosi użytkownika o podanie hasła. Na rysunku 20.11 przedstawiłem główne okno programu Synaptic.

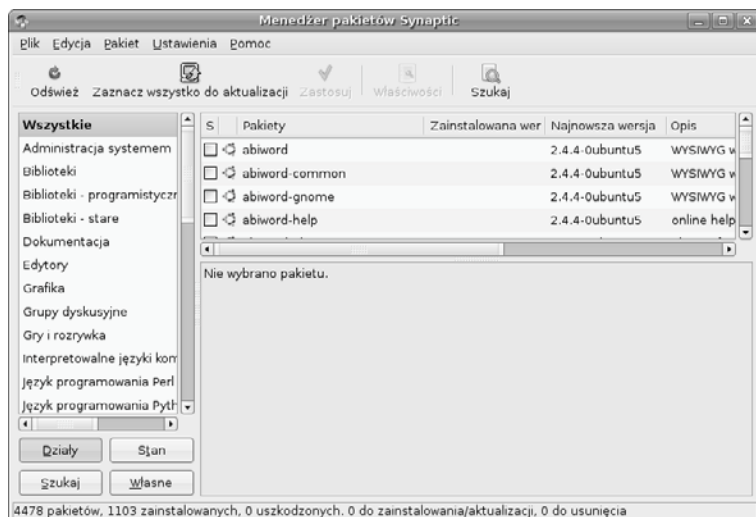
W lewym panelu okna przedstawionego na rysunku 20.11 można zobaczyć kategorie, w których zostało zorganizowane oprogramowanie w ramach repozytoriów uaktywnionych domyślnie podczas instalacji Ubuntu. Na rysunku 20.12 zostały uwzględnione dodatkowe repozytoria Ubuntu, co zwiększyło listę dostępnych kategorii.

Warto zwrócić uwagę, że kategorie oprogramowania zdefiniowane w lewym panelu okna programu Synaptic są nadal takie same, jedynie dopisano do nich nazwy dodatkowych elementów repozytorium. Chodziło o to, żeby można było łatwo zidentyfikować wymagania licencyjne poszczególnych pakietów, co zostało omówione w punkcie „Repozytoria i ich komponenty w systemie Ubuntu”.

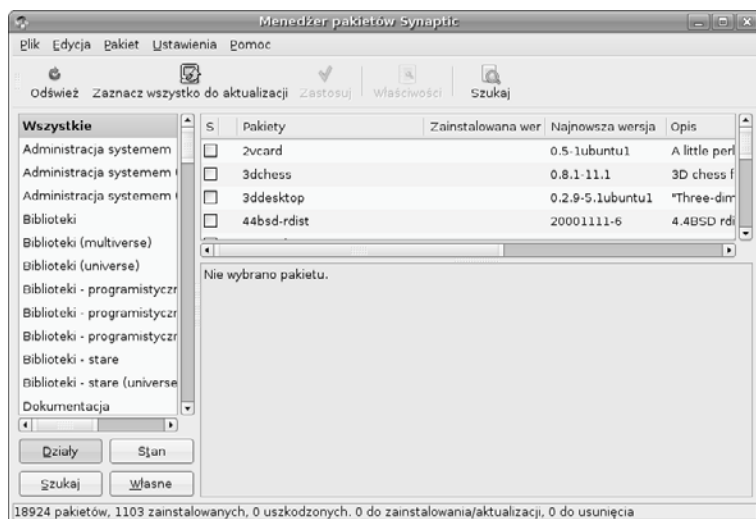
Główne okno programu Synaptic składa się z następujących elementów podstawowych:

- ♦ **lewy panel** zawiera listę dostępnych kategorii oprogramowania w oparciu o komponenty repozytorium zdefiniowane w pliku */etc/apt/sources.list*;
- ♦ **panel po prawej stronie u góry** zawiera oprogramowanie dostępne w kategorii zaznaczonej w lewym panelu; domyślnie w lewym panelu zaznaczona jest pozycja

Rysunek 20.11.
Okno pojawiające się
po uruchomieniu
programu Synaptic



Rysunek 20.12.
Większy wybór
kategorii
oprogramowania



Wszystkie, zatem w tym panelu można przeglądać wszystkie dostępne pakiety; wyszukiwanie w programie Synaptic tworzy nową, wirtualną kategorię i ukrywa pozostałe, z wyjątkiem kategorii *Wszystkie*, ale o tym szerzej wspomnę za chwilę;

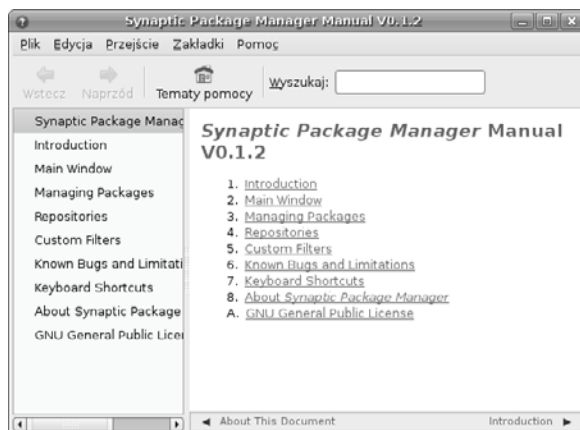
- ♦ **panel po prawej u dołu** przedstawia informacje o pakiecie zaznaczonym w liście pakietów w panelu ponad nim.

Dla programu Synaptic dostępna jest doskonała dokumentacja. Na rysunku 20.13 przedstawiam okno pomocy systemowej z otwartą dokumentacją programu Synaptic.

Dokumentacja systemowa programu Synaptic jest wyczerpująca, nie będę więc zanudzał czytelnika wszelkimi zawartymi tam informacjami. W kilku kolejnych punktach skupię się jedynie na zagadnieniach konfiguracji programu Synaptic oraz sposobach realizacji podstawowych zadań, takich jak wyszukiwanie i instalacja oraz usuwanie pakietów.

Rysunek 20.13.

Pomoc systemowa programu Synaptic

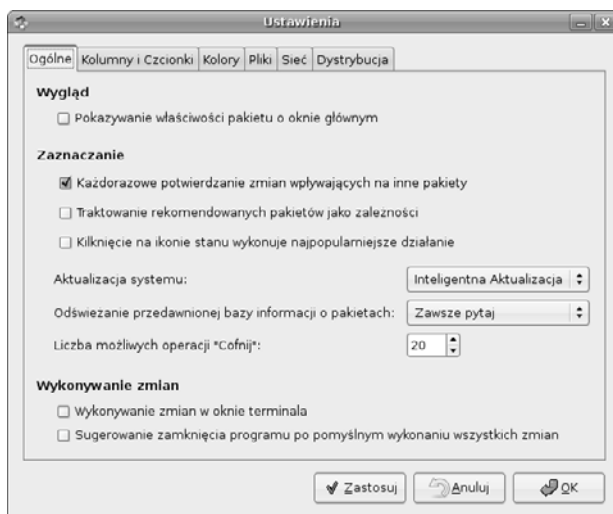


Konfiguracja ustawień programu Synaptic

Okno konfiguracyjne programu Synaptic pozwala ustawić domyślne zachowanie programu, zdefiniować jego repozytoria, ustalić sposób działania operacji wyszukiwania, sposób wyświetlania ikon w pasku narzędzi, jak również określić wartości wewnętrznych zmiennej programu. Większość tych ustawień można zdefiniować w oknie *Ustawienia*, które wywołuje się z menu *Ustawienia/Preferencje*. Okno to przedstawiam na rysunku 20.14.

Rysunek 20.14.

Okno preferencji programu Synaptic



Najczęściej modyfikowane opcje zostały zgromadzone na stronie *Ogólne*. Oto one.

- ♦ **Traktowanie rekomendowanych pakietów jako zależności:** zaznaczenie tej opcji spowoduje, że pakiety zalecane (ale niewymagane) do instalacji z pakietem będą instalowane razem z nim. Zależności tego typu są omówione szerzej w punkcie „Instalacja zalecanych pakietów z użyciem aptitude” — chodzi o to, że przy instalacji pakietu istnieją takie, których instalacja nie jest niezbędna, ale warto ją wziąć pod uwagę.

- ♦ **Aktualizacja systemu:** ta pozycja powinna być zawsze ustawiona na wartości *Inteligentna aktualizacja*. Spowoduje to, że Synaptic podejmie próbę identyfikacji pakietów, które muszą zostać zainstalowane, zaktualizowane lub usunięte w celu zaktualizowania każdego pakietu. Synaptic jest jednak znacznie bardziej konserwatywny, jeśli chodzi o tego typu aktualizacje, w porównaniu z programem `apt-get` — jak pamiętamy, odpowiednikiem tej opcji jest polecenie `apt-get dist-upgrade`, ale ono działa znacznie skuteczniej i w sposób bardziej radykalny. Mimo że jestem zagorzałym zwolennikiem programu Synaptic, co jakiś czas wywołuję polecenie `apt-get dist-upgrade`, które aktualizuje pakiety pominięte przez program Synaptic. Symptodem sytuacji, w której pakiety nie zostały zaktualizowane, jest to, że niektóre pakiety otrzymują stan „wstrzymany” i pojawia się okienko informujące o konieczności uruchomienia polecenia `apt-get dist-upgrade`.
- ♦ **Wykonywanie zmian w oknie terminala:** ustawienie tej opcji spowoduje, że Synaptic będzie w osobnym oknie terminala wyświetlał tekstowe komunikaty stanu procesu instalacji lub aktualizacji pakietu. Jak wspominałem przy okazji omawiania programów `apt-get` i `aptitude`, w procesie instalacji pakietów czasem pojawia się konieczność uzupełnienia informacji systemu o dane niezbędne do jego instalacji. Również Synaptic w takim przypadku wyświetla okno konfiguracyjne podobne do prezentowanego na rysunku 20.7. Gdy pojawi się takie okno, należy odpowiedzieć na pytania zadawane przez program konfigurujący pakiet, po czym instalacja będzie kontynuowana. Synaptic wyświetla to okno w standardowo zwiniętym panelu terminala, zatem w czasie instalacji zawsze warto monitorować to okno. Jeśli zatem Synaptic zatrzyma się na etapie instalacji pakietu, warto rozwinąć panel terminala i sprawdzić, czy program nie oczekuje na wprowadzenie dodatkowych informacji.

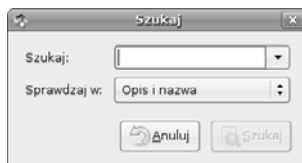
Po zmodyfikowaniu ustawień w dowolnej zakładce okna preferencji programu Synaptic należy kliknąć przycisk *Zastosuj*, co spowoduje ich zapisanie; wciśnięcie przycisku *OK* spowoduje zamknięcie okna *Ustawienia* i powrót do głównego okna programu Synaptic.

Wyszukiwanie oprogramowania w programie Synaptic

Jedną z największych zalet programu Synaptic jest funkcja wyszukiwania niewymagająca korzystania z wiersza poleceń. Aby znaleźć pakiet związany z określonym słowem kluczowym (lub większą ich liczbą), należy kliknąć przycisk *Szukaj* w pasku narzędzi. Pojawi się okno wyszukiwania, prezentowane na rysunku 20.15.

Rysunek 20.15.

Okno wyszukiwania programu Synaptic



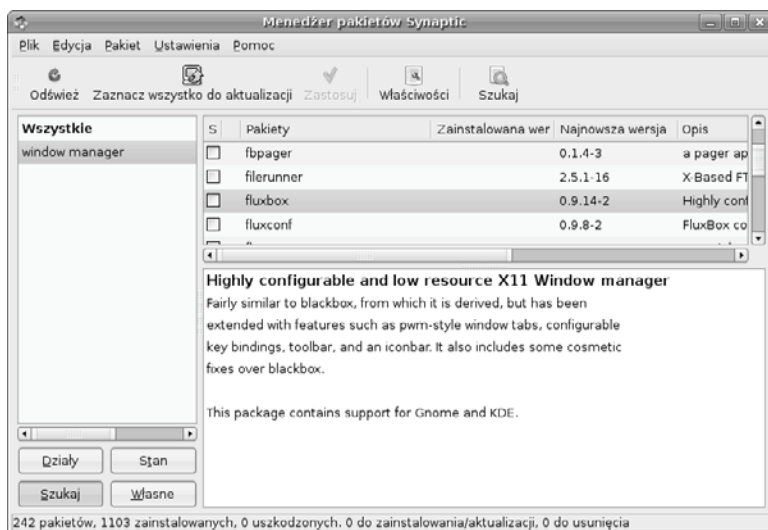
Większość przykładów instalacji oprogramowania w dalszej części tego rozdziału opiera się na funkcji wyszukiwania programu Synaptic. Aby znaleźć pakiety pasujące do zadanych słów kluczowych, należy określić, gdzie Synaptic ma ich szukać. Zwykle można pozostawić domyślną wartość listy rozwijanej *Sprawdź w:* (*Opis i nazwa*), co spowo-

duje wyszukiwanie słów kluczowych zarówno w opisie pakietu, jak i w jego nazwie. Od czasu do czasu trzeba jednak zmienić tę wartość w celu uszczegółowienia wyszukiwania. Oto inne dostępne opcje.

- ♦ **Nazwa:** wyszukiwanie słów kluczowych obejmie tylko nazwy pakietów.
- ♦ **Opiekun:** pozwala szukać pakiety, którymi opiekuje się określona osoba.
- ♦ **Wersja:** wyszukuje określone wersje, co może być użyteczne przy szukaniu pakietów o wersjach przystosowanych do wersji innych pakietów, takich jak np. pakiety związane ze środowiskiem GNOME.
- ♦ **Zależności:** pozwala zidentyfikować pakiety zależne od wyszukiwanego pakietu.
- ♦ **Dostarczane pakiety:** pozwala zidentyfikować pakiety na podstawie innych pakietów, które powodują ich instalację.

Aby szukać pakiety w programie Synaptic, należy wpisać jedno lub większą liczbę słów kluczowych w oknie wyszukiwania i kliknąć *Szukaj* (albo nacisnąć *Enter*). Synaptic w lewym panelu utworzy nową kategorię o nazwie w postaci wyszukiwanych słów kluczowych i ukryje wszystkie kategorie, z wyjątkiem *Wszystkie*. Po zakończeniu wyszukiwania ta kategoria jest zaznaczana automatycznie, a znalezione pakiety są wyświetlane w panelu po prawej stronie u góry. Na rysunku 20.16 przedstawiam wynik wyszukiwania słów kluczowych *window manager*.

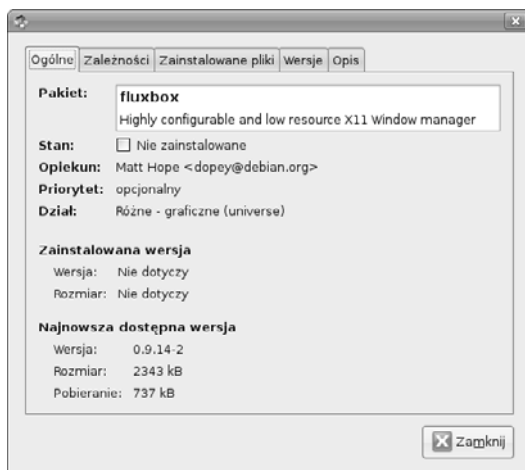
Rysunek 20.16.
Wyniki wyszukiwania
w programie Synaptic



Aby zaznaczyć pakiet do instalacji, należy szukać go w prawym górnym panelu. Po kliknięciu jego nazwy w prawym dolnym panelu pojawią się informacje o nim, można również kliknąć ikonę *Właściwości* (albo z menu kontekstowego wybrać opcję *Właściwości*), co spowoduje wyświetlenie okna z dalszymi informacjami. Na rysunku 20.17 przedstawiam okno właściwości pakietu *fluxbox*, zawierającego menedżer okien Fluxbox, czyli lekkie środowisko okienkowe — lubię je stosować na laptopach.

Rysunek 20.17.

Okno Właściwości
pakietu w programie
Synaptic



Strona *Ogólne* wyświetlana po otwarciu okna właściwości pakietu zawiera ogólne informacje o pakiecie. Na stronie *Zależności* można znaleźć użyteczne informacje o pakietach wymaganych, zalecanych, sugerowanych oraz będących w konflikcie z analizowanym pakietem. Strona *Zainstalowane pliki* jest również bardzo użyteczna, ale, niestety, zawiera informacje wyłącznie na temat zainstalowanych pakietów (informacje na temat plików wchodzących w skład niezainstalowanych pakietów można uzyskać, wykorzystując aplikację `apt-file`, omówioną w punkcie „Określenie pakietu, w którym znajduje się brakujący plik”).

Po zapoznaniu się z właściwościami wybranego pakietu można zamknąć okno, klikając przycisk *Zamknij*, co spowoduje powrót do głównego okna programu Synaptic.

Instalacja pakietów za pomocą programu Synaptic

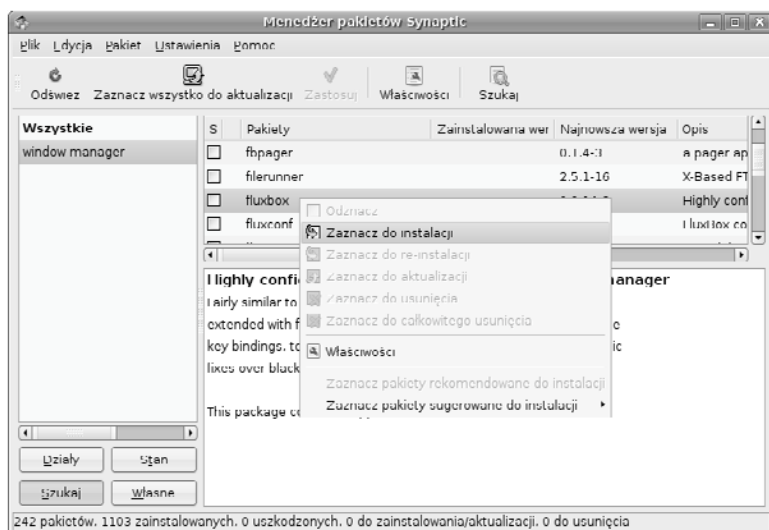
Po zlokalizowaniu pakietu, który chcemy zainstalować, klikamy jego nazwę prawym przyciskiem myszy, co powoduje wyświetlenie menu kontekstowego. W tym menu klikamy funkcję *Zaznacz do instalacji*, co przedstawiam na rysunku 20.18.

Gdy instalacja wybranego pakietu wymaga zainstalowania innych pakietów, pojawia się okno informujące o tym fakcie, przedstawione na rysunku 20.19.

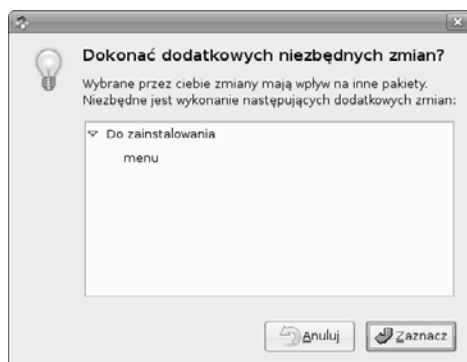
To okno zawiera listę wszystkich pakietów, których instalacja jest wymagana do instalacji wybranego pakietu. Jeśli została oznaczona opcja *Traktowanie rekomendowanych pakietów jako zależności*, znajdą się tu również pakiety zalecane, ale nie wymagane do instalacji. Należy kliknąć przycisk *Zaznacz*, co potwierdzi chęć instalacji, i okno informacyjne zostanie zamknięte.

Następnie należy ponownie kliknąć prawym przyciskiem myszy nazwę pakietu, aby sprawdzić, czy istnieją pakiety sugerowane do instalacji z wybranym. Można je znaleźć w podmenu, co przedstawiam na rysunku 20.20.

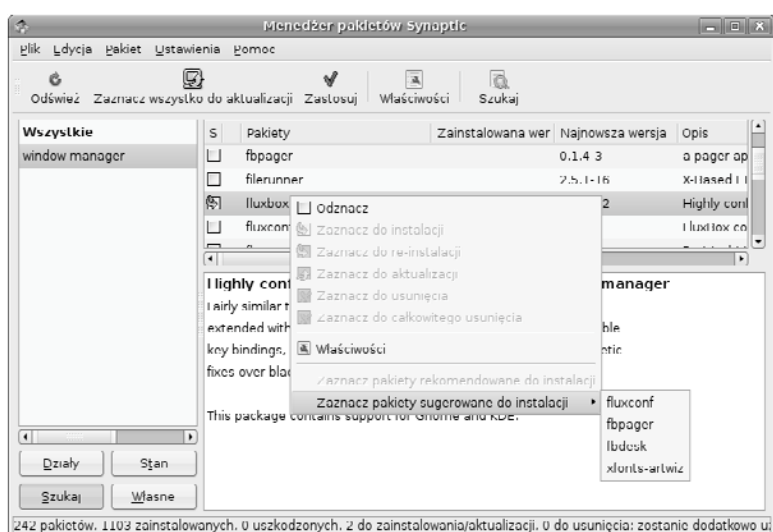
Rysunek 20.18.
Zaznaczanie pakietu
do instalacji
w programie Synaptic



Rysunek 20.19.
Identyfikacja innych
pakietów niezbędnych
do instalacji



Rysunek 20.20.
Zaznaczanie
sugerowanych
pakietów do instalacji
w programie Synaptic

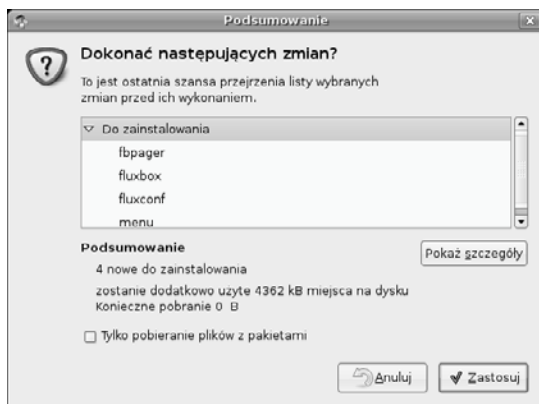


To podmenu identyfikuje pakiety, które z reguły należy instalować z wybranym pakietem, ale nie są niezbędne do pracy. Pakiety te można również zaznaczyć do instalacji, korzystając z tego menu (jeden po drugim).

Po zaznaczeniu do instalacji sugerowanych pakietów jesteśmy gotowi do instalacji, a zatem należy kliknąć ikonę *Zastosuj* w pasku narzędzi, co spowoduje rozpoczęcie procesu instalacji. Pojawi się okno potwierdzenia przedstawione na rysunku 20.21, zawierające podsumowanie planowanych działań instalacyjnych.

Rysunek 20.21.

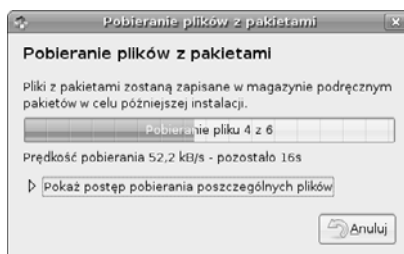
*Podsumowanie
procedury instalacji
pakietów w programie
Synaptic*



Aby kontynuować, należy kliknąć przycisk *Zastosuj*. W tym momencie Synaptic rozpocznie pobieranie pakietów z repozytoriów, wyświetlając okno z paskiem postępu, co widać na rysunku 20.22.

Rysunek 20.22.

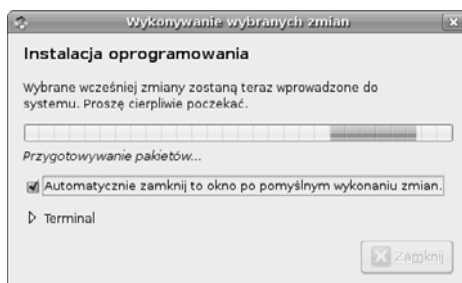
*Status pobierania
plików podczas
instalacji pakietów
w programie Synaptic*



Po zakończeniu pobierania pakietów Synaptic rozpoczyna ich instalację i konfigurację. Jeśli w konfiguracji nie została zaznaczona opcja *Wykonywanie zmian w oknie terminala*, Synaptic wyświetli proste okno stanu instalacji, takie jak na rysunku 20.23.

Rysunek 20.23.

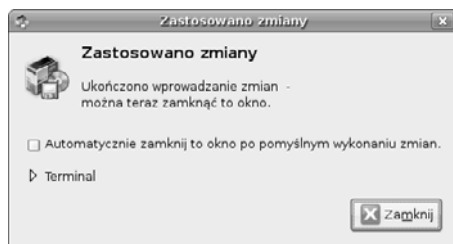
*Proste okno stanu
instalacji w programie
Synaptic*



W tym oknie można otworzyć panel z niewielkim terminalem zawierającym status procesu instalacji i konfiguracji. W tym celu należy kliknąć strzałkę u dołu okna, przy etykiecie *Terminal*. Po zakończeniu instalacji pakietów pojawi się okno potwierdzenia instalacji, przedstawione na rysunku 20.24.

Rysunek 20.24.

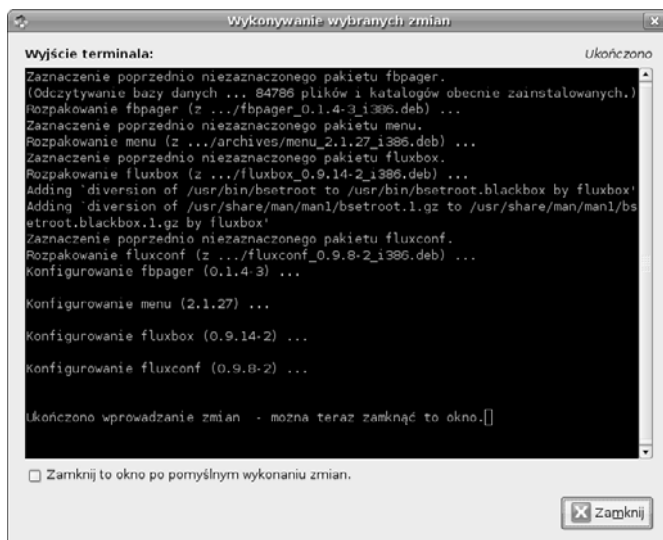
Okno potwierdzenia zakończenia instalacji w programie Synaptic



Jeśli w konfiguracji programu Synaptic została zaznaczona opcja *Wykonywanie zmian w oknie terminala*, pojawi się osobne okno terminala zawierające komunikaty z procesu instalacji i konfiguracji pakietów, co przedstawiam na rysunku 20.25. W takim przypadku nie zobaczymy okien przedstawionych na rysunkach 20.23 i 20.24.

Rysunek 20.25.

Okno terminala przedstawiające status instalacji i konfiguracji



Niezależnie od tego, jakie okno jest wyświetlane po zakończeniu procesu instalacji, w celu jego zamknięcia i powrotu do głównego okna programu Synaptic należy kliknąć przycisk *Zamknij*.

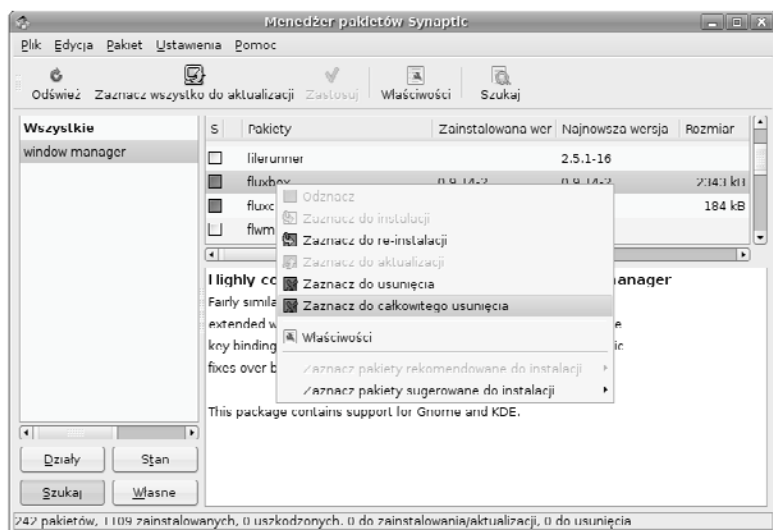
Po zakończeniu instalowania pakietów można zamknąć program Synaptic, wybierając z menu polecenie *Plik/Zakończ* lub stosując kombinację klawiszy *Ctrl+q*.

Usuwanie pakietów za pomocą programu Synaptic

Usuwanie pakietów w programie Synaptic realizuje się w sposób zbliżony do instalowania. Należy odszukać pakiet przewidziany do usunięcia, wybierając odpowiednią kategorię w lewym panelu, lub wykorzystać funkcję wyszukiwania. Po odnalezieniu odpowiedniej

nazwy pakietu należy kliknąć ją prawym przyciskiem myszy, co spowoduje wyświetlenie menu podręcznego (patrz rysunek 20.26).

Rysunek 20.26.
Zaznaczanie pakietu
do kompletnego
usunięcia
w programie Synaptic



Opcja *Zaznacz do całkowitego usunięcia* spowoduje jego usunięcie wraz z plikami konfiguracyjnymi, być może zmodyfikowanymi przez użytkownika, natomiast opcja *Zaznacz do usunięcia* usuwa pakiety, ale pozostawia w systemie ich pliki konfiguracyjne.



Niestety, Synaptic nie potrafi automatycznie usunąć pakietów wymaganych przez usuwany pakiet lub sugerowanych do instalacji, na co użytkownik zdecydował się przy instalowaniu pakietu. Aby je usunąć, należy zaznaczyć je indywidualnie. Można również nie przejmować się tym, że pozostały w systemie, i zastosować się do porad opisanych w punkcie „Utrzymanie porządku w systemie”.

Po wyborze pakietów do usunięcia należy kliknąć ikonę *Zastosuj*. Przed rozpoczęciem procedury usuwania zainstalowanych pakietów zostanie wyświetlone okno potwierdzenia, przedstawione na rysunku 20.27.

Rysunek 20.27.
Okno potwierdzenia
usunięcia pakietu
w programie Synaptic



Aby kontynuować usuwanie, należy kliknąć przycisk *Zastosuj*. Podczas tej procedury Synaptic wyświetla okno stanu zbliżone do wyświetlanego podczas instalacji. Po zakończeniu procedury usuwania pakietów należy kliknąć przycisk *Zamknij*, co spowoduje powrót do głównego okna programu.

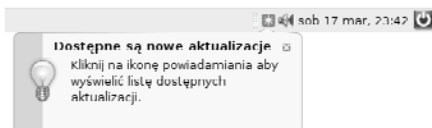
Po zakończeniu usuwania pakietów można zamknąć program Synaptic, wybierając z menu polecenie *Plik/Zakończ* lub stosując kombinację klawiszy *Ctrl+q*.

Wykorzystanie menedżera aktualizacji

Menedżer aktualizacji systemu Ubuntu sprawdza repozytoria zdefiniowane w pliku */etc/apt/sources.list* i informuje użytkownika o dostępności nowych wersji pakietów. Do tego celu Menedżer aktualizacji wykorzystuje obszar powiadamiania znajdujący się standardowo w górnym panelu. Gdy w tym obszarze pojawi się pomarańczowa ikona słoneczka, to znak, że dostępne są nowe wersje pakietów. Na rysunku 20.28 przedstawiam tę ikonę wraz z towarzyszącym jej dymkiem informacyjnym.

Rysunek 20.28.

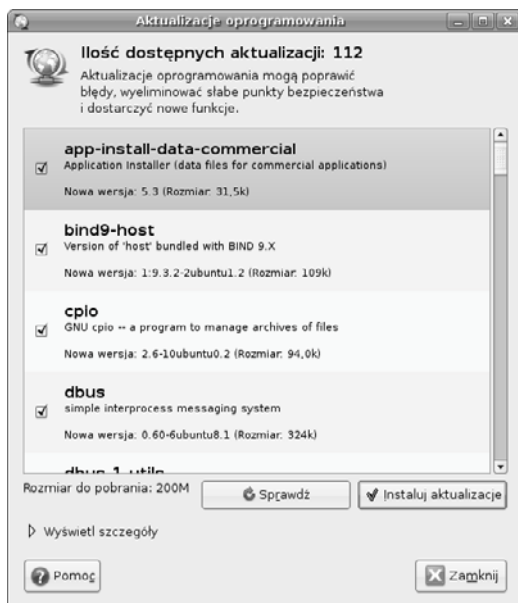
Ikona i okienko informujące o dostępności aktualizacji



Po kliknięciu tej ikony zostanie uruchomione główne okno Menedżera aktualizacji, przedstawione na rysunku 20.29.

Rysunek 20.29.

Menedżer aktualizacji



Używanie Menedżera aktualizacji jest proste — wystarczy kliknąć przycisk *Instaluj aktualizacje*. Menedżer aktualizacji automatycznie pobiera, instaluje i konfiguruje wszystkie

dostępne, nowsze wersje pakietów, wyświetlając w tym czasie okno postępu bardzo zbliżone do wykorzystywanych w programie Synaptic. Po zakończeniu procesu aktualizacji należy kliknąć przycisk *Zamknij*.



Podobnie jak w programie Synaptic, należy uważać, czy podczas instalacji nowszych wersji pakietów nie pojawiają się pytania związane z ich konfiguracją. Gdy instalacja aktualizacji zatrzyma się lub zajmuje dłuższy czas, należy kliknąć przycisk *Terminal* w oknie postępu i sprawdzić, czy program nie czeka na wprowadzenie informacji przez użytkownika.

Konwersja pakietów z innych formatów

Choć repozytoria Ubuntu zawierają większość oprogramowania dostępnego na rynku systemów Linux, może się zdarzyć, że użytkownik będzie musiał poradzić sobie z instalacją pakietów dostępnych w innych formatach niż stosowany w Ubuntu. Ta sytuacja dotyczy głównie pakietów oprogramowania komercyjnego, dostarczanego najczęściej w formatach RPM lub LSB. Na szczęście, istnieje rozwiązanie tego typu problemów — jest nim pakiet *alien* (www.kitenet.net/~joey/code/alien.html), instalujący skrypt napisany w Perlu i dokonujący konwersji między formatami pakietów. Co najważniejsze, *alien* potrafi przekształcać większość formatów pakietów na format DEB wykorzystywany w dystrybucjach Ubuntu i Debian. W zależności od formatu oryginalnego pakietu, *alien* może wymagać zainstalowania dodatkowych pakietów.



Program *alien* należy uruchamiać z uprawnieniami użytkownika root (najlepiej z użyciem polecenia *sudo*), ponieważ jest to niezbędne do odpowiedniego ustawiania uprawnień do plików wchodzących w skład pakietu wynikowego.

Program *alien* potrafi dokonywać konwersji pomiędzy następującymi formatami pakietów:

- ♦ **DEB**: przy konwersji pakietów do formatu DEB niezbędne są narzędzia udostępniane w pakietach *debhelper*, *dpkg*, *dpkg-dev*, *gcc* i *make*; pakiety *gcc*, *make* i *dpkg-dev* są niezbędne do samodzielnego kompilowania pakietów ze źródeł i są zależnościami pakietu *build-essential*, co zostało omówione w rozdziale 18. Instalacja pakietu *alien* powoduje zainstalowanie zależności *debhelper*. Jeśli użytkownik wykorzystuje inną dystrybucję, niż Ubuntu, za pomocą programu *alien* można również przekształcić pakiety DEB na inne formaty pakietów.
- ♦ **LSB**: format pakietów wykorzystywany do dystrybucji pakietów i całych dystrybucji zgodnych ze specyfikacją Linux Standard Base (<http://www.freestandards.org/en/LSB>). Specyfikacja LSB narzuca standard w postaci formatu RPM, format LSB stanowi w zasadzie nadzbiór formatu RPM, co powoduje, że do obsługi formatu LSB niezbędne są narzędzia do zarządzania pakietami RPM. Odpowiedni do tego celu pakiet *rpm* jest instalowany jako zależność pakietu *alien*. Jeśli ktoś planuje dokonać konwersji pakietu LSB zawierającego zależności od innego pakietu, program *alien* prawidłowo dokona przekształcenia zależności, ale dodatkowo zapisze zależność od pakietu *lsb*, dostępnego w standardowych repozytoriach Ubuntu. Należy również zainstalować pakiet *lsb-rpm*, który nie jest wymagany do instalacji pakietu *alien*, ale niezbędny do konwersji z formatu LSB.

- ♦ **PKG**: format pakietów wykorzystywany w systemach Solaris. Konwersja z formatu PKG na inne formaty wymaga instalacji pakietów do obsługi tego formatu: *pkginfo* i *pkgtrans*. Można również przekształcać pakiety PKG na RPM w samym systemie Solaris: w tym celu należy zainstalować w Solarisie narzędzie *rpm* dostępne na dysku CD z cyklu Sun Free Software. Takie przekształcenia prawdopodobnie mają sens wyłącznie w przypadku pakietów zawierających skrypty, kod źródłowy lub pliki poleceń, ponieważ binaria systemu Solaris nie mogą być uruchamiane w systemie Linux.
- ♦ **RPM**: format pakietów wykorzystywany przez wiele dystrybucji, takich jak Red Hat, Fedora Core, Mandrake, Yellow Dog itp. Aby przekształcić pakiety z formatu RPM, należy w Ubuntu zainstalować pakiet z oprogramowaniem RPM Package Manager, znanym wcześniej pod nazwą Red Hat Package Manager (pakiet *rpm* jest instalowany automatycznie z pakietem *alien*, ponieważ należy do jego zależności wymaganych).
- ♦ **SLP**: format pakietów wykorzystywany w nieistniejącej już dystrybucji Stampede Linux (zaprzeszono jej rozwoju w roku 2002). Nie sądzę, żeby obsługa tego formatu była nadal aktywnie rozwijana, ale skoro dystrybucja Stampede Linux przestała się rozwijać, nie powinny następować już zmiany w formacie jej pakietów.
- ♦ **TGZ**: format dystrybucji pakietów wykorzystywany w większości projektów open source oraz w dystrybucji Slackware Linux. Plik w formacie TGZ to archiwum *tar* skompresowane programem *gzip*. Pliki zapisane w formacie TGZ to skompresowane programem *gzip* archiwa *tar* zawierające dodatkowo metadane o pakiecie.



Więcej informacji na temat formatów pakietów dla Linuksa oraz o ich cechach szczególnych można znaleźć na stronie <http://kitenet.net/~joey/pkg-comp>.

Wykorzystanie programu *alien* do przekształcania jednego formatu pakietów w inny to proste zadanie. Aplikacja *alien* obsługuje opcje poszczególnych formatów docelowych, takich jak `--to-deb`, `--to-lsb`, `--to-rpm`, czy `--to-tgz`. Poniższy przykład służy do przekształcenia pakietu Adobe Acrobat Reader z formatu RPM do formatu DEB:

```
$ sudo alien --scripts --to-deb AdobeReader_enu-7.0.0-2.i386.rpm
Password:
adobereader-enu_7.0.0-3_i386.deb generated
```

W zasadzie nic nadzwyczajnego. Opcja `--scripts` informuje program *alien* o tym, że powinien przekształcić również odpowiednie skrypty przedinstalacyjne i poinstalacyjne, lecz poprawność działania przekształconych skryptów nie jest gwarantowana; instalując taki pakiet, należy zwrócić uwagę na to, czy nie występują niepokojące komunikaty o błędach itp. Opcja `--to-deb` informuje program *alien* o tym, że przekształcenie ma zostać wykonane do formatu DEB. Opcja `-v` spowoduje wyświetlenie bardziej szczegółowych komunikatów o procesie przekształcania, ale rzadko bywają one przydatne zwykłemu użytkownikom.

Przed zainstalowaniem lub przekształceniem pakietów należy zweryfikować ich integralność przy pomocy programu *dpkg*, np.:

```
$ dpkg --info adobereader-enu_7.0.0-3_i386.deb
new debian package, version 2.0.
size 37932742 bytes: control archive= 7436 bytes.
    484 bytes,   11 lines    control
   20061 bytes,  207 lines   md5sums
   1340 bytes,   60 lines *  postinst      #!/bin/sh
   980 bytes,   45 lines *  postrm       #!/bin/sh
    211 bytes,    7 lines    shlibs
Package: adobereader-enu
Version: 7.0.0-3
Section: alien
Priority: extra
Architecture: i386
Installed-Size: 92316
Maintainer: root <root@vmdesktop>
Description: Adobe Reader for Linux. An application that reads a PDF document.
 Adobe Reader 7.0.0 can read documents in PDF format. Adobe Reader also allows you to
 search within PDF files, search for PDF files on the internet and participate in
 collaborative document reviews.
.
(Converted from a rpm package by alien version 8.64.)
```

Te informacje dość dokładnie odpowiadają informacjom zapisanym w oryginalnym pakiecie RPM (wywołanie z innej dystrybucji Linuksa):

```
$ rpm -q --info -v -p AdobeReader_enu-7.0.0-2.i386.rpm
Name       : AdobeReader_enu           Relocations: /usr/local/Adobe/Acrobat7.0
Version    : 7.0.0                   Vendor: Adobe Systems, Incorporated
Release    : 2                       Build Date: Mon 28 Mar 2005 06:34:42 AM
EST
Install date: (not installed)         Build Host: acrolinux2.corp.adobe.com
Group      : Applications/Publishing  Source RPM: AdobeReader_enu-7.0.0-
2.src.rpm
Size       : 97613446                 License: Commercial
Signature  : (none)
Packager   : Adobe Systems, Incorporated
URL        : www.adobe.com
Summary    : Adobe Reader for Linux. An application that reads a PDF document.
Description:
Adobe Reader 7.0.0 can read documents in PDF format. Adobe Reader also allows you to
search within PDF files, search for PDF files on the internet and participate in
collaborative document reviews.
Distribution: (none)
```

Bardziej szczegółowe informacje o pakiecie *alien* można znaleźć w jego podręczniku systemowym (man *alien*).



Nie należy używać programu *alien* do instalacji pakietów zawierających niskopoziomowe narzędzia lub biblioteki systemowe. Może się bowiem okazać, że mimo poprawnego ich przekształcenia i instalacji bez błędów, nie będą działać prawidłowo, a co więcej, mogą spowodować poważne problemy, takie jak kłopoty z uruchomieniem systemu.

Utrzymanie porządku w systemie

Jeśli ktoś lubi eksperymenty z pakietami (tak jak ja), często instaluje je i używa przez chwilę, chyba że okażą się użyteczne i wtedy są wykorzystywane systematycznie. Jeśli jest użytkownikiem dbającym o porządek, po zaniechaniu użytkowania pakietu może nawet zdecydować się na jego usunięcie. Niestety, taka operacja nie usunie pakietów zainstalowanych do spełnienia zależności wymaganych, zalecanych lub sugerowanych (chyba że do instalacji i usuwania został użyty program `aptitude`).

Co jakiś czas warto zatem uruchamiać program skanujący wszystkie zainstalowane pakiety niebędące zależnościami innych pakietów, niebędące również częścią żadnego systemowego procesu instalacyjnego. Dobrymi przykładami narzędzi tego typu są `deborphan` i `debfooster`. Żaden z nich nie jest domyślnie zainstalowany w Ubuntu, ale można je zainstalować przy użyciu dowolnego narzędzia do zarządzania pakietami.

Podstawowa różnica między tymi pakietami polega na tym, że `deborphan` po prostu tworzy listę pakietów niewykorzystywanych przez inne pakiety i niestanowiących zależności innych pakietów, natomiast `debfooster` wykorzystuje własną bazę danych zawierającą informację o niezbędnych pakietach, stara się podejmować inteligentne decyzje dotyczące tego, które pakiety są niezbędne, i zadaje mnóstwo pytań przy pierwszym uruchomieniu. Z tych wszystkich powodów preferuję program `deborphan`, ale sugeruję, aby każdy zapoznał się z działaniem obu tych narzędzi i dostosowaniem moich sugestii do własnych potrzeb.

Jak wspomniałem, program `deborphan` po prostu wypisuje pakiety niewykorzystane i niezwiązane z innymi. Wynik działania tego programu można przekierować przez inne polecenie albo zapisać w pliku i wykorzystać go do skonstruowania skryptu powłoki usuwającego pakiety z systemu. Inne rozwiązanie, wyglądające na nieco skomplikowane, polega na wykorzystaniu potoku za pomocą programu `xargs`, przekształcającego listę w argumenty polecenia `apt-get remove`:

```
$ sudo deborphan | xargs apt-get remove --purge -y
```

Przed zastosowaniem wywołania tego typu należy zapoznać się z wynikiem działania polecenia `deborphan`, aby upewnić się, że nie uwzględni pakietów, które z różnych powodów chcemy zachować w systemie.

Systematyczne uruchamianie programów `deborphan` lub `debfooster` pozwoli zredukować ilość zbędnych pakietów zaśmiecających system. Utrzymanie porządku w systemie pozwala również oszczędzić miejsce na dyskach, które można wykorzystać do „prawdziwej pracy”, takiej jak przechowywanie muzyki czy kolekcji artystycznych zdjęć.

Podsumowanie

Zarządzanie pakietami to cudowna rzecz, pozwalająca systemowi na identyfikację zainstalowanych pakietów oraz, dzięki pomocy doskonałego formatu pakietów DEB, na identyfikację i spełnienie wymaganych zależności gwarantujących że zainstalowane zostanie kompletne, w pełni funkcjonalne oprogramowanie. W tym rozdziale omówiłem cztery

najważniejsze, dostępne w Ubuntu narzędzia do zarządzania pakietami oprogramowania (dpkg, apt-get, aptitude i Synaptic), przedstawiłem podstawy ich wykorzystania do wyszukiwania pakietów, określania potrzeb dotyczących instalacji pakietów oraz omówiłem wiele interesujących i użytecznych trików ułatwiających zarządzanie pakietami w systemie.

W rozdziale 21. kontynuuję temat zarządzania systemem, omawiam bowiem metody zarządzania kontami użytkowników i grupami w systemie Ubuntu oraz znaczenie tego zagadnienia. W tym rozdziale omówione zostaną również zaawansowane zagadnienia dotyczące zabezpieczania plików i katalogów, takie jak listy kontroli dostępu (ang. *Access Control Lists*, ACL) wybiegające poza tradycyjne mechanizmy uprawnień użytkowników i grup, zapewniające precyzyjną kontrolę dostępu do danych (a przede wszystkim niewymagające posiadania do tego celu uprawnień administratora).

Rozdział 21.

Zarządzanie użytkownikami, grupami i zaawansowane uprawnienia

W tym rozdziale:

- ♦ Tworzenie użytkowników i grup
- ♦ Standardowe uwierzytelnianie w Linuksie
- ♦ Konfiguracja polecenia sudo
- ♦ Wykorzystanie ACL-i z wiersza poleceń
- ♦ Graficzna edycja ACL-i

Choć wielu użytkowników instaluje i wykorzystuje Ubuntu na komputerach typowo osobistych, Linux jest zaprojektowany i zaimplementowany jako system komputerowy do stosowania przez wielu użytkowników. Nawet wtedy, gdy komputer z zainstalowanym Ubuntu ma tylko jednego użytkownika, w systemie istnieje większa liczba kont użytkowników, którzy są właścicielami procesów systemowych działających z ich uprawnieniami. Proces logowania się w systemie wygląda prosto, ale w rzeczywistości wykorzystuje wiele operacji sprawdzania użytkowników i grup, podobnie jak wiele innych operacji systemowych wymagających specjalnych uprawnień.

W tym rozdziale omawiam zagadnienia zarządzania użytkownikami i grupami w systemach Ubuntu. Rozdział 4. zawierał wprowadzenia do stosowanych w Ubuntu i ogólnie w systemach Linux koncepcji zarządzania użytkownikami i grupami, uprawnieniami do plików i wykonywania uprzywilejowanych operacji. Oto krótkie przypomnienie.

- ♦ Punkt rozdziału 4. pod tytułem „Podstawowe koncepcje: użytkownicy i grupy” zawierał przegląd zagadnień związanych z użytkownikami i grupami w Ubuntu i innych systemach uniksowych, wyjaśniał zawartość i sposób organizacji plików systemowych zawierających informacje o użytkownikach i grupach.

- ♦ Punkt rozdziału 4. zatytułowany „Uprawnienia do plików i katalogów w Linuksie” zawierał wprowadzenie do zagadnień związanych z uprawnieniami użytkowników do plików i katalogów, wyjaśniając pokrótce, co oznaczają nieczytelne na pierwszy rzut oka oznaczenia uprawnień w szczegółowych listingach zawartości katalogów.
- ♦ Punkt rozdziału 4. zatytułowany „Wykonywanie czynności wymagających specjalnych uprawnień” wyjaśniał, w jaki sposób Linux obsługuje wykonanie uprzywilejowanych operacji, co stanowi pewne odstępstwo od tradycyjnych technik stosowanych w innych dystrybucjach Linuksa i systemach uniksowych.

W niniejszym rozdziale omawiam tę tematykę bardziej szczegółowo. Informacje podane tutaj obejmują szeroki zakres — od praktycznego omówienia dotyczącego modyfikowania użytkowników i grup z użyciem narzędzi graficznych, po szczegółową dyskusję na temat tego, w jaki sposób użytkownicy i grupy wykorzystują system Ubuntu oraz zapisane w nim pliki i katalogi. W pierwszym podrozdziale omawiam narzędzia graficzne i obsługiwane z wiersza polecenia, służące do tworzenia i zarządzania użytkownikami i grupami. W następnym podrozdziale wyjaśniam zasady funkcjonowania mechanizmów uwierzytelniających w trakcie logowania się użytkownika w systemie. Ten podrozdział powinien zainteresować administratorów większych sieci korporacyjnych lub akademickich, w których często konieczne jest wprowadzanie modyfikacji w systemie uwierzytelniania. W tym rozdziale szczegółowo omawiam także sposób obsługi uprzywilejowanych poleceń w Ubuntu, który różni się nieco od sposobów realizacji tych zadań w innych dystrybucjach Linuksa i innych systemach uniksowych. W ostatnim podrozdziale opisuję listy ACL, które pozwalają na większy poziom szczegółowości kontroli dostępu do plików. Czytelnik znajdzie tu zrozumiały słownik poleceń obsługi ACL-i oraz objaśnienia komunikatów tych poleceń.

Tworzenie i zarządzanie grupami i użytkownikami

Użytkownicy i grupy to przede wszystkim sposób nadawania różnych uprawnień osobom i analogicznie, grupom osób. Do uprawnień zalicza się możliwość odczytu, zapisu i dostępu do poszczególnych plików i katalogów, możliwość wykorzystania urządzeń systemowych w określony sposób oraz możliwość odczytywania i zapisywania zasobów specjalnego przeznaczenia, takich jak systemy plików, specjalne obszary w pamięci czy zasoby jądra.

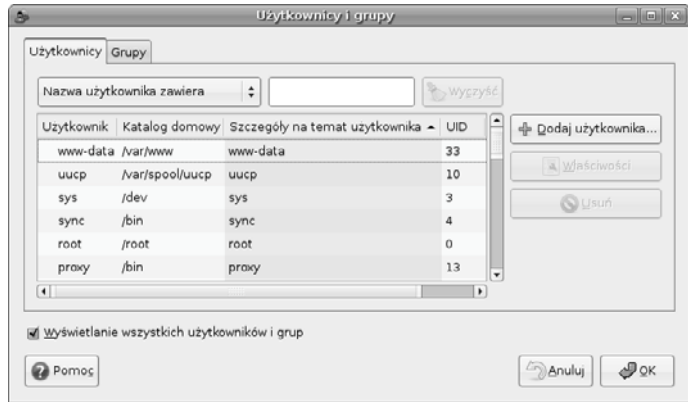
Ubuntu zawiera wygodne graficzne narzędzie do zarządzania użytkownikami i grupami. Narzędzie to można uruchomić z menu *System/Administracja/Użytkownicy i grupy*. Po podaniu hasła użytkownika pojawi się okno przedstawione na rysunku 21.1.

Na rysunku warto zauważyć, że zaznaczyłem opcję *Wyświetlanie wszystkich użytkowników i grup*, co powoduje, że w oknie są widoczne wszystkie konta istniejące w systemie, nie tylko konta zwykłych użytkowników.

Okno tekstowe i lista rozwijana powyżej listy użytkowników pozwalają wyszukiwać ich w oparciu o różne kryteria, takie jak fragment nazwy użytkownika (*Nazwa użytkownika zawiera*), fragment nazwy grupy (*Nazwa grupy zawiera*), ID użytkownika (*Identyfikatorem użytkownika jest*) oraz ID domyślnej grupy (*Identyfikatorem grupy jest*).

Rysunek 21.1.

Narzędzie
do zarządzania
użytkownikami
i grupami



W Ubuntu dostępne są — oczywiście — narzędzia obsługiwane z wiersza poleceń, służące do tworzenia, modyfikacji i usuwania użytkowników oraz grup. Więcej informacji na temat tych narzędzi można znaleźć w podręczniku systemowym programów `adduser`, `useradd`, `userdel`, `usermod`, `groupadd`, `groupdel` i `groupmod`. Do tworzenia pojedynczych kont użytkowników z domyślnymi ustawieniami preferuję jednak narzędzie graficzne *Użytkownicy i grupy*. Narzędzia obsługiwane z wiersza poleceń są dość użyteczne w skryptach, są też doskonale udokumentowane i nie sądzę, żeby jakkolwiek sens miało kopiowanie tej dokumentacji w książce. Podzielę się jednak jedną informacją przydatną przy zarządzaniu grupami za pomocą narzędzi obsługiwanych z wiersza poleceń: grupę domyślną można określić dla użytkownika już na etapie jego tworzenia (polecenie `adduser nazwa_użytkownika nazwa_grupy`). Przy dodawaniu użytkownika do innych grup za pomocą polecenia `usermod` należy pamiętać o zastosowaniu opcji `-a` (`usermod -a -G nazwa_grupy nazwa_użytkownika`). Pomińnięcie opcji `-a` spowoduje, że pełna definicja grup zostanie określona na podstawie danego wywołania, tzn. użytkownik będzie członkiem tylko tej jednej grupy podanej w wywołaniu, a więc zostanie usunięty z grup, do których mógł należeć wcześniej. Opcja `-a` (ang. *add*) działa jak operator dodawania do zbioru grup. Pamiętam, jak kłopotliwa jest sytuacja, gdy zapomni się o tej opcji i należy ponownie wpisać użytkownika do wszystkich grup, z których został omyłkowo wyrzucony.

Informacji o narzędziu *Użytkownicy i grupy* wykraczających poza opis zawarty w tym rozdziale należy szukać w pomocy systemowej, przedstawionej na rysunku 21.2. Uruchomienie tej pomocy odbywa się przez kliknięcie przycisku *Pomoc* w dowolnym oknie narzędzia *Użytkownicy i grupy*.

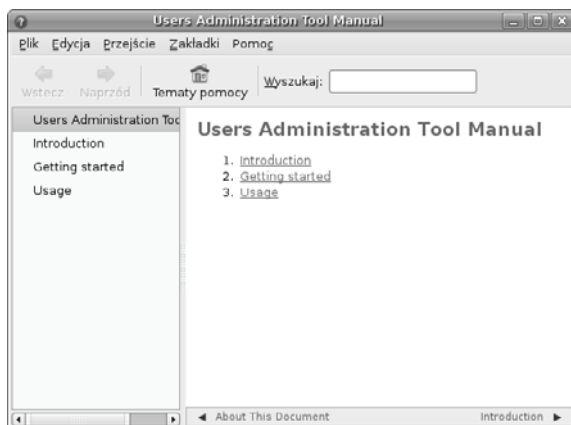
Tworzenie nowych użytkowników

Aby dodać nowego użytkownika z wykorzystaniem narzędzia *Użytkownicy i grupy*, należy kliknąć przycisk *Dodaj użytkownika*. Spowoduje to wyświetlenie okna *Edytor konta użytkownika*, przedstawionego na rysunku 21.3.

Tylko pola *Nazwa użytkownika* i *Hasło* (oraz jego potwierdzenie) są wymagane, ale warto uzupełnić również inne dane na temat konta, gdy bowiem system zawiera większą liczbę kont użytkowników, może pojawić się pytanie: „A czyje to konto?”. Po wypełnieniu pól na tej stronie można przejść na stronę *Zaawansowane*, która została przedstawiona na rysunku 21.4.

Rysunek 21.2.

Podręcznik
systemowy narzędzia
Użytkownicy i grupy

**Rysunek 21.3.**

Edytor konta
użytkownika

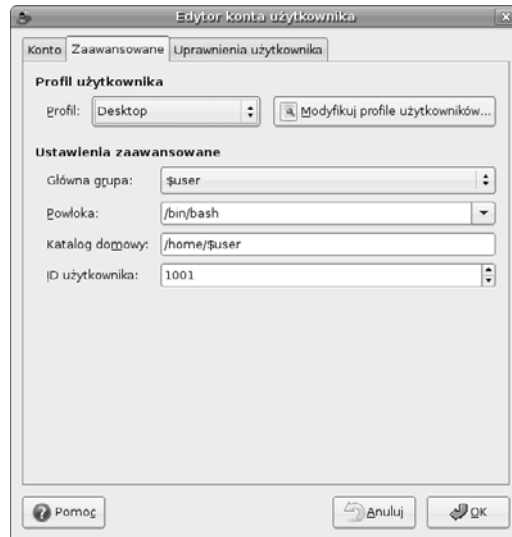


W tym oknie można zdefiniować podstawowe parametry tworzonego konta, ale najciekawszą opcją jest możliwość określenia profilu użytkownika. Profil to definicja uprawnień wykorzystywana przy tworzeniu użytkownika. Standardowe profile użytkowników dostępne w Ubuntu to *Default*, *Desktop* i *Administrator*. Każdy z tych profili określa takie elementy jak dostęp do urządzeń, podstawowe uprawnienia użytkownika oraz grupy, do których będzie należał tworzony użytkownik. Właściwości każdego z profili można modyfikować po kliknięciu przycisku *Edytuj profile użytkowników...*, co spowoduje wyświetlenie okna przedstawionego na rysunku 21.5.

W tym oknie trzeba kliknąć nazwę potrzebnego profilu i później przycisk *Właściwości*. Zostanie wyświetlone okno właściwości profilu, przedstawione na rysunku 21.6.

Na rysunku 21.6 zostały przedstawione uprawnienia użytkowników w ramach profilu *Desktop*. Jak widać, wszystkie uprawnienia są udostępnione, z wyjątkiem uruchamiania zadań administracyjnych. W tym oknie można dodatkowo ustawić bardziej ogólne ustawienia, dostępne na stronie *Ogólne*, przedstawionej na rysunku 21.7.

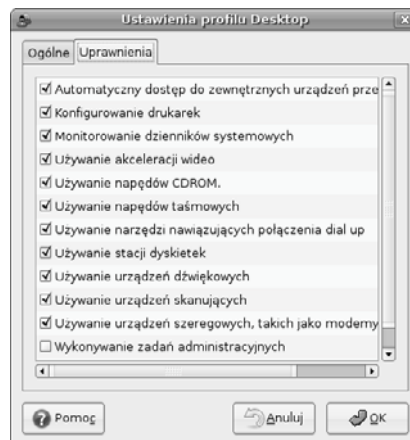
Rysunek 21.4.
*Strona Zaawansowane
okna Edytor konta
użytkownika*



Rysunek 21.5.
*Okno Profile
użytkowników*

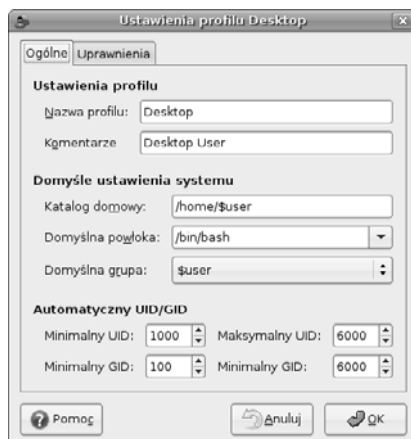


Rysunek 21.6.
*Uprawnienia
związane z profilem
użytkowników*



Jak widać na rysunku 21.7, ogólne parametry profilu są bardzo zbliżone do parametrów nowo tworzonego konta (na stronie *Zaawansowane* w oknie *Edytor konta użytkownika*, przedstawionej na rysunku 21.4). Właśnie tu zdefiniowana jest ścieżka do katalogu

Rysunek 21.7.
*Ogólne ustawienia
 profilu użytkowników*



użytkownika (*/home/* plus nazwa konta), domyślna grupa użytkownika (taka sama jak nazwa konta) oraz sposób generowania identyfikatora użytkownika (UID) i identyfikatora grupy (GID).

Możliwość definiowania ogólnych ustawień kont użytkowników jest wygodna w zastosowaniach domowych, gdzie najczęściej w zupełności wystarczają podstawowe profile zdefiniowane w Ubuntu. Jednak możliwość definiowania własnych profili może być bardzo użyteczna w środowiskach korporacyjnych lub akademickich, gdzie często konieczne jest stworzenie dużej liczby kont użytkowników o zbliżonych uprawnieniach.

Jeśli w profilach nie chcemy niczego zmieniać i wolimy wrócić do okna *Profile użytkowników*, wystarczy kliknąć przycisk *Anuluj*. To okno z kolei zamyka się przyciskiem *Zamknij*, co powoduje powrót do okna *Edytor konta użytkownika*.

Program do zarządzania kontami i grupami użytkowników pozwala również modyfikować uprawnienia indywidualnie na poziomie kont użytkowników. W tym celu należy przejść na stronę *Uprawnienia użytkowników* w oknie *Edytor konta użytkownika* (patrz rysunek 21.8).

Jak łatwo zauważyć, ustawienia na tej stronie są identyczne z uprawnieniami w ramach profilu użytkownika (patrz rysunek 21.6). Domyślne ustawienia na tej stronie odpowiadają dokładnie ustawieniom w ramach profilu wybranego dla użytkownika.

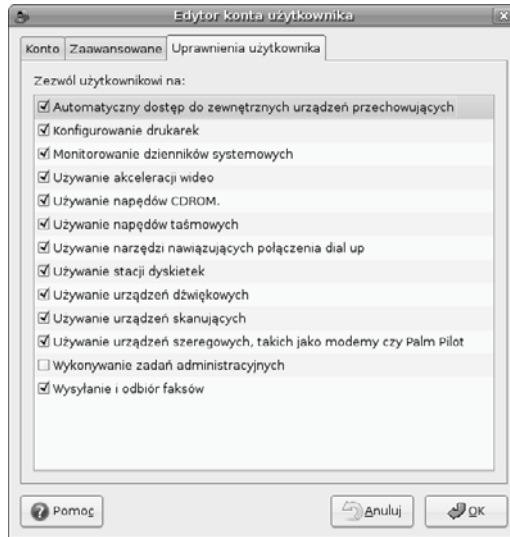
Po dostosowaniu parametrów konta użytkownika nadszedł czas na jego utworzenie. W tym celu należy kliknąć przycisk *OK* w oknie *Edytor konta użytkownika*. Okno to zostanie zamknięte, a system dokona odpowiednich wpisów w plikach systemowych (*/etc/passwd*, */etc/group* i */etc/shadow*). W głównym oknie programu *Użytkownicy i grupy* pojawi się nowe, utworzone przed chwilą konto użytkownika.

Zarządzanie istniejącymi użytkownikami

Program *Użytkownicy i grupy* nie tylko ułatwia tworzenie nowych kont użytkowników, pozwala też w prosty sposób modyfikować parametry istniejących kont, jak również je usuwać. Aby zmodyfikować lub usunąć istniejące konto, należy uruchomić program

Rysunek 21.8.

Ustawianie uprawnień
użytkownika na etapie
tworzenia konta



Użytkownicy i grupy (menu *System/Administracja/Użytkownicy i grupy*), co spowoduje wyświetlenie okna przedstawionego na rysunku 21.1.

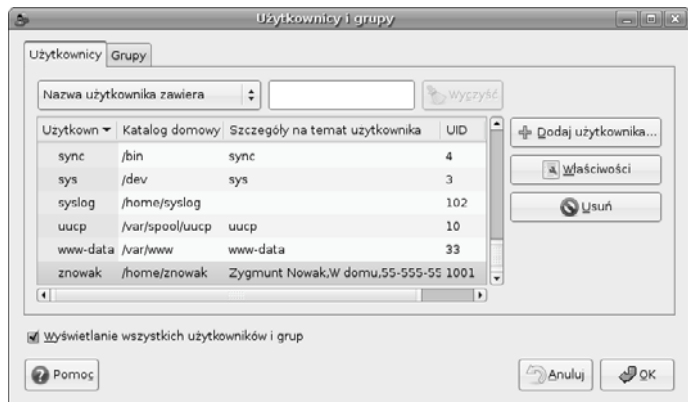


Najczęściej wykonywanym zadaniem administracyjnym na istniejącym koncie jest dodawanie go do grup lub usuwanie z nich. Zadanie to jest opisane w podrozdziale „Zarządzanie istniejącymi grupami”, w dalszej części rozdziału.

Aby zmodyfikować lub usunąć istniejące konto, należy zaznaczyć je na liście, co spowoduje uaktywnienie przycisków *Właściwości* i *Usuń* (patrz rysunek 21.9).

Rysunek 21.9.

Wybór konta
do usunięcia
lub modyfikacji



Aby usunąć konto, trzeba kliknąć przycisk *Usuń*. Program wyświetli okno z żądaniem potwierdzenia tej operacji oraz informacją, że to spowoduje jedynie usunięcie danych na temat konta z plików *passwd*, *group* i *shadow*, ale nie spowoduje usunięcia plików i katalogów związanych z użytkownikiem, takich jak zawartość jego katalogu domowego. Potwierdzenie usunięcia konta polega na kliknięciu przycisku *Usuń*. Aby wrócić do głównego okna narzędzia *Użytkownicy i grupy*, rezygnując z usunięcia konta, należy kliknąć przycisk *Anuluj*.

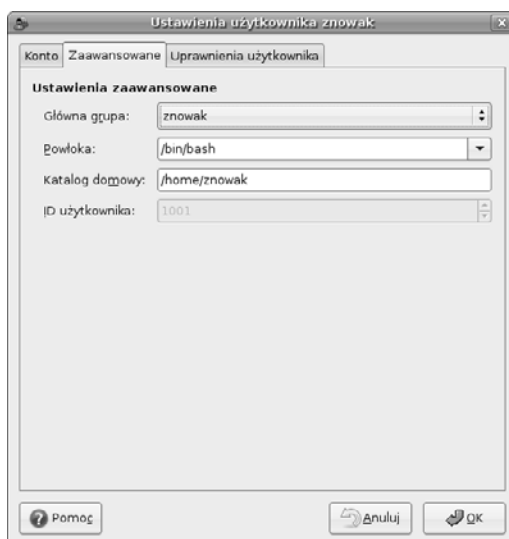
Aby zmodyfikować istniejące konto, trzeba w oknie *Użytkownicy i grupy* kliknąć przycisk *Właściwości*. To spowoduje wyświetlenie okna o zawartości zbliżonej do okna *Edytor konta użytkownika*, z kilkoma różnicami:

- ♦ tytuł okna to tym razem *Ustawienia konta nazwa_konta*;
- ♦ pola, których nie można modyfikować, takie jak nazwa konta czy UID, są nieaktywne (wyszarzone);
- ♦ nie można zmodyfikować profilu użytkownika, ponieważ służy on jedynie określeniu domyślnych uprawnień, czyli jest szablonem dla nowo tworzonego konta użytkownika.

Na rysunku 21.10 przedstawiam zawartość karty *Zaawansowane* edytora ustawień istniejącego konta użytkownika. Stanowi ona odpowiednik karty o tej samej nazwie w oknie *Edytor konta użytkownika* (patrz rysunek 21.4).

Rysunek 21.10.

*Modyfikacja
zaawansowanych
ustawień istniejącego
konta*



Aby zmodyfikować uprawnienia danego konta, należy przejść na kartę *Uprawnienia użytkownika*, zawierającą te same elementy, co jej odpowiednik z rysunku 21.8.

Po wprowadzeniu zmian w parametrach konta trzeba kliknąć *OK*, co spowoduje zamknięcie okna *Ustawienia użytkownika* i powrót do głównego okna narzędzia *Użytkownicy i grupy*.

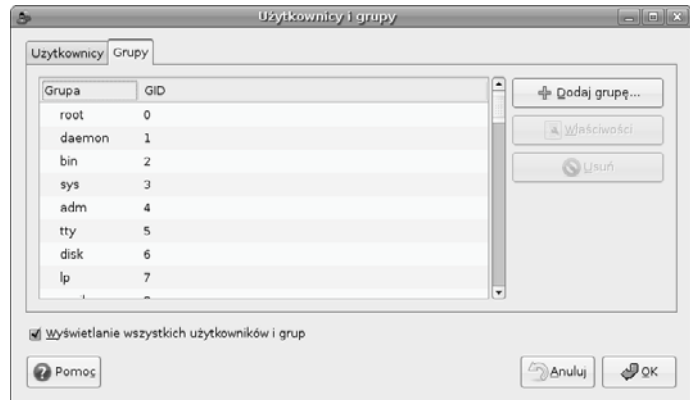
Tworzenie nowych grup

Nowe grupy systemowe są z reguły tworzone na etapie instalacji oprogramowania wymagającego zastosowania grupy nieistniejącej jeszcze w Ubuntu. Grupy systemowe to określenie stosowane do grup tworzonych na potrzeby określonego podsystemu lub aplikacji, np. grup wykorzystywanych przez podsystem obsługi drukarek, baz danych i różnego rodzaju sprzętu. W systemie Ubuntu Linux grupy systemowe używają identyfikatorów grupy mniejszych od 1000.

Użytkownik może tworzyć własne grupy np. w celu identyfikacji członków zespołu projektowego, do identyfikacji wybranych kategorii użytkowników lub po to, by ograniczyć zbiór użytkowników upoważnionych do uruchomienia wybranych programów. Narzędzie *Użytkownicy i grupy* w Ubuntu udostępnia graficzny interfejs upraszczający zadania tworzenia grup i zarządzania nimi (to drugie zadanie będzie omówione w dalszej części rozdziału). W oknie *Użytkownicy i grupy* funkcje zarządzania grupami są dostępne na karcie *Grupy*, co można zobaczyć na rysunku 21.11.

Rysunek 21.11.

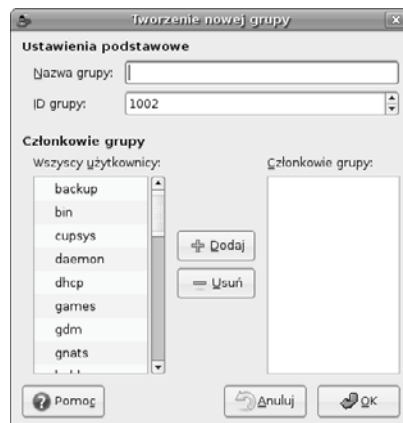
Karta *Grupy*
w głównym oknie
narzędzia
Użytkownicy i grupy



Aby utworzyć nową grupę, należy kliknąć przycisk *Dodaj grupę*. Zostanie wyświetlone okno przedstawione na rysunku 21.12.

Rysunek 21.12.

Okno *Tworzenie nowej grupy*



Nazwę tworzonej grupy wpisuje się w polu *Nazwa grupy*. Następnie można opcjonalnie zmodyfikować ID grupy, co może być przydatne podczas wykorzystania środowiska udostępniania plików NFS bez wykorzystania usługi NIS (szczegóły na temat NFS i NIS można znaleźć w rozdziale 16., w punkcie „Dostęp do katalogów NFS z systemów Linux” i w rozdziale 31., „Konfiguracja serwera NFS”).

W tym miejscu można również dodać użytkowników do tworzonej grupy. Aby to zrobić, należy kliknąć nazwy użytkowników w panelu *Wszyscy użytkownicy*. W systemie Ubuntu Linux identyfikatory grup systemowych są mniejsze od wartości 1000.

Po zakończeniu modyfikowania parametrów i definiowania członków grupy należy kliknąć przycisk *OK*, co spowoduje utworzenie grupy i powrót do karty *Grupy* narzędzia *Użytkownicy i grupy*. Z kolei to narzędzie można zamknąć, klikając przycisk *OK*. w jego głównym oknie.

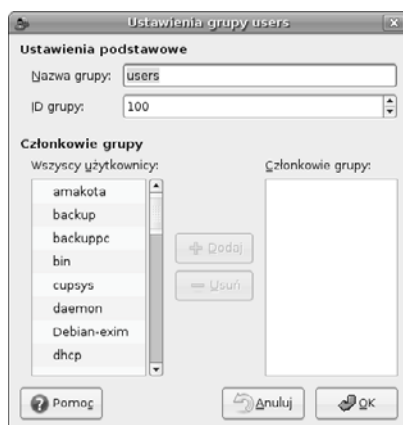
Zarządzanie istniejącymi grupami

Oprócz tworzenia nowych grup, narzędzie *Użytkownicy i grupy* pozwala na dodawanie użytkowników do grup i ich usuwanie.

W karcie *Grupy* głównego okna narzędzia *Użytkownicy i grupy* należy zaznaczyć nazwę grupy, do której chcemy dodać użytkowników lub z której chcemy ich usunąć. Po kliknięciu przycisku *Właściwości* pojawi się okno przedstawione na rysunku 21.13.

Rysunek 21.13.

Okno Ustawienia grupy



To okno jest w zasadzie identyczne z oknem towarzyszącym operacji tworzenia nowej grupy, włącznie z możliwością modyfikacji nazwy i identyfikatora grupy. Główną rolą tego okna jest umożliwienie modyfikacji danych członków wybranej grupy.



Należy unikać modyfikowania identyfikatora istniejącej grupy. Zmiana identyfikatora grupy w tym oknie powoduje zmianę wpisu w pliku */etc/group*, ale nie spowoduje zmiany tego identyfikatora we wszystkich kontekstach, w którym został użyty. Dlatego taka operacja może uniemożliwić dostęp do plików z ustawionym dostępem dla danej grupy, przez co użytkownicy, którzy mieli np. możliwość odczytu plików należących do ich grupy, nagle tę właściwość stracą.

Aby dodać użytkownika do istniejącej grupy, należy wybrać jego nazwę w lewym panelu (patrz rysunek 21.13), co spowoduje uaktywnienie przycisku *Dodaj*, używanego do dodawania użytkowników do grupy.

Aby usunąć użytkownika z grupy, trzeba zaznaczyć jego nazwę w prawym panelu. Po wybraniu użytkownika uaktywni się przycisk *Usuń*, służący do usuwania użytkowników z grupy.

Wszelkie zmiany członków, nazwy czy identyfikatora grupy w oknie przedstawionym na rysunku 21.13 zostaną zapisane w pliku */etc/group* (oraz potencjalnie w pliku */etc/passwd*)

dopiero po kliknięciu przycisku *OK*. Jeśli zamiast tego kliknie się przycisk *Anuluj*, wszelkie wprowadzone zmiany zostaną porzucone, a kontrola programu powróci do głównego okna narzędzia *Użytkownicy i grupy* (patrz rysunek 21.11). Po zakończeniu wprowadzania zmian w ustawieniach użytkowników i grup można zamknąć i to okno, klikając przycisk *OK*.

Podsystem PAM i proces uwierzytelniania użytkowników w Linuksie

W wielu dystrybucjach Linuksa oraz w systemach uniksowych aplikacje wymagają uwierzytelniania, jak również specjalnych uprawnień w celu uzyskania dostępu do urządzeń lub do uruchamiania procesów z podwyższonymi uprawnieniami albo z prawami określonych użytkowników i grup. We wczesnym okresie istnienia Linuksa funkcjonowała zasada pochodząca z tradycji uniksowej: wszystkie aplikacje miały na sztywno wkompiowaną informację o wykorzystywanych mechanizmach uwierzytelniania. Zmiana lub rozbudowa mechanizmu uwierzytelniania wymagały zmian w kodzie źródłowym i ponownej kompilacji aplikacji. Taki sposób realizacji tych funkcji jest niewygodny, mimo że kod źródłowy aplikacji jest powszechnie dostępny.

Aby poradzić sobie z tym problemem, firma Sun Microsystems opracowała koncepcję rozwiązania znanego dziś pod nazwą *Pluggable Authentication Modules* (PAM), elastycznego i dynamicznego mechanizmu uwierzytelniania aplikacji i usług. Ten model przyjął się doskonale i został wykorzystany praktycznie przez wszystkie systemy Linux i Unix, w tym również przez Ubuntu Linux. Aplikacje lub usługi skompilowane z obsługą biblioteki Linux-PAM wykorzystują konfigurację w postaci plików tekstowych, identyfikującą i opisującą ich wymagania dotyczące uwierzytelniania oraz biblioteki współdzielone niezbędne do jej implementacji. Model PAM pozwala modyfikować wymagania dotyczące uwierzytelniania istniejących aplikacji przez prostą modyfikację pliku konfiguracyjnego wykorzystywanego przez daną aplikację. Aplikacje wykorzystujące mechanizm PAM to aplikacje zgodne z PAM (*PAM-aware applications*), a elementy mechanizmu PAM są realizowane przez tzw. moduły PAM.

Moduły PAM obsługują pojedyncze elementy procesu uwierzytelniania wykorzystywanego przez aplikacje zgodne z PAM, dzięki czemu duża ilość kodu może być współużytkowana przez wiele różnych aplikacji. Przykładowo program do logowania użytkowników zgodny z PAM może wykorzystywać kilka różnych modułów PAM, sprawdzając szereg warunków, takich jak to, czy użytkownik loguje się jako root na terminalu zdefiniowanym jako zaufany, czy użytkownik ma prawo zalogować się w systemie w danym momencie oraz wiele innych, podobnych wymagań dotyczących uwierzytelniania. Moduły PAM to po prostu biblioteki dołączane dynamicznie, dzięki temu program *ssh* w wersji zgodnej z PAM może wykorzystać ten sam moduł realizujący funkcję kontroli, czy „w tej chwili użytkownicy mają prawo logować się w systemie”, co wersje programu *login* zgodne z PAM. Jednak po tym sprawdzeniu program *ssh* może wykorzystać inne moduły, specyficzne dla siebie, a niewykorzystywane przez program *login*.

Pliki definiujące kolejność ładowania modułów PAM dla poszczególnych aplikacji lub usług są zapisane w katalogu */etc/pam.d*. Same moduły natomiast (w postaci bibliotek

ładowanych dynamicznie) są zapisane w katalogu */lib/security*. W starszych implementacjach mechanizmu PAM moduły wykorzystywane przez poszczególne aplikacje były definiowane w jednym, centralnym pliku */etc/pam.conf*. Ten plik jest nadal wykorzystywany, ale wyłącznie w charakterze pliku domyślnego, używanego wtedy, gdy dla danego programu nie istnieje specyficzna konfiguracja PAM w katalogu */etc/pam.d*.

W repozytoriach Ubuntu dostępnych jest wiele różnych bibliotek PAM, niektóre z nich są domyślnie instalowane w systemie, inne są związane ze specyficznymi mechanizmami uwierzytelniania, takimi jak LDAP czy Kerberos. Wynik wyszukiwania w repozytoriach ciągu znaków *libpam* ujawnia, że obecnie dostępnych jest 31 pakietów związanych z mechanizmem PAM, co widać w poniższym listingu:

```
$ apt-file search libpam | sed -e 's:.*:;' | grep libpam | \
  grep -v dev | sort | uniq
libpam0g
libpam-ccreds
libpam-chroot
libpam-cracklib
libpam-doc
libpam-dotfile
libpam-encfs
libpam-foreground
libpam-heimdal
libpam-krb5
libpam-ldap
libpam-modules
libpam-mount
libpam-musclecard
libpam-mysql
libpam-ncp
libpam-openafs-kaserver
libpam-openafs-session
libpam-opie
libpam-p11
libpam-passwdqc
libpam-pgsql
libpam-pwdfilere
libpam-pwgen
libpam-radius-auth
libpam-runtime
libpam-smbpass
libpam-ssh
libpam-tmpdire
libpam-umask
libpam-unix2

$ apt-file search libpam | sed -e 's:.*:;' | grep libpam | \
  grep -v dev | sort | uniq | wc -l
31
```

Moduły PAM wykorzystywane przez specyficzne, niestandardowe mechanizmy uwierzytelniające są definiowane jako zależności pakietów dostarczających oprogramowanie z nich korzystające, dzięki czemu są instalowane automatycznie podczas instalacji tego oprogramowania.

Pliki konfiguracji mechanizmów PAM dla aplikacji i usług

Pliki konfiguracyjne mechanizmów PAM zapisane w katalogu */etc/pam.d* mają takie same nazwy jak wykorzystujące je aplikacje zgodne z mechanizmem PAM. W tych plikach zdefiniowana jest sekwencja ładowania modułów PAM podczas procesu uwierzytelniania i weryfikacji uprawnień dla danej aplikacji lub usługi. W plikach zapisanych w katalogu */etc/pam.d* można, oprócz reguł PAM, znaleźć puste wiersze (które są ignorowane) oraz komentarze, czyli ciągi znaków zapisane w wierszach rozpoczynających się od znaku #.

W pliku konfiguracyjnym zapisanym w katalogu */etc/pam.d* każdy wiersz niebędący komentarzem powinien być instrukcją `@include`, włączającą zawartość innego pliku konfiguracyjnego PAM, lub regułą PAM definiującą sposób wykorzystania modułu PAM w ramach procesu uwierzytelniania aplikacji lub usługi. Pliki włączane wykorzystuje się po prostu w celu uproszczenia plików konfiguracyjnych PAM, dzięki czemu powtarzalne reguły PAM mogą być pisane tylko raz i wykorzystywane wielokrotnie. Każda reguła PAM składa się z czterech pól odseparowanych białymi znakami, pierwsze trzy pola są obowiązkowe. Oto kolejność i znaczenie pól.

- ♦ `module-type` — typ modułu PAM definiowanego w regule. Typy modułów definiują sposób wykorzystania modułu w procesie uwierzytelniania. Prawidłowe wartości w tym polu są następujące:
 - ♦ `auth` — określa moduł wykorzystywany do weryfikacji tożsamości użytkownika oraz spełniania wymogów systemowych. Typowymi wymaganiami systemowymi są: możliwość uruchomienia usługi w danym momencie (np. kontrolowane istnieniem pliku */etc/nologin* w momencie próby zalogowania się w systemie), wykorzystanie do zalogowania w systemie określonego, uprawnionego urządzenia (urządzenia te są wymienione w pliku */etc/securetty*), posiadanego przez użytkownika w momencie uruchomienia usługi uprawnień konta root itp.
 - ♦ `account` — identyfikuje moduły weryfikujące prawo użytkownika do uwierzytelnienia w oparciu o wymogi systemowe, takie jak istnienie konta systemowego, możliwość zalogowania się w danym momencie (w oparciu o reguły definiowane na poziomie konta, nie ogólne reguły systemowe), możliwość dostępu do określonej aplikacji lub usługi itp.
 - ♦ `password` — identyfikuje moduły weryfikujące możliwość uwierzytelnienia użytkownika w systemie.
 - ♦ `session` — identyfikuje moduły związane z zadaniami, które muszą być wykonane przed aktywowaniem usługi lub aplikacji albo przed ich zakończeniem. Moduły tego typu z reguły wykonują funkcje systemowe, takie jak montowanie katalogów, zapisywanie informacji audytorskich lub przygotowanie do pracy zasobu systemowego.
- ♦ `control-flag` — znacznik kontrolny określający, jaki jest efekt działania danego modułu PAM na cały proces. Prawidłowe wartości tego pola są następujące:
 - ♦ `optional` — dany moduł PAM nie jest krytyczny w procesie uwierzytelnienia dostępu do usługi lub aplikacji, chyba że jest jedynym modułem zdefiniowanym w danym zbiorze reguł PAM. W takim przypadku sukces lub niepowodzenie działania tej reguły determinują sukces lub niepowodzenie całego procesu.

- ♦ **required** — sukces działania tej reguły jest niezbędny dla powodzenia całego zbioru reguł PAM. Należy jednak pamiętać, że niepowodzenie działania modułu PAM zawierającego reguły PAM ze znacznikiem kontrolnym **required** jest sygnalizowane dopiero po sprawdzeniu wszystkich reguł.
- ♦ **requisite** — sukces działania modułu jest uzależniony od sukcesu tej reguły, jednak porażka jest sygnalizowana natychmiast po wystąpieniu błędu w pierwszej z reguł typu **requisite**.
- ♦ **sufficient** — sukces tej reguły jest wystarczający do sukcesu całego zbioru reguł PAM. Jeśli żadna z poprzedzających tę regułę reguł PAM typu **required** nie zakończyła się niepowodzeniem, po sukcesie tej reguły nie zostaną wywołane żadne następne. Niepowodzenie reguły typu **sufficient** jest ignorowane, jeśli wszystkie następujące po niej reguły typu **required** zakończą się powodzeniem. Jeśli dowolna reguła typu **required** poprzedzająca tę regułę zakończy się niepowodzeniem, sukces tej reguły również jest ignorowany.
- ♦ **module-name** — nazwa modułu PAM związanego z tą regułą. Standardowo moduły PAM znajdują się w katalogu */lib/security*, ale w pliku definicji reguł PAM mogą być umieszczone definicje modułów znajdujących się w innych katalogach. W tym celu należy zastosować pełną ścieżkę do pliku modułu PAM.
- ♦ **arguments** — opcjonalne argumenty specyficzne dla modułu.

W kolejnym punkcie omówię przykład struktury i organizacji pliku konfiguracyjnego PAM wykorzystywanego do obsługi procesu logowania w systemie Ubuntu.

Przykład: konfiguracja PAM dla procesu logowania w systemie

Aby zrozumieć zasadę działania konfiguracji PAM, warto przyjrzeć się przykładowi. Plik konfiguracyjny PAM wykorzystywany w procesie logowania w systemie jest zapisany na ścieżce */etc/pam.d/login*. Po usunięciu pustych wierszy i komentarzy otrzymamy następujący zbiór reguł:

```
1: auth      requisite pam_securetty.so
2: auth      requisite pam_nologin.so
3: session   required pam_env.so readenv=1
4: @include  common-auth
5: auth      optional pam_group.so
6: @include  common-account
7: @include  common-session
8: session   required pam_limits.so
9: session   optional pam_lastlog.so
10: session  optional pam_motd.so
11: session  optional pam_mail.so standard
12: @include  common-password
```



Rzeczywisty plik */etc/pam.d/login* nie zawiera numeracji wierszy. Numery te dodałem w celu ułatwienia omawiania poszczególnych reguł.

Omówię teraz każdy wiersz tej sekwencji.

1. Wywołanie modułu `pam_securetty.so` w celu zweryfikowania, czy użytkownik próbuje zalogować się z bezpiecznego terminala, który jest zdefiniowany w pliku `/etc/securetty`. To sprawdzenie musi zakończyć się powodzeniem, ale nie jest wystarczające do zalogowania się użytkownika w systemie.
2. Wywołanie modułu `pam_nologin.so` w celu sprawdzenia, czy w danej chwili użytkownicy mogą logować się w systemie. Polega na sprawdzeniu, czy istnieje plik `/etc/nologin`. Jeśli tak, oznacza to, że system znajduje się w trybie blokady administratora (np. na potrzeby działań serwisowych) i możliwość zalogowania się w systemie jest tymczasowo zablokowana. To sprawdzenie musi zakończyć się sukcesem (tzn. plik nie może istnieć), który nie jest wystarczający do zalogowania się użytkownika.
3. Wywołanie modułu `pam_env.so` ładującego listę zmiennych środowiska zapisanych w pliku `/etc/security/pam.conf` (który w Ubuntu zawiera wyłącznie wiersze komentarza, czyli w praktyce nic nie robi). Ten moduł PAM musi zakończyć pracę sukcesem (tzn. plik musi dać się odczytać), ale nie jest to wystarczające do zalogowania użytkownika w systemie.
4. Włączenie zawartości pliku konfiguracyjnego `/etc/pam.d/common-auth`, który zawiera wspólny zestaw reguł PAM wykorzystywanych w procedurach uwierzytelniania. W dystrybucji Ubuntu 6.06 standardowo plik ten zawiera jeden wiersz, definiujący wykorzystanie klasycznych uniksowych plików `/etc/passwd` i `/etc/shadow`:

```
auth        required    pam_unix.so nullok_secure
```

Ta reguła wywołuje moduł `pam_unix.so`, który weryfikuje istnienie odpowiedniej pozycji w pliku `/etc/shadow`. Argument `nullok_secure` informuje, że w tym systemie puste hasła są dopuszczalne pod warunkiem, że są wykorzystywane z poziomu terminali wymienionych w pliku `/etc/securetty`. Ta reguła musi zakończyć się sukcesem, ale nie jest wystarczająca do tego, by użytkownik został zalogowany w systemie.

W systemach Ubuntu wykorzystujących inne mechanizmy uwierzytelniające, takie jak LDAP lub Kerberos, w tym pliku znajdują się definicje PAM odpowiednie dla nich.

5. Wywołanie modułu `pam_group.so`, który weryfikuje przynależność użytkownika do jednej lub większej liczby grup, oraz to, czy identyfikator liczbowy takiej grupy znajduje się w pliku `/etc/group`. Ten moduł jest wykonywany, ale jest opcjonalny: jego sukces lub niepowodzenie nie ma wpływu na powodzenie całego zbioru reguł, chyba że ta reguła byłaby jedyną (co w tym przypadku nie jest prawdą), z tego powodu nie jest wystarczająca do uwierzytelnienia użytkownika.
6. Włączenie zawartości pliku `/etc/pam.d/common-account`, w którym znajduje się wspólny zestaw reguł PAM służących do weryfikowania prawidłowości konta w systemie. W Ubuntu 6.06 wykorzystującym standardowe bazy użytkowników `/etc/passwd` i `/etc/shadow` plik ten zawiera następującą regułę PAM:

```
account     required    pam_unix.so
```

Ta reguła wywołuje moduł `pam_unix.so` w celu wykonania standardowego linuksowego mechanizmu uwierzytelniania w celu sprawdzenia, czy konto użytkownika nie jest oznaczone w pliku `/etc/shadow` jako nieaktywne (*expired*). Ta reguła PAM musi zakończyć się sukcesem, ale nie jest wystarczająca do zalogowania użytkownika.

W systemach Ubuntu wykorzystujących inne mechanizmy uwierzytelniające, takie jak LDAP lub Kerberos, w tym pliku znajdują się definicje PAM odpowiednie dla nich.

7. Włączenie zawartości pliku `/etc/pam.d/common-session`, zawierającego wspólny zestaw reguł PAM obsługujących interaktywne i nieinteraktywne sesje logowania. W Ubuntu 6.06 wykorzystującym standardowe bazy użytkowników `/etc/passwd` i `/etc/shadow` w tym pliku znajdują się następujące dwie reguły PAM:

```
session    required    pam_unix.so
session    optional    pam_foreground.so
```

Pierwsza reguła wywołuje moduł `pam_unix.so`, który w standardowym dzienniku systemowym (z użyciem podsystemu `syslog`) tworzy wpis z informacją o rozpoczęciu sesji. Ta reguła musi zakończyć się powodzeniem, ale nie jest wystarczająca do zalogowania użytkownika w systemie. Druga reguła wywołuje moduł `pam_foreground.so`, który tworzy plik blokady `/var/run/console/user: Numer-ty` na potrzeby logującego się użytkownika. Ten plik identyfikuje użytkownika jako właściciela określonej konsoli, co pozwala aplikacjom, takim jak GNOME Volume manager, określić, czy użytkownik jest zalogowany na konsoli. Ten moduł PAM jest wykonywany jako opcjonalny: jego sukces lub niepowodzenie nie mają wpływu na powodzenie logowania, chyba że ta reguła była jedyną (co nie jest prawdą w tym przypadku), a poprzez to nie jest wystarczający do zalogowania użytkownika w systemie.

8. Wywołanie reguły `pam_limits.so` definiującej ograniczenia użytkowników na podstawie zawartości pliku `/etc/security/limits.conf` (który standardowo w Ubuntu zawiera jedynie wiersze komentarza, czyli w praktyce nic nie robi). Ten moduł PAM musi zakończyć się sukcesem (tzn. musi wczytać plik konfiguracyjny), ale nie jest wystarczający do zalogowania użytkownika.
9. Wywołanie modułu `pam_lastlog.so`, który dopisuje informację o użytkowniku w pliku `/var/log/lastlog`: jeden wpis otwierający sesję, drugi z informacją o jej zakończeniu. Ta reguła PAM jest wykonywana, ale jako opcjonalna: jej sukces lub niepowodzenie nie mają wpływu na proces logowania, chyba że byłaby jedyną regułą w zbiorze (co w tym przypadku nie jest prawdą), a zatem nie jest regułą wystarczającą do zalogowania użytkownika w systemie.
10. Wywołuje moduł `pam_motd.so`, który wyświetla zawartość pliku `/etc/motd.conf` na konsoli logowania. Ten moduł PAM jest opcjonalny: sukces lub niepowodzenie reguły w nim zawartej nie mają wpływu na proces logowania, chyba że byłaby jedyną regułą w zbiorze (co w tym przypadku nie jest prawdą). Nie jest to zatem reguła wystarczająca do zalogowania użytkownika w systemie.
11. Wywołuje moduł `pam_mail.so`, sprawdzający zawartość katalogu poczty systemowej w poszukiwaniu nieprzeczytanej poczty i wypisujący na konsoli logowania stosowną informację. Ten moduł PAM jest opcjonalny: sukces lub niepowodzenie reguły

w nim zawartej nie mają wpływu na proces logowania, chyba że byłaby jedyną regułą w zbiorze (co w tym przypadku nie jest prawdą). Zatem nie jest regułą wystarczającą do zalogowania użytkownika w systemie.

12. Włącza zawartość pliku `/etc/pam.d/common-password`, w którym zdefiniowane są podstawowe reguły PAM dotyczące kontroli hasła. W Ubuntu 6.06, wykorzystującym standardowe bazy użytkowników `/etc/passwd` i `/etc/shadow`, w tym pliku znajduje się następująca reguła PAM:

```
password    required    pam_unix.so nullok obscure min=4 max=8 md5
```

Ta reguła wywołuje moduł `pam_unix.so` sprawdzający, czy w pliku `/etc/passwd` znajduje się odpowiedni wpis dla danego użytkownika, oraz weryfikujący prawidłowość hasła w oparciu o wpis w pliku `/etc/shadow`. Argument `nullok` oznacza, że w danym systemie dopuszczalne jest stosowanie pustych haseł. Argument `obscure` oznacza, że przy zmianie hasła zastosowanie będą miały pewne mechanizmy sprawdzające, takie jak weryfikacja, czy nowe hasło nie jest palindromem albo wersją starego hasła uzyskaną w wyniku rotacji znaków, że hasło nie różni się od poprzedniego jedynie pod względem wielkości liter, natomiast argumenty `min=` i `max=` określają minimalny i maksymalny rozmiar hasła. Weryfikowana jest również sytuacja „podobieństwa” nowego hasła do starego. Nie zagłębiałem do kodu, aby dowiedzieć się, jakie kryteria są stosowane przy określaniu „stopnia podobieństwa”, ale znane mi zasady jego określania wydają się dość zagmatwane. Ta reguła PAM musi zakończyć się sukcesem, ale nie jest wystarczająca do zalogowania użytkownika w systemie.



W pliku `/etc/pam.d/common-password` zdefiniowane są również inne reguły sprawdzające siłę hasła, ale są zablokowane znakiem komentarza. Warto rozważyć odblokowanie niektórych i zablokowanie oryginalnej reguły wykorzystującej `pam_unix.so`, jeśli np. zechcemy skorzystać z możliwości, jakie daje moduł `pam_cracklib.so`, które mogą być irytujące dla użytkowników, ale zapewniają, że hasła będą trudniejsze do złamania, ponieważ nie będą zawierały słów ze słownika.

Po wykonaniu wszystkich modułów PAM zdefiniowanych w regułach określonych w pliku `/etc/pam.d/login` (i w plikach włączanych przez reguły z tego pliku) i spełnieniu warunków sukcesu zbioru reguł, procedura logowania użytkownika w systemie jest kontynuowana. Tworzona jest powłoka użytkownika oraz wykonywane są inne operacje związane z rozpoczęciem sesji użytkownika, zgodnie z systemowymi i prywatnymi plikami inicjującymi.

Proces uwierzytelniający wykorzystujący podsystem PAM jest skomplikowany i może niejednego początkującego administratora przyprawić o ból głowy. Mimo kłopotów, jakie może powodować, zapewnienie bezpieczeństwa jest jednak najważniejszym obowiązkiem administratora. W środowisku domowym trudno wyobrazić sobie potrzebę ingerencji w konfigurację PAM, ale w środowisku korporacyjnym lub akademickim z pewnością pojawi się potrzeba modyfikowania (a przede wszystkim zrozumienia) podsystemu PAM. Być może pocieszające będzie to, że bez elastyczności, jaką daje mechanizm PAM, poziom komplikacji kodu aplikacji byłby znacznie większy.

Konfiguracja plików różnych modułów PAM

Każdy tekstowy plik konfiguracyjny umieszczony w katalogu */etc/pam.d* definiuje mechanizm PAM dla jednej aplikacji lub usługi zgodnej z tym mechanizmem. Jak wspominałem przy okazji omawiania pliku */etc/pam.d/login* w poprzednim podrozdziale, niektóre moduły PAM wykorzystują pliki konfiguracyjne o znaczeniu specyficznym dla tych modułów. Te pliki konfiguracyjne modułów PAM są zapisane w katalogu */etc/security*. Choć te pliki muszą istnieć, nic nie stoi na przeszkodzie, aby nie zawierały żadnej znaczącej zawartości. Istnieją po prostu po to, by użytkownik mógł uzupełnić je o odpowiednie ustawienia. W domyślnej instalacji Ubuntu 6.06 można znaleźć następujące pliki konfiguracyjne modułów PAM:

- ♦ *access.conf* — konfiguracja kontroli dostępu na potrzeby modułu *pam_access.so*,
- ♦ *group.conf* — konfiguracja kontroli przynależności do grup, na potrzeby sesji, wykorzystywana przez moduł *pam_group.so*,
- ♦ *limits.conf* — konfiguracja mechanizmu ograniczeń dostępu do zasobów systemowych (dla użytkownika), wykorzystywana przez moduł *pam_limits.so*,
- ♦ *pam_env.conf* — konfiguracja zmiennych środowiska, wykorzystywana przez moduł *pam_env.so*,
- ♦ *time.conf* — konfiguracja czasowych ograniczeń dostępu do usług i aplikacji (dla użytkownika), wykorzystywana przez moduł *pam_time.so*.

Po zainstalowaniu innych systemów uwierzytelniających, zgodnych z podsystemem PAM, w katalogu */etc/security* mogą pojawić się dodatkowo ich pliki konfiguracyjne. Powyższa lista obejmuje pliki konfiguracyjne tylko standardowo instalowanych mechanizmów uwierzytelniających.

Co zrobić, gdy brakuje plików konfiguracyjnych?

Podsystem PAM to bardzo użyteczny mechanizm uwierzytelniający wykorzystywany przez wiele aplikacji; przy jego użyciu administratorzy mogą zapewnić dostęp do aplikacji i usług jedynie tym użytkownikom, którzy ten dostęp mieć powinni. Nietrudno zatem zrozumieć, że prawidłowa konfiguracja mechanizmu PAM to sprawa kluczowa. Co jednak zrobić, gdy aplikacja wykorzystuje PAM, a my nie skonfigurujemy odpowiedniego pliku z definicją reguł PAM lub gdy odpowiedni plik zostanie omyłkowo usunięty? Aby obsłużyć przypadki tego typu, biblioteka Linux-PAM zawiera domyślny plik konfiguracyjny wykorzystywany przez aplikacje i usługi nieposiadające własnych plików konfiguracyjnych. Ten plik to */etc/pam.d/other*; pomijając puste wiersze i komentarze, znajdziemy w nim następujące reguły:

```
@include common-auth
@include common-account
@include common-password
@include common-session
```

Zawartość każdego z tych plików została omówiona wcześniej w tym rozdziale, w punkcie zatytułowanym „Przykład: konfiguracja PAM dla procesu logowania w systemie”. W Ubuntu plik konfiguracyjny */etc/pam.d/other* zawiera rozsądne minimum, wystarczające

do weryfikacji uprawnień dostępu do różnych aplikacji, ponieważ upewniamy się, że są one wykonywane przynajmniej przez użytkowników, którzy są w stanie zalogować się w tym systemie, a ponadto tworzone są odpowiednie zapisy w dziennikach systemowych.



Kiedy brakuje pliku konfiguracyjnego PAM, z reguły mamy problem z konfiguracją systemu lub zdarzyła się sytuacja, gdy ktoś zdecydował się wykorzystać mechanizm PAM bez należytego namysłu. Dlatego warto rozważyć użycie pliku `/etc/pam.d/other` do zdefiniowania skrajnie paranoidalnego mechanizmu bezpieczeństwa, odmawiającego dostępu do nienależycie skonfigurowanych programów i usług:

```
auth        required pam_deny.so
auth        required pam_warn.so
account     required pam_deny.so
account     required pam_warn.so
password    required pam_deny.so
password    required pam_warn.so
session     required pam_deny.so
session     required pam_warn.so
```

Ponieważ wpisy typu `required` dla każdego z modułów są wykonywane nawet mimo zwrócenia kodu niepowodzenia, po wywołaniu modułu `pam_deny.so` (który zawsze zwraca kod niepowodzenia operacji uwierzytelniania) jest wykonywany moduł `pam_warn.so`, zapisujący w dzienniku systemowym komunikat ostrzeżenia o próbie użycia aplikacji lub usługi. Takie zapisy w dzienniku pozwolą administratorowi zidentyfikować problem z niesubordynowanym użytkownikiem, chyba że ten wcześniej zjawi się u administratora z pretensjami, że nie może uruchomić jakiegoś nieautoryzowanego oprogramowania.

Konfiguracja polecenia `sudo` w systemie Ubuntu

Jak wspominałem w rozdziale 4., w punkcie zatytułowanym „Wykonywanie czynności wymagających specjalnych uprawnień”, do wykonywania zadań administracyjnych Ubuntu nie wykorzystuje tradycyjnego konta `root`. Każdego z użytkowników można dodać do grupy administracyjnej, pozwalając im wykonywać zadania wymagające podwyższonych uprawnień po podaniu własnych haseł. Uzasadnienie rezygnacji z wykorzystania konta `root` (oraz związanego z tym sposobu polegającego na użyciu polecenia `su root`) znajduje się w dokumentacji, dostępnej pod adresem <https://help.ubuntu.com/community/RootSudo>. Mówiąc w skrócie, decyzja ta jest postrzegana jako udoskonalenie bezpieczeństwa, co nie jest dalekie od prawdy. Niezależnie od tego, czy ktoś lubi takie podejście, nienawidzi go lub po prostu nie do końca rozumie, należy przyjąć, że Ubuntu działa właśnie w ten sposób. Gdy nawet ktoś jest święcie przekonany, że takie podejście jest dziwaczne, istnieją marne szanse, iż uda się odwieść od tej decyzji twórców Ubuntu i udowodnić im, że są w tragicznym błędzie. Naprawdę szkoda wysiłku.



Strona <https://help.ubuntu.com/community/RootSudo> zawiera również opis sposobu modyfikacji systemu Ubuntu w taki sposób, aby przywrócić istnienie tradycyjnego konta `root`. Mimo że jest to możliwe, gorąco sugeruję, aby tego nie robić, chyba że istnieją bardzo poważne powody ku temu, by swój system Ubuntu zmodyfikować tak, że będzie inny niż pozostałe systemy Ubuntu zainstalowane w tym układzie słonecznym.

W tym podrozdziale omówię techniczne szczegóły funkcjonowania polecenia `sudo`, jego plik konfiguracyjny oraz sposoby udostępnienia uprawnień administracyjnych innym użytkownikom systemu.

Wykorzystanie polecenia `sudo` w systemach Ubuntu wiąże się z pewnymi specyficznymi zagadnieniami, które wyjaśnia poniższa lista.

- ♦ `sudo` jest poleceniem umożliwiającym wykonywanie poleceń działających w wierszu poleceń, takich jak `apt-get`, `aptitude`, `dpkg` itp. Wywołując polecenia wyposażone w graficzny interfejs użytkownika, które wymagają uprawnień administratora, należy korzystać z polecenia `gksudo`.

Problem z wykorzystaniem polecenia `sudo` zamiast `gksudo` do uruchamiania programów z interfejsem graficznym polega na tym, że może to spowodować zmianę właściciela pliku `.ICEauthority` zapisanego w domowym katalogu użytkownika. Plik ten jest używany przez wiele aplikacji KDE, GNOME oraz systemu X Window do identyfikowania aplikacji, które wykorzystują dany ekran (*display*) X Window. Jeśli polecenie `sudo` zostanie zastosowane do uruchomienia aplikacji z interfejsem graficznym, a po tej operacji nie uda się uruchomić innej aplikacji, to może być to sygnał, że mamy do czynienia z taką właśnie sytuacją. Plik `.ICEauthority` można usunąć z poziomu wiersza poleceń za pomocą polecenia `sudo rm .ICEauthority`.

- ♦ Po uruchomieniu polecenia `sudo` lub `gksudo` fakt uzyskania uprawnień administratora jest przechowywany w systemie przez 15 minut. W tym czasie kolejne wywołania tych poleceń nie wymagają podania hasła użytkownika, ale polecenia `sudo` lub `gksudo` należy — oczywiście — wywoływać. Po 15 minutach nieużywania poleceń `sudo` i `gksudo` ponowne ich wywołanie spowoduje konieczność podania hasła użytkownika.



Naprawdę leniwy użytkownik (jakim również byłem kiedyś) szybko spostrzeże, że wywołanie polecenia `sudo bash` spowoduje, że powłoka systemowa zostanie uruchomiona z uprawnieniami administratora, dając nam „wirtualne konto” `root`. Dzięki temu nie będziemy zmuszeni do podawania swojego hasła podczas wydawania kolejnych poleceń. Ta metoda jest wygodna, ale stanowi przykład złych nawyków i spowoduje nieprzychylny komentarz każdego wielbiciela Ubuntu, który nas na tym przyłapie. Zatem zwolennikom tej metody szczerze doradzam dyskrecję.

Działanie polecenia `sudo` jest kontrolowane przez zawartość pliku `/etc/sudoers`. Ze względów bezpieczeństwa ten plik tekstowy nie jest dostępny, nawet do wglądu, zwykłym użytkownikom. Aby zobaczyć jego zawartość, należy posłużyć się poleceniem `visudo` z uprawnieniami administratora, np.:

```
$ sudo visudo
```

Polecenie `visudo` to specjalna modyfikacja polecenia `vi`, która zapewnia, że plik `/etc/sudoers` może być modyfikowany tylko przez jednego użytkownika. Nie trzeba też podawać argumentu, ponieważ to polecenie wie, że ma otworzyć do edycji plik `/etc/sudoers`. Oczywiście, plik ten można otworzyć w dowolnym edytorze tekstu, po prostu wówczas nie będzie zabezpieczenia przed jednoczesnym otwarciem pliku przez kilka osób, co zapewnia polecenie `visudo`.

Jak zobaczymy dalej, w pliku `/etc/sudoers` mogą występować puste wiersze (które są ignorowane) i komentarze, czyli ciągi znaków rozpoczynające się od znaku `#`. Oto znaczące wpisy w domyślnym pliku `/etc/sudoers` w systemie Ubuntu:

```
Defaults    !lecture,tty_tickets,!fqdn
root        ALL=(ALL) ALL
%admin       ALL=(ALL) ALL
```

Pierwszy z wierszy określa ustawienia domyślnych parametrów polecenia `sudo`. Omówię je kolejno.

- ♦ `!lecture` — wyłącza wypisywanie tekstu informującego o konsekwencjach wykorzystania polecenia `sudo`. W systemie Ubuntu opierającym swoje standardowe działanie na mechanizmie `sudo` byłoby to dość problematyczne, dlatego ta funkcja została wyłączona.
- ♦ `tty_tickets` — wymusza konieczność uwierzytelniania użytkowników niezależnie dla każdego `tty` lub `pty`. Przykładowo po zalogowaniu się i uruchomieniu programu Terminal ta opcja powoduje, że dla każdej powłoki (na poszczególnych kartach) uruchomionej w terminalu użytkownik musi uwierzytelniać się niezależnie od pozostałych. Innymi słowy, uwierzytelnienie polecenia `sudo` (i związany z tym 15-minutowy okres „pamiętania” tego faktu przez system) jest ważne tylko dla jednej karty aplikacji terminala.
- ♦ `!fqdn` — wyłącza konieczność wykorzystania kwalifikowanych nazw domenowych w pliku `/etc/sudoers`. Zastosowanie tej opcji powoduje, że polecenie `sudo` będzie sprawdzało za pomocą usługi DNS, czy próba jego użycia pochodzi z systemu o poprawnej nazwie domenowej, co wymusza poprawną konfigurację i dostęp do serwera DNS. W systemie Ubuntu, który wykorzystuje `sudo` jako podstawowy mechanizm administracyjny, wykorzystanie opcji `fqdn` w przypadku wadliwie działającego serwera DNS może spowodować poważne problemy.

Jednak najbardziej interesujące są następne dwa wpisy. Pierwszy z nich definiuje, że użytkownik `root` ma dostęp do każdego polecenia w systemie:

```
root        ALL=(ALL) ALL
```

Drugi wiersz określa dostęp dla wszystkich członków grupy `admin` (zdefiniowanej w pliku `/etc/group`). Ci użytkownicy również mogą wykonywać dowolne polecenia (czyli w praktyce mają takie same uprawnienia jak `root`):

```
%admin      ALL=(ALL) ALL
```

Aby przydzielić uprawnienia administratorskie innym użytkownikom, wystarczy dodać ich do grupy `admin`. Do tego celu można posłużyć się narzędziem Użytkownicy i grupy (co zostało omówione szczegółowo wcześniej w niniejszym rozdziale) lub skorzystać z wiersza poleceń. Przykładowo w celu dodania użytkownika `znowak` do grupy użytkowników upoważnionych do użycia polecenia `sudo` wywołamy następujące polecenie:

```
$ sudo adduser znowak admin
```

Naprawianie uszkodzonego polecenia `sudo`

Instalacja niektórych wersji Ubuntu w trybie eksperta wykorzystuje standardowy mechanizm `root`, co może spowodować, że system pozostanie w stanie, w którym zwykli użytkownicy nie będą mieli możliwości wywoływania uprzywilejowanych poleceń. Również w systemie Ubuntu zainstalowanym standardową metodą usunięcie plików `/etc/group` lub `/etc/sudoers` może spowodować problem z wykonywaniem uprzywilejowanych operacji. W tym drugim przypadku należy ponownie uruchomić system, nacisnąć klawisz `Esc` w celu wyświetlenia menu programu rozruchowego GRUB i wybrać pozycję *Recovery Mode*. To spowoduje uruchomienie systemu z powłoką konta `root`.

W trybie naprawczym należy sprawdzić, czy plik `/etc/group` zawiera grupę `admin` i użytkowników, którzy mają mieć dostęp do funkcji administracyjnych, należących do tej grupy. Jeśli grupa `admin` nie istnieje, można ją dodać za pomocą następującego polecenia:

```
# addgroup --system admin
```

Następnie należy upewnić się, że plik `/etc/sudoers` zawiera wpisy udostępniające prawa administratora członkom grupy `admin`:

```
%admin ALL=(ALL) ALL
```

Jeśli taki wpis nie istnieje w pliku `/etc/sudoers`, należy go tam dodać.

Po upewnieniu się, że grupa `admin` istnieje i w pliku `/etc/sudoers` jest wpis nadający jej uprawnienia do użycia polecenia `sudo`, wystarczy dodać do niej odpowiednich użytkowników, którzy mają mieć uprawnienia administratora. Można tego dokonać za pomocą następującego polecenia:

```
# adduser znowak admin
```

To polecenie doda użytkownika `znowak` do grupy `admin`, co pozwoli mu na wykonywanie programów z uprawnieniami administratora przy pomocy poleceń `sudo` i `gksudo`.

Wykorzystanie ACL-i do zaawansowanego zarządzania dostępem do plików

Standardowe uprawnienia do plików znane z systemów Unix są w zupełności wystarczające, gdy z systemu korzysta niewielka liczba użytkowników o niewygórowanych wymaganiach związanych ze współużytkowaniem tych samych plików. W rozdziale 4., w punkcie „Podstawowe koncepcje: użytkownicy i grupy” wprowadziłem podstawowe zagadnienia związane z grupami, natomiast w punkcie „Uprawnienia do plików i katalogów w Linuksie” omówiłem zagadnienia uprawnień do plików i katalogów powiązane z grupą zdefiniowaną jako ich właściciel.

Grupy w Linuksie są tradycyjnym mechanizmem definiowania dostępu do plików i katalogów dla większej liczby użytkowników. Niestety, wykorzystanie grup do kontroli dostępu wymaga interwencji administratora systemu za każdym razem, gdy trzeba dodać lub usunąć użytkownika uprawnionego do dostępu, to samo dotyczy przypadku konieczności zdefiniowania nowego zasobu sieciowego. Mechanizm znany pod nazwą *Access Control Lists* (ACL), obsługiwany przez większość nowoczesnych wersji Linuksa, w tym również przez Ubuntu Linux, eliminuje ten problem, oferując szczegółowy zestaw uprawnień, które mogą być definiowane przez użytkowników dla ich własnych plików i katalogów; dają one możliwości znacznie przekraczające te, które oferowane są przez standardowe mechanizmy użytkowników, grup i uprawnień dostępu do plików. W dystrybucjach Linuksa

i w innych systemach uniksowych ACL-e są dostępne od dłuższego czasu, ale nigdy nie stały się tak popularne, jak na to zasługują, ponieważ składnia poleceń służących do ich obsługi jest bardzo skomplikowana. Jak się jednak przekonamy w dalszej części rozdziału, istnieją już narzędzia graficzne służące do zarządzania ACL-ami, zatem użytkownicy Ubuntu mogą pracować i stosować ACL-e bez większych trudności.

ACL według najprostszej definicji jest listą użytkowników i (lub) grup wraz z ich prawami dostępu do określonego pliku lub katalogu. ACL-e umożliwiają definiowanie szczegółowych zestawów uprawnień typu: „tylko *jkowalski* i *znowak* mogą zapisywać ten plik, a użytkownik *amakota* może ten plik tylko odczytywać”. Do tego nie potrzeba definiowania specjalnych grup w systemie Linux. Najważniejszą ich cechą jest jednak to, że każdy użytkownik może definiować i modyfikować ACL-e dowolnych plików i katalogów, których jest właścicielem, bez konieczności angażowania do tego zadania użytkownika o uprawnieniach administratora. ACL-e dają zatem zwykłym użytkownikom możliwość zarządzania dostępem do ich własnych plików.

Wielbiciele akronimów z pewnością zainteresują się tym, że ACL-e (ang. *Access Control Lists*) w formie zaimplementowanej w Linuksie zostały zdefiniowane w postaci szkicu standardu POSIX (ang. *Portable Operating System Interface*) o numerze 1003.1e, draft 17, przez organizację IEEE (ang. *Institute of Electrical and Electronics Engineers*). Definicja ACL nie jest oficjalnym standardem, ale jest dostępna publicznie i stała się podstawą implementacji ACL w nowoczesnych systemach operacyjnych, m.in. w Linuksie.

W dalszej części tego rozdziału wyjaśnię, w jaki sposób zainstalować polecenia niezbędne do użycia ACL-i, omówię inne wymagania, jakie muszą być spełnione w celu ich wykorzystania w systemie, oraz wyjaśnię, w jaki sposób przeglądać, definiować, aktualizować i usuwać ACL-e plików i katalogów z użyciem poleceń obsługiwanych z wiersza poleceń oraz narzędzi wyposażonych w graficzny interfejs użytkownika.



W tym punkcie nie omawiam zagadnienia rozszerzonych atrybutów plików. Jest ono często utożsamiane z ACL-ami, ale dotyczy zupełnie innych problemów.

Przegląd ACL-i w Linuksie

Linux obsługuje dwa podstawowe typy ACL-i:

- ♦ ACL-e wykorzystywane do kontroli dostępu do plików i katalogów;
- ♦ ACL-e definiowane na poziomie katalogów, określane jako maski ACL-i, a służące do definiowania domyślnych ACL-i dla plików i katalogów tworzonych w danym katalogu.

ACL-e są reprezentowane w postaci standardowego formatu składającego się z trzech pól oddzielonych przecinkami.

- ♦ Pierwsze pole wpisu ACL to typ wpisu (*entry type*) o wartości **u** (*user*), **g** (*group*), o (*other*) lub **m** (*mask*).
- ♦ Drugie pole wpisu ACL to nazwa użytkownika, jego identyfikator liczbowy (UID), nazwa grupy lub jej liczbowy identyfikator (GID), w zależności od wartości

pierwszego pola. Jeśli to pole jest puste, ACL odnosi się do użytkownika lub grupy będących właścicielem pliku lub katalogu. To pole jest puste w przypadku ACL-i typu *m* lub *o*.

- ♦ Trzecie pole to definicja uprawnień dostępu dla danego wpisu. Tutaj istnieje możliwość zastosowania jednej z dwóch postaci:
 - ♦ Standardowe uprawnienia w stylu systemu Unix: *rwx* (**R**ead, **W**rite, **eX**ecute, przy czym uprawnienie *x* w przypadku katalogu określa możliwość przeglądania katalogu). Każda litera może być zastąpiona znakiem *-* (znak minus), co symbolizuje brak uprawnienia danego typu. Uprawnienia muszą wystąpić w określonej kolejności.
 - ♦ Względna postać uprawnień, gdzie każde uprawnienie może być poprzedzone znakiem *+* (plus) lub *^* (daszek), podobnie jak w składni definicji uprawnień w ramach polecenia *chmod* stosowanej przez użytkowników, którzy mają problem ze stosowaniem ósemkowych wartości uprawnień. W tej postaci ACL-i symbole *+* i *^* poprzedzają symbole *r*, *w* lub *x*, co oznacza, że do bieżącego zbioru uprawnień użytkownika lub grupy do danego pliku lub katalogu należy dodać (znak *+*) lub usunąć (znak *^*) określone uprawnienie.

Wpisy ACL zapisane w plikach są oddzielane białymi znakami lub znakami nowego wiersza. Znak *#* oraz wszystkie znaki następujące po nim aż do końca wiersza są ignorowane.

Instalacja w Ubuntu poleceń do obsługi ACL-i

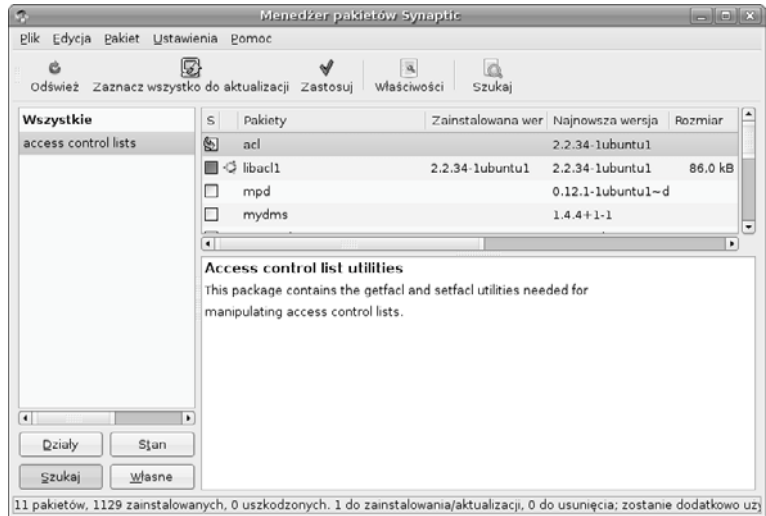
Polecenia specjalistyczne, m.in. służące do obsługi ACL-i, nie są instalowane w ramach standardowej instalacji Ubuntu, ponieważ same ACL-e są, spójrzmy prawdzie w oczy, dość mało popularną technologią. Jednak, jak ma to miejsce w przypadku wszystkich pakietów dostępnych dla Ubuntu, można je zainstalować z łatwością, wykorzystując polecenie *apt-get* lub *aptitude* (w systemach niewyposażonych w graficzny interfejs użytkownika) albo program Synaptic w standardowej, biurkowej instalacji Ubuntu. Polecenia do obsługi ACL-i z wiersza poleceń znajdują się w pakiecie o nazwie *acl*.

Aby zainstalować ten pakiet z wiersza poleceń, wystarczy wywołać polecenie *apt-get install acl* lub *aptitude -r install acl*.

Za pomocą programu Synaptic pakiet *acl* również instaluje się w standardowy sposób. Trzeba uruchomić Synaptic: z menu systemowego wywołujemy funkcję *System/Administracja/Menedżer pakietów Synaptic*. Po uruchomieniu programu należy wyszukać ciąg znaków *access control lists* (zwracając uwagę, aby w oknie wyszukiwania zaznaczone były opcje *Opis* i *Nazwa*). Po zakończeniu wyszukiwania w liście wyników trzeba znaleźć pakiet *acl*, kliknąć jego nazwę prawym przyciskiem myszy i z menu kontekstowego wybrać pozycję *Zaznacz do instalacji*. Na rysunku 21.14 przedstawiam okno programu Synaptic z zaznaczonym do instalacji pakietem *acl*.

Po zaznaczeniu do instalacji pakietu *acl* należy wykonać drugie wyszukiwanie, tym razem w poszukiwaniu ciągu znaków ACL. Po zakończeniu wyszukiwania w wynikach należy znaleźć pakiet *eiciel*. To graficzne narzędzie do zarządzania ACL-ami. Trzeba kliknąć jego

Rysunek 21.14.
Instalacja narzędzi
do zarządzania
ACL-ami



nazwę prawym przyciskiem myszy, z menu kontekstowego wybrać pozycję *Zaznacz do instalacji*. Więcej informacji na temat programu *eiciel* podam w dalszej części tego rozdziału.

Po zaznaczeniu do instalacji pakietu *eiciel* należy kliknąć ikonę *Zastosuj* z paska narzędziowego programu Synaptic, co spowoduje instalację podstawowych narzędzi do zarządzania ACL-ami plików i katalogów. Po zakończeniu instalacji można zamknąć Synaptic. Jesteśmy już prawie u celu!

Uaktywnienie obsługi ACL-i w systemie plików

Aby skorzystać z ACL-i w celu zwiększenia szczegółowości definiowania uprawnień dostępu do plików i katalogów w wybranym systemie plików, należy system ten zamontować z opcją obsługi ACL-i. W Ubuntu opcja ta nie jest domyślnie aktywna, ale można to z łatwością zmienić. Służy do tego słowo kluczowe `acl` w opcjach montowania systemu plików w pliku `/etc/fstab`. Standardowo w tym pliku znajdziemy następujący (mniej więcej) wpis:

```
/dev/hda1 / ext3 defaults,errors=remount-ro 0 0
```

Aby uaktywnić obsługę ACL-i w tym systemie plików, należy zmodyfikować ten wpis w pliku `/etc/fstab`. Można tego dokonać za pomocą ulubionego edytora tekstów, np.:

```
$ sudo emacs /etc/fstab
```

Ten sam wpis z obsługą ACL-i będzie miał następującą postać:

```
/dev/hda1 / ext3 defaults,acl,errors=remount-ro 0 0
```

Aktualizacja tego pliku spowoduje, że przy następnym automatycznym zamontowaniu systemu plików `/dev/hda1` zostanie w nim uaktywniona obsługa ACL-i. Główny system plików jest montowany z reguły tylko raz na każdą sesję działania systemu, ale nie ma

konieczności ponownego uruchamiania systemu w celu wykorzystania ACL-i. W tym celu wystarczy ponownie zamontować system plików `/dev/hda1` z odpowiednimi opcjami, np. wywołując następujące polecenie:

```
$ sudo mount -o remount,acl /dev/hda1
```

Od tego momentu możemy zacząć korzystać z możliwości, jakie dają ACL-e.

Praca z ACL-ami z wiersza poleceń

Jak się przekonamy w tym punkcie, obsługa ACL-i z wiersza poleceń nie należy do przyjemności, efekty naszego działania dla postronnych obserwatorów niewiele będą się różnić od przypadkowego wciskania klawiszy czy wyświetlenia na ekranie zawartości pliku binarnego. Niemniej jednak warto rozumieć wyniki działania poleceń obsługiwanych z wiersza poleceń do obsługi ACL-i. Jeśli ktoś będzie musiał kiedyś napisać skrypt do masowego definiowania ACL-i, z pewnością z dumą stwierdzi: „Oczywiście, wiem, jak obsługiwać ACL-e z wiersza poleceń”.

Przegląd narzędzi do obsługi ACL-i z wiersza poleceń

Pakiet *acl* dla Linuksa zawiera następujące narzędzia do tworzenia, modyfikowania i odczytu ACL-i.

- ♦ `chacl` — służy do modyfikowania, odczytu i usuwania ACL-i plików oraz katalogów na poziomie grup, użytkowników i masek.
- ♦ `getfacl` — służy do przeglądania ACL-i plików i katalogów.
- ♦ `setfacl` — służy do ustawiania ACL-i plików i katalogów.

Kolejne podpunkty omawiają zasady stosowania tych narzędzi.

Odczyt aktualnych ACL-i z wiersza poleceń

W ramach przykładu użycia ACL-i wykorzystamy katalog o następujących uprawnieniach:

```
$ ls -al
razem 52
drwxr-xr-x  2 jkowalski jkowalski  4096 2007-03-18 23:41 .
drwxr-xr-x 18 jkowalski jkowalski  4096 2007-03-18 23:39 ..
-rw-----  1 jkowalski jkowalski  43824 2007-03-18 23:41 informacje.xml
```

Na początku ten katalog ma aktywny domyślny zestaw ACL-i:

```
$ getfacl .
# file: .
# owner: jkowalski
# group: jkowalski
user::rwx
group::r-x
other::r-x
```

Domyślny zestaw ACL-i dla pliku *informacje.xml* jest następujący:

```
$ getfacl informacje.xml
# file: informacje.xml
# owner: jkowska
# group: jkowska
user::rw-
group::---
other::---
```

Domyślny zestaw ACL-i dla pliku w katalogu, w którym nie został zdefiniowany domyślny zestaw ACL-i, jest ustalany na podstawie domyślnych uprawnień do pliku zdefiniowanych dla użytkownika, który go tworzy. Domyślne uprawnienia są określone na podstawie maski użytkownika (*umask*), to zagadnienie zostało szczegółowo omówione w rozdziale 4., w punkcie „Nadawanie uprawnień domyślnych podczas tworzenia plików i katalogów”.

Ustawianie ACL-i z wiersza poleceń

Istnieją trzy sposoby definiowania ACL-i pliku lub katalogu:

- ♦ przez jawne ustawienie odpowiednich wartości z użyciem polecenia `setfacl`, co powoduje nadpisanie istniejących wartości;
- ♦ za pomocą polecenia `setfacl` z opcją `-m` (modyfikacja);
- ♦ za pomocą polecenia `chacl`.

W kilku kolejnych przykładach wykorzystam polecenie `chacl` do zmiany ACL-i, ponieważ jest bezpieczniejsze i łatwiejsze w użyciu, gdyż nie powoduje usunięcia istniejących ACL-i. Dodatkowo nieco lepiej, w porównaniu z poleceniem `setfacl`, obrazuje mechanizmy działania ACL-i.

Przykładowo w celu dodania użytkownika *znowak* do listy użytkowników uprawnionych do odczytu pliku *informacje.xml* zastosuję polecenie `chacl`:

```
$ chacl u::rw-,g::r--,o::---,u:znowak:r--,m::rw- informacje.xml
```

Nie, to nie błąd w druku: właśnie w ten sposób definiuje się ACL-e. Taka jest cena, jaką przychodzi płacić za zwiększenie możliwości. Jak wspominałem wcześniej, ACL-e definiuje się w postaci trzech pól odseparowanych przecinkami. W tych kolumnach określa się uprawnienia użytkownika (właściciela pliku), grupy (grupy zdefiniowanej dla pliku) oraz pozostałych użytkowników systemu. Modyfikując ACL za pomocą polecenia `chacl`, należy określić ACL pliku, po czym określić zmiany, jakie mają zostać wprowadzone w tym ACL-u. Fragment `u::rw-,g::r--,o::---` w tym przykładzie określa istniejący ACL danego pliku, natomiast fragment `u:znowak:r--,m::rw-` określa efektywną maskę uprawnień nowego użytkownika, które mają zostać dodane do istniejących. Efektywna maska uprawnień jest unią wszystkich uprawnień użytkownika, grupy oraz pozostałych uprawnień do pliku lub katalogu. Dodając nowego użytkownika do ACL-i pliku, należy zdefiniować właśnie tę maskę.

Trzeba zwrócić uwagę, że nazwa użytkownika dodawanego do ACL-i jest umieszczana między dwoma przecinkami w elemencie „u” wpisu ACL. Gdy po wywołaniu powyższego

polecenia sprawdzimy ACL-e pliku *informacje.xml* za pomocą polecenia `getfacl`, otrzymamy następujący wynik:

```
$ getfacl informacje.xml
# file: informacje.xml
# owner: jkowalski
# group: jkowalski
user::rw-
user:znowak:r--
group::r--
mask::rw-
other::---
```

Gdy jednak wywołamy polecenie `ls -l`, przekonamy się, że standardowe uprawnienia uniksowe nie uległy zmianie:

```
$ ls -al
razem 52
drwxr-xr-x  2 jkowalski jkowalski  4096 2007-03-18 23:41 .
drwxr-xr-x 18 jkowalski jkowalski  4096 2007-03-18 23:39 ..
-rw-----  1 jkowalski jkowalski 43824 2007-03-18 23:41 informacje.xml
```

Zagadnieniem ciekawszym od nadawania prawa odczytu jest umożliwianie użytkownikom zapisu do określonych plików; spróbujmy np. dać użytkownikowi *znowak* prawo zapisu i odczytu pliku *informacje.xml*. W tym celu ponownie posłużę się poleceniem `chac1`:

```
$ chac1 u::rw-,g::r--,o::---,u:znowak:rw-,m::rw- informacje.xml
```

Ponownie nazwa użytkownika, dla którego definiowane są uprawnienia, jest ujęta w ramach elementu „u” wpisu ACL. Polecenie `getfacl` pokazuje, że użytkownik *znowak* posiada prawa zapisu i odczytu pliku *informacje.xml*:

```
$ getfacl informacje.xml
# file: informacje.xml
# owner: jkowalski
# group: jkowalski
user::rw-
user:znowak:rw-
group::rw-
other::---
mask::rw-
```

Bardzo niefortunnie się składa, że ACL-e obsługuje się w tak mało elegancki sposób, ale głównym powodem takiego stanu rzeczy są bardzo duże możliwości tego mechanizmu. Nietrudno domyślić się, że obsługa ACL-i z wiersza poleceń jest narażona na liczne błędy literowe. Gdyby znaleźć jakieś przyjazne użytkownikowi narzędzie do zarządzania tą cenną funkcją systemu...

Praca z ACL-ami przy użyciu graficznego narzędzia

Na szczęście, z pomocą jak zwykle przychodzi Święty Mikołaj, tym razem skrywający się pod nazwiskiem Roger Ferrer Ibanez. Roger jest głównym autorem programu *eiciel* — edytora ACL-i dla środowiska graficznego GNOME. Co oznacza nazwa *eiciel*? To lekko zdeformowana („z francuska”) forma wymowy tematu naszego podrozdziału, czyli ACL.

Edytor *eiciel* udostępnia interfejs graficzny, za którego pomocą użytkownicy mogą ustawiać, modyfikować i usuwać ACL-e w systemach plików ext2, ext3, XFS, JFS, ReiserFS oraz potencjalnie we wszystkich innych, obsługujących mechanizm POSIX ACL.

Pakiet *eiciel* zainstalowaliśmy wcześniej w tym rozdziale razem z pozostałymi narzędziami do obsługi ACL-i (w punkcie „Instalacja w Ubuntu poleceń do obsługi ACL-i”). Niestety, instalacja pakietu *eiciel* nie tworzy dla niego pozycji w menu Aplikacje, z tego powodu jesteśmy zmuszeni uruchamiać ten program z wiersza poleceń, np. z terminala GNOME, z programu *xterm*. Możemy też utworzyć dla niego aktywator na pulpicie. Pozycję w menu dla programu *eiciel* można również zdefiniować samodzielnie, odpowiednie informacje na ten temat można znaleźć w podrozdziale rozdziału 5. zatytułowanym „Dostosowywanie menu”. Na rysunku 21.15 przedstawiam główne okno programu *eiciel*.

Rysunek 21.15.

Główne okno
programu *eiciel*



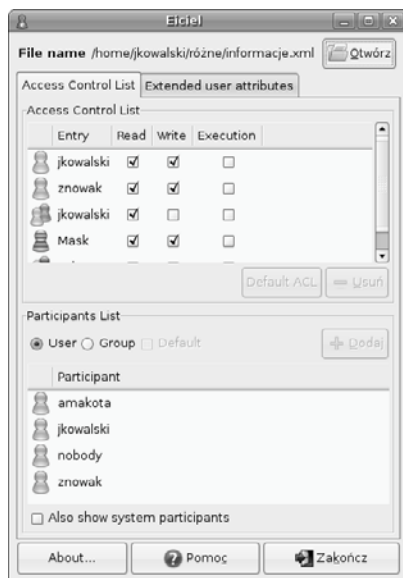
Uruchamiając program *eiciel* z wiersza poleceń, w ramach argumentu wywołania można od razu wskazać plik lub katalog, dla którego chcemy zdefiniować ACL-e. Plik lub katalog można również wskazać z poziomu programu, w tym celu należy kliknąć przycisk *Otwórz* w prawym górnym oknie programu. Na rysunku 21.16 przedstawiam zestaw ACL-i dla pliku *informacje.xml* (który również służył jako przykład stosowania ACL-i za pomocą narzędzi obsługiwanych z wiersza poleceń).

W górnej części okna *Eiciel* znajdują się aktualne ACL-e pliku lub katalogu otwartego w programie *eiciel*. Domyślnie dolna część okna zawiera listę zwykłych użytkowników systemu. Jeśli chcemy zdefiniować ACL dla grupy, a nie dla użytkownika, należy kliknąć przełącznik *Group*. Aby zdefiniować ACL-e dla użytkowników systemowych (tzn. tych o identyfikatorach UID poniżej 1000), należy zaznaczyć opcję *Also show system participants* w dolnej części okna *Eiciel*.

Aby dodać użytkownika lub grupę do listy ACL pliku otwartego w programie *eiciel*, trzeba zaznaczyć odpowiednią pozycję w dolnej części okna *Eiciel*. To spowoduje uaktywnienie przycisku *Dodaj*, co przedstawiam na rysunku 21.17.

Rysunek 21.16.

Przeglądanie
definicji ACL dla
przykładowego pliku
w programie eiciel

**Rysunek 21.17.**

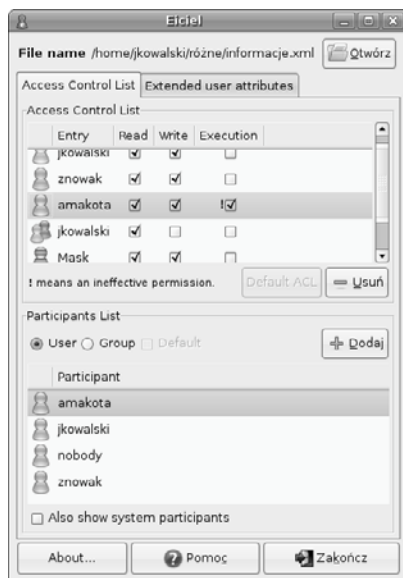
Wybór użytkownika
do przydziału
ustawień ACL



Dodanie użytkownika do ACL-i następuje po kliknięciu przycisku *Dodaj*. Nazwa wybranego użytkownika lub grupy pojawi się w górnej części okna *Eiciel*, natychmiast też dodawany jest odpowiedni wpis do ACL-i danego pliku lub katalogu.

Gdy przyjrzeć się bliżej dodanym przeze mnie ACL-om (patrz rysunek 21.18), można zauważyć, że uprawnienie wykonawcze (*Execute*) jest określone jako nieefektywne (*ineffective*). Przyczyną tego stanu jest to, że sam plik nie ma ustawionego znacznika wykonywalności (uprawnienie x).

Rysunek 21.18.
Domyślny wpis ACL
dla nowo dodanego
użytkownika



Za pomocą programu *eiciel* można dowolnie ustawiać uprawnienia do plików lub katalogów, zmiany będą prezentowane w górnej części okna *Eiciel*. Zmiany są wprowadzane natychmiast w ACL-ach, nie ma konieczności jawnego ich zapisywania. Po zakończeniu pracy z programem można zamknąć jego okno, klikając przycisk *Zakończ*.

Edytor *eiciel* jest doskonałym przykładem użytecznego i prostego narzędzia. To świetny przykład tego, że najnowsze generacje dystrybucji Linuksa oddalają użytkownika od konieczności stosowania tajemnych sztuczek z wierszem poleceń, co naraża ich na popełnianie błędów, a przede wszystkim zniechęca do wykorzystania mało przyjaznych dla użytkownika narzędzi. Więcej informacji na temat programu *eiciel* można znaleźć w jego pomocy, dostępnej po kliknięciu przycisku *Pomoc*. Informacje na temat nowszych wersji programu znajdują się na stronie WWW: <http://rofi.pinchito.com/eiciel>. Roger, należą Ci się podziękowania za niezły program!

Podsumowanie

Na początku tego rozdziału wyjaśniłem podstawowe koncepcje związane z użytkownikami i grupami w Ubuntu. Opisałem sposób działania mechanizmu Pluggable Authentication Modules, służącego do weryfikowania tożsamości użytkownika, określania czasów, w których użytkownik może zalogować się w systemie, oraz innych zadań związanych z logowaniem do systemu Ubuntu. Wyjaśniłem podstawy zarządzania dostępem do plików i katalogów z użyciem standardowych grup użytkowników. Omówiłem zasadę wykorzystywania polecenia *sudo* służącego do nadawania użytkownikom uprawnień administratora. W ostatnim podrozdziale wyjaśniłem rolę i zasady działania mechanizmu Access Control Lists, nowoczesnego i cechującego się dużą elastycznością mechanizmu służącego do nadawania użytkownikom uprawnień do plików i katalogów.

W rozdziale 22. omawiam różne mechanizmy wykonywania kopii zapasowych oraz odzyskiwania plików z tych kopii. Na początek wspomnę o poleceniach pozwalających na tworzenie archiwów plikowych, po czym skupię się na dedykowanych mechanizmach do wykonywania kopii i do synchronizacji plików oraz katalogów z innymi systemami w sieci. Czytelnik znajdzie tam szczegółowe omówienie programu BackupPC, elastycznej, sieciowej aplikacji ułatwiającej automatyczne wykonywanie kopii zapasowych na jednym, centralnym serwerze kopii zapasowych.

Rozdział 22.

Wykonywanie i przywracanie kopii zapasowych

W tym rozdziale:

- ♦ Podstawowe zagadnienia związane z kopiami zapasowymi
- ♦ Wykonywanie lokalnych kopii zapasowych na nośnikach wymiennych
- ♦ Wykorzystanie programu rsync do wykonywania kopii zapasowych na zdalnych systemach
- ♦ Wykorzystanie aplikacji backuppc do wykonywania kopii zapasowych całych sieci

Kopie zapasowe to nadmiarowe kopie plików i katalogów przechowywanych w systemie komputerowym, zapisane na zewnętrznym nośniku, z reguły umieszczonym z dala od komputera źródłowego. Wykonywanie kopii zapasowych to zajęcie czasochłonne, ale absolutnie konieczne, jeśli mamy do czynienia z plikami, pocztą i innymi danymi o jakiegokolwiek wartości dla użytkownika komputera.

Kopie zapasowe można porównać do ubezpieczeń motoryzacyjnych, z tą różnicą, że nie są wymagane prawem (w każdym razie nie w większości znanych mi krajów). Są rzadko potrzebne i mamy nadzieję, że się nigdy nie przydadzą. Są z reguły czasochłonne i kosztowne (w końcu czas też ma swoją wartość). Jeśli jednak pewnego mrocznego wieczora okaże się, że usunąłeś swój katalog domowy, albo pojawi się wystraszony użytkownik z informacją, że „omyłkowo” usunął wszystkie dane użytkowników wraz z listą płac czy wszystkie kody źródłowe produktów firmy, lepiej, żeby administrator znalazł dobrą odpowiedź. Oczywiście, najlepsza odpowiedź w takiej sytuacji to: „Natychmiast odtwórz te dane z kopii zapasowej”.

Trudno znaleźć drugie zagadnienie tak niewdzięczne i tak potrzebne jak wykonywanie kopii zapasowych. To jest nudne. To zajmuje mnóstwo czasu. No i — oczywiście — ma kluczowe znaczenie dla bezpieczeństwa danych. W tym rozdziale omawiam problem z punktu widzenia administratora systemu, niezależnie od tego, jak dużą liczbą systemów się zajmuje. Administrator systemu jest odpowiedzialny za to by system był bezpieczny,

dobrze utrzymany, w którym rygorystycznie przestrzegane są reguły wykonywania kopii zapasowych. Powinniśmy czuć się tym bardziej odpowiedzialni, jeśli pod naszą opieką jest tylko jeden komputer (nasz własny), ponieważ w tym przypadku nie ma na kogo zrzucić winy w razie katastrofy. Nawet w takim jednokomputerowym środowisku każdy z pewnością jest świadom znaczenia danych i systemu. Kopie zapasowe zwiększają ich bezpieczeństwo.

W tym rozdziale omawiam różne rozwiązania dotyczące wykonywania zapasowych kopii danych w systemie Ubuntu Linux, począwszy od rozwiązań obsługiwanych z wiersza poleceń, po dość imponujące narzędzia wyposażone w interfejs graficzny. Przedstawiam również drugą stronę kopii zapasowych, czyli odtwarzanie z nich plików. W końcu to właśnie jest główny powód ich wykonywania.

Podstawowe zagadnienia dotyczące kopii zapasowych

Przed zagłębieniem się w zagadnienia związane z narzędziami wykorzystywanymi do wykonywania kopii zapasowych warto przejrzyć podstawowe zagadnienia i podejścia dotyczące tego problemu w systemie komputerowym. Choć z pewnością większość czytelników zna na pamięć wszystkie te zagadnienia do tego stopnia, że zdarza się im mamrotać przez sen polecenia związane z wykonywaniem kopii zapasowych, to uważam, iż warto przyjrzeć się tematowi, aby lepiej przyswoić tajniki narzędzi, które zostaną omówione w dalszej części rozdziału. Nie jestem w stanie stwierdzić, czy czytelnik tej książki wziął ją do ręki, ponieważ używa Ubuntu na komputerze biurkowym, czy też jest administratorem rozległej sieci korporacyjnej wykorzystującej Ubuntu na wielu serwerach, zatem rozgrzeszam się z nadgorliwości. To ona powoduje, że poruszę wiele zagadnień, które w przypadku systemów domowych z pewnością nie znajdują praktycznego zastosowania, ale są obowiązkowe w wielosystemowych instalacjach biznesowych lub akademickich.

Po co wykonywać kopie zapasowe?

W idealnym świecie kopie zapasowe nie byłyby koniecznością. Sprzęt i oprogramowanie komputerowe zawsze działałyby prawidłowo, użytkownicy nie popełnialiby błędów, a ja byłbym gwiazdą rocka. Niestety, jest nieco inaczej. Administratorzy systemów komputerowych i inni pracownicy departamentów IT są zmuszeni do wykonywania kopii zapasowych, które pozwalają zabezpieczyć się przed wieloma problemami. Oto niektóre z nich.

- ♦ Naturalne kataklizmy, takie jak pożary, powódzie i trzęsienia ziemi, które niszczą systemy komputerowe.
- ♦ Awarie dysków twardych i innych nośników danych, które uniemożliwiają odczyt zapisanych na nich danych.
- ♦ Problemy z oprogramowaniem, takie jak uszkodzenia systemów plików, które mogą spowodować usunięcie plików i katalogów na etapie naprawy i sprawdzania spójności.

Kilka słów o nośnikach

Wykonywanie kopii zapasowych jest procesem czasochłonnym, a same kopie zajmują znaczne ilości miejsca na nośnikach. W czasach, gdy systemy komputerowe wykorzystują wielogigabajtowe zasoby informacji, wiąże się to z koniecznością posiadania wydajnych i bardzo pojemnych urządzeń do wykonywania kopii zapasowych lub ze znaczną ilością pracy przy ładowaniu płyt CD, DVD dla każdej wykonywanej kopii zapasowej. Inne, historyczne już rozwiązania, takie jak dyski Zip, Jazz, LS-120 itp., okazały się tak niewielkim pojemnościowo zasobem, że stały się bezużyteczne do celów wykonywania kopii zapasowych znacznych ilości danych i nadają się właściwie do zapisu pojedynczych, znaczących plików lub zestawów plików konfiguracyjnych. Tego typu nośniki nie będą omawiane w tym rozdziale.

Jeśli dla kogoś określenie „nośnik taśmowy” wywołuje skojarzenia rodem z czasów komputerów klasy mainframe lub wręcz ze starych filmów science-fiction, warto zweryfikować swoją wiedzę. Nawet na potrzeby domowe współczesne napędy taśmowe stanowią szybkie rozwiązanie, zapisują znaczne ilości danych, są relatywnie tanie i wykorzystują taśmy o rozmiarach mieszczących się w zamkniętej dłoni. Mimo że bardzo popularne stają się kopie zapasowe na dyskach twardych, szczególnie w środowiskach sieciowych, napędy taśmowe są nadal popularne i opłacalne.

Dyski CD i DVD są szczególnie użyteczne do wykonywania kopii zapasowych systemów domowych, ponieważ są tanie i z reguły zapewniają wystarczającą ilość miejsca do zachowania wybranych plików i katalogów. Do użytku domowego preferuję nośniki CD-R i DVD-R, ponieważ ich odpowiedniki wielokrotnego zapisu i odczytu są znacznie droższe, a ich cecha „wielokrotnej możliwości zapisu” dotyczy tylko ograniczonej liczby cykli zapisu-odczytu. Z drugiej strony, dyski CD-R i DVD-R rzadko nadają się do zapisywania kopii zapasowych w środowiskach korporacyjnych, ponieważ nawet dyski DVD-R nie mają wystarczającej pojemności, aby wykonać pełną kopię systemu na pojedynczym nośniku. Dzielenie kopii zapasowych na wiele nośników DVD-R to dość trudne zadanie, a dodatkowo dyski DVD-R zapisują się dość wolno. Mogą być użyteczne przy przywracaniu systemu, ponieważ taki nośnik z łatwością można przenieść w miejsce, w którym ma zostać odtworzony, bez konieczności przenoszenia np. napędu taśmowego, czy wykorzystać sieć do odtworzenia kopii. Osobiście, w środowiskach korporacyjnych lub akademickich preferuję przenośne dyski twarde lub taśmy.

- ♦ Awarie oprogramowania, np. zawieszenia programów prowadzące do uszkodzenia lub usunięcia otwartych w nich plików.
- ♦ Błąd ludzki, czyli omyłkowe usunięcie ważnych plików i katalogów.



Wiele osób mylnie traktuje rozwiązania klasy RAID (ang. *Redundant Array of Independent Disks*) jako kopie zapasowe. Macierze RAID stanowią cenny element zwiększający bezpieczeństwo danych i niezawodność systemów w przypadku awarii jednego z napędów dyskowych wchodzących w skład macierzy, ale nie stanowią żadnego zabezpieczenia przed problemami wymienionymi na powyższej liście. W czasie pożaru wszystkie dyski w macierzy spalą się (prawie) dokładnie tak samo.

Oprócz zabezpieczenia przed wymienionymi problemami z dostępem do danych, istnieje wiele proceduralnych i biznesowych powodów wykonywania kopii zapasowych systemów komputerowych. Kompletnie i dokładne kopie zapasowe stanowią:

- ♦ kompletne, historyczne odzwierciedlenie osobistych, korporacyjnych czy też organizacyjnych danych biznesowych i finansowych; tego typu kopie zapasowe mogą pewnego dnia okazać się przydatne, gdy osoba, firma lub organizacja zostaną zmuszone do obrony w sądzie lub w przypadku innych problemów prawnych,

- ♦ źródło informacji historycznych o prowadzonych projektach badawczych i rozwoju oprogramowania,
- ♦ sposób na zachowanie danych, które nie są niezbędne w codziennej pracy, ale mogą być przydatne sporadycznie; dotyczy to również ukończonych projektów, katalogów domowych użytkowników, którzy już nie istnieją w systemie itp.

Na zakończenie rozważań o nośnikach należy wspomnieć o tym, że ich przechowywanie powinno odbywać się z dala od systemów, których dane zawierają. Historia komputeryzacji, niezależnie od tego, czy osobistej, biznesowej, czy też akademickiej, pełna jest opowiadań mrozących krew w żyłach o ludziach, którzy rygorystycznie przestrzegali zasad wykonywania kopii zapasowych, ale przechowywali je w pudełku umieszczonym blisko komputera. Po wystąpieniu klęski żywiołowej administratorzy trafiali na bezrobocie z listą bezużytecznych wymówek. Przechowywanie kopii zapasowych z daleka od źródła ich danych jest kluczowe dla odtworzenia systemu ze stanu fizycznej katastrofy, ale też wiąże się z innym problemem, czyli zapewnieniem odpowiedniego poziomu bezpieczeństwa w miejscu przechowywania kopii.

Większość z nas nie pozostawia otwartych na oścież drzwi wejściowych, wyjeżdżając na wakacje, czy też nie przłącza do internetu niezabezpieczonego (choćby hasłem) systemu. Z tych samych powodów nie należy przechowywać kopii zapasowych w niezaufanym środowisku. To zagadnienie jest szczególnie istotne, gdy jesteśmy odpowiedzialni za system komputerowy wykorzystywany do zadań biznesowych. Miejsce, w którym przechowywane są bieżące i zapasowe nośniki z kopiami zapasowymi, powinno odpowiadać poziomem bezpieczeństwa miejscom, w których znajdują się same systemy informatyczne. Choć zwykły włamywacz być może nie zainteresuje się stosami nagrywanych dysków CD czy szafką pełną taśm do streamera, konkurencja z pewnością zachwyciłaby się możliwością łatwego dostępu do danych skopiowanych z systemów informatycznych. Dlaczego nie oszczędzić wszystkim czasu i wysiłku, i nie wysłać do nich pocztą całego kodu źródłowego czy bazy danych klientów?

Różne typy kopii zapasowych

Po uzasadnieniu znaczenia kopii zapasowych oraz przekazaniu podstawowych zasad ich przechowywania omówię strategię związaną z samym fizycznym wykonywaniem kopii zapasowych. Jak wspominałem poprzednio, wykonywanie kopii zajmuje sporo czasu i wiąże się z pewnymi, nieodłącznymi kosztami, takimi jak koszt nośników. Istnieje jednak kilka sposobów na zarządzanie tymi kosztami i ich minimalizację.

Wyróżniamy trzy podstawowe typy kopii zapasowych:

- ♦ kopie archiwalne, które stanowią kompletny obraz zawartości systemu plików w danym punkcie czasu;
- ♦ kopie przyrostowe, które odzwierciedlają zmiany dokonane w zawartości systemu plików od ostatniej kopii zapasowej;
- ♦ kopie migawkowe, które stanowią odzwierciedlenie plików lub katalogów w danym punkcie czasu.



Kopie migawkowe są stosowane najczęściej w zastosowaniach domowych, ponieważ zapisanie kopii aktualnie prowadzonych projektów, folderów z pocztą czy nawet całego katalogu domowego na kilka dysków CD-R lub DVD-R to stosunkowo szybkie i tanie rozwiązanie. Właściwie nie ma o czym się rozpisywać, ponieważ najczęściej stosowana jest tu metoda „przeciągnij i upuść”. W dalszej części tego podrozdziału skupię się zatem na klasycznym modelu wykonywania kopii zapasowych, polegającym na wykonywaniu kopii archiwalnych i przyrostowych. W dalszej części rozdziału wrócę do tematu kopii migawkowych, omawiając kilka przydatnych technik.

Kopie archiwalne są często nazywane archiwami lub pełnymi kopiami zapasowymi (ang. *full backup*). Ten sposób stanowi maksymalistyczną technikę wykonywania kopii, ponieważ archiwa z reguły zawierają wszystkie pliki i katalogi występujące w momencie wykonania archiwum danego systemu plików w komputerze. W idealnym świecie najlepiej byłoby za każdym razem wykonywać kopie archiwalne po prostu dlatego, że ta metoda gwarantuje, iż w przypadku codziennego wykonywania kopii, utracie ulegnie w najgorszym wypadku praca tylko jednego dnia, niezależnie od typu nieszczęścia, jaki spotka system informatyczny. Niestety, kopie archiwalne cechują się pewnymi znaczącymi wadami.

- ♦ Zajmują najwięcej czasu ze wszystkich form kopii zapasowych, ponieważ wymagają skopiowania każdego pliku i katalogu na każdym systemie plików systemu komputerowego.
- ♦ Ilość danych zachowanych w ramach kopii archiwalnej wiąże się z rozmiarem samej kopii, czyli zajmują one najwięcej miejsca na nośniku ze wszystkich form kopii zapasowych.
- ♦ Zajęcie większej ilości miejsca na nośniku kopii zapasowej wiąże się z kolei z faktem, że przechowywanie kopii zapasowych tego typu jest najbardziej kłopotliwe.
- ♦ Archiwa najlepiej wykonywać w okresie, gdy żaden użytkownik nie pracuje na komputerze. To skraca czas niezbędny na wykonanie kopii zapasowej (ponieważ użytkownik nie konkuruje o moc komputera z procesem wykonującym kopię), a ponadto gwarantuje spójność całej kopii, ponieważ nie występuje obawa, że część zapisanych plików pochodzi sprzed zmiany dokonanej przez użytkownika w czasie wykonywania kopii, a część po dokonaniu tej zmiany. Ten wymóg może wydać się niezbyt kłopotliwy z punktu widzenia systemu domowego, w biznesie uzyskanie pewności, że nikt nie korzysta z systemu (serwera) w okresie wykonywania kopii, może być zupełnie niewykonalne (szczególnie dla usług działających 24 godziny na dobę, 7 dni w tygodniu), a w najlepszym przypadku będzie wiązało się z obniżeniem komfortu pracy pracowników firmy i niezadowolaniem klientów.

Choć zalety archiwalnych kopii zapasowych w charakterze pełnego odwzorowania systemu mają zasadnicze znaczenie, wymienione wyżej problemy powodują, że pełne kopie zapasowe nie są wykonywane codziennie, zarówno w środowiskach domowych, jak i biznesowych czy w ramach organizacji. Kopie tego typu można wykonywać rzadziej, ale redukcja częstotliwości ich wykonywania zwiększa podatność na utratę znaczących ilości danych w przypadku awarii dysku lub pożaru.

W tym miejscu na ratunek przybywają przyrostowe kopie zapasowe. Jak wspomniałem wcześniej, kopie przyrostowe stanowią kopię wszystkich plików i katalogów, które uległy modyfikacji w systemie komputerowym od momentu dokonania poprzedniej kopii zapasowej. Jeśli pojawi się problem i wystąpi konieczność odtworzenia wszystkich plików

i katalogów z kopii, można najpierw odtworzyć kopię archiwalną, zawierającą pełną zawartość systemu plików, po czym wgrać odpowiednią ilość kopii przyrostowych. Kopie te w połączeniu z kopiami archiwalnymi cechują się następującymi zaletami.

- ♦ Pozwalają zminimalizować rozmiar nośników niezbędny do przechowywania kopii. Archiwa z reguły wymagają dużych ilości nośnika, natomiast kopie przyrostowe znacznie mniej, ponieważ nie zapisują tak wielkich ilości danych.
- ♦ Można je wykonać szybciej, ponieważ zapisują znacznie mniej danych, w porównaniu z kopiami archiwalnymi.
- ♦ Nośnik, na którym zapisana jest kopia przyrostowa, może mieć mniejsze rozmiary, ponieważ kopia przyrostowa jest mniejsza.
- ♦ Nawet w środowiskach biznesowych i akademickich kopie przyrostowe mogą być wykonywane również w okresie, gdy systemy komputerowe są w użyciu.

Inną ciekawą cechą kopii przyrostowych jest to, że zapisują zmiany w plikach i katalogach w komputerze od momentu wykonania poprzedniej kopii zapasowej, również przyrostowej. W środowiskach korporacyjnych, większość administratorów systemów, nośniki i procedury sporządzania kopii zapasowych porządkuje w następujący sposób.

- ♦ Archiwa są sporządzane rzadko, nawet raz na pół roku albo przed wprowadzeniem znaczących modyfikacji do systemu.
- ♦ Miesięczne kopie przyrostowe obejmują modyfikacje, jakie wystąpiły od wykonania ostatniego archiwum. Jeśli pozwala na to budżet i rozmiar nośnika, miesięczne kopie przyrostowe są przechowywane aż do momentu wykonania kolejnej kopii archiwalnej, po czym nośniki te mogą być użyte ponownie.
- ♦ Tygodniowe kopie przyrostowe obejmują wszystkie zmiany od ostatniej kopii miesięcznej. Nośniki te można używać od nowa w każdym miesiącu, po wykonaniu nowej kopii miesięcznej.
- ♦ Dienne kopie zapasowe obejmują zmiany od wykonania ostatniej kopii tygodniowej. Te nośniki mogą być wykorzystane ponownie po wykonaniu tygodniowej kopii przyrostowej. W niektórych instalacjach kopie dienne są wykonywane przyrostowo od ostatniej kopii dziennej lub od wybranego dnia tygodnia.

Nie istnieje taki system wykonywania kopii zapasowych, który pozwoliłby odtworzyć dowolny plik z dowolnego dnia tygodnia. Jeśli nawet ktoś ma takie szczęście (lub tyle samozaparcia), że codziennie wykonuje pełne kopie systemu, pliki utworzone lub zmodyfikowane po ostatniej kopii zapasowej nie mogą być odtworzone, również w przypadku plików starszych może być odtworzona wyłącznie taka wersja pliku, która została zapisana w kopii. Projektując schemat wykonywania kopii zapasowych oraz zależności między archiwami i kopiami przyrostowymi, należy zdecydować się na poziom szczegółowości, z jaką chcemy ewentualnie odtwarzać utracone pliki. Przykładowy, opisany wyżej ogólny schemat, wykorzystujący archiwa, kopie miesięczne, tygodniowe i dienne, nie gwarantuje możliwości odtworzenia wersji pliku nowszej niż poprzednie archiwum.

- ♦ Jeśli plik zostanie usunięty w dniu poprzedzającym rozpoczęcie cyklu kopii miesięcznej następującej po wykonaniu kopii archiwalnej, będzie dostępny w kopii archiwalnej oraz w kopiach tygodniowych maksymalnie przez jeden miesiąc.

Następnie nośnik z kopiami tygodniowymi zostanie nadpisany i najnowszą wersją pliku dostępną do odtworzenia będzie ta, która została zapisana na kopii archiwalnej.

- ♦ Jeśli plik zostanie usunięty jeden dzień przed rozpoczęciem cyklu kopii miesięcznej następującej po wykonaniu kopii archiwalnej, będzie dostępny w kopii archiwalnej oraz w kopiach miesięcznych przez okres maksymalnie siedmiu miesięcy. W tym momencie zostanie utworzone nowe archiwum, a taśma miesięczna nie będzie nadpisywana przez miesiąc, aż do zakończenia cyklu. Następnie taśma zawierająca miesięczną kopię przyrostową zostanie nadpisana i najnowszą wersją pliku dostępną do odtworzenia będzie wersja zapisana w kopii archiwalnej.

Wybór strategii wykonywania kopii zapasowych jest w zasadzie formą kalkulacji dotyczącej oczekiwanego okresu następującego między usunięciem ważnego pliku a zgłoszeniem jego utraty i prośby o odtworzenie z kopii. W tym przypadku należy uwzględnić uzgodniony poziom usług oraz koszt poszczególnych poziomów usług z punktu widzenia nośników, czasu wykonywania kopii oraz zarządzania nośnikami. Czasem zauważenie brakującego pliku następuje natychmiast, np. w sytuacji usunięcia ważnego pliku, nad którym użytkownik pracuje w danym momencie. W przypadku innych problemów, takich jak utrata plików w efekcie stopniowego występowania uszkodzeń dysku i błędów w systemie plików zauważenie problemu może zająć dłuższy czas.

Prawie wszystkie systemy wykonywania kopii zapasowych zapewniają automatyczną obsługę wykonywania kopii przyrostowych od ostatniego wykonania kopii archiwalnej. Program *dump*, który omówię w następnym podrozdziale, przypisuje różne liczby do różnych „poziomów” wykonania kopii zapasowej i śledzi poziomy wykonania tych kopii w oparciu o nazwę urządzenia, na którym zlokalizowany jest system plików.

Ważnym aspektem, który należy wziąć pod uwagę przy planowaniu strategii wykonywania kopii zapasowych, jest to, kiedy mają być wykonywane oraz jakie uprawnienia użytkowników są niezbędne do ich wykonania. Kopie zapasowe z reguły wykonują się najszybciej w czasie najmniejszego obciążenia systemu, gdy dodatkowo mamy największą szansę na to, że kopiowane pliki nie ulegną modyfikacji. W środowisku korporacyjnym istnieje duże prawdopodobieństwo, że kopie zapasowe będą obsługiwane przez większą grupę operatorów. W takim przypadku należy zastanowić się nad tym, w jaki sposób zapewnić tym operatorom odpowiedni zestaw uprawnień. Skuteczne skopiowanie solidnie zabezpieczonych plików lub wykorzystanie narzędzia do wykonywania kopii zapasowych pracującego na poziomie systemu plików z reguły wymaga uprawnień administratora (root). Wielu użytkowników wykorzystuje programy pozwalające nadawać uprawnienia konta root, takie jak *sudo* (które już znamy z Ubuntu), lub odpowiednie ustawienia znaczników *s* (*suid root*) na plikach binarnych, dzięki czemu programy zapisujące i odtwarzające kopie zapasowe nie wymagają znajomości hasła konta root, co jest bardzo ważne dla operatorów kopii zapasowych o niskim poziomie zaufania. Informacje na temat konfiguracji polecenia *sudo* w oparciu o grupy użytkowników można znaleźć w podrozdziale „Konfiguracja polecenia *sudo* w systemie Ubuntu” w rozdziale 21.

Weryfikacja i testowanie kopii zapasowych

Samo wykonywanie kopii zapasowych nie gwarantuje jeszcze tego, że możemy spać spokojnie, należy jeszcze upewnić się, że wykonywane kopie są czytelne i można bez problemów odczytać z nich pliki. Obecnie ten problem występuje rzadziej, ale zawsze istnieje niebezpieczeństwo, że głowice w napędzie taśmowym są źle ustawione. To może doprowadzić do sytuacji, że kopie dadzą się odczytać tylko na tym samym napędzie, na którym zostały wykonane, lub nie dadzą się odczytać w ogóle. Należy zweryfikować, czy kopie zapasowe dają się czytać i można z nich odzyskiwać dane również z urządzeń innych niż te, na których zostały zapisane. Nie ma potrzeby codziennego sprawdzania każdej taśmy, ale od czasu do czasu warto wykonać losowe sprawdzenie, choćby dla spokoju ducha. Taśmy w wyniku używania rozciągają się i zużywają, dlatego należy uwzględnić konieczność wymiany nośników po określonym czasie użytkowania.

Jednym z problemów typowych dla kopii zapasowych jest ograniczony okres żywotności nośnika w trybie „przechowywania”. Ta żywotność jest uzależniona od typu nośnika, środowiska, w którym jest przechowywany, oraz od szczęścia. Natomiast należy założyć, że nie istnieją nośniki o nieograniczonej żywotności, np. taśmy mogą wytrzymać wiele lat, ale też może okazać się, że stracą swoje właściwości po krótkim okresie przechowywania. Nośniki o długim okresie żywotności, takie jak dyski CD i DVD jednokrotnego zapisu, mają znaczące zalety, ale cechują się innymi problemami, o których wspomniałem pokrótce w ramce „Kilka słów o nośnikach”. Tego typu nośniki mogą być użyteczne do określonych typów kopii zapasowych, w zależności od tego, czy oprogramowanie obsługujące zapisuje kopie zapasowe na nośniku, wykorzystując jego system plików albo surowy dostęp do urządzenia (ang. *raw device*). Ponadto nie wiadomo dokładnie, jaką rzeczywistą żywotność mają te nośniki, ale z pewnością zajmują mniej miejsca niż praktycznie każdy typ taśm lub stos dysków twardych. Oprócz testowania sprawności nośników wykorzystywanych obecnie do wykonywania kopii zapasowych, należy okresowo sprawdzać sprawność nośników archiwalnych.



Pomijając fakt, że kopie zapasowe bywają zależne od urządzeń, na których są wykonane, należy uwzględnić dostępność odpowiednich urządzeń na potrzeby odtworzenia systemu. Tajemnicą poliszyneła jest fakt, że wiele ośrodków informatycznych, militarnych i rządowych szczytiło się posiadaniem gigantycznych kolekcji kopii zapasowych wykonanych na nieistniejących już urządzeniach, takich jak napędy o bardzo niskich prędkościach przesuwu lub siedmiościeżkowe taśmy jednocalowe. Jeśli nawet takie urządzenie istnieje, dane z reguły i tak nie dają się odczytać, ponieważ zostały zapisane w jakimś przestarzałym, niestandardowym formacie. Gdy system komputerowy jest już przestarzały i w związku z tym ulega likwidacji, często zapomina się o podjęciu decyzji, czy dane z jego kopii zapasowych będą kiedykolwiek potrzebne. Przy odrobinie szczęścia w nowym systemie uda się odczytać stare archiwa, aby ponownie zapisać je na nowszym nośniku, wykorzystując nowszy format kopii zapasowych. Jeśli tego szczęścia zabraknie, zostaniemy ze sporą ilością dużych, nieporęcznych przycisków do papieru przypominających w nieskończoność o tej niefortunnej sytuacji.

Decyzja o tym, co kopiować

Oprócz zagadnień związanych z obniżaniem kosztów, takich jak użycie nośników o wyższej gęstości zapisu (np. dysków CD-R), sposobem na zmniejszenie ilości nośników niezbędnych do przechowywania kopii oraz minimalizację czasu ich wykonania jest dostosowanie

strategii wykonania kopii zapasowych do typu kopiowanych danych. Przykładowo, oprogramowanie systemowe ulega modyfikacjom sporadycznie, zatem partycje zawierające system operacyjny można kopiować wyłącznie w momencie wykonywania kopii archiwalnych. Nawet oprogramowanie tworzone lokalnie nie ulega częstym modyfikacjom, zatem warto zastanowić się, czy nie wystarczy wykonywać kopii tych katalogów raz w tygodniu zamiast codziennie. Na palcach jednej ręki (po amputacji czterech palców) mogę policzyć przypadki, gdy byłem zmuszony odzyskać z kopii zapasowej starszą wersję aplikacji. Jednak może okazać się, że ktoś będzie miał mniej szczęścia ode mnie.

Utrzymywanie kopii zapasowych systemu operacyjnego oraz jego aplikacji to ważne zagadnienie i z pewnością ma kluczowe znaczenie, gdy trzeba odzyskać cały system (*od zera*). Jeśli jednak w grę wchodzi częstotliwość wykonywania kopii zapasowych (dzięki doskonałemu Menedżerowi aktualizacji), z reguły wystarczy zachować nośniki instalacyjne (a nawet ponownie pobrać je z internetu, gdy przyjdzie taka potrzeba) i skorzystać z nich do instalacji systemu. Jeśli jednak system wykorzystuje samodzielnie skompilowane jądro albo specjalne moduły, należy zawsze upewnić się, czy istnieje kopia zapasowa bieżącej konfiguracji i wszystkich jej plików konfiguracyjnych (czyli np. katalogu */etc* z całą zawartością) opisujących stan systemu. Ten wysiłek z pewnością opłaci się, gdy pewnej ponurej nocy stanie w płomieniach dysk twardy zawierający misternie skonfigurowany system operacyjny i aplikacje.

Zagadnienia omawiane w pierwszych podrozdziałach tego rozdziału przyprawiają wielu administratorów o bóle głowy. Utrata krytycznych danych jest bardzo bolesna nawet w przypadku, gdy dane te były ważne tylko dla jednej osoby. Obmyślanie, projektowanie i implementacja rozsądnej polityki kopii zapasowych, terminarzy i planów odtwarzania systemu po katastrofach to ważne zadania, niezależnie od tego, jak wiele osób może być dotkniętych problemem utraty danych. Kopie zapasowe są jak ubezpieczenia — mamy nadzieję, że nigdy z nich nie skorzystamy, ale lepiej mieć je na podorędziu.

Oprogramowanie dla Linuksa do wykonywania kopii zapasowych

Dla systemu Ubuntu istnieje wiele narzędzi do sporządzania kopii zapasowych. Większość z nich to tradycyjne narzędzia obsługiwane z wiersza poleceń, służące do wykonywania plików z archiwami lub do zapisywania tych plików na nośnikach w różnych formatach. Istnieją również interesujące narzędzia wyposażone w graficzny interfejs użytkownika.

W kilku kolejnych podrozdziałach omawiam najczęściej spotykane narzędzia na licencji open source, wykorzystywane do wykonywania kopii zapasowych w systemach Linux. Narzędzia te pogrupowałem, w zależności od tego, czy mają działanie lokalne, czy też stanowią rozwiązania sieciowe. Jak wspominałem w poprzednim podrozdziale, ważnym wymogiem solidnej strategii bezpieczeństwa kopii zapasowych jest przechowywanie ich z dala od systemów źródłowych. We współczesnych konfiguracjach sieciowych tego typu konfigurację można zaimplementować na jeden z dwóch sposobów: zapisując dane lokalnie na nośniku, który jest następnie transportowany w oddalone miejsce przechowywania, lub przez zastosowanie sieciowego mechanizmu wykonywania kopii zapasowych, zapisującego kopie w systemach fizycznie ulokowanych w innym miejscu niż system źródłowy.

Komercyjne rozwiązania dla Linuksa

Komercyjni dostawcy aplikacji stopniowo zaczynają dostrzegać również rynek linuksowy, a zwłaszcza konieczność tworzenia aplikacji do wykonywania kopii zapasowych. Dzieje się tak, ponieważ Linux w charakterze platformy serwerowej stał się bardzo popularny. Moim zdaniem, komercyjne rozwiązania z dziedziny kopii zapasowych są zbędne w środowisku domowym, ale warto rozważyć je w przypadku wykorzystania Ubuntu w wielosystemowym środowisku korporacyjnym lub akademickim. Zanim jednak ktoś zdecyduje się na inwestycję w komercyjne rozwiązania instalowane w wolno dostępnym systemie operacyjnym, warto zastanowić się, jak krytyczne są dane, które będą zabezpieczane na kopiach. Ta książka nie omawia komercyjnych rozwiązań do wykonywania kopii zapasowych, jedynie pokrótce wspomnę o kilku producentach takich rozwiązań i ich produktach.

- ♦ Arkeia Software: aplikacje Network Backup i Smart backup to dobrze znane rozwiązania międzyplatformowe, obsługujące również systemy Microsoft Windows, dostępne dla wielu komercyjnych dystrybucji Linuksa, takich jak Red Hat, SUSE, czy Mandrake. Więcej informacji ma temat tych produktów można znaleźć na stronie <http://www.knox-software.com>.
- ♦ Lone Star Software: aplikacja LONE-TAR jest dostępna na rynku od ponad dziesięciolecia, oferuje rozwiązanie do wykonywania kopii zapasowych dla Linuksa i innych systemów uniksowych. Aplikacja ta posiada również interfejs graficzny dla systemu X Window, LTX. Więcej informacji ma temat tego produktu można znaleźć na stronie <http://www.cactus.com>.
- ♦ MicroLITE: aplikacje BackupEDGE i RecoverEDGE to pakiet narzędzi o dużej wydajności działania, obsługujący lokalne i zdalne kopie zapasowe oraz oferujący ciekawe możliwości odtwarzania systemów komputerowych od zera. Więcej informacji ma temat tego produktu można znaleźć na stronie <http://www.microlite.com>.
- ♦ Storix: system Backup Administrator to graficzne narzędzie dla Linuksa (i AIX-a), obsługujące również opcję wykonywania kopii zapasowych z systemów Windows za pośrednictwem protokołu SMB. Dla Linuksów narzędzie to oferuje również ciekawe możliwości odtwarzania systemu od zera. Więcej informacji ma temat tego produktu można znaleźć na stronie <http://www.storix.com>.
- ♦ Symantec/Veritas: aplikacja Backup Exec, napisana przez Veritas, obecnie będąca własnością firmy Symantec, to znane rozwiązanie do wykonywania kopii zapasowych dla systemów Microsoft Windows, dostępne również dla Linuksa. Więcej informacji ma temat tego produktu można znaleźć na stronie <http://www.symantec.com/enterprise/veritas>.

Żaden z tych producentów nie ogłosił dotychczas oficjalnego wsparcia dla dystrybucji Ubuntu, ale wszyscy potwierdzają zgodność z systemem Linux, jestem więc pewny, że w przypadku zainteresowania tymi produktami warto podjąć dyskusję z producentami lub dystrybutorami na temat tego, czy będą działać pod Ubuntu.

Oprogramowanie dla Linuksa do lokalnego wykonywania kopii zapasowych i ich odtwarzania

Korzenie narzędzi systemowych dla Linuksa leżą w systemie Unix, nie powinno więc dziwić, że również klasyczne, uniksowe narzędzia do wykonywania kopii zapasowych są dostępne we wszystkich dystrybucjach Linuksa. Niektóre z nich już nie są w stanie ukryć swojego wieku, ale jedno jest pewne, były używane przez lata i gwarantują przenośność kopii zapasowych między różnymi wersjami systemu Linux.

Do klasycznych narzędzi do wykonywania kopii zapasowych dla Linuksa i Uniksa należą programy, wymienione niżej w kolejności alfabetycznej.

- ♦ **cpio**: program ten (jego nazwa to skrót od określenia *copy input to output*, czyli kopiowanie z wejścia na wyjście) został napisany z myślą o wykonywaniu kopii zapasowych. Program na standardowym wejściu oczekuje listy plików, które mają być skopiowane, a wynik (archiwum) zapisuje na standardowym wyjściu (zapis na urządzeniu jest możliwy przez zwykłe przekierowanie w ramach powłoki). Program **cpio** może być używany z systemami plików dowolnego typu, ponieważ działa na poziomie systemu plików i nie posiada żadnych mechanizmów specyficznych dla struktur danych systemu plików.
- ♦ **dd**: Jego nazwa pochodzi od określenia *dump device*, czyli zrzut urządzenia, i dokładnie to robi: zrzuca zawartość jednego urządzenia i zapisuje na inne. Program **dd** nie rozumie, co to są systemy plików, poziomy zrzutów czy poprzednie wywołania programu; po prostu odczytuje dane ze źródła i zapisuje je w innym miejscu. Danymi można operować „w drodze”, wykonując znane zabiegi, takie jak konwersja znaków ASCII na format EBCDIC. Program **dd** potrafi skopiować całą zawartość urządzenia, taką jak partycja dyskowa, np. na taśmę. Narzędzie to w zasadzie nie zostało zaprojektowane na potrzeby wykonywania kopii zapasowych, ale istnieją sytuacje, w których **dd** jest doskonałym narzędziem. Przykładowo za pomocą **dd** można przekopiować jedną partycję na inną w przypadku wykrycia problemów z dyskiem twardym, można użyć go do tworzenia kopii partycji na potrzeby klonowania dysków lub wykorzystać na potrzeby aplikacji zapisującej i odczytującej surowe partycje dyskowe, których zawartość kopiuje się na zasadzie „wszystko albo nic”. Program **dd** czyta bezpośrednio z urządzenia, przez co nie rozróżnia typów systemów plików, zatem niemożliwe jest odtwarzanie pojedynczych plików z kopii zapasowych utworzonych tym sposobem. Należy wówczas odtworzyć całą partycję i dopiero z niej odczytać odpowiednie pliki.
- ♦ **dump** i **restore**: te programy zostały zaprojektowane do wykorzystania w parze: jeden z nich zapisuje kopie zapasowe, a drugi służy do ich odtwarzania. Narzędzia są dostępne dla systemów Unix od wersji 6. Choć polecenia **cpio** i **tar** łączą możliwości zapisu i odtwarzania kopii zapasowych, a polecenie **dd** nie potrafi odtworzyć z kopii pojedynczych plików, program **dump** jedynie tworzy kopie zapasowe, a program **restore** potrafi tylko odtwarzać z nich pliki i katalogi. Zarówno program **dump**, jak i program **restore** działają na poziomie struktur systemu plików, z tego powodu można ich używać do kopiowania i odtwarzania plików wyłącznie w systemach plików **ext2** i **ext3** (w czasie pisania tej książki). Jednak **dump** i **restore** posiadają cechę szczególną: potrafią zapisać kopię i odtworzyć każdy rodzaj pliku, zapisanego na partycji **ext2** lub **ext3**, dotyczy to zarówno plików urządzeń, jak i plików typu **sparse** (tzn. że przy wykonywaniu kopii zapasowej zachowują również specjalne właściwości tych plików). Programy **dump** i **restore** mogą być używane do wykonywania kopii zapasowych całych systemów plików, choć mają wbudowaną obsługę kopii przyrostowych, tzn. zachowują informacje na temat tego, jakie systemy plików zostały skopiowane i który poziom kopii zapasowej został wykonany w danym systemie plików. Informacje tego typu są zapisywane w łatwym do odczytu pliku */etc/dumpdates*. Archiwa tworzone za pomocą programu **dump** mogą być automatycznie dzielone na wiele taśm lub innych nośników (gdy program napotka koniec lub zapełni cały nośnik), można też wskazać programowi **dump** rozmiar nośnika, aby wymusić mechanizm podziału kopii zapasowej na części. Najciekawszą cechą programu **restore** jest możliwość uruchomienia go w trybie interaktywnym, w którym można odczytywać informacje

bezpośrednio z taśmy i przeglądać hierarchię plików katalogów zapisanych w archiwum. Do przeglądania tej struktury służą standardowe polecenia, takie jak `cd`. W tym trybie można również zaznaczać pliki i katalogi do odtworzenia.



Programy `dump` i `restore` nie są instalowane w ramach standardowej instalacji Ubuntu, można je jednak z łatwością zainstalować za pomocą polecenia `apt-get` (znajdują się w pakiecie o nazwie *dump*).

- ♦ `tar`: to prawdopodobnie najbardziej znany i najszerzej wykorzystywany program do wykonywania kopii zapasowych w systemach Unix. Jego nazwa pochodzi od określenia *tape archiver* (archiwizator taśmowy), ale znajomość pochodzenia nazwy może być nieco myląca. Program wymaga określenia nazw plików i katalogów, które mają zostać zapisane w archiwum. Pliki te zostają zapisane na urządzeniu wyjściowym lub na standardowym wyjściu. Wersja programu `tar` na licencji GNU (znana wcześniej jako `gtar`) zawiera znaczące różnice, w porównaniu z oryginalną wersją pochodzącą z systemów Unix. To jeszcze jeden zdumiewający przykład możliwości Free Software Foundation. GNU `tar` zawiera bowiem funkcje wykraczające poza możliwości klasycznego programu `tar`, w tym wbudowaną możliwość zapisu i odczytu skompresowanych archiwów `tar` (za pomocą programu `gzip`), obsługę kopii przyrostowych, obsługę wielowoluminowych archiwów i wiele innych. Program `tar` jest niezależny od systemu plików i potrafi odczytywać pliki i katalogi, niezależnie od niskopoziomowych struktur danych. Program `tar` jest z pewnością najbardziej popularnym programem do archiwizacji danych dla systemów Linux, wykorzystywanym w praktycznie wszystkich mechanizmach zarządzania pakietami, np. formaty archiwów RPM zawierają archiwa `tar` skompresowane programem `gzip`, a „czyste” archiwa `tar` spakowane programem `gzip` (pliki z rozszerzeniami `.tgz` lub `.tar.gz`) są wykorzystywane do rozpowszechniania większości pakietów z kodami źródłowymi jądra Linuksa i jego aplikacji.

Narzędzia omówione w tym punkcie tworzą archiwa lokalnie lub zapisują te archiwa na lokalnych urządzeniach pamięci masowej. Oczywiście, w przypadku wykorzystania systemów operacyjnych z obsługą sieci, jakim jest Ubuntu Linux, pojęcie „lokalny zasób” uwzględnia wszelkie typy zasobów pozornie lokalnych w systemie, lecz dostępnych za pośrednictwem sieci, takich jak zasoby sieciowe zamontowane w systemie lokalnym. Do przykładów tego typu zasobów sieciowych zamontowanych lokalnie należą katalogi udostępnione NFS czy też zasoby sieciowe montowane za pośrednictwem mechanizmu Samba. Katalogi sieciowe zamontowane lokalnie pozwalają zintegrować zdalne zasoby z lokalnymi mechanizmami wykonywania kopii zapasowych i umożliwiają:

- ♦ wykonywanie kopii zapasowych zdalnych katalogów do archiwów lokalnych przez zamontowanie tych zasobów w systemie lokalnym i uwzględnienie ich w wykonywanych kopiach zapasowych,
- ♦ zapisywanie kopii zapasowych na zdalnych zasobach przez tworzenie plików z archiwami w katalogach sieciowych zamontowanych w systemie lokalnym.

Oba te scenariusze dzięki zastosowaniu katalogów sieciowych spełniają wymóg wykonywania kopii zapasowych z dala od plików źródłowych.

Oprogramowanie dla systemu Linux do wykonywania sieciowych kopii zapasowych

Narzędzia omówione w poprzednim punkcie tworzą pliki archiwów na lokalnych urządzeniach zasobów pamięci masowej (lub zasobach sprawiających wrażenie lokalnych). Programy omówione w tym punkcie są nieco inne; z założenia zostały oparte na koncepcji sieciowej, a przez to pozwalają tworzyć i zarządzać kopiami zapasowymi na zdalnych systemach.

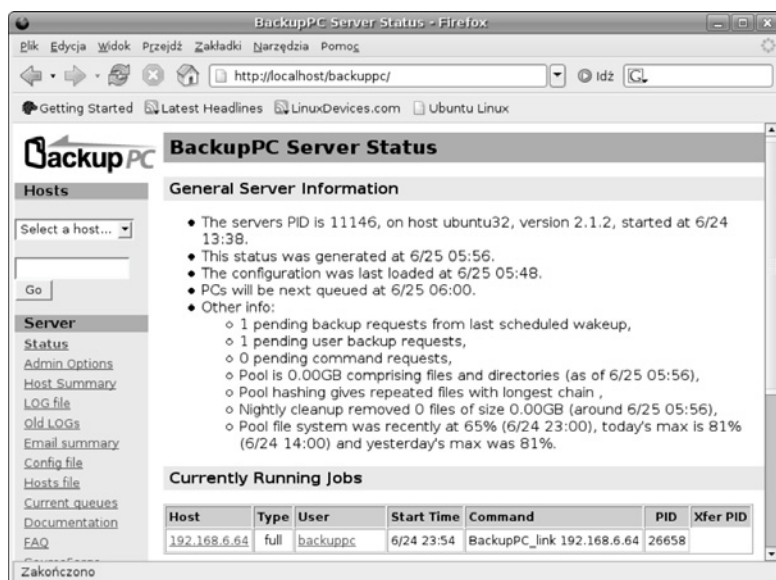
Poniższa lista zawiera sieciowe systemy najczęściej stosowane do wykonywania kopii zapasowych, które są dostępne dla Ubuntu. Jest ich o wiele więcej, można je znaleźć za pomocą menedżera pakietów Synaptic — w tym celu należy wyszukać ciąg znaków *backup* w kategorii *Nazwa i opis*. Wymienione niżej należą do moich ulubionych.

♦ **Amanda:** *Advanced Maryland Automated Network Disk Archiver* to system do zarządzania kopiami zapasowymi na licencji open source, napisany na uniwersytecie Maryland na początku lat 90. z myślą o systemach uniksowych. Za pomocą Amandy wykonywanie kopii zapasowych z wielu stacji klienckich na jeden serwer centralny to dość proste zadanie; obsługuje wykonywanie kopii zapasowych za pomocą Samby i zawiera pełny system zarządzania kopiami zapasowymi dla Ubuntu. Amanda pozwala definiować wiele zbiorów kopii zapasowych o różnych ustawieniach, wykonuje kopie na dysku i taśmach, umożliwia śledzenie poziomów kopii zapasowych na systemach klienckich i przechowuje wyczerpujące dzienniki systemowe ułatwiające diagnostykę i rozwiązywanie problemów. Komunikacja między klientami a serwerem Amandy może być szyfrowana, co dodatkowo zwiększa ich bezpieczeństwo. Amanda nie jest domyślnie instalowana w Ubuntu, ale jest dostępna w repozytoriach, dzięki czemu można ją z łatwością zainstalować, wykorzystując program Synaptic, `apt-get` lub `aptitude`. Amanda składa się z dwóch pakietów: *amanda-server* i *amanda-client*. Strona domowa projektu to <http://www.amanda.org>.

♦ **BackupPC:** to interesujący system wykonywania kopii zapasowych wykorzystujący interfejs użytkownika obsługiwany z przeglądarki WWW. BackupPC umożliwia wykonywanie kopii zapasowych systemów zdalnych z wykorzystaniem `smb`, `tar` oraz `rsync`. Na rysunku 22.1 przedstawiam stronę WWW przykładowego serwera BackupPC. System ten tworzy kopie zapasowe zdalnych systemów, które są zapisywane na serwerze BackupPC, pozwala autoryzowanym użytkownikom odtwarzać ich własne pliki z tych archiwów, co może stanowić rozwiązanie jednego z ważniejszych problemów nękających wielu administratorów. Konfiguracja każdego klienta jest zapisana na serwerze BackupPC, co pozwala wykonywać kopie różnych systemów z wykorzystaniem różnych poleceń i protokołów i z łatwością identyfikować zdalne katalogi i systemy plików, które mają być skopiowane. Jedną ze szczególnie ciekawych cech systemu BackupPC jest to, że do tworzenia kopii wykorzystuje standardowe polecenia Linuksa wywoływane na serwerze, dzięki czemu nie wymaga instalacji żadnego oprogramowania, choć w przypadku niektórych systemów klienckich potrzebne są pewne modyfikacje.

Strona domowa projektu BackupPC to <http://backuppc.sourceforge.net>. Informacje na temat instalacji, konfiguracji i wykorzystania systemu BackupPC można znaleźć w dalszej części rozdziału.

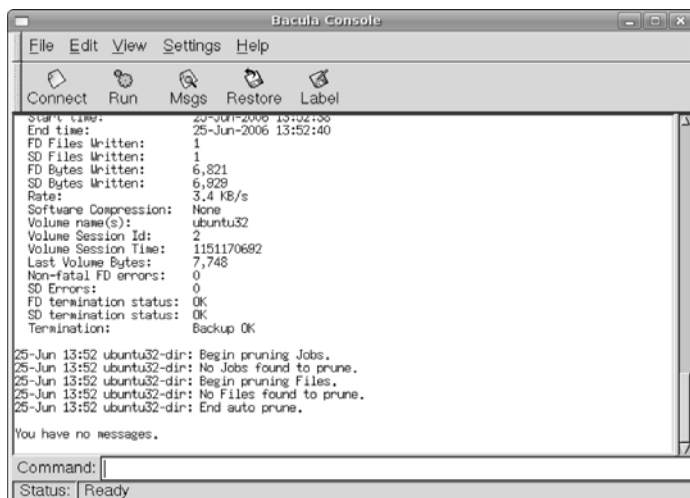
Rysunek 22.1.
Strona interfejsu
systemu BackupPC



- ♦ **Bacula:** niezwykle rozbudowany zestaw programów tworzących skalowany system wykonywania i odtwarzania kopii zapasowych dla systemów Linux, Unix i Microsoft Windows. Możliwości i elastyczność Baculi z powodzeniem konkurują z systemem Amanda, ale Bacula jest znacznie elastyczniejsza pod względem tego, gdzie i w jaki sposób zapisywane są kopie zapasowe. Bacula nie jest domyślnie instalowana w Ubuntu, ale jest dostępna w repozytoriach i można ją z łatwością zainstalować za pomocą programów Synaptic, apt-get czy aptitude. Bacula ma duże możliwości, ale też jest dość skomplikowanym systemem. Administratorzy zainteresowani wykorzystaniem Baculi na początek powinni zainstalować pakiet *bacula-doc* i przeczytać dokumentację, aby stwierdzić, czy to rozwiązanie im odpowiada. Bacula to głównie system oparty na narzędziach obsługiwanych z wiersza poleceń, ale posiada także konsolę graficzną, prezentowaną na rysunku 22.2, która jest w zasadzie nakładką na narzędzia obsługiwane z wiersza poleceń. Strona domowa projektu Bacula to <http://www.bacula.org>.
- ♦ **rsync (remote sync):** to program do synchronizacji plików i katalogów obsługiwany z wiersza poleceń, ułatwiający kopiowanie plików i katalogów z jednego systemu na drugi. Gdy istnieje lokalna i zdalna kopia tego samego pliku lub katalogu, rsync zawiera mechanizm wyrównania różnic między wersjami. Protokół zdalnej synchronizacji wykorzystywany przez program rsync pozwala wykonywać kopie zapasowe przez przesłanie tylko różnic między wersjami. Program rsync jest automatycznie instalowany z Ubuntu, ale wymaga dodatkowej konfiguracji, jeśli chcemy wykorzystać go do wykonywania kopii zapasowych zdalnych systemów.

Rysunek 22.2.

Konsola programu
Bacula w środowisku
GNOME



Wykonywanie kopii zapasowych na lokalnych nośnikach wymiennych

Podrozdział, zatytułowany „Podstawowe zagadnienia dotyczące kopii zapasowych”, zawierał wprowadzenie koncepcji kopii zapasowych, z których wiele sprawia wrażenie mało praktycznych w zastosowaniach domowych. Jednak to, czy są praktyczne, czy też nie, zależy przede wszystkim od problemów, jakie spodziewamy się rozwiązać za pomocą kopii zapasowych. Jeśli główny cel leży w zabezpieczeniu przed awariami dysków lub przypadkowym usunięciem ważnych plików, można odpuścić sobie wykonywanie kopii archiwalnych i kopii przyrostowych. Wystarczy wykonywać kopie migawkowe ważnych plików i katalogów na dysk CD-R lub DVD-R. Analogicznie, jeśli nie ma potrzeby odtworzenia pliku w dowolnym punkcie czasu, a wystarczą solidne kopie plików, ponownie w zupełności wystarczą kopie migawkowe wybranych plików i katalogów, wykonywane z dogodną częstotliwością. Jeśli nie przejmujemy się tym, że kopie zapasowe zginą w przypadku pożaru mieszkania lub domu, nie ma potrzeby przechowywania kopii zapasowych z dala od komputera źródłowego.

Chodzi przede wszystkim o to, że ja nie jestem w stanie stwierdzić, jakie rozwiązanie będzie najlepsze dla czytelnika. W kolejnych kilku punktach omawiam sposoby wykorzystania narzędzi wymienionych wcześniej w tym rozdziale (włącznie ze standardowym poleceniem `cp`) do wykonywania kopii zapasowych ważnych plików.



Do użytku domowego najpopularniejszą metodą wykonywania kopii zapasowych jest przeciągnięcie myszą plików i katalogów na nośnik CD-R lub DVD-R. Drugą pod względem popularności metodą jest wykorzystanie dysków twardych przypinanych za pomocą złącza USB lub FireWire. Zaletą takiego rozwiązania polega na tym, że mamy do dyspozycji większy zasób na kopie (chyba że wykorzystywane są dyski naprawdę niewielkich rozmiarów), w porównaniu z dyskami CD lub DVD, i pozwala przechowywać kilka kopii zapasowych lub tylko jedną. Wadą zewnętrznych dysków twardych jest to, że są bardziej kosztowne w porównaniu z dyskami CD-R lub DVD-R i są nieco bardziej kłopotliwe w transporcie w przypadku przechowywania nośnika z dala od komputera źródłowego.

Archiwizacja i przywracanie danych z użyciem programu tar

Program tar jest jednym z najstarszych i najbardziej klasycznych narzędzi do wykonywania kopii zapasowych w systemach Unix i Linux. Potrafi zapisywać kopie zapasowe bezpośrednio na nośniku urządzenia, takim jak napęd taśmowy, ale najczęściej jest używany do tworzenia plików archiwów, np. pakietów źródłowych, które można z łatwością rozpowszechniać. Polecenie GNU tar, będące odmianą klasycznego polecenia dla systemów Unix, opracowaną specjalnie dla Linuksa, posiada dodatkowe możliwości konwersji, włącznie z możliwością kompresji archiwów w locie. Skompresowane archiwa tar mają najczęściej rozszerzenie *.tgz* lub *.tar.gz*, co sugeruje, że należy je zdekompresować przed rozpakowaniem za pomocą polecenia *gzip*. Obsługiwane jest też rozszerzenie *.tar.bz2*, które informuje, że do kompresji został użyty program *bzip2*. Pliki archiwów utworzone za pomocą programu tar określa się nazwą archiwów tar (po angielsku to *tarball*, czyli kula smoły).



Program tar jest już bardzo wiekowy, zatem przy definiowaniu jego argumentów należy zachować uwagę, ponieważ w przypadku niektórych z nich, kolejność ma kluczowe znaczenie.

Tworzenie archiwum to proste zadanie, np. w celu utworzenia archiwum o nazwie *home_dir_backup.tgz* zawierającego podkatalogi katalogu */home* wywołamy następujące polecenia:

```
$ cd /home
$ sudo tar czvf /tmp/home_dir_backup.tgz *
```



Plik archiwum należy zapisać poza drzewem katalogów będących obiektem kopii zapasowej. Utworzenie kopii zapasowej w katalogu, który jest kopiowany, spowoduje, że program utworzy w kopii kopię samego pliku kopii, co spowoduje błąd wykonania programu, a przede wszystkim doprowadzi do zmarnowania gigantycznych ilości miejsca.

Program tar obsługuje następujące opcje.

- ♦ **c** — utworzenie nowego archiwum. Jeśli plik o podanej nazwie istnieje, zostanie nadpisany, a jego oryginalna zawartość będzie utracona.
- ♦ **z** — wykonanie kompresji archiwum z wykorzystaniem algorytmu programu *gzip*.
- ♦ **v** — tryb "gadatliwy", powodujący wypisywanie nazwy każdego pliku zapisywanego w archiwum.
- ♦ **f** — wynik działania programu zostanie zapisany we wskazanym pliku, którego nazwa musi wystąpić jako pierwszy argument po opcji **f**. W naszym przypadku wynik działania polecenia zostanie zapisany w pliku */tmp/home_dir_backup.tgz*.

Po wywołaniu powyższego polecenia pojawi się duża liczba komunikatów, a po zakończeniu zostanie utworzony plik */tmp/home_dir_backup.tgz*, zawierający kompletną, rekurencyjną kopię zawartości katalogu */home*. Ten plik można skopiować na nośnik zewnętrzny, taki jak CD lub DVD, lub na zewnętrzny dysk twardy.

Po uzyskaniu archiwum tar z wybranego zbioru katalogów można utworzyć kolejne archiwum zawierające pliki i katalogi, które uległy zmianie po ustalonej dacie (np. po dacie utworzenia pierwszego archiwum). W tym celu posłużymy się następującym poleceniem:

```
$ cd /home
$ sudo tar czvf /tmp/home_dir_backup.tgz * --newer "2006-06-23"
```

To polecenie również wypisuje dużą liczbę komunikatów, nawet bez opcji `-v`, co przy pierwszym zetknięciu może być nieco niepokojące. Jest to efekt uboczny pewnej cechy archiwów tar. Nawet z opcją `--newer` nagłówek archiwum musi zawierać pełną informację na temat struktury zapisywanych katalogów. Jest to niezbędne, aby program tar był w stanie odtworzyć pliki i katalogi zachowując właściwą strukturę. Innymi słowy, jeśli została użyta opcja `--newer`, program tar odtworzy tylko te katalogi ze struktury, które zawierają pliki zmienione po wskazanej dacie.

Tworzenie archiwów na niewiele się zda, jeśli nie można odtworzyć zapisanych w nich plików. Z archiwum tar dane można odtwarzać na różne sposoby.

- ♦ Odtworzenie kompletnej zawartości. Poniższy przykład odtworzy całą zawartość archiwum tar o nazwie *home_dir_backup.tgz*, w katalogu, z którego zostało wywołane to polecenie, odtwarzając pełną strukturę podkatalogów:

```
$ sudo tar xzvf home_dir_backup.tgz
```

- ♦ Odtworzenie jednego lub większej liczby katalogów, a co za tym idzie, automatyczne rekurencyjne odtworzenie struktury ich podkatalogów. Poniższy przykład spowoduje odtworzenie katalogu *Ubuntu_Bible* oraz wszystkich jego podkatalogów i plików, zapisanych w archiwum *home_dir_backup.tgz*, w katalogu, z którego zostało wywołane to polecenie, odtwarzając pełną strukturę podkatalogów:

```
$ sudo tar xzvf home_dir_backup.tgz Ubuntu_Bible
```

- ♦ Odtworzenie jednego lub większej liczby wskazanych plików, wraz z drzewem katalogów, w którego strukturze pliki te są zapisane. Poniższy przykład spowoduje odtworzenie pliku *chap22.txt* z archiwum *home_dir_backup.tgz*, zawierającego katalog *Ubuntu_Bible*, w katalogu, z którego zostało wywołane to polecenie, odtwarzając pełną strukturę katalogów nadrzędnych:

```
$ sudo tar xzvf home_dir_backup.tgz Ubuntu_Bible/chap22.txt
```

Bardziej szczegółowe informacje na temat polecenia tar można znaleźć w jego podręczniku systemowym (`man tar`). Jako jedno z najstarszych poleceń systemów Unix i Linux, tar przez lata zgromadził bardzo długą listę opcji wywołania, z których większości zapewne nie uda się wykorzystać. Jednak opcje wywołania z wiersza poleceń są jak naboje: każda ilość może się przydać.

Wykonywanie aktualnych kopii katalogu lokalnego z użyciem polecenia cp

Jeśli wykonujemy jedynie kopię zapasową jednego katalogu i przede wszystkim staramy się zachować kopie plików, nad którymi pracujemy, z reguły najprościej zapisywać kopie tych katalogów na nośnikach wymiennych. Tradycyjne polecenie `cp` systemów Unix i Linux

obsługuje opcje ułatwiające wykonywanie kopii wskazanego katalogu oraz systematyczne aktualizowanie tylko tych plików, które uległy zmianie lub nie istnieją jeszcze w kopii. Przykładowo w celu wykonania kopii zapasowej wszystkich podkatalogów zapisanych w `/home` na nośniku zamontowanym w katalogu `/media/LACIE` (LACIE to popularna marka dysków twardych USB) posłużymy się następującym wywołaniem:

```
$ sudo cp -dpRuvx /home /media/LACIE/home
```

W tym wywołaniu zastosowaliśmy następujące opcje polecenia `cp`:

- ♦ `d` — nie śledź dowiązań symbolicznych, tzn. mają być kopiowane jako dowiązania symboliczne zamiast kopiowania obiektów przez nie wskazywanych;
- ♦ `p` — zachowaj w kopii uprawnienia i własności do oryginalnego pliku;
- ♦ `R` — rekurencyjne kopiowanie wskazanego katalogu;
- ♦ `u` — kopiowanie plików wyłącznie wtedy, kiedy oryginał jest nowszy od kopii lub gdy kopia nie istnieje;
- ♦ `v` — wypisuje informacje o każdym kopiowanym pliku (tę opcję można pominąć, ale jest przydatna przy śledzeniu postępów wykonania, przynajmniej przy pierwszych kilku wywołaniach polecenia);
- ♦ `x` — nie przechodź do innych systemów plików (wtedy, gdy kopiowane są katalogi zawierające punkty montowania).

Po wywołaniu tego polecenia w podkatalogu `/media/LACIE/home` powstanie kopia katalogu `/home`. Następnie można odłączyć dysk zewnętrzny i przechować go w bezpiecznym miejscu (najlepiej z dala od kopiowanego komputera). Za każdym razem gdy pojawi się potrzeba aktualizacji kopii, wystarczy podłączyć napęd i wykonać dokładnie to samo polecenie.

Wykonywanie aktualnych kopii katalogu lokalnego z użyciem polecenia `rsync`

Jak wspominałem wcześniej, polecenie `rsync` jest powszechnie wykorzystywanym narzędziem pozwalającym na wymianę plików ze zdalnym systemem. Program `rsync` musi zostać skonfigurowany w systemie zdalnym, zanim będzie można przystąpić do wymiany plików lub katalogów.

Aby wykorzystać program `rsync` w systemie Ubuntu, należy uaktywnić w nim opcję automatycznego uruchamiania demona `rsyncd`, dodatkowo trzeba zmodyfikować nieco plik konfiguracyjny, dopisując w nim określone katalogi, które mają być dostępne do zdalnego odczytu i zapisu. W tym celu należy otworzyć plik `/etc/default/rsync` w edytorze tekstu:

```
$ sudo emacs /etc/default/rsync
```

W wierszu definiującym zmienną `RSYNC_ENABLE` należy zmienić wartość `false` na `true`, po czym zapisać plik. Następnie, jeszcze przed uruchomieniem demona `rsyncd`, trzeba utworzyć plik konfiguracyjny.



Większość systemów Linux wykorzystuje menedżer usług internetowych, taki jak `inetd` lub `xinetd`, zarządzający obsługą przychodzących żądań związanych z usługami `ftp`, `tftp`, `rsync` czy `vnc`. Menedżery usług internetowych są dostępne również w Ubuntu, ale nie są domyślnie instalowane. W Ubuntu uruchomieniem demona usługi `rsync` zajmuje się skrypt rozruchowy `/etc/init.d/rsync`. Jeśli ktoś zechce zainstalować `xinetd` i wykorzystać go do zarządzania zadaniami `rsync`, należy zablokować standardowy skrypt rozruchowy i utworzyć plik `/etc/xinetd.d/rsync` definiujący obsługę demona `rsync` przez mechanizm `xinetd`.

Plik `/etc/default/rsync` zawiera jedynie informację o tym, czy demon `rsync` jest uruchamiany przy starcie systemu. Konfiguracja samego demona znajduje się w pliku `/etc/rsyncd.conf`, którego nie ma w domyślnej instalacji Ubuntu. Aby utworzyć ten plik, należy posłużyć się edytorem tekstu:

```
$ sudo emacs /etc/rsyncd.conf
```

Minimalny plik konfiguracyjny definiujący usługę zdalnych konfiguracji podkatalogów katalogu `/home` będzie miał następującą zawartość:

```
uid = root
transfer logging = true
log format = %h %o %f %l %b
log file = /var/log/rsyncd.log
hosts allow = 192.168.6.255/3

[homes]
  path = /home
  comment = Katalogi domowe
  auth users = jkowalski
  secrets file = /etc/rsyncd.secrets
```

Pierwszy fragment tego pliku zawiera definicję parametrów uruchamiania demona `rsyncd`. I tak po kolei: demon jest uruchamiany z uprawnieniami konta `root` (`uid`), transfery zapisuje w dzienniku (`transfer logging`), wykorzystuje specjalny format zapisów w dzienniku (`log format`) oraz sam plik dziennika (`log file`), dostęp jest zdefiniowany dla wszystkich hostów o adresach IP należących do podsieci `192.168.6.x` (`hosts allow`). Drugi fragment pliku identyfikuje synchronizowany element, nazwany `homes`, który obejmuje katalog `/home` w systemie plików. Synchronizacja tego katalogu jest wykonywana z wykorzystaniem użytkownika `jkowalski`, którego hasło musi być podane w pliku `/etc/rsyncd.secrets`.

Po zapisaniu tego pliku należy utworzyć plik `/etc/rsync.secrets`, wykorzystując do tego polecenie `sudo` i edytor tekstu:

```
$ sudo emacs /etc/rsyncd.secrets
```

Ten plik powinien zawierać wpisy dla każdego użytkownika określonego w pliku `/etc/rsync.conf`, w tym przypadku `jkowalski`. Każdy wpis w tym pliku zawiera nazwę użytkownika, dwukropek oraz jawną postać hasła użytkownika np.:

```
jkowalski:hellotere
```

Następnie należy zapisać ten plik i upewnić się, że jest czytelny jedynie dla użytkownika `root`, np.:

```
$ sudo chmod 600 /etc/rsyncd.secrets
```

Teraz można uruchomić polecenie *rsync*, wykorzystując następujące polecenie:

```
$ sudo /etc/init.d/rsync restart
```

Kopię zapasową katalogu */home* można wykonać za pomocą poniższego polecenia, gdzie *system-ubuntu* to nazwa lub adres IP systemu, na którym został uruchomiony demon *rsync*:

```
$ rsync -Havz system-ubuntu::home /media/LACIE/home
```

Argumenty polecenia *rsync* w tym przykładzie są następujące:

- ♦ **H** — zachowanie dowiązań trwałych w kopiowanych katalogach,
- ♦ **a** — tryb archiwum, który zachowuje własności, dowiązania symboliczne, nazwy urządzeń itp.; właściwie jest to skrót, pozwalający uniknąć wpisywania serii innych opcji należących do najczęściej wykorzystywanej kombinacji,
- ♦ **v** — tryb gadatliwy, powodujący wypisanie nazw kopiowanych plików (ta opcja nie jest niezbędna, ale może być interesująca, szczególnie przy kilku pierwszych uruchomieniach),
- ♦ **z** — zastosowanie kompresji przy przesyłaniu plików.

Jeśli ktoś ma problemy w wykorzystaniu polecenia *rsync*, warto sprawdzić zawartość pliku dziennika */var/log/rsyncd.log* (w systemie, z którego próbujemy pobrać pliki) w poszukiwaniu komunikatów o błędach i wskazówek ich rozwiązania. Opcja *v* również bywa użyteczna podczas szukania przyczyn problemów.



Przedstawiony przykładowy plik konfiguracyjny programu *rsync* to jedynie wersja minimalistyczna, dodatkowo nie cechuje się wysokim poziomem bezpieczeństwa. Szczegóły na temat wszystkich opcji obsługiwanych przez program *rsync* oraz opis sposobu zwiększania poziomu bezpieczeństwa mechanizmu można znaleźć w podręczniku systemowym pliku *rsync.conf* (*man rsyncd.conf*).

Instalacja i wykorzystanie programu BackupPC

W tym podrozdziale wyjaśnię, w jaki sposób zainstalować, skonfigurować i stosować program BackupPC do wykonywania kopii zapasowych różnych hostów w sieci lokalnej z centralnego serwera kopii zapasowych z zainstalowanym systemem Ubuntu. Wspomniałem wcześniej w tym rozdziale pakiet *backupper* to doskonała aplikacja, łatwa w użyciu oraz dająca duże możliwości autoryzowanym użytkownikom. Każdy autoryzowany użytkownik może zainicjować wykonanie kopii zapasowej w systemach, dla których jest zdefiniowany jako administrator w konfiguracji *backupper*. Służy do tego wygodny interfejs WWW.

Jeśli w sieci domowej posiadamy więcej niż jeden komputer lub pracujemy w wielokomputerowym środowisku korporacyjnym czy akademickim, oprogramowanie BackupPC

z pewnością warto jest sprawdzenia. Jego interfejs obsługiwany przez przeglądarkę WWW jest prosty w konfiguracji i użyciu, łatwo skonfigurować obsługiwane przez system typy kopii zapasowych, równie łatwo je uruchamiać i monitorować. Program może wykonywać kopie zapasowe z systemów Linux, Unix, Windows i Mac OS X. Dodatkową zaletą jest fakt, że nie wymaga instalacji dodatkowego oprogramowania w systemach, z których będą wykonywane kopie zapasowe.

Program BackupPC obsługuje cztery różne mechanizmy wykonywania kopii zapasowych (znane w dokumentacji BackupPC jako *transporty*), które pozwalają na wykonanie kopii zapasowych w różnych systemach operacyjnych. Należą do nich:

- ♦ `rsync` — kopie zapasowe i ich odzyskiwanie odbywa się z użyciem protokołów `rsh` i `ssh`; ta opcja jest doskonała do wykonywania kopii zapasowych z systemów Linux, Unix i Mac OS X, ale można jej użyć do wykonania kopii również z systemu Microsoft Windows obsługującego mechanizm `rsync`, zainstalowany np. w ramach środowiska Cygwin Linux.
- ♦ `rsyncd` — wykonuje kopie zapasowe z użyciem demona `rsync` działającego na systemie klienckim. To najlepsza opcja dla Linuksa, Uniksov i Mac OS X z działającym demonem `rsync`. Tego mechanizmu można użyć również do wykonania kopii z systemu Microsoft Windows obsługującego mechanizm `rsyncd`, zainstalowany np. w ramach środowiska Cygwin Linux.
- ♦ `smb` — wykonywanie i odtwarzanie kopii zapasowych z użyciem mechanizmu `smbclient` i protokołu SMB. To najlepszy (i najprostszy) sposób wykonywania kopii zapasowych z systemów Microsoft Windows, można jej również użyć do wykonania kopii zapasowej systemu Mac OS X oraz Linux i Unix z uruchomionym serwerem Samba.
- ♦ `tar` — wykonywanie i odtwarzanie kopii zapasowych z użyciem polecenia `tar`, za pośrednictwem protokołów `ssh`, `rsh` lub `nfs`. Opcja jest wykorzystywana do wykonywania kopii zapasowych systemów Linux, Unix i Mac OS X. Tego mechanizmu można użyć również do wykonania kopii z systemu Microsoft Windows obsługującego polecenia `tar`, `ssh`, `rsh` i (lub) `nfs`, zainstalowanego np. w ramach środowiska Cygwin Linux.

Domyślny transport wykonywania kopii zapasowych w BackupPC jest zdefiniowany w pliku `/etc/backuppc/config.pl`. Mechanizm używany do wykonywania kopii zapasowych z każdego indywidualnego hosta również jest zdefiniowany w tym pliku, którego składnię omówię z punktach „Definiowanie kopii zapasowej z użyciem protokołu `rsyncd`” oraz „Definiowanie kopii zapasowych z użyciem protokołu SMB”.



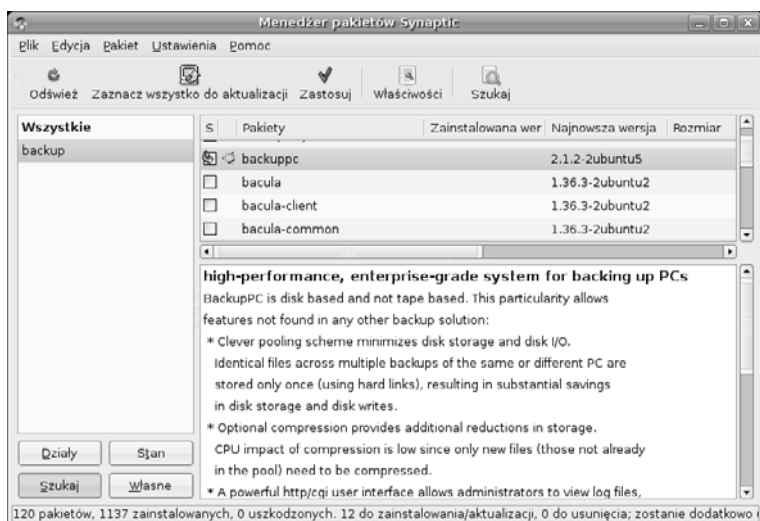
Choć program BackupPC doskonale spisuje się przy wykonywaniu kopii zapasowych systemów Microsoft Windows i Mac OS X, to należy mieć na uwadze kilka szczegółów. Po pierwsze, BackupPC nie nadaje się do wykonywania kopii zapasowych systemu Windows pozwalających na odtworzenie systemu od zera. BackupPC wykorzystuje aplikację `smbclient`, za której pomocą może skopiować dyski systemu Windows z poziomu Linuksa, ale nie wykonuje kopii ACL-i systemu plików Windows i nie jest w stanie skopiować plików zablokowanych w systemie Windows (m.in. skrzynek pocztowych programu Outlook). Podobne uwagi dotyczą również systemu Mac OS X — w tym przypadku BackupPC nie zachowuje rozszerzonych atrybutów plików. Listę aktualnych ograniczeń programu BackupPC można znaleźć na stronie <http://backuppc.sourceforge.net/faq/limitations.html>. Na szczęście, to zaskakująco krótki dokument!

Instalacja pakietu backuppc

Specjalistyczne rozwiązania tworzenia kopii zapasowych, takie jak BackupPC, nie są domyślnie instalowane w Ubuntu, ponieważ dla większości użytkowników z reguły stanowią rzeczy absolutnie zbędne. Podobnie jak w przypadku większości pakietów w Ubuntu, menedżer pakietów Synaptic pozwala zainstalować pakiet backuppc wraz z oprogramowaniem wspomagającym. W tym celu należy uruchomić Synaptic z menu systemowego *System/Administracja/Menedżer pakietów Synaptic*. Po uruchomieniu aplikacji Synaptic kliknąć przycisk *Szukaj*, co spowoduje wyświetlenie okna wyszukiwania. W polu wyszukiwania wpisać *backup*, zaznaczyć opcję *Nazwa i opis* i kliknąć *Szukaj*. Po zakończeniu wyszukiwania należy odszukać pakiet *backuppc*, kliknąć jego nazwę prawym przyciskiem myszy, z menu podręcznego wybrać opcję *Zaznacz do instalacji*. Na rysunku 22.3 przedstawiam pakiet *backuppc* zaznaczony do instalacji.

Rysunek 22.3.

Instalacja pakietu
backuppc

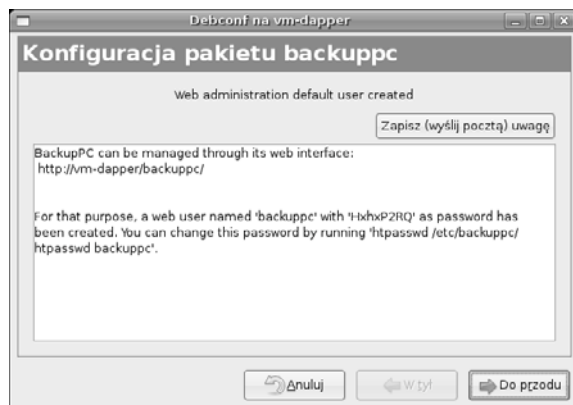


W zależności od tego, jakie oprogramowanie zostało poprzednio zainstalowane w Ubuntu i od innych pakietów zaznaczonych w programie Synaptic, może pojawić się komunikat o konieczności zainstalowania innych pakietów w celu spełnienia zależności. Gdy pojawi się takie okno, należy kliknąć przycisk *Zaznacz* w celu zaznaczenia pakietów niezbędnych do instalacji.

Po zakończeniu zaznaczania pakietów do instalacji trzeba kliknąć ikonę *Zastosuj* na pasku narzędzi programu Synaptic, co rozpocznie instalację pakietu *backuppc* i z nim powiązanych. Po zakończeniu instalacji rozpocznie się faza konfiguracji. Podczas tej fazy Synaptic uruchomi skrypt konfiguracyjny, które będzie wykorzystywane do obsługi programu *backuppc* za pośrednictwem serwera WWW. Na rysunku 22.4 zostało zaprezentowane okno dialogowe, w którym należy zdefiniować hasło dostępu do interfejsu WWW programu *backuppc*.

Należy w nim wprowadzić hasło dostępu do interfejsu *backuppc*, po czym kliknąć przycisk *Do przodu*. Po zakończeniu instalacji i konfiguracji system jest gotowy do wykonywania kopii zapasowej systemu oraz innych systemów w sieci.

Rysunek 22.4.
Hasło dla interfejsu
backuppc



Więcej informacji na temat instalacji aplikacji za pomocą menedżera pakietów Synaptic można znaleźć w rozdziale 20.

Konfiguracja programu BackupPC

W systemie Ubuntu backuppc przechowuje swoje informacje konfiguracyjne w dwóch lokalizacjach. Ogólne informacje konfiguracyjne oraz hasła są zapisane w plikach w katalogu `/etc/backuppc`. Pliki kopii zapasowych oraz konfiguracje specyficzne dla poszczególnych hostów są zapisane w katalogu `/var/lib/backuppc`.



Kopie zapasowe pojedynczego systemu zajmują określony obszar na dysku, co jest dodatkowo zwielokrotnione, gdy na jednym serwerze są zapisywane dane kilku hostów. Z tego powodu przed rozpoczęciem korzystania z serwera backuppc warto dołożyć do systemu nowy dysk twardy o odpowiedniej pojemności i sformatować go jako wolumin logiczny. Następnie zawartość katalogu `/var/lib/backuppc` należy skopiować na ten nowy dysk (z zachowaniem uprawnień do plików i własności), po czym zamontować ten dysk w katalogu `/var/lib/backuppc`. Jeśli w przyszłości okaże się, że potrzebujemy więcej miejsca na kopie zapasowe, można do systemu dodać kolejne dyski twarde i połączyć je z logicznym woluminem, zwiększając jego rozmiar. Program backuppc posiada również mechanizmy pozwalające na migrację starych kopii zapasowych na inne hosty w celach archiwalnych, co pozwala zmniejszyć zapotrzebowanie na miejsce na głównym serwerze kopii zapasowych. Zagadnienie konfiguracji hosta na potrzeby przechowywania archiwów nie jest omówione w tej książce, ale odpowiednie informacje można znaleźć w dokumentacji systemu BackupPC, która jest na bardzo wysokim poziomie.

W pierwszej kolejności należy zmodyfikować hasło dostępu do interfejsu backuppc na prostsze do zapamiętania od losowego ciągu znaków generowanego podczas instalacji programu. Można tego dokonać, wywołując następujące polecenie:

```
$ sudo htpasswd /etc/backuppc/htpasswd backuppc
```

To wywołanie wykorzystuje polecenie `sudo` do wywołania programu `htpasswd` w celu zmiany hasła użytkownika `backuppc` zapisanego w pliku `/etc/backuppc/htpasswd`. Gdy system poprosi o nowe hasło, należy wpisać coś prostszego do zapamiętania niż `HxhxP2RQ`, które było domyślnym hasłem w mojej instalacji backuppc, co widać na rysunku 22.4.

System poprosi o ponowne wpisanie nowego hasła, aby upewnić się, że zostało wprowadzone prawidłowo.

Identyfikacja hostów do wykonywania kopii zapasowych

Każdy host do wykonywania kopii zapasowych musi być zidentyfikowany w pliku */etc/backuppc/hosts*. Podobnie jak pozostałe pliki konfiguracyjne systemu backuppc, ten plik ma bardzo prostą składnię. Znak # w dowolnym miejscu wiersza rozpoczyna komentarz, co pomaga wprowadzać informacje opisowe. Oto minimalna konfiguracja programu backuppc:

host	dhcp	user	moreUsers
localhost	0	backuppc	

Pierwszy wiersz w pliku */etc/backuppc/hosts* (niebędący komentarzem) definiuje nazwy poszczególnych pól w każdym z wierszy. Tego wiersza nie należy modyfikować (w powyższym przykładzie wspomniany wiersz rozpoczyna się od słowa kluczowego host). Wszystkie pozostałe wiersze reprezentują poszczególne pozycje hostów, których dane mają być kopiowane. Pierwszy wpis (jedyne w tym przykładzie) ma znaczenie specjalne: dotyczy wykonywania kopii zapasowych na serwerze backuppc i nie należy go modyfikować.

Pola definiujące hosty w poszczególnych pozycjach mają następujące znaczenie.

- ♦ Pierwsze pole identyfikuje maszynę: jest to nazwa hosta, adres IP lub nazwa NetBios.
- ♦ Drugie pole powinno mieć wartość 0 w przypadku hosta, którego nazwę można odczytać z DNS-u, lokalnego pliku hosts lub za pomocą zapytania *nmblookup*. To pole może też mieć wartość 1 identyfikującą systemy, których nazwy muszą być odczytywane przez próbkowanie zakresu adresów DHCP, co ma zastosowanie w środowiskach, w których mechanizmy DHCP i WINS nie są w pełni zintegrowane. Ustawienie tej wartości na 1 wymaga modyfikacji zmiennej `$Conf{DHCPAddressRanges}` zapisanej w pliku konfiguracji hosta, definiującej bazowy adres IP oraz zakres adresów, które mają być próbkowane.
- ♦ Trzecie pole identyfikuje nazwę użytkownika odpowiedzialnego za wykonywanie kopii zapasowych na tym hoście. Ten podstawowy użytkownik kopii zapasowej będzie otrzymywał wiadomości e-mail z informacjami o statusie wykonywanych zadań kopii zapasowych. Z reguły w tym polu ustawiam użytkownika backuppc, aby historię komunikatów z systemu backuppc przechowywać na jednym koncie e-mail, ale można tu wpisać dowolną nazwę konta.
- ♦ Czwarte pole (opcjonalne) składa się z jednej nazwy lub większej liczby nazw użytkowników posiadających prawa administratora do tworzenia kopii zapasowych lub odtwarzania plików w danym systemie. Gdy w tym polu ma być zapisana większa liczba kont, należy je odseparować przecinkami.

Oto przykład bardziej skomplikowanej konfiguracji systemu backuppc z jednego z moich serwerów:

host	dhcp	user	moreUsers
localhost	0	backuppc	

192.168.6.64	0	backuppc	wvh
64bit	0	backuppc	wvh,djf
64x2	0	backuppc	juser
win2k	0	backuppc	wvh,djf



Program backuppc sprawdza znacznik czasu pliku `/etc/backuppc/hosts` za każdym razem, gdy uruchamiany jest główny proces programu. W przypadku wykrycia zmiany pliku zostaje on załadowany. Z tego powodu nie należy zapisywać zmian w pliku `hosts` aż do momentu, gdy zostaną utworzone odpowiednie pliki konfiguracyjne dla poszczególnych hostów, co zostanie wytłumaczone na przykładach w ramach dwóch kolejnych punktów. Jeśli proces backuppc wczyta plik konfiguracyjny, zanim zostaną utworzone pliki konfiguracyjne poszczególnych hostów, a inny autoryzowany użytkownik uruchomi wykonanie kopii zapasowej swojego systemu, powstaną niewłaściwe kopie albo wystąpi błąd. Zmiany w pliku można wprowadzić w postaci zablokowanej znakiem komentarza (`#` na początku wiersza) aż do momentu prawidłowego skonfigurowania hostów, których dane mają być kopiowane.

Definiowanie kopii zapasowej z użyciem protokołu rsyncd

We wcześniejszym podrozdziale, w punkcie zatytułowanym „Wykonywanie aktualnych kopii katalogu lokalnego z użyciem polecenia `rsync`” wyjaśniłem, w jaki sposób w Ubuntu skonfigurować mechanizm `rsync` w trybie demona oraz jak zdefiniować wpisy poszczególnych zasobów, które mają być poddawane synchronizacji z użyciem programu `rsync`. Przykładowy plik konfiguracyjny dla mechanizmu `rsync` przedstawiony w tamtym punkcie definiował zasób synchronizacyjny o nazwie `homes`, który pozwalał autoryzowanemu użytkownikowi synchronizować wszystkie podkatalogi znajdujące się w katalogu `/home` w przykładowym systemie Ubuntu. Dokładnie tę samą konfigurację wykorzystamy w naszych przykładach z tego punktu.

W poprzednim punkcie pokazałem, w jaki sposób w pliku `/etc/backuppc/hosts` definiować wpisy dla poszczególnych hostów, których dane mają być kopiowane przez mechanizm backuppc. Na pierwszym etapie pisania indywidualnych plików konfiguracyjnych poszczególnych hostów należy utworzyć katalog zawierający dane konfiguracyjne, dzienniki itp. W tym punkcie wykorzystam przykładowy wpis `64bit`, który zdefiniowałem w przykładzie z punktu „Identyfikacja hostów do wykonywania kopii zapasowych”.

Katalog dla hosta `64bit` (`/var/lib/backuppc/64bit`) utworzymy, wykorzystując polecenie `sudo`:

```
$ sudo mkdir /var/lib/backuppc/64bit
```

Następnie wykorzystamy polecenie `sudo` oraz edytor tekstów do utworzenia w tym katalogu pliku konfiguracyjnego dla naszego hosta o nazwie `config.pl`:

```
$ sudo emacs /var/lib/backuppc/64bit/config.pl
```

Zawartość tego pliku powinna być następująca:

```
$Conf{XferMethod} = 'rsyncd';
$Conf{CompressLevel} = '3';
$Conf{RsyncShareName} = 'homes';
$Conf{RsyncdUserName} = 'jkowalski';
$Conf{RsyncdPasswd} = 'hellothere';
```

Pierwszy wiersz identyfikuje mechanizm wykonywania kopii zapasowych, w tym przypadku jest to *rsyncd*, który przesłania domyślny mechanizm zdefiniowany w pliku */etc/backuppc/config.pl*. Drugi wiersz ustawia kompresję na poziomie 3., co stanowi dobry kompromis między obciążeniem procesora, czasem niezbędnym na wykonanie kompresji i rozmiarem plików wynikowych. Ostatnie trzy wpisy w tym pliku odpowiadają wpisom z pliku konfiguracyjnego *rsyncd.conf* oraz *rsyncd.secrets*, omówionym w punkcie „Wykonywanie aktualnych kopii katalogu lokalnego z użyciem polecenia *rsync*”, wcześniej w tym rozdziale.



Podczas korzystania z systemu backuppc do automatycznego wykonywania kopii zapasowych z reguły tworzę osobnego użytkownika upoważnionego do synchronizacji plików za pośrednictwem mechanizmu *rsync*, wykorzystywanego wyłącznie do wykonywania kopii zapasowych, dzięki czemu w dziennikach systemowych od razu widać, czy żądania wykonania kopii pochodzą od upoważnionych użytkowników. W tym celu należy zdefiniować takiego, specjalnego, użytkownika (używam nazwy konta backuppc) w zmiennej *auth users* w pliku */etc/rsyncd.conf*, po czym utworzyć odpowiednią parę *nazwa_użytkownika:hasło* w pliku */etc/rsyncd.secrets*. Następnie trzeba odpowiednio zmodyfikować plik konfiguracyjny programu backuppc dla danego hosta, wprowadzając konkretną nazwę użytkownika i hasło. Dla uproszczenia opisu w przykładach nie zastosowałem się do tej zasady, ale polecam ją, ponieważ upraszcza zarządzanie i zwiększa czytelność dzienników systemu kopii zapasowych.

Jeśli zdalny system wykorzystuje program *rsync* w ścieżce innej niż domyślna */usr/bin/rsync* lub program *rsync* nasłuchuje na niestandardowym porcie (innym niż 873), należy odpowiednio skonfigurować plik programu backuppc dla danego hosta. Ustawienia dla standardowych wartości są następujące:

```
$Conf{RsyncdClientPort} = 873;
$Conf{RsyncClientPath} = '/usr/bin/rsync';
```

Następnie należy zmodyfikować właściciela i grupę katalogu */var/lib/backuppc/64bit* (na backuppc), po czym zmodyfikować zabezpieczenie pliku */var/lib/backuppc/64bit/config.pl* w taki sposób, aby nie był publicznie czytelny (ponieważ zawiera hasło), np. za pomocą następujących poleceń:

```
$ sudo chown -Rv backuppc:backuppc /var/lib/backuppc/64bit
$ sudo chmod 600 /var/lib/backuppc/64bit/config.pl
```

W ostatnim kroku opracowywania konfiguracji programu backuppc dla hosta należy poinformować sam program, że musi od nowa załadować swoją konfigurację. Można tego dokonać na kilka sposobów: przez wymuszenie ponownego załadowania pliku konfiguracyjnego, ponowne uruchomienie procesu backuppc lub przez wysłanie do tego procesu sygnału HUP. Innymi słowy, do ponownego załadowania konfiguracji przez program backuppc wystarczy wywołać następujące polecenie:

```
$ sudo /etc/init.d/backuppc reload
```

W interfejsie WWW pojawi się definicja hosta kopii zapasowych. W tym momencie można przejść do instrukcji zawartych w punkcie „Uruchamianie kopii zapasowych w programie BackupPC”.

Przykład opisany tutaj wykonuje kopie zapasowe wyłącznie katalogu */home* hosta z całą zawartością. Aby wykonać kopie innych katalogów, należy po prostu utworzyć inne

wpisy w pliku `/etc/rsyncd.conf` zdalnego hosta i dopisać odpowiadające im wpisy w pliku programu `backuppc` dla tego hosta. Przykładowo w celu wykonywania kopii zapasowych wpisów o nazwach `homes`, `/` i `/boot` należy definicję zmiennej `RsyncShareName` w pliku konfiguracyjnym programu `backuppc` dla danego hosta zmodyfikować w następujący sposób:

```
$Conf{RsyncShareName} = [ '/', 'homes', '/boot' ];
```

Jeśli wykonujemy kopie zapasowe wielu systemów plików lub punktów synchronizacyjnych, można w pliku konfiguracyjnym dla danego hosta zdefiniować odpowiedni zestaw argumentów wywołania polecenia `rsync`. Dzięki temu możemy użyć specjalnej opcji, takiej jak `--one-file-system`, która spowoduje, że program `backuppc` nie będzie wykonywał kopii zapasowych systemów plików podmontowanych w kopiowanych katalogach (co w rezultacie będzie zmuszało do zdefiniowania osobnych wpisów w celu wykonania kopii tych zamontowanych systemów plików, co z kolei może uprościć procedury odtwarzania plików). Można również wykorzystać opcje wykluczające wybrane katalogi z kopii. Z tej możliwości z pewnością należy skorzystać, wykonując kopię zapasową głównego katalogu systemu plików (`/`), co demonstruje poniższy przykład:

```
$Conf{RsyncArgs} = [  
    # tutaj lista oryginalnych argumentów  
    '--one-file-system',  
    '--exclude', '/dev',  
    '--exclude', '/proc',  
    '--exclude', '/media',  
    '--exclude', '/mnt',  
    '--exclude', '/lost+found',  
];
```

Te ustawienia zapobiegają wykonaniu kopii zapasowej katalogu `/dev`, który zawiera pliki obsługi urządzeń i w większości nowoczesnych systemów Linux jest tworzony dynamicznie przy starcie systemu. Pomijany jest też katalog `/proc`, który jest punktem montowania systemu plików przechowywanego w pamięci, a zawierającego różne przejściowe dane jądra systemu, jak również katalogi `/media` i `/mnt`, w których montowane są nośniki wymienne. Nie należy też wykonywać kopii zapasowej katalogu `/lost+found`, który jest wykorzystywany na etapie sprawdzania spójności systemu plików. Katalogi można również wykluczać z kopii zapasowych z użyciem klauzuli konfiguracyjnej samego `backuppc`, wykorzystując zmienną `BackupFilesExclude` np.:

```
$Conf{BackupFilesExclude} = [ '/dev', '/proc', '/media', '/mnt', '/lost+found' ];
```

Program `backuppc` w pierwszej kolejności odczytuje konfigurację z pliku `/etc/backuppc/config.pl`, po czym ładuje ustawienia specyficzne dla danego hosta. Dzięki temu w pliku `/etc/backuppc/config.pl` można zdefiniować ustawienia domyślne, wspólne dla wszystkich konfiguracji. Wraz z nabieraniem doświadczenia w korzystaniu z systemu `backuppc` można stopniowo modyfikować domyślne wartości z pliku `/etc/backuppc/config.pl`, np. definiując zmienne `$Conf{RsyncArgs}`, `$Conf{BackupFilesExclude}` i `$Conf{CompressLevel}`, co pozwoli zmniejszyć liczbę ustawień w plikach konfiguracyjnych dla poszczególnych hostów.

Definiowanie kopii zapasowych z użyciem protokołu SMB

W punkcie „Identyfikacja hostów do wykonywania kopii zapasowych” pokazałem, w jaki sposób w pliku `/etc/backuppc/hosts` definiować pozycje dla poszczególnych hostów, których dane mają być kopiowane przez program backuppc. W pierwszym etapie pisania indywidualnych plików konfiguracyjnych poszczególnych hostów należy utworzyć katalog zawierający dane konfiguracyjne, dzienniki itp. W tym punkcie wykorzystam przykładowy wpis `win2k`, który zdefiniowałem w przykładzie konfiguracji pliku `hosts`. Jak można domyślić się po nazwie, tym razem będziemy mieli do czynienia z systemem Microsoft Windows 2000.

Na pierwszym etapie definiowania pliku konfiguracyjnego hosta należy utworzyć katalog `/var/lib/backuppc/win2k`, np.:

```
$ sudo mkdir /var/lib/backuppc/win2k
```

Następnie za pomocą polecenia `sudo` i dowolnego edytora tekstu trzeba w tym katalogu utworzyć plik konfiguracyjny o nazwie `config.pl`, np.:

```
$ sudo emacs /var/lib/backuppc/win2k/config.pl
```

Plik ten powinien mieć następującą zawartość:

```
$Conf{XferMethod} = 'smb';  
$Conf{CompressLevel} = '3';  
$Conf{SmbShareName} = ['jkowalski', 'znowak'];  
$Conf{SmbShareUserName} = 'backuppc';  
$Conf{SmbSharePasswd} = 'hellothere';
```

Pierwszy wiersz identyfikuje mechanizm wykonywania kopii zapasowych, w tym przypadku `smb`, co przesłania domyślną wartość zdefiniowaną w globalnym pliku `/etc/backuppc/config.pl`. Drugi wiersz ustawia poziom kompresji kopii zapasowych tego hosta na wartość 3, co stanowi dobry kompromis między obciążeniem procesora, czasem wykonania kompresji i rozmiarem plików wynikowych. Ostatnie wpisy w tym pliku definiują zasoby sieciowe systemu Windows, które mają być kopiowane, nazwę użytkownika uprawnionego do odczytu tych zasobów oraz jego hasło.



Podczas korzystania z programu backuppc do wykonywania kopii zapasowych systemu Windows warto utworzyć w tym systemie konto użytkownika, które będzie stosowane wyłącznie do wykonywania kopii zapasowych. Konto to należy dodać do grupy *Operatorzy kopii zapasowych*. Dzięki temu można uniknąć wpisywania hasła administratora systemu Windows w konfiguracji programu backuppc. Mimo że te pliki konfiguracyjne są dobrze chronione przed odczytem przez osoby postronne, jednak im mniej miejsc, gdzie są zapisane kluczowe hasła, tym lepiej, szczególnie wtedy, kiedy są to hasła pozwalające na wszystko.

Następnie należy zmodyfikować właściciela i grupę katalogu `/var/lib/backuppc/win2k` (na `backuppc`), po czym zmodyfikować uprawnienia do pliku `/var/lib/backuppc/win2k/config.pl`, blokując możliwość jego odczytu przez innych użytkowników (poza właścicielem):

```
$ sudo chown -Rv backuppc:backuppc /var/lib/backuppc/win2k  
$ sudo chmod 600 /var/lib/backuppc/win2k/config.pl
```

W ostatnim kroku tworzenia konfiguracji programu backuppc dla hosta należy poinformować tenże program, że musi od nowa załadować swoją konfigurację. Można tego dokonać na kilka sposobów: przez wymuszenie ponownego załadowania pliku konfiguracyjnego, ponowne uruchomienie procesu backuppc lub przez wysłanie do tego procesu sygnału HUP. Innymi słowy, do ponownego załadowania konfiguracji przez program backuppc wystarczy wywołać następujące polecenie:

```
$ sudo /etc/init.d/backuppc reload
```

W interfejsie WWW pojawi się definicja hosta kopii zapasowych. W tym momencie można przejść do instrukcji zawartych w punkcie „Uruchamianie kopii zapasowych w programie BackupPC”.

Przykład opisany tutaj wykonuje kopie zapasowe wyłącznie katalogów prywatnych użytkowników hosta. Jak wspominałem wcześniej, backuppc nie posiada mechanizmów pozwalających na odtwarzanie systemów operacyjnych od zera, dlatego z reguły nie wykonuję kopii zapasowych takich zasobów jak C\$, czyli domyślnego zasobu reprezentującego całą partycję systemową. Jeśli ktoś zdecyduje się kopiować sterowniki, rejestr itp., można — oczywiście — uaktywnić wykonywanie kopii z tego zasobu. Jednak uważam, że bardziej opłaca się zainstalować i skonfigurować system Windows od zera. Systemy z tej rodziny gromadzą duże ilości śmieci w swoich systemach plików, a zatem ponowna instalacja systemu może zwolnić sporą ilość miejsca na dysku. Gdy wykorzystuje się kilka systemów o identycznej konfiguracji, warto rozważyć wykorzystanie obrazów partycji wykonanych specjalizowanymi narzędziami, takimi jak Norton Ghost, partimage lub g4u. Te narzędzia pozwalają wykonać pełny obraz partycji systemowej i odtworzyć go, gdy wystąpi problem, bez konieczności ponownego instalowania sterowników dla każdego urządzenia przyłączonego do systemu oraz bez potrzeby ponownego instalowania wszystkich aplikacji.



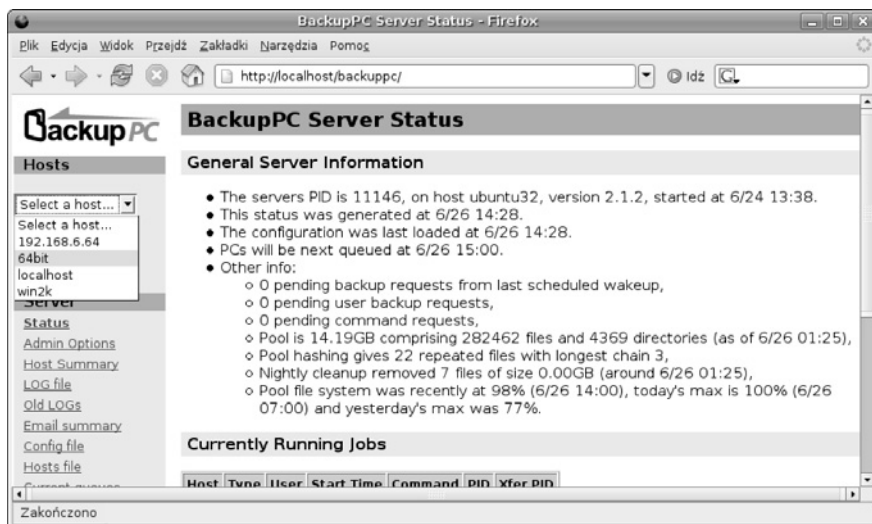
Program backuppc w pierwszej kolejności odczytuje konfigurację z pliku `/etc/backuppc/config.pl`, po czym łąduje ustawienia specyficzne dla danego hosta. Dzięki temu w pliku `/etc/backuppc/config.pl` można zdefiniować ustawienia domyślne, wspólne dla wszystkich konfiguracji. Mając już doświadczenie w używaniu systemu backuppc można stopniowo modyfikować domyślne wartości z pliku `/etc/backuppc/config.pl`, np. definiując zmienną `$Conf{CompressLevel}`, co pozwoli zmniejszyć liczbę ustawień w plikach konfiguracyjnych dla poszczególnych hostów.

Uruchamianie kopii zapasowych w programie BackupPC

Dzięki temu, że interfejs programu backuppc jest oparty na przeglądarce WWW, obsługa podstawowych funkcji, takich jak uruchamianie kopii zapasowych, sprawdzanie ich statusów czy przeglądanie historii wykonanych kopii, to niezwykle proste zadania. Interfejs programu backuppc uruchamia się, wywołując w przeglądarce WWW adres `http://nazwa_hosta/backuppc`, gdzie ***nazwa_hosta*** to nazwa serwera, na którym uruchomiony jest demon backuppc. Pojawi się okno dialogowe z prośbą o zalogowanie w systemie. Po wpisaniu nazwy konta i hasła zdefiniowanych w pliku `/etc/backuppc/htpasswd` pojawi się strona serwera backuppc (patrz rysunek 22.5).

Po pojawieniu się tej strony z listy rozwijanej *Select host...* należy wybrać nazwę odpowiedniego hosta, co przedstawiam na rysunku 22.6.

Rysunek 22.5.
Strona główna
interfejsu WWW
programu backuppc



Rysunek 22.6. Wybór hosta do wykonania kopii zapasowej

Po wybraniu nazwy hosta pojawi się strona podsumowania zawierająca informacje o statusie, listę użytkowników autoryzowanych do wykonania kopii i odtworzenia plików na tym hoście oraz treść ostatniej wiadomości e-mail na temat tego hosta wysłanej przez system backuppc, co przedstawiam na rysunku 22.7.

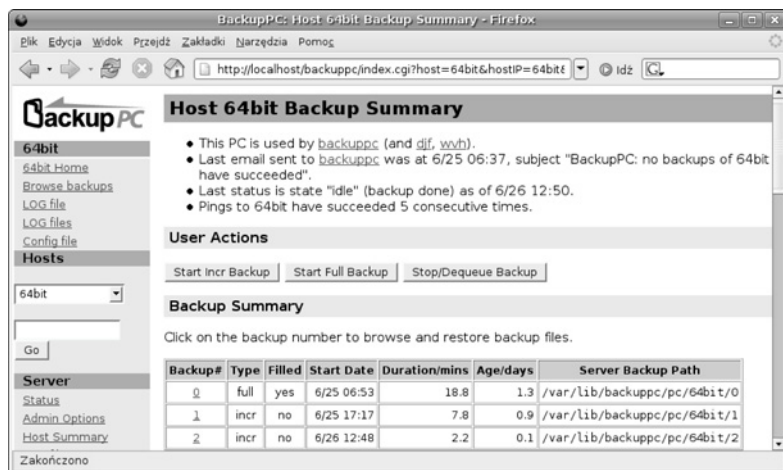


Strony domowe skonfigurowanych systemów zawierają temat ostatniego wysłanego e-maila. Tego typu wiadomości są wysyłane wyłącznie w problematycznych sytuacjach, dlatego należy zwrócić uwagę na to, że informacja o błędzie na stronie hosta nie oznacza, że ten problem nadal istnieje.

Stronę tę po wyświetleniu można przewinąć w dół w celu zapoznania się z dodatkowymi informacjami o statusie dostępnych kopii zapasowych, błędach transmisji, które pojawiły się w trakcie wykonania kopii. Możemy też przyjrzeć się tabeli zawierającej dane

Rysunek 22.7.

Strona podsumowania
na temat hosta
w interfejsie WWW
programu backuppc

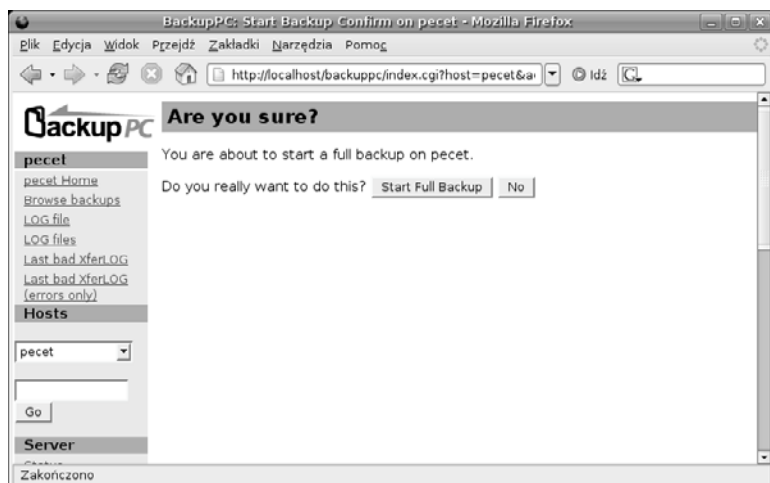


statystyczne o puli zasobów dyskowych, w której zapisywane są kopie zapasowe, oraz o procentowej oszczędności miejsca na dysku uzyskanej dzięki kompresji danych.

Aby rozpocząć tworzenie kopii zapasowej, należy kliknąć łącze *Start Full Backup*, co spowoduje rozpoczęcie wykonywania pełnej (archiwalnej) kopii zapasowej, lub łącze *Start Incr Backup*, co spowoduje rozpoczęcie wykonywania kopii przyrostowej zawierającej zmiany od ostatniej pełnej kopii. W efekcie pojawi się strona potwierdzenia przedstawiona na rysunku 22.8.

Rysunek 22.8.

Potwierdzenie
wykonania pełnej
kopii zapasowej



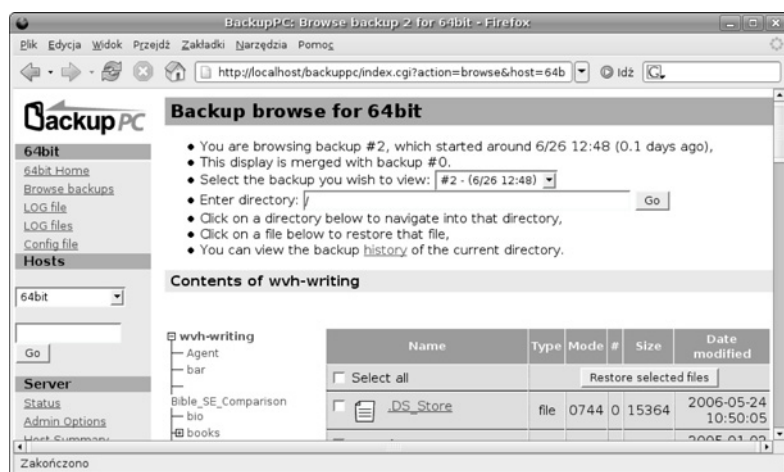
Kliknięcie przycisku *Start Full Backup* (lub *Start Incr Backup*) spowoduje wrzucenie zadania do kolejki i wyświetli łącze powrotu do głównej strony hosta w celu monitorowania stanu kopii zapasowej.

Odtwarzanie danych z kopii zapasowej w programie BackupPC

Odtwarzanie plików z kopii zapasowej w programie backuppc odbywa się w trybie online, taką możliwość mają wszyscy autoryzowani użytkownicy skojarzeni z hostami w pliku */etc/backuppc/hosts*. Program backuppc umożliwia przeglądanie kopii zapasowych, interaktywny wybór plików i katalogów do odtworzenia oraz daje kilka możliwości odtworzenia zasobów.

Aby odtworzyć pliki lub katalogi, należy kliknąć nazwę pełnej lub przyrostowej kopii zapasowej, w której znajdują się wybrane pliki. Pojawi się strona przedstawiona na rysunku 22.9.

Rysunek 22.9.
Odtwarzanie plików i katalogów w interfejsie WWW programu backuppc



U dołu ekranu przedstawionego na rysunku 22.9 widać hierarchiczny listing plików i katalogów zawartych w pełnej lub przyrostowej kopii zapasowej. Jeśli wybrana została przyrostowa kopia zapasowa, zawartość tej kopii jest nakładana na zawartości poprzedniej pełnej kopii zapasowej, dzięki czemu mamy dostęp do pełnej migawki systemu z momentu wykonania tej kopii. Po tym drzewie można nawigować, przechodząc do podkatalogów, zawartość katalogu jest wyświetlana po prawej stronie.

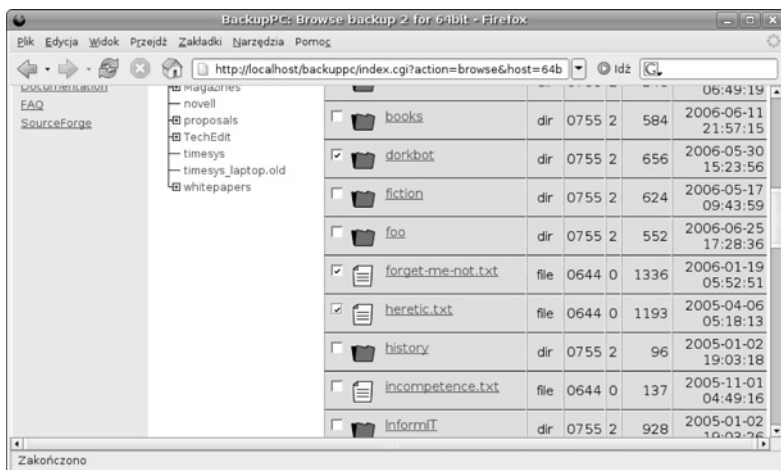
Aby zaznaczyć pliki i katalogi do odtworzenia, trzeba zaznaczyć pola opcji przy odpowiednich nazwach, tak jak to przedstawiam na rysunku 22.10.

Po zaznaczeniu wszystkich plików i katalogów do odtworzenia należy przewinąć okno do końca i kliknąć przycisk *Restore selected files*. Pojawi się strona z pytaniem o sposób odtworzenia pliku (patrz rysunek 22.11).

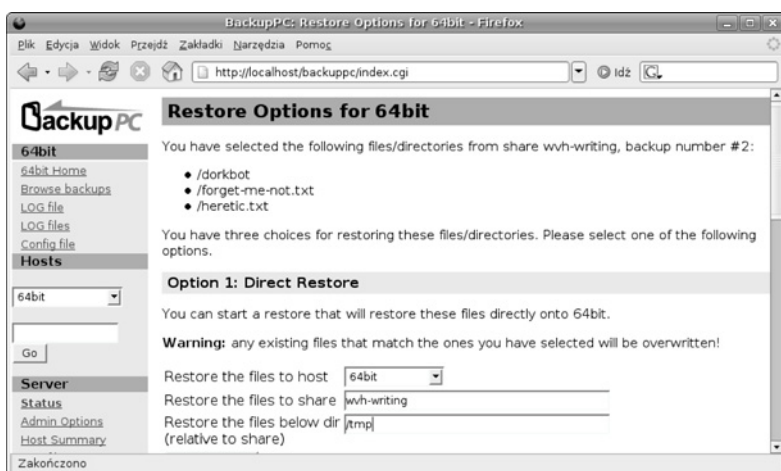
Do wyboru są trzy opcje.

- ♦ **Direct restore** — odtworzenie bezpośrednie, pliki zostaną odtworzone w to miejsce, z którego zostały skopiowane. Wykonując ten rodzaj odtworzenia, uzyskujemy dodatkową opcję odtworzenia plików w dokładnie to samo miejsce, lub do podkatalogu, który zostanie utworzony przez backuppc (chyba że już istnieje).

Rysunek 22.10.
Zaznaczanie plików
i katalogów
do odtworzenia



Rysunek 22.11.
Określenie sposobu
odtworzenia plików
w interfejsie WWW
programu backuppc



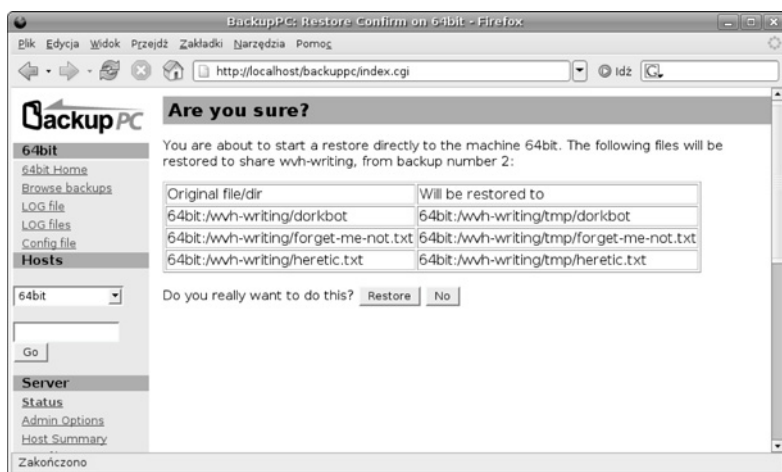
Ta druga opcja jest bardzo rozsądna, ponieważ pozwala uniknąć sytuacji, gdy jakieś ważne pliki zostaną omyłkowo nadpisane ich starszą wersją. W przypadku wyboru tej opcji należy podać nazwę podkatalogu, do którego mają być odtworzone pliki (z reguły korzystam z nazwy *tmp*), po czym kliknąć *Start restore*.

- ♦ **Download Zip** — pobranie plików z serwera w postaci archiwum ZIP, skąd mogą być ręcznie odtworzone przez użytkownika w systemie lokalnym. Przy wyborze tej opcji można opcjonalnie określić poziom kompresji wykorzystywany w archiwum ZIP, co może mieć znaczenie podczas odtwarzania dużej liczby plików. Aby wybrać tę opcję, należy kliknąć *Download Zip file*.
- ♦ **Download Tar archive** — pobranie plików z serwera w postaci archiwum tar, skąd mogą być ręcznie odtworzone przez użytkownika w systemie lokalnym. Aby wybrać tę opcję, należy kliknąć *Download Tar file*.

Po wybraniu opcji *Direct restore* program backuppc dodatkowo wyświetli stronę potwierdzenia (patrz rysunek 22.12).

Rysunek 22.12.

Potwierdzenie bezpośredniego odtworzenia plików w systemie zdalnym

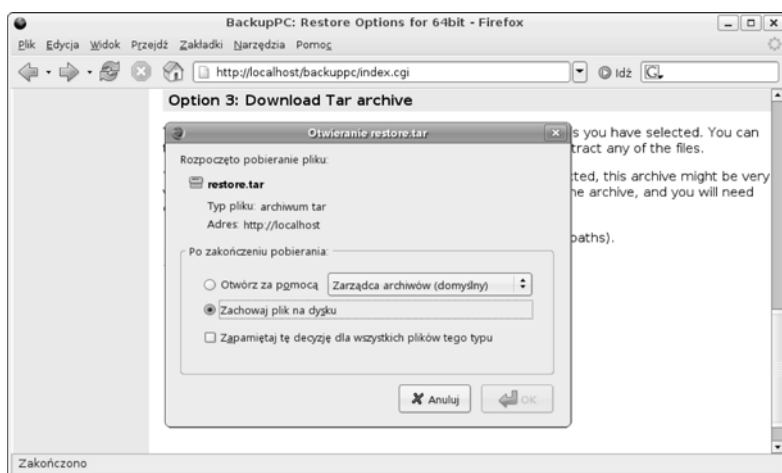


Potwierdzenie to zawiera listę plików i katalogów zaznaczonych do odtworzenia oraz potwierdzenie lokalizacji, do której mają być odtworzone wraz z nazwą podkatalogu (jeśli został podany). Aby kontynuować, należy kliknąć *Restore*.

W przypadku wykorzystania opcji *Zip* lub *Tar* przeglądarka wyświetli standardowe okno dialogowe z potwierdzeniem pobierania pliku. Na rysunku 22.13 przedstawiam przykładowe okno dialogowe dla odtworzenia w formacie tar.

Rysunek 22.13.

Odtwarzanie wybranych plików z kopii w formacie archiwum tar



Jak można się było przekonać, program backuppc stanowi rozbudowane, elastyczne narzędzie do wykonywania kopii zapasowych z wielu systemów w sieci lokalnej i odtwarzania plików z tych kopii. Wystarczy opracować kilka plików konfiguracyjnych oraz zapewnić odpowiednią ilość miejsca na dysku, a utrata plików (oraz czasu, jak to zwykle bywa w takich sytuacjach) stanie się koszmarem z przeszłości.

Podsumowanie

W tym rozdziale omówiłem kopie zapasowe, które należą do najbardziej czasochłonnych, nudnych, ale ważnych zagadnień związanych z komputeryzacją. Dla Ubuntu można znaleźć mnóstwo doskonałego oprogramowania, które można łatwo wykorzystać do tworzenia kopii zapasowych różnych typów, od prostego kopiowania plików w celu zapisania na nośnikach wymiennych, po skomplikowane systemy do zarządzania kopiami zapasowymi w sieci. W rozdziale opisałem proste metody wykonywania kopii zapasowych w sieci, a na końcu opowiedziałem o programie BackupPC, doskonałym narzędziu na licencji open source, służącym do wykonywania regularnych kopii zapasowych w sieci z wielu systemów na jeden serwer centralny.

W rozdziale 23. opiszę sposoby dodawania sprzętu do systemu Ubuntu, który, jak każdy nowoczesny system operacyjny, doskonale radzi sobie z automatycznym rozpoznawaniem peryferiów przyłączanych do systemu, dzięki czemu użytkownik ma łatwy dostęp do drukarek, skanerów i nośników wymiennych, takich jak napędy USB, flash, zewnętrzne dyski twarde, urządzenia FireWire itp. Dodawanie nowych napędów wewnętrznych wymaga nieco więcej pracy, ale jest kluczowym elementem procesu aktualizacji sprzętu i na tyle prostym zadaniem, że nie ma problemu z jego opanowaniem szczególnie wtedy, gdy skorzysta się z porad zawartych w następnym rozdziale.

Rozdział 23.

Dodawanie sprzętu i przyłączanie urządzeń peryferyjnych

W tym rozdziale:

- ♦ Dodawanie drukarek
- ♦ Dodawanie i używanie skanera
- ♦ Wykorzystywanie zewnętrznych napędów
- ♦ Dodawanie wewnętrznych napędów
- ♦ Używanie kart PCMCIA
- ♦ Dodawanie kart PCI

Kluczowym problemem systemu komputerowego jest obsługa ogromnej różnorodności urządzeń. Osprzęt staje się coraz powszechniejszy, użytkownicy komputerów coraz chętniej uzupełniają je o nowe karty i peryferia zewnętrzne, co powoduje, że zakres obsługi sprzętu przez system komputerowy staje się o wiele ważniejszym problemem, niż to było dawniej. Zwiększa się wybór urządzeń zaprojektowanych do spełniania standardu plug-and-play, przyłączanych do portu USB lub FireWire, dostępnych w każdym sklepie komputerowym, a nawet w co większych supermarketach. Użytkownicy oczekują, że te urządzenia po przyłączeniu do komputera będą „po prostu działały”. I słusznie.

Każdy sprzęt wymaga do działania kodu uruchamianego przez jądro systemu, nazywanego sterownikiem urządzenia. Taka sytuacja ma miejsce zarówno w jądrze systemu Microsoft Windows (tak, ten system też posiada jądro), Mac OS X, jak i jądrze Linuksa. To, że dyski CD z etykietą „Sterowniki dla systemu Windows” wypadają z każdego pudełka ze sprzętem, może sugerować, iż dane urządzenie nie będzie działać w Linuksie. W końcu w pudełku nie ma dysku z opisem „Sterowniki dla Linuksa”.

Na szczęście, sterowniki większości sprzętu są już dostępne w ramach jądra Linuksa i są zainstalowane wraz z systemem Ubuntu. Sterowniki urządzeń są wbudowane w jądro albo występują w postaci modułu jądra, ładowanego w momencie wykrycia urządzenia

(stąd popularne określenie tych modułów: **ładowane moduły jądra**). Taka sytuacja ma miejsce w znaczącej większości sprzętu, w końcu w świecie open source wystarczy jeden zdeterminowany użytkownik Linuksa, z umiejętnościami programistycznymi, posiadający sprzęt, który „nie za bardzo działa”. Taki sprzęt z reguły zaczyna pracować w stosunkowo niedługim czasie.

Choć jestem zwolennikiem oraz propagatorem Ubuntu i ogólnie Linuksa, nie mogę zaprzeczyć, że brak sterowników dla Linuksa jest czasem problemem. Do uzyskania najwyższej wydajności niektórych urządzeń, dotyczy to przede wszystkim kart graficznych i dźwiękowych, niezbędne mogą być specjalne sterowniki. Niestety, większość producentów sprzętu kieruje się przewidywaną wielkością sprzedaży opartą na analizie rynku, a obecnie rynek Windows jest od pięćdziesięciu do stu razy większy od rynku Linuksa. Z tego właśnie powodu w pudełkach z urządzeniami znajdziemy dyski CD z naklejką „Sterowniki Windows”. Poziom obsługi tych urządzeń przez sterowniki open source często pozostaje w tyle za ich odpowiednikami dla systemu Windows. Na szczęście coraz większa liczba producentów sprzętu, w takiej czy innej formie, oferuje sterowniki dla Linuksa, najczęściej w postaci archiwów dostępnych na stronach WWW producenta.

Dobra wiadomość jest taka, że sterowniki niezbędne do działania większości nowego sprzętu są dostępne z dystrybucją Ubuntu Linux. Niektóre specjalizowane sterowniki nie są udostępniane na licencji open source, inne posiadają ograniczenia licencyjne, co powoduje, że mogą być oferowane w ramach repozytoriów *restricted*, *universe* lub *multiverse*, których znaczenie szerzej opisałem w rozdziale 20. Inne sterowniki można pobrać bezpośrednio ze strony WWW producenta. W efekcie okazuje się, że większość urządzeń, wbrew obawom, „po prostu działa” w Linuksie.

W tym rozdziale omówię wykorzystanie w systemie Ubuntu różnych typów urządzeń, skupiając się na sprzęcie najczęściej wykorzystywanym z komputerami biurowymi lub laptopami. Wykrywanie sprzętu i jego obsługa w systemie Ubuntu Linux są wciąż udoskonalane. Poszukując nowego sprzętu, warto spędzić nieco czasu w wyszukiwarce WWW i sprawdzić, czy Linux obsługuje wybrane urządzenie. Generalnie, im więcej informacji, tym lepiej.

Dodawanie drukarek

Większość nowoczesnych drukarek podłącza się do komputera poprzez interfejs USB, choć niektóre wciąż wykorzystują klasyczny port równoległy. Niezależnie od interfejsu, drukarkę należy podłączyć do komputera, a następnie włączyć. W przypadku drukarki USB należy sprawdzić, czy jest ona widoczna w systemie, korzystając z polecenia `lsusb` (ang. *list USB*). Po wykonaniu tego polecenia otrzymamy szereg informacji, których przykład prezentuję poniżej:

```
$ lsusb
Bus 004 Device 001: ID 0000:0000
Bus 001 Device 001: ID 0000:0000
Bus 003 Device 001: ID 0000:0000
Bus 002 Device 004: ID 04b8:0005 Seiko Epson Corp. Stylus Printer
Bus 002 Device 001: ID 0000:0000
```


Powyższy listing dostarcza dane o tym, że do systemu podłączona została drukarka Epson Stylus.

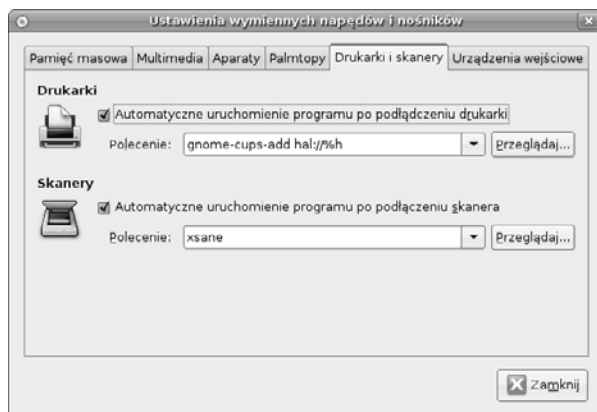


Jeśli ktoś nosi się z zamiarem kupna drukarki i chciałby zasięgnąć informacji, czy konkretna drukarka jest obsługiwana w systemie Linux, warto zajrzeć na stronę *LinuxPrinting.org*. Strona ta dostarcza informacji na temat współpracy konkretnej drukarki (bądź drukarek konkretnego producenta) z systemem Linux. Więcej informacji o drukowaniu w systemie Linux można znaleźć w rozdziale 30., „Konfiguracja serwera wydruków”.

Pozycje z menu *System/Preferencje/Napędy i nośniki wymienne* umożliwiają zdefiniowanie reakcji systemu po podłączeniu lub wykryciu nowej drukarki. W oknie *Ustawienia wymiennych napędów i nośników* należy wybrać kartę *Drukarki i skanery* (patrz rysunek 23.1)

Rysunek 23.1.

Ustawianie
właściwości
wykrywania drukarki



Po zaznaczeniu pozycji *Automatyczne uruchomienie programu po podłączeniu drukarki* w polu pojawia się wpis `gnome-cups-add hal://%h` (co oznacza wywołanie aplikacji *gnome-cups-add* z parametrem zgodnym z systemowym numerem ID urządzenia w podsystemie obsługi sprzętu HAL), co spowoduje uruchomienie kreatora dodawania nowej drukarki w systemie (patrz rysunek 23.3). Więcej informacji na temat obsługi tego okna znajduje się w dalszej części punktu, zaraz pod rysunkiem 23.3.

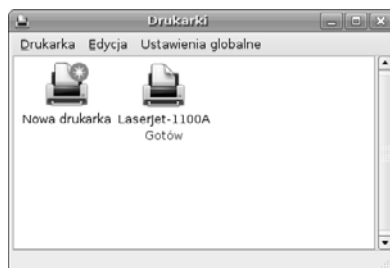
Jeśli wyłączysz opcje automatycznego uruchamiania *gnome-cups-add* w przypadku podłączenia nowej drukarki lub autodetekcja z jakiegoś powodu nie zadziała, drukarkę zawsze można zainstalować ręcznie. W tym celu należy wybrać z menu *System/Administracja/Drukarki*. Ta czynność spowoduje wyświetlenie okna przedstawionego na rysunku 23.2, które zawiera listę wszystkich aktualnie zainstalowanych w systemie drukarek, jak również ikonę *Nowa drukarka*, której można użyć do zdefiniowania i zainstalowania nowej drukarki.

Dwukrotne kliknięcie ikony *Nowa drukarka* lub wybranie pozycji z menu *Drukarka/Dodaj drukarkę* powoduje uruchomienie kreatora ułatwiającego konfigurację nowego sprzętu. Po uruchomieniu programu narzędziowego *Drukarki* nastąpi odczytanie bazy drukarek i wyświetlenie okna konfiguracji nowej drukarki, przedstawionego na rysunku 23.3.

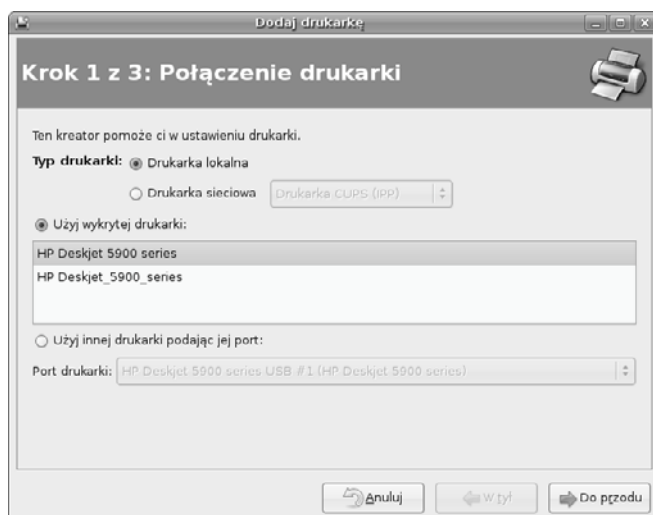
Jeżeli system Ubuntu jest w stanie automatycznie wykryć informacje o nowej drukarce, okno konfiguracji drukarki zostanie automatycznie wypełnione odpowiednimi danymi

Rysunek 23.2.

Program narzędziowy
Drukarki w systemie
Ubuntu

**Rysunek 23.3.**

Określanie drukarki
i typu połączenia



dotyczącymi podłączanej drukarki oraz sposobu jej podłączenia. Przykładowo w oknie przedstawionym na rysunku 23.3 widzimy informację, że w systemie została wykryta drukarka HP DeskJet 5900, przyłączona na porcie USB. Jeśli te informacje są prawidłowe, należy kliknąć przycisk *Do przodu*, co spowoduje przejście na następną stronę kreatora, przedstawioną na rysunku 23.4.



Jeśli drukarka nie zostanie wykryta automatycznie, należy wskazać jej model, zaznaczając opcję *Użyj innej drukarki podając jej port*, po czym z listy rozwijanej *Port drukarki* wybrać port, do którego jest przyłączona.

Jeśli drukarka zostanie automatycznie (i poprawnie) wykryta, w oknie wyświetli się odpowiedni dla niej sterownik, tak jak na rysunku 23.4. Po kliknięciu *Do przodu* otrzymamy okno *Informacje o drukarce* przedstawione na rysunku 23.5.

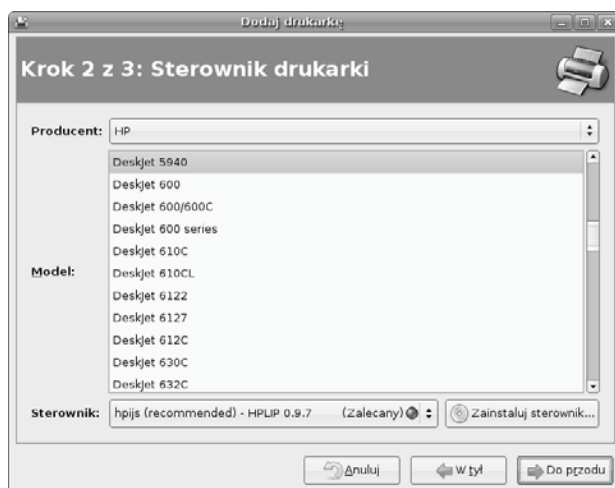


Jeśli nie został wykryty prawidłowy sterownik dla podłączonej drukarki, można go wskazać ręcznie, wybierając producenta z listy rozwijanej *Producent*, a następnie model z listy rozwijanej *Model*.

Okno dialogowe przedstawione na rysunku 23.5 pozwala na wprowadzenie istotnych (ale opcjonalnych) informacji na temat drukarki, umożliwia też modyfikację jej nazwy. Po wprowadzeniu informacji należy kliknąć przycisk *Zastosuj*, co zakończy proces definiowania parametrów drukarki. To spowoduje zamknięcie okna kreatora, a w oknie narzędzia Drukarki (patrz rysunek 23.2) pojawi się ikona symbolizująca właśnie dodaną drukarkę.

Rysunek 23.4.

Wybór sterownika drukarki

**Rysunek 23.5.**

Wprowadzanie informacji o drukarce



Aby sprawdzić działanie nowej drukarki, wystarczy kliknąć jej ikonę prawym przyciskiem myszy i wybrać z menu podręcznego funkcję *Właściwości*. To spowoduje wyświetlenie okna *Właściwości*, przedstawionego na rysunku 23.6.

Po kliknięciu przycisku *Wydrukuj stronę testową* możemy zweryfikować, czy instalacja odbyła się prawidłowo. To spowoduje umieszczenie w kolejce wydruku, a następnie wydrukowanie strony testowej drukarki. Jeśli strona nie zostanie wydrukowana, należy przejrzeć poszczególne karty okna właściwości drukarki, sprawdzając poprawność konfiguracji i informacje o producencie oraz modelu urządzenia. Wydrukowanie strony testowej jest wystarczającym sygnałem, że ustawienia są poprawne.



Jeśli drukarka nie została wykryta automatycznie lub nie uda się na niej drukować za pomocą ustawień kreatora drukarek, warto ponownie uruchomić komputer z drukarką podłączoną do systemu i włączoną do zasilania. W sporadycznych przypadkach drukarka i podsystemy odpowiedzialne za jej obsługę mogą nie być prawidłowo zainicjalizowane, jeśli urządzenie nie zostanie wykryte w trakcie procedury rozruchu systemu.

Rysunek 23.6.

Okno właściwości drukarki



Wykorzystanie skanera

Większość nowoczesnych skanerów to urządzenia USB, dość dobrze obsługiwane w Linuksie, podobnie jak starsze skanery wykorzystujące interfejs SCSI. Do obsługi skanerów system Linux wykorzystuje mechanizm SANE (ang. *Scanner Access Now Easy*) oraz związane z nim aplikacje. Stroną domową projektu SANE jest <http://www.sane-project.org>. Jeśli ktoś zastanawia się nad zakupem skanera i zainteresowany jest modelem działającym prawidłowo w systemie Linux, powinien zajrzeć na stronę <http://www.sane-project.org/sane-supported-devices.html>, na której można znaleźć łatwą w użyciu wyszukiwarkę modeli skanerów wraz z ich możliwościami.

W systemie Ubuntu domyślnie instalowane jest oprogramowanie XSane Image Scanner, wykorzystujące mechanizm SANE i oferujące graficzny interfejs użytkownika pozwalający na podgląd procesu skanowania, definiowanie opcji konfiguracyjnych skanera, wykorzystanie szybkiego podglądu obszaru skanowania na potrzeby ustawienia źródła obrazu i definiowanie różnych rozdzielczości skanowania. W dalszej części rozdziału w odniesieniu do oprogramowania XSane Image Scanner będę stosował określenie XSane.



Ten podrozdział nie jest kompletnym poradnikiem objaśniającym każdy możliwy niuans oprogramowania do obsługi skanerów dla systemów Linux. Moim celem jest przekazanie kilku porad, które pozwolą na wygodne i łatwe wykorzystywanie skanera w Ubuntu.

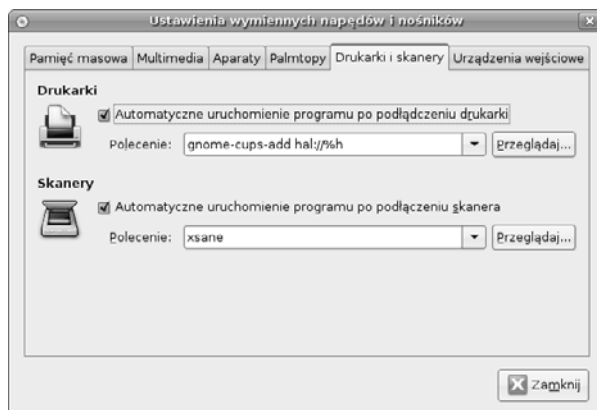
Narzędzie *System/Preferencje/Napędy i nośniki wymienne* pozwala zdefiniować reakcję systemu na zdarzenia związane z urządzeniami zewnętrznymi, dotyczy to również skanerów. Do konfiguracji zdarzeń związanych ze skanerem służy karta *Drukarki i skanery*, przedstawiona na rysunku 23.7.

Zaznaczenie opcji *Automatyczne uruchomienie programu po podłączeniu skanera* spowoduje, że po podłączeniu skanera zostanie uruchomiony program xsane, wchodzący w skład rodziny narzędzi XSane.

Jeśli z różnych względów nie chcemy, aby po podłączeniu skanera był uruchamiany jakikolwiek program, lub z jakiegoś powodu zawiedzie mechanizm autodetekcji, program XSane można uruchomić ręcznie. Aby uruchomić aplikację XSane, należy wywołać funkcję

Rysunek 23.7.

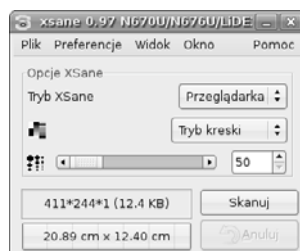
Konfiguracja
preferencji detekcji
skanera



Aplikacje/Grafika/Skaner obrazów XSane. To spowoduje wyświetlenie kilku okien, które początkowo można pozamykać, z wyjątkiem przedstawionego na rysunku 23.8, głównego okna programu XSane.

Rysunek 23.8

Podstawowe
okno dialogowe
programu XSane



Aby sprawdzić, jaki skaner został rozpoznany przez program, należy w tym oknie wywołać z menu funkcję *Plik/Informacje*, co spowoduje wyświetlenie okna przedstawionego na rysunku 23.9.

Rysunek 23.9.

Informacje programu
XSane na temat
skanera



To okno dialogowe identyfikuje model i producenta skanera wykrytego w systemie, przedstawia również podstawowe informacje na temat sterownika SANE użytego do obsługi tego urządzenia.

Aby wykonać skanowanie wstępne (podgląd o niskiej rozdzielczości, wykonywany z reguły w celu sprawdzenia czy prawidłowo ułożono skanowany materiał itp.), należy uaktywnić w menu opcję *Okno/Podgląd*, co spowoduje otwarcie okna przedstawionego na rysunku 23.10.

Rysunek 23.10.

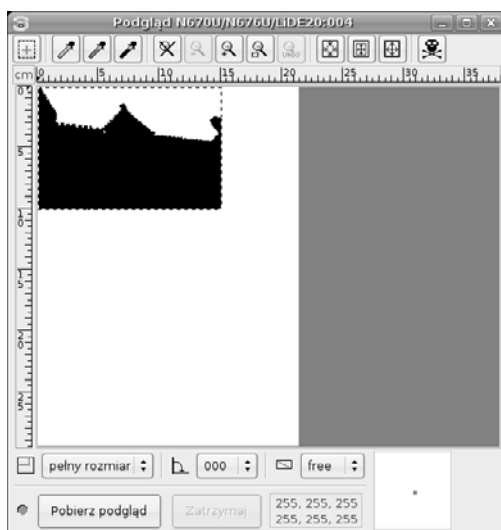
Okno podglądu obszaru skanowania



Aby uzyskać podgląd obszaru skanowania, należy kliknąć przycisk *Pobierz podgląd* w lewym dolnym rogu okna przedstawionego na rysunku 23.10. Skaner uruchomi się, wykona operację skanowania, a w oknie pojawi się podgląd, co przedstawiam na rysunku 23.11.

Rysunek 23.11.

Podgląd skanowanego obrazu



Niewiele widać, ale nie należy się niepokoić. Na razie nie modyfikowaliśmy parametrów skanera, które domyślnie ustawione są na dość niskim poziomie. Nawet na podstawie tak niewyraźnego podglądu można stwierdzić, czy układ źródła obrazu jest prawidłowy,

i zdecydować się na wykonanie skanowania. Zwykle jednak podgląd o takiej jakości będzie niewystarczający, zatem należy wrócić do głównego okna programu XSane (patrz rysunek 23.8), aby nieco zmodyfikować opcje skanowania. Najczęściej zastosowanie będą miały następujące opcje.

- ♦ **Tryb skanowania** (druga lista rozwijana od góry): dostępne opcje to *Tryb kreski* (ustawienie domyślne wykonujące skanowanie w trybie czerni i bieli), *Szarości* oraz *Kolor*. Wybór opcji *Szarości* i *Kolor* powoduje zmianę wyglądu głównego okna, bo dodaje opcje specyficzne dla tych trybów skanowania, takie jak jasność, kontrast, nasycenie itp. Na rysunku 23.12 przedstawiam okno programu XSane w trybie skanów kolorowych.

Rysunek 23.12.

Dodatkowe
elementy kontrolne
dla kolorowych
skanów



- ♦ **Rozdzielczość skanowania**: dostępny zakres rozdzielczości rozpoczyna się od 50 dpi (punktów na cal), a kończy na 3200 dpi. Górny zakres jest zależny od możliwości danego modelu skanera.

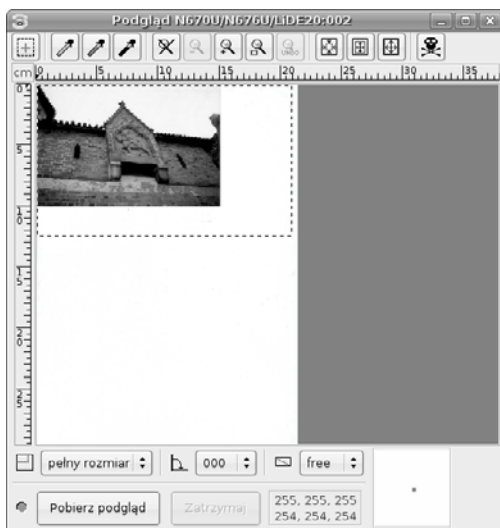
Kombinacje trybu i rozdzielczości skanowania determinują rozmiar obrazu wygenerowanego przez XSane, np. obraz w trybie kresek (czerni i bieli) o rozmiarze A4 w rozdzielczości 72 dpi będzie miał rozmiar kilkudziesięciu kilobajtów, natomiast obraz w kolorze o tej samej rozdzielczości i wymiarach będzie miał rozmiar około 1,5 MB. Ten sam dokument zeskanowany w rozdzielczości 240 dpi będzie miał rozmiar kilkunastu megabajtów. Jak widać, jakość ma swoją cenę.

Po zdefiniowaniu ustawień należy ponownie wykonać podgląd, klikając przycisk *Pobierz podgląd* w oknie podglądu, co tym razem spowoduje uzyskanie podglądu znacznie wyższej jakości (patrz rysunek 23.13).

Tym razem na podglądzie widać, że mamy do czynienia z fragmentem zabytkowej budowli. Jeśli podgląd jest zadowalający, można zamknąć jego okno i w głównym oknie programu XSane kliknąć przycisk *Skanuj*. Dokument zostanie zeskanowany zgodnie ze zdefiniowanymi ustawieniami. Po zakończeniu skanowania program wyświetli jego efekt w oknie przeglądarki przedstawionym na rysunku 23.14.

Warto pobawić się nieco, wykonując kilka skanów z różnymi ustawieniami rozdzielczości, jasności, nasycenia czy dowolnych innych parametrów dostępnych w głównym oknie

Rysunek 23.13.
 Ulepszony podgląd
 skanowanego obrazu



Rysunek 23.14.
 Wynik skanowania
 w oknie przeglądarki



Xsane. Po uzyskaniu zadowalających efektów można zapisać obraz z poziomu przeglądarki, wywołując funkcję *Plik/Zapisz obraz*, co spowoduje wyświetlenie standardowego okna zapisu pliku z możliwością określenia formatu pliku graficznego.

Informacje na temat SANE i ogólnie na temat skanowania w Linuksie można znaleźć w internecie pod następującymi adresami:

- ♦ Linux Scanner HOWTO: <http://tldp.org/HOWTO/Scanner-HOWTO/index.html>;
- ♦ SANE FAQ: <http://www.xs4all.nl/~ljm/SANE-faq.html>;
- ♦ Strona domowa projektu SANE: <http://www.sane-project.org>.



Jeśli system Ubuntu ma problem ze zlokalizowaniem skanera lub konieczne jest podanie nazwy pliku urządzenia wykorzystywanego przez skaner, wówczas należy zainstalować pakiet *sane-utils* (np. za pomocą menedżera pakietów Synaptic), w którym znajduje się wygodne narzędzie *sane-find-scanner*, pomocne podczas wyszukiwania skanerów SCSI i USB. Po zainstalowaniu pakietu *sane-utils* w systemie znajdziemy podręczniki użytkownika dla *sane-scsi* i *sane-usb*, w których można odszukać porady i sugestie pomocne w rozwiązywaniu problemów ze skanerami tych typów.

Wykorzystanie zewnętrznych dysków i napędów CD/DVD

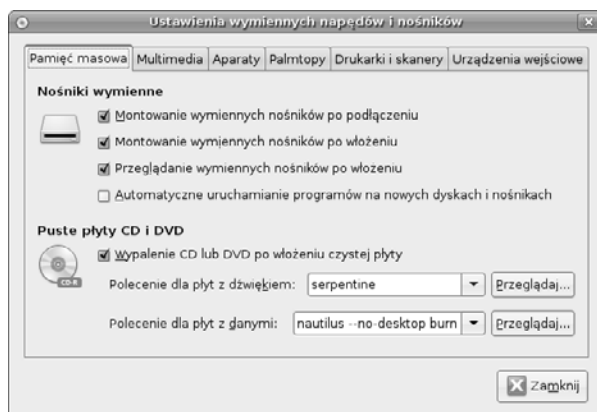
Przenośne napędy USB i FireWire są obecnie bardzo popularne, podobnie jak pamięci flash z interfejsem USB (widziane przez system jako dyski twarde mniejszej pojemności). Te przenośne, podłączane w locie urządzenia znacznie upraszczają wykonywanie kopii zapasowych czy przenoszenie znacznych ilości danych między komputerami. Zwykle te przenośne urządzenia są prawidłowo wykrywane przez system Ubuntu Linux, który reaguje na zdarzenie przyłączenia urządzenia do systemu w sposób skonfigurowany w systemie, o czym opowiem w kolejnym punkcie rozdziału. W następnym punkcie, „Rozwiązywanie problemów z automatycznym wykrywaniem urządzeń”, omówię samodzielną konfigurację systemu w taki sposób, aby prawidłowo wykrywał urządzenia nośników przenośnych, oraz wyjaśnię, jak formatować urządzenia, aby mogły być użyte w systemie.

Konfiguracja automatycznego wykrywania urządzeń

W rozdziale 13. zawarłem przegląd funkcji narzędzia *System/Preferencje/Napędy i nośniki wymienne* w celu konfiguracji zachowania systemu po podłączeniu i wykryciu przenośnego nośnika danych. Po uruchomieniu narzędzie to wyświetla kartę *Pamięć masowa*, na której można skonfigurować zachowanie systemu po wykryciu zewnętrznego nośnika danych. Zawartość karty *Pamięć masowa* jest przedstawiona na rysunku 23.15.

Rysunek 23.15.

Konfiguracja preferencji zewnętrznych nośników pamięci masowej



Sekcja *Nośniki wymienne* na tej karcie pozwala określić zachowanie systemu w reakcji na wykrycie wymiennych nośników USB i FireWire oraz w przypadku włożenia do napędu czystej nagrywanej płyty CD lub DVD. Mamy tu do dyspozycji następujące ustawienia.

- ♦ **Montowanie wymiennych nośników po podłączeniu:** partycje na nośniku wymiennym zostaną automatycznie podłączone po podłączeniu do systemu.
- ♦ **Montowanie wymiennych nośników po włożeniu:** partycje na nośniku wymiennym zostaną automatycznie podłączone po włożeniu do napędu (dyski CD i DVD).
- ♦ **Przeglądanie wymiennych nośników po włożeniu:** po włożeniu lub podłączeniu nośnika wymiennego zostanie automatycznie uruchomiona przeglądarka plików Nautilus.
- ♦ **Automatyczne uruchamianie programów na nowych dyskach i nośnikach:** zaznaczenie tej opcji spowoduje, że zostanie automatycznie uruchomiony skrypt *autorun.sh* lub *autorun.inf* znalezione na nośniku.

Sekcja *Puste płyty CD i DVD* w tym oknie pozwala określić zachowanie systemu po włożeniu czystego, zapisywalnego nośnika CD lub DVD do napędu. Na karcie *Multimedia* sekcja *Płyty CD-Audio* możemy określić zachowanie systemu po włożeniu do napędu płyty audio CD. Analogicznie sekcja *Płyty DVD video* pozwala określić zachowanie systemu dla nośników DVD zawierających materiał filmowy.

Rozwiązywanie problemów z automatycznym wykrywaniem urządzeń

Po podłączeniu do systemu wymiennego nośnika danych, niezależnie od tego, czy jest to napęd CD lub DVD, urządzenie pamięci flash (obsługiwane przez system Linux tak jak dysk twardy SCSI), jądro automatycznie zapisuje komunikaty związane z wykryciem tego urządzenia. Jeśli nowo przyłączone urządzenie nie zostanie automatycznie zamontowane w systemie, należy odnaleźć nazwę pliku urządzenia przydzielonego przez system i spróbować zamontować je ręcznie.

Oto przykład komunikatów wypisanych przez jądro w wyniku przyłączenia pamięci flash USB:

```
kernel: usb 4-1: new high speed USB device using ehci_hcd and address 5
kernel: scsi3 : SCSI emulation for USB Mass Storage devices
kernel: Vendor: WDC           Model: WD2500JB-00GVC0   Rev: 08.0
kernel: Type:   Direct-Access              ANSI SCSI revision: 02
kernel: 3:0:0:0: Attached scsi generic sgl type 0
kernel: Driver 'sd' needs updating - please use bus_type methods
kernel: SCSI device sda: 488397168 512-byte hdwr sectors (250059 MB)
kernel: SCSI device sda: 488397168 512-byte hdwr sectors (250059 MB)
kernel: sda: sda1
kernel: sd 3:0:0:0: Attached scsi disk sda
```



Pominałem pewne nadmiarowe informacje z każdego wiersza komunikatów, aby zmieściły się w poziomie na stronie druku.

Pierwsze sześć wierszy tego wyniku informuje o sterowniku USB i sterowniku emulacji SCSI wykorzystywanych do obsługi tego urządzenia. Ostatnie cztery wiersze są najbardziej interesujące, zawierają bowiem informację o plikach urządzeń przydzielonych temu nośnikowi USB. Znajdziemy tu informację o tym, że urządzenie zostało przyłączone jako

pierwsze urządzenie SCSI, *sda*, i że na urządzeniu wykryto jedną partycję, *sda1*. Informację tę można zweryfikować, wykorzystując program *fdisk* do odczytania parametrów urządzenia, tak jak w poniższym przykładzie:

```
$ fdisk -l /dev/sda
Disk /dev/sda: 250.0 GB, 250059350016 bytes
255 heads, 63 sectors/track, 30401 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
   Device Boot      Start         End      Blocks   Id  System
/dev/sda1  *           1         30401     244196001   c   W95 FAT32 (LBA)
```

Ten wynik potwierdza, że urządzenie */dev/sda* posiada partycję */dev/sda1*, która zawiera system plików *vfat* - informacja ta przyda się podczas ręcznego montowania partycji. Można tego dokonać za pomocą następującego polecenia:

```
$ sudo mount -t vfat /dev/sda1 /mnt
```

Komunikaty jądra towarzyszące przyłączaniu do tego samego systemu zewnętrznego napędu DVD+RW przy pomocy interfejsu USB były następujące:

```
kernel: usb 4-2: new high speed USB device using ehci_hcd and address 4
kernel: Initializing USB Mass Storage driver...
kernel: scsi2 : SCSI emulation for USB Mass Storage devices
kernel: usbcore: registered new driver usb-storage
kernel: USB Mass Storage support registered.
kernel: Vendor: _NEC      Model: DVD+RW ND-1100A   Rev: 1.A3
kernel: Type:   CD-ROM                      ANSI SCSI revision: 00
kernel: sr0: scsi3-mmc drive: 40x/40x writer cd/rw xa/form2 cdda tray
kernel: sr 2:0:0:0: Attached scsi generic sg0 type 5
```

W tym przykładowym wyniku pierwsze siedem wierszy informuje o sterownikach USB i SCSI wykorzystywanych do obsługi napędu DVD+RW USB. Ponownie, najbardziej interesujące są dwa ostatnie wiersze. Informują o tym, że napęd został skojarzony z urządzeniami */dev/sr0* i */dev/sg0*. Szybkie sprawdzenie tych plików za pomocą polecenia `ls -l` pozwala stwierdzić, że jedno z nich jest urządzeniem blokowym (*/dev/sr0*), natomiast drugie — znakowym (*/dev/sg0*), pozwalającym na bezpośredni odczyt umieszczonych w nim nośników.

Jak widać, komunikaty jądra są bardzo użyteczne do identyfikacji nowo przyłączonego sprzętu oraz związanych z nim plików urządzeń. Więcej informacji na temat narzędzi z graficznym interfejsem użytkownika, przydatnych w pozyskiwaniu podobnych informacji, można znaleźć w ostatnim podrozdziale tego rozdziału, „Rozwiązywanie problemów z urządzeniami za pomocą narzędzi graficznych”.

Dodawanie wewnętrznych dysków i napędów CD/DVD

W tym podrozdziale podaję informacje na temat dodawania do systemu napędów wewnętrznych. Obecnie w komputerach PC i laptopach najczęściej stosowane są dyski twarde z interfejsem EIDE/ATA (ang. *Enhanced Integrated Drive Electronics/Advanced Technology Attachment*) oraz SATA (ang. *Serial Advanced Technology Attachment*). Z powodu

popularności napędów SATA dyski EIDE/ATA są często określane skrótem PATA (ang. *Parallel Advanced Technology Attachment*), co podkreśla główną różnicę między interfejsami (*serial*, czyli interfejs szeregowy i *parallel*, czyli interfejs równoległy) i pozwala wykorzystać zbliżoną nazwę do urządzeń o podobnej funkcjonalności. Kolejną opcją to urządzenia SCSI (ang. *Small Computer System Interface*), ale są znacznie mniej powszechne w systemach klasy desktop, choć nadal bardzo popularne w systemach serwerowych. Większość napędów CD i DVD to urządzenia EIDE/ATA, wykorzystujące standard ATAPI, choć można również napotkać urządzenia CD i DVD z interfejsem SCSI i SATA.

Aby zainstalować w komputerze urządzenia tych typów, należy zamknąć system i zdjąć zewnętrzną pokrywę obudowy, żeby uzyskać dostęp do *zatok napędów*, służących do montażu dysków twardych i napędów CD/DVD.

W kolejnych punktach wyjaśnię, w jaki sposób dodawać do systemu napędy różnych typów i jak identyfikować przydzielone im pliki urządzeń, oraz przedstawię ogólne informacje na temat sposobu tworzenia na tych urządzeniach systemu plików.

Dodawanie do systemu napędów EIDE/ATA

Po wyłączeniu systemu i zdjęciu pokrywy należy zlokalizować przewody IDE przyłączone do gniazd EIDE płyty głównej (lub, w rzadszych przypadkach, do karty kontrolera IDE lub EIDE). Przewody IDE to płaskie, 40-żyłowe taśmy zakończone po obu stronach prostokątnymi wtyczkami. Większość komputerów PC posiada dwa interfejsy: podstawowy (*primary*, czyli IDE-1) i drugorzędny (*secondary*, czyli IDE-2). Przewód przyłączony do każdego interfejsu może służyć do podłączenia jednego lub dwóch dysków twardych lub napędów CD/DVD. Jeśli do jednego przewodu przyłączone są dwa urządzenia, muszą być skonfigurowane jako, odpowiednio, nadrzędne (*master*) i podrzędne (*slave*); najczęściej do tego celu stosuje się przełączniki lub zworki łączące odpowiednie piny. Urządzenie nadrzędne jest identyfikowane jako pierwsze na danym interfejsie IDE, podrzędne jest podłączane do systemu jako drugie.

Aby dodać do systemu nowy napęd, należy odpowiednio skonfigurować zworki na pinach urządzenia, identyfikując je jako nadrzędne lub podrzędne, tak aby nie konkurowały wzajemnie o sterowanie interfejsem IDE. W tym celu należy zapoznać się z konfiguracją urządzeń podłączonych już do interfejsów IDE w systemie. Można tego dokonać w sposób automatyczny, weryfikując informacje podsystemu BIOS lub samodzielnie odczytując ustawienia przełączników czy zwerek na urządzeniach. Skupię się na tej drugiej metodzie, ponieważ zaleciłem wyłączenie komputera, a ustawienia PC BIOS oraz boot monitora w systemach Macintosh i SPARC cechują się zasadniczymi różnicami, których analiza wykracza poza tematykę tego rozdziału.

Śledząc przewody i analizując ustawienia zwerek, można łatwo zidentyfikować zasady przyłączenia urządzeń. W tym celu należy prześledzić każdy przewód IDE od płyty głównej do napędu. Dysk twardy, z którego jest uruchamiany system, jest nazywany napędem rozruchowym i z reguły przyłącza się go jako urządzenie nadrzędne (*master*) do pierwszego kontrolera (IDE-1). Jeśli do tego samego interfejsu (przewodu IDE) przyłączony jest też napęd CD/DVD, powinien być skonfigurowany jako podrzędny (*slave*). Jeśli jednak napęd CD/DVD jest przyłączony jako jedyny do drugiego interfejsu (IDE-2), zapewne jest skonfigurowany jako nadrzędny.



Jeśli do jednego interfejsu IDE były wcześniej podłączone dwa urządzenia i system działał prawidłowo, nie warto psuć tej konfiguracji, ponieważ oznacza to, że system był skonfigurowany jak należy.

Gdy dowiemy się, w jaki sposób podłączone są dyski, należy odpowiednio skonfigurować zworki na przyłączanym napędzie, aby nie spowodować konfliktu z urządzeniami już zainstalowanymi w systemie. Instrukcja konfiguracji napędu w charakterze urządzenia nadrzędnego lub podrzędnego z reguły jest nadrukowana na samym napędzie.



Przy odczycie informacji o zworkach należy zachować czujność, ponieważ są drukowane z oznaczeniem numerów pinów od lewej do prawej, czasem zdarza się, że ułożenie pinów zgodnie z instrukcją wymaga odwrócenia napędu „do góry nogami”, czyli numeracja pinów jest w stylu „od prawej do lewej”, a zatem odwrotnie w stosunku do instrukcji. Należy po prostu poprawnie zidentyfikować pierwszy pin.

Po ustawieniu zworek dysku trzeba zamontować go w obudowie i podłączyć przewód IDE oraz zasilanie. W tym momencie można włączyć zasilanie komputera, uruchomić Linuksa i przejść do punktu „Lokalizowanie, partycjonowanie i formatowanie dysków twardych”, w dalszej części rozdziału. Jeśli po dodaniu nowego urządzenia występują problemy z uruchomieniem systemu, należy zapoznać się z treścią punktu „Rozwiązywanie problemów z uruchamianiem systemu po dodaniu nowych urządzeń”.



Najpowszechniejszym źródłem problemów podczas podłączania napędów EIDE/ATA do istniejącego systemu jest nieprawidłowe ustawienie zworek. Jeśli nowy napęd będzie miał takie same ustawienie zworek (*master* lub *slave*), co istniejący napęd, system z reguły nie zobaczy żadnego z tych urządzeń. W systemie PC można to sprawdzić jeszcze przed uruchomieniem systemu, wystarczy wejść w tryb konfiguracji BIOS-u i zobaczyć, czy urządzenia IDE zostały wykryte prawidłowo.

Dodawanie napędów SATA

Napędy SATA znacznie łatwiej przyłączać do systemu, w porównaniu z napędami EIDE/ATA czy SCSI, przede wszystkim dzięki temu, że nie ma potrzeby ustawiania żadnych zworek. Dyski SATA są identyfikowane w systemie w oparciu o numer portu SATA, do którego zostały przyłączone. W większości płyt głównych występują dwa lub cztery porty SATA w postaci niewielkich gniazd SATA w kształcie małej litery L. Dostępne są również karty kontrolerów SATA, oferujące cztery i więcej portów.

Standard SATA posiada wiele zalet w porównaniu z tradycyjnym standardem EIDE/ATA (który dla uproszczenia będę nazywał napędami PATA, od „parallel ATA”). Przede wszystkim każdy napęd SATA jest przyłączany do kontrolera osobnym przewodem, dzięki czemu nie ma problemów z ustawieniami master/slave, typowymi dla interfejsu PATA. Wydajność dysków SATA II i PATA jest zbliżona, lecz dedykowany przewód każdego napędu SATA pozwala osiągnąć wyższe średnie wydajności w zastosowaniach wykorzystujących jednocześnie większą liczbę napędów. Przewody SATA mogą mieć długość do 1 metra, natomiast przewody PATA mogą mieć długość maksymalnie 46 centymetrów. Przewody SATA mają znacznie mniejszy przekrój w porównaniu z taśmami PATA, dzięki czemu montaż jest prostszy, a przewody mogą być ułożone w sposób bardziej uporządkowany, co może mieć niemałe znaczenie w ciasnych obudowach komputerów. Największą zaletą standardu SATA jest to, że dyski SATA można odłączać w czasie pracy

komputera, zatem doskonale nadają się do pracy w konfiguracjach RAID lub systemach o podwyższonej niezawodności.

Aby dodać napęd SATA do systemu (wyłączonego!), wystarczy zamontować go w dostępnej zatoce na dyski twarde, podłączyć do dostępnego portu SATA i wpiąć przewód zasilający; w zależności od modelu dysku może to być tradycyjny przewód stosowany również w napędach PATA lub specjalny przewód zasilający SATA. W tym momencie można uruchomić komputer w systemie Linux, po czym zastosować się do instrukcji zawartych w punkcie „Lokalizowanie, partycjonowanie i formatowanie dysków twardych”. Jeśli po dodaniu urządzenia występują problemy z uruchomieniem komputera, należy zastosować się do porad opisanych w punkcie „Rozwiązywanie problemów z uruchamianiem systemu po dodaniu nowych urządzeń”.



Niektóre płyty główne zawierają dwa różne zestawy portów SATA, jeden na potrzeby sprzętowej obsługi RAID, drugi na potrzeby zwykłego trybu SATA. Nie należy używać portów do sprzętowej obsługi RAID, chyba że dany kontroler RAID jest poprawnie obsługiwany w Linuksie no i — oczywiście — zależy nam na wykorzystaniu sprzętowego mechanizmu RAID. Z reguły preferuję programową obsługę RAID oferowaną w Linuksie, ponieważ sprzętowe rozwiązania RAID są ściśle zależne od modelu kontrolera. Oznacza to, że w przypadku awarii płyty głównej należy wymienić ją na identyczny model lub inny, ale zawierający identyczny kontroler RAID. W przeciwnym razie, dane zapisane na dyskach sprzętowej macierzy RAID nie zostaną rozpoznane przez inny kontroler RAID.

Dodawanie napędów SCSI

Dyski twarde SCSI od lat były stosowane w serwerach (głównie RAID) oraz w systemach o podwyższonej dostępności. Głównym powodem tej popularności była większa wydajność (napędy SCSI są szybsze od napędów SATA II), obsługa trybu hot-swap (możliwości przyłączania i odłączania napędu podczas pracy systemu) i podwyższona kontrola jakości (dzięki czemu niektórzy producenci oferują dłuższe gwarancje na napędy SCSI niż na ich odpowiedniki z interfejsem PATA czy SATA). Wadą tego rozwiązania jest wyższy koszt napędów SCSI w porównaniu z napędami SATA II, które oferują zbliżone możliwości (np. hot-swap). Tradycyjnie obsługę SCSI posiadają jedynie płyty główne z najwyższej półki, przez co w większości systemów kontrolery SCSI występują w postaci kart rozszerzeń.

Złącza i przewody SCSI są dość różnorodne, w zależności od standardu SCSI i liczby obsługiwanych urządzeń. Przez lata powstało wiele typów złączy SCSI, co można dość luźno powiązać z różnymi wersjami standardu SCSI i odmiennymi charakterystykami wydajności. Najpopularniejszymi typami są złącza równoległe typu Centronics oraz IDC-50 (stosowane z 8-bitowymi kontrolerami i urządzeniami SCSI-1), DB-50 (stosowane z 8-bitowymi kontrolerami i urządzeniami Fast SCSI-2) oraz DB-68 i SCA-80 (stosowane z 16-bitowymi kontrolerami i urządzeniami Fast/Wide SCSI-3). Kontrolery SCSI-1 i SCSI-2 pozwalają na przyłączenie do siedmiu urządzeń na każdy kontroler, natomiast kontrolery SCSI-3 i nowsze pozwalają przyłączać do piętnastu urządzeń na kontroler.

Każde urządzenie SCSI w systemie musi być zidentyfikowane w sposób unikalny z wykorzystaniem numeru ID. W większości przypadków występuje konieczność ustawienia tego numeru za pomocą zworek na urządzeniu, jednak w droższych kontrolerach wielo-

portowych istnieje możliwość automatycznego przydzielenia numerów ID urządzeniom w oparciu o port, do którego przyłączone jest urządzenie. Łańcuch urządzeń SCSI (napędy można łączyć ze sobą szeregowo, tzn. kolejne urządzenie jest przyłączane do poprzedniego) musi być zakończony terminatorem, umieszczonym za ostatnim urządzeniem. Istnieje również możliwość automatycznej terminacji magistrali SCSI przez kontroler.

Aby dodać urządzenie SCSI do wyłączonego systemu, należy określić jego numer SCSI ID, zamontować urządzenie w zatoce, przyłączyć do kontrolera SCSI lub do jednego z urządzeń w łańcuchu i podłączyć zasilanie (tradycyjny przewód zasilający). Następnie można uruchomić system Linux i zastosować się do porad opisanych w punkcie „Lokalizowanie, partycjonowanie i formatowanie dysków twardych”. Gdy wystąpią problemy z uruchomieniem systemu po dodaniu nowego urządzenia, warto zajrzeć do kolejnego punktu, w którym podam kilka przydatnych porad.

Rozwiązywanie problemów z uruchamianiem systemu po dodaniu nowych urządzeń

Często zdarza się sytuacja, że dodanie do systemu nowego urządzenia uniemożliwia jego uruchomienie. Zwykle problem sprowadza się do ustawień BIOS lub innych mechanizmów definiujących kolejność przeglądania urządzeń w celu zainicjowania rozruchu systemu. W kilku kolejnych punktach omówię najczęściej spotykane problemy oraz sposoby ich rozwiązania.

Zmiany w BIOS-ie dotyczące dysku twardego i kolejności sprawdzania urządzeń

W większości BIOS-ów można znaleźć sekcję Boot lub Boot Settings, w której znajdują się opcje dysków twardych i urządzeń używanych do rozruchu systemu (często noszą one nazwy *Hard Disk Drives* i *Boot Device Priority*). Ekran *Hard Disk Drives* z reguły zawiera listę wszystkich dysków twardych dostępnych w systemie. Pierwszy dysk twardy wymieniony na tym ekranie jest również domyślnym dyskiem w sekwencji rozruchu systemu na ekranie *Boot Device Priority* (czyli ma najwyższy priorytet rozruchu). Należy upewnić się, że pierwszy dysk twardy w tej sekwencji to ten sam dysk, na którym znajduje się system operacyjny. Trzeba też upewnić się, że ten dysk twardy jest uwzględniony na ekranie *Boot Device Priority* i elementy w ramach tego ekranu występują we właściwej kolejności. Wiele ekranów *Boot Device Priority* zawiera jedynie dwie lub trzy pozycje na potrzeby urządzeń rozruchowych, sam w dwóch pierwszych konfiguruję *Floppy Disk* i *CD-ROM*. Niezależnie od tego typu decyzji, należy upewnić się, że dysk twardy zawierający system operacyjny znajduje się na tej liście i występuje w odpowiedniej kolejności w stosunku do pozostałych urządzeń.

Problemy po dodaniu urządzeń IDE do systemu wykorzystującego dyski twarde SATA

Większość systemów jest skonfigurowana w taki sposób, że w poszukiwaniu dysku rozruchowego sprawdzają urządzenia IDE przed urządzeniami SATA. Jeśli do komputera zawierającego dysk twardy SATA z systemem operacyjnym dodamy dysk IDE, należy

sprawdzić, czy sekwencja rozruchowa jest prawidłowa. Spotkało mnie dość zabawne zdarzenie, gdy do systemu z dyskiem SATA dodałem dysk IDE, z którego zapomniałem usunąć poprzednią zawartość, a po rozruchu zdziwiłem się, że zamiast ulubionego Ubuntu uruchomiła się stara wersja dystrybucji SUSE.

Problemy związane ze zmianą nazwy napędu SATA

Dodanie dysku SATA do istniejącego systemu może spowodować problemy z rozruchem i to z kilku przyczyn. Najdziwniejsza z nich wiąże się ze sposobem rozpoznawania i nazywania urządzeń SATA. Dyski SATA w Linuksie wykorzystują emulację SCSI (dlatego ścieżki plików urządzeń rozpoczynają się od `/dev/sd`), natomiast litera następująca w nazwie pliku urządzenia po `sd` jest uzależniona od sekwencji inicjalizacji urządzeń SATA. Jeśli osoba konfigurująca system działała w sposób systematyczny, rozruchowy (główny) dysk twardy będzie przyłączony do pierwszego kontrolera, dzięki czemu zawsze będzie obsługiwany jako `/dev/sda`. Jeśli dysk twardy został przyłączony w sposób nieuważny, do „pierwszego lepszego” kontrolera (np. o numerze 4), to dopóki w systemie nie znajdą się dodatkowe dyski, wszystko będzie działało prawidłowo — jednemu urządzeniu SATA zostanie przypisany plik urządzenia `/dev/sda`. Sytuacja zmieni się po podłączeniu kolejnego dysku do innego kontrolera (o mniejszym numerze). To jemu zostanie przypisany plik `/dev/sda`, a bieżący dysk twardy będzie dostępny jako `/dev/sdb`. Przypinanie kolejnych urządzeń spowoduje, że za każdym razem litera pliku urządzenia dysku SATA będzie się zmieniała. Tego typu błąd w montażu przewodów w gniazdach na płycie głównej wprowadza również zamieszanie w konfiguracji BIOS, co zostało omówione w poprzednim punkcie, a z całą pewnością mocno przeszkadza w konfiguracji programu rozruchowego GRUB.

W przypadku wystąpienia problemu tego typu mamy dwa możliwe rozwiązania.

- ♦ Otworzyć obudowę i podłączyć dysk twardy w prawidłowy sposób. To da gwarancję, że dysk twardy będzie zawsze widziany w systemie jako `/dev/sda`. Zmiany w BIOS-ie mogą nadal być konieczne, ale nie będzie trzeba modyfikować żadnych systemowych plików konfiguracyjnych.
- ♦ Zmodyfikować ustawienia programu rozruchowego GRUB, uwzględniając zmiany w nazewnictwie dysku.

Pierwsza z tych opcji jest preferowana, chyba że mamy naprawdę dobry powód, aby zmodyfikować konfigurację programu GRUB. Jeśli ktoś zdecyduje, że druga opcja jest dla niego lepsza i chce zmodyfikować program GRUB, powinien zajrzeć do kolejnego punktu.

Modyfikacja konfiguracji programu rozruchowego GRUB

Jeśli pojawi się konieczność modyfikacji ustawień menedżera rozruchu GRUB, w pierwszej kolejności należy poprawić sekwencję dysków rozruchowych w BIOS-ie komputera, co zostało omówione w punkcie „Zmiany w BIOS-ie dotyczące dysku twardego i kolejności sprawdzania urządzeń”. Ta operacja umożliwi rozruch z istniejącego dysku twardego, co prawie na pewno zakończy się niepowodzeniem, ponieważ GRUB usiłuje załadować jądro systemu np. z partycji `/dev/sda1`, która zapewne będzie istniała w systemie, lecz tym razem może zawierać zupełnie inne dane niż w poprzedniej konfiguracji sprzętowej.

Aby tymczasowo zmodyfikować ustawienia menedżera GRUB w celu naprawienia tego problemu, należy wykonać następujące czynności.

1. Po załadowaniu BIOS-u należy nacisnąć klawisz *Esc* w celu wyświetlenia menu rozruchowego programu GRUB.
2. Nacisnąć klawisz *E*, aby wyświetlić polecenia związane z domyślną pozycją rozruchową.
3. Używając klawiszy strzałek, ustawić kursor na pozycji zawierającej słowo kluczowe `kernel`.
4. Nacisnąć klawisz *E*, aby uaktywnić tryb edycji, a następnie za pomocą klawiszy strzałek przejść do nazwy partycji zawierającej jądro Linuksa (w ramach parametru `root=`). Gdy np. dysk początkowo był widoczny jako `/dev/sda`, a teraz będzie widziany przez system jako `/dev/sdd`, główna partycja dysku będzie identyfikowana przez `/dev/sdd1`, a nie `/dev/sda1`, co należy odpowiednio zmodyfikować w parametrze `root=`.



Jeżeli system już wcześniej zawierał kilka dysków i na wszystkich znajdowały się używane przez niego partycje wówczas problem z uruchomieniem może istnieć nadal. Dzieje się tak ponieważ nazwy partycji zapisane w pliku `/etc/fstab` będą niezgodne z ich nowym nazewnictwem. W takim przypadku, oprócz wprowadzenia powyższych modyfikacji, konieczne może okazać się dopisanie słowa kluczowego `single` na końcu wiersza zawierającego słowo kluczowe `kernel`, co spowoduje uruchomienie systemu w trybie jednego użytkownika i umożliwi edycję pliku `/etc/fstab`.

5. Aby uruchomić system z zastosowaniem zmodyfikowanej (tymczasowo) konfiguracji programu GRUB, należy nacisnąć klawisz *B*.

Po prawidłowym uruchomieniu systemu trzeba za pomocą edytora tekstu wprowadzić te zmiany na stałe w konfiguracji programu GRUB, czyli w pliku `/boot/grub/menu.lst`. Oprócz definicji danej pozycji, w menu programu GRUB należy zmodyfikować wartości szablonu wykorzystywanego do automatycznego tworzenia pozycji w menu GRUB podczas instalacji nowej wersji jądra.

Trzeba również sprawdzić zawartość pliku `/etc/fstab`, aby upewnić się, czy zdefiniowane w nim systemy plików są zgodne z obecnym nazewnictwem urządzeń. Często zdarza się, że system uruchamia się bez prawidłowej inicjalizacji przestrzeni wymiany lub bez zamontowania niektórych systemów plików z powodu niewłaściwych definicji w pliku `/etc/fstab`. Warto zwrócić uwagę, że partycja główna jest już zamontowana w trakcie procesu rozruchu, ponieważ to z niej odczytywana jest zawartość pliku `/etc/fstab`.

Problemy po dodaniu napędów SATA do systemu SCSI

Kolejny dość powszechny problem może pojawić się po przyłączeniu dysku SATA do systemu zawierającego już dyski SCSI. Dyski SATA w systemie Linux wykorzystują pliki urządzeń SCSI, zatem otrzymują nazwy z tego samego zakresu nazw, co urządzenia SCSI. Dyski SATA są inicjalizowane wcześniej od dysków SCSI na etapie inicjalizacji jądra. Połączenie tych dwóch faktów w konsekwencji oznacza, że dodanie jednego lub większej liczby dysków SATA do systemu zawierającego już dyski SCSI może spowodować zmianę nazw plików urządzeń obsługujących wspomniane dyski SCSI. Jeśli np.

w systemie znajduje się dysk SCSI `/dev/sda` i do systemu zostaną dodane dwa dyski SATA, wtedy dyskowi SCSI zostanie nadana nazwa `/dev/sdc`. Ta sytuacja wiąże się z dokładniej tymi samymi problemami z ustawieniami sekwencji rozruchowej w BIOS-ie oraz konfiguracją menedżera rozruchu GRUB i nazwami partycji w `/etc/fstab`, które zostały omówione w punkcie „Problemy związane ze zmianą nazwy napędu SATA”. Tym razem nie uda się ich jednak poprawić przez przepinanie przewodów dysków i należy zastosować się do zaleceń opisanych w punkcie „Modyfikacja konfiguracji programu rozruchowego GRUB”.

Lokalizowanie, partycjonowanie i formatowanie dysków twardych

Po ponownym uruchomieniu komputera należy odszukać nowo dodany dysk twardy i po prostu wykorzystać oferowaną przez niego przestrzeń. W tym celu należy zidentyfikować pliki urządzeń związane z urządzeniem fizycznym.

W kolejnych trzech punktach omawiam wykorzystanie narzędzi graficznych oraz tekstowych służących do lokalizowania dysków twardych oraz tworzenia i formatowania nowych partycji.



Gdy powstawała ta książka, opisywane tu graficzne narzędzie *Dyski*, ani jakiegokolwiek graficzne narzędzie instalowane domyślnie, nie obsługiwało opcji podziału dysków na partycje. Przyszłe wersje narzędzia *Dyski* powinny obsługiwać tę funkcję, dodatkowo trwa dyskusja na temat udostępnienia do instalacji narzędzia *partman*, czyli modułu dostępnego w ramach instalacji Ubuntu (którego obecnie nie można wykorzystać w zainstalowanym już systemie). Niestety, obecnie nie jest dostępna żadna z tych opcji, zatem jesteśmy skazani na wykorzystanie narzędzi trybu tekstowego, omówionych w punkcie „Wykorzystanie narzędzi trybu tekstowego do partycjonowania i formatowania dysków”.

Wykorzystanie narzędzia graficznego do analizy i formatowania dysków

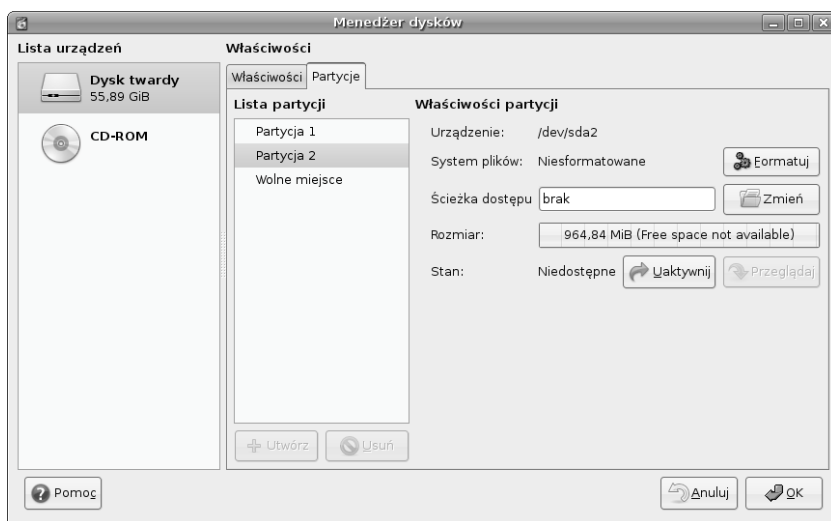
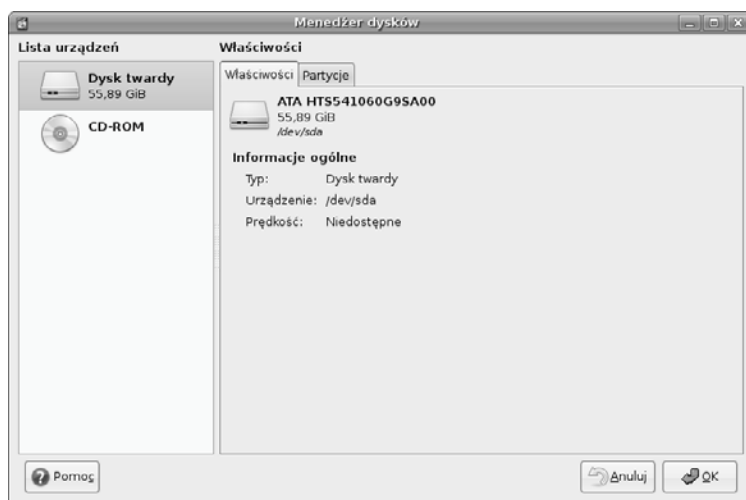
Narzędzie *Dyski*, instalowane domyślnie w systemie Ubuntu, pozwala przeglądać w trybie graficznym strukturę dysków twardych zainstalowanych w systemie. Można dzięki niemu w łatwy sposób zapoznać się z podstawowymi informacjami, takimi jak pojemność, dostępne partycje czy też pliki urządzeń związane z tymi urządzeniami i ich partycjami. Za pomocą tego programu można również formatować dyskiety i istniejące partycje dysków twardych.

Uruchamiamy narzędzie z menu *System/Administracja/Dyski*. Po wpisaniu hasła wyświetli się główne okno programu, przedstawione na rysunku 23.16.

W panelu po lewej stronie można znaleźć listę wszystkich dostępnych nośników danych. Po zaznaczeniu dowolnej pozycji w prawym panelu zostaną wyświetlone informacje o właściwościach wybranego urządzenia. Wśród tych informacji można znaleźć dane o pojemności nośnika oraz ścieżkę pliku urządzenia służącego do obsługi tego nośnika danych. Karta *Partycje* zawiera podsumowanie informacji o podziale urządzenia na partycje, co przedstawiam na rysunku 23.17.

Rysunek 23.16.

Informacje o dysku
twardym w programie
Dyski

**Rysunek 23.17.** Informacje o partycjach w programie Dyski

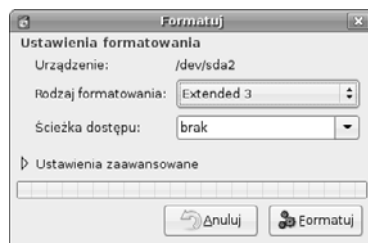
Aby uzyskać informacje na temat konkretnej partycji, należy wybrać jej nazwę z listy wyświetlonej w prawym panelu. Przykładowo na rysunku 23.17 widzimy informacje na temat partycji `/dev/sda2`, zidentyfikowanej aktualnie jako niesformatowana. Aby ją sformatować, należy wybrać partycję i kliknąć przycisk *Formatuj*, a zostanie otworzone okno dialogowe (patrz rysunek 23.18).

Okno dialogowe przedstawione na rysunku 23.18 pozwala na ustawienie kilku różnych opcji.

- ♦ **Format:** rodzaj systemu plików, jaki chcemy założyć na wybranej partycji. Dostępne są następujące możliwości: Extended 2, Extended 3, JFS, Memory Swap, ReiserFS, Windows Virtual FAT oraz XFS.

Rysunek 23.18.

Formatowanie
wybranej partycji
za pomocą
narzędzia Dyski



- ♦ **Ścieżka:** punkt zamontowania nowo tworzonej partycji w systemie. Ta ścieżka jest zapamiętywana podczas formatowania partycji, lecz nie jest rzeczywiście wykorzystywana, dopóki nie zostanie wybrany przycisk *Włącz*, tak jak na rysunku 23.17.

W celu rozpoczęcia formatowania należy kliknąć przycisk *Formatuj* (patrz rysunek 23.18). Zostanie wyświetlone okno dialogowe z prośbą o potwierdzenie. Kliknięcie przycisku *Tak* spowoduje sformatowanie partycji, natomiast *Nie* — powrót do okna dialogowego, co przedstawiam na rysunku 23.17.

Gdy sprawdzanie lub formatowanie nośnika danych zostanie ukończony, należy kliknąć przycisk *OK* w oknie prezentowanym na rysunkach 23.16 oraz 23.17, co spowoduje zamknięcie programu *Dyski*.

Wykorzystanie narzędzi obsługiwanych z wiersza poleceń do identyfikacji napędów i partycji

Polecenie `df` pozwala w prosty sposób sprawdzić rozmiar partycji zamontowanych w systemie plików. Niestety, nie ujawnia ono informacji o nowych dyskach i partycjach (które często nie zawierają nawet systemu plików) ani o urządzeniach zawierających systemy plików, które nie są w danej chwili zamontowane. Na szczęście, istnieją dość proste sposoby sprawdzenia listy urządzeń dyskowych w systemie, niezależnie od tego, czy są sformatowane lub zamontowane. W tym przypadku jesteśmy skazani na magię wiersza poleceń, ale w końcu czytelnik już może nazwać się przynajmniej adeptem linuxowej sztuki magicznej, nie ma więc powodów do obaw. W końcu właśnie przed chwilą udało się zamontować w systemie nowy dysk twardy!

Najprostszy sposób identyfikacji plików urządzeń związanych z nowo dodanymi dyskami i ich partycjami polega na przeanalizowaniu komunikatów rozruchowych jądra w poszukiwaniu wzorców opisanych w punkcie „Rozwiązywanie problemów z automatycznym wykrywaniem urządzeń”. Sugerowałem tam, że w plikach dzienników systemowych należy poszukiwać komunikatów związanych z mechanizmem `hotplug`, dotyczących zdarzeń detekcji nowych urządzeń oraz ich odłączenia od systemu. Aby przejrzeć komunikaty jądra (które są zapisywane bezpośrednio w pamięci operacyjnej, ponieważ dotyczą etapu rozruchu systemu, gdy nie jest jeszcze zamontowany jakikolwiek system plików, na którym taki dziennik można by zapisać), wystarczy wywołać polecenie `dmesg`. Wiemy, że urządzenia IDE noszą nazwy rozpoczynające się przedrostkiem `hd`, po którym następuje jedna litera, natomiast dyski SATA mają nazwy rozpoczynające się przedrostkiem `sd`, po którym następuje jedna litera, zatem do wyszukania interesujących nas komunikatów możemy wykorzystać wzorec dopasowań `'[hs]d[a-z]: '`. Dla czytel-

ników, którym obca jest składnia wyrażeń regularnych: wzorzec '[hs]d[a-z]: ' oznacza „dopasuj ciąg znaków zawierający znak h lub s, po którym występuje znak d, po którym występuje litera z zakresu od a do z, po której występuje dwukropek i spacja”. Proste, nieprawdaż?

Wynik wyszukiwania tego ciągu znaków w komunikatach jądra jednego z moich systemów Ubuntu daje mniej więcej oczekiwane wyniki:

```
$ dmesg | grep '[hs]d[a-z]: '
[ 846.564436] SCSI device sda: 234441648 512-byte hdwr sectors (120034 MB)
[ 846.564453] SCSI device sda: drive cache: write back
[ 846.564514] SCSI device sda: 234441648 512-byte hdwr sectors (120034 MB)
[ 846.564526] SCSI device sda: drive cache: write back
[ 846.564531] sda: sda1 sda2
[ 846.575071] SCSI device sdb: 156299375 512-byte hdwr sectors (80025 MB)
[ 846.575086] SCSI device sdb: drive cache: write back
[ 846.578550] SCSI device sdb: 156299375 512-byte hdwr sectors (80025 MB)
[ 846.578826] SCSI device sdb: drive cache: write back
[ 846.578830] sdb: sdb1 sdb2 < sdb5 >
[ 848.272430] hdc: DVDWR 16X16X, ATAPI CD/DVD-ROM drive
[ 848.952828] hdc: ATAPI 48X DVD-ROM DVD-R CD-R/RW drive, 2048kB Cache
[ 852.169148] SCSI device sdc: 8910423 512-byte hdwr sectors (4562 MB)
[ 852.170635] SCSI device sdc: drive cache: write back
[ 852.171452] SCSI device sdc: 8910423 512-byte hdwr sectors (4562 MB)
[ 852.172941] SCSI device sdc: drive cache: write back
[ 852.172943] sdc: sdc1 sdc2
```

Ten komunikat informuje, że zostały wykryte trzy urządzenia SCSI (lub SATA, wykorzystujące nazewnictwo SCSI) oraz jedno urządzenie IDE ATA CD/DVD-ROM widoczne jako */dev/hdc*. Pierwsze urządzenie SCSI, */dev/sda*, nie zawiera partycji. Drugie urządzenie, */dev/sdb*, zawiera dwie partycje podstawowe, */dev/sdb1* i */dev/sdb2*, oraz partycję logiczną */dev/sdb5* (co prawdopodobnie oznacza, że */dev/sdb2* jest partycją rozszerzoną zawierającą partycję */dev/sdb5*, ale do tego tematu wrócę za chwilę). Trzecie urządzenie, */dev/sdc*, zawiera dwie partycje podstawowe: */dev/sdc1* i */dev/sdc2*.

Do uzyskania bardziej szczegółowych informacji na temat partycji dysku można wykonać polecenie *fdisk* z opcją *-l*. Oto przykład:

```
$ sudo fdisk -l /dev/sdb
Disk /dev/sdb: 80.0 GB, 80025280000 bytes
255 heads, 63 sectors/track, 9729 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks    Id System
/dev/sdb1    *           1         9566     76838863+   83  Linux
/dev/sdb2             9567         9729     1309297+    5  Extended
/dev/sdb5             9567         9729     1309266    82  Linux swap / Solaris
```

Ten wynik udowadnia, że zgodnie z oczekiwaniami */dev/sdb2* jest partycją rozszerzoną zawierającą partycję wirtualną */dev/sdb5*.

Wywołując to samo polecenie wyszukujące informacje o dyskach twardych w systemie zawierających tylko urządzenia IDE, uzyskamy:

```
$ dmesg | grep 'hd[a-z]: '
```

```
[17179574.600000] hda: HITACHI_DK23DA-30B, ATA DISK drive
[17179576.012000] hdc: UJDA720 DVD/CDRW, ATAPI CD/DVD-ROM drive
[17179576.696000] hda: max request size: 128KiB
[17179576.708000] hda: 58605120 sectors (30005 MB) w/2048KiB Cache...
[17179576.708000] hda: cache flushes supported
[17179576.708000] hda: hda1 hda2 < hda5 >
[17179576.768000] hdc: ATAPI 24X DVD-ROM CD-R/RW drive, 2048kB Cache, UDMA(33)
[17179944.340000] hde: LEXAR ATA FLASH, CFA DISK drive
[17179944.676000] hde: max request size: 128KiB
[17179944.676000] hde: 1001952 sectors (512 MB) w/1KiB Cache, CHS=994/16/63
[17179944.676000] hde: cache flushes not supported
[17179944.676000] hde: hde1
[17179944.680000] ide-cs: hde: Vcc = 3.3, Vpp = 0.0
```

Ten listing zawiera informacje o trzech urządzeniach IDE: `/dev/hda`, `/dev/hdc` i `/dev/hde`. Urządzenie `/dev/hda` jest dyskiem ATA zawierającym dwie partycje podstawowe, `/dev/hda1` i `/dev/hda2`, oraz partycję logiczną, `/dev/hda5`. Podobnie jak w poprzednim przykładzie, oznacza to prawdopodobnie, że `/dev/hda2` jest partycją rozszerzoną zawierającą partycję logiczną `/dev/hda5`. Urządzenie `/dev/hdc` to napęd ATAPI DVD-ROM CD-R/RW niezawierający — oczywiście — żadnych partycji. Trzecie urządzenie, `/dev/hde`, to karta Flash zawierająca jedną partycję, `/dev/hde1`. Podobnie jak napędy SATA, również urządzenia USB są widziane przez Linux jako urządzenia SCSI, natomiast urządzenia Flash są obsługiwane jak dyski IDE.

Tak spora dawka tajemniczych wywołań poleceń i komunikatów jądra systemu może przyprawić osoby niedoświadczone o zawrót głowy, ale w tym punkcie przekazałem mnóstwo ważnych informacji na temat systemu i urządzeń dyskowych. Jedno z urządzeń SATA napotkanych w przykładzie nie zawiera partycji, jest zatem dość bezużyteczne w systemie. Z tego powodu w następnym punkcie omówię sposoby formatowania i podziału na partycje nowo zamontowanego dysku twardego.

Wykorzystanie narzędzi trybu tekstowego do partycjonowania i formatowania dysków

W rozdziale 4. opisałem podstawowe zagadnienia dotyczące nośników danych systemu komputerowego, takich jak dyski twarde, partycje i systemy plików. W niniejszym rozdziale wspomniałem również o tym, że nośnika danych nie można używać, jeśli nie został prawidłowo podzielony na partycje, a te partycje nie zostały sformatowane do wykorzystania w formie systemu plików. W tym punkcie opiszę sposób wykorzystania narzędzi trybu tekstowego, takich jak *fdisk*, *mkfs* i (okazjonalnie) *mkswap*. Oto ich krótkie omówienie.

- ♦ *fdisk*: służy do tworzenia partycji na dysku twardym i zarządzania nimi. Dokładnie nie wiadomo, co oznacza litera *f* w nazwie, prawdopodobnie chodzi o *format*.
- ♦ *mkfs*: służy do tworzenia na partycji systemu plików określonego typu.
- ♦ *mkswap*: służy do formatowania określonej partycji do wykorzystania w charakterze linuksowej partycji wymiany, tzn. partycji wykorzystywanej przez Ubuntu w charakterze pamięci wirtualnej.

Po dodaniu dysku do systemu należy podzielić go na partycje. W poprzednim punkcie wykorzystałem program *fdisk* do wyświetlenia informacji o partycjach dysku, tym razem

posłużymy się tym programem do zdefiniowania partycji na dysku. W przykładach użyję dwóch partycji na dysku `/dev/sda` (w rzeczywistości układ partycji, jakie utworzymy w tym przykładzie, jest dokładnie taki sam jak w poprzednim przykładzie, przedstawionym na rysunku 23.17).

Aby podzielić dysk na partycje za pomocą programu *fdisk*, należy wykonać następujące kroki.

1. Trzeba uruchomić program *fdisk* z użyciem polecenia *sudo*, podając jako parametr nazwę pliku urządzenia dysku twardego, jak w poniższym przykładzie:

```
$ sudo fdisk /dev/sda
The number of cylinders for this disk is set to 14593.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
 1) software that runs at boot time (e.g., old versions of LILO)
 2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)
Command (m for help):
```

Ostatni wiersz przykładu to wiersz zachęty programu *fdisk*, sygnalizujący, że program oczekuje na interakcję z użytkownikiem (wprowadzenie polecenia).

2. Teraz należy wprowadzić polecenie *p* (*print*), co spowoduje wypisanie listy istniejących partycji. W końcu wolelibyśmy uniknąć przypadkowego usunięcia informacji zapisanych na dysku, jeśli np. omyłkowo wpisaliśmy nazwę urządzenia dysku zawierającego ważne dane. To polecenie (litera *p* i *Enter*) spowoduje wypisanie informacji o partycjach na dysku, np.:

```
Command (m for help): p
Disk /dev/sda: 120.0 GB, 120034123776 bytes
255 heads, 63 sectors/track, 14593 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
   Device Boot      Start         End      Blocks   Id  System
Command (m for help):
```

Jak widać, dysk nie zawiera obecnie żadnych partycji, co jest zgodne z oczekiwaniami.

3. Następnie wywołujemy polecenie *n* (*new*), aby utworzyć nową partycję:

```
Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
```

4. Po wywołaniu tego polecenia program poprosi o określenie typu partycji. Wpisujemy literę *e* (*extended*, rozszerzona) lub *p* (*primary*, podstawowa). Wprowadźmy literę *p*, aby utworzyć partycję podstawową:

```
p
Partition number (1-4):
```

5. Wybór partycji fizycznej spowoduje zadanie pytania o jej numer. Dyski w Linuksie mogą składać się z czterech partycji, z których jedna może być partycją rozszerzoną. Tworzymy partycję o numerze 1:

First cylinder (1-14593, default 1):

6. Wskazanie numeru partycji spowoduje wyświetlenie prośby o podanie fizycznego cylindra, od którego rozpoczyna się partycja. Program wypisze dostępny zakres cylindrów (w tym przypadku 1-14593). Aby zaakceptować domyślną wartość (w tym przypadku 1), wystarczy nacisnąć klawisz *Enter*.

Using default value 1

Last cylinder or +size or +sizeM or +sizeK (1-14593, default 14593):

7. Przy tworzeniu partycji można zdefiniować jej ostatni blok, można też określić jej rozmiar w blokach lub w klasycznych jednostkach pojemności. Przykładowo określenie +30G spowoduje utworzenie partycji o rozmiarze 30 gigabajtów.

Last cylinder or +size or +sizeM or +sizeK (1-14593, default 14593): +30G

Command (m for help):

8. To już wszystkie informacje niezbędne do utworzenia partycji, niestety, program nie wyświetla informacji o tym, czy operacja zakończyła się powodzeniem. Aby sprawdzić, czy partycja została utworzona, najlepiej spowodować wypisanie istniejących partycji, wywołując polecenie *p*:

Command (m for help): p

Disk /dev/sda: 120.0 GB, 120034123776 bytes

255 heads, 63 sectors/track, 14593 cylinders

Units = cylinders of 16065 * 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1		1	3648	29302528+	83	Linux

Oczywiście, utworzona partycja o rozmiarze 30 GB to */dev/sda1*, czyli pierwsza partycja na dysku.



W tym momencie na dyskach nie zostały jeszcze wprowadzone jakiekolwiek fizyczne modyfikacje. Wszelkie zmiany są dokonane jedynie w pamięci, zatem w przypadku rezygnacji ze zmian w podziale dysku na partycje wystarczy po prostu zamknąć program *fdisk* bez zapisu (polecenie *q*).

9. Powyższą sekwencję powtarzamy w celu utworzenia drugiej partycji, o mniejszym rozmiarze. Tym razem bez omawiania każdego kroku, kompletna sekwencja poleceń jest następująca:

Command (m for help): n

Command action

e extended

p primary partition (1-4)

p

Partition number (1-4): 2

First cylinder (3649-14593, default 3649):

Using default value 3649

Last cylinder or +size or +sizeM or +sizeK (3649-14593, default 14593): +1G

10. Tyle wystarczy, aby utworzyć drugą partycję, ponownie bez informacji o powodzeniu operacji. Sprawdźmy więc, czy partycja została utworzona (polecenie *p*):

Command (m for help): p

Disk /dev/sda: 120.0 GB, 120034123776 bytes


```
255 heads, 63 sectors/track, 14593 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1		1	3648	29302528+	83	Linux
/dev/sda2		3649	3771	987997+	83	Linux

Command (m for help):

Ponownie wszystko w porządku: mamy partycję 30 GB i 1 GB.

11. Zapisujemy zmiany na dysku i kończymy pracę z programem *fdisk*. W tym celu należy wywołać polecenie zapisu (*w*), co zakończy działanie programu i zapisze wprowadzone zmiany do tablicy partycji dysku.

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

Gratulacje, dysk jest podzielony na partycje. Z reguły nie warto pozostawiać niewykorzystanej przestrzeni na dysku, ale możemy sobie na to pozwolić w naszym przykładzie, ponadto z reguły uruchamiam program *fdisk* do utworzenia każdej partycji osobno.

W kolejnym etapie sformatujemy każdą z utworzonych partycji, pierwszą z zastosowaniem określonego systemu plików, drugą jako przestrzeń wymiany. W rzeczywistości rzadko konieczność jest tworzenie dodatkowych przestrzeni wymiany po zainstalowaniu systemu, ale warto nauczyć się tego, choćby na wszelki wypadek. W prawie 100% przypadków tworzy się partycje na potrzeby systemów plików, np. aby zamontować je w celu zwiększenia ilości dostępnego miejsca na dyskach.

Do utworzenia systemu plików stosuje się program *mkfs*, który jest właściwie jedynie interfejsem do innych poleceń tworzących właściwe systemy plików. W większości systemów linuksowych wykorzystuje się następujące systemy plików: ext2, ext3, JFS, Reiser4, ReiserFS, Windows Virtual FAT (vfat) oraz XFS. Wszystkie te systemy plików zostały omówione w rozdziale 4., w punkcie „Lokalne systemy plików: standardowy i z kro-niką”. Aby utworzyć jeden z tych systemów plików, wystarczy wskazać programowi *mkfs* odpowiedni typ systemu za pomocą opcji *-t*. W poniższym przykładzie utworzę system plików *ext3* na partycji */dev/sda1*:

```
$ sudo mkfs -t ext3 /dev/sda1
mke2fs 1.38 (30-Jun-2005)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
3662848 inodes, 7325632 blocks
366281 blocks (5.00%) reserved for the super user
First data block=0
224 block groups
32768 blocks per group, 32768 fragments per group
16352 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632,
    2654208, 4096000
```

```
Writing inode tables: 224/224
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

This filesystem will be automatically checked every 25 mounts or 180 days, whichever comes first. Use tune2fs -c or -i to override.

To wszystko. Partycję można zamontować tymczasowo, wykorzystując polecenie `mount`, można też dopisać definicję ją do pliku `/etc/fstab` dzięki czemu będzie ona automatycznie montowana po ponownym uruchomieniu komputera. Obie te opcje zostały opisane szczegółowo w rozdziale 4.

Do utworzenia na partycji przestrzeni wymiany służy polecenie `mkswap`. Formatuje ono partycję w sposób niezbędny do jej wykorzystania przez menedżer obsługi pamięci wirtualnej. Utwórzmy zatem przestrzeń wymiany na partycji `/dev/sda2`:

```
# mkswap /dev/sda2
Setting up swapspace version 1, size = 1011703 kB
no label, UUID=731c0f53-bf25-4f91-9ac5-73a2b5bef1eb
```

Utworzoną w ten sposób przestrzeń wymiany można wykorzystać natychmiast, wywołując polecenie `swapon /dev/sda2`. Aby na stałe skonfigurować tę partycję w charakterze przestrzeni wymiany, należy dopisać odpowiednią definicję w pliku `/etc/fstab`, co zostało opisane w rozdziale 4.

Wykorzystanie kart PCMCIA

Karty w standardzie PCMCIA (ang. *Personal Computer Memory Card International Association*) stanowią oryginalny mechanizm zaprojektowany do obsługi wymiennych urządzeń dla komputerów przenośnych. Karty PCMCIA często nazywa się kartami PC, ponieważ niewiele osób pamięta, co oznacza ten akronim (co sugeruje nieoficjalne rozwinięcie akronimu PCMCIA: „People Can’t Memorize Computer Industry Acronyms”, czyli „ludzie nie są w stanie zapamiętać akronimów przemysłu komputerowego”).

Istnieją karty PCMCIA implementujące interfejsy sieci przewodowych i bezprzewodowych, modemy, interfejsy SCSI, USB i FireWire, przydatne w systemach nieobsługujących tych standardów itp. Wraz z wersją 2.6 jądra Linuksa, wykorzystywaną w systemach Ubuntu, tradycyjny pakiet PCMCIA został przeniesiony do podsystemu `hotplug`, którego zadaniem jest automatyczne tworzenie odpowiednich plików urządzeń po włożeniu karty PCMCIA do komputera.

Do zarządzania kartami PCMCIA podłączonymi do systemu służy narzędzie `cardctl`. Aby sprawdzić listę kart podłączonych do systemu, należy wywołać polecenie `cardctl ident`. Spowoduje to wypisanie wyniku w następującej postaci:

```
$ cardctl ident
Socket 0:
  product info: "3Com", "Megahertz 589E", "TP/BNC LAN PC Card", "005"
  manfid: 0x0101, 0x0589
  function: 6 (network)
```

```
Socket 1:
product info: " LEXAR ATA FLASH CARD      ", " STORM ", "TTE01"
manfid: 0x4e01, 0x0200
function: 4 (fixed disk)
```

Z tego wynika, że w moim laptopie znajdują się dwie karty PCMCIA. W ocenie, czy te dwie karty są zainicjowane prawidłowo oraz jakie sterowniki zostały użyte do ich obsługi, pomocny jest wynik polecenia `dmesg`, np. w efekcie poszukiwania w wyniku polecenia `dmesg` ciągu znaków `3com` znalazłem następujący fragment:

```
[17179870.988000] PCMCIA: registering new device PCMCIA0.0
[17179871.248000] eth2: 3Com 3c589, bio 0x300, irq 5, hw_addr 00:00:86:58:D3:E1
[17179871.248000]      8K FIFO split 5:3 Rx:Tx, auto xcvr
[17179881.916000] eth2: no IPv6 routers present
[17179886.384000] eth2: flipped to 10baseT
```

Z tego wynika, że moja karta Ethernet PCMCIA jest przyłączona do interfejsu sieciowego `eth2`. Dzięki tej wiedzy mogę posłużyć się standardowym poleceniem `ifconfig`, aby zbadać stan tego interfejsu sieciowego:

```
$ ifconfig eth2
eth2      Link encap:Ethernet  HWaddr 00:00:86:58:D3:E1
          inet addr:192.168.6.186  Bcast:192.168.6.255  Mask:255.255.255.0
          inet6 addr: fe80::200:86ff:fe58:d3e1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:22 errors:0 dropped:0 overruns:0 frame:0
          TX packets:11 errors:0 dropped:0 overruns:0 carrier:7
          collisions:0 txqueuelen:1000
          RX bytes:3424 (3.3 KiB)  TX bytes:1878 (1.8 KiB)
          Interrupt:5 Base address:0x300
```

Druga karta w moim gnieździe PCMCIA to LEXAR ATA Flash. Po przeszukaniu wyniku polecenia `dmesg` znalazłem następujące informacje na jej temat:

```
[17179603.088000] hde: LEXAR ATA FLASH, CFA DISK drive
[17179603.424000] ide2 at 0x3100-0x3107,0x310e on irq 3
[17179603.424000] hde: max request size: 128KiB
[17179603.424000] hde: 1001952 sectors (512 MB) w/1KiB Cache, CHS=994/16/63
[17179603.424000] hde: cache flushes not supported
[17179603.424000] hde: hde1
[17179603.428000] ide-cs: hde: Vcc = 3.3, Vpp = 0.0
```

Jak widzimy, karta jest dostępna jako urządzenie IDE `/dev/hde` i zawiera jedną partycję, `/dev/hde1`, którą można zamontować poleceniem `mount`.

Więcej informacji na temat kart PCMCIA w systemach Linux można znaleźć pod adresem <http://pcmcia-cs.sourceforge.net/ftp/doc/PCMCIA-HOWTO.html>.

Przyłączanie kart PCI

Karty PCI to nieco inne urządzenia od pozostałych, opisanych w tym rozdziale, ponieważ z reguły umożliwiają wykorzystanie innych urządzeń zewnętrznych, takich jak głośniki (karty dźwiękowe PCI), zewnętrzne nośniki danych (karty USB, FireWire czy SCSI na interfejsie PCI), lub zawierają podstawowe urządzenia np. modemy, dodatkowe porty

szeregowe lub równoległe itp. Przed podłączeniem karty PCI należy wyłączyć komputer. Wykrycie przez BIOS karty PCI może umknąć uwadze użytkownika, ale odpowiednie informacje na temat, czy karta została prawidłowo wykryta i zainicjowana można znaleźć w dzienniku jądra systemu.

Warto zapoznać się z listą urządzeń PCI przyłączonych do systemu, aby zweryfikować, które z nich zostały prawidłowo zidentyfikowane przez system. Do tego celu służy polecenie `lspci`, oto przykład wyniku jego działania:

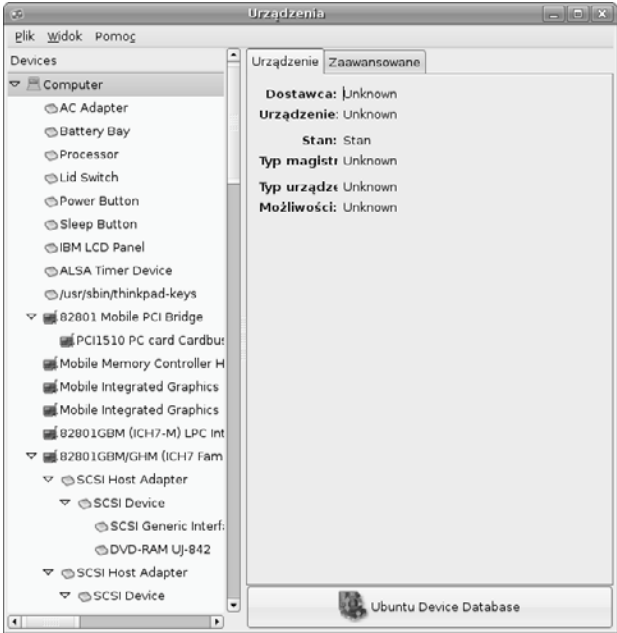
```
$ lspci
0000:00:00.0 RAM memory: nVidia Corporation C51 Host Bridge (rev a2)
0000:00:00.1 RAM memory: nVidia Corporation C51 Memory Controller 0 (rev a2)
0000:00:00.2 RAM memory: nVidia Corporation C51 Memory Controller 1 (rev a2)
0000:00:00.3 RAM memory: nVidia Corporation C51 Memory Controller 5 (rev a2)
0000:00:00.4 RAM memory: nVidia Corporation C51 Memory Controller 4 (rev a2)
0000:00:00.5 RAM memory: nVidia Corporation C51 Host Bridge (rev a2)
0000:00:00.6 RAM memory: nVidia Corporation C51 Memory Controller 3 (rev a2)
0000:00:00.7 RAM memory: nVidia Corporation C51 Memory Controller 2 (rev a2)
0000:00:02.0 PCI bridge: nVidia Corporation C51 PCI Express Bridge (rev a1)
0000:00:03.0 PCI bridge: nVidia Corporation C51 PCI Express Bridge (rev a1)
0000:00:04.0 PCI bridge: nVidia Corporation C51 PCI Express Bridge (rev a1)
0000:00:05.0 VGA controller: nVidia Corporation C51 PCI Express Bridge
0000:00:09.0 RAM memory: nVidia Corporation MCP51 Host Bridge (rev a2)
0000:00:0a.0 ISA bridge: nVidia Corporation MCP51 LPC Bridge (rev a2)
0000:00:0a.1 SMBus: nVidia Corporation MCP51 SMBus (rev a2)
0000:00:0b.0 USB Controller: nVidia Corporation MCP51 USB Controller (rev a2)
0000:00:0b.1 USB Controller: nVidia Corporation MCP51 USB Controller (rev a2)
0000:00:0d.0 IDE interface: nVidia Corporation MCP51 IDE (rev a1)
0000:00:0e.0 IDE interface: nVidia Corporation MCP51 Serial ATA Controller
0000:00:0f.0 IDE interface: nVidia Corporation MCP51 Serial ATA Controller
0000:00:10.0 PCI bridge: nVidia Corporation MCP51 PCI Bridge (rev a2)
0000:00:10.1 0403: nVidia Corporation MCP51 High Definition Audio (rev a2)
0000:00:14.0 Bridge: nVidia Corporation MCP51 Ethernet Controller (rev a1)
0000:00:18.0 Host bridge: HyperTransport Technology Configuration
0000:00:18.1 Host bridge: [Athlon64/Opteron] Address Map
0000:00:18.2 Host bridge: [Athlon64/Opteron] DRAM Controller
0000:00:18.3 Host bridge: K8 [Athlon64/Opteron] Miscellaneous Control
0000:04:05.0 FireWire (IEEE 1394): VIA, Inc. IEEE 1394 Host Controller 0000:04:08.0
SCSI storage controller: LSI Logic / Symbios Logic 53c895
0000:04:09.0 Ethernet controller: 3Com Corporation 3c595 100BaseTX [Vortex]
```

Powyższy listing może wydać się mało czytelny, ale łatwo go przejrzeć w poszukiwaniu interesującej nas karty. Jeśli została prawidłowo wykryta, należy zastosować odpowiednie narzędzie konfiguracyjne, właściwe dla typu urządzenia obsługiwanego przez tę kartę.

Rozwiązywanie problemów z urządzeniami za pomocą narzędzi graficznych

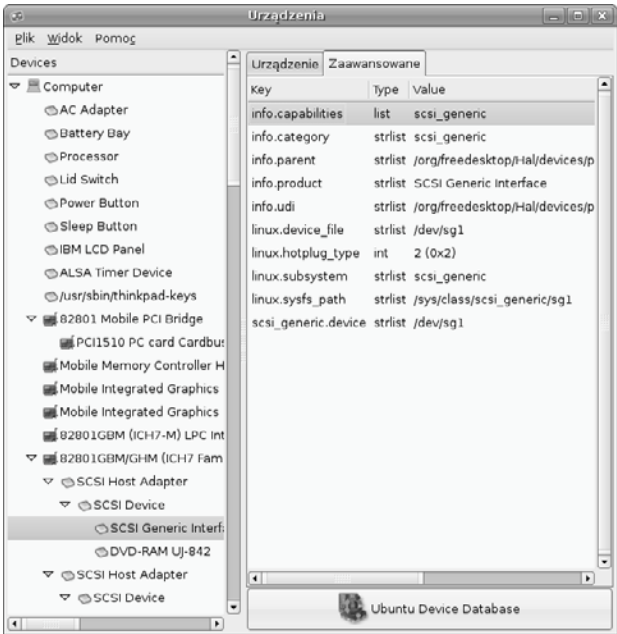
Ubuntu oferuje doskonałe narzędzie graficzne, menedżer urządzeń (pod nazwą Urządzenia), za którego pomocą można przeglądać sprzęt zainstalowany w systemie z punktu widzenia Ubuntu. Narzędzie to wywołuje się z menu systemowego *System/Administracja/Urządzenia*, główne okno programu zostało przedstawione na rysunku 23.19.

Rysunek 23.19.
*Menedżer urządzeń
w Ubuntu*



Po wyświetleniu okna Urządzenia w lewym panelu można przeglądać urządzenia przyłączone do systemu, zarówno na interfejsach wewnętrznych, jak i zewnętrznych. Po zaznaczeniu urządzenia pojawiają się w prawym panelu szczegóły dotyczące jego konfiguracji. W tym panelu znajdują się dwie karty. Pierwsza z nich, *Urządzenie*, zawiera ogólne informacje, co przedstawiam na rysunku 23.19. Druga, *Zaawansowane* (patrz rysunek 23.20), zawiera dodatkowe szczegóły na temat zaznaczonego urządzenia.

Rysunek 23.20.
*Zaawansowane
informacje
menedżera urządzeń*



Na przykładzie z rysunku 23.20 przedstawiam szczegóły połączenia SCSI ustanowionego dla interfejsu SATA do którego podłączony został dysk twardy Western Digital. Na tej karcie można znaleźć także informację o pliku urządzenia wykorzystywanym do jego obsługi, co może uprościć lokalizację i dostęp do nowo dołączanych urządzeń.

Menedżer urządzeń pozwala również dodawać do centralnej bazy urządzeń informacje na temat obsługiwanego (i nieobsługiwanego) sprzętu. W tym celu należy kliknąć przycisk *Ubuntu Device Database*, co spowoduje wyświetlenie okna przedstawionego na rysunku 23.21.

Rysunek 23.21.

Wysyłanie do bazy
danych informacji
o systemie



W tym oknie, obsługiwanym w formie kreatora, przechodzimy przez serię stron testujących poszczególne kwestie sprzętowe systemu, które pozwalają na wprowadzenie komentarzy na ich temat. Wysyłanie informacji na temat sprzętu za pomocą tego narzędzia pomoże nie tylko projektowi Ubuntu i jego użytkownikom, lecz również może stanowić pomoc dla użytkownika, kiedy ma problem z konfiguracją swojego sprzętu. Wysyłanie do bazy informacji o posiadanym sprzęcie to drobny wkład, na jaki może sobie pozwolić każdy użytkownik społeczności Ubuntu.

Podsumowanie

W tym rozdziale wyjaśniłem, w jaki sposób można do systemu Ubuntu dodawać różne typy sprzętu, począwszy od zewnętrznych nośników danych wykorzystywanych tymczasowo lub do przenoszenia danych, po bardziej trwale modyfikacje konfiguracji, takie jak podłączenie drukarki, nowego dysku twardego czy skanera.

W rozdziale 24. wyjaśnię tajniki zabezpieczania systemu Ubuntu. Choć Linux jest zasadniczo odporny na większość typów wirusów, koni trojańskich i oprogramowania szpiegującego (*spyware*) powszechnych we współczesnym internecie (i większości systemów Windows), bezpieczeństwo sieciowe jest w systemie Linux tak samo ważnym zagadnieniem jak w jakimkolwiek innym systemie informatycznym przyłączonym do internetu. Istnieje ogromna liczba książek poświęconych bezpieczeństwu sieciowemu i informatycznemu, lecz w rozdziale 24. umieszczę podstawowe objaśnienie kluczowych koncepcji i problemów, a także wyjaśnię podstawowe sposoby zabezpieczenia systemu i wskażę dalszą drogę dla czytelników pragnących pogłębić swoją wiedzę na ten ważny temat.

Rozdział 24.

Konfiguracja sieci i bezpieczeństwo

W tym rozdziale:

- ◆ Konfiguracja interfejsów sieciowych
- ◆ Konfiguracja połączeń PPP
- ◆ Testowanie interfejsów sieciowych
- ◆ Wskazówki dotyczące zabezpieczenia sieci

Komputery osobiste próbowały wzajemnie się komunikować prawie od początku historii internetu, w latach 70.. Na długo przed powstaniem pojęcia dostawcy internetowego, przed powstaniem internetu fani komputerów domowych używali modemów do łączenia się z biuletynami, zdalnymi komputerami klasy mainframe lub minikomputerami, a starożytni dostawcy usług, tacy jak CompuServe i AOL, wykorzystywali programy emulatorów terminala do komunikacji, wymiany plików itp. Wczesne mechanizmy przesyłania danych, np. UUCP (ang. *Unix-to-Unix Calling Program*) i fidonet, stanowiły doskonałe sposoby przesyłania plików i innych informacji za pośrednictwem powolnych sieci komputerowych, które sieciami były dlatego, że jeden system znał numer telefoniczny drugiego.

Przekształcenie ARPANET-u w internet oraz komercjalizacja tego ostatniego spowodowały narodzenie się instytucji dostawcy internetu, tzn. firmy oferującej klientom mechanizmy przyłączania się do sieci, na razie z użyciem niezbyt wydajnych łączy typu punkt-punkt (*point-to-point*), nadal zależnych od modemów, a przez to umożliwiających dostęp do internetu z prędkościami transferu o wielkościach niewiele większych od zera. Mimo to pojawienie się dostawców internetu spowodowało, że komputer PC przestał być samotną wyspą, a stał się równorzędnym uczestnikiem internetu, na razie pracującym z dość niewielką prędkością.

Gdy dostawcy internetu zaczęli funkcjonować jako najważniejsi usługodawcy dla wielu użytkowników komputerów domowych, koszty sprzętu sieciowego zaczęły spadać, zbliżając się do cen powszechnego sprzętu codziennego użytku, takich jak tostery czy lodówki. W wielu domach pojawiła się większa liczba komputerów, co z kolei doprowadziło do powstania zjawiska lokalnych sieci domowych, często niezależnych lub podłączonych do internetu za pośrednictwem modemu, oferującego bramę do sieci o prędkości przesyłu

od 9600 bitów do 56 kilobitów na sekundę. Wiele zawdzięczamy tym pionierom komputerów domowych, którzy z poświęceniem przecierali szlaki, ściągając pornosy przez te niewiarygodnie powolne łącza.

Szerokopasmowy Ethernet, jeszcze tańszy przewodowy sprzęt sieciowy oraz eksplozja sieci bezprzewodowych spowodowały, że wykorzystanie sieci stało się codziennością dla wielu użytkowników komputerów. Systemy domowe uzyskały prawdziwe adresy IP i połączenia z internetem o prędkościach pozwalających na realizowanie rzeczywistych zadań, a często są też elementami sieci domowych wykorzystujących te łącza za pośrednictwem mechanizmu NAT (ang. *Network Address Translation*). Czytelnicy, dla których sieci to nowość, na początku tego rozdziału znajdą podstawowy przegląd zagadnień i terminologii sieciowych.

Lepsze sieci i szeroki dostęp do internetu mają jednak swoją cenę. Wszechobecne sieci dają tysiącom osób swobodny dostęp do komputera użytkownika za pośrednictwem adresu IP albo serwera WWW lub innych procesów sieciowych. Większość osób niewiele to obchodzi, ale zdarzają się osobniki ciekawe, bywają też osoby o po prostu złośliwych intencjach. Ta ostatnia grupa psuje opinię wszystkim użytkownikom internetu, starając się włamać do innych systemów i wykorzystać je na niejeden z niecnych sposobów. Nie mam nic przeciwko hakerom, którzy są ciekawi — eksploracja zawsze była fundamentalną cechą ludzkości. Prawdziwi **hakerzy** to elektroniczni eksploratorzy — na wzór członków *National Geographic Society* lub załogi z filmu *Star Trek* — śmiało stawiający kroki tam, gdzie nikt wcześniej się nie odważył. Natomiast osoby włamujące się do systemów tylko po to, by coś popsuć lub wykorzystać do niecnych celów, to **krakerzy** (ang. *cracker*), którzy pracują na złe imię tych pierwszych.

Niestety, istnieje mnóstwo krakerów o złych zamiarach, którzy z rozkoszą włamią się do każdego komputera pozbawionego ochrony, aby przekształcić go w bezwolny system zombie, lub po to, by zademonstrować swoją „wyższość”, a często po prostu w celach zarobkowych. Niestety, wszechobecność sieci spowodowała również wielką dostępność narzędzi ułatwiających zadanie wszelkiej maści złoczyńcom sieciowym. Osoby, wykorzystujące takie narzędzia, określa się terminem „dzieciaków ze skryptami” (ang. *script kiddies*): to oni demonstrują swój kunszt sieciowy, w podobny sposób jak dziecko wyposażone w pistolet maszynowy UZI demonstruje swoje umiejętności bojowe.

Zmierzam do tego, że wszechobecność sieci powoduje, iż bezpieczeństwo sieciowe staje się obowiązkiem wszystkich użytkowników. Jeśli ktoś mieszka w niewielkim miasteczku w USA, w którym kradzież gazety w kiosku za rogiem uznaje się za wielkie przestępstwo, zamykanie drzwi domu na noc można uznać za przejaw przewrażliwienia. Jeśli jednak mieszkaniec tego cichego i spokojnego miasteczka jest przyłączony do internetu, musi uznać, że jest mieszkańcem wielkiego miasta. Administratorzy sieci korporacyjnych i akademickich wiedzą o tym aż za dobrze. Co najsmutniejsze, dziś musi o to dbać babcia i wnuczek, i jego rodzice też. Bezpieczeństwo jest dziś zagadnieniem ważniejszym niż kiedykolwiek wcześniej, a jutro będzie jeszcze gorzej.

W tym rozdziale podaję podstawy będące wprowadzeniem do sieci, omawiam narzędzia oferowane przez Ubuntu Linux do graficznej konfiguracji i testowania sieci, jednocześnie (co najważniejsze) przekazuję podstawowe zasady najskuteczniejszego zabezpieczania systemu przed atakami. Istnieje stare jak informatyka powiedzenie, że naprawdę

bezpieczny jest tylko ten system, który nie jest przyłączony do czegośkolwiek. Choć to najświętsza prawda, to takie podejście jest — oczywiście — zupełnie niepraktyczne. Istnieją jednak proste reguły, których przestrzeganie pozwala zminimalizować szanse włamania do naszego systemu. Jeśli już ktoś używa systemu Ubuntu Linux, to przez sam ten fakt jest znacznie mniej podatny na ataki niż użytkownik Windows 98, ME i 2000.

Podstawowe zagadnienia dotyczące sieci

Większość nowoczesnych systemów komputerowych potrafi komunikować się z innymi systemami i urządzeniami, wykorzystując do tego celu sieć Ethernet, z zastosowaniem protokołów TCP/IP (ang. *Transmission Control Protocol/Internet Protocol*) i UDP (ang. *Universal Data Packet*). Ethernet został wynaleziony przez Xerox Corporation w instytucie Xerox PARC (ang. *Palo Alto Research Center*) we wczesnych latach 70. Jak większość wynalazków tej firmy (chlubny wyjątek stanowi kserokopiarka), Ethernet nie przyniósł pieniędzy Xeroxowi, mimo że stał się standardem przemysłowym i został skomercjalizowany przez wiele firm, np. 3COM, która została założona przez jednego z wynalazców Ethernetu, Boba Metcalfa. On, w przeciwieństwie do swoich pracodawców, potrafił docenić wagę swojego wynalazku.

Jeszcze dwadzieścia lat temu „internet” był pojęciem mocno technicznym, wykorzystywanym przez osoby, którym pracodawcy (często uczelnie) umożliwili dostęp do internetu lub jego poprzednika, ARPANET-u. Powstanie i popularyzacja WWW (ang. *World Wide Web*) oraz poczty elektronicznej zastępującej w wielu obszarach kontaktów rozmowy telefoniczne zmieniły tę sytuację diametralnie. Nagle pojawił się powód, aby ludzie chcieli mieć dostęp do internetu.

Wczesne mechanizmy dostępu do internetu opierały się na połączeniach wdzwanianych emulujących połączenia TCP/IP przy pomocy protokołów, takich jak SLIP (ang. *Serial Line Internet Protocol*), CSLIP (ang. *Compressed SLIP*) czy PPP (ang. *Point-To-Point Protocol*). Jeśli ktoś nie był maniakiem komputerowym, deweloperem lub naukowcem, rzadko miał dostęp do sieci domowych, ale to zmieniło się wraz z pojawieniem się szerokopasmowego dostępu do internetu za pośrednictwem kabla lub dostawców telefonii. Jak wspomniałem we wprowadzeniu do tego rozdziału, sieci domowe stają się coraz bardziej powszechne, ale większość użytkowników nie musi samodzielnie ich konfigurować. Jeśli ktoś ma pojedynczy komputer PC, Mac lub stację roboczą, wystarczy bezpośrednie połączenie z modemem kablowym lub DSL. Jednak posiadanie większej liczby komputerów, które chcemy połączyć w sieci lokalnej, powoduje, że musimy poznać niezrozumiałe terminy, takie jak koncentratory, przełączniki, 10-BaseT, RJ45, kable skrzyżowane, pakiety, bramy, routery, Cat5 i wiele innych, ulubionych określeń ze słownika maniaków sieciowych. W tym podrozdziale umieściłem przegląd tychże terminów. Informuję o tym, jak skonfigurować prostą sieć domową; mam nadzieję, że również pomogę czytelnikom poczuć się nieco pewniej w sieciowych zagadnieniach przewijających się w różnych miejscach tej książki. Więcej szczegółów należy jednak szukać w innych książkach omawiających zagadnienia sieci.

Podstawowym elementem nowoczesnej sieci jest standardowy przewód ethernetowy, zbudowany z kilku przewodów skręconych wzajemnie, stąd jego popularna nazwa — skrętka. Na końcach skrętki znajdują się wtyczki odpowiedniego typu. Najpopularniejsze są wtyczki

typu RJ-45, zbudowane najczęściej z przezroczystego plastiku, a podobne do standardowej wtyczki telefonicznej. Tego typu przewody ethernetowe czasem są nazywane 10-BaseT, 100-BaseT lub 1000-BaseT, gdzie część liczbową nazwy w rzeczywistości określa prędkość sieci, ponieważ przewody w sieciach tego typu mają takie same parametry. 1000-BaseT jest również określany jako gigabit Ethernet i stanowi nadchodzący standard wydajności. Ethernet 10 i 100 (10 megabitów i 100 megabitów) to odpowiednio standard odchodzący w przeszłość oraz standard obecny.



Czasem wśród starszego sprzętu można natrafić na określenie 10-Base2. To starszy typ Ethernetu 10 megabitowego, wykorzystującego inny przewód (koncentryczny) i wtyk typu BNC (ang. *Bayonet Neill-Concelman* lub *Baby N Connector*), który nie jest obsługiwany przez większość współczesnego sprzętu sieciowego.

Najlepszy sposób wizualizacji współczesnego Ethernetu to niezwykle długi przewód, do którego przyłączone są komputery i inne urządzenia sieciowe. W najprostszym przypadku dołączenia urządzeń w sieć wykorzystuje się urządzenia centralne, takie jak koncentrator (ang. *hub*), przełącznik (ang. *switch*) lub router. Koncentrator to urządzenie pozwalające podłączyć większą liczbę urządzeń (gniazda wejściowe) do innego urządzenia (gniazdo wyjściowe), np. modemu kablowego, innego koncentratora, przełącznika, routera lub bramy. Komunikacja na gnieździe wejściowym koncentratora jest przekazywana do wszystkich innych gniazd wejściowych oraz do gniazda wyjściowego. Przełączniki działają na zasadzie bardzo zbliżonej do zasady działania koncentratorów, z tą różnicą, że potrafią śledzić adresy sprzętowe urządzeń przyłączonych do poszczególnych portów i ruch sieciowy kierują od razu do odpowiedniego portu, bez konieczności przesyłania go do wszystkich. Przełączniki, ponieważ mają więcej funkcji, są szybsze i bardziej kosztowne od koncentratorów.

Bramy i routery działają podobnie do koncentratorów i przełączników, ale są zaprojektowane do przekazywania ruchu między dwoma różnymi sieciami. Jeśli komputer, do którego jest kierowany ruch, znajduje się w innej sieci, żądanie jest przekazywane przez bramę (ang. *gateway*), która przesyła je dalej. Komunikacja sieciowa jest realizowana z wykorzystaniem dyskretnych jednostek informacji, noszących nazwę **pakietów**. Pakiety zawierają m.in. adres IP hosta docelowego. Adres IP ma postać *NNN.NNN.NNN.NNN* i jest odpowiednikiem skrzynki pocztowej, w sposób unikalny identyfikując określony komputer. Pakiety wysyłane do komputera o adresie spoza sieci lokalnej są przekazywane za pośrednictwem bramy. Routery są kosztownymi, zaawansowanymi urządzeniami kierującymi komunikacją między sieciami, tłumaczącymi pakiety między protokołami sieciowymi i ograniczającymi ruch sieciowy do wybranych sieci, dzięki czemu żądanie pobrania pliku z komputera w pokoju syna nie musi być wysyłane do całego internetu.

W sieciach domowych komputery najczęściej łączy się za pomocą koncentratorów lub routerów podłączonych do modemu kablowego lub DSL-owego. Różnica między tymi urządzeniami polega na tym, że koncentrator po prostu przekazuje pakiety na *port wychodzący*, który służy jedynie jako gniazdo łączące port jednego urządzenia z portem drugiego. Pakiety są po prostu przekazywane w niezmienionej formie. Brama natomiast może w pakietach adresowanych do internetu tłumaczyć adresy sieci lokalnej na adresy rozpoznawane w internecie, po czym dopiero przesyła je do modemu DSL. Podczas stosowania koncentratora komputery w sieci lokalnej muszą mieć adresy rozpoznawane w internecie, natomiast bramy pozwalają w sieci lokalnej zastosować specjalną pulę adresów, nazywanych nieroutowalnymi adresami IP. Brama przetłumaczy te wewnętrzne adresy na adresy

publiczny bramy we wszystkich pakietach wysyłanych do stacji w internecie. W takich zastosowaniach najczęściej używane są adresy z nieroutowalnej puli *192.168.X.Y*, gdzie *X* i *Y* są specyficzne dla konfigurowanej sieci domowej.



Zainteresowanych zagadnieniem nieroutowalnych adresów IP odsyłam do dokumentów RFC (ang. *Request for Comment*), w których zostały zdefiniowane, czyli 1597 i 1918. Łatwo je znaleźć za pomocą wyszukiwarki internetowej, a na skróty można tam trafić za pośrednictwem następujących adresów: <http://www.safety.net/sum1597.html> i <http://www.howstuffworks.com/nat2.htm>.

Adresy IP są przydzielane komputerom na jeden z dwóch sposobów: statycznie lub dynamicznie. Statyczne adresy muszą być unikalne w sieci, a charakteryzują się tym, że adres jest cechą szczególną konkretnej maszyny. Dynamiczne adresy są przydzielane automatycznie komputerom, które o to poproszą. Większość dostawców internetu wykorzystuje adresy dynamiczne, ponieważ mają do przydzielenia skończoną ich pulę. Wykorzystanie dynamicznych adresów pozwala dostawcy internetu przydzielać te adresy, które nie są w danej chwili używane (bo stosujące je komputery są w danym momencie odłączone od internetu). Większość dynamicznych adresów IP jest przydzielana za pomocą protokołu **DHCP** (ang. *Dynamic Host Configuration Protocol*), który oprócz adresu IP przekazuje systemowi informacje o konfiguracji sieci, takie jak adres IP bramy i adresy IP serwerów DNS, tłumaczących nazwy hostów na ich właściwe adresy IP.

Aby użyć statycznego adresu IP w swojej sieci, wystarczy przydzielić maszynom unikalne, nieroutowalne adresy IP z odpowiedniej puli. Przykładowo większość moich maszyn domowych wykorzystuje statyczne adresy z puli *192.168.6.Y*. Korzystałem z bramy, więc skonfigurowałem ją w taki sposób, aby dokonywała translacji adresów (znanej jako mechanizm **NAT**, ang. *Network Address Translation*), dzięki czemu pakiety wysyłane do stacji w internecie mają odpowiedni publiczny adres nadawcy należący właśnie do bramy.

Jeśli ktoś chce wykorzystać dynamiczne adresy IP w sieci domowej, jedna z maszyn w tej sieci musi działać jako serwer DHCP. Większość bram (m.in. popularne urządzenia marki DLink lub Linksys) posiada wbudowane serwery DHCP, pozwalające skonfigurować zakres przydzielanych adresów (w moim przypadku są to adresy od *192.168.6.240* do *192.168.6.250*). Po włączeniu funkcji translacji adresów brama będzie odpowiednio routowała pakiety. Należy pamiętać, że często brama zna swój publiczny adres IP dzięki połączeniu z serwerem DHCP dostawcy internetu, natomiast hosty w sieci lokalnej biorą swoje adresy IP z serwera DHCP. Nie należy konfigurować hostów w sieci wewnętrznej w taki sposób, aby pobierały adresy z serwera DHCP dostawcy internetu, chyba że w sieci lokalnej znajduje się jeden komputer lub zależy nam, aby maszyny te były publicznie dostępne. Jeśli nasza brama nie posiada serwera DHCP, czy też chcemy lepiej kontrolować działanie serwera DHCP lub mamy serwer Ubuntu Linux w konfiguracji komercyjnej, warto rozważyć wykorzystanie samodzielnie skonfigurowanego serwera DHCP w systemie Ubuntu. Zagadnienie to zostało omówione w rozdziale 28., „Konfiguracja serwera DHCP”.

Kolejne zagadnienie sieciowe to sposób identyfikacji i lokalizacji komputerów w internecie. Najczęściej dokonuje się tego za pomocą mechanizmu DNS (ang. *Domain Name Service*), który omówiłem w rozdziale 29., w podrozdziale „Podstawowe informacje o DNS i serwerze BIND”.

Podałem dość informacji, aby dać czytelnikowi podstawową orientację w zagadnieniach sieci domowych. Jak można się spodziewać, w internecie łatwo znaleźć mnóstwo stron WWW zawierających szczegółowe poradniki konfiguracji sieci domowych. Zainteresowani konkretnymi informacjami o sieciach domowych z użyciem określonego sprzętu i systemów operacyjnych mogą również poszukać odpowiedniej literatury w księgarniach.

Ręczna konfiguracja sprzętu sieciowego

Konfiguracja sprzętu sieciowego w komputerze należy do procesu instalacji systemu, ponieważ do instalacji systemu Ubuntu Linux z reguły potrzebne jest działające połączenie z internetem. Jednak wiadomo, że wszystko ulega zmianom. Może się zdarzyć, że musimy zainstalować dodatkowy sprzęt, zmienić istniejącą konfigurację z wykorzystującej DHCP na statyczne adresowanie, ustawić priorytety urządzeń w przypadku posiadania kilku interfejsów sieciowych (np. interfejsu przewodowego i bezprzewodowego w laptopie). Warto też po prostu lepiej zrozumieć sposób działania sieci oraz jej konfigurację w posiadanych systemach.

Ubuntu oferuje wygodne narzędzie do konfiguracji interfejsów sieciowych. Z menu systemowego należy wybrać funkcję *System/Administracja/Sieć* (z wiersza poleceń narzędzie to wywołuje się poleceniem `network-admin`). Po wpisaniu hasła pojawi się okno, takie jak na rysunku 24.1.

Rysunek 24.1.

Narzędzie
do konfiguracji sieci
w Ubuntu



Zawartość tego okna jest uzależniona od liczby interfejsów Ethernet dostępnych w systemie. Przykładowo na rysunku 24.1 przedstawiam okno w systemie zawierającym jeden interfejs Ethernet. Domyślnie w oknie konfiguracji sieci znajduje się dodatkowo jeden element reprezentujący połączenie *Point-to-Point Protocol* (PPP), niezależnie od tego, czy w systemie występuje modem, ponieważ połączenia PPP można nawiązywać również m.in. za pośrednictwem portów szeregowych. Na rysunku 24.2 przedstawiam okno *Sieć* w systemie zawierającym interfejsy Ethernet zarówno dla sieci bezprzewodowej, jak i przewodowej.

W dalszej części tego podrozdziału wykorzystam przykładowy system posiadający interfejs przewodowy i bezprzewodowy (patrz rysunek 24.2), ponieważ taka konfiguracja jest bardzo typowa dla współczesnych laptopów. Komputer klasy biurowej z reguły posiada

Rysunek 24.2.

Konfiguracja sieci
z dostępnym
interfejsem sieci
bezprzewodowej



pojedynczy przewodowy interfejs Ethernet, wykorzystanie większej liczby przewodowych interfejsów jest dość rzadko spotykane i z reguły ma praktyczny sens wyłącznie w systemach kierujących ruchem na styku dwóch sieci lub wymagających osobnej sieci na potrzeby programowania i testowania systemów lub aplikacji.

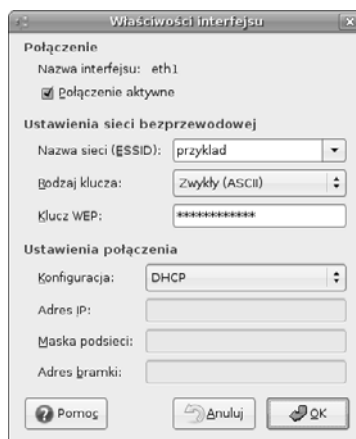


Systemy, w których występuje większa liczba interfejsów Ethernet, są obecnie bardzo rozpowszechnione. Jeśli wykorzystujemy jednocześnie kilka interfejsów Ethernet, z reguły sensowne jest przypięcie ich do kilku sieci, ponieważ inaczej routing pakietów może być nieco utrudniony. Systemy wyposażone w kilka interfejsów Ethernet, z których każdy jest przyłączony do innej sieci, nazywamy systemami wielosieciowymi (ang. *multi-homed systems*).

W oknie konfiguracji sieci można zaznaczyć dowolny interfejs, po czym kliknąć przycisk *Właściwości* w celu zapoznania się z konfiguracją lub dokonania jej modyfikacji. Na rysunku 24.3 przedstawiam okno *Właściwości* dla bezprzewodowego interfejsu sieci Ethernet z rysunku 24.2.

Rysunek 24.3.

Konfiguracja
właściwości interfejsu
sieci bezprzewodowej

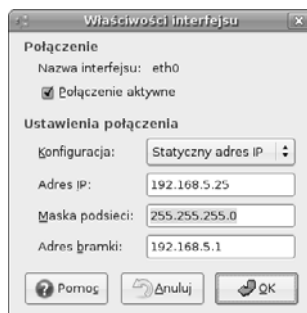


Jak widać, okno to pozwala skonfigurować podstawowe parametry interfejsu sieciowego. Szczegółowe objaśnienia tego okna dialogowego można znaleźć w podrozdziale rozdziału 25., zatytułowanym „Konfiguracja interfejsów sieci bezprzewodowej”.

Jak można zauważyć, ten interfejs jest skonfigurowany do wykorzystania mechanizmu DHCP, który dynamicznie przydziela adresy sieciowe interfejsom Ethernet przyłączającym się do sieci. Z tego właśnie powodu nieaktywne są pola służące do określania adresu IP i związanych z nim parametrów. Na rysunku 24.4 przedstawiam konfigurację tego samego interfejsu Ethernet, ale tym razem ze statyczną definicją adresowania IP.

Rysunek 24.4.

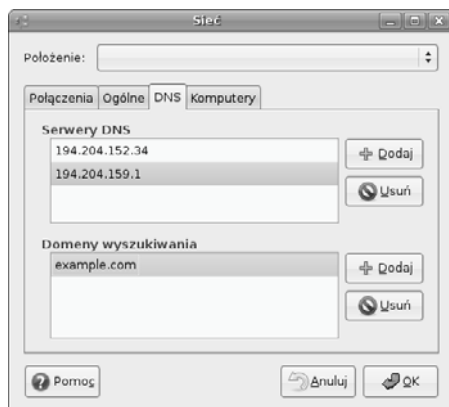
*Konfiguracja
statycznego adresu
interfejsu sieciowego*



Po zdefiniowaniu właściwości interfejsu sieciowego należy kliknąć przycisk *OK*, co spowoduje zamknięcie okna *Właściwości*. Jeśli nie jest wykorzystany mechanizm DHCP, trzeba przejść na kartę *DNS* w oknie *Sieć*, której zawartość przedstawiam na rysunku 24.5.

Rysunek 24.5.

*Określanie
serwerów DNS*



Jak wspomniałem w podrozdziale „Podstawowe zagadnienia dotyczące sieci” na początku tego rozdziału, tematy związane z mechanizmem DNS szczegółowo omówię w rozdziale 29., „Konfiguracja serwera DNS”. Dość stwierdzić, że większość współczesnych systemów wykorzystuje mechanizm DNS do wyszukiwania adresów IP systemów o znanych nazwach sieciowych. Wielu z nas w wyszukiwarce wpisuje po prostu `www.google.com`, ale do połączenia komputer potrzebuje liczbowego adresu sieciowego tego systemu. Okno przedstawione na rysunku 24.5 pozwala wprowadzić adresy serwerów DNS w sieci korporacyjnej lub serwerów DNS dostawcy internetu. To okno jest wspólne dla wszystkich interfejsów Ethernet zainstalowanych w systemie, zatem podczas konfigurowania drugiego interfejsu sieciowego nie trzeba ich już podawać. Ponadto, jak już wspominałem, wprowadzanie adresu serwera DNS jest niezbędne przede wszystkim, kiedy używamy statycznych adresów IP, ponieważ w przypadku stosowania mechanizmu DHCP w większości konfiguracji adresy serwerów DNS są przysyłane do stacji roboczej razem z innymi danymi w ramach usługi DHCP.



Ponieważ serwery DNS stanowią źródło informacji o adresach IP odpowiadających znanym nazwom hostów, w tym oknie należy wpisywać tylko adresy IP. Jeśli zostałaby wprowadzona nazwa hosta, system musiałby wykorzystać inny serwer DNS do uzyskania adresu IP serwera DNS określonego jego nazwą sieciową, co spowodowałoby zapętlenie logiczne w stylu: „Co było pierwsze, kura czy jajko?”.

Po wprowadzeniu lub zweryfikowaniu konfiguracji DNS interfejsu sieciowego należy przejść na kartę *Połączenia*, co spowoduje otwarcie początkowej karty okna *Sieć*. Jak można zauważyć, skonfigurowany przed chwilą interfejs jest oznaczony jako nieaktywny. Aby aktywować interfejs i umożliwić jego wykorzystanie, należy kliknąć przycisk *Aktywuj*. Po kilku sekundach okno *Sieć* zmieni swoją zawartość na przedstawioną na rysunku 24.6; będzie to informacja, że nasz interfejs sieciowy jest skonfigurowany i aktywny.

Rysunek 24.6.

Aktywacja nowo skonfigurowanego interfejsu sieciowego



W większości systemów biurkowych okno dialogowe dotyczące ustawień sieci będzie zawierało tylko jeden interfejs sieciowy. W tym momencie z pewnością warto przetestować nowo skonfigurowany interfejs sieciowy, aby upewnić się, czy wszystko działa prawidłowo. Ubuntu oferuje ciekawe narzędzie z interfejsem graficznym służące do testowania możliwości sieciowych systemu. Szczegółowe informacje na jego temat można znaleźć w dalszej części rozdziału, w podrozdziale „Testowanie sieci z użyciem narzędzi sieciowych dla środowiska GNOME”.

Jeśli wykorzystywany jest system wyposażony w większą liczbę interfejsów sieciowych, warto zapoznać się z następnym podrozdziałem, w którym omawiam sposoby wykorzystania odpowiednich interfejsów sieciowych w przypadku zmiany lokalizacji.

Ręczna konfiguracja połączeń modemowych

Jak wspominałem wcześniej, wszystkie systemy Ubuntu Linux dają możliwość skonfigurowania połączeń sieciowych za pośrednictwem protokołu PPP, co stanowi nowoczesny sposób realizacji połączeń na łączu szeregowym lub za pośrednictwem modemu. Mimo że szerokopasmowy internet staje się coraz bardziej powszechnym rozwiązaniem, modemy wykorzystujące protokół PPP są nadal dość popularnym sposobem nawiązywania połączeń z internetem. Spodziewam się, że wkrótce ten stan ulegnie zmianie przede wszystkim dlatego, że wielu użytkowników ma dość niekończącego się oczekiwania na

załadowanie nieco bardziej skomplikowanych stron WWW, jak również dlatego, że dostawcy internetu mogą zarobić znacznie więcej pieniędzy, jeśli przyzwyczają użytkownika do stałego korzystania z internetu i pozwolą mu poznać jego pełne możliwości dostępne dzięki szerokopasmowym łączom stałym. Wiele osób (również i ja) posiada dwa łącza. Sam wykorzystuję łącze modemowe głównie awaryjnie, gdy nie jest dostępne łącze stałe, np. podczas awarii punktu zbiorczego. Łącze modemowe bywa przydatne również do celów testowych. Oczywiście konta PPP są bardzo użyteczne z punktu widzenia przenośności. Często korzystam z tej możliwości np. na wakacjach, gdy muszę być w kontakcie (pocztą czy korekta rozdziałów pisanych przeze mnie książek), a moja żona na wypoczynek wybrała przeuroczy, ale mocno archaiczny zakątek świata.



Moje osobiste podziękowania dla Jerry'ego i Nancy Tibbettów, właścicieli Aurora Inn w Bar Harbor, stan Maine, za zainstalowanie szerokopasmowego łącza, dzięki czemu nie musiałem wozić kilkunastometrowego przewodu telefonicznego, jak to było w przypadku innych miejsc, w których spędzałem wakacje. Zacytuje Arnolda: „Jeszcze tu wróć”!

Połączenia PPP z internetem przy użyciu modemu są w wielu przypadkach nadal bardzo użyteczną formą komunikacji. Mój pierwszy system Linux zmusił mnie do napisania niewielkiego skryptu, połączenia z dostawcą internetowym, złożenia ofiary z kurczaka i żarliwej modlitwy o odrobinę szczęścia. Zarówno obsługa protokołu, jak i jakość usług dostawców internetowych znacznie się poprawiły od tamtych czasów. Narzędzia *Sieć* ułatwiają konfigurację połączeń PPP, sprowadzając cały kłopot do poziomu komplikacji konfiguracji interfejsu sieci Ethernet.

Aby zatem skonfigurować połączenie PPP do internetu, należy zaznaczyć pozycję *Połączenia modemowe* w oknie *Sieć* (patrz rysunki 24.1 i 24.2) i kliknąć przycisk *Właściwości*. Pojawi się okno przedstawione na rysunku 24.7.

Rysunek 24.7.

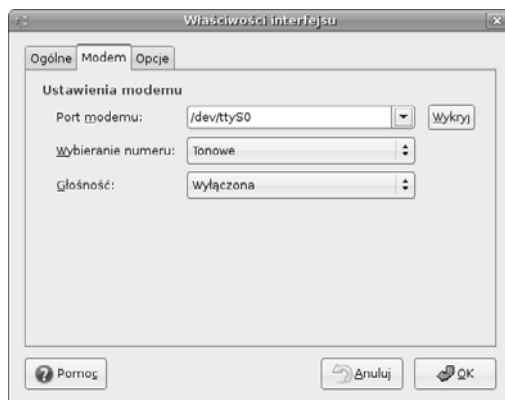
Początkowa zawartość okna konfiguracji połączenia modemowego



Zaznaczamy opcję *Połączenie aktywne* i wypełniamy pozostałe pola. Zawartość tych pól jest specyficzna dla każdego dostawcy internetu. Po wypełnieniu niezbędnych informacji należy przejść na kartę *Modem*, której zawartość przedstawiłem na rysunku 24.8.

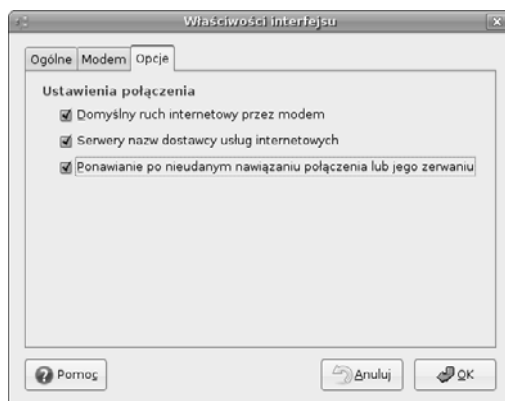
Większość połączeń modemowych wykorzystuje pierwszy port szeregowy systemu Linux, czyli urządzenie `/dev/ttyS0`, ale wybór warto dostosować do sytuacji w danym systemie (zależy od komputera i modemu). Aby automatycznie wykryć modem zainstalowany

Rysunek 24.8.
Konfiguracja modemu



w systemie, należy podłączyć modem do łącza telefonicznego i kliknąć przycisk *Wykryj*. System dokona przeszukania standardowego zbioru urządzeń wykorzystywanych do połączeń modemowych, sprawdzając, czy uda się wykryć sygnał telefoniczny. Większość użytkowników pozostawia domyślne wartości opcji *Wybieranie numeru* (*Tonowe*) i *Głośność* (*Wyłączona*), ale — oczywiście — można to zmienić na jedną z dostępnych opcji pól wyboru. Następnie należy przejść na kartę *Opcje*, której zawartość przedstawiłem na rysunku 24.9.

Rysunek 24.9.
*Konfiguracja
opcji PPP*



Pierwsze dwie opcje w tym oknie są zaznaczone po to, aby po nawiązaniu połączenia modemowego domyślny ruch do internetu był wysyłany właśnie przez nie; druga opcja powoduje, że informacja o serwerach DNS zostanie pobrana od dostawcy internetu. Z reguły zaznaczam również i trzecią opcję, wymuszającą automatyczne ponowienie próby połączenia w przypadku problemu (kiedy np. połączenie zostanie zerwane). Wielu dostawców internetu posiada dużą liczbę modemów, ale jeszcze większą liczbę klientów. W zależności od dostawcy, może okazać się konieczne kilkakrotne ponowienie próby nawiązania połączenia, zanim uda się uzyskać sprawne łącze.

Po zakończeniu konfiguracji połączenia PPP należy kliknąć *OK*, co spowoduje zamknięcie okna i ponowne wyświetlenie głównego okna konfiguracji sieci (patrz rysunki 24.1 i 24.2). Aby aktywować połączenie PPP, wystarczy kliknąć przycisk *Aktywuj* i jeśli wszystko zostało skonfigurowane prawidłowo, jesteśmy w sieci!

Definiowanie i wykorzystanie wielu konfiguracji sieciowych

Jak wspominałem wcześniej, jeśli jesteś szczęśliwym posiadaczem komputera wyposażonego w kilka interfejsów sieciowych, naprawdę nie chcesz aby kilka z nich było dostępnych w tej samej sieci, w tym samym czasie. Taka konfiguracja z pewnością wprowadzi zamieszanie w systemie, ponieważ nie będzie on w stanie zdecydować, który interfejs ma być wykorzystany do komunikacji z tą samą siecią. W oknach przedstawionych na rysunkach 24.3 i 24.4 konfigurowaliśmy dwa różne interfejsy sieciowe przyłączone do dwóch odmiennych sieci, więc ten problem nie zachodził.

Również przypadek jednoczesnego przyłączenia do kilku różnych sieci to dość rzadka sytuacja. O wiele częściej zdarza się, że trzeba ustawić różne konfiguracje tego samego interfejsu sieciowego podczas przyłączania systemu do różnych sieci w różnych lokalizacjach (np. w domu i biurze) lub korzystania z różnych interfejsów sieciowych w odmiennych lokalizacjach. Przewodowy interfejs Ethernet jest znacznie szybszy od bezprzewodowego interfejsu Ethernet, zatem w przypadku laptopa wyposażonego w oba typy interfejsów zawsze warto wykorzystywać łącze przewodowe, o ile to tylko możliwe.

Szerzej ten temat omówię w podpunkcie rozdziału 25., „Inteligentne zarządzanie interfejsami z użyciem narzędzia Network Manager”, ale i w tym miejscu warto krótko wspomnieć o narzędziu *Network Manager*, które można zainstalować z repozytoriów Ubuntu za pomocą programu Synaptic. Narzędzie służy właśnie do przełączania konfiguracji sieciowych w zależności od lokalizacji, niestety, nie odniosłem większych sukcesów podczas jego użytkowania. Do tego samego zadania można wykorzystać podobne narzędzie, o wdzięcznej nazwie *whereami* (więcej informacji o nim można znaleźć pod adresem <http://www.ubuntuforums.org/showthread.php?t=24994>).

Automatyczna konfiguracja sieci to wygodny mechanizm, ale określanie ustawień może powodować problemy szczególnie wtedy, kiedy tego zadania podejmie się osoba niebędąca sieciowym guru, znającym znaczenie wszystkich dziwnie brzmiących określeń i skrótowców. Narzędzie do konfiguracji sieci dostępne w Ubuntu upraszcza natomiast konfigurowanie wielu profili sieciowych, między którymi można bez problemu przełączać się ręcznie, w miarę potrzeb. Proces przełączania konfiguracji, mimo że wykonywany ręcznie, opiera się jednak na decyzji użytkownika (który wie, w jakiej lokalizacji się znajduje) i nie wymaga żadnych dodatkowych ustawień z wyjątkiem dokonania prawidłowej wstępnej konfiguracji interfejsów sieciowych w każdym ze zdefiniowanych profili.

Narzędzie *Sieć* upraszcza definiowanie kombinacji profili sieciowych i pozwala zapisać je pod wspólną nazwą, określaną jako *Położenie*.

W pierwszym kroku definiowania położenia należy określić konfigurację wszystkich interfejsów sieciowych, które mają być użyte w danej fizycznej lokalizacji. Następnie trzeba kliknąć listę rozwijaną *Położenie*, co spowoduje wyświetlenie menu przedstawionego na rysunku 24.10.

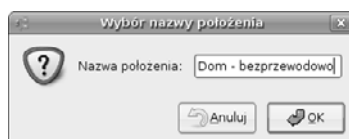
Z listy rozwijanej należy wybrać opcję *Utworzenie położenia*, jak to przedstawiam na rysunku 24.10. Pojawi się okno z polem do wpisania nazwy dla zdefiniowanej konfiguracji interfejsów sieciowych (patrz rysunek 24.11).

Rysunek 24.10.

Tworzenie nowego profilu sieciowego za pomocą listy rozwijanej *Położenie*

**Rysunek 24.11.**

Wprowadzanie nazwy nowego położenia



Wprowadzana nazwa powinna odzwierciedlać sposób i miejsce jej wykorzystania (w przykładzie *Dom - bezprzewodowo*), po wpisaniu należy ją zatwierdzić, klikając przycisk **OK**.

Od tego momentu przy każdej zmianie lokalizacji wystarczy wybrać funkcję z menu systemowego *System/Administracja/Sieć*, wpisać hasło i z listy rozwijanej *Położenie* wybrać odpowiednią nazwę profilu.



Utworzenie nowego położenia nie powoduje zmiany domyślnych ustawień sieci, po prostu do menu *Położenie* dodaje nazwę specyficznej konfiguracji. Po wyborze nowej lokalizacji nie ma jednak łatwego sposobu na powrót do ustawień domyślnych. Dlatego przed zdefiniowaniem różnych położzeń warto utworzyć położenie o nazwie *Domyślne* (lub podobnej), w którym będą zapisane te ustawienia. Dzięki temu w każdym momencie będziemy mieli możliwość powrotu do domyślnej konfiguracji interfejsów sieciowych.

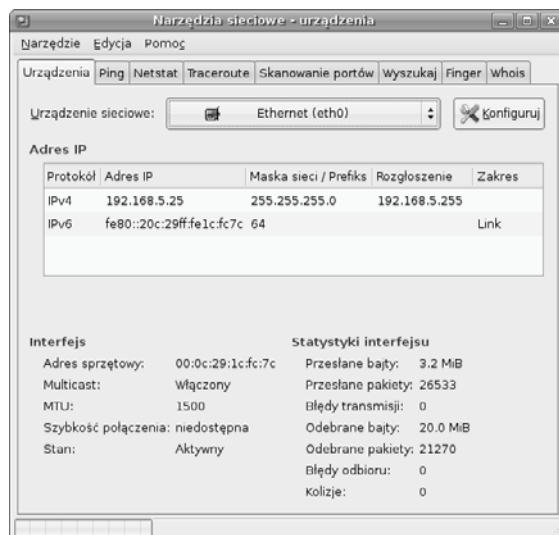
Testowanie sieci z użyciem narzędzi sieciowych dla środowiska GNOME

W obszernej kolekcji łatwych w użyciu narzędzi z interfejsem graficznym, Ubuntu oferuje narzędzie do analizy konfiguracji dowolnego interfejsu sieciowego. Mowa o *Narzędziach sieciowych* dla środowiska GNOME, za których pomocą można zapoznać się z informacjami o konfiguracji sieci, z tego interfejsu można też uruchamiać podstawowe narzędzia diagnostyki sieci. Narzędzia te uruchamia się z menu systemowego *System/Administracja/Narzędzia sieciowe*. Po wpisaniu hasła pojawi się okno przedstawione na rysunku 24.12.

Po pierwszym uruchomieniu aplikacja *Narzędzia sieciowe* wyświetla informacje o urządzeniu *loopback*. Aby zobaczyć informacje o innym interfejsie sieciowym, należy wybrać jego nazwę z listy rozwijanej *Urządzenie sieciowe*. Na rysunku 24.12 demonstruję informacje, jakie mogę uzyskać w jednym z moich systemów o interfejsie sieciowym *eth0*.

Rysunek 24.12.

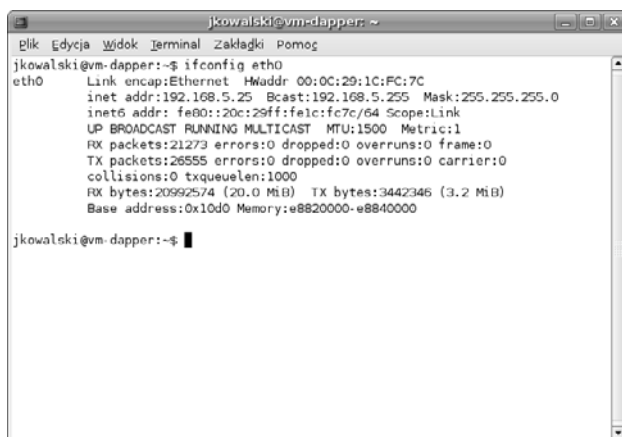
Informacja
o interfejsie *eth0*
w ramach aplikacji
Narzędzia sieciowe



Najprostszy i najszybszy sposób identyfikacji bieżącej konfiguracji jednego z moich interfejsów Ethernet mógłby prawdopodobnie polegać na uruchomieniu polecenia `ifconfig` w oknie terminala GNOME. Na rysunku 24.13 przedstawiam wynik działania tego polecenia z opcją *eth0* (nazwa interfejsu) w tym samym systemie:

Rysunek 24.13.

Informacje
o interfejsie
sieciowym Ethernet
uzyskane w wierszu
poleceń



Jak widać, informacje tekstowe na temat interfejsu Ethernet uzyskane z wiersza poleceń wymagają nieco więcej doświadczenia i umiejętności analizy, w porównaniu z analogicznymi informacjami uzyskanymi w programie graficznym, przedstawionymi na rysunku 24.12.

Oprócz bardziej czytelnych informacji o konfiguracji sieci, aplikacja *Narzędzia sieciowe* pozwala zapoznać się z wynikami działania szeregu narzędzi dostępnych w trybie tekstowym. W ramach okna *Narzędzia sieciowe* dostępne są następujące karty, reprezentujące różne narzędzia diagnostyczne.

- ♦ **Urzędnia:** wyświetla konfigurację i podsumowanie ruchu sieciowego dla każdego z interfejsów dostępnych w systemie. Informacje te odpowiadają tym, które można uzyskać z aplikacji `ifconfig` uruchamianej z wiersza poleceń.
- ♦ **Ping:** wyświetla informacje o możliwości dostępu do zdalnego systemu, uzyskane za pomocą analizy pakietów zwrotnych wysłanych ze zdalnego systemu w odpowiedzi na pakiety ICMP. W oparciu o liczbę pakietów zwrotnych i czas odpowiedzi wyświetlana jest informacja o dostępności lub braku dostępu do hosta. Dane te odpowiadają informacjom, które można uzyskać z aplikacji `ping` uruchamianej z wiersza poleceń.
- ♦ **Netstat:** wyświetla informacje o wszystkich aktywnych i dostępnych w systemie sieciach TCP i UDP. Odpowiadają one danym, które można uzyskać z aplikacji `netstat` uruchamianej z wiersza poleceń.
- ♦ **Traceroute:** wyświetla informacje o routerach pośredniczących (tzw. *hopach*), przez które odbywa się komunikacja ze wskazanym hostem oraz czas niezbędny do przeprowadzania komunikacji pomiędzy każdym z tych systemów. Informacje te odpowiadają danym, które można uzyskać z aplikacji `traceroute` uruchamianej z wiersza poleceń.
- ♦ **Skanowanie portów:** wyświetla informacje o portach i usługach sieciowych dostępnych w zdalnym systemie. Odpowiadają one w przybliżeniu informacjom, które można uzyskać z aplikacji `nmap` uruchamianej z wiersza poleceń.
- ♦ **Wyszukaj:** wyświetla informacje o adresie IP i dostępnych aliasach DNS określonego systemu. Odpowiadają one informacjom, które można uzyskać z aplikacji `nslookup` lub `host`, uruchamianych z wiersza poleceń.
- ♦ **Finger:** wyświetla wszelkie dostępne informacje personalne na temat wskazanego użytkownika zdalnego hosta. Odpowiadają one informacjom, które można uzyskać z aplikacji `finger` uruchamianej z wiersza poleceń. Obecnie niewiele systemów udostępnia informacje `finger`.
- ♦ **Whois:** wyświetla informacje na temat instytucji rejestrującej oraz techniczne dane wskazanej domeny internetowej. Odpowiadają one informacjom, które można uzyskać z aplikacji `whois` lub `bwhois` uruchamianych z wiersza poleceń.

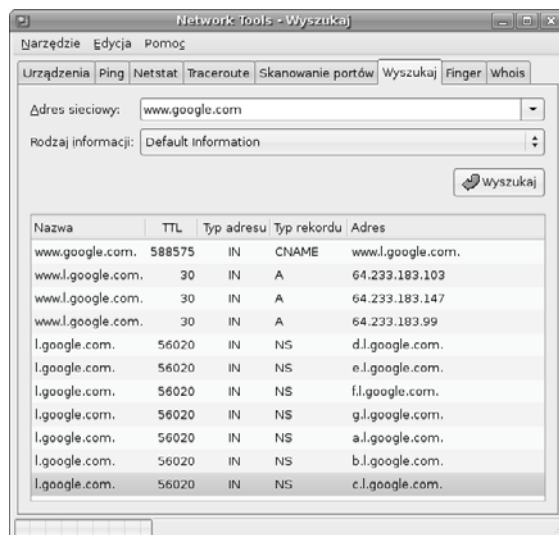
Jako przykład działania aplikacji *Narzędzia sieciowe* niech posłuży wyszukanie adresu IP nazwy hosta `www.google.com`, przedstawione na rysunku 24.14.

Wskazówki dotyczące zabezpieczenia systemu

Bezpieczeństwo systemu to zagadnienie szerokie i o wielu obliczach. Obejmuje bezpieczeństwo fizyczne, uwierzytelnianie użytkowników, zabezpieczenia plików i systemów plików itp.; na temat bezpieczeństwa informatycznego napisano mnóstwo książek, a z pewnością powstanie o wiele więcej. Jak wspomniałem na wstępie tego rozdziału, bezpieczeństwo w przeróżnych formach staje się coraz poważniejszym zagadnieniem, ponieważ sieci

Rysunek 24.14.

Graficzne
przedstawienie
wyników wyszukiwania
adresu IP hosta



są już ogólnie dostępne, a wraz z ich powszechnością zwiększa się dostępność zdalnych systemów oraz narzędzi do skanowania, wyszukiwania słabych punktów, eksplorowania i włamywania się do zdalnych maszyn.

Poniżej przedstawię kilka sugestii dotyczących poprawiania bezpieczeństwa systemów w sieci. Jak można się spodziewać, obejmują one niektóre aspekty innych zagadnień związanych z bezpieczeństwem, ale każda z nich posiada pewne cechy szczególne, warte rozważenia.

1. Domowi użytkownicy „fabrycznie skonfigurowanych” urządzeń typu punkt dostępu (ang. *access point*) lub brama (ang. *gateway*) powinni ustawić solidne hasło, zanim przyłączą się do internetu. O ile to możliwe, należy również zmodyfikować nazwę użytkownika upoważnionego do zalogowania się w takim urządzeniu. Gdybym dostał grosz za każde skuteczne włamanie do urządzenia sieciowego, w którym nie zostało zmienione domyślne hasło, nie wiedziałbym, ile mam groszy, ponieważ większość tych włamań na zawsze pozostanie nieujawniona.
- ♦ Należy wyłączyć zbędne usługi w systemie. Do ich identyfikacji można posłużyć się kartą *Skanowanie portów* programu *Narzędzia sieciowe*. Trzeba wyłączyć wszelkie usługi, które nie są używane, wykorzystując do tego celu narzędzie *Boot-Up Manager*, omówione szczegółowo w rozdziale 19., w podrozdziale „Optymalizacja procesu rozruchowego dystrybucji Ubuntu”.
 - ♦ Należy usunąć konta użytkowników, które nie są już używane w systemie. Dotyczy to również kont utworzonych na potrzeby usług systemowych, które odinstalowano z systemu.
 - ♦ System zawsze powinien być zaktualizowany za pomocą *Menedżera aktualizacji*. Wszak istnieje jakiś powód (i, wiercie mi, jest to powód poważny), dla którego deweloperzy ciężko pracują nad poprawkami do systemu.

- ♦ Należy regularnie monitorować pliki dziennika systemowego. Pliki `/var/log/messages` i `/var/log/syslog` mogą być cennym źródłem informacji o tym, że ktoś próbuje włamać się do systemu i w jaki sposób to robi.
- ♦ Hasła powinny być zmieniane regularnie. Zależność administracji systemu Ubuntu od mechanizmu `sudo` (zamiast tradycyjnego sposobu administracji systemem wykorzystującego konto `root`) stanowi użyteczną dla bezpieczeństwa komplikację zadania włamywacza, ale pamiętaj, Twój dedykowany włamywacz gdzieś w Pekinie nie ma kompletnie nic innego do roboty prócz poszukiwania dróg wejścia.

Jak wspominałem wcześniej, bezpieczeństwo jest obowiązkiem każdego użytkownika komputera. Istnieją użyteczne aplikacje pomagające testować i sprawdzać stan zabezpieczeń systemu, co może być tak samo pouczające jak użyteczne. Oto moje faworytki.

- ♦ `chkrootkit`: sprawdza istnienie tzw. *rootkitów*, czyli specjalnie spreparowanych aplikacji instalowanych w systemie, do którego udało się włamać. Takie *rootkity* ułatwiają włamywaczowi „powrót na miejsce zbrodni” i zbierają informacje o nazwach użytkowników oraz hasłach w systemie.
- ♦ `nmap`: sprawdza udostępniane usługi sieciowe w systemie, pozwalając znaleźć potencjalne słabe punkty.

Jak można się spodziewać, obie te aplikacje są dostępne w repozytoriach Ubuntu i można je z łatwością zainstalować za pomocą programów `apt-get`, `aptitude` czy menedżera pakietów `Synaptic`.

Instalacja zapory sieciowej

Zapora sieciowa to termin wykorzystywany do określenia systemu działającego między dwoma (lub większą liczbą) systemami komputerowymi, monitorującego ruch sieciowy i zarządzającego nim. Podobnie jak ściana przeciwogniowa w samochodzie, zapobiegająca przedostaniu się ognia z komory silnika do kabiny pasażerskiej, zapora sieciowa chroni przed przedostaniem się złośliwego, nadmiarowego lub po prostu zbędnego ruchu sieciowego z jednej sieci do drugiej. Wiele zapór sieciowych realizuje dodatkowe funkcje, takie jak NAT (ang. *Network Address Translation*), ale głównym ich zadaniem jest zabezpieczenie systemów przed atakami sieciowymi.

W nowoczesnych systemach Linux funkcję zapory sieciowej z reguły realizuje moduł jądra obsługujący funkcję filtra sieciowego. Podsystem ten nosi nazwę *netfilter*, a do jego obsługi służy polecenie `iptables`. **Filtrowanie pakietów** wykorzystuje analizę pakietów sieciowych i wykonuje określone działania w oparciu o ich cechy, takie jak źródło, cel, typ oraz inne informacje. Obsługa filtrowania pakietów jest wbudowana w jądro Linuksa, każdy system Linux podłączony do sieci może funkcjonować jako własna zapora sieciowa monitorująca ruch sieciowy, wchodzący i wychodzący z działających w nim procesów, i zarządzająca nim. Oczywiście, jako zapora sieciowa może służyć również dedykowane urządzenie lub system Linux i jego mechanizm *iptables/netfilter*, wielu dostawców rozwiązań sieciowych oferuje tego typu produkty.

Wśród fanów Ubuntu od dłuższego czasu trwa debata na temat, czy ten system w ogóle potrzebuje zapory sieciowej. Standardowa instalacja Ubuntu nie udostępnia żadnych usług sieciowych, zatem nie ma portów, które powinny być dodatkowo chronione. Oczywiście, nie jest to prawda w przypadku serwerów Ubuntu, które udostępniają porty usług, takie jak DNS, poczta e-mail, serwer WWW itp. W takim przypadku zainstalowanie zapory sieciowej z odpowiednią konfiguracją jest jak najbardziej wskazane.



Jeśli system Ubuntu jest wykorzystywany w środowisku zabezpieczonym zaporą sieciową, prawdopodobnie nie ma konieczności konfigurowania takiej zapory w systemie. Należy jednak upewnić się, że tego typu centralna zaporą sieciową prawidłowo realizuje powierzone jej funkcje. W tym celu warto skontaktować się z producentem urządzenia, administratorami sieci w środowisku korporacyjnym lub akademickim, czy też z dostawcą usług internetowych. Sam fakt występowania naklejki „Firewall” na obudowie urządzenia nie oznacza, że nadaje się do czegokolwiek, oprócz podgrzewania powietrza.

Również w biurkowych zastosowaniach Ubuntu z czasem może okazać się, że zostaną zainstalowane aplikacje otwierające porty sieciowe, a zaporą wykorzystująca mechanizm *iptables/netfilter* nie wprowadza nadmiernego obciążenia, zatem z reguły zalecam użytkownikom skonfigurowanie choćby prostej zapory sieciowej. Kiedy z czasem system stanie się bardziej wyekspozowany na zdalne działania, zaporą sieciową będzie już na miejscu, by go chronić. W przyszłości może okazać się, że konieczna będzie ponowna konfiguracja zapory sieciowej, gdy zdecydujemy się udostępnić kilka portów na zewnątrz. Instalacja prostej zapory sieciowej ma sens również wtedy, gdy konfigurujemy system dla przyjaciół, rodziny lub w niewielkiej firmie, gdzie nie mamy większego wpływu na to, jakie aplikacje będą instalowane i uruchamiane w systemie.

Podstawy zapór sieciowych i filtrowania pakietów w Linuksie

Filtrowanie pakietów w formie dostępnej w jądrze Linuksa serii 2.6.x jest kombinacją mechanizmu (API) obsługiwanego przez moduł jądra *netfilter* oraz interfejsu użytkownika obsługiwanego przez polecenie *iptables*. Interfejs *iptables* jest jednym z modułów jądra opartych na mechanizmie realizowanym przez *netfilter*. Pozostałymi modułami wspomagającymi *netfilter* są m.in. moduł realizujący funkcje NAT (ang. *Network Address Translation*), pozwalającą kilku komputerom sieci lokalnej łączyć się z internetem za pośrednictwem jednego publicznego adresu IP, czy też moduł implementujący i obsługujący mechanizm śledzenia połączeń. Do końca tego rozdziału będę już tylko omawiał mechanizm *iptables*, ponieważ jest najczęściej wykorzystywany w nowoczesnych systemach Linuksa do obsługi zapór sieciowych i filtrowania pakietów.

Interfejsy programów *iptables* i *netfilter* stanowią w rzeczywistości czwartą generację rozwiązań związanych z zaporami sieciowymi dla Linuksa. Oryginalne rozwiązanie mechanizmu filtrowania pakietów w Linuksie nosiło nazwę *ipfw*, zostało zapożyczone z systemów BSD i wprowadzone do jądra Linuksa w wersji 1.1 przez Alana Coksa. Narzędzie to zostało zaprojektowane z myślą o tworzeniu prostych zapór sieciowych i routerów, a wykorzystuje inspekcję i filtrowanie pakietów. Do jąder serii 2.0 zostało dodane narzędzie *ipfwadmin* oraz odpowiednie zmiany w API *ipfw*, upraszczające tworzenie zapór sieciowych wykorzystujących ten mechanizm. Tę wersję nazywa się drugą generacją zapór sieciowych dla Linuksa. Trzecia generacja mechanizmów filtrowania pakietów dla Linuksa

wiązała się z całkowitym przepisaniem podsystemu i wprowadzeniem nowego narzędzia, czyli ipchains. Nastąpiło to w jądrze w wersji 2.1. Obecny mechanizm netfilter oraz związany z nim iptables został wprowadzony w jądrze w wersji 2.4 i od tamtej pory jest standardowym mechanizmem filtrowania pakietów, translacji adresów sieciowych i portów oraz manipulacji pakietami, również w aktualnej serii 2.6 jąder Linuksa.

Filtrowanie pakietów w Linuksie działa w oparciu o analizę pakietów przychodzących i wychodzących. Na tych pakietach wykonywane są operacje w oparciu o reguły filtrowania, załadowane do mechanizmu netfilter za pomocą polecenia iptables. Domyślnie polecenie iptables obsługuje trzy typy reguł, zwane łańcuchami. Domyślnymi łańcuchami są INPUT, OUTPUT i FORWARD. Reguły w łańcuchu INPUT są wykorzystywane do analizy i przetwarzania pakietów przychodzących, których adresatami są porty w systemie lokalnym zapory. Reguły w łańcuchu OUTPUT służą do analizowania i przetwarzania pakietów wychodzących z systemu lokalnego. Reguły w łańcuchu FORWARD analizują i przetwarzają pakiety przekazywane za pośrednictwem zapory.

Każdy z domyślnych łańcuchów reguł posiada własne reguły filtrowania. Można również definiować własne zestawy reguł i wykorzystywać je do własnych celów. Wiele nowoczesnych systemów Linux i Unix jest dostarczanych z predefiniowanymi łańcuchami INPUT, OUTPUT i FORWARD, ładowanymi podczas uruchamiania systemu. Własne zestawy reguł można definiować z użyciem łatwych w obsłudze narzędzi, również graficznych, z których jedno zostanie omówione pod koniec tego rozdziału.

Inne moduły uzupełniające mechanizm netfilter korzystają z własnych łańcuchów. Przykładowo moduł NAT wykorzystuje tabelę NAT składającą się z trzech wbudowanych łańcuchów: OUTPUT, POSTROUTING i PREROUTING. Specjalizowane operacje do manipulowania pakietami stosują tabelę mangle, zawierającą łańcuchy FORWARD, INPUT, OUTPUT, PREROUTING i POSTROUTING. Moduł śledzenia połączeń wykorzystuje tabelę raw, zawierającą łańcuchy OUTPUT i PREROUTING.

Do przeglądania, tworzenia i modyfikowania łańcuchów mechanizmu netfilter użytkownik musi mieć uprawnienia konta root. Polecenia iptables można umieścić w skrypcie wykonywanym w ramach procesu rozruchowego systemu lub przy pomocy polecenia sudo.

Instalacja i konfiguracja zapory sieciowej z użyciem programu Lokkit

Jak wspomniałem w poprzednim podrozdziale, istnieje wiele różnych programów pomagających skonfigurować i aktywować zaporę sieciową w Ubuntu. Należy do nich Lokkit (omówiony w tym punkcie), Firestarter, Fwbuilder, Guarddog i wiele innych. Uważam, że Lokkit świetnie sprawuje się w konfiguracji podstawowej zapory sieciowej, zadaje odpowiednie pytania i jest łatwy w użyciu, zatem właśnie jego wybrałem do omówienia.

Instalacja pakietu Lokkit

Mimo iż kwestia, czy stosować zaporę sieciową w Ubuntu, jest dość gorącym tematem na forach, nie jest ona instalowana w domyślnej instalacji tej dystrybucji. Jednak wśród pakietów oprogramowania dostępnych w repozytoriach Ubuntu istnieją odpowiednie

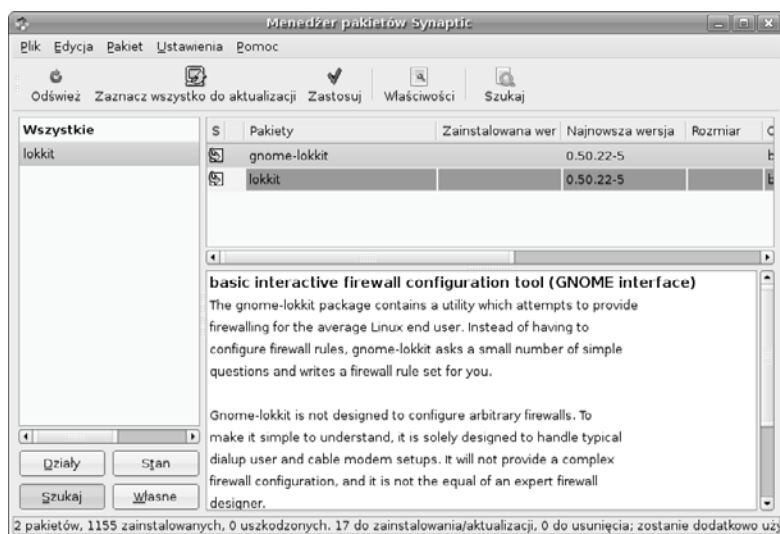
narzędzia, które łatwo zainstalować za pomocą poleceń `apt-get`, `aptitude` i menedżera pakietów Synaptic. Proponuję zainstalować pakiet Lokkit, który jest udostępniany w ramach pakietu *lokkit*. Sugeruję również instalację pakietu *gnome-lokkit*, który oferuje czytelny i łatwy w użyciu graficzny interfejs użytkownika upraszczający konfigurację i dostosowanie zapory sieciowej do konkretnych potrzeb.

Aby zainstalować pakiet *lokkit* z wiersza poleceń, wystarczy posłużyć się poleceniami `sudo apt-get install lokkit` lub `sudo aptitude -r install lokkit`. Jeśli pracujemy w systemie niewyposażonym w środowisko graficzne, nie ma sensu instalowania pakietu *gnome-lokkit*.

Aby zainstalować pakiet z użyciem narzędzia graficznego, należy uruchomić menedżer pakietów Synaptic (*System/Administracja/Menedżer pakietów Synaptic*) i podać hasło użytkownika. Po uruchomieniu programu trzeba kliknąć przycisk *Szukaj* i w polu tekstowym wpisać *lokkit*, wybierając z listy rozwijanej opcję *Opis i nazwa*, po czym kliknąć *Szukaj*. Po zakończeniu wyszukiwania należy zaznaczyć pakiet *lokkit*, kliknąć jego pozycję prawym przyciskiem myszy i z menu podręcznego wybrać opcję *Zaznacz do instalacji*.

Po zaznaczeniu do instalacji pakietu *lokkit* należy powtórzyć operację dla pakietu *gnome-lokkit*, który jest graficzną nakładką na *lokkit*, służącą do konfigurowania zapory sieciowej. Na rysunku 24.15 przedstawiam okno programu Synaptic z pakietami *lokkit* i *gnome-lokkit* zaznaczonymi do instalacji.

Rysunek 24.15.
Instalacja pakietu Lokkit i pakietów pomocniczych za pomocą programu Synaptic



Przy zaznaczaniu do instalacji pakietu *lokkit* pojawi się okno z informacją o pakietach zależnych, które muszą być zainstalowane do sprawnej pracy pakietu *lokkit*. Należy kliknąć przycisk *Zaznacz*, co spowoduje, że również te pakiety zostaną automatycznie zaznaczone do instalacji.

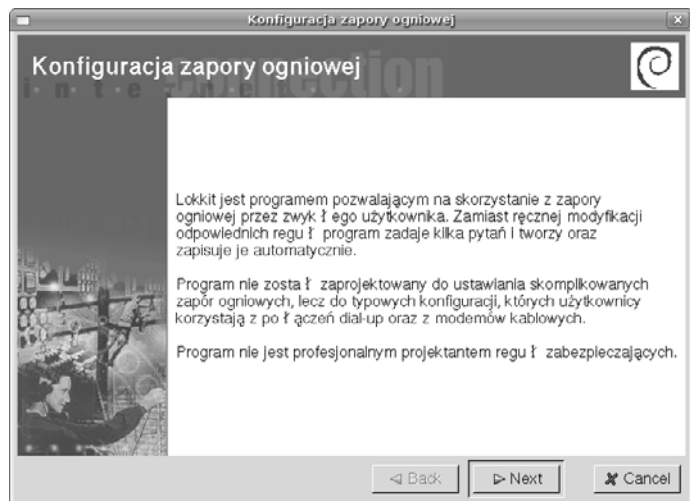
Po zaznaczeniu pakietów należy kliknąć ikonę *Zastosuj* w pasku narzędzi programu Synaptic, co spowoduje instalację pakietu *lokkit* i jego graficznej nakładki. Po zakończeniu instalacji można zakończyć program Synaptic.

Wykorzystanie programu Lokkit do konfiguracji prostej zapory sieciowej

Instalacja pakietów *lokkit* i *gnome-lokkit* nie dodaje pozycji w menu systemowym, co jest zrozumiałe, gdyż zwykle program ten jest uruchamiany tylko raz na danym komputerze w celu wstępnego skonfigurowania zapory sieciowej. Aby uruchomić graficzne narzędzie *gnome-lokkit*, należy wywołać polecenie `gksudo gnome-lokkit` z dowolnego wiersza poleceń systemu Ubuntu oraz wpisać hasło użytkownika w oknie, które się pokaże. Zostanie wyświetlone okno programu *gnome-lokkit*, zawierające podstawowe informacje o programie *Lokkit* (patrz rysunek 24.16).

Rysunek 24.16.

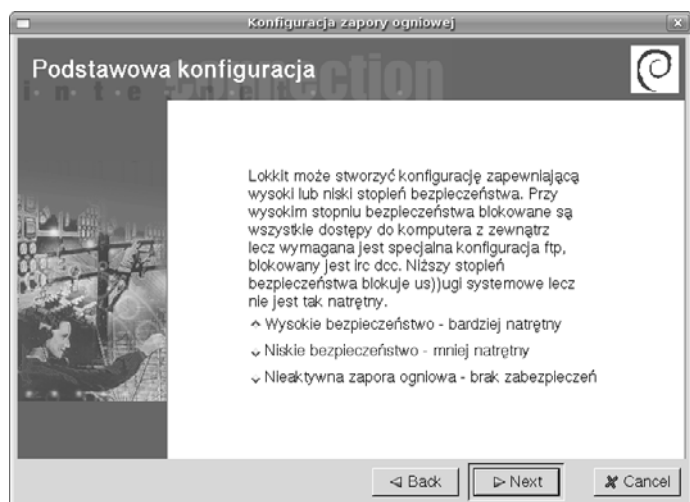
Pierwsza strona kreatora konfiguracji zapory sieciowej Lokkit



Aby kontynuować, należy kliknąć przycisk *Next*. Pojawi się okno przedstawione na rysunku 24.17.

Rysunek 24.17.

Wybór typu zapory sieciowej



W większości przypadków najlepszym wyborem jest opcja *Low Security*. Jak wspominałem wcześniej, domyślna instalacja Ubuntu Desktop nie otwiera żadnych portów, zatem zaporą sieciową to jedynie dodatkowe zabezpieczenie „na wszelki wypadek”, gdyby w systemie jednak zostały otwarte jakieś porty (lub zainstalowane zostały pakiety, które to robią). Konfigurując zaporę sieciową w systemie Ubuntu Server, warto zaznaczyć opcję *High Security*, ale w takim wypadku należy liczyć się z koniecznością konfigurowania reguł filtrowania utworzonych przez Lokkit (albo zmiany konfiguracji zainstalowanych usług), aby upewnić się, że usługi, które ma świadczyć serwer, nie są blokowane przez zaporę. Żeby kontynuować, należy kliknąć przycisk *Next*. Pojawi się okno przedstawione na rysunku 24.18.

Rysunek 24.18.

*Opcja zaufania
do hostów z sieci
lokalnej*

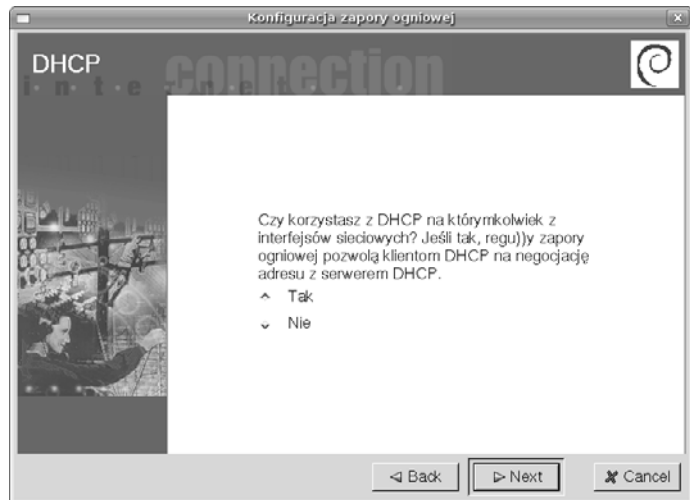


Okno dialogowe z rysunku 24.18 zawiera pytanie, czy hosty sieci lokalnej (czyli hosty o identycznej pierwszej części adresów sieciowych) mają być traktowane jako zaufane. Jeśli np. system ma adres IP równy *192.168.6.121*, wybór opcji *Yes* spowoduje, że wszystkie hosty o adresach typu *192.168.XXX.XXX* będą traktowane jako uprawnione do dostępu do usług świadczonych przez system. Jeśli w sieci lokalnej występuje większa liczba komputerów, a konfigurowany system nie jest podłączony bezpośrednio do internetu, na to pytanie warto odpowiedzieć *Yes*. Aby kontynuować, należy kliknąć *Next*, pojawi się okno przedstawione na rysunku 24.19.

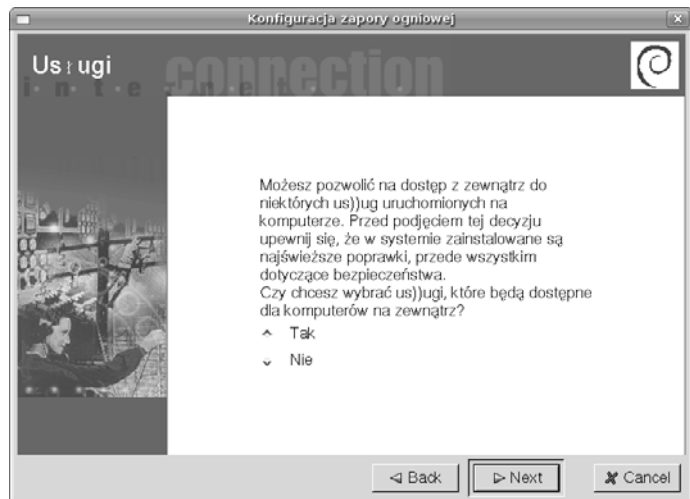
Okno dialogowe przedstawione na rysunku 24.19 służy określeniu, czy system korzysta z usługi DHCP. Jeśli w systemie jest używany adres dynamiczny przydzielany przez DHCP lub ma on służyć jako serwer tej usługi, na to pytanie należy odpowiedzieć *Yes*. Aby kontynuować, należy kliknąć *Next*, a pojawi się okno przedstawione na rysunku 24.20.

W oknie przedstawionym na rysunku 24.20 można wybrać usługi działające w konfigurowanym systemie, do których ma być możliwy zdalny dostęp. Jeśli na danym komputerze nie działają (i nie będą w przyszłości działać) usługi, takie jak DNS, FTP, serwer e-mail czy serwer WWW, należy zaznaczyć opcję *No*. Jeśli jednak któreś z tych usług są udostępniane, należy zaznaczyć opcję *Yes*. Aby kontynuować, należy kliknąć *Next*. Pojawiają się kolejne okna pozwalające określić dostępność usług (w kolejności): *serwera WWW*, *serwera poczty*, *SSH* i *Telnet*. Sugeruję zaznaczyć opcję *Yes* dla wszystkich tych usług,

Rysunek 24.19.
Wybór opcji DHCP



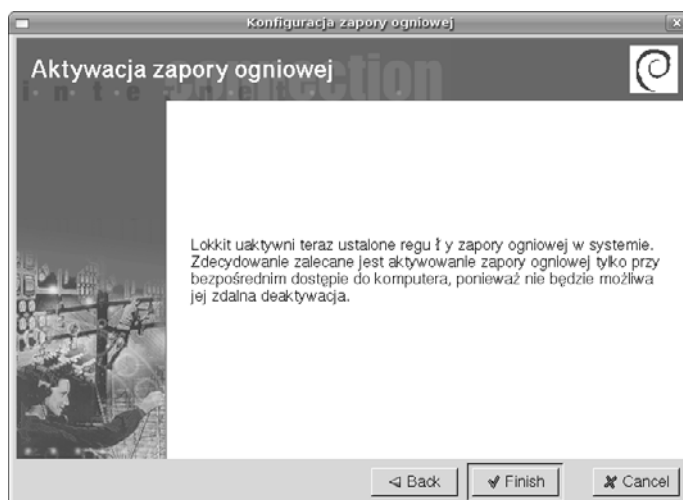
Rysunek 24.20.
Otwieranie dostępu do usług serwowanych przez system



z wyjątkiem usługi *Telnet*, który jest starym mechanizmem o bardzo niskim poziomie bezpieczeństwa służącym do zdalnego łączenia się z systemem. Ten przestarzały standard już dawno został zastąpiony bezpiecznym mechanizmem *SSH*.

Po udzieleniu odpowiedzi na pytania (lub w przypadku, gdy w oknie przedstawionym na rysunku 24.20 udzielono odpowiedni *No*) pojawi się okno przedstawione na rysunku 24.21.

Aby aktywować zaporę sieciową, należy kliknąć przycisk *Finish*, co spowoduje zamknięcie programu *gnome-lokkit*. Jeśli zmieniliśmy zdanie, w tym miejscu wystarczy kliknąć przycisk *Cancel* — program można ponownie uruchomić w innym momencie i spokojnie skonfigurować opcje zapory sieciowej. Jeśli jednak klikniemy *Yes*, *lokkit* wykona podstawowe testy zapory sieciowej, po czym uruchomi ją, dopisując jej uruchamianie do procedury rozruchowej systemu przez utworzenie skryptu */etc/init.d/lokkit* oraz dowiązań dla odpowiednich poziomów uruchomienia systemu.

Rysunek 24.21.*Aktywacja zapory sieciowej*

Podsumowanie

W tym rozdziale omówiłem podstawowe zagadnienia związane z konfiguracją w systemie Ubuntu przewodowych, bezprzewodowych i modemowych łączy sieciowych. Dla środowiska GNOME dostępne są łatwe w użyciu narzędzia graficzne upraszczające konfigurację interfejsów tego typu, co eliminuje konieczność stosowania tradycyjnej magii wiersza poleceń z użyciem polecenia `ifconfig`. W tym rozdziale podałem też podstawowe informacje na temat zabezpieczania sieciowego systemu Ubuntu i omówiłem prostą konfigurację zapory sieciowej.

W rozdziale 25. bardziej szczegółowo omawiam wykorzystanie technologii bezprzewodowych w systemie Ubuntu. Zamieściłem tam szczegółowe omówienie konfiguracji i wykorzystania interfejsów sieci bezprzewodowej, o których tutaj zaledwie wspomniałem. W rozdziale 25. zawarłem ponadto szczegółowe informacje na temat instalacji i konfiguracji w Ubuntu sterowników systemu Windows, co jest bardzo przydatne w przypadku posiadania problematycznych urządzeń Ethernet (i innego sprzętu), dla których w standardowym jądrze Linuksa nie ma działających sterowników.

Rozdział 25.

Sieć bez kabli

W tym rozdziale:

- ♦ Terminologia sieci bezprzewodowych
- ♦ Konfiguracja urządzeń sieci bezprzewodowej
- ♦ Wykorzystanie w Ubuntu sterowników dla systemu Windows

Współczesne wymagania dotyczące wymiany danych byłyby farsą, gdyby nie wszechobecność łączności bezprzewodowej. Nieważne, czy mamy na myśli telefon komórkowy łączący się ze stacją bazową, urządzenie PDA wymieniające dane z komputerem PC za pomocą Bluetooth lub podczerwieni, czy laptopy łączące się wzajemnie za pośrednictwem sieci bezprzewodowej z użyciem punktu dostępowego ukrytego w hotelowym schowku na miotły. Możliwość podróżowania bez konieczności przewożenia przewodów, a może nawet adapterów pozwalających połączyć urządzenie A z urządzeniem B, daje niesamowite poczucie wolności szczególnie w czasach, gdy ochrona na lotniskach patrzy wilkiem na przewożone w walizkach sploty kabli, zasilaczy i sprzętu elektronicznego, jako żywo przywodzące na myśl laboratorium nowoczesnego doktora Victora von Frankensteina (niespokrewnionego ze mną). Nareszcie wolni!

W rozdziale 17. omówiłem zagadnienia podłączenia urządzeń PDA i telefonów do systemu Ubuntu, w tym rozdziale skupię się na sieciach bezprzewodowych. Podobnie jak w większości dziedzin, w których króluje zaawansowana technologia, sieci bezprzewodowe mają własny specyficzny słownik i oszałamiający zbiór skrótowców, wyglądających podobnie, ale o zupełnie odmiennym znaczeniu. W pierwszym podrozdziale tego rozdziału skupię się zatem na ogólnym opisie technologii związanych z sieciami bezprzewodowymi. W dalszej części zajmę się zagadnieniami konfiguracji i obsługi technologii dostępnych obecnie w Linuksie, następnie wyjaśnię sposoby pracy z urządzeniami sieciowymi oficjalnie nieobsługiwanymi w Linuksie (co jest samo w sobie dość imponującą sztuczką) oraz przekażę kilka cennych, moim zdaniem, uwag dotyczących użycia w systemie Ubuntu Linux sterowników napisanych dla systemu Microsoft Windows.

Ogólne informacje o sieciach bezprzewodowych

Sieci bezprzewodowe to niezwykle użyteczna technologia, zarówno w pracy biurowej (żeby już nigdy w trakcie ważnej konferencji nie trzeba było zastanawiać się, które gniazdo jest do sieci Ethernet, które do telefonicznej, a w którym sygnału nie ma w ogóle), a przede wszystkim w trakcie podróży, gdzie mamy możliwość podłączenia laptopa lub PDA do sieci w bibliotece, kawiarni, hotelu i w wielu innych miejscach publicznych. Sieci bezprzewodowe zapewniają ponadto doskonały mechanizm do łączenia sieci komputerowych zlokalizowanych w niedostępnych lokalizacjach lub modernizacji istniejących struktur, gdzie nie jest możliwa modernizacja, taka jak położenie dodatkowego okablowania. Z pominięciem pewnych zagadnień bezpieczeństwa, sieci bezprzewodowe pomagają uniknąć wiercenia dziur w Watykanie czy Luwrze.

W tym rozdziale zdecydowałem się na pewne uproszczenie. Przez pojęcie sieci bezprzewodowych rozumiem bowiem technologie pozwalające komunikować się komputerom z wykorzystaniem standardowych protokołów sieciowych, ale bez użycia przewodów sieciowych. W ogólnym rozumieniu sieci bezprzewodowe to sieci lokalne (LAN), których stacje robocze wykorzystują do połączeń sieciowe standardy przemysłowe, m.in. *Institute of Electrical and Electronics Engineers* (IEEE) 802.11. Oryginalny standard 802.11 został opracowany w 1997 roku i definiuje częstotliwości, przepustowości oraz metody dostępu do nośnika wykorzystywane w komunikacji między uczestnikami sieci bezprzewodowej. Standard ten był przez lata unowocześniany przez tworzenie kolejnych podwersji o ogólnym zapisie 802.11X, które bardziej szczegółowo opiszę w dalszej części rozdziału.

Pomijając takie szczegóły jak częstotliwości fali radiowej czy prędkości wymiany danych, istnieją dwa podstawowe typy sieci bezprzewodowych: sieci typu *ad hoc* oraz sieci zarządzane. Ich podstawowe cechy (oraz różnice) są następujące.

- ♦ **Sieci *ad hoc*:** określane również jako sieci bezprzewodowe typu peer-to-peer składają się z określonej liczby komputerów posiadających interfejsy sieci bezprzewodowej, komunikujące się bezpośrednio ze wszystkimi pozostałymi komputerami w ramach danej sieci. Dzięki temu komputery mają możliwość wymiany plików i drukarek, ale nie mają dostępu do zasobów internetu, chyba że jeden (lub więcej) z komputerów w sieci *ad hoc* jest przyłączony do internetu i został skonfigurowany do pracy w charakterze bramy, mostu lub routera do tej sieci. Wielbicieli akronimów ucieszą się z tego, że sieci *ad hoc* mają — oczywiście — własny akronim: IBSS (ang. *Independent Basic Service Sets*).
- ♦ **Sieci zarządzane (*managed*):** określane również jako infrastruktury sieci bezprzewodowych wykorzystują punkt dostępowy (*access point*, nazywany też czasem stacją bazową, *base station*), zarządzający komunikacją między poszczególnymi stacjami sieci bezprzewodowej, działający w pewnym sensie jako odpowiednik koncentratora lub przełącznika sieci przewodowych. Punkt dostępowy jest najczęściej przyłączony do sieci przewodowej i działa dodatkowo jako brama między siecią bezprzewodową a przewodową. Punkty dostępowe są najczęściej dedykowanymi urządzeniami (określanymi akronimem HAP, ang. *Hardware Access Point*), np. są to urządzenia Apple Airport lub popularne

punkty dostępowe firm LinkSys, NetGear, Cisco itp. Punktami dostępowymi mogą być również komputery wyposażone w odpowiednie oprogramowanie (określane akronimem SAP, ang. *Software Access Points*) oraz w interfejsy sieci bezprzewodowej i przewodowej. Zarządzane sieci bezprzewodowe to dla wielbicieli akronimów BSS-y (ang. *Basic Service Set*, to N wiele komputerów i jeden punkt dostępowy) oraz ESS-y (ang. *Extended Service Set*, to N wiele punktów dostępowych i N wiele komputerów, które tworzą jedną logiczną podsieć).

Większe obszary wymagające pokrycia siecią bezprzewodową mogą wykorzystywać większą liczbę punktów dostępowych i mechanizm roamingu, tzn. możliwość dostosowania się połączenia bezprzewodowego stacji roboczej do dostępnego punktu dostępowego i dynamiczne przełączanie między nimi (przy zmianie lokalizacji). Ten proces jest z reguły niewidoczny dla użytkownika, choć niektóre konfiguracje wymagają podawania haseł lub innej formy uwierzytelniania przy przełączaniu między punktami dostępowymi. Wielu producentów urządzeń sieci bezprzewodowej oferuje również specjalizowane urządzenia określane nazwą **punktów rozszerzających** (ang. *extension point*), służące do zwiększania zasięgu sieci bezprzewodowej. Obsługa mechanizmu roamingu lub punktów rozszerzających jest zależna od producenta sprzętu. Zgodnie z obowiązującą regułą, największą szansę wzajemnej kompatybilności sprzętu do obsługi sieci bezprzewodowych będziemy mieli wówczas, gdy cały sprzęt będzie pochodził od tego samego producenta.

Najważniejszym zagadnieniem przy konfiguracji sieci bezprzewodowej jest zgodność niskopoziomowych standardów sieci bezprzewodowych. Jeśli komputer nie jest w stanie skomunikować się z punktem dostępowym, nie ma mowy o skonfigurowaniu sieci. Choć IEEE 802.11 jest publicznie dostępnym standardem komunikacji bezprzewodowej, istnieje kilka różnych protokołów 802.11 oraz związanych z nimi standardów, a nie wszystkie z nich są wzajemnie kompatybilne. Oto lista aktualnie istniejących standardów 802.11, wraz z prędkościami transferu danych i deklaracjami zgodności.

- ♦ **802.11a**: standard komunikacji wykorzystujący regulowaną częstotliwość 5,0 GHz, oferujący maksymalną przepustowość na poziomie od 25 do 54 Mbit/s. Nie jest kompatybilny z żadnym innym standardem 802.11.
- ♦ **802.11b**: standard komunikacji wykorzystujący nieuregulowaną częstotliwość 2,4 GHz, oferujący maksymalną przepustowość od 5 do 11 Mbit/s. Standard 802.11b jest kompatybilny w przód ze standardem 802.11g i proponowanym standardem 802.11n.



Niektórzy producenci oferują karty bezprzewodowe typu *Dual Band* obsługujące zarówno standard 802.11a, jak i 802.11b, działające w odpowiednim zakresie częstotliwości i z prędkościami komunikacji dostosowanymi do takiego typu połączenia, jakie uda się nawiązać.

- ♦ **802.11g**: standard komunikacji wykorzystujący nieuregulowaną częstotliwość 2,4 GHz, oferujący maksymalną przepustowość od 25 do 54 Mbit/s. Standard 802.11g jest zgodny wstecz ze standardem 802.11b i w przód z proponowanym standardem 802.11n.
- ♦ **802.11n**: standard komunikacji wykorzystujący nieuregulowaną częstotliwość 2,4 GHz, oferujący maksymalną przepustowość od 100 do 200 Mbit/s. Standard 802.11g jest zgodny wstecz ze standardami 802.11b i 802.11g.



W czasie pisania tej książki 802.11n nie był jeszcze ustanowionym standardem. Ta specyfikacja była nadal poddana dyskusji, a oczekiwany termin ustanowienia standardu to rok 2008. Jednakże brak oficjalnego zatwierdzenia standardu 802.11n nie powstrzymuje producentów sprzętu przed wytwarzaniem urządzeń zgodnych z tą specyfikacją. Az do opublikowania oficjalnej specyfikacji 802.11n należy jednak trzymać się starej reguły: „Wszystkie urządzenia kupuj u tego samego producenta”. Większość producentów sprzętu testuje swoje urządzenia głównie z innymi oferowanymi przez siebie urządzeniami. W najgorszym wypadku będziemy mieli tylko jednego producenta, aby zgłaszać reklamacje.

Urządzenia sieci bezprzewodowych są wyposażone w różne rozwiązania związane z zapewnieniem bezpieczeństwa, począwszy od kontroli nazw sieciowych, aż po mechanizmy kryptograficzne. Najczęściej stosowane są następujące rozwiązania.

- ♦ **Filtrowanie MAC** (ang. *Media Access Control*). Ten mechanizm bezpieczeństwa wymaga od punktu dostępowego możliwości zdefiniowania listy adresów interfejsów sieciowych Ethernet (MAC), które mogą się do niego przyłączyć. Filtrowanie MAC zabezpiecza jedynie przed nieautoryzowanym przyłączeniem do punktu dostępowego, ale po przyłączeniu nie zapewnia poufności samej komunikacji.
- ♦ **SSID** (ang. *Service Set Identifier*). Identyfikator SSID działa jako proste hasło, stanowiąc identyfikator sieci bezprzewodowej. Punkty dostępowe o określonych SSID można skonfigurować w taki sposób, aby nie pozwalały na przyłączanie stacji, które nie znają danego SSID na wstępnym etapie negocjacji połączenia. SSID zabezpiecza wyłącznie przed nieautoryzowanym dostępem, nie zapewnia poufności komunikacji po nawiązaniu połączenia. ESSID (ang. *Extended Service Set Identifier*) to forma SSID, która pozwala na identyfikowanie sieci w ramach wielu różnych punktów dostępowych.
- ♦ **WEP** (ang. *Wired Equivalent Privacy*). Mechanizm bezpieczeństwa wymagający na etapie negocjacji połączenia podania 48, 64 lub 128-bitowego klucza zdefiniowanego w punkcie dostępowym. Ten klucz jest wykorzystywany do szyfrowania i odszyfrowywania komunikacji bezprzewodowej. Jeśli klucz podany przez stację jest zgodny z kluczem zapisanym w punkcie dostępowym, zostaje nawiązane połączenie. WEP służy do zabezpieczania dostępu oraz poufności komunikacji, ponieważ cała wymiana danych jest szyfrowana.
- ♦ **WPA** (ang. *WiFi Protected Access*). Mechanizm bezpieczeństwa wykorzystujący protokół *Temporary Key Integrity Protocol*, zastępujący mechanizm WEP i zapewniający zwiększony poziom bezpieczeństwa. Mechanizm WPA wykorzystuje serwer kluczy lub wstępnie wygenerowany zestaw kluczy do szyfrowania komunikacji na poziomie poszczególnych pakietów. Istnieją dwa różne standardy WPA: WPA1, opracowany w ramach szkicu standardu IEEE 802.11i, oraz WPA2, opracowany w ramach zatwierdzonego standardu 802.11i. Standard WPA nie jest obecnie obsługiwany przez niektóre aplikacje do obsługi sieci bezprzewodowych w Ubuntu, ale należy się spodziewać, że będzie odgrywać ważną rolę w zakresie bezpieczeństwa sieci bezprzewodowych.

Wszystkie te zabezpieczenia można wykorzystywać razem, w dowolnych kombinacjach, lub osobno (jedynie mechanizmy WEP i WPA wykluczają się wzajemnie). Każdorazowe wpisywanie 128-bitowego klucza przy przyłączeniu do sieci bezprzewodowej może być męczące i podatne na błędy. W wielu sieciach do zabezpieczenia dostępu wykorzystuje

się zatem filtrowanie MAC oraz SSID/ESSID, po czym sama komunikacja jest zabezpieczana z użyciem standardowych tuneli SSH lub technologii VPN, działających niezależnie od warstwy sieci bezprzewodowych.

Jądro Linuksa zawiera sterowniki dla wielu urządzeń sieci bezprzewodowych w standardach PCI, PCMCIA i USB. W Ubuntu bezprzewodowe urządzenia Ethernet są konfigurowane w ten sam sposób, co przewodowe sieci Ethernet, co omówię w kolejnym podrozdziale.

Jeśli okaże się, że dla posiadanego interfejsu sieci bezprzewodowej nie powstał jeszcze sterownik dla Linuksa, nie wszystko jest stracone. Dzięki sprytności społeczności open source powstał moduł jądra Linuksa oraz odpowiednie narzędzia pozwalające wykorzystać sterowniki napisane dla systemu Microsoft Windows. Opis instalacji tych narzędzi oraz sterowników dla Windows pod Linuxem znajduje się w dalszej części rozdziału.

Dość powszechne jest obecnie wykorzystanie systemów wyposażonych w urządzenia sieci przewodowych i bezprzewodowych. Mimo tego, że nowoczesne specyfikacje sieci bezprzewodowych, takie jak 802.11g oraz powstająca specyfikacja 802.11n, zapewniają wysokie prędkości komunikacji, sieci bezprzewodowe nadal nie są w stanie dorównać połączeniom 100 Mbit/s a przede wszystkim sieciom gigabitowym. Z tego powodu szkoda rezygnować z wykorzystania sieci przewodowej, gdy jest dostępna. Za pomocą narzędzia *System/Administracja/Sieć* można tego dokonać ręcznie, ale istnieje również automatyczny mechanizm *Network Manager*, omówiony w ramce w dalszej części niniejszego rozdziału.

Konfiguracja interfejsów sieci bezprzewodowej

Jak wspomniałem wcześniej, interfejsy sieci bezprzewodowej konfiguruje się w Ubuntu w sposób zbliżony do konfiguracji standardowych sieci przewodowych, z tą różnicą, że wymagają podania dodatkowych informacji specyficznych dla sieci bezprzewodowych. Aby skonfigurować interfejs sieci bezprzewodowych w Ubuntu, należy podłączyć go do systemu i wywołać z menu funkcję *System/Administracja/Sieć*, co spowoduje uruchomienie aplikacji do zarządzania siecią (z wiersza poleceń tę samą aplikację można wywołać poleceniem `gksudo network-admin`), przedstawionej na rysunku 25.1.

Po uruchomieniu tej aplikacji urządzenie sieci bezprzewodowej powinno być już uwzględnione, tak jak na rysunku 25.1, ale będzie też informacja o tym, że nie jest skonfigurowane. Jeśli na liście nie pojawi się urządzenie sieci bezprzewodowej, może to oznaczać, że nie jest ono obsługiwane przez jądro Linuksa. W tym przypadku należy zastosować się do porad z podrozdziału „Instalacja i wykorzystanie sterowników dla systemu Windows”, w dalszej części rozdziału.

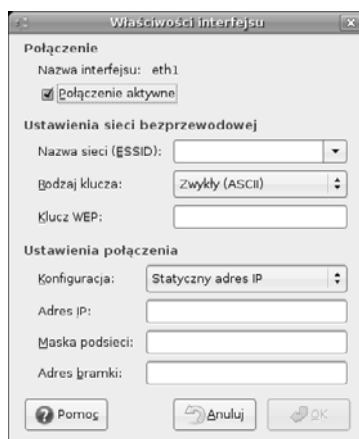
Jeśli interfejs sieci bezprzewodowej pojawi się w oknie przedstawionym na rysunku 25.1, należy zaznaczyć odpowiednią pozycję i kliknąć przycisk *Właściwości*, co spowoduje wyświetlenie okna konfiguracji urządzenia sieci bezprzewodowej. Należy zaznaczyć opcję *Połączenie aktywne*, co uaktywni opcje konfiguracji poszczególnych elementów konfiguracji interfejsu (patrz rysunek 25.2).

Rysunek 25.1.

Konfiguracja
interfejsów sieciowych
w Ubuntu

**Rysunek 25.2.**

Konfiguracja
właściwości interfejsu
sieci bezprzewodowej



Jeśli chcemy przyłączyć się do określonej sieci, w tym miejscu należy wpisać ESSID tej sieci, pozostawienie pustego pola pozwoli przyłączyć się do dowolnych publicznych (otwartych) punktów dostępowych niewymagających uwierzytelniania, niezależnie od nazwy. Jeśli sieć bezprzewodowa, do której chcemy się podpiąć, wykorzystuje klucze WEP, należy wybrać odpowiedni typ klucza z listy rozwijanej *Typ klucza*, a w polu *Klucz WEP* wpisać odpowiednią wartość klucza. Jeśli sieć bezprzewodowa nie jest szyfrowana (nie wykorzystuje klucza WEP), wartość listy *Typ klucza* należy pozostawić na *Plain (ASCII)* i zostawić puste pole *Klucz WEP*. Na końcu konfiguracji należy wprowadzić odpowiednie informacje na temat sieci, jeśli chcemy używać statycznego adresu IP dla połączenia do sieci bezprzewodowej. Jeśli posiadana sieć bezprzewodowa wykorzystuje DHCP, trzeba wybrać opcję *DHCP* z listy rozwijanej *Konfiguracja*.

Po zakończeniu wprowadzania ustawień należy kliknąć *OK*, aby potwierdzić zmiany. Okno *Właściwości* zostanie zamknięte i pojawi się okienko z informacją o stanie konfiguracji interfejsu sieci bezprzewodowej. Po zakończeniu konfiguracji urządzenia ponownie pojawi się okno przedstawione na rysunku 25.1, zawierające informację o tym, że interfejs jest skonfigurowany prawidłowo, co przedstawiam na rysunku 25.3.

Jeśli system jest wyposażony w urządzenia sieci przewodowej oraz bezprzewodowej i mają one być używane równolegle, należy odpowiednio wybrać urządzenie domyślnej bramy.

Rysunek 25.3.

Zaktualizowana
informacja
o urządzeniu sieci
bezprowadowej



Służy do tego lista rozwijana *Urządzenie domyślnej bramki* u dołu okna (patrz rysunek 25.3). W większości przypadków należy tu wybrać urządzenie sieci przewodowej. W zależności od konfiguracji można również dezaktywować inne urządzenia sieci Ethernet. W tym celu należy zaznaczyć po kolei każde z nich, kliknąć przycisk *Właściwości*, po czym zdjąć zaznaczenie z opcji *Połączenie aktywne*. Więcej informacji na temat konfiguracji sieci, bram sieciowych, routingu itp. można znaleźć w rozdziale 24., „Konfiguracja sieci i bezpieczeństwa”.

Aby zakończyć proces konfiguracji sieci bezprzewodowych, należy kliknąć przycisk *OK*, co spowoduje zapisanie zmian. Kliknięcie przycisku *Anuluj* sprawi, że okno zamknie się bez zapisu modyfikacji.

Aby zweryfikować, czy urządzenie zostało właściwie skonfigurowane, można wykorzystać polecenie wywoływane z wiersza poleceń `ifconfig`, które wypisze informacje o stanie interfejsów sieciowych, np.:

```
$ ifconfig wlan0
wlan0    Link encap:Ethernet  HWaddr 00:06:25:07:F7:0E
         inet addr:192.168.6.81  Bcast:192.168.6.255  Mask:255.255.255.0
         inet6 addr: fe80::206:25ff:fe07:f70e/64  Scope:Link
         UP BROADCAST RUNNING MTU:1500  Metric:1
         RX packets:44 errors:0 dropped:0 overruns:0 frame:0
         TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:7697 (7.5 KiB)  TX bytes:1284 (1.2 KiB)
```

Okno dialogowe przedstawione na rysunku 23.3 zawiera informację o nazwie urządzenia sieci bezprzewodowej. Zewnętrzne urządzenia sieci bezprzewodowej (np. podłączone do gniazda USB lub PCMCIA) otrzymują nazwy rozpoczynające się przedrostkiem *wlan* (wireless LAN) oraz identyfikatorem liczbowym służącym do ich rozróżniania, podobnie jak ma to miejsce w przypadku interfejsów sieci Ethernet (przedrostek *eth*). Decyzję o tym, jaki dane urządzenie ma mieć przedrostek (*wlan* czy też *eth*), podejmuje sterownik urządzenia.

Dodatkowe informacje o narzędziu do zarządzania sieciami i o konfiguracji sieci można znaleźć w rozdziale 24., „Konfiguracja sieci i bezpieczeństwa”.

Narzędzia do konfiguracji sieci bezprzewodowych obsługiwane z wiersza poleceń

Pakiet *wireless-tools* instalowany domyślnie w Ubuntu zawiera kilka poleceń do wyszukiwania i konfigurowania interfejsów sieci bezprzewodowych. W podręczniku systemowym tych narzędzi (`man wireless`) można znaleźć informacje na temat pakietu *wireless-tools* i dostępnych w nim narzędzi.

Najczęściej używanymi przeze mnie poleceniami z pakietu *wireless-tools* są `iwconfig` i `iwlist`. Polecenie `iwconfig` wypisuje informacje zbliżone do wypisywanych przez polecenie `ifconfig`. Można dzięki niemu uzyskać informacje o statusie urządzeń sieci bezprzewodowych w systemie. Poniższy przykład zawiera wynik polecenia `iwconfig`, sprawdzającego stan bezprzewodowych urządzeń sieci Ethernet zainstalowanych w poprzednim podrozdziale:

```
$ iwconfig wlan0
wlan0      IEEE 802.11-DS  ESSID:"wvh"  Nickname:"okuwlan"
           Mode:Managed  Frequency:2.427 GHz  Access Point: 00:03:93:E9:89:22
           Bit Rate:11 Mb/s   Tx-Power=15 dBm
           Retry limit:8      RTS thr=1536 B   Fragment thr=1536 B
           Power Management:off
           Link Quality=0/0  Signal level=95/255  Noise level=0/0
           Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
           Tx excessive retries:0  Invalid misc:0  Missed beacon:0
```

Jak widać, informacje zwracane przez polecenie `iwconfig` są zbliżone do informacji uzyskiwanych z polecenia `ifconfig`, ale mamy tu dodatkowo informacje specyficzne dla sieci bezprzewodowych. Polecenie `iwconfig` może być użyte do ustawiania konfiguracji sieci bezprzewodowych z wiersza poleceń; więcej informacji na ten temat można uzyskać z podręcznika systemowego (`man iwconfig`).

Polecenie `iwlist` pozwala skanować eter w poszukiwaniu sieci bezprzewodowych i sprawdzać różne charakterystyki interfejsu sieciowego. Przykładowo w wyniku skanowania sieci bezprzewodowych w mojej okolicy za pomocą polecenia `iwlist wlan0` otrzymałem następujący wynik:

```
$ sudo iwlist wlan0 scan
wlan0      Scan completed :
           Cell 01 - Address: 00:03:93:E9:89:22
                   ESSID:"wvh"
                   Mode:Master
                   Channel:4
                   Encryption key:off
                   Quality:0/0  Signal level:38/255  Noise level:0/0
                   Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s
```

To polecenie jest użyteczne do weryfikacji tego, czy interfejs bezprzewodowy rzeczywiście działa i jest w stanie połączyć się z punktem dostępowym. Informacje zwracane przez

polecenie `iwlist` są bardziej interesujące w publicznych punktach dostępowych, gdzie z reguły można wykryć kilka sieci, jak miało to miejsce w hotelu podczas wakacji:

```
$ sudo iwlist eth0 scan
Password:
eth0      Scan completed :
          Cell 01 - Address: 00:09:5B:36:22:38
                      ESSID:"Wireless"
                      Mode:Master
                      Frequency:2.412 GHz (Channel 1)
                      Signal level:20/153  Noise level:12/153
                      Encryption key:off
                      Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s
          Cell 02 - Address: 00:12:17:A6:E8:CC
                      ESSID:"linksys"
                      Mode:Master
                      Frequency:2.437 GHz (Channel 6)
                      Signal level:83/153  Noise level:13/153
                      Encryption key:off
                      Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s
          Cell 03 - Address: 00:12:17:A6:E8:CC
                      ESSID:"linksys"
                      Mode:Master
                      Frequency:2.437 GHz (Channel 6)
                      Signal level:81/153  Noise level:13/153
                      Encryption key:off
                      Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s
          Cell 04 - Address: 00:12:17:79:CF:D3
                      ESSID:"linksysR79cfd3"
                      Mode:Master
                      Frequency:2.437 GHz (Channel 6)
                      Signal level:49/153  Noise level:13/153
                      Encryption key:off
                      Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 18 Mb/s
                                24 Mb/s; 36 Mb/s; 54 Mb/s
```

W tym przypadku informacje uzyskane za pomocą polecenia `iwlist` były bardzo użyteczne do identyfikacji publicznych punktów dostępowych, ich nazw oraz informacji na temat tego, czy wykorzystują mechanizmy uwierzytelniania i szyfrowania. Polecenie `iwlist` pozwala również sprawdzić szczegóły konfiguracji sieci bezprzewodowych; więcej informacji na ten temat można znaleźć w podręczniku systemowym (`man iwlist`).

Instalacja i wykorzystanie sterowników dla systemu Windows

Obsługa sprzętu to jeden z obszarów, w których Linux ustępuje systemowi Microsoft Windows. Mam nadzieję, że to moje stwierdzenie nie spowoduje telefonu od Linusa Torvaldsa z żądaniem zwrotu mojej odznaki wielbiciela Linuksa. Niemniej jednak taka jest prawda. Powód braku sterowników do niektórych urządzeń nie wynika z jakiegokolwiek ułomności Linuksa, ale przede wszystkim z tego, że producenci sprzętu analizują rynek i decydują, iż z ekonomicznych względów sterowniki opłaca im się pisać tylko dla najszerszego spektrum odbiorców, a do tej pory są nim użytkownicy systemów z rodziny Microsoft Windows.

Sprytni programiści ze środowiska open source piszą sterowniki dla nowych urządzeń, a każde nowe wydanie jądra zawiera znacznie więcej sterowników urządzeń niż poprzednie. Jednak, mimo że sytuacja nie jest zła, użytkownik zdecydowany na wykorzystanie konkretnego urządzenia (niewystępującego na liście oficjalnie obsługiwanych w Linuksie) może nie mieć szczęścia. Z pomocą przychodzi pewien moduł jądra Linuksa oraz związane z nim oprogramowanie, które są bardzo pomocne użytkownikom urządzeń sieci bezprzewodowych, ponieważ umożliwiają wykorzystanie sterowników napisanych z myślą o systemie Microsoft Windows. Dzięki pakietowi *ndiswrapper* można więc wykorzystywać w Ubuntu sterowniki systemu MS Windows do czasu, gdy powstaną dedykowane specjalnie dla Linuksa.

Oczywiście, natychmiast nasuwa się pytanie, dlaczego ktoś, kupując sprzęt, który ma pracować pod Linuxem, nie kupuje urządzeń, dla których już istnieją odpowiednie sterowniki dla tego systemu. Sam właśnie w ten sposób postępuję, ale istnieją powody, dla których ktoś może mieć ochotę (lub potrzebę) wykorzystania sprzętu działającego w systemie Windows, dla którego w Linuksie nie ma do tej pory zaimplementowanej obsługi. Do takich powodów należą przede wszystkim następujące przypadki.

- ♦ Użytkownik posiada już urządzenie sieciowe, nieobsługiwane w Linuksie. Po co kupować nowe urządzenie, jeśli już się jedno posiada?
- ♦ Laptopy wykorzystywane do uruchamiania Windows i Linuksa z wbudowanym interfejsem sieci bezprzewodowej, nieobsługiwanym przez Linuksa. Podobnie w przypadku, gdy użytkownik otrzymał laptop służbowy razem z kartą sieci bezprzewodowej ze złączem PCMCIA lub USB nieobsługiwaną w Linuksie, użycie sterowników dla systemu Windows pod Ubuntu pozwala wykorzystać ten sprzęt pod kontrolą właściwego systemu operacyjnego, bez konieczności wydawania dodatkowych pieniędzy.
- ♦ Niektóre protokoły sieci bezprzewodowych, przede wszystkim 802.11a, nie są zbyt dobrze obsługiwane w Linuksie, a nadal spotyka się je w niektórych korporacjach. Ponieważ standard 802.11a jest już na wymarciu, wątpię, że ta sytuacja ma szansę się zmienić. Wykorzystując sterowniki dla systemu Windows dla standardu 802.11a, pozwolimy wydłużyć życie urządzeniom w standardzie 802.11a również w systemie Linux.

W kilku kolejnych punktach omówię wygodne narzędzie wyposażone w graficzny interfejs użytkownika oraz sposób instalacji w Ubuntu sterowników napisanych dla Windows. Przekazę też kilka porad i sztuczek przydatnych podczas rozwiązywania problemów w systemie Ubuntu ze sterownikami dla systemu Windows.



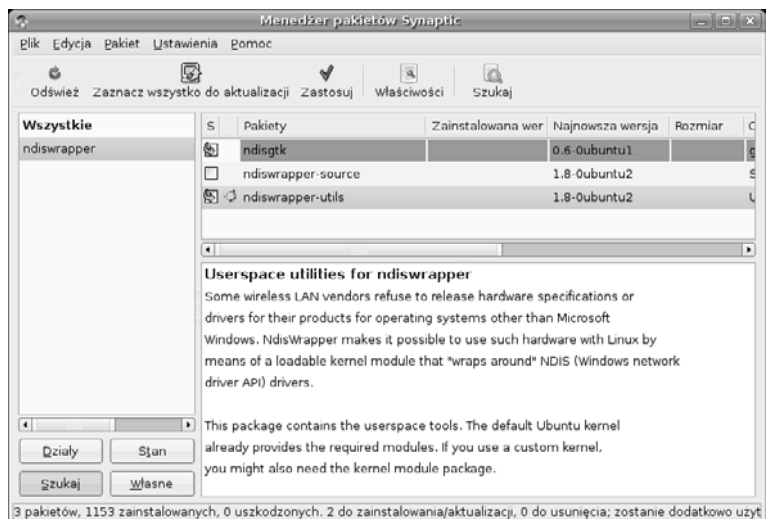
Choć pakiet *ndiswrapper* został napisany z myślą o wykorzystaniu sterowników systemu Microsoft Windows do interfejsów sieci bezprzewodowych, istnieją znane przypadki skutecznego zastosowania tego narzędzia do instalacji sterowników urządzeń do obsługi portów szeregowych USB, przewodowych kart sieciowych Ethernet i wielu innych (kompletną listę urządzeń, które udało się nakłonić do pracy pod Linuxem z użyciem sterowników Windows za pomocą pakietu *ndiswrapper* można znaleźć pod adresem <http://ndiswrapper.sourceforge.net/mediawiki/index.php/List>). Dla posiadaczy urządzeń wyposażonych wyłącznie w sterowniki dla systemu Windows, których nie udało uruchomić się przy pomocy sterowników linuxowych, warty uwagi jest pakiet *ndiswrapper*, od którego dzieli ich zaledwie kilka kliknięć w menedżerze pakietów Synaptic.

Instalacja pakietu NDIS Wrapper i narzędzi pomocniczych

Moduł jądra Linuksa `ndiswrapper` jest standardowo wkompiłowany w każde jądro udostępniane w ramach dystrybucji Ubuntu, ale w celu jego wykorzystania należy zainstalować odpowiednie narzędzia upraszczające instalację i konfigurację sterowników dla systemu Microsoft Windows. Pakiet `ndisgtk` zawiera narzędzie z graficznym interfejsem użytkownika służące do instalacji i konfiguracji sterowników, pakiet `ndiswrapper-utils` zawiera programy wykorzystywane przez to graficzne narzędzie oraz narzędzie obsługiwane z wiersza poleceń dla tych użytkowników, którzy nie są zwolennikami interfejsów graficznych. Narzędzia te nie są zainstalowane domyślnie w Ubuntu, ponieważ nie każdy ich potrzebuje, a po co marnować miejsce na dysku?

Aby zainstalować te pakiety, należy uruchomić *Menedżer pakietów Synaptic* z menu *System/Administracja*, kliknąć przycisk *Szukaj*, w polu *Szukaj* wpisać `ndiswrapper`, z listy rozwijanej *Sprawdzaj w:* wybrać opcję *Opis i Nazwa*, po czym kliknąć przycisk *Szukaj*. Po zakończeniu wyszukiwania trzeba zaznaczyć do instalacji pakiety `ndisgtk` i `ndiswrapper-utils`, czyli kliknąć prawym przyciskiem myszy nazwę pakietu i z menu podręcznego wybrać opcję *Zaznacz do instalacji*. Na rysunku 25.4 przedstawiam wspomniane pakiety zaznaczone do instalacji.

Rysunek 25.4.
*Instalacja
pakietów ndisgtk
i ndiswrapper-utils*



Po zaznaczeniu pakietów należy kliknąć ikonę *Zastosuj* w pasku narzędzi programu Synaptic, co rozpocznie procedurę instalacji pakietów w systemie. Po zakończeniu instalacji system jest gotowy do zainstalowania, skonfigurowania i wykorzystania sterowników dla systemu Windows dla sprzętu w teorii nieobsługiwanego w Linuksie.

Więcej informacji na temat instalowania pakietów za pomocą programu Synaptic można znaleźć w rozdziale 20., „Dodawanie, usuwanie i aktualizacja oprogramowania”.

Instalacja sterowników systemu Microsoft Windows

W tym punkcie omówię sposób instalowania sterowników systemu Microsoft Windows z użyciem programu *ndisgtk*, który stanowi wygodne narzędzie wspomagające pracę z pakietem narzędzi *ndiswrapper*, modulem jądra oraz mechanizmem konfiguracji sieci w dystrybucji Ubuntu. Sterowniki dla systemu Windows do współpracy z modulem *ndiswrapper* można również zainstalować za pomocą komendy wiersza poleceń *ndiswrapper -i nazwa_pliku_INF* i skonfigurować, odpowiednio modyfikując pliki konfiguracyjne. Osobiście jednak preferuję narzędzie z interfejsem graficznym, czyli *ndisgtk*.



Przed rozpoczęciem instalacji sterowników systemu Windows pod Ubuntu należy upewnić się, że są dostępne. Najczęściej wystarczy włożyć do napędu nośnik z odpowiednimi sterownikami lub skopiować plik ZIP zawierający sterowniki na partycję Ubuntu i rozpakować archiwum (za pomocą polecenia *unzip*).

Po zainstalowaniu pakietu *ndisgtk* w menu systemowym tworzy się nowa pozycja *System/Administracja/Windows Wireless Drivers*. Po wybraniu tej pozycji otwiera się program *ndisgtk*. Na rysunku 25.5 przedstawiam okno programu w takiej postaci, jaka pojawia się zaraz po uruchomieniu.

Rysunek 25.5.

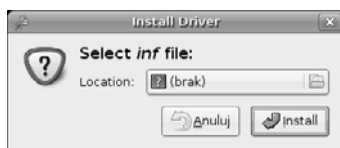
Początkowe okno programu *ndisgtk*



Aby zainstalować sterownik, należy kliknąć przycisk *Install New Driver*. To spowoduje wyświetlenie okna służącego do wskazania pliku *INF* związanego ze sterownikiem urządzenia dla systemu Windows, co przedstawiam na rysunku 25.6. Plik *INF* to tekstowy plik o rozszerzeniu *.inf* zawierający informacje na temat sterownika, jego opcje instalacyjne itp.

Rysunek 25.6.

Okno wyboru pliku *INF* z definicją sterownika dla systemu Microsoft Windows

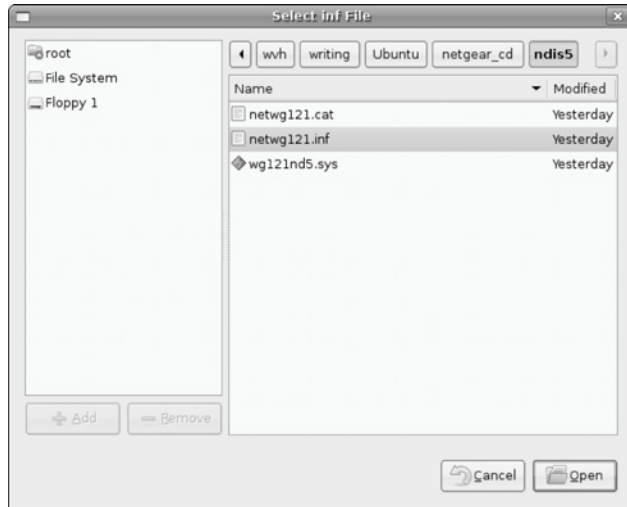


Po kliknięciu przycisku *Location* pojawi się standardowe okno dialogowe służące do wyboru pliku, za którego pomocą należy odszukać w systemie plików odpowiedni plik *INF* instalowanego sterownika (patrz rysunek 25.7). W tym konkretnym przypadku instalowałem sterownik dla systemu Windows urządzenia NetGear WG121 przyłączanego do portu USB.

Po odnalezieniu i wybraniu odpowiedniego pliku należy kliknąć przycisk *Otwórz*, co zamknie okno i otworzy wskazany plik. Ponownie pojawi się okno przedstawione na rysunku 25.6, tym razem z nazwą zaznaczonego pliku *INF*. Po kliknięciu przycisku *Install* sterownik

Rysunek 25.7.

Przeglądanie systemu plików w poszukiwaniu pliku INF sterownika dla systemu Microsoft Windows



zostanie zainstalowany i zarejestrowany do użycia przez moduł jądra `ndiswrapper`. Ponownie pojawi się okno przedstawione na rysunku 25.5, tym razem z informacją o zainstalowanym sterowniku oraz informacją, czy w systemie został odnaleziony sprzęt odpowiadający sterownikom, co demonstruję na rysunku 25.8.

Rysunek 25.8.

Ładowanie sterownika i detekcja sprzętu



Na zakończenie konfiguracji nowego sterownika należy kliknąć przycisk *Configure Network*, a wyświetli się okno dialogowe konfiguracji sieci omówione w podrozdziale „Konfiguracja interfejsów sieci bezprzewodowej”. Interfejs sieci bezprzewodowej należy skonfigurować zgodnie z zawartymi tam instrukcjami.

Po ustawieniu parametrów interfejsu należy kliknąć *OK*, zamykając okno konfiguracji sieci, a następnie kliknąć przycisk *Close* kończący proces konfiguracji sterownika sieci bezprzewodowej.

Aby zapewnić, że moduł `ndiswrapper` będzie uruchamiał się za każdym razem, gdy uruchamiany jest system Windows, należy dopisać odpowiednie ustawienia w pliku `/etc/modules`, definiującym moduły ładowane przez jądro na starcie systemu. Można tego dokonać za pomocą edytora tekstu (który należy uruchomić z prawami administratora) lub przy użyciu prostego polecenia wywołanego z wiersza poleceń:

```
$ sudo echo "ndiswrapper" >> /etc/modules
```

Należy zwrócić uwagę na **dwa** nawiasy trójkątne (>>)! Jeśli omyłkowo wpisze się jeden taki znak, zawartość pliku `/etc/modules` zostanie zastąpiona tekstem, który mieliśmy dopisać na jego końcu (`ndiswrapper`), co prawie na pewno spowoduje problemy z działaniem urządzeń, takich jak np. mysz lub drukarka.

Sztuczki i kruczki związane z mechanizmem NDIS Wrapper

Jak sugerowałem w poprzednim punkcie, sterowniki systemu Windows dla interfejsów sieci bezprzewodowych „po prostu działają” w Linuksie, a to dzięki modułowi jądra `ndiswrapper`, wystarczy prawidłowo je zainstalować. Niestety, to nie do końca prawda. Nie będę twierdził, że wypróbowałem wszystkie dostępne na świecie urządzenia sieci bezprzewodowych i związane z nimi sterowniki dla systemu Windows (ale chętnie je wypróbuję, jeśli ktoś zechce podarować mi na zawsze odpowiednie próbki sprzętu wraz ze sterownikami). Jednak używałem naprawdę wielu urządzeń, konfigurując laptopy dla klientów, współpracowników i przyjaciół.

Najwięcej stresów przeżyłem z różnymi kartami PCMCIA marki LinkSys WPC54G, z których wszystkie udało mi się zmusić do pracy, ale wymagały sporej dawki gimnastyki, drapania w głowę, wyszukiwania w Googlach i godzin konsultacji ze znajomymi specjalistami od sieci w Linuksie (początkowo chciałem zatytułować ten punkt „LinkSys WPC54G, czyli jak spędziłem wakacje”, ale zdecydowałem, że lepiej będzie napisać ten fragment rozdziału w sposób bardziej ogólny). Przy okazji, to są doskonałe urządzenia, obsługujące standard 802.11g (a przez to również i 802.11b) i ponieważ uważam sprzęt firmy LinkSys za niedrogi i godny zaufania, z reguły polecam te karty znajomym. Niestety, **wielki** problem z tymi kartami polega na tym, że LinkSys przez okres ich produkcji wykorzystywał w nich różne komponenty, a co za tym idzie oferował różne sterowniki, z których każdy wymaga nieco innej konfiguracji i miewa własne, unikalne humory przy działaniu za pośrednictwem modułu `ndiswrapper`.

Kluczowym zagadnieniem jest upewnienie się, że instalowany sterownik systemu Windows jest dokładnie tym, właściwym dla posiadanego urządzenia. Przykładowo niektóre z urządzeń 54G wykorzystują chipset Realtek 8180, a przez to będą działać wyłącznie ze sterownikami dla tego chipsetu. Inne stosują chipset Texas Instruments ACX 1xx, a jeszcze inne Broadcom BCM43xx, a nowsze modele z kolei podobno wykorzystują chipset Marvell. Jeśli ktoś nie jest w stanie zidentyfikować chipsetu w oparciu o literaturę dla produktu, z reguły powinien poradzić sobie za pomocą polecenia `lspci`, które wypisuje wszystkie urządzenia PCI znajdujące się w systemie. W wyniku działania tego polecenia należy szukać informacji o urządzeniach sieci bezprzewodowych.

Przykładem dziwaczności kombinacji sterowników i chipsetów jest to, że chipset Texas Instruments ACX 1xx jest obsługiwany bezpośrednio przez jądro Linuksa w postaci modułów `acx` i `acx_pci` (w tym przypadku nie jest potrzebny moduł `ndiswrapper`, choć sam preferuję tutaj zastosowanie tego modułu), ale wymaga eksperymentów z mikrokodem zapisanym w pliku `/lib/firmware/wersja-jądra/acx` w celu odszukania odpowiedniej wersji i nazwy tego firmware (cenne sugestie można znaleźć na forach Ubuntu). Aby użyć modułu `ndiswrapper` do obsługi tego chipsetu, należy zablokować wykorzystanie z nim standardowego modułu jądra (dopisać nazwy tych modułów w pliku `/etc/modprobe.d/blacklist`, aby nie były ładowane na etapie detekcji sprzętu).

Gdy jesteśmy pewni, że posiadamy właściwy sterownik, ale nadal nie udaje się zmusić go do pracy, warto zapoznać się z wynikami działania programu `dmesg`, który wyświetla ostatnie komunikaty jądra systemu. W komunikatach należy odszukać te, które dotyczą modułu `ndiswrapper`, wykorzystywanej karty sieci bezprzewodowej lub chipsetu. To pomoże zidentyfikować przyczynę problemów.

Karty WPC54G wykorzystujące chipset Broadcom BCM43xx również charakteryzowały się dość interesującą cechą: wymagały specyficznego mikrokodu firmware zaszytego wewnątrz pliku sterownika (`bcmwl5.sys`), który do działania musiał być zapisany osobno w katalogu `/lib/firmware`, w jakim jądro Linuksa poszukuje mikrokodów firmware. Aby z pliku sterownika odczytać sam mikrokod, użytkownik musiał posłużyć się programem `bcm43xx-fwcutter`, zapisać uzyskany w ten sposób mikrokod w katalogu `/lib/firmware` i dopiero wówczas mógł prawidłowo skonfigurować tę kartę sieci bezprzewodowej. Posiadacze tej wersji chipsetu najnowszą wersję programu służącego do wydobywania firmware z pliku sterownika znajdą na stronie http://developer.berlios.de/project/showfiles.php?group_id=4547.

Jak wspominałem wcześniej, to są jedynie przykłady problemów, jakie można napotkać w trakcie prób skonfigurowania interfejsów sieci bezprzewodowych z wykorzystaniem sterowników systemu Windows za pomocą pakietu `ndiswrapper`. Wiele osób sprawia wrażenie niezadowolonych z jakości kodu źródłowego na licencji open source, ale uważam, że jakość sterowników urządzeń dla systemu Windows jest nie mniej podejrzana. Jak mawiał mój dziadek: „Jeden słabo działający sterownik w ręku jest wart więcej niż wiele planowanych projektów sterowników na dachu”. Czy jakoś tak?



Programiści pakietu `ndiswrapper` oferują pomoc online na różnych formach, których listę można znaleźć pod adresem <http://ndiswrapper.sourceforge.net/support.html>. Jeśli komuś nie uda się znaleźć tam rozwiązania swojego problemu i jest absolutnie zdeterminowany do użycia w Linuksie sprzętu sieci bezprzewodowych „tylko dla Windows”, istnieje firma pod nazwą Linuxant (<http://www.linuxant.com>), która oferuje doskonałe oprogramowanie pod nazwą DriverLoader działające na podobnych zasadach, co `ndiswrapper`. Programiści firmy Linuxant przetestowali swój produkt z dużą ilością sprzętu sieci bezprzewodowych. Oferują również sterowniki dla Linuksa do wybranych kart softmodemów (modemów, których obsługa jest zaimplementowana w całości programowo, co od zawsze stanowiło problem dla użytkowników Linuksa). Ze strony firmy można pobrać 30-dniową wersję demo programu DriverLoader i sprawdzić, jak się sprawuje z posiadanym sprzętem. Klienci firmy Linuxant mogą skorzystać ze wsparcia pracowników tej firmy oraz licznej społeczności związanej z jej oprogramowaniem.

Podsumowanie

W tym rozdziale omówiłem jedną z najbardziej obiecujących i przyszłościowych technologii w informatyce, czyli sieci bezprzewodowe. Sieci bezprzewodowe to nie tylko wygoda z powodu mniejszej liczby przewodów, ale przede wszystkim możliwość wykonywania pracy na laptopach, przenoszenia jej z jednego pokoju konferencyjnego do drugiego, z biura do biura, a nawet z jednego miasta do drugiego bez konieczności wpinania komputerów i pracochłonnego konfigurowania oraz synchronizowania każdej sieci z osobna.

Inteligentne zarządzanie interfejsami z użyciem narzędzia Network Manager

Wiele komputerów posiada zarówno łącza przewodowe, jak i bezprzewodowe, wykorzystywane na zmianę w różnych okolicznościach. Można je skonfigurować w taki sposób, żeby działały z różnymi sieciami bez potencjalnych konfliktów; w przypadku gdy oba są aktywne jednocześnie i mają skonfigurowane domyślne bramy, mogą pojawić się problemy z wydajnością pracy internetu. Oczywiście, użytkownik może wyłączyć niepotrzebne w danej chwili łącze internetowe, ale częste wykonywanie tych działań prowadzi do irytacji.

Aplet GNOME Network Manager (*nm-applet*) został zaprojektowany do wykonywania automatycznej rekonfiguracji sieci za każdym razem, kiedy występuje zmiana w połączeniach sieciowych. Gdy np. do systemu przyłączane jest łącze przewodowe, aplet dezaktywuje łącze bezprzewodowe, a gdy łącze przewodowe przestaje być aktywne, aktywuje łącze bezprzewodowe. Narzędzie Network Manager można zainstalować za pomocą menedżera pakietów Synaptic: należy zainstalować pakiety *network-manager* i *network-manager-gnome* i sprawdzić, czy pakiet *wpa_supplicant* jest już zainstalowany w systemie. Aplet *Network Manager* za pomocą programu *wpa_supplicant* obsługuje uwierzytelnianie w standardzie WPA, które jest kolejnym standardem zwiększającym poziom bezpieczeństwa sieci bezprzewodowych.

W teorii *Network Manager* działa zgodnie z oczekiwaniami, zwalniając użytkownika z konieczności dokonywania ręcznych modyfikacji ustawień sieci w pliku */etc/network/interfaces*. Jednak jego dokumentacja jest dość uboga (i w tym określeniu jestem niezwykle subtelny). Zainteresowanym użytkownikom proponuję zapoznać się z forami dyskusyjnymi (<http://www.ubuntuforums.org>), na których można znaleźć sporo informacji na temat apletu *Network Manager*. Z tego, co udało mi się stwierdzić, ten program będzie bardzo przydatny podczas częstego zmieniania sieci (*roaming*) pod warunkiem, że zostanie płynnie zintegrowany z Ubuntu.

Nie udało mi się wykorzystać możliwości aplikacji *Network Manager* do konfiguracji większej liczby interfejsów i z tego powodu jej nie używam. Wiele osób go uwielbia, ja na niego narzekam. Żadnych gwarancji.

W rozdziale 26. rozpoczynam serwerową część tej książki, omawiając konfigurację serwera WWW Apache 2 w systemie Ubuntu. Na początku rozdziału 26. podaję informacje na temat serwerów WWW, następnie omawiam szczegóły instalacji, konfiguracji i monitorowania serwera Apache 2, najnowszego i najlepszego z serii serwerów WWW autorstwa legendarnego projektu Apache.

Część IV

Konfiguracja serwerów w Ubuntu

W tej części:

Rozdział 26. Konfiguracja serwera WWW

Rozdział 27. Konfiguracja serwera poczty

Rozdział 28. Konfiguracja serwera DHCP

Rozdział 29. Konfiguracja serwera DNS

Rozdział 30. Konfiguracja serwera wydruków

Rozdział 31. Konfiguracja serwera NFS

Rozdział 32. Konfiguracja serwera Samba

Rozdział 26.

Konfiguracja serwera WWW

W tym rozdziale:

- ◆ Podstawy sieci WWW
- ◆ Wprowadzenie do serwera Apache
- ◆ Instalacja serwera Apache
- ◆ Konfiguracja serwera Apache
- ◆ Rozwiązywanie problemów z serwerem WWW

Znaczące postępy w informatyce widać najlepiej, kiedy powstaje kluczowa technologia, tzw. *killer app* (w tym przypadku *application* dotyczy nie konkretnej aplikacji, a raczej dziedziny zastosowania komputerów) — unikalna, niezwykle znacząca, o ogromnych możliwościach, powodująca, że użytkownicy nagle masowo zaczynają interesować się nią, i stanowiąca przełom w postrzeganiu jakiejś dziedziny informatyki. Dla komputerów biurowych takim przełomem był arkusz kalkulacyjny. W systemach Apple Macintosh były to aplikacje do składu publikacji, tzw. *desktop publishing*. A w internecie takim przełomem było powstanie WWW. Oczywiście, każdy doceniał wygodę poczty e-mail, ale WWW zmieniło internet w bezkresny ocean handlu e-commerce, informacji o zakresie osobistym i technicznym, sieci społecznych oraz rzeczy, które trudno dziś przewidzieć, ale które z pewnością zaskoczą nas w najbliższej przyszłości.

W rozdziale 9. omówiłem wykorzystanie programu Firefox, najlepszej przeglądarki WWW na świecie. Tu zajmę się drugą stroną medalu, a raczej sieci, czyli konfiguracją serwera WWW w taki sposób, aby dostarczać strony WWW i inne treści internetowe każdemu zainteresowanemu użytkownikowi. Większość środowisk biznesowych i akademickich posiada zarówno publiczne serwery WWW, jak i prywatne, do wewnętrznych zastosowań. Wiele osób konfiguruje serwery WWW w swoich sieciach domowych po to, by umożliwić wykorzystanie wspólnych kalendarzy online, współużytkowanie dokumentów oraz tworzenie centralnego repozytorium zdjęć oraz realizację wszelkich innych zadań, które mogą być przydatne.

Ten rozdział rozpoczynam od podstaw na temat sieci WWW, zasad jej działania. Następnie pokrótce przedstawię historię serwera Apache, najpopularniejszego obecnie serwera WWW. W kolejnych podrozdziałach omówię proces instalacji serwera Apache, jego konfiguracji oraz umieszczania treści internetowych na serwerze w taki sposób, by były dostępne dla użytkowników. Opiszę też dostosowanie serwera Apache do własnych potrzeb oraz wyjaśnię, w jaki sposób rozwiązywać problemy z połączeniami i uprawnieniami.

Podstawy sieci WWW

Istnieje mnóstwo doskonałych książek na temat sieci WWW i języka HTML, a ta nie ma zamiaru ich zastąpić. Jednak wszyscy czytelnicy, dla których zagadnienia związane z serwerami WWW są nowością, w tym podrozdziale mogą znaleźć nieco historii oraz wyjaśnienia pojęć, które będą potem wykorzystywane w omawianiu zagadnień technicznych.

W 1989 roku Tim Berners-Lee, pracownik instytutu CERN (Conseil Européen pour la Recherche Nucleaire, czyli Europejskiego Ośrodka Badań Jądrowych w Genewie, w Szwajcarii) opracował koncepcję, która stała się podstawą sieci WWW. Termin World Wide Web nie istniał aż do roku 1990, gdy Tim Berners-Lee i Robert Cailliau opublikowali oficjalną propozycję projektu opracowania sieci WWW. Zasugerowali nowy sposób wymiany informacji między badaczami instytutu CERN wykorzystującymi różne typy terminali i stacji roboczych. Unikalne cechy proponowanego modelu wymiany polegały na tym, że dane przechowywane na różnych serwerach będą przesyłane do klientów w ujednoliconej formie, niezależnej od urządzenia, a za formatowanie informacji na ekranie (nazwane **renderowaniem**) odpowiedzialny będzie sam klient. Klienci WWW komunikują się z serwerem za pomocą języka (protokołu) nazywanego HTTP, co stanowi skrót od HyperText Transfer Protocol (protokół przesyłania hipertekstu).



Hipertekst to zwykły tekst zawierający osadzone łącza. Przykładem popularnych hipertekstów poza siecią WWW są systemy pomocy online, w których można nawigować między zagadnieniami, klikając wyodrębnione słowa kluczowe. W ramach WWW najpopularniejszą techniką wykorzystującą hipertekst jest HTML, czyli HyperText Markup Language (język znaczników hipertekstowych), który omówię w dalszej części rozdziału.

Uczestnikami sieci WWW są z jednej strony serwery, a z drugiej po prostu przeglądarki WWW, takie jak Firefox, Opera, SeaMonkey, Netscape, Microsoft Internet Explorer czy Safari firmy Apple. Aby pobrać stronę WWW z serwera, należy wprowadzić jej adres w formacie URI (Uniform Resource Identifier) przez bezpośrednie wpisanie w polu adresu lub przez kliknięcie łącza odwołującego się do tego adresu URI. Przeglądarka łączy się z odpowiednim serwerem WWW, który wykorzystuje ten adres URI do zlokalizowania zasobu i zwraca jego zawartość w postaci strumienia informacji hipertekstowych, które przeglądarka renderuje na ekranie. Ot i cała tajemnica.

Współczesne przeglądarki potrafią porozumiewać się innymi, oprócz HTTP, protokołami, takimi jak m.in. FTP (File Transfer Protocol, wykorzystywany do pobierania plików), *file* (do otwierania plików z zasobów lokalnych), POP (Post Office Protocol, wykorzystywany do pobierania poczty elektronicznej) czy NNTP (Network News Transfer

Protocol, wykorzystywany do wymiany wiadomości Usenet News). Protokół zastosowany do wymiany danych jest zakodowany w URI w ramach tzw. schematu (ang. *scheme*). Typowy adres URI składa się z trzech elementów:

schemat://host/ścieżka

Schemat, `http`, `ftp`, `file` lub wiele innych, określa sposób, w jaki przeglądarka łączy się z serwerem, a przez serwer wykorzystywany jest do ustalenia metody obsługi żądania. Ścieżka jest opcjonalną częścią URI i identyfikuje lokalizację na serwerze zasobu, który ma być przesłany do przeglądarki.

URL-e a URI

Jak można zauważyć w innych rozdziałach tej książki, aplikacje wykorzystujące sieć WWW często stosują odmienną terminologię do określania potocznego pojęcia „adresu w sieci WWW”. URL (Uniform Resource Locator) to tradycyjny akronim stosowany w odniesieniu do adresu w sieci WWW, ale bardziej poprawny technicznie jest akronim i termin URI (Uniform Resource Identifier). Jeszcze inny akronim i termin stosowany w odniesieniu do adresów to URN (Universal Resource Name).

Związek między tymi akronimami jest następujący — URI to dowolny sposób identyfikacji zasobu WWW, URL to URI określający lokalizację zasobu i protokół wykorzystywany do jego odczytu, URN to URI definiujący nazwę zasobu i może, ale nie musi, informować o tym, gdzie go można odnaleźć.

Zmierzam do tego, że większość użytkowników wykorzystuje i postrzega terminy URI, URL i „adres w sieci WWW” zupełnie zamiennie. Jeśli jednak ktoś chciałby wybrać jeden, URI jest najodpowiedniejszy w ogólnym zastosowaniu.

Strony WWW składają się ze statycznych i dynamicznie generowanych dokumentów tekstowych, które mogą zawierać tekst, łącza do innych stron lub serwisów WWW, osadzoną grafikę w różnych formatach, odwołania do powiązanych dokumentów, takich jak arkusze kalkulacyjne i wiele innych. Te dokumenty tekstowe są tworzone z użyciem języka znaczników HTML (HyperText Markup Language, języki znaczników zostały omówione w rozdziale 10.). Strukturalny język znaczników jest językiem znaczników ze zdefiniowanymi regułami zachowania hierarchii, tzn. pewne znaczniki mogą występować wyłącznie w określonych kontekstach. Użycie strukturalnego języka znaczników może być wykorzystane do zagwarantowania, że np. nagłówek nie wystąpi w środku akapitu. Podobnie jak dokumenty utworzone w innych nowoczesnych językach znaczników, dokumenty w HTML-u składają się z logicznych elementów identyfikujących typ każdego elementu, a zadaniem przeglądarki jest identyfikacja elementów i określenie sposobu ich wyświetlenia na ekranie (czyli rendering). Wykorzystanie języka znaczników niezależnego od urządzenia upraszcza tworzenie narzędzi wyświetlających strony WWW na różne sposoby (w zależności od typu wyświetlającego je urządzenia) czy konwersje informacji ze stron WWW na inne formaty strukturalne (i vice versa).

Wprowadzenie do serwerów WWW i Apache

Jak wspominałem w poprzednim podrozdziale, kontrapunkt dla przeglądarek WWW stanowią serwery WWW, aplikacje, których zadaniem jest odszukiwanie i dostarczanie treści internetowych wskazanych przez wybrany URI. Jakie zadania wykonuje serwer WWW? W najprostszym wydaniu serwer po prostu dostarcza dokumenty w formacie HTML oraz inne treści, zgodnie z przychodzącymi zadaniami. Aby jednak serwer WWW sprawdzał się w nowoczesnych zastosowaniach związanych z WWW, musi spełniać kilka założeń, z których najważniejsze to:

- ♦ elastyczność i łatwość konfiguracji ułatwiające dodawanie nowych możliwości, wirtualnych hostów i realizowanie rosnących oczekiwań bez konieczności ponownej kompilacji;
- ♦ obsługa mechanizmów uwierzytelniających pozwalających ograniczać grupy użytkowników upoważnionych do uzyskiwania dostępu do określonych stron WWW;
- ♦ obsługa mechanizmów generujących strony WWW w sposób dynamiczny, takich jak Perl czy PHP, pozwalających na dostosowanie treści do potrzeb użytkowników serwisu;
- ♦ zapisywanie dzienników umożliwiających śledzenie żądań do stron WWW, pozwalających na identyfikację problemów i ustalanie popularności stron i serwisów WWW;
- ♦ obsługa szyfrowania wymiany danych między serwerem a przeglądarką, co zapewnia zwiększenie bezpieczeństwa tej komunikacji.

Znaczenie poszczególnych wymogów jest uzależnione od wymagań stawianych serwerowi WWW przez administratora lub handlowca e-commerce, ale wszystkie nowoczesne serwery WWW powinny je spełniać.

Obecnie istnieje wiele różnych serwerów WWW, wybieranych w zależności od platformy sprzętowej, wymagań związanych z oprogramowaniem serwera WWW, skłonności do wybranego systemu operacyjnego lub tendencji do wykorzystywania oprogramowania open source (lub wręcz przeciwnie).

Jak można się spodziewać, pierwszy aktywny serwer WWW powstał w instytucie CERN, obok pierwszej na świecie przeglądarki WWW. Oba zostały napisane na stacjach roboczych NeXT, platformie o niezbyt dużej popularności (co jest smutne). Pierwszy poza Europą serwer WWW został uruchomiony w USA, w Stanford Linear Accelerator Center (SLAC).

Pierwszy serwer WWW na bardziej popularne platformy powstał w NCSA (National Center for Supercomputing Applications), był znany jako NCSA httpd (HTTP Daemon). Prace nad tym serwerem WWW były prowadzone równolegle z pracami nad przeglądarką NCSA, znaną jako Mosaic (więcej informacji na jej temat oraz roli, jaką odegrała, podaję w historii przeglądarek WWW). Gdy jeden z najważniejszych programistów pracujących przy NCSA httpd (Rob McCool) opuścił NCSA, grupa fanów i programistów pracujących przy projekcie NCSA httpd założyła projekt zmierzający do utrzymywania i wspierania rozwoju NCSA httpd. Ten poprawiony serwer z czasem przekształcił się w serwer Apache. Na oficjalnej stronie projektu Apache istnieje formalne wyjaśnienie

pochodzenia nazwy — jego twórcy utrzymują, że nazwa pochodzi od znanego z wytrwałości i wojowniczości szczepu Indian Apaczów. Jednak większość osób (do których sam się zaliczam) uważa, że nazwa jest żartem, bowiem ten serwer początkowo składał się z wielkiego zbioru łat (ang. *patch*), innymi słowy był to „połatany serwer” („a patchy Web server”, co fonetycznie sprowadziło się do *Apache Web server*).

W Ubuntu do instalacji dostępne są dwie różne wersje serwera Apache, w postaci pakietów *apache* i *apache2*. Podstawowe różnice między nimi polegają na bazie kodu źródłowego, wieku oraz sposobie instalacji i zarządzania. Pakiet *apache* zawiera najnowszą wersję serii serwerów WWW Apache 1.x, która w swoim czasie stanowiła doskonałą opcję i nadal jest bardzo popularna, działa w tysiącach instalacji. Pakiet *apache2* zawiera najnowszą wersję serwerów z rodziny Apache 2.x, którą można nazwać „następną generacją serwerów Apache”. Mechanizmy działania serwera Apache 2 znacząco się zmieniły, szczególnie z punktu widzenia administratora, Apache 2.x jest znacznie lepszy od swojego poprzednika, wszystkie najnowsze rozszerzenia serwera Apache są rozwijane już z myślą o tej wersji. W tym rozdziale skupię się zatem na wyjaśnieniu instalacji i konfiguracji serwera Apache 2. Wszelkie odwołania do Apache odnoszą się do wersji 2.x tego serwera.

Obecnie liczba stron serwowanych przez instalację serwera Apache przewyższa liczbę stron WWW obsługiwanych przez inne serwery WWW. Nie pamiętam w tej chwili nazwy drugiego w rankingu serwera WWW, ale jest dostępny tylko na jedną platformę systemową (którą nie jest Linux), przez co ma poważne wady, nie tylko statystyczne, ale i ideologiczne.

Instalacja serwera Apache

W zależności od tego, czy system Ubuntu został zainstalowany z *Ubuntu Server CD*, *Ubuntu Alternate CD* czy też z *Ubuntu Desktop CD*, serwer Apache instaluje się na różne sposoby. Różnice sprowadzają się jednak do tego, czy system posiada interfejs GUI.

- ♦ Jeśli system został zainstalowany z *Ubuntu Server CD* z zastosowaniem opcji *Instalacja na dysku twardym*, system nie będzie posiadał środowiska graficznego, chyba że użytkownik zainstalował je osobno. W takim przypadku serwer Apache najlepiej zainstalować za pomocą programu *aptitude*, co zostało omówione w punkcie „Instalacja serwera Apache z wiersza poleceń”, ponieważ ten program zainstaluje również inne użyteczne pakiety, takie jak np. dokumentacja serwera Apache.
- ♦ Jeśli system został zainstalowany z *Ubuntu Server CD* z zastosowaniem opcji *Instalacja serwera LAMP*, system nie będzie posiadał środowiska graficznego, chyba że użytkownik zainstalował je osobno. Jednak w tej opcji serwer Apache 2 jest instalowany standardowo w ramach platformy LAMP (Linux, Apache, MySQL i Perl). W takim przypadku można pominąć fragment rozdziału omawiający instalację.
- ♦ Podczas instalacji systemu z *Ubuntu Alternate CD* użytkownik ma jeszcze więcej opcji.

- ♦ W przypadku wykorzystania opcji *Instalacja w trybie tekstowym* system jest wyposażony w środowisko graficzne i do instalacji serwera Apache najlepiej użyć programu Synaptic, co zostało omówione w punkcie „Instalacja Apache za pomocą programu Synaptic”.
- ♦ W przypadku wykorzystania opcji *Instalacja w trybie OEM* system jest wyposażony w środowisko graficzne i do instalacji serwera Apache najlepiej użyć programu Synaptic, co zostało omówione w punkcie „Instalacja Apache za pomocą programu Synaptic”.
- ♦ W przypadku wykorzystania opcji *Instalacja serwera* system nie jest wyposażony w środowisko graficzne, chyba że zostało zainstalowane osobno. Do instalacji serwera Apache najlepiej użyć programu *aptitude*, co zostało omówione w punkcie „Instalacja serwera Apache z wiersza poleceń”, ponieważ ten program zainstaluje również inne użyteczne pakiety, takie jak np. dokumentacja serwera Apache.
- ♦ Jeśli system został zainstalowany z *Ubuntu Desktop CD*, jest wyposażony w środowisko graficzne i do instalacji serwera Apache najlepiej użyć programu Synaptic, co zostało omówione w punkcie „Instalacja Apache za pomocą programu Synaptic”.

Instalacja serwera Apache z wiersza poleceń

Serwer WWW Apache można zainstalować z wiersza poleceń, najłatwiej do tego celu użyć programu *apt-get* lub *aptitude*. Polecam użycie *aptitude*, co pozwala wykorzystać jego opcję automatycznej instalacji pakietów zalecanych obok pakietów niezbędnych do uruchamiania i monitorowania pracy serwera Apache w systemie Ubuntu.



Jak wspominałem wcześniej, w Ubuntu dostępne są dwie wersje serwera Apache w postaci dwóch pakietów o odmiennych zależnościach i pakietach zalecanych. W tym punkcie skupię się na instalacji serwera WWW Apache 2. Aby zainstalować starszą wersję Apache 1.3.x, należy uaktywnić repozytorium *universe* i zamiast pakietu *apache2* zainstalować pakiet *apache*. Zalecam używanie serwera Apache 2, chyba że ze względu na kompatybilność ktoś jest zmuszony stosować serwer Apache 1.3.x (np. z powodu bibliotek lub modułów niedostępnych dla Apache 2).

Aby zainstalować serwer Apache 2 z wiersza poleceń, należy wywołać następujące polecenie:

```
$ sudo aptitude -r install apache2
```

Program *sudo* poprosi najpierw o podanie hasła użytkownika, a następnie o potwierdzenie instalacji pakietu *apache2* i pakietów wymaganych oraz zalecanych do instalacji z tym pakietem. Aby zaakceptować instalację, należy nacisnąć klawisz *Enter* (lub *Y* i *Enter*), co spowoduje instalację pakietów, odpowiednią modyfikację sekwencji startowej systemu oraz uruchomienie serwera Apache 2. W tym momencie można zacząć konfigurację serwera i jego treści. Teraz możemy pominąć poniższy fragment rozdziału aż do punktu „Konfiguracja serwera Apache”.

Szczegółowe informacje na temat wykorzystania programu *apt-get* i *aptitude* można znaleźć w rozdziale 20.

Instalacja Apache za pomocą programu Synaptic

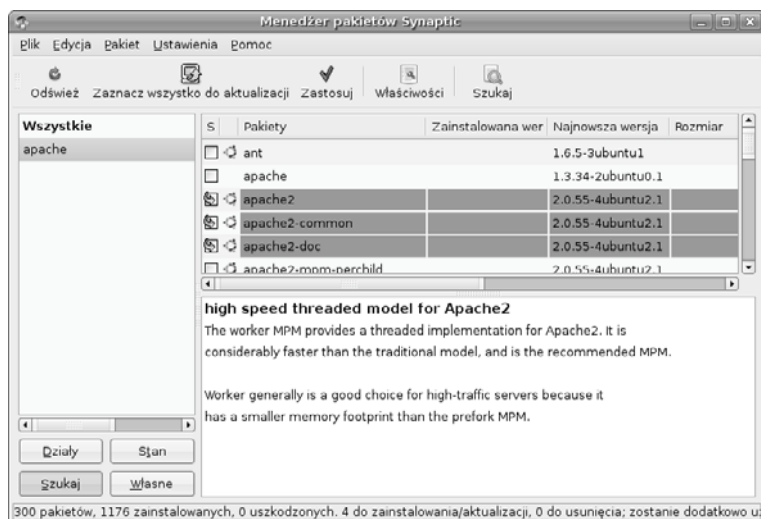
Aby zainstalować pakiety niezbędne do uruchomienia i monitorowania serwera Apache, należy z menu systemowego wywołać *System/Administracja/Menedżer pakietów Synaptic*, kliknąć przycisk *Szukaj*, co spowoduje wyświetlenie okna wyszukiwania. Z listy rozwijanej *Szukaj* w należy zaznaczyć opcję *Nazwy i opisy*, w polu wyszukiwania wpisać *apache* i kliknąć *Szukaj*.



Po zakończeniu wyszukiwania, w zależności od konfiguracji repozytoriów, na liście znalezionych pakietów mogą pojawić się dwa: jeden o nazwie *apache*, drugi o nazwie *apache2*. Pakiet *apache* zawiera najnowszą wersję serwera Apache z gałęzi 1.x. Pakiet *apache2* zawiera najnowszą wersję serwera z rodziny Apache 2.x, którą można nazwać „serwerem następnej generacji”. Choć mechanizmy działania serwera Apache 2 znacząco się zmieniły, szczególnie z punktu widzenia administratora, Apache 2.x jest znacznie lepszy od swojego poprzednika, a wszystkie najnowsze rozszerzenia tworzone są właśnie dla niego. Nakłanianie do zainstalowania innego serwera WWW to iście niedźwiedzia przysługa.

Należy kliknąć prawym przyciskiem myszy pakiet *apache2* i z menu podręcznego wybrać funkcję *Zaznacz do instalacji*. Warto również zaznaczyć pakiet *apache2-doc*, który zawiera całą dokumentację dostępną w ramach projektu Apache 2. Na rysunku 26.1 przedstawiam podstawowe pakiety serwera Apache zaznaczone do instalacji.

Rysunek 26.1. Instalacja serwera Apache 2 i związanych z nim pakietów



Po potwierdzeniu instalacji pojawi się okno, w którym musimy zatwierdzić instalację wymaganych pakietów. W tym oknie należy kliknąć przycisk *Zaznacz*, aby zaakceptować instalację niezbędnych pakietów.

Następnie w pasku narzędziowym programu Synaptic trzeba kliknąć ikonę *Zastosuj*, co rozpocznie instalację serwera Apache 2. Po zakończeniu instalacji serwer Apache 2 zostanie uruchomiony w systemie, choć jego możliwości będą nieco ograniczone domyślną konfiguracją. W kolejnych punktach omawiam podstawy konfiguracji, instalacji stron WWW i zwiększania użyteczności świeżo zainstalowanego serwera Apache 2.

Szczegółowe informacje na temat wykorzystania programu Synaptic można znaleźć w rozdziale 20.

Pliki serwera Apache 2

W tym podrozdziale omówię domyślne lokalizacje plików konfiguracyjnych, binarnych i plików treści serwowanych przez serwer Apache 2 w systemie Ubuntu. Oto one.

- ♦ **/etc/apache2**: katalog zawierający pliki konfiguracyjne dla serwera WWW Apache 2. Najważniejszym plikiem w tym katalogu jest *apache2.conf*.
- ♦ **/etc/apache2/conf.d**: katalog zawierający lokalne dyrektywy konfiguracyjne serwera Apache 2, np. związane z niestandardowymi lub lokalnie zainstalowanymi modułami serwera.
- ♦ **/etc/apache2/envvars**: plik zawierający zmienne środowiska potrzebne do działania skryptu *apache2ctl* zarządzającego serwerem Apache 2.
- ♦ **/etc/apache2/mods-available**: katalog zawierający moduły serwera Apache 2 i ich pliki konfiguracyjne.
- ♦ **/etc/apache2/mods-enabled**: katalog zawierający dowiązania symboliczne uaktywniające moduły serwera Apache 2, zlokalizowane w katalogu */etc/apache2/mods-available*. Zastosowany tu mechanizm jest analogiczny do mechanizmu skryptów rozruchowych systemu na różnych poziomach pracy, czyli stosuje dowiązania do skryptów zapisanych w katalogu */etc/init.d*.
- ♦ **/etc/apache2/sites-available**: katalog zawierający pliki definiujące serwisy WWW obsługiwane przez serwer.
- ♦ **/etc/apache2/sites-enabled**: katalog zawierający dowiązania symboliczne uaktywniające definicje serwisów WWW, zlokalizowanych w katalogu */etc/apache2/sites-available*. Zasada działania jest taka sama jak w przypadku */etc/apache2/mods-enabled*.
- ♦ **/etc/default/apache2**: plik konfiguracyjny określający, czy serwer Apache 2 ma być uruchamiany automatycznie przy starcie systemu.
- ♦ **/etc/init.d/apache2**: skrypt startowy serwera Apache 2 wywołujący program *apache2ctl* służący do uruchamiania i zatrzymywania serwera Apache 2.
- ♦ **/etc/mime.types**: domyślny plik definiujący w jaki sposób dana treść MIME (*Multipurpose Internet Mail Extensions*) ma być obsługiwana przez serwer.
- ♦ **/usr/lib/cgi-bin**: lokalizacja skryptów CGI-BIN (*Common Gateway Interface scripts*) dla serwera Apache 2.
- ♦ **/usr/sbin/apache2**: program wykonawczy serwera Apache 2.
- ♦ **/usr/sbin/apache2ctl**: skrypt administracyjny ułatwiający uruchamianie, zatrzymywanie, ponowne uruchamianie i monitorowanie statusu działającego serwera Apache 2.

- ♦ ***/usr/share/apache2-doc***: katalog zawierający dokumentację serwera Apache 2 (w podkatalogu *manual*). Ten katalog będzie istniał wyłącznie w przypadku zainstalowania pakietu *apache2-doc* (co sugerowałem wcześniej).
- ♦ ***/usr/share/apache2/error***: katalog zawierający definicje domyślnych komunikatów o błędach.
- ♦ ***/usr/share/apache2/icons***: katalog zawierający domyślny zestaw ikon wykorzystywanych przez serwer Apache 2. Zawartość katalogu jest mapowana jako */icons* w głównej konfiguracji twojego serwera Apache.
- ♦ ***/var/log/apache2/access.log***: domyślny plik dziennika dostępu dla serwera Apache 2. Ten dziennik śledzi próby dostępu do stron WWW obsługiwanych przez serwer oraz adresy hostów klienckich.
- ♦ ***/var/log/apache2/error.log***: domyślny plik dziennika błędów serwera Apache 2. Ten plik zawiera komunikaty o wewnętrznych problemach serwera WWW, próby odczytania nieistniejących zasobów itp.
- ♦ ***/var/run/apache2/apache2.pid***: plik tekstowy zawierający numer procesu (PID) serwera Apache 2. Ten plik jest wykorzystywany przez skrypt */etc/init.d/apache2* do zatrzymywania serwera.
- ♦ ***/var/www/apache2-default***: katalog zawierający domyślną stronę domową serwera WWW. Należy zwrócić uwagę, że zawartość tego katalogu nie jest prawidłowo obsługiwana przez serwer Apache 2. W dalszej części rozdziału wyjaśnię, w jaki sposób skonfigurować serwer do jego obsługi.

Niektóre z katalogów, a szczególnie katalog z konfiguracją */etc/apache2*, zawierają pliki, które są włączane przez inne pliki zapisane w tym samym katalogu.

Konfiguracja serwera Apache

Jak wspominałem w poprzednim podrozdziale, pliki konfiguracyjne serwera Apache 2 są zapisane w katalogu */etc/apache2*. Pliki konfiguracyjne stron WWW są zapisane w katalogu */etc/apache2/sites-available*. Aby włączyć obsługę serwisu WWW przez dany serwer, należy w katalogu */etc/apache2/sites-available* utworzyć plik konfiguracyjny dla danego serwisu, po czym w katalogu */etc/apache2/sites-enabled* stworzyć dowiązanie symboliczne do tego pliku.

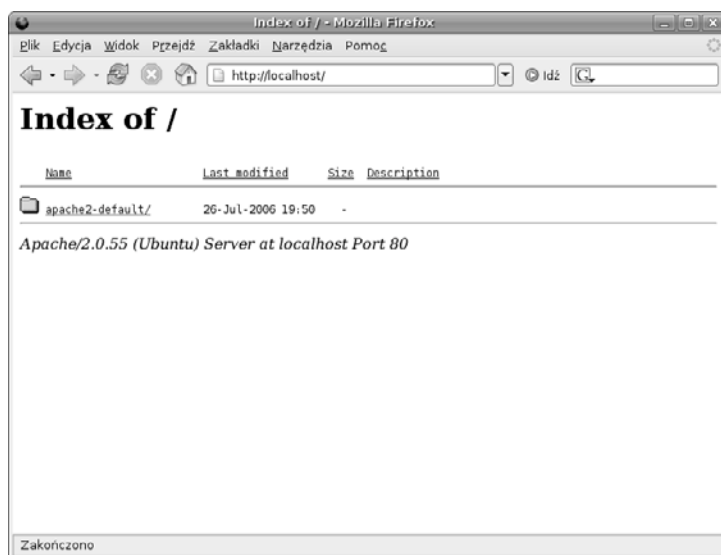
Jedyny serwis skonfigurowany w domyślnej instalacji serwera Apache 2 jest dostępny pod adresem *http://nazwa.hosta*. Niestety, próba wywołania takiego adresu w domyślnej instalacji Apache 2 w Ubuntu spowoduje wyświetlenie strony prezentowanej na rysunku 26.2.



Jeśli planujemy wykorzystać prostą stronę domową serwera WWW, wystarczy skopiować odpowiednie pliki do katalogu */var/www*, co powinno spowodować, że wszystko zadziała bez problemu. Kłopot z domyślną stroną domową serwera Apache 2 w Ubuntu wykorzystam do zademonstrowania kilku dyrektyw konfiguracyjnych.

Rysunek 26.2.

Domyślna strona
domowa świeżo
zainstalowanego
serwera Apache 2



Wykorzystamy tę sytuację do analizy zawartości i składni pliku konfiguracyjnego serwera WWW. Dokonamy kilku modyfikacji, aby zmusić serwer WWW do prezentowania właściwej strony WWW. W poniższym listingu przedstawiam domyślną zawartość pliku */etc/apache2/sites-available/default*, do którego utworzone zostało dowiązanie symboliczne */etc/apache2/sites-available/000-default* aktywujące ten serwis na serwerze (liczby na początku wierszy nie występują w pliku, stanowią jedynie numerację dla uproszczenia analizy, nie należy ich przepisywać!).

```

1. NameVirtualHost *
2. <VirtualHost *>
3.     ServerAdmin webmaster@localhost
4.     DocumentRoot /var/www
5.     <Directory />
6.         Options FollowSymLinks
7.         AllowOverride None
8.     </Directory>
9.     <Directory /var/www/>
10.        Options Indexes FollowSymLinks MultiViews
11.        AllowOverride None
12.        Order allow,deny
13.        allow from all
14.        # Uncomment this directive is you want to see apache2's
15.        # default start page (in /apache2-default) when you go to /
16.        #RedirectMatch ^/$ /apache2-default/
17.    </Directory>
18.    ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
19.    <Directory "/usr/lib/cgi-bin">
20.        AllowOverride None
21.        Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
22.        Order allow,deny
23.        Allow from all
24.    </Directory>
25.    ErrorLog /var/log/apache2/error.log
26.    # Possible values include: debug, info, notice, warn, error, crit,
```

```
27. #alert, emerg.  
28. LogLevel warn  
29. CustomLog /var/log/apache2/access.log combined  
30. ServerSignature On  
31. Alias /doc/ "/usr/share/doc/"  
32. <Directory "/usr/share/doc/">  
33.     Options Indexes MultiViews FollowSymLinks  
34.     AllowOverride None  
35.     Order deny,allow  
36.     Deny from all  
37.     Allow from 127.0.0.0/255.0.0.0 ::1/128  
38. </Directory>  
39. </VirtualHost>
```

W pierwszej kolejności warto zwrócić uwagę na wiersz 3., w którym jest skonfigurowany adres e-mail do webmastera tej strony WWW. Domyślnie jest to adres *webmaster@localhost*, czyli konto pocztowe, które prawdopodobnie nie istnieje. Można skonfigurować alias pocztowy w konfiguracji serwera poczty (patrz rozdział 27., „Konfiguracja serwera poczty”), który będzie przekazywał pocztę na inne, istniejące konto, lub po prostu wpisać tu odpowiedni adres, np. adres prawdziwego webmastera wszystkich obsługiwanych serwisów WWW. Ja z reguły ustawiam ten adres na *webmaster@vonnhagen.org*.

W drugiej kolejności należy zwrócić uwagę na wiersz 19., w którym znajduje się odwzorowanie głównej części adresu URI (tzn. każdy adres rozpoczynający się od ukośnika w prawo (/), po którym natychmiast następuje koniec ciągu znaków) na podkatalog */apache2-default* w katalogu określonym dyrektywą *DocumentRoot*. Ten wiersz ma ustawić znak komentarza, więc w celu przywrócenia działania tej dyrektywy należy usunąć znak # znajdujący się na początku wiersza.

Następnie trzeba ponownie uruchomić serwer WWW, aby sprawdzić nasze zmiany w działaniu:

```
$ sudo /etc/init.d/apache2 restart
```

Uruchomienie tego samego adresu URI, co na rysunku 26.2, spowoduje wyświetlenie innej strony powitalnej, przedstawionej na rysunku 26.3.

Jak można zauważyć na tej stronie, jej autor utworzył łącznie do dokumentacji wskazujące katalog */manual/*. Jednak w całej instalacji serwera WWW oraz w jego konfiguracji nie występuje katalog o takiej nazwie. Utwórzmy go zatem. Utworzymy sekcję zbliżoną zawartością do definicji katalogu */doc/* (wiersze od 31. do 38.), ale w wersji nieco uproszczonej:

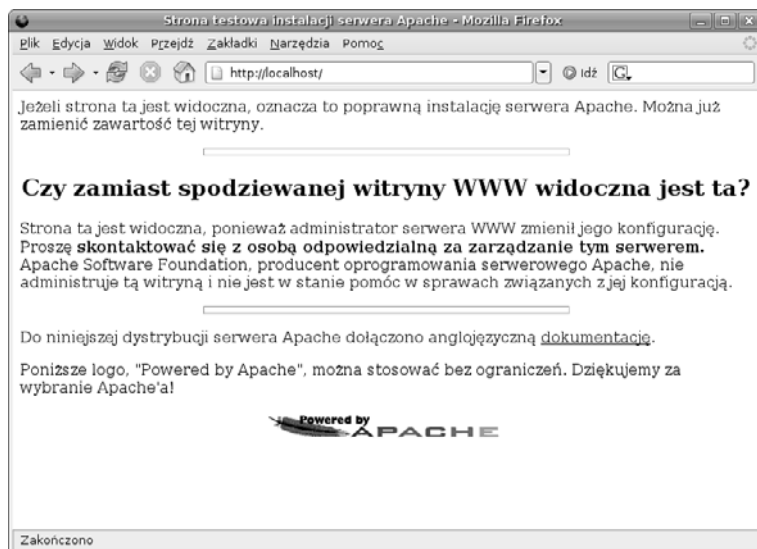
```
1. Alias /manual/ "/usr/share/doc/apache2-doc/manual/"  
2. <Directory "/usr/share/doc/apache2-doc/manual/">  
3.     Order deny,allow  
4.     Deny from all  
5.     Allow from 192.168.6 127.0.0.1  
6. </Directory>
```

Ponownie przypominam, nie należy wpisywać numerów wierszy w pliku, umieściłem je w przykładzie, aby łatwiej było analizować jego zawartość.



Rysunek 26.3.

*Lepsza strona
domyślna dla nowo
zainstalowanego
serwera Apache*



Wiersz 1. definiuje alias o nazwie `/manual/`, wskazujący na katalog `/usr/share/doc/apache2-doc/manual/`, w którym rzeczywiście znajduje się podręcznik serwera Apache. Pozostałe wiersze definiują warunki dostępu do tego katalogu. Wiersz 2. określa początek sekcji odnoszącej się do katalogu `/usr/share/doc/apache2-doc/manual/`, a wiersz 6. identyfikuje koniec tej sekcji. Wiersze 3., 4. i 5. opisują sposób autoryzacji dostępu. Wiersz 3. informuje o tym, że wpisy dotyczące blokad (`deny`) są analizowane przed definicjami udostępnień (`allow`). Wiersz 4. definiuje pełną blokadę dostępu do tego katalogu, natomiast wiersz 5. określa wyjątki od poprzedniej blokady, tzn. hosty z podsieci `192.168.6.*` (czyli sieci lokalnej, w której działa serwer) oraz interfejs `loopback` serwera. Po dopisaniu tych zmian (muszą wystąpić przed dyrektywą `</VirtualHost>` znajdującą się w naszym przykładzie w wierszu 39., ponieważ należą do globalnych definicji serwera WWW, wykorzystywanych również przez ten host wirtualny) należy ponownie uruchomić serwer WWW:

```
$ sudo /etc/init.d/apache2 restart
```

Po kliknięciu URI wskazującego dokumentację pojawi się strona przedstawiona na rysunku 26.4, zawierająca łącza do odpowiednich sekcji dokumentacji.

Należy zwrócić uwagę, że w dopisanej przez nas sekcji nie było odpowiednika wiersza 33., który znajdował się w naszej sekcji wzorcowej (z oryginalnej wersji pliku). Nie ma potrzeby definiowania tych opcji przeglądania, ponieważ wiemy, że w tym katalogu znajdują się odpowiednie pliki HTML samodzielnie realizujące funkcję przeglądania zawartości. Omówmy je teraz.

- ♦ **Indexes:** serwer sam buduje indeks zawartości katalogu w przypadku braku pliku `index.html` (lub jego odpowiednika);
- ♦ **MultiViews:** uaktywnia mechanizm negocjacji zawartości, w którym przeglądarka próbuje znaleźć najlepszą postać treści. W przypadku dokumentacji mamy do czynienia z jedną wersją języka, lokalizacji i strony kodowej, nie ma potrzeby stosowania negocjacji.

Rysunek 26.4.

Dokumentacja serwera Apache po wprowadzeniu odpowiednich modyfikacji w pliku konfiguracyjnym serwera



- ♦ FollowSymLinks: wiem, że w tym katalogu nie ma dowiązań symbolicznych, nie ma więc potrzeby wymuszania na serwerze śledzenia dowiązań.

Dokumentacja serwera Apache zawiera bardzo dobre omówienie dyrektyw konfiguracji serwera. Nie mam możliwości pełnego ich omówienia w tym krótkim rozdziale.

Usuwanie problemów

Jak w każdym przypadku występowania problemów, dzienniki systemowe serwera Apache stanowią bezcenne źródło informacji. W wierszach 25., 28. i 29. w oryginalnej konfiguracji serwera (przedstawionej wyżej) znajduje się definicja plików dziennika wykorzystywanych przez serwer oraz ich poziomu szczegółowości. Wiersz 25. identyfikuje nazwę pliku dziennika błędów: `/var/log/apache2/error.log`. Wiersz 28. definiuje poziom szczegółowości, który domyślnie obejmuje ostrzeżenia (*warnings*), co jest nieco bardziej użyteczne od zapisywania samych komunikatów o błędach (*errors*), ale nieco mniej szczegółowe od poziomu *debug*, który przydaje się podczas skomplikowanych konfiguracji nowych serwerów lub serwisów WWW. Wiersz 29. zawiera dyrektywę określającą plik dziennika dostępu `/var/log/apache2/access.log`, w którym zapisywane są żądania dostępu do serwera w formacie *NCSA combined*.

Do celów diagnostycznych szczególnie dobrze nadają się następujące pliki dzienników serwera Apache:

- ♦ **access.log**: zapisuje próby dostępu do serwera, zawiera adres IP hosta nawiązującego połączenie, znacznik czasu próby, przesłane żądanie oraz informację o wykorzystanej przeglądarce,
- ♦ **error.log**: zawiera komunikaty diagnostyczne od poziomu ostrzeżeń wzwyż (tzn. o wyższym poziomie ważności) dotyczące zdarzeń serwera występujących

przy przetwarzaniu żądań dostępu. Znajdują się tu komunikaty o nieodnalezionych stronach, katalogach, do których dostęp został zabroniony itp.

Poziomy diagnostyczne stosowane przez serwer Apache 2 pozwalają na precyzyjne kontrolowanie rozmiaru i typu informacji pojawiających się w dziennikach. Mamy do dyspozycji następujące poziomy:

- ♦ **emerg**: zdarzenia awaryjne, które mogą doprowadzić serwer WWW do utraty stabilności;
- ♦ **alert**: zdarzenia wymagające natychmiastowego działania, które mogą identyfikować problemy w systemie;
- ♦ **crit**: błędy krytyczne, które mogą informować o problemach bezpieczeństwa lub innych problemach serwera czy systemu;
- ♦ **error**: błędy niekrytyczne informujące o brakujących stronach, błędnych dyrektywach konfiguracyjnych i ogólnych sytuacjach błędów;
- ♦ **warn**: komunikaty ostrzeżeń o niekrytycznych problemach lub sytuacjach wymagających zbadania;
- ♦ **notice**: komunikaty o zdarzeniach normalnych, ale o dużym znaczeniu, na które należałoby zwrócić uwagę;
- ♦ **info**: komunikaty informacyjne, które mogą być pomocne w identyfikacji potencjalnych problemów lub sugerują możliwe rozwiązania;
- ♦ **debug**: zapisuje informacje o praktycznie każdej zmianie stanu w systemie: o każdym otwartym pliku czy każdej aktywności serwera podczas jego inicjalizacji i działania.

Na serwerach produkcyjnych nie należy ustawiać szczegółowości dzienników na poziomie diagnostycznym niższym niż **crit**. Z reguły poziom **warn** wydaje się najlepszy dla serwerów produkcyjnych, a w przypadku wystąpienia problemów z wydajnością lub funkcjonowaniem można tymczasowo ustawić poziom **notice**, **info** lub **debug**. Ponadto należy pamiętać, że aby pliki dzienników spełniały swoje zadanie, należy po prostu do nich zaglądać.

Więcej informacji

Istnieje sporo ciekawych źródeł informacji na temat serwera Apache 2 w internecie i w książkach. Do moich ulubionych zasobów należą:

- ♦ <http://httpd.apache.org/docs/2.0/> — nigdzie nie można znaleźć więcej informacji niż w dokumentacji Apache 2; dokumentacja Apache 2, oprócz podręczników referencyjnych, zawiera poradniki oraz artykuły typu „jak to zrobić”, w których można znaleźć porady praktyczne,
- ♦ *Apache Server 2 Bible*, Mohammed J. Kabir (Wiley, 2002, ISBN: 0-7645-4821-2),
- ♦ *Hardening Apache*, Tony Mobily (Apress, 2004; ISBN: 1590593782).

Wiele przydatnych informacji można znaleźć za pomocą wyszukiwarki, np. Google lub Clusty, trzeba poszukać napotkanego komunikatu o błędzie, szczególnej konfiguracji itp.

Podsumowanie

W tym rozdziale omówiłem instalację i konfigurację serwera WWW Apache 2, jednej z najpowszechniej wykorzystywanych aplikacji w instalowanych przeze mnie sieciowych systemach Linux. Serwer WWW jest użyteczny nie tylko do dostarczania stron WWW, ale również umożliwia tworzenie aplikacji obsługiwanych przez WWW, stosowanych często do administracji i monitorowania systemu.

W następnym rozdziale kontynuuję rozpoczętą tutaj tematykę serwerową. Omówię w nim zagadnienia związane z serwerami poczty elektronicznej w systemach Linux. Zawarłem w nim omówienie instalacji i konfiguracji serwera Postfix: elastycznego, łatwego w konfiguracji, a jednocześnie oferującego duże możliwości serwera poczty, będącego popularną alternatywą w stosunku do tradycyjnego serwera sendmail. W rozdziale 27. omówię również instalację i integrację oprogramowania wspierającego działanie serwera poczty, czyli rozwiązania antyspamowe i oprogramowanie antywirusowe.

Rozdział 27.

Konfiguracja serwera poczty

W tym rozdziale:

- ♦ Serwerowe technologie poczty elektronicznej
- ♦ Instalacja Postfixa, kontroli antyspamowej i antywirusowej
- ♦ Konfiguracja Postfixa
- ♦ Obsługa protokołu POP3
- ♦ Konfiguracja filtrów antyspamowych
- ♦ Konfiguracja skanerów antywirusowych

W rozdziale 26. stwierdziłem, że większość znaczących postępów w komputeryzacji wiąże się z powstaniem przełomowego rozwiązania — jednej, znacząco unikalnej, niezwykle użytecznej i inspirującej technologii, która przyciąga wielu entuzjastów i na długi czas staje się integralną częścią informatyki. W przypadku internetu takim rozwiązaniem była sieć WWW (ang. *World Wide Web*), ale ogólnie dla sieci tego typu technologią tą z pewnością była poczta elektroniczna. Trudno zaprzeczyć, że internet jest największą i najbardziej użyteczną siecią na świecie, ale wiele firm wykorzystywało pocztę elektroniczną na długo przedtem, zanim miały dostęp do internetu.

Poczta elektroniczna może być niesamowitym pożeraczem czasu, ale też jest krwioobiegiem osobistej komunikacji współczesnego świata. Do największych zalet wykorzystania poczty e-mail w zastosowaniach biznesowych należy fakt, iż jest to medium asynchroniczne (e-mail można wysłać do dowolnej osoby z firmy, o dowolnej porze dnia i nocy i istnieje duże prawdopodobieństwo, że prędzej lub później osoba ta zapozna się z wiadomością) i niezależne od lokalizacji (wiadomość e-mail wysyła się do osoby mającej dostęp do internetu, nie do konkretnego adresu fizycznego). Osobista poczta e-mail pozwala utrzymać kontakt z przyjaciółmi i rodziną, niezależnie od tego, jak daleko się znajdują. Oczywiście technologia ta, do działania potrzebuje niezawodnych mechanizmów transportowych (internetu lub sieci korporacyjnej) oraz oprogramowania obsługującego transmisję poczty między komputerami i dostarczanie tej poczty do skrzynki pocztowej oraz umożliwiającego odczyt, zapisywanie i możliwość udzielania odpowiedzi.

W rozdziale 8. zamieściłem opis możliwości programu Evolution, doskonałej aplikacji do zarządzania informacjami osobistymi, instalowanej w ramach standardowej dystrybucji Ubuntu. W tym rozdziale omówię instalację i konfigurację serwera Postfix, nie mniej potężnego oprogramowania służącego do obsługi transmisji poczty. Opiszę sposoby integracji Postfixa z różnymi pakietami dostarczania poczty oraz sposoby rozszerzenia jego możliwości o filtrowanie spamu, będącego największą bolączką trapiącą systemy poczty elektronicznej współczesnego świata. Zanim jednak zagłębię się w szczegóły, w pierwszym podrozdziale omówię mechanizmy działania poczty e-mail, zdefiniuję podstawową terminologię oraz niezbędne elementy oprogramowania. Będzie to przegląd serwerowych technologii poczty elektronicznej. Podam też argumenty, dla których warto zdecydować się na samodzielne utrzymanie serwera poczty. Doświadczeni administratorzy (i ryzykanci) mogą przejść do kolejnych podrozdziałów, w których omówię instalację, konfigurację i mechanizmy antyspamowe i antywirusowe.

Serwerowe technologie poczty elektronicznej

Elementy systemu poczty elektronicznej mogą początkowo wydać się niezrozumiałe, dlatego z reguły dzielę je na mechanizmy dostarczania i pobierania poczty. Serwer poczty otrzymuje pocztę, a klient poczty (Evolution) pozwala mi pobrać pocztę z serwera, przeczytać ją, zarchiwizować lub usunąć itd. W oficjalnej terminologii, mechanizmy poczty elektronicznej dzielimy na trzy grupy.

- ♦ **Mail Transfer Agent (MTA)**: program obsługujący przesyłanie poczty z jednego komputera do drugiego w wykorzystaniem protokołu *Simple Mail Transfer Protocol* (SMTP). Do popularnych MTA zalicza się Microsoft Exchange, Sendmail, Postfix, qmail i exim. MTA często określa się po prostu serwerami poczty, co jest mylące, ponieważ serwerami poczty nazywa się też komputery, na których działają MTA.
- ♦ **Mail Delivery Agent (MDA)**: program, który faktycznie dostarcza wiadomość odebraną przez MTA do skrzynki pocztowej użytkownika, często po drodze realizując dodatkowo funkcję wstępnej obróbki (np. poszukiwanie wirusów czy filtrowanie spamu) oraz autorespondera (wysyłającego wiadomość w stylu „jestem na wakacjach”). MDA często obsługuje mechanizmy pośrednie między skrzynką pocztową a klientem poczty, implementując protokoły odbierania poczty, np. *Post Office Protocol* (POP and POP3) oraz *Internet Message Access Protocol* (IMAP). Do popularnych MDA dla Linuksa zalicza się procmail, Qpopper, Cyrus, pop3d, Courier IMAP oraz */bin/mail*. MDA mogą być zintegrowane bezpośrednio z MTA, jak ma to miejsce w przypadku serwera Microsoft Exchange oraz wielu MTA dla Linuksa, w których poczta jest odczytywana bezpośrednio z tradycyjnych uniksowych plików poczty.
- ♦ **Mail User Agent (MUA)**: program pozwalający użytkownikowi wysyłać pocztę i odczytywać wiadomości z osobistej skrzynki pocztowej przy pomocy protokołów POP, POP3 i IMAP. Większość MUA oferuje również narzędzia do tworzenia wiadomości i zarządzania nimi w skrzynce. Do popularnych MUA zalicza się Evolution, Microsoft Outlook, mutt, kmail, pine i Thunderbird.

Zwykle użytkownicy nie zauważają istnienia MDA, ponieważ są zintegrowane z MTA przez dostawcę oprogramowania lub administratora systemu. Podobnie większość użytkowników nie stosuje określenia MTA i mówi potocznie „serwer poczty” w odniesieniu do MTA oraz hosta, na którym działa MTA. MUA jest najważniejszym elementem systemu pocztowego z punktu widzenia użytkownika, MTA jest najważniejszym elementem systemu pocztowego z punktu widzenia administratora, ponieważ realizuje funkcje transferu poczty z serwera na serwer, a dodatkowo identyfikuje MDA i inne aplikacje biorące udział w przetwarzaniu poczty po drodze do skrzynki pocztowej.

Popularne linuksowe aplikacje MTA

Jak łatwo przewidzieć, dla Linuksa istnieje kilka MTA. W tym rozdziale większość uwagi skupiam na instalacji i konfiguracji Postfixa, ale istnieją inne MTA posiadające rzesze zwolenników. Z tego powodu warto wiedzieć co nieco na temat różnych popularnych MTA, aby decyzję o zastosowaniu określonego rozwiązania podjąć w oparciu o rzetelne przesłanki (oczywiście, postaram się dowiedzieć, że warto użyć opisanego w tym rozdziale Postfixa). W kilku kolejnych podpunktach rozdziału omówię różne popularne rozwiązania MTA.

Exim

Exim to popularny, szybki i elastyczny MTA napisany przez Philipa Hazela z University of Cambridge. Program ten powstał w celu zastąpienia popularnego kiedyś MTA o nazwie smail. Exim jest bardzo popularny wśród administratorów, którzy oprócz poczty opiekują się listami dyskusyjnymi, takimi jak mailman, ponieważ można go skonfigurować w taki sposób, aby sprawdzał listy e-mailowe przed użytkownikami. Exim posiada dużą liczbę opcji pozwalających na zdefiniowanie mechanizmów sprawdzania poczty przychodzącej pod kątem problemów i nie potrzebuje do tego zewnętrznych programów, co wśród MTA jest rzadkością. Exim jest instalowany domyślnie w niektórych dystrybucjach Linuksa, np. w Debianie.

Więcej informacji na temat serwera Exim można znaleźć:

- ♦ na stronie domowej projektu Exim: <http://www.exim.org>;
- ♦ w dokumentacji i FAQ projektu Exim: <http://www.exim.org/docs.html>.

Postfix

Postfix to popularny, skalowalny i bezpieczny MTA napisany przez Wietse Venema w okresie, gdy pracował w firmie IBM. Postfix początkowo nosił nazwę VMailer, a przez firmę IBM był reklamowany jako Secure Mailer. W 1999 roku nazwa została zmieniona na Postfix.

Postfix jest stabilny, szybki i łatwy w administracji. Jego plik konfiguracyjny jest czytelny i łatwy w modyfikacji, choć — oczywiście — należy poznać opcje i ich możliwe wartości. Postfix to serwer, którym można z łatwością zastąpić serwer Sendmail, z zaznaczeniem, że ze względów bezpieczeństwa, wymagane jest wykonanie pewnych operacji z uprawnieniami użytkownika i grupy *postfix*.

Więcej informacji o serwerze Postfix można znaleźć:

- ♦ na stronie projektu Postfix: <http://www.postfix.org>;
- ♦ w dokumentacji i FAQ projektu Postfix: <http://www.seaglass.com/postfix>;
- ♦ na Postfix Shrine: <http://www.stahl.bau.tu-bs.de/~hildeb/postfix>.

Qmail

Qmail jest niezwykle szybkim i skalowalnym MTA napisanym przez Dana Bernsteina, profesora nauk informatycznych na Uniwersytecie Illinois w Chicago, który jest doskonale znany ze swoich prac z zakresu kryptografii i bezpieczeństwa. Te zainteresowania są wyraźnie zaznaczone w projekcie qmail — od 1997 roku istnieje nagroda w wysokości 500 dolarów dla każdego, komu uda się znaleźć dziurę bezpieczeństwa w najnowszej wersji qmaila. Dość powiedzieć, że nigdy nie musiała być wypłacona.

Podstawowe cechy Qmaila to niezwykła wydajność, bezpieczeństwo i skalowalność. Niestety, jego licencja jest obciążona dziwnymi wymogami (np. nie wolno dystrybuować zmodyfikowanych źródeł qmaila bez wyraźnej zgody autora), nie jest zgodny ze standardem Linux Filesystem Hierarchy Standard (qmail wykorzystuje katalog `/var`), a poziom komplikacji początkowej konfiguracji związanej z dodawaniem użytkowników jest przyczyną większości moich siwych włosów. Problemy licencyjne wpłynęły na mniejszą popularność qmaila w środowisku open source, ponieważ mimo dostępnych kodów źródłowych licencja qmaila nie jest zgodna z ideą open source.

Więcej informacji o serwerze qmail można znaleźć:

- ♦ na stronie serwera qmail Dana Bernsteina: <http://cr.jp.to/qmail.html>;
- ♦ na Life with qmail: <http://www.lifewithqmail.org>;
- ♦ na stronie domowej projektu qmail: <http://www.qmail.org>.

Sendmail

Jak wspominałem wcześniej, sendmail jest najczęściej stosowanym MTA w systemach Unix i Linux. Celowo nie napisałem „najpopularniejszym”, ponieważ po prostu sendmail jest instalowany domyślnie z większością systemów Unix i dystrybucji Linuksa, przez co jest używany po prostu dlatego, że „już jest”. Jeśli sendmail jest już wykorzystywany na wielu serwerach w twojej firmie, trudno zmuszać kogoś by w nowym systemie zainstalował inny MTA, zgodnie z zasadą administratorów, mówiącą: „Jeśli nie jest zepsute, nie naprawiaj”.

Sendmail został napisany przez Erica Allmana, którego delivermail był standardowym systemem dostarczania poczty ARPANET-u, wywodzącym się z 4.0 BSD Unix i wcześniejszych wersji 4.1 BSD. ARPANET rozrósł się jednak i w końcu przekształcił się w Internet, delivermail stał się za mało elastyczny przede wszystkim z tego powodu, że jego konfiguracja była wkompiłowana w program. W związku z tym został opracowany Sendmail wykorzystujący zewnętrzny plik konfiguracyjny. Początkowo był udostępniony razem z ostatnią wersją BSD 4.1. Plik konfiguracyjny Sendmaila miał niezwykle rozbudowaną składnię o bardzo dużym poziomie komplikacji, co było jednym z głównych argumentów

przeciwko jego wykorzystaniu. Pliki konfiguracyjne Sendmaila są generowane z plików pośrednich w formacie m4 edytowanych przez administratora serwera, a następnie przekształcane w pliki Sendmaila za pomocą procesora makr *m4*. Taki dwuetapowy proces edycji konfiguracji znacznie komplikuje zarządzanie oraz proces diagnostyki ewentualnych błędów. Pytania na temat konfiguracji Sendmaila są niezwykle powszechne w ramach rozmów kwalifikacyjnych na administratorów systemów.

Zaletą Sendmaila jest to, że nie ma w internecie dojrzałego serwera poczty, ponieważ projekt jest aktywny od ponad dwudziestu lat. Sendmail jest nieźle skalowalny i można go zintegrować z większością popularnych pakietów antyspamowych i antywirusowych. Sendmail ma nie najlepszą reputację (nie do końca zasłużenie) z powodu problemów z bezpieczeństwem, co — moim zdaniem — w znacznej mierze wynika z faktu, że jest on najstarszym i największym celem ataków.

Więcej informacji o serwerze Sendmail można znaleźć:

- ♦ na stronie projektu Open Source Sendmail: <http://www.sendmail.org>;
- ♦ na stronie firmy Sendmail, Inc.: <http://www.sendmail.com> (w której Eric Allman pełni funkcję CTO).

Po co instalować własny serwer poczty?

Istnieje sporo argumentów za tym, aby wykorzystywać własny serwer poczty, ale taka decyzja wiąże się ze zwiększeniem odpowiedzialności oraz kosztów wewnętrznych. Niemal wszystkie firmy zarządzające domenami oraz dostawcy usług oferują konta pocztowe na swoich serwerach oferując możliwość ich drobnej konfiguracji przez klienta. Często dostępne są panele administracyjne wyposażone w interfejs WWW, umożliwiające tworzenie kont na serwerze oraz aliasów do nich. Jednak żaden dostawca usług internetowych nie zagwarantuje pełnej kontroli nad serwerem poczty. Oto najważniejsze powody samodzielnego utrzymania serwera poczty.

- ♦ Brak ograniczeń liczby użytkowników i aliasów. Większość dostawców usług internetowych oferuje pakiety składające się z określonej liczby kont, a za dodatkowe żądają opłat.
- ♦ Brak ograniczenia liczby domen obsługiwanych przez pojedynczy serwer poczty. Większość dostawców usług internetowych pobiera opłaty od każdej domeny, a ich narzędzia administracyjne działają niezależnie dla każdej z domen. Pojedynczy serwer poczty można jednak skonfigurować w taki sposób, aby obsługiwał wiele domen, a pojedyncza konfiguracja serwera będzie miała zastosowania do nich wszystkich.
- ♦ Możliwość wykorzystania dowolnego serwera poczty; dostawcy usług internetowych nie dają z reguły wyboru serwera poczty. Można skonfigurować dowolną kombinację MTA i MDA najlepiej dostosowaną do potrzeb, a nie dla wygody dostawcy internetu.
- ♦ Możliwość integracji scentralizowanej kontroli antyspamowej i antywirusowej pod pełną kontrolą użytkownika, którą można bez większych kosztów dostosować do własnych potrzeb.

- ♦ Łatwa integracja dodatkowych możliwości, takich jak obsługa list e-mailowych, na co nie zawsze można liczyć u dostawców usług internetowych.

Choć posiadanie własnego serwera poczty ma znaczące zalety, to jednak wprowadza pewne koszty i zwiększa odpowiedzialność, i należy być na to przygotowanym. Oto problemy, które trzeba będzie uwzględnić.

- ♦ **Bezpieczeństwo staje się wewnętrznym problemem i odpowiedzialnością organizacji.** Gdy serwerem poczty zarządza dostawca internetu, to on jest odpowiedzialny za bezpieczeństwo. Gdy firma samodzielnie zarządza serwerem poczty, przejmuje odpowiedzialność za jego bezpieczeństwo, co jest szczególnie ważne dla poczty IMAP, która jest przechowywana na serwerze.
- ♦ **Należy zapewnić stały dostęp do serwera poczty.** Oprócz konieczności utrzymania pracy serwera bez przerw, należy zapewnić, że rekordy MX domeny są skonfigurowane prawidłowo dla serwera poczty. Konieczny będzie stały adres IP lub prawidłowa konfiguracja solidnego i stale dostępnego systemu dynamicznego DNS-u.
- ♦ **Administrator systemu będzie odpowiedzialny za wszystkie zadania związane z serwerem.** Oprócz zadbania o jego prawidłową konfigurację (bardzo proszę, bez otwartych pośredników!), będzie odpowiedzialny za pełną dostępność serwera pod adresem IP wskazywanym przez rekord MX w DNS-ie, za wykonywanie kopii zapasowych, za aktualizację systemu oraz prawidłowe działanie kontroli antyspamowej i antywirusowej oraz wielu innych.

Decydując się na zainstalowanie własnego serwera poczty, łatwo przeoczyć wspomniane wyżej trudności. Gdy wszystko działa prawidłowo, a dzieje się tak w końcu przez większą część czasu, wszystko wygląda doskonale. Jednak podczas awarii administrator znajduje się na cenzurowanym.

Instalacja Postfixa i pakietów pomocniczych

W tym podrozdziale wyjaśniam, w jaki sposób zainstalować pakiety niezbędne do zbudowania prostej konfiguracji serwera Postfix, automatycznie skanującej wiadomości e-mail w poszukiwaniu wirusów, z możliwością pobierania poczty z użyciem protokołu POP3. Postfix i jego pakiety pomocnicze instaluje się na różne sposoby, w zależności od wersji systemu: Ubuntu Server CD, Ubuntu Alternate CD, czy też Ubuntu Desktop CD. Różnice sprowadzają się jednak głównie do tego, czy przy instalacji istnieje możliwość użycia graficznego interfejsu użytkownika.

- ♦ Gdy system nie posiada graficznego interfejsu użytkownika, należy zastosować się do instrukcji z punktu „Instalacja Postfixa z wiersza poleceń”.
- ♦ Gdy system posiada graficzny interfejs użytkownika, należy zastosować się do instrukcji z punktu „Instalacja Postfixa i pakietów pomocniczych z użyciem menedżera pakietów Synaptic”.

Oczywiście, gdy użytkownik jest posiadaczem graficznego interfejsu, może wykorzystać instrukcje dotyczące instalacji z wiersza poleceń, wywołując je z terminala GNOME lub okna programu xterm. Ubuntu pozwala każdemu wykorzystać takie narzędzia, jakie są dla niego najwygodniejsze.

W poniższej liście opisałem w kolejności alfabetycznej wszystkie pakiety, które należy zainstalować.

- ♦ *clamav*: pakiet antywirusowy skanujący pocztę przychodzącą w poszukiwaniu załączników zawierających wirusy. W razie potrzeby dokonuje ich kwarantanny, zapobiegając tym samym przypadkowemu otwarciu zainfekowanego załącznika przez użytkownika.
- ♦ *mailx*: pakiet zawierający proste narzędzie obsługiwane z wiersza poleceń, służące do wysyłania poczty, wykorzystywane przez wiele narzędzi systemowych, a w naszym przypadku bardzo użyteczne do szybkiego testowania poprawności działania serwera.
- ♦ *mailscanner*: pakiet służący do filtrowania poczty przychodzącej w celu blokowania spamu, obsługujący wiele technik rozpoznawania i filtrowania niechcianej poczty.
- ♦ *postfix*: serwer pocztowy Postfix i jego narzędzia wraz z plikami konfiguracyjnymi.
- ♦ *postfix-doc*: dokumentacja serwera Postfix, instalowana w katalogach `/usr/share/doc/postfix` oraz `/usr/share/doc/postfix-doc`.
- ♦ *postgrey*: obsługa mechanizmu szarych list (*greylisting*) dla serwera Postfix. Technika ta polega na odrzucaniu pierwszej próby połączenia z obcego serwera poczty, ale drugie żądanie jest już akceptowane. Wielu spamerów podejmuje tylko jedną próbę, dzięki czemu to proste narzędzie może znacząco zredukować ilość otrzymywanego spamu.
- ♦ *qpopper*: pakiet umożliwiający użytkownikom zdalne czytanie poczty z wykorzystaniem protokołu POP/POP3, obsługiwanego przez większość czytników poczty (MUA).

Oczywiście, nie trzeba instalować wszystkich pakietów, ale w dalszej części rozdziału zakładam, że zostały zainstalowane.

Wbudowane modele konfiguracji Postfiksa

Niezależnie od tego, czy Postfix i pakiety pomocnicze są instalowane z wiersza poleceń, czy za pomocą graficznego interfejsu użytkownika, podczas instalacji pojawi się okno wyboru modelu konfiguracji serwera. W pakiecie Postfiksa do wyboru jest pięć modeli konfiguracji, a każdy z nich został dostosowany do innego scenariusza. Mamy do dyspozycji następujące opcje.

- ♦ **Internet site**: serwer internetowy. Poczta jest dostarczana do tego serwera bezpośrednio przez inne serwery SMTP; służy on również jako serwer SMTP dla programów MUA. To domyślna konfiguracja, gdy serwer ma służyć jako serwer poczty przychodzącej i wychodzącej przy bezpośrednim podłączeniu do internetu.

- ♦ **Internet site using smarthost:** serwer internetowy z użyciem smarthosta. Ten serwer otrzymuje pocztę bezpośrednio z internetu lub pobiera ją z innego serwera z użyciem programów dodatkowych, takich jak fetchmail, ale pocztę wysyła za pomocą innego serwera poczty. Ta opcja jest korzystna w przypadku konfiguracji serwera domowego lub w niewielkim biurze (SOHO) posiadającym własny serwer poczty, korzystający z serwera pocztowego dostawcy internetu lub firmy hostingowej..
- ♦ **Local delivery only:** wyłącznie dostarczanie lokalne. Serwer dostarcza wyłącznie pocztę lokalnie, nie wysyła jej do innych sieci. Ta opcja jest odpowiednia w systemach, które przesyłają komunikaty diagnostyczne w sieci lokalnej, lub dla serwera służącego do przesyłania poczty między określoną grupą hostów.
- ♦ **No configuration:** brak konfiguracji. Proces instalacyjny nie zmodyfikuje konfiguracji Postfixa. Do poprawnego działania konieczne będzie dokonanie skomplikowanego procesu konfiguracji ręcznej.
- ♦ **Satellite system:** system satelicki. Ta maszyna nie otrzymuje poczty, a pocztę wysyłaną przekazuje do innego serwera poczty.

W systemie, którego konfiguracja nie pasuje do żadnego z opisanych wyżej scenariuszy, najlepiej wybrać domyślną wartość, czyli *Internet Site*, ponieważ stanowi ona rozsądną szablona, na podstawie którego można zdefiniować własną, nietypową konfigurację.

Instalacja Postfixa z wiersza poleceń

Serwer Postfix wraz z pakietami dodatkowymi łatwo zainstalować z wiersza poleceń, używając programu `apt-get` lub `aptitude`. Sugeruję zastosowanie programu `aptitude`, bo można skorzystać z jego funkcji automatycznej instalacji pakietów zalecanych wraz z pakietami głównymi, niezbędnymi do uruchomienia Postfixa w systemie Ubuntu.

Aby zainstalować serwer poczty Postfix i związane z nim aplikacje za pomocą `aptitude`, należy wywołać następujące polecenie:

```
$ sudo aptitude -r install postfix postfix-doc postgrey mailscanner mailx qpopper clamav
```

Program `sudo` poprosi o podanie hasła użytkownika, po czym wypisze komunikat potwierdzenia instalacji pakietów określonych w wierszu poleceń, pakietów zależności oraz pakietów zalecanych do instalacji. Efekt będzie zbliżony do poniższego listingu (dokładna zawartość tej listy może różnić się, w zależności od pakietów zainstalowanych w systemie):

```
$ sudo aptitude -r install postfix postfix-doc postgrey mailscanner mailx qpopper clamav
Password:
Reading package lists... Done
Building dependency tree... Done
Reading extended state information...
Initializing package states... Done
Building tag database... Done
The following NEW packages will be automatically installed:
  arj ca-certificates clamav-base clamav-freshclam debconf-utils emacs21
  emacs21-bin-common emacs21-common emacsen-common laptop-detect
  libarchive-zip-perl libberkeleydb-perl libclamav1 libcompress-zlib-perl
  libconvert-binx-perl libconvert-tnef-perl libcurl3 libdigest-hmac-perl
```



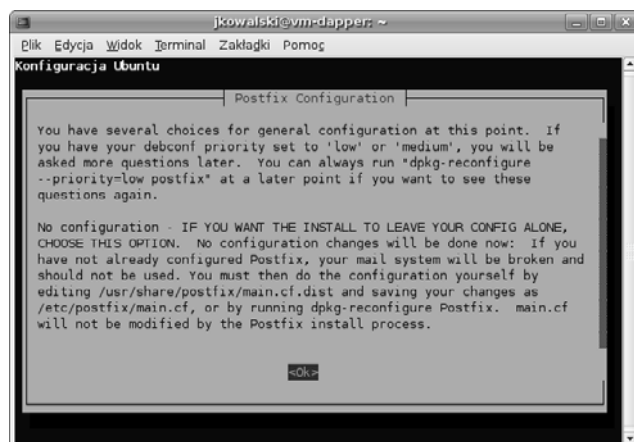
```
libdigest-shal-perl libgmp3c2 libhtml-parser-perl libhtml-tagset-perl
libice6 libidn11 libio-multiplex-perl libio-stringy-perl libjpeg62
liblockfile libmail-spf-query-perl libmailtools-perl libmime-perl
libnet-cidr-lite-perl libnet-cidr-perl libnet-dns-perl libnet-ip-perl
libnet-server-perl libpng12-0 libsm6 libsocket6-perl libssl0.9.7 libtiff4
libtimedate-perl libungif4g liburi-perl libx11-6 libxau6 libxext6 libxmu6
libxpm4 libxt6 mew-beta mew-beta-bin openssl perl perl-doc perl-modules
resolvconf spamassassin spamc ssl-cert tnef ucf unzip unzoo x11-common
xaw3dg
```

The following NEW packages will be installed:

```
arj ca-certificates clamav clamav-base clamav-freshclam debconf-utils
emacs21 emacs21-bin-common emacs21-common emacsen-common laptop-detect
libarchive-zip-perl libberkeleydb-perl libclamav1 libcompress-zlib-perl
libconvert-binhex-perl libconvert-tnef-perl libcurl3 libdigest-hmac-perl
libdigest-shal-perl libgmp3c2 libhtml-parser-perl libhtml-tagset-perl
libice6 libidn11 libio-multiplex-perl libio-stringy-perl libjpeg62
liblockfile libmail-spf-query-perl libmailtools-perl libmime-perl
libnet-cidr-lite-perl libnet-cidr-perl libnet-dns-perl libnet-ip-perl
libnet-server-perl libpng12-0 libsm6 libsocket6-perl libssl0.9.7 libtiff4
libtimedate-perl libungif4g liburi-perl libx11-6 libxau6 libxext6 libxmu6
libxpm4 libxt6 mailscanner mailx mew-beta mew-beta-bin openssl perl
perl-doc perl-modules postfix postfix-doc postgrey qpopper resolvconf
spamassassin spamc ssl-cert tnef ucf unzip unzoo x11-common xaw3dg
0 packages upgraded, 73 newly installed, 0 to remove and 0 not upgraded.
Need to get 5028kB/45.3MB of archives. After unpacking 129MB will be used.
Do you want to continue? [Y/n/?]
```

W tym momencie można nacisnąć klawisz *Enter* lub klawisz *Y* i potwierdzić go klawiszem *Enter*, co spowoduje instalację serwera poczty Postfix wraz z pakietami pomocniczymi. Postfix zostanie również dopisany do sekwencji rozruchowej systemu, ponadto instalator automatycznie uruchomi ten serwer. Na etapie konfiguracji Postfixa instalator wyświetli okno konfiguracyjne przedstawione na rysunku 27.1.

Rysunek 27.1.
Dostępne modele
konfiguracji serwera
Postfix



Ten ekran przedstawia dostępne modele konfiguracji serwera Postfix, opisane wcześniej w punkcie „Wbudowane modele konfiguracji Postfixa”. Aby przejść do ekranu konfiguracyjnego (patrz rysunek 27.2), należy nacisnąć klawisz *Enter*.

Rysunek 27.2.

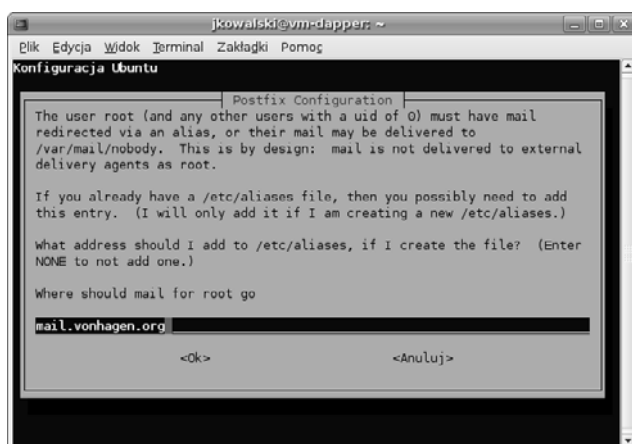
Wybór modelu
konfiguracji serwera
Postfix



Zwykle warto zdecydować się na standardowy model konfiguracji, *Internet Site*. Można również zmienić model, posługując się klawiszami strzałek w górę i w dół i zatwierdzając wybór klawiszem *Enter*. To spowoduje przejście do kolejnego ekranu konfiguracji, przedstawionego na rysunku 27.3.

Rysunek 27.3.

Określanie nazwy
serwera pocztowego
w trybie tekstowym



Ten ekran pozwala określić nazwę hosta serwera pocztowego, wykorzystywaną przez hosty zewnętrzne. Domyślnie pojawi się rzeczywista nazwa systemu. Wykorzystując klawisze strzałek w lewo i w prawo oraz klawisz *Del*, można wprowadzić niezbędne poprawki i zatwierdzić konfigurację klawiszem *Enter*. Dalsza konfiguracja powinna odbyć się bez dodatkowych pytań.

To jeszcze nie koniec konfiguracji serwera Postfix. Dalsze informacje podaję w podrozdziale „Konfiguracja serwera Postfix”.

Więcej informacji na temat instalacji pakietów z wiersza poleceń można znaleźć w rozdziale 20.

Instalacja Postfixa i pakietów pomocniczych z użyciem menedżera pakietów Synaptic

Aby zainstalować Postfix i pakiety pomocnicze w trybie graficznym, należy z menu systemowego wywołać funkcję *System/Administracja/Menedżer pakietów Synaptic*. W oknie programu trzeba wcisnąć przycisk *Szukaj*. Z listy rozwijanej należy wybrać opcję *Opis i nazwa*, a w polu wyszukiwania wpisać ciąg znaków postfix, po czym kliknąć przycisk *Szukaj*. Po odnalezieniu na liście pakietów *postfix*, *postfix-doc* i *postgrey* należy kliknąć je kolejno prawym przyciskiem myszy i z menu rozwijanego wybrać opcję *Zaznacz do instalacji*.



Przy zaznaczaniu tych pakietów do instalacji pojawi się okno z informacją o konieczności zainstalowania dodatkowych pakietów. W tym oknie należy kliknąć przycisk *Zaznacz*, akceptując instalację tych zależnych (i wymaganych) pakietów.

Następnie trzeba znaleźć ciąg znaków mail i zaznaczyć do instalacji pakiety *mailx* i *mailscanner*.

Później należy wyszukać ciąg znaków qpopper i zaznaczyć do instalacji pakiet o tej samej nazwie.

Na końcu trzeba wyszukać i zaznaczyć do instalacji pakiet *clamav*.

Na rysunku 27.4 przedstawiam pakiety *postfix* i *postfix-doc* zaznaczone do instalacji w menedżerze pakietów Synaptic.

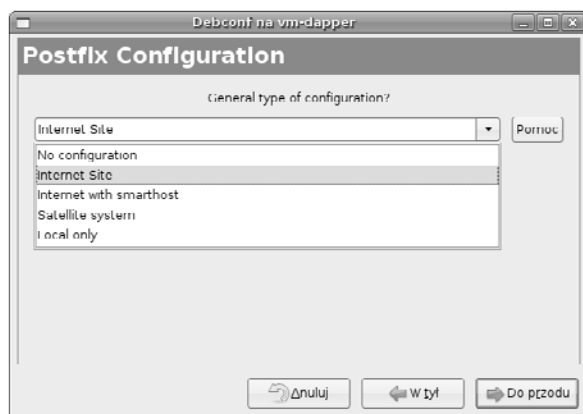
Rysunek 27.4.
Instalacja serwera poczty Postfix i pakietów pomocniczych



Po zaznaczeniu do instalacji wszystkich pakietów należy kliknąć ikonę *Zastosuj* w pasku narzędzi programu Synaptic. Na tym etapie konfiguracji serwera Postfix pojawi się okno przedstawione na rysunku 27.5.

Rysunek 27.5.

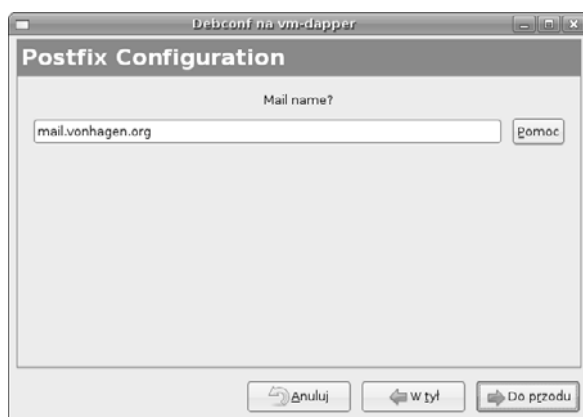
Wybór modelu
konfiguracji serwera
Postfix w trybie
graficznym



W oknie należy wybrać jeden z modeli konfiguracji Postfixa, opisanych wcześniej w punkcie „Wbudowane modele konfiguracji Postfixa”. Zwykle trzeba potwierdzić domyślną opcję *Internet Site*, ale można też wybrać dowolny z pozostałych modeli. Po dokonaniu wyboru trzeba go zatwierdzić, klikając przycisk *Dalej*. Pojawi się okno przedstawione na rysunku 27.6.

Rysunek 27.6.

Określanie nazwy
hosta pocztowego
w trybie graficznym



To okno pozwala wpisać nazwę hosta, która będzie wykorzystywana przez inne hosty. Domyślnie wprowadzona jest tu nazwa systemu. Po odpowiednim zmodyfikowaniu nazwy należy kliknąć przycisk *Do przodu*. Dalsza część konfiguracji powinna odbyć się bez dodatkowych pytań.

To jeszcze nie koniec konfiguracji serwera Postfix. Dalsze informacje zamieściłem w podrzdziale „Konfiguracja serwera Postfix”.

Więcej informacji na temat instalacji pakietów za pomocą menedżera pakietów Synaptic można znaleźć w rozdziale 20.

Konfiguracja serwera Postfix

W domyślnej instalacji serwer Postfix jest prawie gotowy do działania. Nadaje się już do lokalnego wykorzystania, co można sprawdzić, uruchamiając narzędzie `mail` obsługiwane z wiersza poleceń (dostępne w pakiecie *mailx* zainstalowanym w poprzednim punkcie). Aby serwer plików przyjmował pocztę i wysyłał ją do innych hostów, należy go dodatkowo skonfigurować. Postfix domyślnie obsługuje tylko lokalne dostarczanie poczty, co jest przydatne tylko dla tych użytkowników, którzy chcą logować się na serwerze poczty i swoją pocztę czytać bezpośrednio na serwerze. Niestety, to nie jest przykład dobrych praktyk z punktu widzenia bezpieczeństwa, a dodatkowo stanowi znaczne utrudnienie dla użytkowników. Z tego powodu należy dokonać modyfikacji konfiguracji Postfiksa, aby stał się bardziej użyteczny.

W następnym punkcie omawiam domyślną konfigurację Postfiksa (taką, jaką zaleciłem w poprzednich punktach). W kolejnych dwóch punktach napiszę, w jaki sposób skonfigurować Postfix, aby działał w sieci oraz jak skonfigurować serwer poczty, aby obsługiwał pobieranie poczty za pomocą protokołu POP.



Jak wspominałem w rozdziale 8. i na początku niniejszego rozdziału, podstawowymi protokołami służącymi do pobierania poczty ze zdalnych serwerów są POP i IMAP. Zasadnicza różnica między nimi polega na tym, że protokół POP został zaprojektowany w celu pobierania poczty przez klienta i przechowywania jej na stacji roboczej, natomiast IMAP przechowuje pocztę na serwerze. Każdy z tych modeli ma zalety, ale sądzę, że większość użytkowników jest bardziej przyzwyczajona do modelu POP, zatem właśnie nim się zajmujemy.

Pliki konfiguracyjne Postfiksa

Serwer Postfix w Ubuntu wykorzystuje trzy pliki konfiguracyjne, znajdujące się w katalogu */etc/postfix*:

- ♦ *dynamicmaps.cf*: identyfikuje dodatkowe funkcje, które mają być ładowane w zależności od wykonywanych operacji,
- ♦ *main.cf*: zawiera konfigurację Postfiksa wykorzystywaną do przetwarzania wiadomości e-mail,
- ♦ *master.cf*: zawiera parametry wykorzystywane przez główny program Postfiksa przy wywoływaniu innych programów z katalogu */usr/lib/postfix*.

Postfix może również wykorzystywać standardowe pliki konfiguracji serwera poczty, takie jak */etc/aliases* (który pozwala na definiowanie wielu wirtualnych adresów e-mail dla jednego konta), ale nie jest to plik specyficzny tylko dla niego.

Plik konfiguracyjny *main.cf* jest jedynym plikiem konfiguracyjnym, z którym administrator na pewno będzie miał do czynienia. W domyślnej instalacji w systemie Ubuntu (w modelu *Internet Site*) plik *main.cf* ma następującą zawartość (po usunięciu zbędnych komentarzy i pustych linii):

```
smtpd_banner = $myhostname ESMTP $mail_name (Ubuntu)
biff = no
append_dot_mydomain = no
#delay_warning_time = 4h
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
smtpd_use_tls=yes
smtpd_tls_session_cache_database = btree:${queue_directory}/smtpd_scache
smtp_tls_session_cache_database = btree:${queue_directory}/smtp_scache
myhostname = ubuntu32.vonhagen.org
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname
mydestination = mail.vonhagen.org, ubuntu32.vonhagen.org,
                localhost.vonhagen.org, localhost
relayhost =
mynetworks = 127.0.0.0/8
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
```

Omówię kolejno wszystkie opcje konfiguracyjne pojawiające się w tym listingu.

- ♦ `smtpd_banner`: ustawia tekst baneru wysyłanego do klienta po nawiązaniu połączenia SMTP. W tym przypadku baner jest konstruowany w oparciu o wartości innych zmiennych.
- ♦ `biff`: określa, czy lokalna usługa powiadomienia o pocście (`biff`) powinna być wykorzystywana za każdym razem, gdy Postfix przetwarza kolejkę poczty przychodzącej. W tym przypadku opcja jest nieaktywna, ponieważ usługa `biff` może powodować znaczne obniżenie wydajności i ma sens tylko na maszynach, na których, oprócz obsługi poczty wychodzącej, znajduje się również repozytorium poczty. Użytkownicy wykorzystujący lokalne pliki poczty mogą uaktywnić tę opcję.
- ♦ `append_dot_mydomain`: określa, czy Postfix powinien dopisywać nazwę domeny do adresów wysyłanych na adres typu *uzytkownik@host*. Obecnie tego typu funkcje są obsługiwane przez MUA, zatem ta opcja jest nieaktywna.
- ♦ `delay_warning_time`: uaktywnienie tej opcji pozwala określić okres między powiadomieniami o tym, że poczta nie została jeszcze dostarczona. W końcu zdalnym serwerem SMTP może być Microsoft Exchange, który może być „chwilowo niedostępny”.
- ♦ `smtpd_tls_cert_file`: identyfikuje pełną ścieżkę do pliku zawierającego certyfikat wykorzystywany przez tę maszynę do wysyłania poczty za pomocą mechanizmu TLS (ang. *Transport Layer Security*).
- ♦ `smtpd_tls_key_file`: identyfikuje pełną ścieżkę do pliku zawierającego prywatny klucz RSA klienta SMTP, wykorzystywany przez Postfix do wysyłania poczty przy pomocy TLS (ang. *Transport Layer Security*).
- ♦ `smtpd_use_tls`: określa, czy serwer Postfix powinien wykorzystywać mechanizm TLS w przypadku, gdy zdalny serwer poinformuje o obsłudze tego mechanizmu (wyśle komunikat STARTTLS). Jeśli zdalny serwer nie poinformuje o obsłudze TLS, komunikacja odbywa się bez szyfrowania.

- ♦ `smtpd_tls_session_cache_database`: identyfikuje format oraz lokalizację bufora sesji TLS (ang. *TLS session cache*) wykorzystywanego przez demon `tlsmgr` dla serwera SMTP.
- ♦ `smtp_tls_session_cache_database`: identyfikuje format oraz lokalizację bufora sesji TLS (ang. *TLS session cache*) wykorzystywanego przez demon `tlsmgr` dla klienta SMTP.
- ♦ `myhostname`: identyfikuje rzeczywistą internetową nazwę hosta. Domyślnie ta wartość jest zgodna z wartością zwracaną przez funkcję `gethostname()`.
- ♦ `alias_maps`: identyfikuje format i pełną ścieżkę do pliku aliasów pocztowych wykorzystywanych do lokalnego dostarczania poczty.
- ♦ `alias_database`: identyfikuje format i pełną ścieżkę do pliku aliasów pocztowych wykorzystywanych do lokalnego dostarczania poczty, aktualizowanego poleceniem `newaliases`. Z reguły jest to ten sam plik, co identyfikowany parametrem `alias_maps`.
- ♦ `myorigin`: identyfikuje nazwę hosta lub domeny, do której wysyłana jest lokalna poczta i z której poczta ta wychodzi. W systemach Ubuntu wpisana jest tu ścieżka do pliku zawierającego nazwę hosta podaną podczas instalacji. Opcja może być wystarczająca dla instalacji jednodomenowych. Dla serwerów obsługujących wiele domen okaże się niewystarczająca..
- ♦ `mydestination`: lista domen, do których poczta jest dostarczana z użyciem lokalnego mechanizmu dostarczania poczty.
- ♦ `relayhost`: gdy poczta wychodząca musi być przesyłana za pośrednictwem innego serwera poczty, ten parametr identyfikuje ten serwer poczty, w przeciwnym wypadku musi być pusty.
- ♦ `mynetworks`: identyfikuje sieci lub poszczególne hosty, które mogą przysyłać wiadomości przez nasz serwer. W tym przypadku serwer poczty będzie wysyłał pocztę wyłącznie otrzymaną na interfejsie *loopback*, co może stanowić pewien problem. To zagadnienie jest omówione w kolejnym punkcie zatytułowanym „Identyfikacja zaufanych hostów i domen”.
- ♦ `mailbox_size_limit`: identyfikuje maksymalny rozmiar skrzynki pocztowej. W tym przypadku liczba 0 oznacza brak limitu.
- ♦ `recipient_delimiter`: identyfikuje separator wykorzystywany wewnętrznie przez serwer Postfix do oddzielania poszczególnych adresów pocztowych.
- ♦ `inet_interfaces`: identyfikuje interfejsy sieciowe, na których serwer może odbierać pocztę. W tym przypadku serwer pocztowy będzie nasłuchiwał na wszystkich interfejsach sieciowych.

Kompletną listę parametrów konfiguracyjnych Postfiksa można znaleźć w podręczniku systemowym polecenia `postconf` (`man 5 postconf`) oraz w internecie, np. pod adresem <http://www.postfix.org/postconf.5.html>.

Identyfikacja zaufanych hostów i domen

Plik konfiguracyjny *main.cf*, omówiony w poprzednim punkcie, do pełnej funkcjonalności systemu wymaga tylko jednej poprawki: należy zmodyfikować parametr *mynetworks* w taki sposób, aby zawierał definicję przynajmniej jednej sieci, najczęściej lokalnej. Postfix w domyślnej instalacji w Ubuntu nasłuchuje na wszystkich interfejsach sieciowych, ale wysyłać pocztę pozwala jedynie hostom przyłączonym do wirtualnego interfejsu sieciowego *loopback*.

Parametr *mynetworks* może składać się z dowolnej liczby parametrów, które mogą przyjmować różne wartości: adres pojedynczego hosta lub adres IP z maską sieciową (definiującą zakres adresów IP podsieci). Wartość parametru *mynetworks* w pliku *main.cf* definiuje politykę pośredniczenia (*relay policy*) dla serwera SMTP. Wartość ta musi występować, inaczej serwer będzie otwartym pośrednikiem (*open relay*), czyli serwerem, który może być użyty do przesyłania poczty przez każdego użytkownika internetu. Otwarte serwery pośredniczące SMTP są powszechnie wykorzystywane przez spamerów, ponieważ można je wykorzystywać bez żadnych ograniczeń.

W naszym przypadku serwer poczty będzie obsługiwał przesyłanie wiadomości od wszystkich systemów w niewielkim biurze. Hosty mają tu adresy z podsieci *192.168.6.**, zatem zastosuję następującą wartość parametru *mynetworks*:

```
mynetworks = 127.0.0.0/8, 192.168.6.0/8
```

Po dokonaniu zmiany należy ponownie uruchomić serwer Postfix (`sudo /etc/init.d/postfix restart`) lub załadować zmodyfikowaną konfigurację bez zatrzymywania serwera (`sudo postfix reload`). Od tego momentu pocztę mogą wysyłać wszystkie hosty w z naszej sieci lokalnej.

Przepisywanie adresów w poczcie wychodzącej

Wszystkie hosty w sieci lokalnej mogą teraz wysyłać pocztę przez nasz nowy serwer, ale wiele MUA wysyła pocztę z użyciem w pełni kwalifikowanych nazw domenowych hostów, z których była wysłana poczta. Dlatego poczta wysyłana przez użytkownika *wvh* z hosta *64bit.vonhagen.org* ma adres nadawcy ustawiony jako *wvh@64bit.vonhagen.org*. Mnie to nie przeszkadza, prawdopodobnie w tym systemie nie mam skrzynki pocztowej, przez co odbiorcy takich e-maili nie będą mogli na nie odpowiedzieć.

Aby naprawić ten problem, możemy wykorzystać parametr konfiguracyjny Postfiksa *masquerade_domains*, który identyfikuje domenę, na jaką będą przepisywane adresy nadawcy. W tym przypadku poczta z adresu *wvh@64bit.vonhagen.org* zostanie zmodyfikowana w taki sposób, aby adresem nadawcy był *wvh@vonhagen.org*, który jest moim właściwym adresem. W tym przypadku muszę ustawić następującą wartość parametru *masquerade_domains*:

```
masquerade_domains = vonhagen.org
```

Po dokonaniu zmiany należy ponownie uruchomić serwer Postfix (`sudo /etc/init.d/postfix restart`) lub załadować zmodyfikowaną konfigurację bez zatrzymywania serwera (`sudo postfix reload`). Od tego momentu poczta wysyłana przez nasz nowy serwer będzie miała prawidłową nazwę domeny nadawcy.

Przyjmowanie poczty dla całej domeny

Jak wspomniałem wcześniej, parametr `mydestination` identyfikuje adresy hostów, dla których poczta będzie dostarczana lokalnie. Domyślna konfiguracja Postfixa akceptuje pocztę dla `localhost.localdomain`, `localhost`, rzeczywistej domeny serwera oraz dla domeny zdefiniowanej na etapie konfiguracji serwera w charakterze jego nazwy publicznej (patrz rysunki 27.3 i 27.6). Aby nasz nowy serwer Postfix mógł działać jako serwer docelowy dla poczty wysyłanej do określonej domeny, należy nazwę domeny dopisać do listy `mydestination` w pliku konfiguracyjnym `main.cf`. Na nazwę serwera poczty wybrałem `mail.vonhagen.org`, dlatego parametr `mydestination` będzie miał następującą zawartość:

```
mydestination = mail.vonhagen.org, ubuntu32.vonhagen.org,  
               localhost.vonhagen.org, localhost
```

Do tej listy mogę dodać moją nazwę domeny lub ustawić wartość parametru `mydomain` i wykorzystać ją w ramach definicji `mydestination`. Dzięki temu w przypadku zmiany domeny wartość `mydestination` będzie nadal aktualna bez dodatkowych modyfikacji:

```
mydomain = vonhagen.org  
mydestination = mail.vonhagen.org, ubuntu32.vonhagen.org,  
               localhost.vonhagen.org, localhost, $mydomain
```

Podobnie jak w przypadku innych zmian konfiguracji, w celu ich zastosowania należy ponownie uruchomić serwer Postfix (`sudo /etc/init.d/postfix restart`) albo załadować zmiany bez ponownego uruchamiania serwera (`sudo postfix reload`).



Wprowadzając zmiany sugerowane w tym punkcie, warto ustawić wartość parametru `myorigin` na `$mydomain`. Podczas instalacji parametr `myorigin` został ustawiony na nazwę systemu, która została również użyta jako wartość parametru `mydestination`, co w tym przypadku jest użyteczne, gdy ktoś wyśle wiadomość na adres typu `root@mail.vonhagen.org`. W praktyce, po instalacji, lepiej jest gdy parametr `mydomain` ma tę samą wartość co `myorigin`.

Uaktywnienie obsługi POP/POP3 za pomocą pakietu Qpopper

Qpopper (<http://www.eudora.com/products/unsupported/qpopper/index.html>) jest popularnym serwerem POP/POP3 na licencji open source, opracowanym przez firmę Qualcomm, twórców Eudory, popularnej aplikacji typu MUA dla pecetów i macintoshy. Lubię program Qpopper, ponieważ obsługuje klasyczny dla Linuksów i Uniksoów format `mbox`, wykorzystywany również przez Postfix w standardowej konfiguracji Ubuntu. Format `mbox` przechowuje wszystkie wiadomości w pojedynczym pliku, początek wiadomości rozpoznaje się po ciągu znaków typu `From użytkownik@domena`, po którym następuje znacznik czasu wysyłki. Na etapie instalacji Postfixa zasugerowałem również instalację pakietu Qpopper. Jeśli ktoś tego nie zrobił, a planuje udostępniać wiadomości pocztowe za pomocą protokołu POP/POP3, powinien wykonać to teraz i przejść do kolejnego punktu rozdziału.



Jak wspomniałem wcześniej, POP/POP3 i IMAP stanowią zamienne protokoły dostępu do poczty. Oba standardy mogą być używane z Postfiksem pod warunkiem, że na serwerze poczty zostanie zainstalowany odpowiedni serwer obsługujący te protokoły. Qpopper jest szybki, lekki i ma solidne wsparcie. Istnieją jednak rozwiązania, które pozwalają na obsługę obu (POP/POP3 i IMAP) standardów jednocześnie, są to Cyrus IMAP (<http://asg.Web.cmu.edu/cyrus/>) oraz Courier IMAP (www.courier-mta.org/imap/); oba są dostępne w repozytoriach Ubuntu. Konfiguracja serwera Cyrus IMAP jest dość skomplikowana, natomiast serwer Courier IMAP obsługuje jedynie format maildir, w którym wiadomości są zapisywane w pojedynczych plikach, co w przypadku Postfiksa w Ubuntu wymaga dodatkowych zabiegów konfiguracyjnych. Dlatego zdecydowałem się na serwer Qpopper — szybki i prosty, który „po prostu działa”.

W pakiecie Qpopper w Ubuntu znajduje się gotowy do użycia demon Qpopper przystosowany do użycia z wykorzystaniem demonów `inetd` lub `xinetd`, omówionych w podpunkcie rozdziału 18., „Konfigurowanie serwera `svnserv` do obsługi Subversion”. Każdy z tych demonów służy do uruchamiania usług w odpowiedzi na nadchodzące żądania. Jeśli ktoś nie zainstalował jednego z tych pakietów, sugeruję, aby jak najszybciej to zrobić. Demon Qpopper może być z łatwością zintegrowany z `xinetd` przez utworzenie pliku `/etc/xinetd.d/pop3` o następującej zawartości:

```
service pop3
{
    port                = 110
    socket_type         = stream
    protocol            = tcp
    wait                = no
    user                = root
    server              = /usr/sbin/in.qpopper
}
```

Następnie należy ponownie uruchomić demon `xinetd`, wywołując polecenie `sudo /etc/init.d/xinetd restart`, co spowoduje uaktywnienie demona Qpopper w systemie. Działanie demona Qpopper można zweryfikować, konfigurując program MUA do odbierania poczty z użyciem protokołu POP/POP3. Można też posłużyć się wierszem poleceń i programem `telnet`, łącząc się na porcie 110 (wykorzystywanym przez protokoły POP/POP3) i sprawdzając, czy uda się zalogować:

```
$ telnet mail.vonhagen.org 110
Trying 192.168.6.182...
Connected to 192.168.6.182.
Escape character is '^J'.
+OK Qpopper (version 4.0.5) at mail starting. <28861.1153954154@ulive>
user wvh
+OK Password required for wvh.
pass *****
+OK wvh has 1 visible message (0 hidden) in 696 octets.
quit
+OK Pop server at ulive signing off.
Connection closed by foreign host.
```

Jak widać, Qpopper zareagował prawidłowo na żądanie POP/POP3, zaakceptował nazwę użytkownika i hasło oraz wypisał liczbę wiadomości oczekujących na serwerze. Brawo!

Konfiguracja mechanizmów antyspamowych i antywirusowych

Po skonfigurowaniu i uruchomieniu serwera poczty oraz zweryfikowaniu, że działają połączenia ze zdalnych serwerów, szybko okaże się, że do serwera przychodzi mnóstwo spamu (określanego również jako UCE, ang. *Unsolicited Commercial E-mail*, czyli niepożądana poczta komercyjna). Niektóre ze spamów mogą również zawierać załączniki zakażone wirusami. Może się też okazać, że otrzymamy wirus od znajomej osoby, przesyłającej w dobrej wierze zainfekowaną ciekawostkę. Jeśli miałeś szczęście, i ktoś kto obsługiwał twoją pocztę odfiltrowywał większość niechcianych wiadomości zanim postanowiłeś przenieść pocztę na własny serwer, to nie zdajesz sobie sprawy co złego dzieje się w krainie e-maili. Teraz, kiedy posiadasz własny serwer pocztowy szybko zrozumiesz, że SPAM i zainfekowane maile przybrały rozmiar epidemii. Nic dziwnego, że łączenie z Amazon.com trwa ostatnio tak długo....

Nie wystarczy poznanie skali epidemii spamu i wirusów poczty elektronicznej, należy uzmysłowić sobie, że fakt utrzymania serwera pocztowego narzuca również konieczność ochrony użytkowników przed tego typu śmieciami pocztowymi. Na szczęście, istnieją doskonałe mechanizmy, w które można uzbroić Postfix, aby radził sobie z tego typu problemami. Jeśli użytkownik zainstalował wszystkie pakiety, o których była mowa w podrödziale „Instalacja Postfiksa i pakietów pomocniczych”, mechanizm *postgrey* i *MailScanner* działają już w systemie. *MailScanner* skanuje pocztę przychodzącą i odrzuca zidentyfikowany spam, potrafi też uruchamiać inne pakiety, takie jak *SpamAssassin* (<http://spamassassin.apache.org>), czyli narzędzie, które jeszcze lepiej radzi sobie z wykrywaniem spamu, czy *ClamAV* (<http://www.clamav.net>), doskonały pakiet antywirusowy na licencji open source. Pozostaje tylko zintegrować zainstalowane pakiety z Postfiksem.

Można się zastanawiać, po co komu program antywirusowy pod Linuksem, skoro ten system jest praktycznie odporny na tego typu zakażenia. Proponuję instalację programu antywirusowego, ponieważ niektórzy użytkownicy serwera pocztowego mogą do czytania poczty korzystać z systemu operacyjnego Microsoft Windows wykorzystywanego inaczej do grania. A poważniej, serwer pocztowy, czy to w domu, firmie czy też korporacji może być wykorzystywany przez użytkowników pracujących w systemie Microsoft Windows. Instalacja pakietu antywirusowego i integracja go z serwerem pocztowym jest praktycznym zastosowaniem reguły „lepiej zapobiegać, niż leczyć”. Nic nie kosztuje, a może oszczędzić mnóstwa problemów związanych z koniecznością oczyszczenia komputerów z wirusów.

W kolejnych trzech punktach omawiam różne mechanizmy zabezpieczenia poczty elektronicznej przed spamem i wirusami. W pierwszym opisuję mechanizm *greylisting* implementowany przez pakiet *postgrey*. W drugim omawiam kilka opcji Postfiksa, o jakie można uzupełnić plik *main.cf*, co spowoduje natychmiastowe odrzucenie spamu i oszczędzi programowi *MailScanner* znacznej ilości pracy. W trzecim przedstawiam sposób integracji mechanizmu *MailScanner* z serwerem Postfix i programem *ClamAV* w celu identyfikacji jak największej ilości SPAM-u pochodzącego z poprawnie skonfigurowanych serwerów wraz z jednoczesną kontrolą antywirusową.

Obsługa mechanizmu greylisting za pomocą pakietu postgrey

W podrozdziale „Instalacji Postfixa i pakietów pomocniczych” zasugerowałem, aby zainstalować pakiet *postgrey*, zawierający aplikację obsługującą mechanizm szarych list (*greylisting*), zaprojektowaną do współpracy z Postfixem. Ten pakiet jest skonfigurowany i uruchamiany w sposób automatyczny i nie wymaga żadnej dodatkowej konfiguracji; jeśli został zainstalowany wraz z serwerem Postfix, będzie działał od razu.

Jak wspomniałem w podrozdziale poświęconym instalacji, mechanizm szarych list to ciekawa technika polegająca na odrzucaniu wszystkich wiadomości przychodzących z zewnętrznych źródeł (do serwera SMTP przesyłany jest kod błędu tymczasowego). Większość spamerów podejmuje tylko jedną próbę dostarczenia poczty, po czym przechodzi do następnej „ofiary”: przy gigantycznej ilości rozsyłanego spamu nie mogą sobie pozwolić na ponawianie prób dostarczenia poczty. Wszystkie prawdziwe serwery SMTP (zgodne ze specyfikacją RFC) po otrzymaniu kodu błędu tymczasowego powinny po określonym czasie oczekiwania ponowić próbę dostarczenia poczty, która tym razem będzie bez problemów zaakceptowana przez serwer Postfix. W tym momencie zastosowanie znajdują kolejne techniki detekcji spamu.



Określenie *szara lista* związane jest z dwiema innymi technikami stosowanymi do zwalczania spamu. *Czarne listy* to technika stosowana do odrzucania poczty pochodzącej ze znanych źródeł spamu. *Biała lista* natomiast jest mechanizmem akceptacji poczty pochodzącej z wiarygodnych źródeł. Fakt, że administratorzy muszą uciekać się do tego typu sztuczek, aby oszczędzić użytkownikom kłopotów związanych z ogromną ilością niechcianej poczty, stanowi rzeczywiste świadectwo niskiego poziomu wielu użytkowników internetu, ale takie są — niestety — realia współczesnego świata.

Choć *postgrey* nie wymaga dodatkowej konfiguracji, istnieje kilka opcji jego wywołania, które nie są aktywne domyślnie, a które mogą okazać się przydatne podczas dostosowywania zachowania mechanizmu greylist. Można np. zdefiniować białą listę nadawców, wpisując ich definicje do odpowiedniego pliku w katalogu `/etc/postgrey`. Przykładowo plik `/etc/postgrey/whitelist_clients` identyfikuje adresy hostów, które nie będą dodawane do szarej listy, czyli wszelkie przychodzące od nich wiadomości pocztowe będą automatycznie akceptowane. Podobnie plik `/etc/postgrey/whitelist_recipients` identyfikuje użytkowników serwera poczty, dla których również nie będzie wykonywana operacja szarej listy. E-maile kierowane do tych odbiorców będą automatycznie przyjmowane przez serwer, niezależnie od tego, z jakiego hosta pochodzą. Więcej informacji na temat opcji programu *postgrey* i możliwości jego dostosowania do własnych wymagań można znaleźć w jego podręczniku systemowym.

Parametry serwera Postfix służące do odrzucania niechcianej poczty

Jak wspomniałem w punkcie „Pliki konfiguracyjne Postfixa”, administratorzy tego serwera mają do dyspozycji ogromną liczbę opcji, z których w domyślnej konfiguracji pliku `main.cf` w Ubuntu wykorzystywany jest zaledwie niewielki podzbiór. To dobry znak, ponieważ pozwala mieć nadzieję, że wartości domyślne są właściwe w większości zastosowań. Jednak istnieje kilka parametrów, które dają dodatkowe możliwości pomagające

serwerowi Postfix w odrzucaniu spamu dzięki weryfikacji cech wiadomości niezgodnych ze standardami (zdefiniowanymi w dokumentach RFC). W kolejnym punkcie, „Integracja programów MailScanner, SpamAssassin i ClamAV z serwerem Postfix”, wyjaśniam, w jaki sposób filtrować treści poczty przychodzącej. Skanowanie poczty zajmuje określony czas, co wprowadza pewne opóźnienia w procesie dostarczania poczty. Z tego powodu każda wiadomość pocztowa odrzucona, zanim zostanie przetworzona przez filtry, stanowi zysk poprawiający wydajność serwera.

Poniższy zbiór reguł dopisany w pliku *main.cf* może okazać się bardzo efektywnym mechanizmem redukcji spamu (numery wierszy służą jedynie zwiększeniu czytelności, nie należy ich przepisywać do pliku *main.cf*):

```
1 smtpd_delay_reject = yes
2 smtpd_helo_required = yes
3 smtpd_helo_restrictions =
4   permit_mynetworks,
5   reject_invalid_hostname,
6   reject_non_fqdn_hostname,
7   permit
8 smtpd_data_restrictions =
9   reject_unauth_pipelining,
10  permit
11 smtpd_sender_restrictions =
12  permit_mynetworks,
13  reject_non_fqdn_sender,
14  reject_unknown_sender_domain,
15  permit
16 smtpd_recipient_restrictions =
17  permit_mynetworks,
18  reject_unknown_recipient_domain,
19  reject_unauth_destination,
20  permit
```

Wiersz 1. definiujący parametr *smtpd_delay_reject* (odłożenie w czasie odrzucenia wiadomości) jest niezbędny do działania poniższych reguł. Jego działanie polega na tym, że połączenie nie zostanie zerwane od razu, nawet w przypadku, gdy będzie miała zastosowanie reguła nakazująca takie działanie. Chodzi o to, że niektóre serwery SMTP zgłaszają błędy, gdy nie osiągną określonego stanu komunikacji, należy zatem doprowadzić połączenie SMTP do stanu, gdy jego zerwanie nie będzie powodować konfliktów. Wiersz 2. odrzuca pocztę pochodzącą od nadawców, którzy nie zidentyfikują się za pomocą prawidłowej instrukcji HELO lub EHLO, co jest zdefiniowane w RFC protokołu SMTP. Jeśli zdalny serwer nie spełni tego wymogu połączenie zostanie przerwane i dalsze reguły filtrujące wiadomości nie muszą być przetwarzane.

Wiersze od 3. do 7., od 8. do 10., od 11. do 16. i od 16. do 20. definiują łańcuchy reguł stosowane do różnych elementów wiadomości pocztowych na kolejnych etapach interakcji serwera SMTP z klientem (serwerem zdalnym).

- ♦ Wiersze od 3. do 7. identyfikują ograniczenia wykorzystania instrukcji HELO i EHLO, które zdalny serwer stosuje do identyfikacji. Wiersz 4. powoduje, że automatycznie akceptowane są serwery zdefiniowane w parametrze *mynetworks*. Wiersz 5. powoduje odrzucenie połączeń od zdalnych serwerów identyfikujących się z użyciem nieprawidłowej nazwy hosta. Wiersz 6. odrzuca połączenia od zdalnych serwerów

poczty, które nie zidentyfikowały się z zastosowaniem w pełni kwalifikowanej nazwy domenowej. Wiersz 7. powoduje, że serwer, który poprawnie przeszedł poprzednie sprawdzenia, zostanie chwilowo zaakceptowany, tzn. dopuszczony do weryfikacji w ramach kolejnego łańcucha kryteriów.

- ♦ Wiersze od 8. do 10. identyfikują ograniczenia w oparciu o sposób przysyłania żądań do serwera. Wiersz 9. powoduje odrzucenie połączenia, w którym klient wysyła żądania ze zbyt dużą prędkością, a w każdym razie zanim nasz serwer poinformuje serwer zdalny o możliwości przesłania szybkiego strumienia żądań SMTP. Wiele serwerów spammerskich wykorzystuje tę opcję do przyspieszenia przysyłania wiadomości: w końcu mają biliony wiadomości do przesłania i nie ma sensu marnować czasu (chyba że to nie ich czas). Wiersz 10. spowoduje, że serwer, który poprawnie przeszedł poprzednie sprawdzenia, zostanie chwilowo zaakceptowany, tzn. dopuszczony do weryfikacji w ramach kolejnego łańcucha kryteriów.
- ♦ Wiersze od 11. do 15. identyfikują ograniczenia nadawcy wiadomości pocztowej (rozpoznawanego w oparciu o żądanie SMTP MAIL FROM). Wiersz 12. powoduje, że automatycznie zostaną zaakceptowane wiadomości od każdego serwera wymienionego w parametrze `mynetworks`. Wiersz 13. odrzuci pocztę z adresów niezawierających w pełni kwalifikowanego adresu (wiadomości z sieci lokalnej mogą zawierać skrócone nazwy hostów, ale te już zostały zaakceptowane w wierszu 12.). Wiersz 14. odrzuca pocztę od nadawców z nieokreślonych domen. Wiersz 15. spowoduje, że serwer, który poprawnie przeszedł poprzednie sprawdzenia, zostanie chwilowo zaakceptowany, tzn. dopuszczony do weryfikacji w ramach kolejnego łańcucha kryteriów.
- ♦ Wiersze od 15. do 20. identyfikują ograniczenia adresata wiadomości (na podstawie instrukcji SMTP RCPT TO). Wiersz 17. powoduje, że automatycznie zostaną zaakceptowane wiadomości od każdego serwera wymienionego w parametrze `mynetworks`. Wiersz 18. odrzuca wiadomości, dla których aktualny serwer pocztowy nie jest serwerem docelowym oraz serwer docelowy nie jest prawidłową nazwą domenową. Wiersz 19. odrzuca wiadomości, gdy adresat nie należy do domen obsługiwanych przez ten serwer (identyfikowanych parametrem `mynetworks`), ani nie należy do domen, dla których bieżący serwer jest pośrednikiem (identyfikowanych parametrem `relayhost`). Te reguły zapobiegają wykorzystaniu serwera w charakterze otwartego pośrednika (`open relay`). Wiersz 20. spowoduje, że serwer, który poprawnie przeszedł poprzednie sprawdzenia, zostanie zaakceptowany, tzn. dopuszczony do procesu dostarczania poczty do adresatów.

Postfix oferuje wiele innych mechanizmów kontrolnych, których nie zastosowałem w tym przykładzie, np. sprawdzenie, czy nadchodząca wiadomość nie jest wysłana ze źródła będącego znanym nadawcą spamu (czarna lista). Z doświadczenia wiem, że tego typu sprawdzenia spowalniają serwer, a czarne listy są często niekompletne lub wręcz błędne. Często zdarza się, że na czarne listy trafiają adresy IP przydzielane w sposób dynamiczny przez dostawców internetu. Czasem takie przypadki wskazują rzeczywistych spamerów, jednak zwykle karę ponoszą niewinni użytkownicy, którzy otrzymają taki „skażony” adres IP.

Przykładowy zestaw konfiguracyjny omówiony w tym punkcie powinien wystarczyć do odfiltrowania pokażnej ilości spamu. Po eksperymentach administrator może zdecydować

się na uzupełnienie tej listy. Kompletną listę mechanizmów kontrolnych pomocnych w filtrowaniu spamu można znaleźć w podręczniku systemowym programu `postconf` (`man 5 postconf`). Ten sam dokument znajduje się na stronie WWW projektu Postfix: <http://www.postfix.org/postconf.5.html>.

Integracja programów MailScanner, SpamAssassin i ClamAV z serwerem Postfix

MailScanner to szybkie narzędzie o bardzo dużych możliwościach, służące do skanowania poczty przychodzącej w celu identyfikacji spamu, a można go dodatkowo zintegrować z pakietem antywirusowym. Pakiet programu MailScanner dla Ubuntu wymaga instalacji pakietu SpamAssassin (omówionego w punkcie „Automatyczne sprawdzanie niechcianej poczty” rozdziału 8.). Dzięki połączeniu siły obu pakietów, po wykonaniu filtrowania spamu przez MailScanner, wywoływany jest SpamAssassin, który wykonuje specyficzne dla siebie działania związane z detekcją spamu.



Pakiet MailScanner informuje o tym, że sprawdza wiadomości w poszukiwaniu wirusów, nawet wtedy, gdy nie jest zintegrowany z pakietem antywirusowym. Dzieje się tak dlatego, że w pliku konfiguracyjnym istnieją ustawienia skanera antywirusowego, który w domyślnej konfiguracji ma wartość `none` (brak). Niedługo napiszę, w jaki sposób usunąć tę niedogodność.

W podrozdziale „Instalacja Postfiksa i pakietów pomocniczych” zasugerowałem, aby wraz z Postfiksem zainstalować MailScanner i pakiet antywirusowy ClamAV. Jeśli ktoś nie zastosował się do tych zaleceń, powinien teraz nadrobić swoje niedopatrzenie.

Integracja pakietu MailScanner z serwerem Postfix jest już niezwykle prostym zadaniem.

1. Na końcu pliku `/etc/postfix/main.cf` należy wpisać następujący wiersz:

```
header_checks = regexp:/etc/postfix/header_checks
```

2. Teraz trzeba utworzyć plik `/etc/postfix/header_checks` o następującej zawartości:

```
/^Received:/ HOLD
```

Ten wiersz powoduje, że Postfix będzie szukał w każdej przychodzącej wiadomości ciągu znaków *Received*: na samym początku wiersza. Wiadomości spełniające ten warunek (a powinny to być wszystkie wiadomości) będą przenoszone do kolejki wstrzymania (*HOLD*). Kolejka wstrzymania to obszar tymczasowego zasobu poczty przychodzącej, wykorzystywany właśnie przez skanery. Zapisujemy plik `/etc/postfix/header_checks` i przechodzimy do kolejnego punktu.

3. W edytorze tekstu należy otworzyć plik konfiguracyjny programu MailScanner (`/etc/MailScanner/MailScanner.conf`) i wprowadzić zmiany niezbędne do poprawnej pracy z Postfiksem (w domyślnej konfiguracji w Ubuntu MailScanner jest skonfigurowany do współpracy z programem `exim`).
4. Teraz trzeba odszukać parametr `Run As User`, usunąć znak komentarza przed wierszem `Run As User = postfix` i zablokować znakiem komentarza pozostałe wpisy `Run As User`. W efekcie ten fragment pliku powinien mieć następującą postać:

```
# Run As User = mail
Run As User = postfix
# Run As User = Debian-exim
```

5. Następnie należy odszukać blok parametrów `Run As Group`, usunąć znak komentarza przed wierszem `Run As Group = postfix` i zablokować znakiem komentarza pozostałe wpisy `Run As Group`. W efekcie ten fragment pliku powinien mieć następującą postać:

```
# Run As Group = mail
Run As Group = postfix
# Run As Group = Debian-exim
```

6. Teraz należy odszukać parametr `Incoming Queue Dir` i zmienić jego wartość na `/var/spool/postfix/hold`. Ta pozycja powinna mieć następującą postać:

```
Incoming Queue Dir = /var/spool/postfix/hold
```

7. Przyszła kolej na odszukanie parametru `Outgoing Queue Dir` i zmianę jego wartości na `/var/spool/postfix/incoming`. Ta pozycja powinna mieć następującą postać:

```
Outgoing Queue Dir = /var/spool/postfix/incoming
```

8. Teraz trzeba odszukać parametr `MTA` i zmienić jego wartość na `postfix`. Ta pozycja powinna mieć następującą postać:

```
MTA = postfix
```

9. Następnie należy wyszukać parametr `Virus Scanners` i zmienić jego wartość na `clamav`. Ta pozycja powinna mieć następującą postać:

```
Virus Scanners = clamav
```

10. Trzeba zapisać plik `/etc/MailScanner/MailScanner.conf`.

11. Należy upewnić się, że właścicielem katalogu roboczego programu MailScanner jest użytkownik i grupa `postfix`, aby serwer Postfix mógł zapisywać w nich pliki. W tym katalogu MailScanner przechowuje wiadomości poddane kwarantannie, tzn. takie, które mogą być zainfekowane:

```
sudo chown -R postfix:postfix /var/spool/MailScanner
```

12. Następnie należy otworzyć w edytorze tekstu plik `/etc/default/mailscanner` i usunąć znak komentarza na początku wiersza definiującego zmienną `run_mailscanner`. Ta pozycja powinna mieć następującą postać:

```
run_mailscanner=1
```

Uaktywnienie tej zmiennej spowoduje automatyczne uruchomienie programu MailScanner przy następnym uruchomieniu systemu.

Następnie trzeba uruchomić program MailScanner, wywołując następujące polecenie:

```
sudo check_mailscanner
```

Teraz należy uruchomić serwer Postfix (`sudo /etc/init.d/postfix restart`) albo ponownie załadować jego plik konfiguracyjny bez ponownego uruchamiania (`sudo postfix reload`).

Aby sprawdzić, czy MailScanner został prawidłowo zintegrowany z serwerem poczty, należy wysłać do siebie wiadomość za pośrednictwem tego serwera. Jeśli MailScanner działa prawidłowo (a wiadomość nie została uznana za spam i nie zawiera wirusa), u dołu wiadomości powinna pojawić się następująca informacja:

This message has been scanned for viruses and dangerous content by MailScanner, and is believed to be clean.

Gratulacje! Oznacza to, że system jest solidnie zabezpieczony przed spamem i wirusami znajdującymi się w załącznikach do poczty przesyłanej przez ten serwer.



Jak wspominałem wcześniej, SpamAssassin jest wstępnie skonfigurowany tak, aby program MailScanner mógł z niego korzystać. W internecie można znaleźć wiele zestawów reguł rozszerzających skuteczność filtrowania programu SpamAssassin. Gdy uda się sprawdzić, że wstępna konfiguracja filtrów działa prawidłowo, warto zastanowić się nad udoskonaleniem zestawów reguł. Zachęcam do zapoznania się z kilkoma zasobami takimi jak *Exit0.us SpamAssassin wiki* (<http://www.exit0.us>), czy fora dyskusyjne *SpamAssassin Rules Emporium* (<http://www.rulesemporium.com/forums/>) i lista e-mailowa (<http://lists.maddoc.net/mailman/listinfo/sare-users>). Należy jedynie pamiętać, że każda dodatkowa reguła wydłuża czas niezbędny do sprawdzenia każdej wiadomości. Nie warto przesadzać, bo można doprowadzić do sytuacji, gdy serwer, przeciążony filtrami antyspamowymi, nie będzie w stanie nadążyć z przetwarzaniem nadchodzących wiadomości.

Więcej informacji

W tym rozdziale zawarłem omówienie podstawowej konfiguracji prostego serwera poczty Postfix w systemie Ubuntu. Jeśli ktoś decyduje się utrzymywać własny serwer, powinien zakupić jedną z książek poświęconych serwerowi Postfix. Osobiście polecam poniższe pozycje, które znam i uważam za doskonałe zasoby wiedzy na ten temat.

- ♦ Richard Blum; *Postfix*, (Sams, 2001, ISBN: 0672321149): wykorzystuję tę książkę od lat i zawsze stanowiła doskonałe źródło podstawowych informacji. Niestety, to dość stara pozycja, dlatego brakuje w niej informacji o parametrach konfiguracyjnych, które pojawiły się po roku 2001.
- ♦ Kyle D. Dent, Wietse Venema; *Postfix. Przewodnik encyklopedyczny*, (Helion, 2004, ISBN: 83-7361-668-3): trudno znaleźć bardziej kompletne źródło informacji o programie Postfix, niż ta książka wydawnictwa O'Reilly, m.in. dlatego, że jednym z jej autorów jest twórca Postfiksa, Wietse Venema.
- ♦ Ralf Hildebrandt, Patrick Koetter; *Postfix. Nowoczesny system przesyłania wiadomości*, (Helion 2006, ISBN: 83-246-0145-7): Ralf Hildebrandt jest znanym ekspertem od Postfiksa, jego strona WWW Postfix Shrine jest jednym z doskonałych zasobów informacji o tym serwerze (szczególnie cennym dla osób mówiących po niemiecku, ponieważ część informacji tam zawartych jest napisana w języku niemieckim). Książka jest nawet lepsza, a jej zaleta polega na tym, że została przetłumaczona na język polski.

W przypadku wszystkich zagadnień związanych z technologią, często najlepszym źródłem informacji jest wyszukiwarka internetowa. Wystarczy wpisać komunikat o błędzie, hasło związane z problemem konfiguracyjnym czy inne słowo kluczowe, aby uzyskać mnóstwo adresów, które warto odwiedzić.

Podsumowanie

Poczta elektroniczna to jedna z kluczowych technologii współczesnego internetu, a posiadanie własnego serwera poczty to standard w wielu korporacjach i środowiskach akademickich; reguła ta staje się powszechna również w sieciach domowych. Pełna kontrola nad serwerem pocztowym jest wielką zaletą w większości środowisk informatycznych, bo pozwala na zredukowanie lub wręcz wyeliminowanie spamu i ataków na systemy z użyciem wirusów i koni trojańskich.

W tym rozdziale przekazałem podstawowe informacje na temat najpopularniejszych systemów pocztowych dla Linuksa, skupiając się na instalacji i konfiguracji serwera Postfix. Wyjaśniłem również, w jaki sposób w standardowej instalacji Postfixa zintegrować usługi dodatkowe, takie jak filtrowanie spamu czy ochrona antywirusowa. Postfix to serwer pocztowy o wielkich możliwościach, z którym można łatwo, ale solidnie połączyć dodatkowe usługi dostępne na licencji open source.

W rozdziale 28. omówię instalację i konfigurację serwera DHCP (ang. *Dynamic Host Control Protocol*), który pozwala na zarządzanie adresami IP w sieci i kontrolę ich wykorzystania. Dynamiczne przydzielanie adresów ułatwia zarządzanie niewielką pulą adresów IP nawet przy znacznie większej puli stacji roboczych, jak również upraszcza zarządzanie konfiguracją sieci, która jest przekazywana klientom w sposób automatyczny. Wykorzystanie serwera DHCP upraszcza wszelkie modyfikacje konfiguracji sieciowych, ponieważ dane te są zapisywane na serwerze tylko w jednym miejscu, skąd są pobierane przez klientów usługi DHCP.

Rozdział 28.

Konfiguracja serwera DHCP

W tym rozdziale:

- ♦ Ogólne informacje o DHCP
- ♦ Konfiguracja serwera DHCP w trybie tekstowym
- ♦ Konfiguracja serwera DHCP w trybie graficznym
- ♦ Usuwanie problemów z DHCP

Dawno temu, gdy byłem chłopcem, internet (a właściwie ARPANET, jak się wówczas nazywał) był niewielką siecią połączonych wzajemnie hostów, z których każdy posiadał stały adres IP, a ich nazwy były zapisane w pliku przekazywanym od jednego administratora do drugiego. Każdy z nich dopisywał w tym pliku własne, lokalne modyfikacje dotyczące hostów i sieci. Wraz z rozwojem, ARPANET powiększał się o nowe hosty, a utrzymanie tak dużej ilości informacji w pojedynczym pliku stało się tyle niepraktyczne, co niepoważne. Ta sytuacja doprowadziła do opracowania elastycznego, programistycznego rozwiązania pod nazwą DNS (ang. *Domain Name System*), które pozwala identyfikować nazwę hosta z powiązaniem adresem IP i vice versa. Więcej informacji na temat mechanizmu DNS można znaleźć w rozdziale 29., „Konfiguracja serwera DNS”.

Dlaczego rozdział poświęcony mechanizmowi DHCP (ang. *Dynamic Host Configuration Protocol*) rozpoczynam od wspomnienia o systemie DNS? Ponieważ DHCP i jego poprzedników opracowano z myślą o osiągnięciu elastyczności i skalowalności przez podobne usługi oferujące różne rodzaje informacji w sieci. Gdy w firmie lub instytucji do sieci przyłączonych jest niewiele stacji i wszystkie należą do wspólnej podsieci, przydzielanie im statycznych adresów IP i utrzymanie informacji o adresach oraz odwzorowanie nazwy domenowej na adres IP nie jest zadaniem zbyt wyczerpującym. Jednak większa i przede wszystkim zmienna liczba hostów, z częstymi zmianami w sieci wymagałaby od działu IT znacznych nakładów pracy na ręczne konfiguracje każdej stacji, co wkrótce stałoby się uciążliwe. Laptopy przyłączane do sieci to typowy przykład stacji pracujących w różnych lokalizacjach, z których każda może mieć odmienną konfigurację.

W tym rozdziale przekażę nieco informacji na temat mechanizmu DHCP i jego poprzedników, problemów, które musieli rozwiązać programiści, a na końcu opiszę konfigurację

własnego serwera DHCP oraz poruszę zagadnienia, jakie należy wziąć pod uwagę przy usuwaniu błędów. Skupię się na serwerach DHCP w tradycyjnych sieciach IPv4, ale podam też kilka uwag dotyczących IPv6.

Ogólne informacje o DHCP

DHCP to skrót od *Dynamic Host Configuration Protocol*, czyli protokół dynamicznej konfiguracji hostów. Co to oznacza dla użytkowników sieci? To, że w sytuacji, gdy wszystko działa prawidłowo, użytkownik może podłączyć swój komputer do gniazda w ścianie za pomocą przewodu ethernetowego, a serwer zarządzający siecią na uczelni rozpozna komputer i skonfiguruje go do pracy.

Mechanizm DHCP jest wykorzystywany do przekazywania informacji o konfiguracji sieci przyłączającym się do niej komputerom skonfigurowanym do dynamicznego przydzielania adresów IP. Te systemy, nazywane klientami DHCP, lokalizują system serwera DHCP i otrzymują od niego informacje o konfiguracji sieci.

Protokół DHCP został opracowany przez *Dynamic Host Configuration Working Group* w ramach *Internet Engineering Task Force* (IETF). Mechanizm ten powstał początkowo jako realizacja dokumentu *Request for Comments* (RFC) o numerze 1531, z którego powstał RFC 1533 (z którego powstał RFC 2132), z którego powstał RFC 1534, z którego powstał RFC 1541, z którego powstał RFC 2131 itd. Podobnie jak rzecz ma się z większością dokumentów RFC, ich lektura pozwala bliżej poznać proces rozwoju koncepcji i rozumowania prowadzącego do opracowania ważnych technologii, które stają się codzienną i nieodłączną częścią sieci lokalnych i internetu. Dokumenty RFC można znaleźć na stronie <http://www.ietf.org/rfc.html> lub (nieco przyjaźniejszej dla użytkownika) <http://www.faqs.org/rfcs>. Niektóre z nich to prawdziwa klasyka, tak jak RFC 1149, 2549 itp.

Protokół DHCP jest oparty na wcześniejszym protokole, służącym do rozruchu komputerów przez sieć o nazwie BOOTP i posiada szereg części wspólnych, np. hosty uruchamiające się za pośrednictwem BOOTP mogą odczytywać wymaganą informację z prawidłowo skonfigurowanego serwera DHCP. Różnica między tymi dwoma protokołami polega na tym, że BOOTP wysyła określone dane konkretnym hostom, natomiast DHCP jest systemem bardziej elastycznym, odpowiadającym na żądania dowolnych klientów. DHCP dodatkowo pozwala na realokację adresów IP hostów, które już mają przydzielone adresy, oraz na odzyskiwanie adresów sieciowych, niewykorzystywanych przez ustalony okres czasu.

Klient DHCP lokalizuje i pobiera informację o konfiguracji sieci z serwera DHCP za pośrednictwem serii zdarzeń (ten przykład jest uproszczony, ponieważ nie bierze pod uwagę hostów pośredniczących DHCP). Oto ona.

1. Klient wysyła żądanie DHCP *discover* na adres rozgłoszeniowy 255.255.255.255 (*broadcast*). To żądanie zawiera informacje unikalnie identyfikujące klienta, m.in. jego adres MAC (ang. *Media Access Control*).
2. Serwery DHCP, które odbiorą to żądanie, wysyłają na adres rozgłoszeniowy propozycje adresów IP w postaci komunikatów-ofert DHCP. Zaoferowany adres IP jest konstruowany w oparciu o interfejs sieciowy oraz segment sieci, z którego

pochodzi żądanie, i może być adresem IP przypisanym do określonego adresu MAC lub jednym z adresów należących do puli zdefiniowanej dla danego segmentu sieci. Adresy IP, których serwer DHCP nie może przydzielić, to adresy już użyte lub zarezerwowane dla klientów o określonych adresach MAC.

Gdy zostaje wysłana propozycja adresu IP, serwer tymczasowo rezerwuje ten adres, aby nie zaoferować go dwukrotnie. Informacja wysłana do klienta zawiera również dane na temat innych parametrów sieci zdefiniowanych na serwerze.

3. Klient DHCP wybiera najlepszą ofertę z otrzymanych od serwerów opierając się na liczbie usług zawartych w komunikatach, po czym wysyła na adres rozgłoszeniowy komunikat zawierający adres IP, który zaakceptował. Ten komunikat służy również do tego, aby pozostałe serwery (których oferty zostały odrzucone) zdjęły rezerwacje z proponowanych adresów IP.
4. Serwer DHCP zidentyfikowany w drugim komunikacie alokuje adres IP zarezerwowany dla klienta i wpisuje ten adres IP do bazy zaalokowanych adresów IP wraz ze znacznikiem czasu ważności dzierżawy. Serwer DHCP wysyła do klienta potwierdzenie (ACK), zawierające pozostałe oferowane parametry konfiguracji sieciowej.
5. Klient DHCP wykorzystuje konfigurację sieci przesłaną przez serwer DHCP do skonfigurowania swoich interfejsów sieciowych, serwerów DNS itp. Klient również odnotowuje czas ważności adresu IP (dzierżawy) i rozpoczyna odliczanie, po zakończeniu którego wyśle do tego samego serwera żądanie jej przedłużenia. Gdy to nastąpi, serwer przedłuży dzierżawę pod warunkiem, że żądanie przedłużenia jest zgodne z polityką dzierżaw zdefiniowaną na serwerze.

Gdy serwer DHCP wysuwa żądanie udzielenia dzierżawy adresu IP, może wykorzystać zegar odliczający, może też, co jakiś czas, odpytywać serwer i sprawdzać, czy dzierżawa nie wygasa. Większość klientów DHCP w ramach procesu zamykania systemu informuje serwer DHCP, że już nie potrzebują dzierżawionych adresów IP. Klienci DHCP, którzy zamykają się w sposób nieprawidłowy (np. wskutek awarii), nie mają — oczywiście — możliwości poinformowania serwera DHCP. W tym przypadku przydzielony adres IP zostanie zwolniony po upływie czasu dzierżawy.

DHCP to bardzo zmyślny i użyteczny mechanizm, pomocny zarówno hostom wymagającym określonych ustawień konfiguracyjnych (identyfikowanych przez serwer DHCP, dzięki odpowiedniej konfiguracji), jak i różnym przypadkowym klientom, przyłączanym do segmentu sieci obsługiwanej przez serwer DHCP.



Wadą tak dużej elastyczności jest łatwość wystąpienia sytuacji problematycznych. Przykładowo serwer DHCP skonfigurowany w sposób błędny lub uruchomiony przez przypadek bez odpowiedniej konfiguracji może udostępniać klientom DHCP adresy IP i informacje o konfiguracji sieci zupełnie niezgodne z polityką obowiązującą w sieci. Z mojego doświadczenia wynika, że tego typu problemy najczęściej występują, kiedy do sieci, w której działa już serwer DHCP, jest przyłączany kolejny, o błędnej konfiguracji. W środowiskach korporacyjnych trudno znaleźć bardziej irytującą sytuację niż przyłączenie błędnie skonfigurowanego serwera DHCP. Wbrew pozorom, jest to dość łatwe, szczególnie w systemach, w których zainstalowałem kilka interfejsów Ethernet i zapomniałem zdefiniować w konfiguracji serwera DHCP, na którym interfejsie ma nasłuchiwać (w związku z czym nasłuchuje na wszystkich). Czy napisałem „zapomniałem”? Oczywiście, miałem na myśli „ktoś zapomniał”...

Jak dobrać okres dzierżawy?

Czas dzierżawy adresu IP dla klienta DHCP to delikatny kompromis między użytecznością a dostępnością. Chciałoby się, aby klienci skonfigurowani za pośrednictwem DHCP wykorzystywali zdobyty adres IP tak długo, jak jest on potrzebny, a jednocześnie aby był on jak najszybciej dostępny dla innych klientów sieci. Ma to szczególne znaczenie w sieciach, w których liczba potencjalnych użytkowników przekracza liczbę dostępnych adresów IP.

W doborze czasu dzierżawy użyteczna może okazać się obserwacja wzorców wykorzystania serwera DHCP przez użytkowników. Gdy liczba klientów przekracza liczbę dostępnych adresów IP, warto skonfigurować jak najkrótszy czas dzierżawy adresu. Z drugiej strony, warto minimalizować ruch sieciowy i obciążenie serwera DHCP. Jeśli liczba użytkowników jest znacznie większa niż liczba adresów IP i są to użytkownicy laptopów często przyłączający się i odłączający od sieci, warto skonfigurować jak najkrótszy czas dzierżawy. Jeśli systemy najczęściej przyłączają się do sieci na ustalony, w miarę stały okres (np. 30 minut, godzinę itp.), okres dzierżawy należy dostosować do tego czasu, aby zminimalizować częstotliwość odświeżania.

Inną rzeczą, którą warto wziąć pod uwagę, jest ilość czasu niezbędnego do przywrócenia systemu do sprawności po awarii sieci lub serwera DHCP. Przykładowo podczas poważnej awarii serwera DHCP dłuższy okres dzierżawy pozwoli utrzymać sieć w sprawności i da czas niezbędny na dokonanie naprawy.

Należy również wziąć pod uwagę „czynnik zwykłego użytkownika”. Jeśli klienci serwera DHCP uruchamiają usługi sieciowe, które mają być dostępne dla innych w ustalonym okresie, warto zadbać o to, aby dzierżawy DHCP były ważne przynajmniej przez ten okres. Mimo że dzierżawy DHCP są odświeżane po upływie ważności, możliwe są sytuacje, gdy taka operacja się nie powiedzie. Warto być na to przygotowanym. Co prawda, w przypadku gdy użytkownik zgłosi problem, można go z łatwością usunąć, zaliczając jeszcze jedną skuteczną interwencję, ale czasem najlepszy komentarz to brak komentarza.

Przedstawiłem koncepcję serwerów DHCP oraz to, jakie informacje udostępniają. Nadszedł czas, aby zainstalować taki serwer.

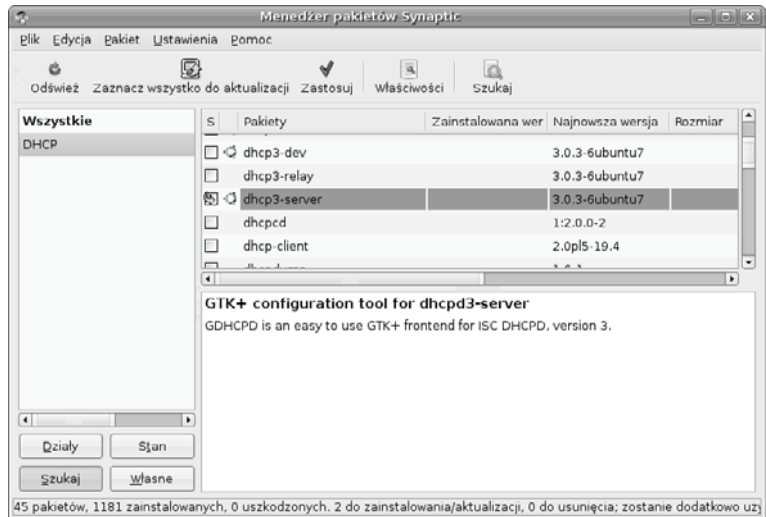
Instalacja serwera DHCP

Większość pakietów serwerowych nie jest instalowana z domyślną instalacją Ubuntu, ponieważ nie każdy chce (lub potrzebuje) je uruchamiać. Podobnie jak w przypadku innych pakietów dla Ubuntu, instalacja pakietów serwera DHCP jest bardzo prosta, można tego dokonać np. za pomocą menedżera pakietów Synaptic. Osobiście proponuję instalację serwera DHCP *Institutional Service Component* (ISC) v3, udostępnianego w postaci pakietu `dhcpcd3-server`. Jest to najnowsza i najlepsza wersja serwera DHCP i z pewnością oferująca największą funkcjonalność. Sugeruję również instalację pakietu `gdhcpd`, który stanowi łatwy w użyciu graficzny interfejs użytkownika upraszczający proces konfiguracji serwera DHCP.

Aby zainstalować pakiety za pomocą programów `apt-get` lub `aptitude` (bez użycia graficznego narzędzia instalacyjnego), można posłużyć się poleceniem `sudo apt-get install dhcpcd3-server` lub `sudo aptitude -r install dhcpcd3-server`.

Aby zainstalować te pakiety w trybie graficznym, należy uruchomić menedżer pakietów Synaptic: *System/Administracja/Menedżer pakietów Synaptic*. Po wpisaniu hasła zostanie wyświetlone okno programu Synaptic, w którym należy kliknąć przycisk *Szukaj*, co spowoduje wyświetlenie okna wyszukiwania. Trzeba upewnić się, że zaznaczona jest opcja *Opis i nazwa*, w polu tekstowym wpisać ciąg znaków `DHCP`, po czym kliknąć przycisk *Szukaj*. Po zakończeniu wyszukiwania należy odszukać pakiet *dhcp3-server*, kliknąć jego nazwę prawym przyciskiem myszy i z menu wybrać funkcję *Zaznacz do instalacji*. Na rysunku 28.1 przedstawiam pakiet *dhcp3-server* zaznaczony do instalacji.

Rysunek 28.1.
Instalacja serwera DHCP i pakietów pomocniczych



Po zaznaczeniu do instalacji pakietu *dhcp3-server* należy odszukać pakiet *gdhcpd*, który jest graficznym narzędziem służącym do konfiguracji serwera DHCP. Trzeba kliknąć prawym przyciskiem myszy nazwę pakietu i z menu wybrać funkcję *Zaznacz do instalacji*.

Po zaznaczeniu obu pakietów należy kliknąć ikonę *Zastosuj* znajdującą się w pasku narzędzi programu Synaptic, co rozpocznie instalację. Po jej zakończeniu można zamknąć okno menedżera pakietów Synaptic. Serwer jest prawie gotowy do działania.

Zarządzanie serwerem DHCP z wiersza poleceń

Zawsze znajdują się chętni do walki, którzy nie chcą się przystosować, innymi słowy tacy, którzy preferują tradycyjne metody konfiguracji serwera DHCP za pomocą edytora w trybie znakowym zamiast wykorzystania wygodnych, graficznych narzędzi konfiguracyjnych. Nie ma problemu, we wprawnych (i odpowiedzialnych) rękach edytor tekstu stanowi znacznie szybszy sposób konfiguracji. W kilku kolejnych punktach omówię konfigurację serwera DHCP z wiersza poleceń wraz z jego najważniejszymi opcjami.

Alternatywne serwery DHCP dla Ubuntu

Repozytoria Ubuntu, oprócz wspomnianego, zawierają też kilka innych serwerów DHCP. Może okazać się, że jeden z alternatywnych pakietów będzie lepiej nadawać się do określonych potrzeb niż ten, którego instalację zalecam. Dzięki temu, że DHCP jest standardem, wszystkie serwery DHCP wykorzystują te same polecenia konfiguracyjne. Mogą posiadać różne składnie plików konfiguracyjnych, ale polecenia zawarte w tych plikach są standardem dla wszystkich odmian serwerów DHCP. Oto lista opcjonalnych serwerów DHCP dostępnych dla Ubuntu.

- ♦ **dhcp**: wersja 2. serwera ISC DHCP. Tę wersję warto wziąć pod uwagę, jeśli chcemy w sposób automatyczny uzupełniać w serwerze DNS dane o hostach, których informacja IP została świeżo przypisana za pomocą mechanizmu DHCP.
- ♦ **dnsmasq**: niewielki serwer DNS (i pośrednika sieciowego) oferujący również usługę DHCP. Nie zalecam tego pakietu, chyba że ktoś ma specyficzne potrzeby spełniane właśnie przez ten pakiet lub jest pewny, że nie chce używać serwera BIND, który stanowi standard i jest najbardziej popularnym serwerem DNS. Więcej informacji na temat serwera BIND można znaleźć w rozdziale 29., „Konfiguracja serwera DNS”.
- ♦ **udhcpd**: niewielki serwer DHCP zaprojektowany przede wszystkim do wykorzystania w systemach osadzonych (embedded). Warto rozważyć instalację tego serwera DHCP w przypadku, gdy dostępna pamięć lub zasoby dyskowe są bardzo ubogie.
- ♦ **wide-dhcpv6-server**: serwer DHCP dla hostów IPv6. Ten serwer współpracuje wyłącznie z klientami pracującymi w standardzie IPv6, takimi jak **wide-dhcpv6-client**.

Wybór serwera DHCP jest uzależniony od wymagań. Sugeruję pakiet *dhcpd3-server*, ponieważ jest najnowszą i najdoskonalszą wersją serwera DHCP oferowaną przez firmę ISC. Nie narzucam oczywiście wyboru (powinien on być uzależniony przede wszystkim od środowiska, w którym ma pracować serwer DHCP).

Tworzenie plików konfiguracyjnych serwera DHCP za pomocą edytora tekstu

W tym podrozdziale omówię podstawy konfiguracji serwera DHCP oraz najczęściej stosowany zestaw dyrektyw konfiguracyjnych, definiujących zakres adresów IP, które ma przydzielać serwer DHCP.



Plik */etc/dhcp3/dhcpd.conf* jest wykorzystywany przez serwer zainstalowany z pakietu *dhcpd3*, który radziłem zainstalować w poprzednim podrozdziale. Jeśli ktoś zainstalował inny serwer (wstyd, że nie zastosował się do porady, ale być może miał ważne powody), jego plikiem konfiguracyjnym będzie prawdopodobnie */etc/dhcpd.conf*, co jest tradycyjną lokalizacją pliku konfiguracyjnego serwera DHCP. Prosty sposób sprawdzenia pliku konfiguracyjnego wykorzystywanego przez serwer DHCP jest wywołanie następującego polecenia (gdzie *nazwa-serwera* to nazwa programu serwera DHCP):

```
$ strings `which nazwa-serwera` | grep dhcpd.conf
```

To polecenie wypisze kilka ciągów znaków, wśród których znajdzie się również ścieżka do pliku konfiguracyjnego serwera DHCP. Powyższe polecenie działało prawidłowo dla wszystkich testowanych przeze mnie serwerów DHCP, ale może okazać się, że konieczna będzie pewna modyfikacja.

Aby skonfigurować serwer DHCP dla wybranej sieci, należy w pliku konfiguracyjnym dopisać następującą sekcję:


```
1: option domain-name "nazwa.domeny";
2: option domain-name-servers adres-serwera-dns1, adres-serwera-dns2,... ;
3: option routers adres-rutera;
4: option subnet-mask 255.255.255.0;
5: subnet adres-podsieci netmask maska-podsieci {
6:   range pełny-adres-początkowy pełny-adres-końcowy;
7: }
```



Rzeczywisty plik konfiguracyjny serwera DHCP nie zawiera numerów wierszy (i nie powinien). Numerację dodałem, by uprościć analizę poszczególnych elementów konfiguracji.

Powyższy listing przedstawia minimalistyczny serwer DHCP, użyteczny, ale wykorzystujący wiele wartości domyślnych. Omówię teraz poszczególne wiersze tego pliku.

1. Nazwa domeny, do której powinien należeć host pobierający adres IP. W systemach Linux ta domena jest zapisywana w pliku `/etc/resolv.conf`, więcej szczegółów na ten temat można znaleźć w rozdziale 29., „Konfiguracja serwera DNS”. To ustawienie jest globalne dla serwera DHCP i będzie stosowane dla wszystkich podsieci (chyba że w ramach podsieci zostanie zdefiniowana lokalna nazwa domeny).
2. Nazwy serwerów DNS, których powinien używać host otrzymujący adres IP od tego serwera. W systemach Linux serwery nazw otrzymane od serwera DHCP są zapisywane w pliku `/etc/resolv.conf` (poprzednia zawartość jest nadpisywana). Więcej informacji na temat pliku `/etc/resolv.conf` można znaleźć w rozdziale 29. To ustawienie jest globalne dla serwera DHCP i będzie stosowane dla wszystkich podsieci (chyba że w ramach podsieci zostaną zdefiniowane inne serwery DNS).
3. Trzeci wiersz identyfikuje adres IP routera, czyli systemu, do którego będzie kierowany ruch nieprzeznaczony dla hostów z sieci lokalnej. To ustawienie jest globalne dla serwera DHCP i będzie stosowane dla wszystkich podsieci (chyba że w ramach podsieci zostanie zdefiniowany inny adres routera).
4. Czwarty wiersz określa domyślną maskę podsieci wykorzystywaną w komputerach otrzymujących adresy IP od serwera DHCP. Maską najczęściej ma wartość `255.255.255.0`, ale może być inna, w zależności od konfiguracji sieci. To ustawienie jest globalne dla serwera DHCP i będzie stosowane dla wszystkich podsieci (chyba że w ramach podsieci zostanie zdefiniowana inna maska).



Każde ustawienie globalne w sekcji hosta kończy się średnikiem.

5. Piąty wiersz identyfikuje podsieć, w której serwer DHCP ma obsługiwać adresy IP. Podsieć identyfikuje się za pomocą adresu podsieci, w którym w polach obsługiwanych przez DHCP pozostawia się zera, oraz maski podsieci.
6. Szósty wiersz identyfikuje zakres adresów IP, jakie serwer DHCP będzie mógł wysyłać do klientów.
7. Ostatni wiersz stanowi zamknięcie sekcji dotyczącej podsieci.

Po wprowadzeniu modyfikacji w pliku konfiguracyjnym serwera DHCP należy uruchomić go ponownie. W przypadku serwera DHCP poleconego przeze mnie należy posłużyć się następującym poleceniem:

```
$ sudo /etc/rc.d/init.d/dhcp3-server restart
```

Inne serwery DHCP również posiadają analogiczne skrypty służące do zarządzania. Należy po prostu przejrzeć zawartość katalogu */etc/init.d* w poszukiwaniu odpowiedniego.

Dodatkowe ustawienia w pliku konfiguracyjnym serwera DHCP

W poprzednim podrozdziale omówiłem podstawową konfigurację serwera DHCP, ale to jeszcze nie wszystkie możliwości konfiguracyjne. Istnieją bowiem opcje pozwalające zdefiniować dodatkowe informacje o sieci, które mogą być przydatne klientom otrzymującym adresy IP od naszego serwera.

Poniżej przedstawiłem dodatkowe opcje pliku konfiguracyjnego, przydatne w wielu konfiguracjach. Przykładowy plik konfiguracyjny wykorzystujący omawiane opcje jest przedstawiony na listingu poniżej.

- ♦ **default-lease-time**: ta opcja serwera DHCP pozwala określić domyślny okres ważności dzierżawy adresu. Okres ten jest definiowany w postaci całkowitej liczby sekund.
- ♦ **ddns-update-style**: ta deklaracja może być użyta, gdy w sieci istnieje serwer DNS pozwalający na dynamiczne definiowanie adresów w oparciu o dzierżawy serwera DHCP. Jeśli w sieci pracuje serwer DNS nie wspierający dynamicznej aktualizacji danych dns oferowanej przez serwer DHCP, należy w pliku konfiguracyjnym serwera DHCP ustawić opcję `ddns-update-style none` (co jest wartością domyślną).
- ♦ **ip-forwarding**: jeśli np. nie chcemy, aby klient DHCP wysyłał zapytania IP do innych sieci, należy ustawić wartość tego parametru na `off`. Ta opcja ma zastosowanie przede wszystkim w sieciach o charakterze lokalnych.
- ♦ **max-lease-time**: ta opcja konfiguracyjna serwera DHCP pozwala określić maksymalny okres ważności dzierżawy. Okres ten jest definiowany w postaci całkowitej liczby sekund.

Serwer DHCP może być wykorzystywany również do przydzielania adresów IP konkretnym hostom określonych podsieci, gdzie każdy z hostów jest identyfikowany za pomocą adresu MAC. Do tego celu służy deklaracja `host` w ramach sekcji podsieci. Oto przykład konfiguracji tego typu:

```
option ddns-update-style interim;
option default-lease-time 28800;
option max-lease-time 86400;
option ip-forwarding off;
subnet 192.168.6.0 netmask 255.255.255.0 {
    range 192.168.6.160 192.168.6.190
    host dorothy {
        hardware ethernet 00:1E:0C:1D:75:1D;
```

```
fixed-address 192.168.6.69;  
server-name "192.168.6.64" ;  
option routers 192.168.6.1;  
option host-name "dorothy" ;  
}  
}
```

Należy zwrócić uwagę, że host o nazwie *dorothy* zawiera ustawienia przesłaniające analogiczne ustawienia globalne, zdefiniowane wcześniej w tym pliku.

Zarządzanie serwerem DHCP za pomocą interfejsu graficznego

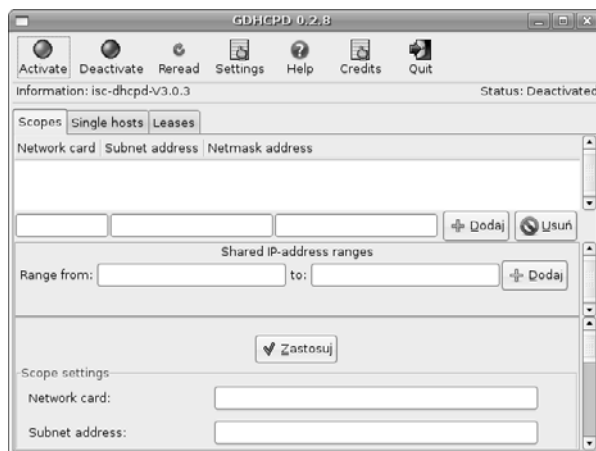
Repozytorium Ubuntu oferuje doskonałe narzędzie graficzne dla środowiska GNOME, służące do konfigurowania serwera DHCP. Ten pakiet (*gdhcpd*) zainstalowaliśmy wcześniej, w podrozdziale „Instalacja serwera DHCP”.

Instalacja pakietu *gdhcpd* powoduje utworzenie odpowiedniej pozycji w menu *System/Administracja*. Niestety, ta pozycja w menu nie uruchamia narzędzia z użyciem mechanizmu *sudo*, przez co użytkownik nie ma możliwości zapisania zmian wprowadzanych za jego pomocą. Aby usunąć ten problem, należy uruchomić program *gdhcpd* z wiersza poleceń terminala GNOME, *xterm* lub podobnego narzędzia pozwalającego na wywoływanie poleceń:

```
$ gksudo gdhcpd
```

Po podaniu hasła zostanie uruchomiony program *gdhcpd*, przedstawiony na rysunku 28.2.

Rysunek 28.2.
Uruchomienie programu GNOME DHCPD

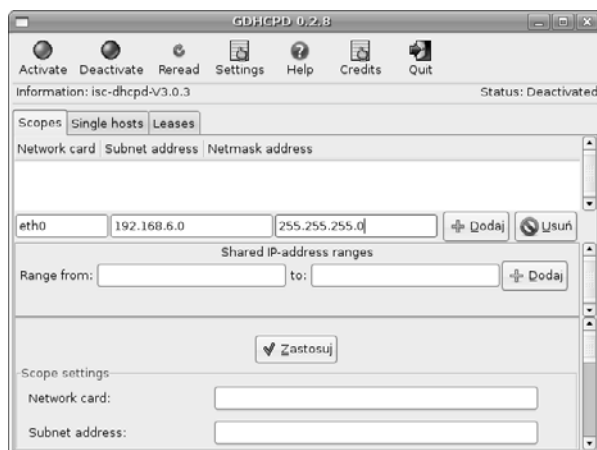


Nie należy się przejmować komunikatem o tym, że aplikacja nie może znaleźć serwera DHCP, ponieważ (w domyślnej konfiguracji) szuka programu o nazwie *dhcpd*, natomiast my zainstalowaliśmy serwer ISC v3 DHCP, którego plik wykonywalny nosi nazwę *dhcp3*.

W pierwszej kolejności należy zdefiniować zakres działania serwera DHCP, tzn. wskazać interfejs sieciowy, na którym będzie nasłuchiwać serwer, zakres adresów IP, jakie będzie przydzielał, oraz wykorzystywaną maskę podsieci. W tym celu trzeba wprowadzić odpowiednie wartości zgodnie z rysunkiem 28.3. W naszym przykładzie serwer DHCP nasłuchuje na interfejsie *eth0*, przydziela adresy IP z podsieci *192.168.6.0* z zastosowaniem maski *255.255.255.0*.

Rysunek 28.3.

Podstawowa konfiguracja serwera DHCP

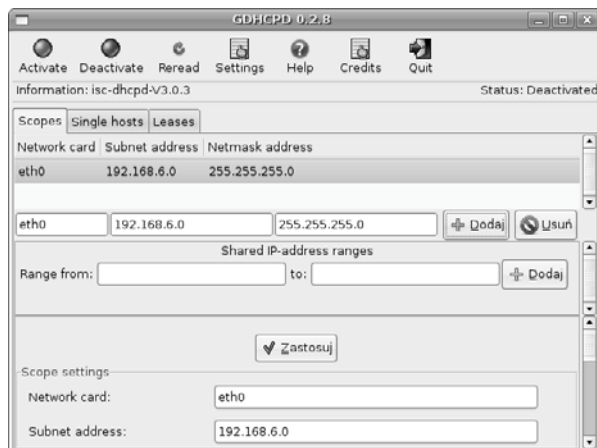


Określenie zakresu (*scope*) jest stosowane w odniesieniu do zakresu adresów IP w sieci. Najczęściej dotyczy to fizycznych podsieci w ramach sieci lokalnej, którym serwer DHCP przydziela adresy. Te zakresy to podstawowe narzędzie pozwalające serwerowi DHCP zarządzać różnymi podsieciami o różnych zakresach adresów IP (oraz innych parametrów sieci) przydzielanych klientom.

Po wprowadzeniu tych informacji należy kliknąć przycisk *Dodaj*. W karcie *Scopes* pojawi się nowa pozycja, co przedstawiam na rysunku 28.4.

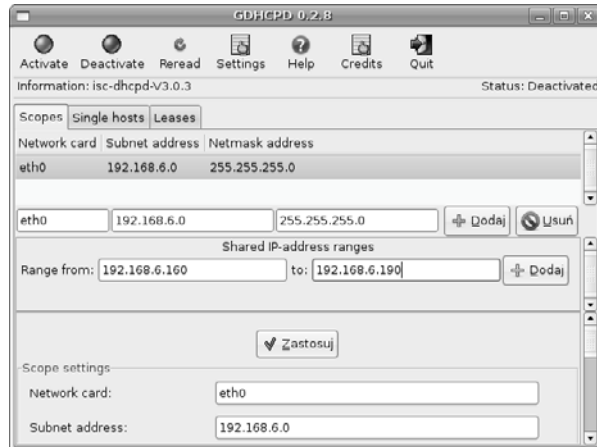
Rysunek 28.4.

Dodanie zakresu serwera DHCP



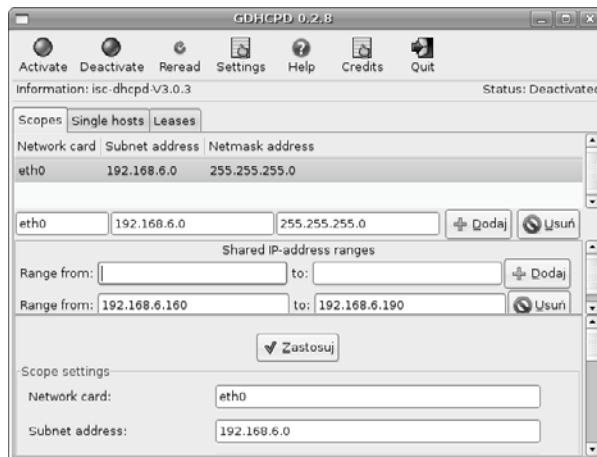
Następnie trzeba wprowadzić zakres adresów IP, które mają być przydzielane klientom przez dany serwer DHCP (patrz rysunek 28.5).

Rysunek 28.5.
Określanie zakresu adresów przydzielanych przez serwer DHCP



Po zdefiniowaniu zakresu należy kliknąć przycisk *Dodaj*. Zdefiniowany zakres adresów pojawi się w głównym oknie programu *gdhcpd*, co przedstawiam na rysunku 28.6.

Rysunek 28.6.
Zdefiniowany zakres adresów przydzielanych przez serwer DHCP



Należy również sprawdzić ustawienia samego serwera, do czego służy ikona *Settings* w pasku narzędzi. Po jej kliknięciu pojawi się okno przedstawione na rysunku 28.7.

Rysunek 28.7.
Ogólne ustawienia serwera DHCP



Na tym ekranie można zdefiniować różne aspekty działania serwera DHCP. Gdy konfigurowany jest serwer inny niż zalecany przeze mnie *dhcp3-server*, niezbędne może okazać się odpowiednie zdefiniowanie ścieżki do pliku konfiguracyjnego oraz do pliku, w którym zapisywane są przydzielone dzierżawy.

Przydatne może okazać się również modyfikowanie ogólnych ustawień serwera. Kiedy np. serwer DHCP ma możliwość dynamicznego aktualizowania konfiguracji serwera DNS, należy ustawić odpowiednią opcję *DDNS update style* na wartość *ad-hoc* lub *interim* (tryb *ad hoc* jest przestarzały i niezalecany, tryb *interim* pozwala serwerowi DHCP na aktualizację danych serwera DNS za każdym razem, gdy zostanie przydzielona dzierżawa).

Po zweryfikowaniu globalnych ustawień serwera DHCP należy kliknąć przycisk *Zastosuj*, co spowoduje zamknięcie tego okna i powrót do głównego okna programu *gdhcpd*.

Jeśli ustawienia zakresów i inne parametry adresów zostały zdefiniowane, można zamknąć i to okno, klikając przycisk *Zakończ*. Serwer DHCP jest skonfigurowany i czas najwyższy, aby go uruchomić.



Przed uruchomieniem serwera DHCP warto sprawdzić, czy serwer nie wymaga dodatkowych ustawień, zgodnie z podrozdziałem „Dodatkowe ustawienia w pliku konfiguracyjnym serwera DHCP”.

Aby ponownie uruchomić serwer DHCP po modyfikacji jego pliku konfiguracyjnego, należy wywołać następujące polecenie:

```
$ sudo /etc/init.d/dhcp3-server restart
```

To zatrzyma serwer *dhcp3-server* i ponownie go uruchomi, co w rezultacie spowoduje odczytanie pliku konfiguracyjnego. Gratulacje: mamy działający serwer DHCP.

W tym momencie należy sprawdzić zawartość pliku */var/log/messages* w celu weryfikacji prawidłowego uruchomienia serwera DHCP. Można to zrobić za pomocą następującego polecenia, które wypisze na ekranie sto ostatnich wierszy tego pliku:

```
$ sudo tail -100 /var/log/messages
```

W kolejnym podrozdziale podam kilka rad przydatnych wtedy, gdy serwer DHCP nie uruchamia się prawidłowo lub stacje klienckie nie mogą uzyskać od niego informacji.

Rozwiązywanie problemów z DHCP

Najczęstsze problemy z DHCP polegają na tym, że serwer nie uruchomił się prawidłowo lub przydzielił już wszystkie dostępne adresy IP. W domyślnej konfiguracji wszelkie komunikaty związane z uruchamianiem, zamykaniem i błędami serwera DHCP są zapisywane w pliku */var/log/messages*. Jeśli serwer nie uruchamia się prawidłowo, przyczyny problemów w pierwszej kolejności należy szukać w tym pliku. Jeżeli plik konfiguracyjny był modyfikowany ręcznie, przyczyną mogą być brakujące średniki lub błędy składni, powodujące zamykanie serwera lub, co gorsza, działanie w sposób niezgodny z oczekiwaniami.

Jeśli komputery klienckie nie są w stanie skontaktować się z serwerem DHCP, z reguły samodzielnie „przydzielają sobie” adresy typu *169.x.x.x*, co stanowi sygnał, że komputer nie został przyłączony do sieci. Jeśli któryś z systemów otrzyma adres IP z takiego zakresu, w pierwszej kolejności należy sprawdzić oczywiste przyczyny: czy komputer jest przyłączony do sieci i czy jego interfejs Ethernet jest skonfigurowany do korzystania serwera DHCP. W Ubuntu trzeba sprawdzić konfigurację interfejsów Ethernet, wywołując z menu funkcję *System/Administracja/Sieć* (należy podać hasło). Trzeba również zweryfikować zawartość pliku */var/log/messages* serwera, sprawdzając, czy żądanie klienta wpłynęło, ale z jakiegoś powodu zostało odrzucone.

W systemach przyłączonych do wielu sieci częstą przyczyną problemów jest nasłuchiwanie serwera DHCP na niewłaściwym interfejsie, przez co komputery w sieci lokalnej nie mogą uzyskać do niego dostępu. Uruchamianie serwera DHCP na określonym interfejsie Ethernet uzyskuje się przez podanie nazwy tego interfejsu Ethernet w wywołaniu serwera DHCP, które najczęściej odbywa się za pośrednictwem skryptu rozruchowego, np. */etc/init.d/dhcpd*. Nazwa tego pliku może być inna dla różnych wersji serwerów DHCP dostępnych dla systemu Ubuntu.

Na koniec warto zasugerować wymuszenie na kliencie zwolnienia posiadanych przez niego dzierżaw adresów IP. Służy do tego polecenie *dhclient3 -r* (lub *dhclient -r* w przypadku klienta v2). Klient DHCP po takim wywołaniu nie odświeży adresu, można tego dokonać przez następujące wywołanie:

```
$ sudo /etc/init.d/networking restart
```

To polecenie uruchamia ponownie wszystkie procesy związane z sieciami, przy okazji też programy klientów DHCP, przez co konfiguracja DHCP odbywa się od nowa.

Podsumowanie

Możliwość dynamicznego przydzielania i centralnego zarządzania zakresami klienckich adresów IP i konfiguracją sieci znacząco upraszcza życie wielu administratorów sieci. W tym rozdziale wyjaśniłem, w jaki sposób zainstalować serwer *Dynamic Host Configuration Protocol* (DHCP) w systemie Ubuntu, wykorzystując do tego celu wiersz poleceń lub graficzne narzędzie do zarządzania oprogramowaniem. Ponadto umieściłem tu informacje o konfiguracji serwera DHCP w trybie tekstowym (przez edycję pliku konfiguracyjnego) oraz w trybie graficznym. Na końcu rozdziału można znaleźć porady dotyczące usuwania potencjalnych problemów z użytkowaniem i konfiguracją DHCP.

Rozdział 29. stanowi kontynuację tematyki niniejszego rozdziału; omawiam tam instalację i konfigurację serwera *Domain Name System* (DNS). Serwer DNS odwzorowuje nazwy hostów na ich adresy IP (i vice versa) i często stanowi ważny element sieci akademickich i korporacyjnych, gdzie występuje konieczność zarządzania nazwami hostów i powiązanymi z nimi adresami IP.

Rozdział 29.

Konfiguracja serwera DNS

W tym rozdziale:

- ◆ Podstawowe informacje o DNS i serwerze BIND
- ◆ Przegląd plików konfiguracyjnych serwera BIND
- ◆ Tworzenie plików stref
- ◆ Tworzenie plików dla zapytań odwrotnych
- ◆ Rozwiązywanie problemów z DNS-em

Jak wspomniałem w rozdziale poświęconym protokołowi DHCP, dawno temu, w czasach swojej młodości internet (który wówczas nosił nazwę ARPANET) był niewielką siecią łączącą ograniczoną, znaną liczbę hostów, z których wszystkie miały ustalone z góry adresy IP. Odzwzorowania tych adresów IP na nazwy hostów były obsługiwane w oparciu o centralny plik tekstowy. Tak, to nie pomyłka, pojedynczy plik tekstowy. Wszystkie hosty pobierały ten plik (*HOSTS.TXT*) co jakiś czas z centralnego hosta obsługiwanego przez Stanford Research Institute Network Information Center (NIC), a wszelkie lokalne zmiany były wysyłane do tego centralnego zasobu w celu wprowadzenia do bazy. Jednakże ARPANET szybko rozrósł się i przyłączano do niego coraz więcej hostów, tak więc utrzymywanie pojedynczej bazy w formacie tekstowym stało się tyle niepraktyczne, co nieodpowiedzialne.

Ten plik miał duże rozmiary (i stawał się coraz większy), a jego format zdefiniowany był raczej w formie konwencji niż formalnej definicji. Internet/ARPANET był opracowany przez niewielką grupę osób, a wiele udoskonaleń tego modelu dokonało się we wczesnych latach 80. ubiegłego wieku. W marcu 1982 roku definicja tabeli hostów została zastąpiona formatem DoD Internet Host Table Specification (RFC 810). Krótco potem w RFC 811 została zdefiniowana funkcja serwera dokonującego translacji nazw hostów na adresy sieciowe, co doprowadziło do opracowania dynamicznego mechanizmu definiowania odzworowań adresów i nazw w ramach RFC 819 („The Domain Naming Convention for Internet User Applications”), co stanowiło prawdziwy przełom, który stał się przyczynkiem do opracowania standardu DNS w ramach RFC 830 („A Distributed System for Internet Name Service”), który sugerował opracowanie rozproszonego systemu serwerów

nazw, z których każdy miał obsługiwać jedną, niezależną domenę. W latach 1982 i 1983 opracowano wiele fundamentalnych koncepcji, które obecnie stały się codziennością.



Po pojawieniu się odpowiednich RFC użytkownicy uzyskali możliwość ich analizy. Ich implementacja była po prostu kolejnym krokiem. W listopadzie 1983 roku opracowano koncepcję i terminarz implementacji systemu DNS, który został opublikowany w ramach RFC 881 („The Domain Names Plan and Schedule”), RFC 882, („Domain Names — Concepts And Facilities”) i RFC 883 („Domain Names — Implementation And Specification”).

W reakcji na te RFC na University of California w Berkeley (UCB) opracowano oprogramowanie o nazwie BIND, od Berkeley Internet Name Daemon. Projekt powstał w ramach pracy dyplomowej. Wkrótce opieka nad projektem BIND została przejęta przez Computer Systems Research Group (CSRG), potem na niedługi czas przekazana firmie Digital Equipment Corporation (DEC), a w końcu BIND trafił do Internet Systems Consortium (<http://www.isc.org>), który rozwija projekt BIND do dziś. Od powstania projektu BIND opracowano wiele dokumentów RFC rozbudowujących koncepcję systemu DNS (RFC 973, 1035 itd.), niektóre z nich rozszerzały zakres istniejących RFC, inne zastępowały istniejące koncepcje nowymi, bardziej nowoczesnymi. Jak wspominałem w rozdziale 28., „Konfiguracja serwera DHCP”, dokumenty RFC można znaleźć w serwisie <http://www.ietf.org/rfc.html> oraz w nieco przyjaźniejszym użytkownikowi serwisie <http://www.faqs.org/rfcs>.



Wszelkie odniesienia do nazwy BIND w dalszej części rozdziału dotyczą serwera BIND w wersji 9., czyli wersji, której wykorzystanie zalecam i opisuję w tym rozdziale.

DNS został zaprojektowany przede wszystkim w celu rozwiązania problemów skalowalności, typowych dla pojedynczego pliku z odwzorowaniami nazw hostów na adresy IP, przez skonstruowanie elastycznego mechanizmu pozwalającego na zadawanie zapytań. Ten mechanizm nie gorzej sprawdza się w środowiskach sieci lokalnych, które mogą posiadać własne, lokalne serwery DNS samodzielnie obsługujące żądania prywatne, natomiast na żądania o adresy globalne odpowiadają, wykonując zapytania rekurencyjne do serwerów DNS wyższych poziomów hierarchii. Lokalne serwery nazw mogą również buforować wyniki zapytań, dzięki czemu ilość niezbędnych do wykonania zapytań DNS będzie mniejsza. Wiele sieci posiada serwery DNS służące wyłącznie do buforowania wyników zapytań DNS, nazywa się je „buforującymi serwerami DNS”.

W dalszej części tego rozdziału opiszę działanie usługi DNS, instalację najnowszej wersji serwera nazw BIND w systemie Ubuntu oraz zasady tworzenia tekstowych plików konfiguracyjnych serwera BIND, jego uruchamianie i testowanie oraz diagnostykę wielu możliwych problemów z systemem DNS.



BIND i DNS to skomplikowane zagadnienia, o których napisano niejedną książkę. Nie mogę pozwolić sobie na umieszczanie tego obszernego materiału w książce o Ubuntu. Dlatego na końcu rozdziału można znaleźć odnośniki do wielu szczegółowych zasobów online dotyczących serwera BIND i systemu DNS. Jeśli ktoś planuje wykorzystanie własnego serwera DNS w skomplikowanym środowisku komercyjnym, sugeruję wizytę w księgarni w celu nabycia solidnej pozycji na temat systemu DNS i serwera BIND, które będą zawsze pod ręką, niezależnie od dostępności sieci. Do moich ulubionych pozycji należą *DNS and BIND*, Paula Albitza i Cricketa Liu (O'Reilly) oraz *Pro DNS and BIND*, Rona Aitchisona (Apress). Tę ostatnią można również znaleźć na stronie <http://www.netwidget.net/books/apress/dns>, ale i tak warto kupić wersję drukowaną.

Podstawowe informacje o DNS i serwerze BIND

System DNS daje sieciom możliwość definiowania hierarchicznej struktury domenowych nazw hostów oraz ich odwzorowania na adresy IP serwerów w sieci lokalnej lub w internecie. Wszystkie hosty w sieci mogą wykorzystywać lokalny serwer DNS w poszukiwaniu adresu IP odpowiadającego lokalnej nazwie hosta. Żądania informacji związanych z nazwami hostów spoza sieci lokalnej są automatycznie przekazywane do nadrzędnych serwerów DNS aż do momentu, gdy jakiś serwer DNS będzie w stanie udzielić autorytatywnej odpowiedzi na zadane zapytanie.

Powszechnym nieporozumieniem jest to, że system musi posiadać własny serwer DNS, żeby mógł korzystać z DNS. System może być zwykłym klientem DNS bez działającego procesu serwera DNS. Klientka strona systemu DNS to *resolver*, mechanizm rozwiązywania nazw opierający swoje działanie na pliku konfiguracyjnym */etc/resolv.conf* (który omówię w dalszej części rozdziału). To, czy system korzysta z usługi DNS, jest definiowane za pomocą pliku */etc/nsswitch.conf* (Name Service Switch), również omówionego w dalszej części rozdziału.

Większość hostów, oprócz serwera DNS, może korzystać z tradycyjnego źródła informacji o odwzorowaniu nazwy hosta na adres IP, chodzi o plik */etc/hosts*. Gdy aplikacja musi skontaktować się z określoną nazwą hosta, system wykonuje następującą sekwencję czynności.

1. Sprawdza zawartość pliku */etc/nsswitch.conf* w celu określenia, czy system jest skonfigurowany tak by wykorzystywać DNS. Plik */etc/nsswitch.conf* identyfikuje sekwencję, w jakiej są sprawdzane pliki zasobów w poszukiwaniu różnych informacji, w tym informacji o adresach IP i nazwach hostów. Wpis *hosts* z reguły ma następującą zawartość:

```
hosts:    files dns [NOTFOUND=return] nis nisplus
```

System, w którym występuje taka definicja, w pierwszej kolejności sprawdza zawartość pliku */etc/hosts*, następnie wykonuje zapytanie DNS. Jeśli adres IP hosta nie zostanie zwrócony z serwera DNS, zapytanie o adres IP zakończy się niepowodzeniem. Pozycje po ciągu znaków *[NOTFOUND=return]* nie są wykorzystane, ale pozostawia się je w celu zasygnalizowania, że również stanowią prawidłowe wartości tej opcji. W tym przypadku niewykorzystane pozostaną opcje *NIS* i *NIS+*.

Niektóre starsze aplikacje nadal wykorzystują inny plik konfiguracyjny o nazwie */etc/host.conf*. Ten plik zawiera analogiczną informację, co plik */etc/nsswitch.conf*, informując o kolejności wykorzystania pliku */etc/hosts* i zapytań DNS, np.:

```
order hosts,bind
```

Funkcje *resolvera* w większości nowszych dystrybucji wykorzystują już plik */etc/nsswitch.conf*. Kiedy jednak plik */etc/nsswitch.conf* zawiera prawidłową konfigurację, a aplikacja ewidentnie sprawdza adresy IP w niewłaściwej kolejności, należy upewnić się, czy nie korzysta z pliku */etc/hosts* i czy ten nie jest błędnie skonfigurowany. Jeśli dalej mamy problemy, warto zgłosić błąd opiekunowi programu, który przysparza nam kłopotów z funkcjami *resolvera*.



2. Zakładając, że nazwa hosta i jego adres IP nie zostaną odnalezione w pliku */etc/hosts*, system skonfigurowany do korzystania DNS-a sprawdza zawartość pliku */etc/resolv.conf* w celu zidentyfikowania serwerów DNS, z którymi powinien się skonsultować (w określonej kolejności). Ten plik często zawiera dodatkowe informacje takie jak domyślna domena, która powinna być dopisana w przypadku wyszukiwania niekwalifikowanych nazw hostów. Przykładowy plik */etc/resolv.conf* (na jednym z moich systemów Ubuntu) ma następującą zawartość:

```
domain vonhagen.org
nameserver 0.0.0.0
nameserver 68.87.75.194
nameserver 68.87.64.146
nameserver 207.44.142.94
```

W tym przykładzie niekwalifikowane nazwy hostów, takie jak *64bit*, będą automatycznie uzupełnione o domenę *vonhagen.org*. Serwery nazw są odpytywane kolejno, aż do uzyskania odpowiedzi, lub gdy wystąpi określona liczba nieudanych prób, co spowoduje zakończenie zapytania DNS z błędem.



Serwer nazw o adresie 0.0.0.0 to w rzeczywistości skrót oznaczający odpytanie komputera lokalnego na pierwszym adresie IP zdefiniowanym w systemie. W praktyce oznacza to adres interfejsu *loopback*, czyli 127.0.0.1, ale niekoniecznie. Adres 0.0.0.0 odnosi się po prostu do pierwszego znalezionej prawidłowego adresu IP komputera lokalnego.

3. Gdy serwer DNS odpowie, w pierwszej kolejności podejmuje próbę samodzielnej odpowiedzi na zapytanie. Serwery DNS są konfigurowane za pomocą plików tekstowych, które zostaną omówione w kolejnym podrozdziale. Pliki definiują tzw. strefy DNS (*zone*), które są analogią do domen i poddomen, dla których dany serwer DNS jest serwerem autorytatywnym. Jeśli adres hosta i jego adres IP należą do domeny, dla której serwer DNS jest autorytatywny, lub gdy ten adres znajduje się w pamięci podręcznej (*cache*) serwera buforującego (i pozwala na to okres ważności, czyli TTL wpisu), serwer odsyła odpowiednie informacje do klienta. Jeśli żądanie znajduje się w poddomenie, której autorytatywny serwer DNS jest znany pierwszemu serwerowi DNS, żądanie zostaje przesłane do tego serwera. Ten drugi serwer DNS odpowie na żądanie, a wynik zostanie przesłany do klienta.
4. Jeśli pierwszy serwer DNS nie potrafi w żaden sposób udzielić odpowiedzi na zapytanie, przekazuje je do serwerów nazw najwyższego poziomu w celu identyfikacji serwera, który jest odpowiedzialny za obsługę domeny lub zakresu adresów IP zawierających poszukiwany adres IP (dla zapytań odwrotnych).
5. Serwer nazw najwyższego poziomu zwraca adres odpowiedniego serwera nazw, który jest następnie odpytywany przez pierwszy serwer DNS w poszukiwaniu adresu IP lub nazwy hosta. Jeśli odnaleziony serwer nazw nie jest serwerem autorytatywnym dla poszukiwanej domeny lub adresu IP, może przekazać żądanie do innych serwerów. Taka ścieżka jest kontynuowana, aż do odnalezienia serwera nazw autorytatywnego dla poszukiwanej domeny lub adresu IP. Z reguły odbywa się to w sposób niewidoczny dla klienta i najczęściej dotyczy zapytań o poddomenę, nigdy o domenę najwyższego poziomu.

Tego typu kombinacja serwerów DNS: lokalnych, najwyższego poziomu i hierarchicznych stanowi przykład elegancji i prostoty. Gdy host otrzyma poszukiwaną nazwę hosta lub adres IP, z reguły zapisuje te informacje w pamięci podręcznej (*cache*), aby kolejne zapytania dotyczące tych samych danych mogły być obsłużone bez konieczności angażowania innych serwerów DNS.

Serwer BIND może być skonfigurowany na jeden z kilku sposobów.

- ♦ Jako serwer podstawowy (*master*), będący autorytatywnym źródłem informacji na temat jednej lub większej liczby stref, które najczęściej odpowiadają domenom lub poddomenom.
- ♦ Jako serwer podrzędny (*slave*), będący autorytatywnym źródłem informacji na temat jednej lub większej liczby stref, ale okresowo odczytujący swoją konfigurację z serwera podstawowego, który jest głównym źródłem tych informacji.
- ♦ Jako serwer wyłącznie buforujący, niebędący autorytatywnym źródłem informacji na temat jakiegokolwiek strefy, ale który buforuje wyniki zapytań DNS, aby zminimalizować ilość ruchu sieciowego i ograniczyć zapytania o te same informacje DNS. Większość serwerów DNS w systemach Ubuntu będzie zapewne skonfigurowana właśnie w taki sposób.
- ♦ Jako klient nieświadczący usług DNS, ale którego resolver jest skonfigurowany do konsultacji serwerów DNS w oparciu o plik */etc/resolv.conf*.

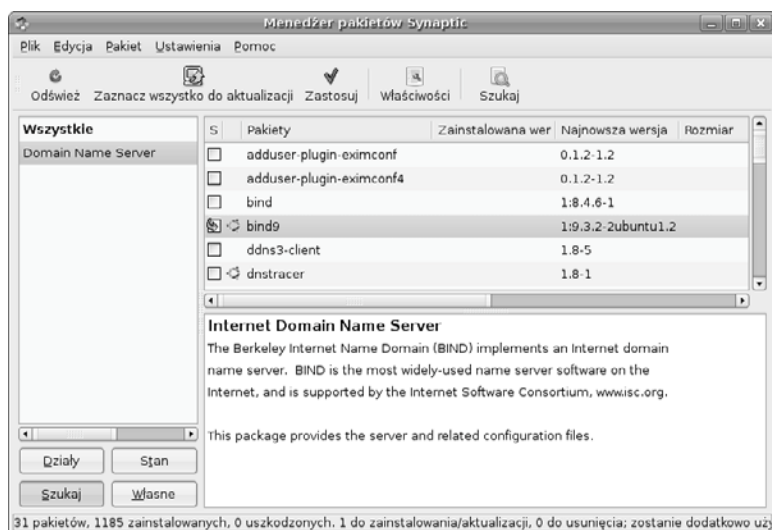
Większość systemów biurowych nie potrzebuje serwera DNS i będzie doskonale działać wyłącznie w charakterze klientów. Jeśli jednak ktoś spędza mnóstwo czasu, przeglądając WWW w mocno obciążonej sieci, warto wziąć pod uwagę konfigurację własnego, buforującego serwera DNS. Serwery DNS nie pochłaniają dużych ilości zasobów systemu, ale potrzebują sporych ilości pamięci, uchwytów procesów itp. Jeśli konfigurowany lub używany system nie musi oferować informacji o hostach wchodzących w skład strefy (co wymuszałoby konfigurację serwera DNS), wówczas decyzja czy lepszy będzie buforujący serwer DNS, czy zwykła konfiguracja kliencka należy do użytkownika.

Instalacja serwera DNS za pomocą menedżera pakietów Synaptic

Choć każdy system Ubuntu zawiera podstawowe polecenia klienckie do zadawania zapytań DNS i konstruowania dynamicznych odwzorowań nazw hostów na adresy IP, serwer DNS nie jest instalowany w standardowej konfiguracji Ubuntu. Nie ma się czemu dziwić, nie każdy użytkownik potrzebuje serwera DNS, a dla niewielkich, domowych sieci utrzymanie pliku */etc/hosts* z danymi wszystkich stacji nie stanowi większego problemu. Podobnie jak w przypadku innych pakietów oprogramowania w Ubuntu, instalacja serwera BIND jest szybka i łatwa. Można to zrobić np. za pomocą menedżera pakietów Synaptic. W tym celu należy uruchomić to narzędzie z menu *System/Administracja/Menedżer pakietów Synaptic* i podać hasło, po czym kliknąć przycisk *Wyszukaj*. W oknie wyszukiwania trzeba wybrać pozycję *Nazwy i opisy*, w polu wyszukiwania wpisać ciąg znaków *Domain Name Server*, po czym kliknąć przycisk *Szukaj*. Po zakończeniu wyszukiwania należy odszukać

pakiet *bind9*, kliknąć jego nazwę prawym przyciskiem myszy i z menu podręcznego wybrać funkcję *Zaznacz do instalacji*. Na rysunku 29.1 przedstawiam pakiet *bind9* zaznaczony i gotowy do instalacji.

Rysunek 29.1.
Instalacja serwera
BIND 9 za pomocą
programu Synaptic



Po zaznaczeniu pakietu należy kliknąć ikonę *Zastosuj* w pasku narzędzi programu Synaptic, co rozpocznie instalację serwera BIND. Po zakończeniu instalacji można zamknąć menedżer pakietów Synaptic. Jesteśmy prawie na miejscu!

Przegląd plików konfiguracyjnych serwera BIND

Pliki */etc/nsswitch.conf* i */etc/resolv.conf* odzwierciedlają strategię systemu dotyczącą pozyskiwania adresów IP i nazw hostów. Podstawową konfigurację serwera nazw BIND, czyli plik *named.conf*, można w systemach Ubuntu znaleźć w katalogu */etc/bind*.

W Ubuntu plik */etc/bind/named.conf* zawiera głównie instrukcje włączające inne pliki konfiguracyjne serwera BIND, co daje prostą, hierarchiczną strukturę plików konfiguracyjnych. Pliki ładowane przez główny plik konfiguracyjny serwera BIND w Ubuntu można z grubsza zaliczyć do trzech kategorii.

- ♦ Pliki konfiguracyjne procesu serwera BIND (*named*). Domyślna konfiguracja serwera BIND w Ubuntu składa się z dwóch części, ogólnej, wspólnej dla większości instalacji, oraz specyficznej dla tej konkretnej instalacji. Taka strategia upraszcza aktualizację systemu i identyfikację plików, które uległy zmianom od momentu instalacji poprzedniej wersji serwera BIND.
- ♦ Pliki stref definiujące odwzorowanie nazw hostów na ich adresy IP.
- ♦ Pliki odwrotnego DNS, definiujące odwzorowanie adresów IP na ich nazwy domenowe.

W ramach przykładu domyślny plik konfiguracyjny serwera BIND */etc/bind/named.conf* w systemie Ubuntu ładuje następujące pliki (lista jest w kolejności alfabetycznej, aby ułatwić odnalezienie pliku na liście, a nie w kolejności ładowania):

- ♦ */etc/bind/db.0*: zawiera dane dla zapytań odwrotnych DNS dla adresów rozgłaszania *0.**,
- ♦ */etc/bind/db.127*: zawiera dane dla zapytań odwrotnych dla adresów interfejsu loopback *127.**,
- ♦ */etc/bind/db.255*: zawiera dane dla zapytań odwrotnych dla adresów rozgłaszania *255.**,
- ♦ */etc/bind/db.empty*: plik wspólny dla wszystkich nieroutowalnych adresów (zgodnych z RFC 1918, „Address Allocation for Private Internets”, <http://www.faqs.org/rfcs/rfc1918.html>),
- ♦ */etc/bind/db.local*: zawiera informacje o nazwie hosta i adresie IP dla lokalnego interfejsu loopback *127.0.0.1*,
- ♦ */etc/bind/db.root*: zawiera informacje o serwerach nazw najwyższego poziomu,
- ♦ */etc/bind/named.conf.local*: identyfikuje strefy i adresy IP, dla których dany serwer DNS jest autorytatywnym źródłem informacji, jest to w pewnym sensie lokalne, zależne od hosta uzupełnienie pliku */etc/named.conf*,
- ♦ */etc/bind/named.conf.options*: definiuje podstawowe opcje wykorzystywane w konfiguracji serwera BIND,
- ♦ */etc/bind/zones.rfc1918*: zawiera dane dla odwrotnych zapytań DNS dla każdej nieroutowalnej rodziny adresów, określonych w ramach RFC 1918, „Address Allocation for Private Internets” (<http://www.faqs.org/rfcs/rfc1918.html>). Ten plik nie jest domyślnie ładowany przez serwer BIND instalowany w Ubuntu, ale może być ładowany do pliku */etc/bind/named.conf.local* w celu skrócenia czasu niezbędnego do rozwiązywania nazw i adresów IP w sieciach nieroutowalnych.

Zwykle wystarczy zmodyfikować zawartość pliku */etc/bin/named.conf.local*. Należy w nim dodać wpisy dla plików, w których znajdują się definicje stref oraz zakresy adresów IP, dla których serwer BIND stanowi autorytatywne źródło informacji DNS.



Opcje serwera BIND dla systemu Ubuntu zawarte w pliku */etc/bin/named.conf.options* konfiguruje te serwery w taki sposób, aby ich katalogiem roboczym był */var/cache/bind*. Wszelkie tymczasowe dane serwera BIND są zapisywane w tym katalogu. Z tego powodu, ścieżki w pliku */etc/bin/named.conf.local* powinny być pełnymi ścieżkami do plików, jak */etc/bind*, w przeciwnym wypadku ścieżki będą interpretowane jako podkatalogi katalogu */var/cache/bin*.

Oto przykładowa zawartość pliku */etc/bin/named.conf.local* jednego z moich serwerów DNS:

```
zone "vonhagen.org" {
    type master;
    file "/etc/bind/vonhagen.zone";
    check-names warn;
};
```

```
zone "168.192.in-addr.arpa" {  
    type master;  
    file "/etc/bind/168.192.in-addr.arpa";  
};  
  
include "/etc/bind/zones.rfc1918.local";
```

Jak widać, ten serwer DNS jest serwerem głównym dla strefy *vonhagen.org*, której adresy IP należą do podsieci *192.168.**. Informacje o strefie *vonhagen.org* znajdują się w pliku */etc/bind/vonhagen.zone*. Informacje dla zapytań odwrotnych dla podsieci *192.168.** znajdują się w pliku */etc/bind/168.192.in-addr.arpa*. Jak wspomniałem wcześniej, pliki zapisane poza katalogiem */var/cache/bind* należy podawać wraz z pełną ścieżką. Ten plik ładuje również lokalną kopię domyślnego pliku */etc/bin/zones.rfc1918*, z którego usunąłem definicję pustych odpowiedzi dla strefy *168.192.in-addr.arpa*, którą wykorzystuję lokalnie.

W następnym podrozdziale opiszę konstrukcję plików stref i danych dla odwrotnych zapytań DNS w dowolnych systemach lokalnych, dla których BIND jest serwerem autorytatywnym.

Tworzenie definicji strefy DNS i pliku danych dla zapytań odwrotnych

W domyślnej konfiguracji w Ubuntu serwer BIND jest tylko buforującym serwerem nazw, czyli nadaje się do pracy w sieciach lokalnych lub w systemach intensywnie korzystających z sieci, w celu zminimalizowania liczby zewnętrznego ruchu, dzięki buforowaniu danych DNS i uniknięciu wielokrotnych zapytań o ten sam adres.

W wielu przypadkach, wykorzystanie serwera DNS dotyczy odczytów informacji DNS dla hostów z sieci domowej, korporacyjnej lub akademickiej. Jak wspominałem wcześniej w tym rozdziale, informacje DNS dla każdego zbioru hostów są grupowane w plikach stref, czyli odwzorowaniach nazwy na adres IP, oraz w plikach zapytań odwrotnych, gdzie w oparciu o adres IP odczytuje się nazwę hosta. W kilku kolejnych punktach omówię obowiązkowe elementy występujące w plikach każdego z tych typów, opiszę też format plików stref oraz format plików zapytań odwrotnych.



Pliki, które nazywam plikami stref oraz plikami zapytań odwrotnych mają w zasadzie ten sam format. Rozróżniam te dwie koncepcje przede wszystkim dlatego, że tak łatwiej zrozumieć ich zasadę działania: w każdym z tych plików zapisywane są rekordy DNS o różnych typach i służą one różnym celom w kontekście serwera DNS.

Wykorzystanie wspólnych wpisów dla stref oraz plików zapytań odwrotnych: SOA i \$TTL

Na początku pliku strefy oraz zapytań odwrotnych znajdują się definicje zmiennych wykorzystywanych przez proces serwera BIND (*/usr/sbin/named*) oraz zmiennych dla każdej strefy lub zakresu adresów IP, dla których serwer jest autorytatywnym źródłem danych DNS.

Plik strefy z reguły na początku zawiera zmienną TTL (ang. *Time To Live*) definiującą domyślny okres ważności każdego rekordu w pliku strefy. Identyfikator zmiennej TTL jest poprzedzony znakiem dolara, a sama wartość określana jest liczbą sekund (domyślnie), minut, godzin lub dni. Najczęściej wykorzystuję wartość jednego dnia (1d), ponieważ często dodaję, modyfikuję lub zmieniam nazwy definicji hostów w swojej sieci. Dlatego u mnie definicja zmiennej TTL ma następującą postać:

```
$TTL 1d
```

Najważniejszą pozycją w pliku definicji strefy lub danych dla zapytań odwrotnych jest rekord SOA (ang. *Start Of Authority*), który definiuje globalne parametry dla strefy lub zakresu adresów IP. Format rekordu SOA jest zdefiniowany w RFC 1035 (<http://www.ietf.org/rfc/rfc1035.html>). Oto przykładowy rekord SOA:

```
@ 1d IN SOA ubuntu32.vonhagen.org. wvh.vonhagen.org. (
    2006070102 ; Serial
    21600      ; Refresh
    1800       ; Retry
    604800     ; Expiration
    900        ; Negative Cache TTL
)
```



Jeden z problemów z rekordem SOA (który dotyczy również innych elementów konfiguracyjnych serwera DNS) polega na tym, że wiele pól posiada wartości domyślne, przez co bywają pomijane w niektórych plikach konfiguracyjnych. Osobiście praktykę pomijania wartości domyślnych uważam za lenistwo oraz przyczynę większości problemów z serwerem DNS. Parser plików konfiguracyjnych serwera DNS jest dość kapryśny i idąc na łatwiznę łatwo zrobić krzywdę sobie, albo innemu administratorowi.. W końcu żaden z parametrów plików definicji stref lub odwzorowań odwrotnych nie jest szczególnie skomplikowany i nie wymaga ogromnej ilości wpisywania. Zrób sobie i innym przysługę: nie omijaj definicji ważnych zmiennych, nawet jeśli przyjmują one wartości domyślne gdy zostaną pominięte.

Rekord SOA składa się z następujących pól.

- ♦ Pierwsze pole identyfikuje nazwę domeny głównej dla strefy związanej z rekordem SOA. Symbol @ stanowi skrót do tej nazwy, zdefiniowanej w deklaracji strefy w pliku konfiguracyjnym serwera BIND (w tym przypadku */etc/bind/named.conf.local*). Zdarza się, że administratorzy wpisują tu właściwą nazwę domeny, ja preferuję znak @, który pozwala jednoznacznie odróżnić rekord SOA od innych rekordów pliku definicji strefy lub odwzorowania odwrotnego.
- ♦ Drugie pole zawiera wartość TTL dla tego rekordu. Mimo że zawsze umieszczam domyślną wartość TTL na początku pliku, dla czytelności powtarzam tę samą wartość w tym miejscu. To pole jest opcjonalne — jeśli nie będzie określone, zostanie zastosowana domyślna wartość TTL serwera lub wewnętrzna wartość domyślna.
- ♦ Trzecie pole identyfikuje klasę rekordu SOA. Pozycja IN informuje, że mamy do czynienia z klasą „Internet”. To pole również jest opcjonalne, ponieważ IN jest klasą domyślną dla każdego rekordu zasobów w pliku definicji strefy lub definicji dla odwrotnych zapytań DNS. Preferuję jednak jawne określanie wartości klasy w celu zwiększenia czytelności. Innymi możliwymi rekordami są HS (Hesiod)

i CH (Chaos), zrozumiałe jedynie dla starszych pracowników MIT lub fanów maszyny Lisp.

- ♦ Czwarte pole identyfikuje nazwę serwera oferującego autorytatywną informację dla danej strefy lub zakresu adresów IP. Ta nazwa wygląda jak standardowa nazwa hosta, ale z reguły kończy się kropką, co stanowi standardowy sposób zasygnalizowania parserowi serwera BIND, że ma do czynienia z w pełni kwalifikowaną domeną internetową. Nazwy nie zakończone kropką są interpretowane jako nazwy lokalne, do których należy dopisać nazwę głównej domeny strefy. Kiedy np. serwer nazw ma nazwę *ns1*, nazwa domeny głównej dla tej strefy to *vonhagen.org*, a adres IP serwera nazw jest identyfikowany w pliku strefy lub w pliku dla zapytań odwrotnych, w tym polu akceptowane są wówczas wpisy *ns1* lub *ns1.vonhagen.org*. (z kropką na końcu). Wpis *ns.vonhagen.org* jest błędny, ponieważ nie kończy się kropką, co spowoduje dopisanie do niego głównej domeny strefy i powstanie adres *ns1.vonhagen.org.vonhagen.org*, który prawdopodobnie nie istnieje w tej domenie. Pomińcie kropki w tym miejscu jest powszechnym błędem i przyczyną dość częstych problemów z plikami stref.
- ♦ Piąte pole to adres e-mail osoby odpowiedzialnej za strefę lub zakres adresów IP. Podobnie jak w przypadku nazw, hostów w pliku strefy, ta pozycja z reguły kończy się kropką, aby zasygnalizować, że jest w pełni kwalifikowaną nazwą domenową. Znak @, wykorzystywany tradycyjnie w adresach e-mail, nie może być użyty w tym miejscu, ponieważ parser plików konfiguracyjnych serwera DNS wykorzystuje go w charakterze skrótu do głównej domeny strefy. Z tego powodu zamiast znaku @ w tym miejscu stosowana jest kropka. Błędy w tym polu są dość powszechne, a ich identyfikacja jest trudna, ponieważ jedynym problemem jest brak wiadomości e-mail od serwera DNS wysyłanych do administratora. Jeśli w dzienniku serwera poczty znajduje się dużo komunikatów o błędach dla adresu e-mail danego użytkownika, warto sprawdzić, czy do rekordu SOA w serwerze BIND nie zakradł się błąd.
- ♦ Szóste pole jest w zasadzie zbiorem pięciu parametrów, oddzielonych białymi znakami i ujętych w nawiasy klamrowe. Wielu administratorów (w tym również ja) dla zwiększenia czytelności wykorzystuje średniki (znaki komentarza) do opisanja każdej pozycji. W tym rekordzie zapisuje się kolejno pięć wartości:
 - ♦ **numer seryjny**: liczba całkowita bez znaku, którą należy modyfikować za każdym razem, gdy wprowadzane są zmiany w strefie lub pliku zapytań odwrotnych; tę wartość z reguły konstruuje się przez połączenie czterech cyfr bieżącego roku, dwóch cyfr miesiąca, dwóch cyfr dnia oraz dwóch cyfr numeru kolejnego od 01 do 99; dzięki temu każdego dnia można dokonać 99 modyfikacji pliku, co powinno w zupełności wystarczyć większości administratorów,
 - ♦ **okres odświeżania**: określa czas w sekundach między odświeżeniami danych przez drugorzędny serwer DNS (*slave*) dla danej strefy,
 - ♦ **okres ponowienia**: określa odstęp czasu w sekundach między kolejnymi próbami skomunikowania się z serwerem nadrzędnym (po upływie okresu odświeżania): ta wartość powinna — oczywiście — być dużo mniejsza od okresu odświeżania,
 - ♦ **okres ważności**: określa czas między kolejnymi udanymi odświeżeniami, po upływie którego dane zarządzane przez serwer podrzędny przestają być

uznawane za autorytatywne; w sytuacji, gdy ich dane są uznane za przestarzałe, a nie ma możliwości skontaktowania się z nadrzędnym serwerem DNS, serwery podrzędne przestają odpowiadać na żądania DNS dotyczące nieaktualnej domeny,

- ♦ **TTL dla informacji negatywnych:** okres czasu przechowywania informacji negatywnych, takich jak brak domeny o poszukiwanej nazwie; serwer BIND do wersji 9. wykorzystywał tę wartość jako domyślną wartość TTL dla strefy lub pliku danych dla zapytań odwrotnych, obecnie do tego celu stosuje się zmienną TTL (o czym dalej).

W kolejnym punkcie opisuję podstawową konstrukcję plików stref i danych dla zapytań odwrotnych.

Tworzenie plików stref

Podstawowy plik strefy składa się z czterech bloków informacji, takich jak:

- ♦ instrukcje \$TTL i SOA opisane w poprzednim podrozdziale;
- ♦ definicja serwera nazw dla danej strefy;
- ♦ definicja serwera poczty dla danej strefy;
- ♦ wpisy dla wszystkich hostów w strefie, definiujące odwzorowania nazw hostów na odpowiednie adresy IP oraz definiujące aliasy (*CNAME*), odwzorowujące jedno nazwy hostów na inne.

Wpis definiujący nazwę serwera nazw ma następującą postać:

```
@ . IN NS ubuntu32.vonhagen.org.
```

Ten wpis identyfikuje nazwę głównej domeny strefy, wykorzystując symbol @, ten sam, który stosuje się w rekordzie SOA. Znajduje się tam również informacja o tym, że mamy do czynienia z rekordem klasy internetowej (IN), definicja jego typu (NS, dla serwera nazw) oraz nazwa hosta serwera nazw. Adres IP serwera nazw jest z reguły zdefiniowany w dalszej części pliku strefy pod warunkiem, że występuje w tej samej domenie lub poddomenie. Podobnie jak w przypadku innych pozycji dotyczących adresów internetowych (rekordy typu IN), nazwa serwera nazw musi kończyć się kropką, inaczej zostanie do niej dołączona nazwa strefy. Sam w rekordach NS wykorzystuję znak @, odwołujący się do nazwy strefy, dzięki czemu do tworzenia plików definicji stref mogę wykorzystać jednolity szablon.

Wpis dotyczący serwera poczty ma składnię zbliżoną do wpisu serwera nazw, z tą różnicą, że w tym przypadku rekord jest typu MX (Mail Exchanger) i jego nazwę można opcjonalnie poprzedzić liczbową wartością określającą preferencję wykorzystania danego serwera. Niższe wartości mają wyższy priorytet. Nie ma to znaczenia gdy używany jest jeden serwer, ale może się przydać w przypadku większej liczby serwerów pocztowych. Oto przykładowy wpis definiujący adres serwera pocztowego:

```
@ IN MX 10 mail.vonhagen.org.
```

Adres IP serwera poczty jest z reguły zdefiniowany w dalszej części pliku strefy pod warunkiem, że znajduje się w tej samej domenie lub jej poddomenie. Podobnie jak w przypadku innych pozycji dotyczących adresów internetowych (rekordy typu IN), nazwa serwera nazw musi kończyć się kropką, inaczej zostanie do niej dołączona nazwa strefy. Sam w rekordach MX również wykorzystuję znak @, odwołujący się do nazwy strefy, dzięki czemu do tworzenia plików definicji stref mogę wykorzystać jednolity szablon.

Po definicji serwera nazw i serwerów poczty następuje z reguły seria rekordów adresów (A) i nazw kanonicznych (CNAME) w danej strefie. Rekordy adresów mają następującą składnię:

```
64bit      IN  A    192.168.6.64
64x2       IN  A    192.168.6.80
ubuntu32   IN  A    192.168.6.90
```

Należy zwrócić uwagę na to, że po skróconych nazwach nie występują kropki, bo w tym przypadku zależy nam na tym, by została do nich dopisana główna domena strefy. Po adresie IP również nie wpisuje się kropki.

Rekordy definiujące nazwy kanoniczne (aliasy nazw hostów) mają następującą składnię:

```
dualcore    IN  CNAME 64x2
```

Po nazwie kanonicznej oraz nazwie wskazywanej nie wpisuje się kropki. Rekordy CNAME wymagają dwóch zapytań DNS: jednego dla aliasu, jednego w celu odczytania adresu IP właściwego hosta, dlatego warto zachować umiar w ich stosowaniu. Rekordy CNAME są wykorzystywane najczęściej w celu odwzorowania powszechnie stosowanych adresów serwerów, takich jak www, ftp itp., i trzeba odwzorować je na rzeczywiste nazwy serwerów.

Oto przykładowa, kompletna zawartość pliku definicji strefy dla mojej domeny *vonhagen.org*:

```
$TTL 1d

@ 1d IN SOA ubuntu32.vonhagen.org. vvh.vonhagen.org. (
    2006070108 ; Serial
    21600      ; Refresh
    1800       ; Retry
    604800     ; Expire
    900        ; Negative Cache TTL
)

@      IN  NS    ubuntu32.vonhagen.org.
@      IN  MX    10 mail.vonhagen.org.

64bit   IN  A    192.168.6.64
64x2    IN  A    192.168.6.80
ubuntu32 IN  A    192.168.6.90
mail    IN  A    207.44.142.34

dualcore IN  CNAME 64x2.vonhagen.org.
www      IN  CNAME mail.vonhagen.org.
ftp      IN  CNAME mail.vonhagen.org.
```

Wpisy dotyczące hostów *mail*, *www* i *ftp* różnią się nieco od pozostałych i wymagają wyjaśnień (oraz, w pewnym sensie, przeprosin). Ponieważ mój dostawca internetu oferuje publiczny serwer nazw, który jest autorytatywny dla domeny *vonhagen.org*, do konfiguracji dopisałem więc adresy IP hostów znajdujących się poza moją lokalną strefą, aby mogły być obsługiwane przez lokalny serwer nazw w charakterze autorytatywnego źródła informacji o mojej domenie. Użytkownicy spoza mojej sieci znajdą adresy tych hostów za pośrednictwem publicznego, autorytatywnego serwera DNS, podczas gdy mogę znaleźć adresy tych serwerów, znajdujących się poza moją siecią, przy pomocy lokalnego serwera DNS. Ten sam efekt można osiągnąć w standardowy lecz nieco bardziej skomplikowany sposób, wykorzystując technikę delegacji, ale ten fortel pozwala zachować czytelność pliku strefy i działa zgodnie z oczekiwaniami.

Tworzenie pliku strefy to zaledwie połowa bitwy o DNS. Po utworzeniu pliku strefy należy utworzyć uzupełniający go plik identyfikujący odwzorowanie adresów IP na odpowiadające im nazwy hostów, czyli plik odwzorowań odwrotnych. Konstrukcję tych plików omawiam w kolejnym punkcie.

Tworzenie pliku odwzorowań odwrotnych DNS

Podstawowy plik danych dla zapytań odwrotnych, nazywany też plikiem odwzorowania odwrotnego (*reverse map file*), składa się z trzech bloków informacji:

- ♦ definicji zmiennej \$TTL i rekordu SOA, omówionych w poprzednim punkcie;
- ♦ wpisu definiującego serwer nazw dla tej strefy;
- ♦ wpisów definiujących odwzorowanie adresu IP na nazwę hosta w strefie.

Serwer nazw w pliku odwzorowania odwrotnego jest zdefiniowany w następujący sposób:

```
@ . IN NS ubuntu32.vonhagen.org.
```

Ten wpis identyfikuje nazwę głównej domeny strefy, wykorzystując symbol @, ten sam, który stosuje się w rekordzie SOA. Znajduje się tam również informacja o tym, że mamy do czynienia z rekordem klasy internetowej (IN), definicja jego typu (NS, dla serwera nazw) oraz nazwa hosta serwera nazw. Adres IP serwera nazw jest z reguły zdefiniowany w pliku strefy pod warunkiem, że występuje w tej samej domenie lub poddomenie. Podobnie jak w innych pozycjach dotyczących adresów internetowych (rekordy typu IN), nazwa serwera nazw musi kończyć się kropką, inaczej zostanie do niej dołączona nazwa strefy. Osobiście w rekordach NS wykorzystuję znak @, odwołujący się do nazwy strefy, dzięki czemu do tworzenia plików definicji stref mogę wykorzystać jednolity szablon.

Wszystkie następne rekordy w typowym pliku odwzorowania odwrotnego są typu PTR i definiują nazwy hostów odpowiadające poszczególnym adresom IP w określonym zakresie adresów. W pełni kwalifikowane rekordy PTR dla adresów IP postaci *AAA.BBB.CCC.DDD* mają następującą postać:

```
DDD.CCC.BBB.AAA.in-addr.arpa. IN PTR hostname.domain.tld.
```

Wpisy w pierwszym polu odpowiadające adresom IP stanowią odwrócenie rzeczywistych adresów IP, ponieważ informują o odwzorowaniu adresu na nazwę, a w tym przypadku najbardziej znaczący jest skrajny prawy fragment adresu IP, najczęściej w sposób unikalny

identyfikujący rekord PTR. Wszystkie rekordy PTR muszą w sposób jawny lub domyślny zawierać rozszerzenie `.in-addr.arpa.` (z kropką na końcu).

Oto kilka przykładowych rekordów PTR w jednym z moich plików odwzorowań odwrotnych:

```
64.6    IN  PTR    64bit.vonhagen.org.
80.6    IN  PTR    64x2.vonhagen.org.
90.6    IN  PTR    ubuntu32.vonhagen.org.
```

Mimo że w pierwszych polach rekordów wykorzystane są jedynie fragmenty adresów IP, są one obsługiwane tak samo jak nazwy, to znaczy w przypadku, gdy nie kończą się kropką, jest do nich dopisywana nazwa głównej domeny strefy (tutaj `168.192.in-addr.arpa`). Nazwy hostów są natomiast zakończone kropkami, co sygnalizuje, że stanowią w pełni kwalifikowane nazwy domenowe.

Oto zawartość przykładowego, kompletnego pliku odwzorowania odwrotnego, definiującego nazwy hostów dla adresów z podsieci `192.168`, zdefiniowanych w pliku odwzorowania jako domena `168.192.in-addr.arpa`:

```
$TTL 1d

@ 1d IN SOA ubuntu32.vonhagen.org. wvh.vonhagen.org. (
    2006070108 ; Serial
    21600      ; Refresh
    1800       ; Retry
    604800     ; Expire
    900        ; Negative Cache TTL
)

@      IN  NS      ubuntu32.vonhagen.org.

64.6   IN  PTR     64bit.vonhagen.org.
80.6   IN  PTR     64x2.vonhagen.org.
90.6   IN  PTR     ubuntu32.vonhagen.org.
```

Utworzenie pliku odwzorowania odwrotnego to druga część bitwy o DNS. Pozostało jedynie wykorzystać je w konfiguracji serwera BIND, co zostanie omówione w kolejnym punkcie.

Instalacja w serwerze BIND pliku strefy i pliku odwzorowania odwrotnego

W poprzednich dwóch punktach objaśniłem składnię pliku strefy i pliku odwzorowania odwrotnego. Pliki utworzone w poprzednich punktach to kolejno *vonhagen.zone*, plik definicji strefy dla domeny *vonhagen.org*, oraz plik odwzorowania odwrotnego *168.192.in-addr.arpa*, w którym znajduje się odwzorowanie adresów IP z podsieci *192.168* na odpowiadające im nazwy w domenie *vonhagen.org*. Aby wykorzystać te pliki, należy dodać je do pliku */etc/bin/named.conf.local* serwera skonfigurowanego jako autorytatywne źródło informacji DNS dla tej domeny i puli adresów. Oto zawartość pliku */etc/bin/named.conf.local* dla mojego przykładowego serwera:

```
zone "vonhagen.org" {
    type master;
    file "/etc/bind/vonhagen.zone";
    check-names warn;
};

zone "168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/168.192.in-addr.arpa";
};

include "/etc/bind/zones.rfc1918.local";
```

Te wpisy włączają pliki i identyfikują ten serwer jako główny serwer dla strefy *vonhagen.org*, wykorzystujący adresy z podsieci *192.168.**. Informacja o strefie *vonhagen.org* jest zdefiniowana w pliku */etc/bind/vonhagen.zone*. Definicja odwzorowań odwrotnych dla adresów IP z podsieci *192.168.** znajduje się w pliku */etc/bin/168.192.in-addr.arpa*. Deklaracje włączające pliki wykorzystują pełne ścieżki, ponieważ są one zapisane w katalogu */etc/bind*, a nie w domyślnym dla serwera BIND katalogu */var/cache/bind*, co jest zdefiniowane w pliku */etc/bin/named.options*. Jak wspomniałem wcześniej, plik */etc/bind/named.conf.local* łączy również zawartość pliku */etc/bin/zones.rfc1918*, z którego usunąłem definicję pustych odpowiedzi dla adresów ze strefy *168.192.in-addr.arpa*, ponieważ prawidłowe wartości dla tego zakresu nieroutowalnych adresów IP są zdefiniowane w pliku */etc/bin/168.192.in-addr.arpa*.

Po zaktualizowaniu pliku */etc/bin/named.conf.local* pozostaje jedynie ponownie załadować zmodyfikowaną konfigurację serwera, zgodnie z opisem w kolejnym podrozdziale.

Ponowne uruchamianie i testowanie serwera DNS

W Ubuntu do uruchamiania serwera BIND służy skrypt */etc/init.d/bind9*, który obsługuje opcje *reload* i *force-reload*, lecz obie z nich wykorzystują program *rndc*, który może nie być poprawnie skonfigurowany w systemie. Z tego powodu z reguły po prostu ponownie uruchamiam serwer BIND, wykorzystując następujące polecenie:

```
$ sudo /etc/init.d/bind9 restart
```

Po wpisaniu hasła na ekranie zostaną wypisane znane komunikaty zamykania i uruchamiania serwera. Następnie należy zweryfikować, czy serwer został prawidłowo uruchomiony, np. sprawdzając istnienie procesu *named*:

```
$ ps alxww | grep named
```

Jeśli plik stref i plik odwzorowania odwrotnego mają prawidłowe składnie, na liście procesów znajdzie się też ten plik odpowiedzialny za pracę serwera DNS. Gratulacje, mamy działający serwer. Jeśli taki proces nie jest uruchomiony w systemie, proponuję zapoznać się z poradami dotyczącymi rozwiązywania problemów z serwerem BIND, opisanymi w kolejnym podrozdziale.

Następnie należy w pliku `/etc/resolv.conf` wpisać odpowiednią pozycję dla nowo uruchomionego serwera DNS. W tym samym systemie wystarczy wpis odwołujący się do tego samego komputera, np.:

```
nameserver 0.0.0.0
```

Jak wspominałem wcześniej, skrót `0.0.0.0` odnosi się do adresu IP pierwszego interfejsu sieciowego uruchomionego w systemie. Jest to najczęściej adres interfejsu *loopback* o adresie `127.0.0.1` (ale nie zawsze). Wykorzystanie adresu `0.0.0.0` spowoduje wysłanie żądania DNS na adres IP lokalnego systemu, niezależnie od tego, jaki w rzeczywistości jest to adres.

Gdy upewnimy się, że serwer nazw działa prawidłowo, do plików `/etc/resolv.conf` w systemach, które będą korzystać z tego serwera DNS, należy wpisać rzeczywisty adres IP serwera.

Najwyższy czas przetestować serwer DNS w działaniu. Do testowania serwera można wykorzystać kilka poleceń. Oto najpopularniejsze z nich.

- ♦ `dig`: narzędzie zalecane do testowania konfiguracji serwerów DNS. Wyniki działania tego programu są dość rozbudowane, dlatego narzędzie wymaga nieco czasu, aby się do niego przyzwyczaić.
- ♦ `host`: najprostsze narzędzie do testowania konfiguracji DNS.
- ♦ `nslookup`: narzędzie to, choć dość użyteczne i nadal dostępne w wielu systemach, zostało oficjalnie uznane za przestarzałe. Niestety, pewnego dnia program `nslookup` przestanie wypisywać komunikaty o tym, że jest przestarzały, ponieważ po prostu zniknie z naszych systemów.

Wszystkie polecenia są łatwe w użyciu. Poniżej przedstawiam sesję testową wykorzystującą każdy z tych programów do weryfikacji istnienia hosta 64bit, skonfigurowanego wcześniej w tym rozdziale w ramach testowego serwera DNS:

```
$ host 64bit
64bit.vonhagen.org has address 192.168.6.64
```

```
$ nslookup 64bit
Server:      127.0.0.1
Address:     127.0.0.1#53
```

```
Name:   64bit.vonhagen.org
Address: 192.168.6.64
```

```
$ dig 64bit.vonhagen.org
; <>> DiG 9.3.2 <>> 64bit.vonhagen.org
;; global options: printcmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 47870
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;64bit.vonhagen.org.      IN      A
```



```
:: ANSWER SECTION:
64bit.vonhagen.org.      86400   IN      A       192.168.6.64

:: AUTHORITY SECTION:
vonhagen.org.           86400   IN      NS      ubuntu32.vonhagen.org.

:: ADDITIONAL SECTION:
ubuntu32.vonhagen.org.   86400   IN      A       192.168.6.90

:: Query time: 1 msec
:: SERVER: 127.0.0.1#53(127.0.0.1)
:: WHEN: Wed Jul  5 01:18:21 2006
:: MSG SIZE   rcvd: 91
```

Wyniki działania poleceń `host` i `nslookup` nie wymagają wyjaśnień. Wynik działania programu `dig` jest najbardziej obszerny i użyteczny w procesie diagnozowania przyczyn problemów z konfiguracją serwera DNS. Program `dig` określa dokładne zapytanie, jakie zostaje wysłane do serwera, oraz otrzymaną od niego odpowiedź. W wyniku można znaleźć również informację o serwerze nazw, który udzielił odpowiedzi. Program `dig` obsługuje wiele opcji, pozwalających dostosować wyniki jego działania do konkretnych potrzeb, szczególnie można znaleźć w podręczniku użytkownika (`man dig`).

Rozwiązywanie problemów z serwerami DNS

Jedną z zalet serwera BIND 9 jest to, że proces `named` nie zostanie uruchomiony w przypadku, gdy w którymkolwiek z plików konfiguracyjnych wystąpią poważne błędy. To może być dość irytujące, ale pozwala uniknąć sytuacji, gdy uruchomimy nie do końca skonfigurowany, publicznie dostępny serwer DNS. Ponadto definicja „poważnych błędów” jest dość płynna. Jednym z przykładów poważnych błędów są błędy składni, przeszkadzające w odpowiedniej interpretacji pliku konfiguracyjnego, natomiast inne sytuacje, które należałoby uznać za poważne, takie jak brak niektórych plików czy błędne wartości w rekordach, jako poważne już nie są uznawane.

Jeśli serwer BIND nie zostanie uruchomiony prawidłowo, w pierwszej kolejności należy sprawdzić zawartość odpowiedniego pliku dziennika, w tym przypadku `/var/log/syslog`. Domyślnie BIND w wyniku zdarzeń, takich jak uruchamianie i zamykanie procesu czy przeładowanie konfiguracji, zapisuje w dzienniku różne informacje. Oto przykładowa zawartość pliku `/var/log/syslog` zawierająca komunikaty serwera BIND (usunąłem znaczniki czasu i identyfikację hosta, które znajdują się na początku wierszy):

```
named[19897]: starting BIND 9.3.2 -u bind
named[19897]: found 1 CPU, using 1 worker thread
named[19897]: loading configuration from '/etc/bind/named.conf'
named[19897]: /etc/bind/named.conf:60: missing ';' before end of file
named[19897]: loading configuration: failure
named[19897]: exiting (due to fatal error)
```

Wyraźnie widać, że brakuje średnika w wierszu 60. pliku `named.conf`. Po poprawieniu błędu należy spróbować uruchomić serwer, aby sprawdzić, czy był to jedyny błąd w pliku.

W Ubuntu, w pakiecie *bind* można znaleźć narzędzie *named-checkconf*, które służy do weryfikowania prawidłowości plików konfiguracyjnych. Uruchomienie tego programu pozwala w prosty sposób sprawdzić składnię plików konfiguracyjnych programu BIND, bez konieczności ponownego uruchamiania procesu *named*, sprawdzania dziennika systemowego i naprawiania błędów jeden po drugim, powtarzając za każdym razem tę samą sekwencję działań.

W pliku */var/log/syslog* można dość często znaleźć następujący komunikat:

```
named[20222]: zone vonhagen.org/IN: loading master file: file not found
```

Co dziwne, brak pliku *vonhagen.zone* definiującego strefę *vonhagen.org* nie jest uznany za błąd krytyczny, zatem BIND kontynuuje działanie, po prostu nie zna informacji na temat domeny *vonhagen.org*. Oprócz błędów składni w tym pliku, najczęstszą przyczyną problemów jest wykorzystanie ścieżki względnej zamiast bezwzględnej. Jak wspominałem wcześniej, BIND w Ubuntu domyślnie wykorzystuje jako katalog główny */var/cache/bind*. Wszelkie pliki wskazane przez ścieżką względną będą zatem poszukiwane w tym katalogu, a nie w */etc/bind*, gdzie znajdują się domyślnie.

Program *named-checkconf* sprawdza składnię pliku */etc/bind/named.conf* i innych plików konfiguracyjnych, ale nie sprawdza składni plików definicji stref. Na szczęście, pakiet *bind* w Ubuntu zawiera jeszcze jedno narzędzie, *named-checkzone*, służące właśnie do tego celu. Choć ten program obsługuje kilka różnych opcji, domyślnie program *named-checkzone* wykorzystuje się z wywołaniem dwóch parametrów: nazwy strefy oraz ścieżki do pliku definiującego tę strefę. Może to być standardowy plik strefy lub plik odwzorowania odwrotnego, ponieważ oba określają dwa typy stref logicznych. Oto kilka przykładowych błędów zgłaszanych przez program *named-checkzone*:

```
$ named-checkzone vonhagen.org vonhagen.zone
dns_rdata_fromtext: vonhagen.zone:8: near '@': extra input text
zone vonhagen.org/IN: loading master file vonhagen.zone: extra input text
```

```
$ named-checkzone 168.192.in-addr.arpa 168.192.in-addr.arpa
168.192.in-addr.arpa:11: unknown RR type 'ubuntu32.vonhagen.org.'
zone 168.192.in-addr.arpa/IN: loading master file: unknown class/type
```

Bez obaw, te błędy nie występują w przedstawionych przeze mnie przykładach plików konfiguracyjnych, popsułem je nieco jedynie dla celów demonstracyjnych. Przykłady przedstawione wcześniej w tym rozdziale są prawidłowe.

Pierwszy błąd, *extra input text*, wskazuje na błąd analizy leksykalnej podczas odczytu rekordów w pliku, tym razem rekordu SOA. Aby sprowokować ten komunikat o błędzie, usunąłem zamykający nawias klamrowy rekordu SOA, co spowodowało, że parser traktował kolejny wiersz jako dalszą część tego rekordu, przez co poinformował o nieoczekiwanym tekście rozpoczynającym się od znaku @.

Kolejne błędy, *unknown RR type 'ubuntu32.vonhagen.org'* oraz *unknown class/type*, wymagają nieco więcej uwagi. Oba informują o błędach analizy leksykalnej w ramach tego samego rekordu, tym razem w wierszu 11. Chodzi tu o deklarację serwera nazw w pliku odwzorowania odwrotnego. Jak wspominałem wcześniej w tym rozdziale, parser serwera BIND jest dość kapryśny. Mimo że radzi sobie doskonale z podstawianiem wartości domyślnych, gdy jest to konieczne, to podczas stosowania białych znaków w pliku

należy zachować szczególną ostrożność. W tym przypadku usunąłem znak @ oraz wszelkie białe znaki na początku tego rekordu, przez co na początku wiersza znalazła się deklaracja klasy rekordu IN. Ten ciąg znaków na początku wiersza, niepoprzedzony białymi znakami, został zinterpretowany jako nazwa głównej domeny serwera nazw, ciąg znaków NS został zinterpretowany jako deklaracja klasy rekordu, a nazwa serwera, czyli `ubuntu32.vonhagen.org.`, została odczytana jako specyfikacja typu rekordu. Próba rozwiązania tego problemu spowodowała również pojawienie się w pliku `/var/log/syslog` komunikatu `zone 168.192.in-addr.arpa/IN: has no NS records (strefa 168.192.in-addr.arpa/IN: nie ma zdefiniowanych rekordów NS)`, co stanowi efekt uboczny błędnej interpretacji tego rekordu.

Próby rozwiązywania błędów analizy leksykalnej mogą być nieco frustrujące. W najgorszym przypadku można zdecydować się na jawne zdefiniowanie każdej wartości, sprawdzenie, czy plik w takiej postaci działa prawidłowo, po czym stopniowe upraszczanie pliku, aż do zadowalającej postaci. Na potrzeby porównawcze często przydatne są poprawne, działające przykłady, takie jak przedstawione wcześniej w tym rozdziale.

Błędy leksykalne i składniowe powodują, że plik konfiguracyjny nie załaduje się i nie będzie prawidłowo zinterpretowany. Oczywiście, również po wczytaniu i zinterpretowaniu tych plików może okazać się, że serwer nadal nie będzie działał prawidłowo. Najczęstszą przyczyną problemów z plikami definicji stref jest brak kropki po w pełni kwalifikowanej nazwie domenowej lub postawienie kropki na końcu nazwy skróconej, do której musi być dopisana główna domena strefy. Nazwy niekończące się kropką są traktowane jako nazwy lokalne i parser dopisuje do nich na końcu główną domenę strefy. Gdy np. nazwa serwera nazw jest zdefiniowana jako `ns1`, nazwa domeny głównej dla tej strefy to `vonhagen.org`, a adres serwera nazw znajduje się w pliku strefy lub jej odwzorowania odwrotnego, to w polu nazwy serwera nazw można wpisać zarówno `ns1`, jak i `ns1.vonhagen.org.` (z kropką na końcu). Pierwsza z tych nazw stanowi formę skróconą i nie może kończyć się kropką, co spowoduje automatyczne dopisanie domeny głównej. Druga forma to w pełni kwalifikowana nazwa domenowa, zatem nie należy dopisywać do niej żadnych informacji. Natomiast wpis `ns.vonhagen.org` jest całkowicie nieprawidłowy, ponieważ nie kończy się kropką, co spowoduje, że parser dopisze do niej główną domenę strefy, w wyniku czego powstanie `ns1.vonhagen.org.vonhagen.org`, czyli prawdopodobnie nieistniejąca nazwa hosta w danej domenie.



Jeśli ktoś używa serwera DNS w systemie skonfigurowanym za pomocą serwera DHCP (najczęściej wykorzystuje się do tego statyczne odwzorowanie adresów MAC na adresy IP), po każdorazowym uzyskaniu konfiguracji DHCP skrypt `/etc/dhcp3/dhclient-script` nadpisze zawartość pliku `/etc/resolv.conf` listą serwerów DNS pobraną z serwera DHCP. Najlepiej zatem dopisać nowo utworzony serwer DNS do konfiguracji serwera DHCP. Jeśli nie ma takiej możliwości, do pliku `/etc/dhcp3/dhclient.conf` można dopisać następujący wiersz:

```
prepend domain-name-servers 0.0.0.0;
```

To spowoduje, że lokalny serwer DNS będzie automatycznie wpisywany na pierwszej pozycji w pliku `/etc/resolv.conf`. Alternatywnie można zablokować (wstawiając znak komentarza) wywołanie funkcji `make_resolv_conf()` w kodzie skryptu `/etc/dhcp3/dhclient-script`, co spowoduje, że klient DHCP nie będzie modyfikował zawartości pliku `/etc/resolv.conf`, ale to wiąże się z zablokowaniem pewnych aspektów dynamiczności mechanizmu DHCP.

W tym punkcie opisałem kilka moich „ulubionych” błędów konfiguracji serwera BIND i plików stref oraz sposoby ich identyfikacji. Z pewnością można trafić na zupełnie inne komunikaty o błędach — niestety — to nieodłączny element procesu konfiguracji i uruchamiania serwera BIND. Należy się na to przygotować. Ale nie warto panikować, bo przeglądarka WWW w połączeniu z dobrą wyszukiwarką pozwalają znaleźć odpowiedzi na prawie każde pytanie.

Więcej informacji na temat DNS i serwera BIND

Jak wspomniałem wcześniej, DNS i BIND to skomplikowane zagadnienia, o których napisano pokaźne książki. W tym rozdziale przedstawiłem przegląd technologii DNS i serwera BIND oraz wyjaśniłem prostą konfigurację wraz ze sposobami rozwiązywania popularnych problemów. Dodatkowe informacje można znaleźć we wspomnianych wcześniej książkach oraz w następujących źródłach informacji w internecie:

- ♦ *BIND 9 Administrator Reference Manual* (<http://www.net.cmu.edu/groups/netdev/docs/bind9/Bv9ARM.html>): kopia online oficjalnego podręcznika administratora serwera BIND napisanego przez członków projektu BIND,
- ♦ *Pro DNS and BIND* (<http://www.netwidet.net/books/apress/dns/>): kopia online jednej z książek poświęconych tematyce DNS/BIND, wspomnianych wcześniej w tym rozdziale,
- ♦ *DNS Resources Directory* (<http://www.dns.net/dnsrd/>): katalog zasobów zawierający odnośniki do ogromnej kolekcji materiałów poświęconych serwerowi BIND.

Te zasoby pozwolą znaleźć odpowiedzi na wiele zapytań dotyczących tematyki DNS.

Podsumowanie

W niniejszym rozdziale omówiłem mechanizmy odwzorowania nazw hostów na ich adresy IP (i odwrotnie) w elastyczny, rozproszony sposób oraz przedstawiłem sposób instalacji i konfiguracji serwera nazw obsługującego sieć lokalną i pozwalającego jednocześnie na wykorzystanie ogromnej ilości serwerów w Internecie. W rzeczywistości w sieciach lokalnych o większych rozmiarach warto zainstalować większą liczbę serwerów nazw, aby zabezpieczyć się przed błędami pojedynczych systemów, ale w tym rozdziale skupiłem się na instalacji pojedynczego serwera nazw, co ma sens w sieciach mniejszych rozmiarów.

W rozdziale 30. kontynuuję przegląd istotnych mechanizmów serwerowych, wyjaśniam bowiem, w jaki sposób zainstalować i skonfigurować serwer wydruków. Jeśli ktoś nie ma za dużo pieniędzy i powierzchni biurowej, z pewnością zechce tak zorganizować pracę, by większa liczba stacji roboczych mogła współużytkować pewną ograniczoną liczbę drukarek, i w takim celu przyda się konfiguracja serwerów wydruku. To podejście pozwala oszczędzić pieniądze, ale również pomaga zminimalizować liczbę miejsc, w których użytkownicy będą szukać swoich zaginionych wydruków.

Rozdział 30.

Konfiguracja serwera wydruków

W tym rozdziale:

- ♦ Uaktywnianie zdalnego dostępu do serwera wydruku
- ♦ Integracja systemów Microsoft Windows
- ♦ Integracja systemów Apple Mac OS X
- ♦ Rozwiązywanie problemów

Jeśli posiada się jeden komputer i jedną drukarkę, konfiguracja podsystemu wydruków jest prostym zadaniem, które zostało omówione w podrozdziale „Dodawanie drukarek”, w rozdziale 23. Jednak we współczesnych, skomplikowanych środowiskach sieciowych istnieje spora szansa, że konieczny okaże się dostęp do drukarki przyłączonej do sieci w zupełnie innym miejscu, w tym również do systemów z działającym systemem operacyjnym innym niż Ubuntu Linux.

W tym rozdziale omawiam konfigurację systemu w taki sposób, aby użytkownicy innych systemów mogli korzystać z drukarki przyłączonej lokalnie. W tym rozdziale można również znaleźć opis sposobów korzystania z drukarki udostępnionej w systemie Ubuntu Linux z systemów Microsoft Windows i Mac OS X. Na końcu rozdziału umieściłem wskazówki dotyczące rozwiązywania częstych problemów oraz sposobów poszukiwania pomocy i informacji na omówione tematy.



Konfiguracja systemu Ubuntu w sposób opisany w tym rozdziale jest zalecana wyłącznie w środowiskach domowych lub małych biur, dobrze odseparowanych od internetu za pomocą zapory sieciowej, w których można ufać wszystkim użytkownikom. W sieciach z zainstalowanymi systemami Windows można po prostu skonfigurować serwer Samba i za jego pomocą udostępnić drukarkę lokalną. Ta metoda umożliwi wykorzystanie istniejących w systemie Windows mechanizmów uwierzytelniania (zależnych od sposobu konfiguracji Samby, co zostało omówione w punkcie „Konfiguracja uwierzytelniania Samby” w rozdziale 32. Metoda pozwala również na wykorzystanie drukarek z systemu Mac OS X, który posiada wbudowane mechanizmy użycia drukarek systemu Windows, czyli protokołu SMB.

Historia drukowania w systemach Linux i Unix

Współczesne drukarki to najczęściej wysokiej jakości drukarki laserowe lub atramentowe, często wyposażone w możliwość wydruku w kolorze o jakości zbliżonej do fotograficznej. Oryginalny system wydruku dla Linuksa, LPD (ang. *Line Printer Daemon*), był opracowany z myślą o kolejkowaniu i zarządzaniu zadaniami wykonywanymi na drukarkach tekstowych. Z czasem, gdy opracowywano coraz bardziej zaawansowane drukarki umożliwiające generowanie wydruków wysokiej jakości (np. drukarki laserowe Xerox x9700, Canon-CX, czy Imagen-300), nadal wykorzystywano oryginalny system wydruku LPD, z tym że wydruki musiały być przetwarzane w taki sposób, by zawierały polecenia w wewnętrznym języku drukarki. Ta metoda szybko stała się bardzo niewygodna, ponieważ zmuszała użytkowników do zdobycia umiejętności dopasowania odpowiednich języków drukarek do ich konkretnych modeli. Wkrótce system LPD został zmodyfikowany i powstał system LP, który pozwalał automatyzować zadanie translacji wydruku na wewnętrzny język drukarki z użyciem odpowiednich filtrów (zwanymi również sterownikami wydruku), a te realizowały funkcję formatowania i wykonywania innych operacji na wydruku przed wysłaniem go do drukarki. Pojawiły się też inne systemy wydruku, takie jak lprng (lpr, Next Generation, nazwa oparta na nazwie polecenia wydruku, znanego z systemu LPD). Zarówno LPD, jak i lprng są dostępne w repozytoriach Ubuntu ze względu na kompatybilność (lub po prostu dlatego, że ktoś je zna i preferuje).

Ewolucja systemów wydruku dla systemów Unix doprowadziła do braku zgodności między systemami wydruku, co z kolei prowadziło do konieczności ponownej kompilacji filtrów wydruku dla różnych wersji Uniksów (oczywiście pod warunkiem, że dostępny był kod źródłowy filtra). W końcu firma Easy Software Products rozpoczęła projekt uogólnionego systemu wydruków dla Uniksów, Linuksów i pokrewnych, który został nazwany w skrócie CUPS (ang. *Common Unix Printing System*). Oryginalna wersja systemu CUPS wykorzystywała standardowy protokół LPD, ale szybko zdecydowano się na przejście na nowy standard, IPP (ang. *Internet Printing Protocol*), dzięki czemu drukarki obsługiwane przez CUPS mogą być wykorzystywane przez systemy nieuniksośowe, takie jak Windows. Firma Easy Software Products posunęła się też do tego, aby kod źródłowy podsystemu CUPS udostępnić na wolnej licencji GPL, co pozwoliło na pełną, międzyplatformową kompatybilność i popularyzację wśród milionów użytkowników i administratorów. I ta strategia zadziałała doskonale — obecnie mechanizm CUPS jest wykorzystywany we wszystkich najważniejszych dystrybucjach Linuksa i w większości innych systemów uniksowych.

Sporo dystrybucji Linuksa, włączając w to Ubuntu, posiada własne mechanizmy konfiguracji drukarek i wykonywania podstawowych operacji w systemie wydruku. Ubuntu wykorzystuje narzędzie konfiguracji drukarek środowiska GNOME, opisane w rozdziale 23., w podrozdziale „Dodawanie drukarek”. CUPS, oprócz standaryzacji podsystemu wydruków w Linuksie, oferuje jeszcze jeden, cenny prezent dla administratorów, czyli wbudowane narzędzie administracyjne do konfiguracji całego mechanizmu CUPS, obsługiwane za pośrednictwem przeglądarki WWW na porcie 631. W tym rozdziale skupię się na standardowych narzędziach Ubuntu i środowiska GNOME służących do konfigurowania systemu wydruku.

Konfiguracja zdalnych hostów do wykorzystania serwera wydruków CUPS

W większości dystrybucji Linuksa (w tym również w starszych wersjach Ubuntu) serwery CUPS oczekują na połączenia na porcie 631 zewnętrznego adresu IP komputera. Z pewnych powodów ta sytuacja została zmieniona i od wersji 6.06 w Ubuntu serwer CUPS domyślnie nasłuchuje wyłącznie na urządzeniu *loopback*, czyli na adresie *127.0.0.1*, co w praktyce uniemożliwia drukowanie z sieci. Na szczęście, przywrócenie możliwości drukowania sieciowego wymaga tylko drobnych modyfikacji.



Aby uniknąć przepisywania pełnej dokumentacji podsystemu wydruków CUPS, opisana konfiguracja jest zupełnie podstawowa, w szczególności nie wykorzystuje mechanizmów uwierzytelniających. Taka konfiguracja nie jest zalecana w systemach stosujących sieci WAN lub w środowiskach akademickich.

Aby uaktywnić drukowanie sieciowe w systemie Ubuntu, należy do listy portów, na których nasłuchuje serwer, dodać zewnętrzny adres IP hosta, który ma oferować usługę drukowania CUPS innym systemom w sieci. Oto, jak można to zrobić.

1. Za pomocą polecenia `sudo` i edytora tekstów (np. `emacs`) należy poddać edycji plik `/etc/cups/cups.d/ports.conf`:

```
$ sudo emacs /etc/cups/cups.d/ports.conf
```

Po podaniu hasła w edytorze zostanie otwarty plik; trzeba odszukać w nim następujący fragment:

```
Listen localhost:631
Listen /var/run/cups/cups.sock
```

2. Teraz trzeba dopisać wiersz informujący serwer o tym, że powinien nasłuchiwać na porcie 631 zewnętrznego interfejsu sieciowego (Ethernet). Przykładowo host wykorzystywany przeze mnie do testów nosi nazwę domenową *ubuntu32.vonhagen.org* i ma adres IP *192.168.6.90*, zatem w pliku `ports.conf` dopisałem następujące dwa wiersze:

```
Listen ubuntu32.vonhagen.org:631
Listen 192.168.6.90:631
```

Pełna zawartość tego pliku będzie zatem następująca:

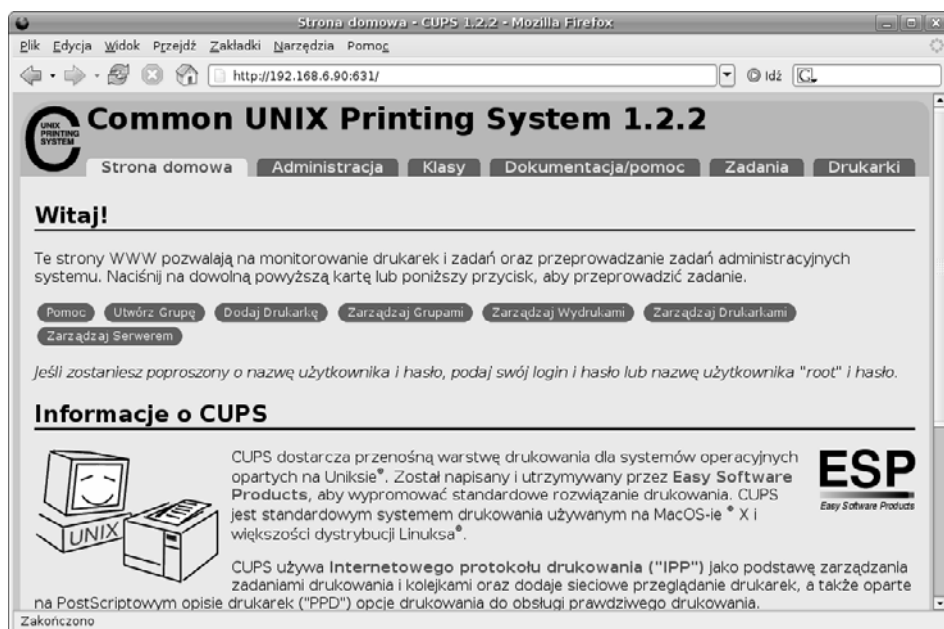
```
Listen localhost:631
Listen ubuntu32.vonhagen.org:631
Listen /var/run/cups/cups.sock
```

Zapisujemy plik i zamykamy edytor tekstu.

3. Uruchamiamy ponownie demona CUPS, wywołując następujące polecenie:

```
$ sudo /etc/init.d/cupsys restart
```

W tym momencie serwer CUPS powinien być dostępny z innych hostów. W ramach szybkiego testu można spróbować wyświetlić interfejs administracyjny w przeglądarce innego hosta w sieci. Wystarczy wywołać adres URL w postaci `http://hostname:631`, ponieważ serwer CUPS nadal nasłuchuje wyłącznie na porcie 631. Jeśli konfiguracja jest prawidłowa, pojawi się strona powitalna systemu CUPS, przedstawiona na rysunku 30.1.



Rysunek 30.1. Interfejs WWW serwera CUPS

W tym momencie można już utworzyć konfigurację drukarek sieciowych na hostach sieci lokalnej. Dodawanie drukarek w Ubuntu zostało omówione w rozdziale 23. Zainteresowani wykorzystaniem drukarki obsługiwanej przez CUPS w systemach Microsoft Windows lub Apple Mac OS X znajdą opisy odpowiednich konfiguracji w dalszej części rozdziału.



Gdy w zdalnym systemie Linux konfiguruje się drukarkę obsługiwaną przez CUPS, należy zastosować adres URL w formacie Internet Printing Protocol o następującej postaci:

```
ipp://hostname/printers/queueName
```

Gdy np. chcemy skonfigurować połączenie z drukarką o nazwie SuperScript-660i działającą w systemie *ubuntu32.vonhagen.org* zarządzaną przez podsystemu CUPS, w adresie sieciowym drukarki trzeba napisać `ipp://ubuntu32.vonhagen.org/printers/SuperScript-660i`. W zależności od typu użytkowanego systemu Linux, konfiguracja można wymagać osobnego podania adresu IP i nazwy kolejki. W takim przypadku dla powyższego przykładu nazwa kolejki to *printers/ SuperScript-660i*.

Uaktywnianie opcji administracyjnych w interfejsie WWW podsystemu CUPS

Wykonywanie zadań administracyjnych z użyciem interfejsu WWW podsystemu CUPS jest wyłączone w domyślnej instalacji Ubuntu, ale funkcję tę można z łatwością uaktywnić. Najprościej dodać użytkownika *cupsys* do grupy *shadow*, dopisując odpowiedni wpis w pliku */etc/group* lub wywołując następujące polecenie:

```
$ sudo adduser cupsys shadow
```


Po wywołaniu tego polecenia należy ponownie uruchomić program CUPS, aby odczytał zaktualizowaną informację o przynależności do grup:

```
$ sudo /etc/init.d/cupsys restart
```

To powinno wystarczyć, aby umożliwić użytkownikowi wykonywanie operacji administracyjnych z użyciem interfejsu WWW, takich jak np. dodawanie drukarek. Po wyświetleniu prośby o podanie nazwy użytkownika i hasła można wpisać dowolnego członka grupy *lpadmin* (użytkownik systemu zakładany w procesie instalacji jest automatycznie do niej dodawany). To jednak nie zawsze wystarcza do wykorzystania wszystkich funkcji administracyjnych interfejsu WWW (trzeba wziąć pod uwagę konfigurację systemu i wersję CUPS). Więcej informacji można znaleźć w punkcie „Odblokowanie dostępu do wybranych funkcji interfejsu administracyjnego WWW podsystemu CUPS”, w dalszej części rozdziału.

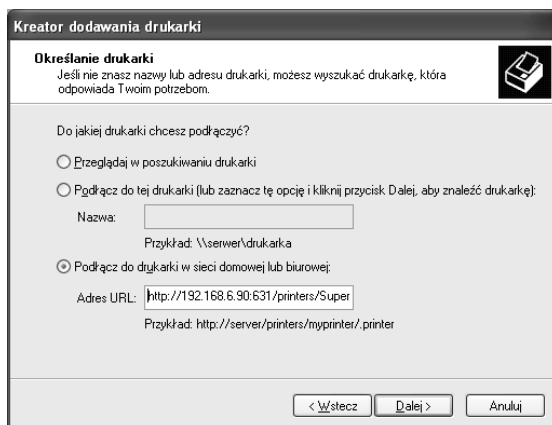
Integracja podsystemu wydruków Windows z serwerem wydruków Ubuntu

Konfiguracja systemu Microsoft Windows 2000 lub XP tak, by mógł wykorzystania ze zdalnej drukarki udostępnianej przez Ubuntu jest tak prosta jak konfiguracja każdej innej drukarki sieciowej. W poniższym przykładzie omawiam taką konfigurację w systemie Windows XP, posiłkując się zrzutami ekranu; we wcześniejszych wersjach systemu Windows procedura jest zbliżona. Oto opis poszczególnych kroków procedury.

1. Z menu systemowego należy wywołać funkcję *Start/Ustawienia/Drukarki i faksy*.
2. Później trzeba dwukrotnie kliknąć ikonę *Dodaj drukarkę*. Pojawi się okno kreatora, w którym należy kliknąć przycisk *Dalej*.
3. Teraz trzeba zaznaczyć opcję *Drukarka sieciowa* i kliknąć przycisk *Dalej*. Pojawi się okno przedstawione na rysunku 30.2.

Rysunek 30.2.

Wprowadzanie adresu URL drukarki udostępnianej przez Ubuntu

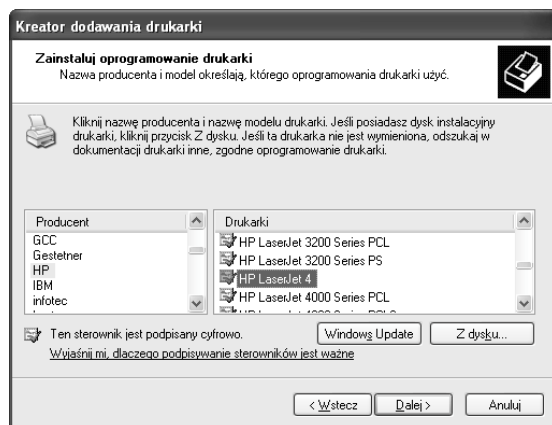


4. Dalej należy wprowadzić adres URL w postaci `http://host:631/printers/nazwa_kolejki`, np. w celu podłączenia się do drukarki SuperScript-660i przyłączonej

do systemu *ubuntu32.vonhagen.org* należy wpisać URL `http://ubuntu32.vonhagen.org:631/printers/SuperScript-660i`. Po wpisaniu URL-a trzeba kliknąć przycisk *Dalej*. Zostanie wyświetlone okno przedstawione na rysunku 30.3.

Rysunek 30.3.

Identyfikacja
producenta i modelu
drukarki



5. Następnie należy wybrać producenta i model drukarki, aby system Windows zastosował właściwy sterownik wydruku. Jeśli producenta lub modelu zdalnej drukarki nie można znaleźć w tym oknie, można wybrać model zbliżony lub wykorzystujący kompatybilny sterownik. Można, oczywiście, wykorzystać sterowniki dostarczone na dysku CD, który wypadł z pudełka przy rozpakowywaniu drukarki. Po dokonaniu wyboru sterownika należy kliknąć przycisk *OK*.
6. Pojawi się okno potwierdzenia, w nim trzeba kliknąć przycisk *Zakończ*, co spowoduje ponowne wyświetlenie okna *Drukarki i faksy*.

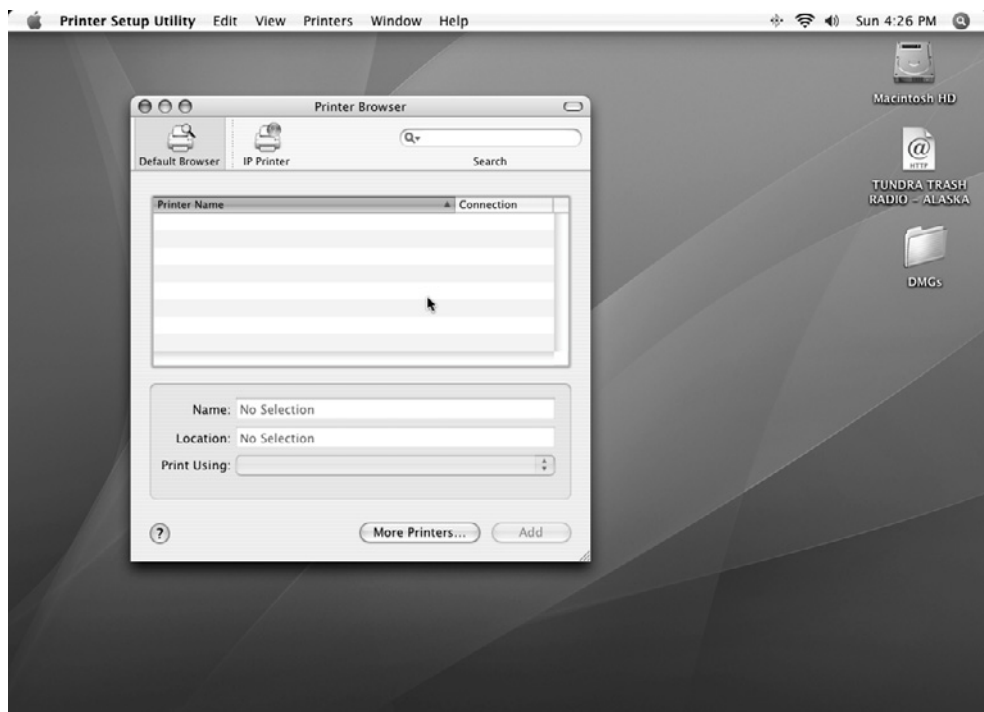
To już wszystko! Aby upewnić się, czy drukarka działa prawidłowo, wystarczy kliknąć prawym przyciskiem myszy ikonę nowo utworzonej drukarki, wybrać funkcję *Właściwości* i kliknąć przycisk *Drukuj stronę testową*. Drukarka powinna wydrukować standardową stronę testową drukarki.

Integracja podsystemu wydruków Mac OS X z serwerem wydruków Ubuntu

System Apple Mac OS X przebył długą drogę od poprzedniej wersji systemu Mac OS. Wersja Mac OS X jest prawdziwym systemem uniksowym wyposażonym w niezwykle estetyczny interfejs użytkownika. Mac OS X wykorzystuje znaczną ilość oprogramowania open source, integracja tego systemu z serwerem wydruków Ubuntu odbywa się za pomocą narzędzi znanych wszystkim zwolennikom systemów Mac OS X, takich jak *Printer Setup* i *Printer Browser*. Zrzuty ekranu zostały wykonane w systemie Mac OS X 10.4, ale procedura jest podobna także we wcześniejszych wersjach systemu Mac OS X.

Aby w systemie Mac OS X utworzyć definicję drukarki udostępnianej przez Ubuntu, należy wykonać kolejno następujące kroki.

1. Trzeba uruchomić *Printer Browser*. Można to zrobić na kilka sposobów. Osobiście z reguły uruchamiam folder *Applications* z okna Findera, otwieram folder *Utilities*, klikam *Printer Setup Utility* i z menu wybieram funkcję *Printer Browser*. Pojawia się okno przedstawione na rysunku 30.4, na razie bez zawartości.



Rysunek 30.4. Uruchomienie narzędzia Przeglądarka drukarek w systemie Mac OS X

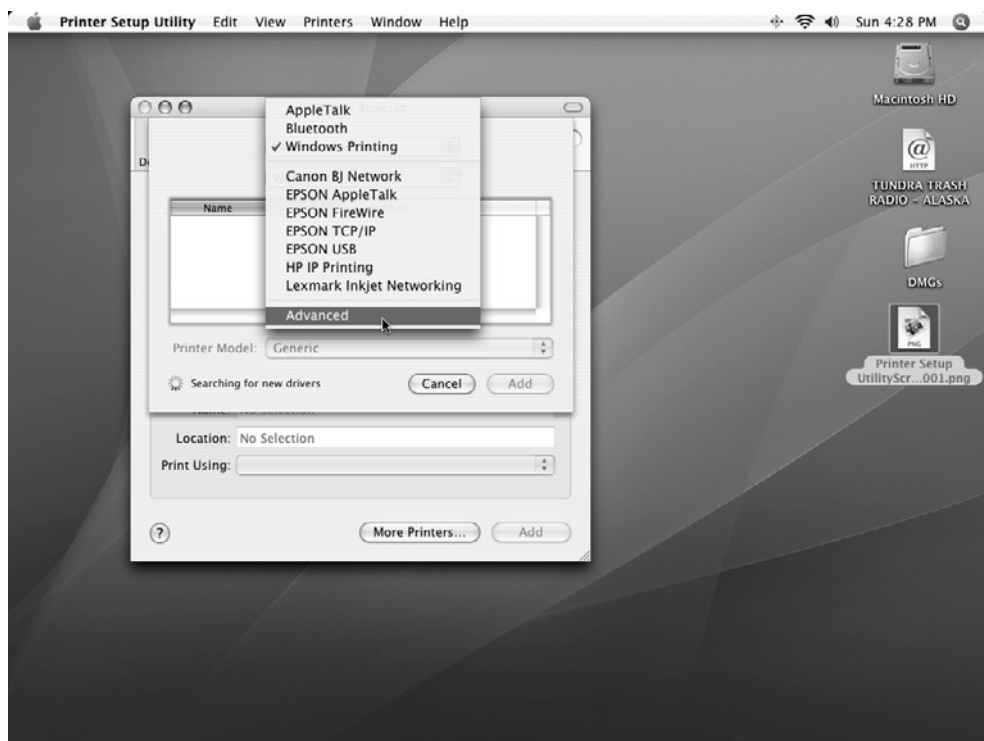


Użytkownicy systemu Mac OS X 10.3 i wcześniejszych powinni przy kliknięciu przycisku *Add* przytrzymać klawisz *Option*, pomijając też punkt 2. niniejszej listy czynności.

2. Trzeba przytrzymać jeden z klawiszy *Option* i kliknąć przycisk *More Printers....*
3. Z listy rozwijanej po prawej stronie okna należy wybrać opcję *Advanced*, co przedstawiam na rysunku 30.5.
4. Z listy rozwijanej *Device* trzeba wybrać opcję *Internet Printing Protocol* z użyciem *IPP*. To spowoduje wyświetlenie okna przedstawionego na rysunku 30.6.
5. Teraz trzeba wprowadzić dowolną nazwę w polu *Device Name* oraz URI urządzenia (ang. *Universal Resource Indicator*) dodawanej drukarki udostępnianej przez Ubuntu w postaci:

```
ipp://host:631/printers/nazwa_kolejki
```

Przykładowo w celu zdefiniowania połączenia z drukarką o nazwie SuperScript-660i udostępnianą przez system *ubuntu32.vonhagen.org* należy wpisać URI
`ipp://ubuntu32.vonhagen.org:631/printers/SuperScript-660i`.



Rysunek 30.5. Uruchamianie zaawansowanej konfiguracji drukarek

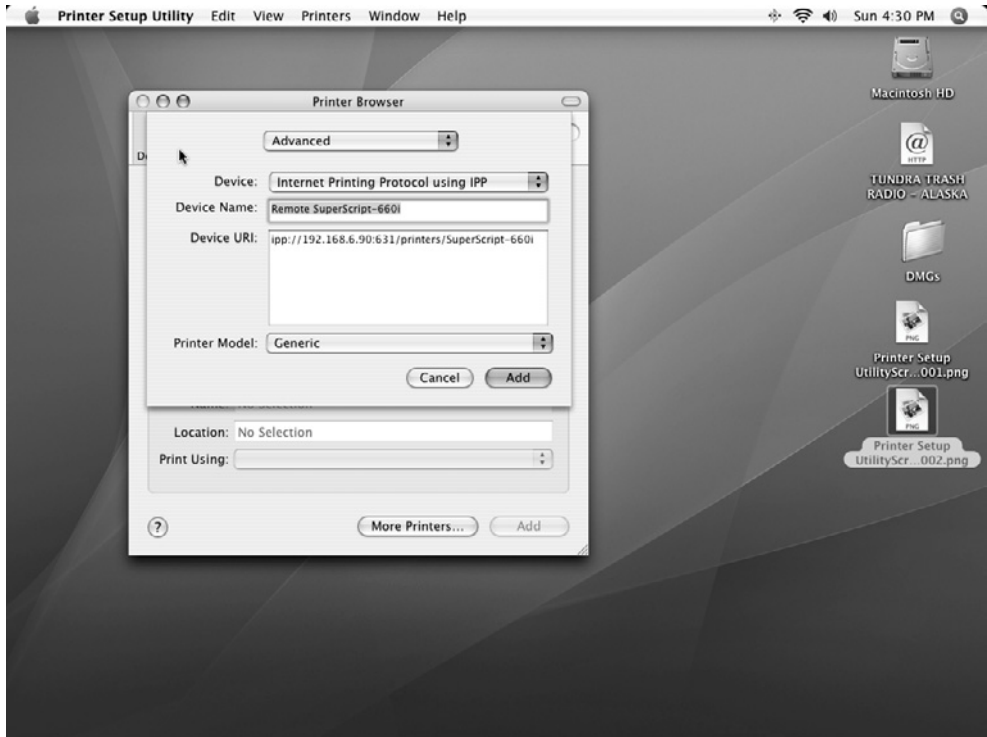
6. Jeśli wykorzystany jest sterownik inny niż standardowy PostScript, należy z listy rozwijanej *Printer Model* wybrać producenta i model drukarki. Na rysunku 30.7 przedstawiam wybrany sterownik HP LaserJet 4, kompatybilny z wieloma popularnymi drukarkami laserowymi.

7. Następnie trzeba kliknąć przycisk *Add*.

To wszystko. Drukarka udostępniana przez Ubuntu jest dostępna dla systemu Mac OS X. Ten system nie posiada łatwej w obsłudze funkcji wydruku próbnego; aby sprawdzić poprawność konfiguracji, wystarczy uruchomić jakikolwiek program i wydrukować dowolną stronę. Jeśli wszystko zostało skonfigurowane prawidłowo, drukarka wykona zadanie wydruku.

Usuwanie problemów z udostępnianymi drukarkami

W tym podrozdziale podsumowuję podstawowe zagadnienia związane z diagnostyką i potencjalnymi problemami, jakie mogą pojawić się w różnych wersjach podsystemu wydruków CUPS oraz w wyniku standardowych błędów popełnianych na etapie konfiguracji.



Rysunek 30.6. Określenie nazwy i URI zdalnej drukarki

Sprawdzenie dzienników podsystemu CUPS

Serwery wydruków CUPS zapisują informacje diagnostyczne w trzech plikach dzienników. Te pliki umieszczone są w katalogu `/var/log/cups`. I tak:

- ♦ ***access_log***: zawiera komunikaty dla każdej próby dostępu zdalnych hostów do serwera CUPS;
- ♦ ***error_log***: zawiera komunikaty zawierające błędy napotkane lub wygenerowane przez serwer wydruków CUPS;
- ♦ ***page_log***: zawiera komunikaty diagnostyczne dla każdej strony wydrukowanej za pośrednictwem serwera CUPS. Te komunikaty podają informacje o hoście, z którego pochodzi polecenie wydruku, wykorzystanej drukarce itp.

Najważniejsze dla diagnostyki są *access_log* i *error_log*. W przypadku napotkania problemu drukowaniem warto zapoznać się z ostatnimi wpisami w tych dziennikach w poszukiwaniu istotnych informacji. Plik *access_log* będzie zawierał adresy IP systemów komunikujących się z serwerem. To pomoże zweryfikować, czy zdalne systemy są rzeczywiście przyłączone do serwera wydruków. Korzystając z tych informacji należy w pliku *error_log* poszukać wskazówek dotyczących przyczyny problemu z wydrukiem.



Rysunek 30.7. Identyfikacja producenta i modelu drukarki

Odblokowanie dostępu do wybranych funkcji interfejsu administracyjnego WWW podsystemu CUPS

Domyślna konfiguracja podsystemu CUPS w Ubuntu zmienia się w zależności od wersji, co odzwierciedla udoskonalenia i odkryte problemy związane z bezpieczeństwem. Można być pewnym, że w którymś momencie użytkownika systemu Ubuntu, serwer wydruków CUPS zostanie zaktualizowany do nowszej wersji. Jeśli zdalne drukowanie działało bez problemów natomiast po aktualizacji drukowanie jest niemożliwe, należy wówczas dodać zdalne hosty do listy hostów uprawnionych do drukowania znajdującej się w pliku `/etc/cups/cupsd.conf`. Lista tych hostów znajduje się w sekcji `<Location />...</Location>`. Jeśli zdalne drukowanie nagle przestaje działać, należy upewnić się, że w tej sekcji zdefiniowany jest domyślny dostęp dla wszystkich hostów lub że znajdują się tam definicje wszystkich uprawnionych hostów (w przypadku jawnej definicji dostępnych hostów). Przykładowo, bardziej restrykcyjna wersja tej sekcji, blokująca dostęp dla innych hostów, prócz `localhost` oraz hostów z podsieci `192.168.6.*`, będzie miała następującą treść:

```
<Location />
  Order Deny,Allow
  Deny From All
  Allow From localhost
  Allow From 192.168.6.*
</Location>
```

Domyślna konfiguracja podsystemu wydruków CUPS w standardowej instalacji Ubuntu (w czasie pisania tej książki) zezwalała na dostęp dowolnych hostów (oczywiście, po modyfikacji konfiguracji podsystemu CUPS, aby nasłuchiwał na interfejsie Ethernet, a nie wyłącznie na interfejsie *loopback*, co zostało omówione wcześniej w tym rozdziale). Niestety, po aktualizacji podsystemu CUPS, może okazać się, że domyślna konfiguracja uległa zmianie. Przykładowo, różne elementy interfejsu administracyjnego CUPS posiadają własne sekcje <Location>. Jeśli okaże się, że z określonego hosta nie można uzyskać do nich dostępu, warto dopisać go za pomocą słowa kluczowego Allow (w tym przypadku trzeba unikać wpisów definiujących dostęp dla całych podsieci, jak zrobiłem to wcześniej).

Obsługa wstępnie sformatowanych wydruków

Innym zagadnieniem, dość podatnym na zmiany wersji podsystemu CUPS, jest sposób obsługi różnych formatów wydruków. Gdy drukujemy ze zdalnego systemu, takiego jak Microsoft Windows czy Apple Mac OS X, tenże system sam wstępnie formatuje wydruk przed wysłaniem go do serwera wydruków. Jeśli drukowanie z określonego hosta nagle przestanie działać, należy sprawdzić komunikaty w pliku *error_log*; jeśli pojawiły się tam informacje o konieczności uaktywnienia obsługi surowych danych wejściowych (*raw input*), trzeba zobaczyć, czy w pliku */etc/cups/mime.types* znajduje się następujący wpis (i nie jest poprzedzony znakiem #):

```
application/octet-stream
```

Znak # jest znakiem komentarza w pliku konfiguracyjnym CUPS. Ten wpis musi występować, aby serwer obsługiwał wstępnie sformatowane wydruki wysyłane do serwera CUPS za pośrednictwem protokołu HTTP.

Po wprowadzeniu zmian w konfiguracji podsystemu CUPS należy ponownie uruchomić serwer, wywołując następujące polecenie:

```
$ sudo /etc/init.d/cupsys restart
```

Gdzie można znaleźć dalsze informacje o podsystemie CUPS?

Konfigurację systemu wydruków CUPS można, w zależności od punktu widzenia, nazwać niezwykle rozbudowaną lub bezgranicznie irytującą. Plik konfiguracyjny */etc/cups/cupsd.conf* posiada tak wiele możliwych opcji, że nawet programy gcc, ls i tar, znane ze swoich rozbudowanych wierszy poleceń, mogłyby poczuć się zazdrosne. Jeśli ktoś próbuje wykorzystać CUPS do wykonywania zadań wykraczających poza obsługę prostego serwera wydruków omówionego w tym rozdziale, będzie potrzebował więcej informacji. Oto kilka doskonałych źródeł informacji o podsystemie wydruków CUPS.

- ♦ <http://www.cups.org/documentation.php>: dokumentacja podsystemu CUPS, udostępniana online przez Easy Software, Inc.
- ♦ <http://www.linuxprinting.org>: strona WWW specjalizująca się w udzielaniu pomocy w zakresie drukowania w Linuksie. Znajduje się tu bardzo użyteczna

baza danych drukarek, które można wykorzystywać w systemie Linux (<http://www.linuxprinting.org/prINTER.list.cgi>) oraz sporo informacji o podsystemie CUPS wraz z doskonałym podręcznikiem rozwiązywania problemów (celowo nie umieściłem tu adresu URL tego podręcznika, bo zapewne nie zmieściłby się w tym rozdziale).

- ♦ **CUPS: Common Unix Printing System** (Michael R. Sweet, SAMS, 2002, ISBN: 0672321963) — Michael Sweet jest jednym z właścicieli Easy Software, Inc., dzięki czemu ta książka jest nie tylko doskonałym podręcznikiem podsystemu drukowania CUPS, lecz również internetowego protokołu drukowania (IPP). Niestety, jest ona już przestarzała, ale nadal stanowi doskonałe źródło informacji; swoją kopię uważam za cenny element mojej bazy wiedzy.

Podobnie jak w przypadku innych zagadnień dotyczących Ubuntu, sporo informacji można znaleźć na forach Ubuntu, omówionych w rozdziale 1. tej książki.

Podsumowanie

Centralizacja usług drukowania to dobre rozwiązanie niemal w każdej sieci. W tym rozdziale zawarłem nieco podstawowych informacji na temat drukowania w systemach Unix i Linux, po czym omówiłem sposób instalacji i konfiguracji podsystemu wydruków CUPS, czyli *Common Unix Printing System*. Podałem również kilka informacji na temat konfiguracji systemów Microsoft Windows i Mac OS X w celu wykorzystania drukarek udostępnianych w systemie Ubuntu.

W rozdziale 31. omówię zagadnienia związane z udostępnianiem w sieci katalogów za pomocą systemu NFS (ang. *Network File System*), który jest standardowym sieciowym systemem plików dla systemów Unix i Linux. Udostępnianie plików, dokumentów i kodu z jednego serwera w środowisku roboczym jest ważnym zagadnieniem, choćby ze względu na wydajność czy wykonywanie kopii zapasowych. Obecnie zagadnienie to jest interesujące również dla wielu środowisk domowych, bowiem pozwala każdemu użytkownikowi sieci domowej uzyskać dostęp do kolekcji muzyki, fotografii z wakacji i innych plików, które mogą zainteresować współlokatorów, małżonka lub dzieci.

Rozdział 31.

Konfiguracja serwera NFS

W tym rozdziale:

- ◆ Podstawy technologii NFS
- ◆ Instalacja serwera NFS
- ◆ Eksportowanie katalogów z wykorzystaniem narzędzi graficznych
- ◆ Przegląd pliku konfiguracyjnego serwera NFS

Udostępnianie grup plików do wykorzystania przez współpracowników to obecnie standardowa procedura w środowiskach biznesowych, a dzięki sieciom domowym staje się powszechna w naszych domach. Zapewnienie scentralizowanego dostępu do kolekcji materiałów audio zgranych z posiadanych płyt CD lub zdjęć z ostatnich wakacji jest nie mniej ważne dla użytkownika domowego niż udostępnianie scentralizowanych zasobów podręczników i procedur czy repozytorium kodów źródłowych w firmie. Na szczęście, system Linux oferuje kilka sposobów udostępniania katalogów w sieci, niektóre z nich są dedykowane tylko systemom Linux i innym systemom uniksowym (w tym również Apple Mac OS X), inne są zorientowane na systemy Microsoft Windows (ale — oczywiście — nie wykluczają współpracy z systemami Unix i Mac OS X). W tym rozdziale omówię instalację systemu Ubuntu Linux w taki sposób, aby stał się centralnym repozytorium plików i katalogów z zastosowaniem mechanizmu NFS (ang. *Network File System*), rozwiązania popularnego w systemach Linux i Unix. Informacje na temat wymiany plików ze zdalnymi systemami Windows można znaleźć w rozdziale 32., „Konfiguracja serwera Samba”.

Mechanizm Network File System, znany powszechnie jako NFS, został opracowany przez firmę Sun Microsystems i jest obecnie najczęściej stosowanym sposobem udostępniania plików i katalogów w systemach uniksowych. Częściowo swoją popularność zawdzięcza temu, że jest domyślnie instalowany z większością dystrybucji Linuksa i systemów Unix, częściowo też temu, że klienci i serwery NFS są dostępne dla większości nowoczesnych systemów operacyjnych, również dla Microsoft Windows i Apple Mac OS X.

W tym rozdziale przyjrzymy się różnym zagadnieniom dotyczącym NFS, omówię tu różne wersje NFS i ich możliwości oraz aplikacje a nim związane. Oprócz poruszenia

zagadnień podstawowych, skupię się na zadaniu skonfigurowania systemu Ubuntu w charakterze serwera NFS. Podłączanie klientów do serwera NFS z innych systemów opisałem w rozdziale 16. Ten kończy się omówieniem mechanizmu NIS, *Network Information System*, rozproszonego mechanizmu uwierzytelniania, często wykorzystywanego w połączeniu z NFS.

Podstawy technologii NFS

NFS to sieciowy system plików zapewniający mechanizmy dostępu do plików dostępnych w zdalnych systemach. Sieciowe systemy plików często nazywa się również rozproszonymi systemami plików, ponieważ udostępniane przez nie pliki i katalogi mogą fizycznie znajdować się na różnych komputerach w sieci domowej, akademickiej lub korporacyjnej. NFS został opracowany przez firmę Sun we wczesnych latach 80. ubiegłego wieku, od tamtego czasu był wielokrotnie udoskonalany i obecnie jest dostępny dla wszystkich Linuksów, Uniksów i systemów uniksowych, a nawet istnieje oprogramowanie dla systemu Windows (dostawców niezależnych od firmy Microsoft). Specyfikacje systemu NFS były dostępne niedługo po opublikowaniu informacji o pierwszej wersji, dzięki czemu NFS stał się faktycznym standardem rozproszonych systemów plików.

Protokół NFS umożliwia serwerom plików eksportowanie centralnych zbiorów plików i katalogów do wielu systemów klienckich. Dobrym przykładem plików i katalogów, które warto udostępnić w formie zasobu centralnego z jednoczesnym dostępem wielu użytkowników, są katalogi domowe użytkowników, zestawy narzędzi programistycznych i scentralizowane zasoby danych, takie jak kolejki poczty i katalogi wykorzystywane do przechowywania internetowych forów dyskusyjnych.

Oto kilka powszechnie występujących scenariuszy wykorzystania zasobów NFS.

- ♦ **Udostępnianie wspólnych zbiorów danych.** Udostępnianie plików, które powinny być osiągalne dla wszystkich użytkowników sieci, takich jak pliki audio, dane biznesowe czy kod źródłowy przełomowej aplikacji, to najczęstsze zastosowanie sieciowych systemów plików.
- ♦ **Udostępnianie katalogów domowych.** Załóżmy, że wszystkie katalogi domowe użytkowników są zapisane w jednym katalogu `/export` serwera NFS, który jest automatycznie montowany przez wszystkie systemy sieciowe podczas uruchamiania. Plik haseł (`/etc/passwd`) dla każdego z tych systemów będzie wskazywał jako katalog domowy użytkownika `/export/nazwa-użytkownika`. Dzięki temu, użytkownicy będą mogli zalogować się do dowolnego systemu klienckiego obsługującego NFS i będą widzieć własny katalog domowy, udostępniany poprzez sieć (często bez ich świadomości).



Alternatywne rozwiązanie polega na automatycznym montowaniu sieciowych katalogów domowych z wykorzystaniem wyeksportowanego katalogu NFS zarządzanego przez demon *NFS automount*. Gdy następuje próba dostępu do katalogu zarządzanego przez demona *automount*, tenże demon automatycznie montuje zasób. Taki mechanizm upraszcza zawartość pliku `/etc/exports` serwera, pozwalając na wyeksportowanie jedynie katalogu nadrzędnego i pozostawienie demonowi *automount* zarządzania tym katalogiem (i jego podkatalogami) na każdym z klientów. Ogólne informacje o mechanizmie automatycznego montowania zasobów można znaleźć w ramce pod koniec tego rozdziału.

- ♦ **Współdzielenie zbioru binariów między systemami.** Załóżmy, że ktoś chce udostępnić określony zestaw narzędzi GNU dla różnych systemów w środowisku programistycznym i chce je scentralizować na serwerze NFS w celu ułatwienia zarządzania i aktualizacji. Aby upewnić się, że pliki konfiguracyjne będą kompatybilne we wszystkich systemach, można udostępnić te binaria w katalogu `/usr/gnu`, niezależnie od systemu. Można po prostu skompilować po jednej wersji binariów dla każdej wykorzystywanej architektury, skonfigurować je w taki sposób, aby wykorzystywały ścieżkę `/usr/gnu`, ale zapisać je w katalogach odpowiednich dla nazwy architektury lub systemu, np. `/export/gnu/ubuntu`, `/export/gnu/solaris8` itp. Każdego klienta można wówczas skonfigurować w taki sposób, aby montował odpowiedni dla siebie zasób sieciowy w katalogu `/usr/gnu`. Innymi słowy, zasób `/export/gnu/ubuntu` zostałby zamontowany jako `/usr/gnu` w systemach Ubuntu, a `/export/gnu/solaris8` byłby zamontowany jako `/usr/gnu` w systemach Solaris. Można wówczas umieścić katalog `/usr/gnu/bin` w ścieżce systemowej i w sposób niemal magiczny wszystko będzie działać prawidłowo.

Jak się przekonamy w tym rozdziale, NFS jest mechanizmem łatwym w instalacji oraz konfiguracji i oferuje elastyczne sieciowe rozwiązanie dostępne dla każdego systemu Ubuntu i innych Linuksów, jak również Uniksov i systemów uniksowych. W niektórych przypadkach należy wziąć pod uwagę kilka specyficznych zagadnień administracyjnych, ale Ubuntu zawiera użyteczne i łatwe w użyciu narzędzia, które znacznie upraszczają konfigurację serwera NFS.

Jak działa NFS?

Jeśli ktoś po prostu chce wykorzystywać NFS bez zastanawiania się, co się dzieje „pod maską”, może z powodzeniem pominąć ten punkt. Postaram się przybliżyć szczegóły dotyczące wielu wewnętrznych mechanizmów NFS-a, które warto poznać, ponieważ dobrze jest wiedzieć, w jaki sposób NFS radzi sobie z pewnymi przeszkodami natury technicznej, co pozwala mu na nawiązanie skutecznej komunikacji między systemami często bardzo różniącymi się wzajemnie, nawet o odmiennych architekturach procesora. Jeśli zatem ktoś jest ciekawy, zapraszam do lektury!

Protokół wykorzystywany przez NFS do komunikacji, znany pod nazwą *Remote Procedure Calls* (RPC), wykorzystuje niskopoziomowy protokół UDP (ang. *Universal Datagram Protocol*, stosowany w NFS v 2) lub standardowy protokół TCP (NFS v 3). Z tego właśnie powodu w pliku `/etc/services` większości systemów Linux występują wpisy dotyczące portu 2049 zarówno dla protokołu UDP, jak i TCP. UDP minimalizuje opóźnienia transmisji, ponieważ nie wykorzystuje szeregowania pakietów ani kontroli przepływu, nie gwarantuje także pewności dostawy — po prostu wysyła pakiety do określonego portu wskazanego hosta, gdzie inny proces oczekuje danych.

Projekt i implementacja protokołu RPC powodują, że NFS jest mechanizmem niezależnym od platformy, tzn. może służyć do wymiany danych między całkowicie odmiennymi systemami komputerowymi. RPC to metoda komunikacji typu klient-serwer wykorzystująca wywołania RPC (zdalne wywołania procedur) z różnymi parametrami. Procedury te są wywoływane na systemie klienta, ale ich wykonanie odbywa się w systemie serwera. Klient nie musi wiedzieć, czy wywołanie procedury odbywa się lokalnie, czy zdalnie na

serwerze — wynik procedury RPC otrzymuje dokładnie w ten sam sposób, jakby to było lokalne wywołanie procedury.

Sposób implementacji mechanizmu RPC jest niezwykle zmyślny. Technika ta działa w oparciu o mechanizm *marshalling*, polegający w gruncie rzeczy na pakowaniu argumentów procedury wywoływanej zdalnie w obustronnie uzgodniony format. Ten obustronnie uzgodniony format jest znany jako *eXternal Data Representation* (XDR) i stanowi w pewnym sensie język esperanto dla komputerów o odmiennych architekturach i uporządkowaniu bajtów (ang. *byte order*). Podsystem RPC klienta przesyła do odpowiedniego serwera niezależny od architektury pakiet zawierający wywołanie wraz z argumentami. Podsystem RPC serwera odbiera pakiet, rozpakowuje go (ang. *unmarshal*) i wydobywa argumenty, przekształcając je do formatu odpowiedniego dla platformy na której działa. Następnie podsystem RPC wywołuje procedurę lokalnie, pakuje wyniki w pakiet zwrotny i wysyła do klienta. Gdy ten pakiet jest odebrany przez klienta, jego podsystem RPC wypakowuje pakiet i wysyła wynik do programu, który wywołał RPC, zwracając wynik w taki sam sposób, jak procedura wywołana lokalnie. Pakowanie (ang. *marshalling*) i rozpakowywanie (ang. *unmarshalling*) oraz odpowiednia reprezentacja danych XDR powodują, że różne typy systemów mogą komunikować się bez problemów.

Komunikacja RPC jest wykorzystywana przez protokół NFS do realizacji różnych zadań, związanych m.in. z uwierzytelnianiem (NIS lub NIS+), zarządzaniem blokadami plików, zarządzaniem żądaniami montowania NFS, oraz żądaniami wysyłanymi do demona automatycznego montowania NFS. Aby aplikacje mogły wykorzystywać tak dużą liczbę usług bez konieczności wykorzystania zdefiniowanych z góry numerów portów, NFS pozwala komunikować się na dowolnym porcie, uzgadnianym w sposób dynamiczny pod warunkiem, że klient zarejestruje się w centralnej usłudze, zwanej demonem *portmapper*. Ten demon działa zawsze na porcie 111 hosta obsługującego komunikację RPC. Serwery rejestrują swoje usługi RPC w usłudze *portmappera*, informując o numerze portu, na którym nasłuchują. Klienci łączą się z *portmapperem* na znanym porcie, zadając pytanie o dostępność usługi oraz o port, na którym ona pracuje.

Błędy komunikacji mogą wystąpić w każdym mechanizmie komunikacji sieciowej i RPC nie jest tu wyjątkiem. Jak wspomniałem na początku tego podrozdziału, protokół UDP nie daje gwarancji uporządkowania pakietów ani powodzenia komunikacji. Z tego powodu w przypadku, gdy przez określony czas nie nadchodzi odpowiedź na żądanie RPC, klient ponawia próbę wysłania pakietu. To z kolei powoduje, że system zdalny może wywołać daną procedurę dwukrotnie w oparciu o te same dane wejściowe (parametry). Jak widać, operacje NFS są powtarzalne, tzn. mogą być wykonywane wielokrotnie z takim samym wynikiem, co oznacza, że operacja NFS nie może modyfikować żadnych danych, od których jest zależna. Choć NFS w wersji 3. do komunikacji wykorzystuje protokół TCP, własność powtarzalności operacji NFS jest nadal zachowana, co gwarantuje kompatybilność z implementacjami NFS v 2.

Serwery NFS są bezstanowe, co oznacza, że nie zachowują informacji między ponownymi uruchomieniami. Kiedy serwer ulega awarii w czasie, gdy klient wykonuje wywołanie RPC, klient będzie ponawiał żądanie do momentu, gdy serwer będzie znów dostępny lub wyczerpie się liczba prób zdefiniowana w konfiguracji klienta, co spowoduje zakończenie operacji błędem. Bezstanowość operacji powoduje, że NFS jest znacznie prostszym protokołem, ponieważ nie ma konieczności obsługi mechanizmów zachowania spójności mię-

dzy danymi klienta a danymi serwera. Klient ma zawsze rację, nawet po ponownym uruchomieniu serwera, ponieważ ani on, ani serwer nie zachowują jakichkolwiek informacji o wzajemnym stanie.

Mimo że bezstanowe operacje upraszczają wiele zagadnień, ta cecha ma też wady, bo powoduje przesyłanie nadmiernej ilości danych, jest mało efektywna i co za tym idzie powolna. Gdy dane przesłane przez klienta są zachowane na serwerze, serwer musi je zapisać w sposób synchroniczny, nie zwracając kontroli klientowi do momentu, gdy wszystkie dane znajdują się na dysku serwera. Nowsze wersje NFS-a obsługują pewne proste formy bufora zapisu, dzięki czemu serwer zwraca kontrolę klientowi tak szybko, jak to tylko możliwe. Ten bufor jest realizowany przez proces *rpciod* klienta (*RPC IO Daemon*), który buforuje oczekujące operacje zapisu na serwerze NFS w nadziei, że w przypadku połączenia kilku operacji zapisu w jedną wykorzystanie sieci będzie bardziej optymalne. W bieżącej, standardowej wersji NFS (NFS v 3) buforowanie operacji zapisu jest nadal dość niebezpieczne, ponieważ operacje są buforowane tylko w pamięci, przez co mogą być utracone w przypadku awarii komputera klienta.

W zupełnie bezstanowym środowisku awaria serwera uniemożliwiłaby zapisanie danych zmodyfikowanych w międzyczasie na kliencie. Serwer nie miałby bowiem możliwości rozpoznania, do którego pliku należą zmodyfikowane dane, ponieważ nie śledziłby poczynań swoich klientów. Aby rozwiązać ten problem, klienty NFS uzyskują od serwerów uchwyty plików przy ich otwarciu. Uchwyty plików to struktury danych identyfikujące serwer i plik, z którymi są związane. W przypadku awarii serwera klienty ponawiają próby operacji zapisu aż do momentu, gdy serwer jest ponownie dostępny lub zostaje przekroczony czas oczekiwania. Gdy serwer stanie się znów dostępny, otrzymuje od klienta zmodyfikowane dane oraz uchwyt pliku, za którego pomocą może określić, w którym pliku mają być zapisane zmodyfikowane dane.

Brak buforowania po stronie klienta ma również zalety, ponieważ pozwala zmniejszyć liczbę zależności klientów NFS od serwerów. Klienty nie buforują danych, muszą je zatem ponownie odczytać z serwerów po ponownym uruchomieniu. Ten fakt z pewnością spowalnia proces ponownego uruchamiania klientów, które uruchamiają programy zapisane na serwerach NFS. Gdy serwer nie jest dostępny, klient nie może się uruchomić. Z tego powodu klienty NFS najczęściej posiadają własne kopie programów wykorzystywanych podczas rozruchu systemu, a współdzielone po NFS-ie są jedynie dane oraz programy użytkownika.

Porównanie różnych wersji NFS

Mechanizm NFS jest dostępny na rynku prawie od początku istnienia systemów Unix na stacjach roboczych, ponieważ pojawił się wraz z pierwszymi stacjami roboczymi firmy Sun na początku lat 80. wieku XX. W tym punkcie omówię różnice między czterema różnymi wersjami NFS. Chodzi mi po części o względy historyczne, ale przede wszystkim zależy mi na tym, by pokazać, że to przemyślane i aktywnie rozwijane rozwiązanie. NFS 4 usuwa najważniejsze ograniczenia NFS 3, przede wszystkim uzupełniając system o buforowanie po stronie klienta odporne na ponowne uruchomienia systemu. Najnowszym wydaniem mechanizmu NFS wykorzystywanym powszechnie jest NFS 3, na którym skupię się w tym rozdziale.

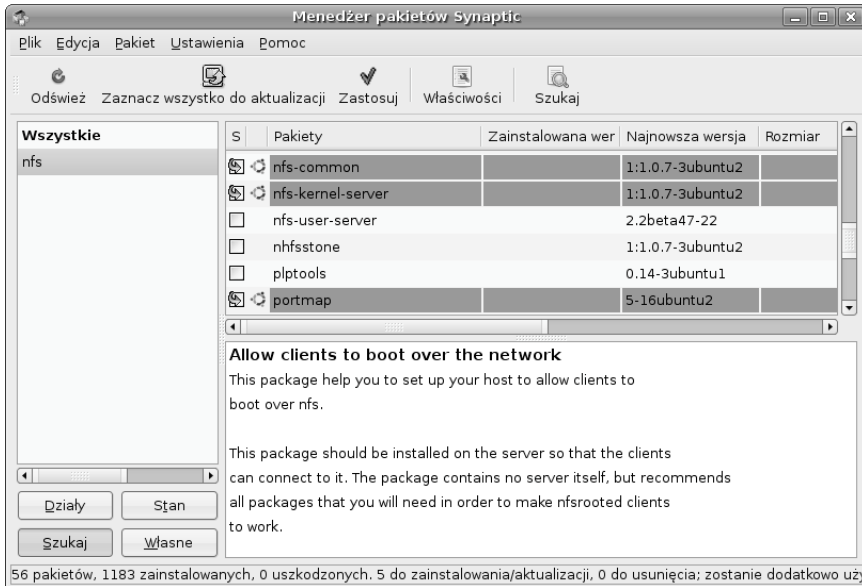
Na poniższej liście zestawilem najważniejsze cechy czterech wydań standardu NFS.

- ♦ **Wersja 1.:** oryginalny protokół NFS był wykorzystywany wyłącznie wewnętrznie przez firmę Sun na potrzeby rozwoju NFS-a i nie udało mi się znaleźć oryginalnej specyfikacji. Znaczenie tego wydania jest przede wszystkim historyczne.
- ♦ **Wersja 2.:** NFS 2 był pierwszą wersją protokołu NFS udostępnioną publicznie. Wersja 2. jako mechanizm transportowy wykorzystuje wyłącznie protokół UDP i definiuje 18 podstawowych procedur RPC. Wersja ta ma implementację 32-bitową, co narzuca ograniczenie rozmiaru plików do 2 GB i wykorzystuje 32-bajtowe uchwyty plików. NFS 2 ma dodatkowo ograniczenie transferu do pakietów o rozmiarach 8 KB.
- ♦ **Wersja 3.:** NFS 3 usuwa wiele niedoskonałości i niejednoznaczności specyfikacji wersji 2. i wykorzystuje wiele wynalazków technologicznych, które pojawiły się w ciągu dziesięciu lat od powstania wersji 2. Wersja 3. dodaje TCP jako sieciowy mechanizm transportowy, który jest wykorzystywany domyślnie, jeśli jest obsługiwany przez klienta i serwer. Do 64 KB został zwiększony maksymalny rozmiar pakietu danych przesyłanego między klientem a serwerem, implementacja protokołu stała się 64-bitowa, co praktycznie usunęło ograniczenia rozmiaru plików. Wszystkie te zmiany stały się możliwe dzięki postępowi w technologii sieciowej i zmianom w architekturze komputerów PC. Wersja 3. wprowadza również kilka dodatkowych procedur RPC oraz usuwa dwie, które nie były używane (a nawet nie były zaimplementowane w wersjach NFS, które miałem okazję sprawdzić). Aby udoskonalić wydajność, w NFS 3 wprowadzono mechanizm łączenia kilku operacji zapisu w jedną oraz automatycznego zwracania atrybutów pliku przy każdym wywołaniu RPC, zamiast wysyłania osobnego żądania w celu odczytu tych informacji, jak to miało miejsce w NFS 2.
- ♦ **Wersja 4.:** większa część nowości w NFS 4 została opracowana z myślą o pracy w internecie i środowiskach WWW, czemu miało służyć poprawienie trwałości połączeń, wydajności i bezpieczeństwa. Wersja 4. obsługuje trwałe buforowanie po stronie klienta, pozwalające na odzyskanie danych po ponownym uruchomieniu klienta przy minimalnej ilości ruchu sieciowego, posiada także obsługę mechanizmu ACL w systemach plików NFS. Wersja 4. wprowadza również udoskonalone, standardowe API poprawiające bezpieczeństwo dzięki mechanizmom bezpieczeństwa usług RPC znanym jako *Remote Procedure Call Security - Generic Security Services* (RPCSEC_GSS). To z kolei wymusza stosowanie *Generic Security Services Application Programming Interface* (GSS-API, zdefiniowane w RFC 2203), które służy do negocjacji najlepszego mechanizmu bezpieczeństwa obsługiwanego przez klienta i serwer.

Instalacja serwera NFS i pakietów pomocniczych

Aby zainstalować pakiety niezbędne do uruchamiania i monitorowania serwera NFS w systemie Ubuntu, należy z menu systemowego uruchomić *System/Administracja/Menedżer pakietów Synaptic*, kliknąć przycisk *Szukaj*, co spowoduje wyświetlenie okna wyszukiwania. Z listy rozwijanej trzeba wybrać opcję *Nazwy i opisy*, w polu wyszukiwania wpisać

ciąg znaków `nfs` i kliknąć przycisk *Szukaj*. Po zakończeniu wyszukiwania należy odszukać pakiety *nfs-common* i *nfs-kernel-server*, kliknąć je prawym przyciskiem myszy, z menu rozwijanego wybrać opcję *Zaznacz do instalacji*. Na rysunku 31.1 przedstawiam pakiety zaznaczone do instalacji.



Rysunek 31.1. Instalacja pakietów serwera NFS



Jak widać na rysunku 31.1, repozytoria Ubuntu zawierają dwa serwery NFS: jeden działający w jądrze Linuksa, drugi — w przestrzeni użytkownika. Serwer NFS działający w jądrze jest nieco szybszy, ale ten z przestrzeni użytkownika łatwiej kontrolować i diagnozować błędy. Pakiet NFS zawiera kilka narzędzi obsługiwanych z wiersza poleceń, takich jak program `exportfs`, pozwalający na ręczne zarządzanie eksportami katalogów serwera NFS oraz monitorowanie stanu udostępnianych katalogów. W tym rozdziale omawiam instalację i wykorzystanie serwera NFS działającego w jądrze Linuksa; jeśli ktoś napotka problemy z tym pakietem, może wypróbować drugą opcję, pozwalającą łatwiej diagnozować przyczyny błędów.

W zależności od tego, jakie oprogramowanie było poprzednio zainstalowane w systemie Ubuntu i od pakietów wybranych w interfejsie programu Synaptic, może pojawić się komunikat informujący o zależnościach, które muszą być zainstalowane dodatkowo. W takim przypadku warto kliknąć *Zaznacz*, aby zaakceptować instalację tych pakietów.

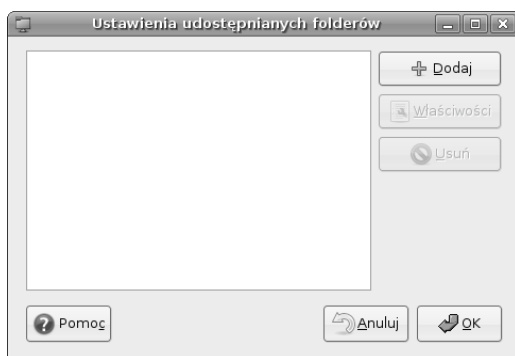
Po zaznaczeniu pakietów należy kliknąć przycisk *Zastosuj*, co rozpocznie instalację serwera NFS działającego w jądrze Linuksa. Po jej zakończeniu można przejść do konfiguracji udostępniania zasobów przez serwer NFS.

Więcej informacji na temat instalowania pakietów z użyciem menedżera pakietów Synaptic można znaleźć w rozdziale 20.

Wykorzystanie narzędzia Foldery współdzielone do udostępniania zasobów

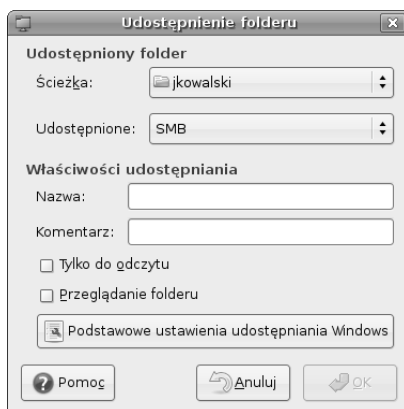
W tym miejscu książki nie powinno nikogo dziwić, że Ubuntu Linux posiada łatwe w użyciu narzędzie z graficznym interfejsem użytkownika, upraszczające proces definiowania i konfigurowania katalogów udostępnianych za pomocą NFS. Narzędzie to uruchamia się z menu systemowego, klikając *System/Administracja/Foldery współdzielone*. Po wprowadzeniu hasła użytkownika w celu uzyskania uprawnień administratora pojawi się główne okno programu, przedstawione na rysunku 31.2.

Rysunek 31.2.
Narzędzie Foldery współdzielone



Aby zdefiniować katalog, który chcemy udostępnić za pośrednictwem NFS, należy kliknąć przycisk *Dodaj*, co spowoduje wyświetlenie okna przedstawionego na rysunku 31.3.

Rysunek 31.3.
Definiowanie folderu udostępnionego

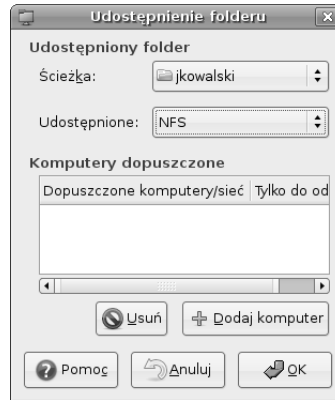


Jak widać na rysunku 31.3, domyślne ustawienia udostępniania dotyczą mechanizmu SMB, obsługiwanego przez serwer Samba, omówiony w rozdziale 32. Aby udostępnić katalog z użyciem mechanizmu NFS należy z listy rozwijanej *Udostępnione* wybrać pozycję *NFS*. To spowoduje wyświetlenie ustawień odpowiednich dla mechanizmu NFS, co zostało przedstawione na rysunku 31.4.

Jak widać na rysunku 31.4., domyślnym udostępnionym katalogiem jest katalog domowy użytkownika. W tym przykładzie zdecydowałem się udostępnić katalog zawierający moją

Rysunek 31.4.

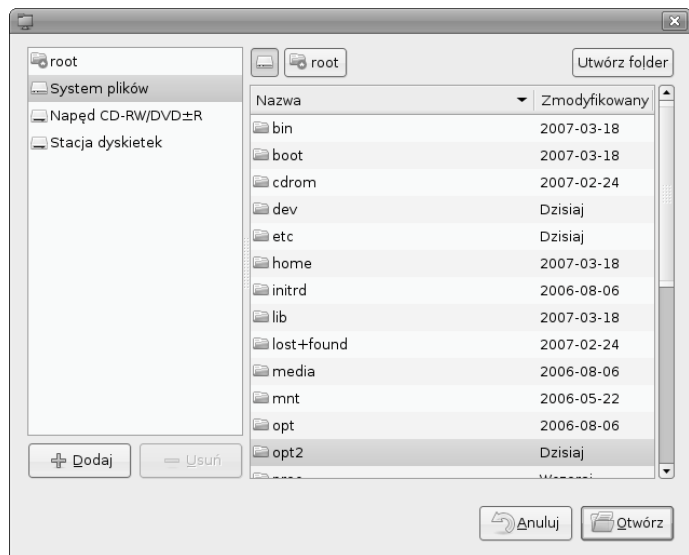
Ustawienia
udostępniania
dla katalogów NFS



kolekcję muzyki zgranej legalnie z posiadanych przeze mnie płyt CD lub pobranej z darmowych kolekcji muzyki w internecie. Aby wskazać katalog do udostępniania, należy z listy rozwijanej *Ścieżka* wybrać opcję *Inne*, co spowoduje wyświetlenie okna wyboru katalogu, przedstawionego na rysunku 31.5.

Rysunek 31.5.

Wybór katalogu
do eksportu

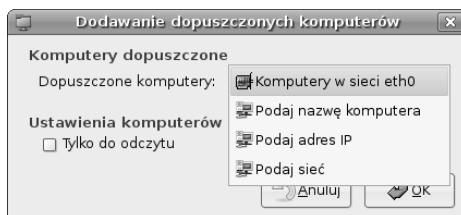


Aby wybrać dowolny katalog systemu plików, należy w lewym panelu tego okna kliknąć pozycję *System plików* i w prawym panelu odszukać odpowiednią ścieżkę do katalogu, który ma być udostępniony. W moim przykładzie to katalog */opt2*. Aby wybrać katalog, trzeba go zaznaczyć i kliknąć przycisk *Otwórz*, co zamknie okno wyboru katalogu, a kontrola powróci do okna przedstawionego na rysunku 31.4, zawierającego tym razem nazwę udostępnianego katalogu.

Następnie należy zidentyfikować hosty, które mają mieć możliwość montowania tworzonego zasobu sieciowego. W tym celu trzeba kliknąć przycisk *Dodaj*, co spowoduje wyświetlenie okna przedstawionego na rysunku 31.6.

Rysunek 31.6.

Definiowanie
użytkowników
uprawnionych
do dostępu
do udostępnionego
katalogu



Istnieje kilka sposobów identyfikacji hostów uprawnionych do montowania udostępnionych zasobów. Menu rozwijane *Dopuszczone komputery*, przedstawione na rysunku 31.6, zawiera cztery opcje. Oto one.

- ♦ **Komputery w sieci eth0:** umożliwia korzystanie z zasobu wszystkim klientom łączącym się z serwerem za pośrednictwem jego interfejsu eth0.
- ♦ **Podaj nazwę komputera:** umożliwia określenie nazwy hosta, którego użytkownicy będą mogli montować zasób. Wybór tej opcji spowoduje wyświetlenie dodatkowego pola tekstowego w oknie przedstawionym na rysunku 31.6, w którym należy wpisać nazwę hosta (kwalifikowaną lub skróconą) uprawnionego do korzystania z zasobu.
- ♦ **Podaj adres IP:** umożliwia określenie adresu IP hosta, którego użytkownicy będą mieli możliwość montowania zasobu. Wybór tej opcji spowoduje wyświetlenie dodatkowego pola tekstowego w oknie przedstawionym na rysunku 31.6, w którym należy wpisać adres IP uprawnionego hosta.
- ♦ **Podaj sieć:** pozwala określić adres IP podsieci, której hosty mogą montować udostępniany zasób. Wybór tej opcji spowoduje wyświetlenie dwóch dodatkowych pól tekstowych w oknie przedstawionym na rysunku 31.6, w których należy wprowadzić adres i maskę, definiującą podsieć uprawnioną do korzystania z katalogu.

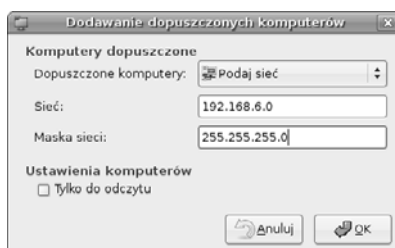


W przypadku określania dostępu do zasobów na podstawie nazwy hosta, adresu IP lub podsieci, zawsze można wprost zdefiniować dostęp dla pojedynczych hostów, klikając przycisk *Dodaj komputer* w oknie przedstawionym na rysunku 31.4.

W tym przykładzie zademonstruję sposób zdefiniowania dostępu do udostępnionego katalogu dla hostów z podsieci *192.168.0.0*, co przedstawiam na rysunku 31.7. Należy zwrócić uwagę, że to okno pozwala zdefiniować dostęp tylko do odczytu do katalogów współdzielonych (opcja *Tylko do odczytu*). Dzięki temu można z łatwością udostępnić dane bez możliwości ich modyfikowania.

Rysunek 31.7.

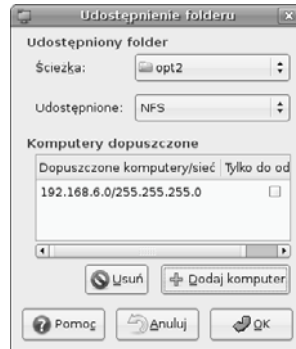
Definiowanie
dostępu do zasobu
dla określonej sieci



Kliknięcie przycisku *OK* w oknie przedstawionym na rysunku 31.7 spowoduje powrót do okna z rysunku 31.4, w którym tym razem pojawi się zasób */opt2* udostępniony w tym przykładzie (patrz rysunek 31.8).

Rysunek 31.8.

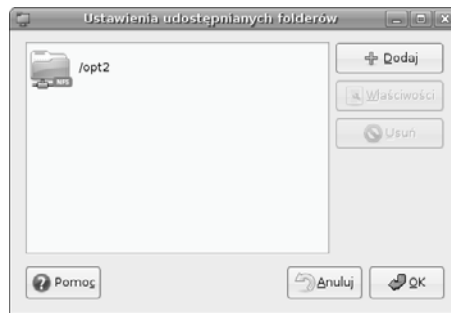
Ustawienia dostępu
sieciowego
dla zasobu NFS



Po kliknięciu przycisku *OK* zostanie ponownie wyświetlone okno przedstawione na rysunku 31.2, teraz zawierające ustawienia nowo zdefiniowanego zasobu *NFS* (patrz rysunek 31.9).

Rysunek 31.9.

Zasób NFS
w narzędziu
Foldery współdzielone



Prawie gotowe! Aby zmodyfikować lub zaktualizować ustawienia udostępnionego katalogu, należy zaznaczyć jego nazwę, po czym kliknąć przycisk *Właściwości*. Kliknięcie w tym oknie klawisza *OK* spowoduje rozpoczęcie udostępniania zasobu i zamknięcie okna narzędzia *Foldery współdzielone*.

Weryfikacja działania mechanizmu NFS

Pakiet serwera NFS zawiera narzędzie o nazwie *exportfs*, którego można użyć do zapoznania się z listą katalogów udostępnianych przez serwer NFS; za jego pomocą można też udostępniać inne katalogi, dopisane do konfiguracji serwera NFS, czyli w pliku */etc/exports*. Po wykonaniu działań opisanych w poprzednim podrozdziale zawartość pliku */etc/exports* będzie następująca:

```
# /etc/exports: the access control list for filesystems which may
#                be exported to NFS clients. See exports(5).
/opt2            192.168.0.0/255.255.0.0(rw)
```

Każdy wiersz w tym pliku, niebędący komentarzem (nierozpoczynający się od znaku #), definiuje katalog udostępniany przez serwer NFS i jest nazywany **specyfikacją eksportu**. Aby upewnić się, że katalog `/opt2` jest udostępniany (oraz, o ile to konieczne, w celu ponownego jego wyeksportowania), można posłużyć się poleceniem `exportfs -av`, które eksportuje wszystkie katalogi zdefiniowane w pliku konfiguracji eksportów serwera NFS:

```
$ sudo exportfs -a
exportfs: /etc/exports [3]: No 'sync' or 'async' option specified \
for export "192.168.0.0/255.255.0.0:/opt2".
Assuming default behavior ('sync').
NOTE: this default has changed from previous versions
exporting 192.168.0.0/255.255.0.0:/opt2
```

Wynik tego wywołania pokazuje, że katalog `/opt2` został wyeksportowany i jest dostępny dla hostów o adresach z podsieci 192.168.0.0.

Warto zwrócić uwagę na to, że program `exportfs` wypisał ostrzeżenie o brakującej opcji w specyfikacji eksportu katalogu `/opt2`. W przykładowym pliku `/etc/exports` cytowanym wyżej łatwo zauważyć, że wpis definiujący zasób `/opt2` kończy się ciągiem znaków `(rw)`. Na końcu specyfikacji eksportu znajdują się opcje związane z definiowanym zasobem. W tym przypadku jedyną wprowadzoną opcją jest `rw`, co oznacza, że eksportowany katalog będzie dostępny w trybie do odczytu i zapisu dla uwierzytelnionych użytkowników (więcej szczegółów na temat identyfikowania użytkowników przez protokół NFS można znaleźć w ramce „Użytkownicy i uwierzytelnianie w NFS”).

Komunikat ostrzeżenia wypisany przez polecenie `exportfs` określa, czy zmiany w plikach są dokonywane natychmiast po ich wykonaniu (*sync*, czyli synchronicznie), czy też są zapisywane w sposób odroczone (*async*, czyli asynchronicznie). Operacje asynchroniczne są powolniejsze, ponieważ system musi czekać na zakończenie zapisu, ale też bezpieczniejsze, bowiem użytkownik ma pewność, że operacja się zakończyła (chyba że wystąpi błąd połączenia, o czym użytkownik również się dowie). Starsze wersje protokołu NFS wykorzystywały wyłącznie operacje synchroniczne, ale jego nowsze wersje pozwalają określić sposób w jaki realizowany jest zapis. Aby wyeliminować wspomniany wyżej komunikat o błędzie, należy zmodyfikować plik `/etc/exports`, zastępując ciąg znaków `rw` ciągiem `rw,async` — polecam stosować tryb asynchroniczny, ponieważ jest znacznie wydajniejszy. Po wprowadzeniu tej zmiany zawartość pliku `/etc/exports` będzie następująca:

```
# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients. See exports(5).
/opt2          192.168.0.0/255.255.0.0(rw,async)
```

Zdefiniowany zasób należy wyeksportować, ponownie wywołując polecenie `exportfs`, które tym razem „nie będzie miało pretensji” do definicji zasobu:

```
$ sudo exportfs -av
exporting 192.168.0.0/255.255.0.0:/opt2
```

Pakiet `nfs-common` zawiera program o nazwie `showmount`, który można uruchomić na serwerze NFS w celu sprawdzenia listy wyeksportowanych zasobów. To polecenie nie wykonuje ponownego eksportu tych zasobów i nie modyfikuje ich w żaden sposób. Wywołanie polecenia `showmount` z opcją `-e` da następujący wynik:

Użytkownicy i uwierzytelnianie w NFS

NFS wykorzystuje ID użytkownika (UID) oraz ID grupy (GID) dostępne w systemowej bazie użytkowników (*/etc/passwd*) i dostęp do plików oraz katalogów na serwerze określa w oparciu o UID i GID zdefiniowane dla pliku na serwerze. Oznacza to, że do poprawności działania NFS niezbędne jest, aby UID i GID zgadzały się we wszystkich współpracujących systemach.

W niewielkich sieciach często wystarczy upewnić się, że we wszystkich systemach tworzeni są ci sami użytkownicy i grupy o odpowiednich identyfikatorach liczbowych, dzięki czemu własności plików udostępnionych z użyciem NFS będą prawidłowe.

W większych sieciach takie podejście jest niepraktyczne, warto wówczas zastosować scentralizowany mechanizm uwierzytelniania, taki jak *Network Information System* (NIS), opracowany przez firmę Sun Microsystems właśnie na potrzeby protokołu NFS. Niestety, omówienie instalacji i konfiguracji mechanizmu NIS znacznie wykracza poza tematykę tego rozdziału, ale w dokumencie NIS HOWTO (<http://www.linux-nis.org/nis-howto/>) można znaleźć mnóstwo informacji na ten temat. Dokument dostępny jest w języku polskim, oprócz angielskiego, francuskiego i portugalskiego.

```
$ sudo showmount -e
Export list for ulaptop:
/opt2 192.168.0.0/255.255.0.0
```

Wyczerpujące informacje na temat programów *exportfs* i *showmount* można znaleźć w ich podręcznikach systemowych: *man exportfs* i *man showmount* (wywołane w wierszu poleceń, np. z poziomu *xterm* lub terminala GNOME).

Ręczne definiowanie eksportów w pliku */etc/exports*

Każdy uwielbia narzędzia graficzne, ale czasem wygodniej po prostu wprowadzić niewielkie zmiany w pliku tekstowym, przetwarzanym przez te narzędzia. Tak jest szybciej i można tego dokonać przy pomocy dowolnego narzędzia, w którym da się uruchomić edytor tekstu.

Jak wspominałem w poprzednim podrozdziale, informację o katalogach wyeksportowanych przez serwer NFS zawiera plik */etc/exports*. Każdy wpis w tym pliku ma następującą składnię:

```
pełna-ścieżka-eksportowanego-katalogu hosty(opcje-montowania)
```

Taki wpis w pliku */etc/exports* nazywa się specyfikacją eksportu. Hosty można zdefiniować w postaci adresów IP, nazw lub podsieci. Przykładowo wpis *192.168.6.61* spowoduje, że do zasobu będzie miał dostęp tylko host o takim adresie IP, natomiast wpis *192.168.6.** lub *192.168.6.0* ograniczy dostęp do hostów z takiej właśnie podsieci. Domyślnie dostęp do wyeksportowanych katalogów mają wszystkie hosty (co jest sygnalizowane znakiem *** poprzedzającym opcje montowania).

Jak można się spodziewać, istnieje wiele opcji montowania. Poniżej przedstawiam najczęściej stosowane.

- ♦ `all_squash`: odwzorowuje żądania zapisu i odczytu na określonego użytkownika, najczęściej o nazwie *anonymous*. Ta opcja jest często wykorzystywana dla publicznych zasobów, takich jak katalogi serwerów USENET, publiczne serwery FTP itp. Wszystkie pliki zapisywane w katalogu NFS wyeksportowanym z zastosowaniem opcji `all_squash` będą miały przypisane UID i GID konta *anonymous* lub konta wskazanego opcjami `anonuid` i `anongid`. Domyślnie zastosowanie ma opcja `no_all_squash`, która zachowuje oryginalne wartości UID i GID plików.
- ♦ `insecure`: uaktywnia dostęp do katalogów NFS przez aplikacje klientów działające na niestandardowych portach. Domyślnie ta opcja jest nieaktywna, a żądania NFS muszą nadchodzić z portów o numerach mniejszych niż 1024. Opcja `insecure` może być niezbędna w przypadku niektórych komputerów PC i Macintosh. Podczas zastosowania tej opcji należy ograniczyć liczbę maszyn uprawnionych do dostępu do serwera NFS. Nie należy jej stosować w systemach dostępnych z internetu, ponieważ może to prowadzić do problemów z bezpieczeństwem.
- ♦ `no_root_squash`: pozwala użytkownikowi `root` na posiadanie na stacjach klienckich takich samych uprawnień na serwerze NFS. Opcja domyślnie nieaktywna.
- ♦ `ro`: katalogi są eksportowane w trybie tylko do odczytu. Domyślnie, zasoby są montowane w trybie `rw`, co oznacza dostęp do zapisu i odczytu.
- ♦ `sync`: wymusza synchroniczny tryb zapisu na serwerze, co powoduje, że klient musi czekać na zakończenie zapisu. Jest to domyślna wartość tej opcji, co zostało omówione w poprzednim podrozdziale. Można również zastosować opcję `async`, wymuszającą asynchroniczny tryb zapisu.

Wyczerpujące informacje na temat opcji pliku `/etc/exports` można znaleźć w jego podręczniku systemowym (`man 5 exports`).

Po zdefiniowaniu wpisu w pliku `/etc/exports` definiującego zasób NFS, należy wyeksportować ten katalog, wywołując polecenie `exportfs` z opcją `-r`, co wymusza ponowne odczytanie pliku `/etc/exports` i dokonanie niezbędnych modyfikacji w liście katalogów eksportowanych przez serwer NFS.

Gdzie znaleźć więcej informacji na temat mechanizmu NFS?

Jak się łatwo domyślić, wiele cennych informacji na temat mechanizmów NFS i NIS można znaleźć w sieci WWW. Szczególnie użyteczne są następujące zasoby:

- ♦ NFS FAQ (<http://nfs.sourceforge.net/>);
- ♦ NFS HOWTO (<http://nfs.sourceforge.net/nfs-howto/>);
- ♦ NIS HOWTO (<http://www.linux-nis.org/nis-howto/>);
- ♦ NFS Automount HOWTO (<http://www.tldp/HOWTO/Automount.html>).

Automatyczne montowanie katalogów domowych użytkowników z zasobów NFS

Gdy chcemy w sposób dynamiczny zarządzać montowaniem zasobów, przyda się mechanizm automounter. Automatyczne montowanie jest kontrolowane przez demona `automount`, który działa w systemie klienta. Oprócz automatycznego montowania zasobów NFS, demon `automount` może również automatycznie odmontowywać zasoby w przypadku, gdy nie są wykorzystywane przez dłuższy czas.

Wykorzystanie demona `automount` pozwala uniknąć konieczności montowania zasobów NFS w czasie, gdy nie są wykorzystywane. Stałe montowanie katalogów NFS na wszystkich klientach może doprowadzić do obniżenia wydajności serwera i nadmiernego ruchu sieciowego, którego większość jest zbędna, gdyż dotyczy montowania niewykorzystanych katalogów. Wykorzystanie demona `automount` pozwala zminimalizować ilość ruchu NFS.

Obecnie dla Linuksa dostępne są dwie wersje demona `automount`. Demon `amd` działa w przestrzeni użytkownika podobnie do oryginalnego automounter systemu SunOS. Demon `amd` jest konfigurowany w pliku `/etc/amd.conf`. Więcej informacji na temat `amd` można znaleźć na stronie domowej projektu Automount Utilities: <http://www.am-utils.org>. Drugim demonem typu `automount` jest `autofs`, zaimplementowany w jądrze Linuksa od wersji 2.2 wzwyż. Automounter jądra dla każdego automatycznie montowanego zasobu uruchamia proces `automount` działający w przestrzeni użytkownika. Demon `autofs` jest konfigurowany w pliku `/etc/auto.master` lub z użyciem map NIS o tej samej nazwie. Demon ten stanowi element jądra, zatem jest szybszy, a przez to stosowany domyślnie w wielu dystrybucjach (w tym w Ubuntu). Oba pakiety są dostępne w repozytoriach Ubuntu, ale szczegółowe omawianie mechanizmu automatycznego montowania zasobów wybiega poza tematykę tego rozdziału.

Podsumowanie

W tym rozdziale omówiłem mechanizm NFS (ang. *Network File System*), będący faktycznym standardem udostępniania katalogów w systemach uniksowych. NFS to prosty, niewymagający znacznych zasobów i szybki sieciowy system plików, a jego implementacje są dostępne dla wszystkich systemów Unix od wczesnych lat 80 ubiegłego wieku. Po kilku zdaniach wprowadzenia omówiłem proces instalacji i konfiguracji serwera NFS w systemie Ubuntu z zastosowaniem narzędzi graficznych oraz w trybie tekstowym.

Mimo że istnieją implementacje protokołu NFS dla systemów Microsoft Windows, większość z nich jest dostępna jedynie odpłatnie, przez co nie są alternatywą dla użytkowników środowisk heterogenicznych, czyli wykorzystujących różne systemy operacyjne, takie jak Ubuntu, Mac OS X czy Microsoft Windows (choć należy pamiętać, że Mac OS X, będący systemem uniksowym o korzeniach w systemach Mach i BSD, posiada natywną implementację protokołu NFS). Dość interesujące jest to, że najbardziej „standardowym” sposobem współdzielenia plików jest mechanizm Samba, czyli dostępny dla Ubuntu i innych systemów uniksowych mechanizm wykorzystujący własnościowy protokół *Server Message Block* (SMB), opracowany na potrzeby systemów operacyjnych Microsoft Windows (podobieństwo nazwy Samba i SMB nie jest przypadkowe). W rozdziale 32. opiszę instalację Samby, która umożliwi każdemu użytkownikowi Ubuntu udostępnienie plików i drukarek innym użytkownikom komputerów w sieci lokalnej, niezależnie od używanego przez nich systemu operacyjnego, na dodatek bez wydawania ani grosza.

Rozdział 32.

Konfiguracja serwera Samba

W tym rozdziale:

- ◆ Instalacja serwera Samba
- ◆ Udostępnianie katalogów systemom Windows
- ◆ Udostępnianie drukarek systemom Windows

Czy to się komu podoba, czy nie, nasza planeta jest przesycona komputerami z systemem Windows. Jak się łatwo domyślić z uwag wplecionych w teksty tej książki, jestem jedną z osób winnych wytworzeniu się postawy „my i oni”, opisującej sytuację Linux i Windows. Moje osobiste preferencje nie mają tu jednak większego znaczenia. Ważniejsze jest to, że w tym rozdziale omówię różne sposoby pozwalające zintegrować systemy Linux i Windows w obu kierunkach, a przy okazji uzyskać kilka dodatkowych funkcji, takich jak na przykład udostępnianie drukarek.

Jeśli ktoś z czytelników nie słyszał o serwerze Samba, zapewne przez kilka ostatnich lat mieszkał w jaskini. To w tej chwili jedna z najpopularniejszych aplikacji dla systemów Linux i Unix. Samba to, w dużym uproszczeniu, zestaw aplikacji napisanych z myślą o zaimplementowaniu w Linuksie protokołów sieciowych firmy Microsoft. Od powstania tej koncepcji Samba została przeniesiona właściwie na każdy istniejący system operacyjny, który korzysta z sieci.

Istnieje mnóstwo książek poświęconych Sambie, wyjaśniających każdy niuans jej plików konfiguracyjnych, instalacji i użytkowania. Cel, jaki postawiłem sobie, pisząc ten rozdział, nie polegał na wciśnięciu jeszcze jednej takiej książki wewnątrz książki poświęconej Ubuntu. Zdecydowałem się, że lepiej będzie przekazać czytelnikowi nieco informacji na temat sieci Windows i Samby, a następnie wyjaśnić, w jaki sposób udostępniać katalogi i drukarki systemu Ubuntu tak, aby stacje robocze z systemem Microsoft Windows były w stanie z nich korzystać w ramach sieci domowej, akademickiej lub korporacyjnej.

Przegląd technologii udostępniania zasobów w sieci Microsoft Windows

Sieci i związane z nimi technologie, takie jak routing, zgromadziły wokół siebie większą liczbę akronimów niż którakolwiek z pozostałych gałęzi informatyki. Sieci systemów MS-DOS i Microsoft Windows miały w tym swój znaczący wkład, przede wszystkim z powodu wszechobecności tych systemów operacyjnych we współczesnych środowiskach komputerowych. Popularność systemu MS-DOS (dawniej) i Windows (obecnie) spowodowała, że współczesne systemy Windows obsługują protokoły wszystkich swoich poprzedników. Na marginesie, jedną z cech, które podziwiam w systemach Windows, jest ich znaczny postęp technologiczny przy zachowaniu zdumiewającej zgodności wstecz z większością aplikacji (aż do systemu DOS) i protokołów sieciowych. Systemy Windows nadal są kompatybilne z protokołami *Internet Packet Exchange* (IPX) i *Sequenced Packet Exchange* (SPX), wprowadzonymi przez firmę Novell wraz z pierwszymi serwerami sieciowymi dla komputerów PC. Jednak w tym rozdziale większe znaczenie będą miały akronimy dotyczące protokołów sieciowych opracowanych przez Microsoft i wykorzystywanych do udostępniania plików i innych zasobów w sieciach komputerów PC bez konieczności stosowania narzędzi firm trzecich.

Podsystem *Basic Input and Output System* wykorzystywany przez komputery PC jest lepiej znany jako *BIOS*. Wraz z popularyzacją sieci lokalnych firma Microsoft rozszerzyła możliwości BIOS-u o funkcje sieciowe, i nazwała swój protokół *network BIOS*, w skrócie znany jako *NetBIOS*. Podobnie do BIOS-u, oferującego podstawowe funkcje obsługi wejścia-wyjścia w systemie, NetBIOS zapewnia podstawowe funkcje pozwalające wykorzystywać usługi sieciowe i zarządzać nimi. Polecenia i funkcje NetBIOS są przesyłane między systemami przyłączonymi do sieci i wymagają do tego niskopoziomowych mechanizmów transportowych przesyłających pakiety z jednego hosta do drugiego. Niskopoziomowe protokoły transportowe nadal stosowane w sieciach PC to IPX, NetBEUI (ang. *Network Basic Extended User Interface*) i TCP/IP (ang. *Transmission Control Protocol/Internet Protocol*). Co ciekawe, termin „internet” w rozwinięciach nazw IPX i TCP/IP odnosi się do „międzysieciowości”, a nie do internetu jaki znamy dzisiaj.

Nowoczesne systemy Windows wysyłają swoje żądania NetBIOS za pośrednictwem protokołu transportowego TCP/IP. Ponad protokołem NetBIOS, Windows udostępnia interfejs wyższego poziomu o nazwie *Server Message Block* (SMB), który w prosty sposób dostarcza funkcje sieciowe dla aplikacji. SMB to protokół połączeniowy (a nie rozgłoszeniowy), co oznacza, że jego działanie opiera się na ustanowieniu połączenia do określonych usług sieciowych oferowanych przez hosty przyłączone do sieci, nie jest zaś wykorzystywane rozgłaszanie informacji o dostępności tych usług. Po nawiązaniu połączenia SMB dostarcza czterech podstawowych typów funkcji. Oto one.

- ♦ Funkcje obsługi sesji, służące do negocjowania i nawiązywania połączeń sieciowych między systemami (określane jako wirtualne obwody, ang. *virtual circuits*) oraz do uwierzytelniania i weryfikowania uprawnień dostępu każdej ze stron do oferowanych usług.
- ♦ Funkcje obsługi plików, pozwalające na otwieranie, odczyt i zapis zdalnych plików, współdzielonych katalogów itp.

- ♦ Funkcje drukowania, pozwalające aplikacjom na przysyłanie żądań wydruków do zdalnych urządzeń.
- ♦ Funkcje wymiany komunikatów, pozwalające aplikacjom na przysyłanie komunikatów o charakterze kontrolnym, statusowym lub informacyjnym.

SMB we wczesnych latach 90. ubiegłego wieku stał się standardem Open Group. Serwer Samba swoją nazwę wziął od skrótu SMB, dzięki czemu nazwa jednoznacznie się kojarzy nie będąc jednocześnie kolejnym, trudnym do wymówienia, akronimem.

W nowszych wersjach systemów Windows wprowadzony został udoskonalony protokół SMB o nazwie CIFS (ang. *Common Internet File System*), przekazany przez Microsoft do standaryzacji instytucji o wdzięcznej nazwie IETF (ang. *Internet Engineering Task Force*). IETF jest otwartym stowarzyszeniem ludzi zainteresowanych architekturą Internetu oraz jego sprawnym działaniem. CIFS został przyjęty jako standard, a rozszerza możliwości protokołu SMB między innymi poprzez rozszerzenie możliwości wymiany zasobów, wprowadzając dodatkowe, międzyplatformowe technologie, takie jak HTTP URL (ang. *Hyper-Text Transfer Protocol Uniform Resource Locators*) czy DNS (ang. *Domain Name System* wykorzystywany do odwzorowania nazw hostów na ich adresy IP i vice versa).

Wprowadzenie do Samby

Gdyby dobrze policzyć, z pewnością okazałoby się, że w systemach Windows przechowywanych jest więcej danych niż w jakimkolwiek innym systemie komputerowym. Te wszystkie domowe i biurowe komputery, choćby każdy posiadał dysk o pojemności zaledwie 10 GB, tworzą w sumie gigantyczny system plików przechowujący niesamowite ilości danych. Samba daje użytkownikom Linuksa możliwość dostępu do tego olbrzymiego systemu plików, ale częściej wykorzystywana jest do tego, by stacje robocze Windows miały dostęp do zasobów systemów Linuksa i innych systemów uniksowych. Samba daje taką możliwość dzięki udostępnieniu protokołu sieciowego kompatybilnego z protokołami systemu Windows służącymi do udostępniania plików i drukarek. Dla systemu Windows, Linux z Sambą wygląda dokładnie tak samo jak dowolny inny system Windows udostępniający pliki w sieci. Użytkownicy systemu Windows mogą korzystać z szybkości, możliwości i pojemności systemów Linux, nie zdając sobie nawet sprawy z faktu, że mają do czynienia z Linuksem.

Samba to otwarty interfejs dla Linuksa, Uniksa oraz innych systemów operacyjnych, pozwalający przyłączyć do nich dowolne urządzenie sieciowe obsługujące protokół SMB — przede wszystkim systemy Windows. Samba pozwala użytkownikom systemu Windows na wykorzystanie systemu plików Linuksa oraz jego zasobów, na przykład drukarek, tak samo, jakby były to zasoby innego systemu Windows. Jeśli np. Samba działa w Linuksie w sieci lokalnej, to użytkownicy systemu Windows mogą montować swoje katalogi domowe z Linuksa i drukować na drukarce przyłączonej fizycznie do komputera z tymże Linuksem. Samba, opracowana przez Andrew Tridgella, jest jednym z najbardziej pożytecznych kawałków kodu napisanych w celu zwiększenia możliwości wymiany informacji i współpracy niezależnej od systemu operacyjnego.

Samba zawiera zarówno oprogramowanie klienckie, jak i serwerowe; innymi słowy, są to programy umożliwiające użytkownikom Linuksa wykorzystanie zasobów SMB udostępnianych przez inne hosty w sieci, jak i oprogramowanie serwerowe, pozwalające świadczyć usługi sieciowe za pośrednictwem protokołu SMB. Wykorzystanie klienckiej części Samby zostało omówione w rozdziale 16. W tym rozdziale skupię się na części serwerowej Samby.

Serwer Samba składa się z dwóch procesów, każdy z nich można uruchomić z wiersza poleceń albo w sposób automatyczny w ramach procedury rozruchowej systemu. Tymi procesami są: *smbd*, demon Samby obsługujący udostępnianie plików i usług drukowania dla klientów systemu Windows, oraz *nmbd*, serwer nazw NetBIOS odwzorowujący nazwy systemu Windows, wykorzystywane w żądaniach SMB, na adresy IP używane przez systemy Linux. Demona Samby konfiguruje się modyfikując plik */etc/samba/smb.conf*. Wybrane katalogi można udostępniać za pomocą narzędzi z graficznym interfejsem użytkownika, co zostało omówione w podrozdziale „Wykorzystanie folderów udostępnionych”, lub przez modyfikację pliku konfiguracyjnego, o czym mowa w podrozdziale „Podstawy konfiguracji serwera Samba”.

Wymiana danych między systemami Linux i Windows to nie tylko Samba. Jądro Linuksa posiada wbudowaną obsługę protokołów niezbędnych do uzyskania dostępu do systemów plików Windows, dzięki czemu istnieje możliwość montowania systemów plików Windows poprzez odpowiednie wpisy w pliku */etc/fstab*, podobnie jak każdego innego systemu plików.

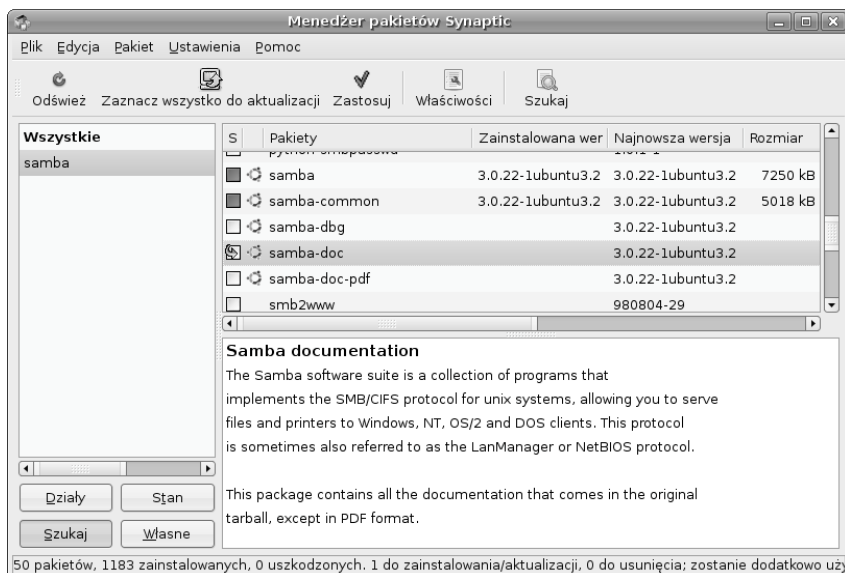
Instalacja serwera Samba i pakietów pomocniczych

Aby zainstalować pakiety niezbędne do uruchomienia i monitorowania serwera Samba w systemie Ubuntu, należy uruchomić menedżer pakietów Synaptic (menu *System/Administracja/Menedżer plików Synaptic*) i kliknąć ikonę *Wyszukaj*. W oknie wyszukiwania trzeba wybrać pozycję *Nazwy i opisy*, w polu wyszukiwania wpisać ciąg znaków *samba*, po czym kliknąć przycisk *Szukaj*. Po zakończeniu wyszukiwania należy odszukać pakiety *samba-common* i *samba-server*, kliknąć prawym przyciskiem myszy i z menu podręcznego wybrać funkcję *Zaznacz do instalacji*. Warto również zaznaczyć pakiety *samba-doc* i *samba-doc-pdf*, które zawierają dokumentację systemu Samba odpowiednio w formacie HTML i PDF oraz wersję elektroniczną książki *Samba 3 By Example* (Samba 3 w przykładach) autorstwa jednego z założycieli projektu Samba, Johna Terpstry. Pakiety zaznaczone do instalacji przedstawiam na rysunku 32.1.

W zależności od tego, jakie oprogramowanie zostało zainstalowane poprzednio, po zaznaczeniu pakietów może pojawić się okno z informacją o tym, że muszą zostać zainstalowane dodatkowe pakiety, i z prośbą o potwierdzenie. Jeśli pojawi się takie okno, należy kliknąć przycisk *Zaznacz*, co spowoduje zatwierdzenie instalacji wybranych pakietów oraz wszystkich pakietów niezbędnych do ich działania.

Następnie trzeba kliknąć przycisk *Zastosuj* na pasku narzędzi programu Synaptic. Po zakończeniu instalacji system jest gotowy do wymiany danych z innymi systemami obsługującymi protokół SMB.

Rysunek 32.1.
Instalacja Samby
i pakietów
pomocniczych



Szczegółowe informacje na temat instalowania pakietów za pomocą menedżera pakietów Synaptic można znaleźć w rozdziale 20.

Podstawy konfiguracji serwera Samba

Obecnie brakuje przyjaznego mechanizmu do konfiguracji serwera Samba w Ubuntu i jest to poważna skaza na przyjaznym obliczu tego systemu. Nie jestem jedyną osobą, która to zauważyła, i trwają aktywne dyskusje na ten temat na wielu forach systemu Ubuntu. Jak na razie narzędzie takie nie istnieje i podstawową konfigurację Samby w Ubuntu trzeba wykonać klasyczną, linuksową metodą, czyli z użyciem edytora tekstu. Serwer Samba konfiguruje się modyfikując plik `/etc/samba/smb.conf`. W tym pliku można znaleźć wiele użytecznych komentarzy, są nimi wiersze rozpoczynające się od znaku `#`. Znajduje się tam również sporo przykładowych, nieaktywnych opcji konfiguracyjnych, rozpoczynających się od znaku średnika. Te opcje można łatwo uaktywnić, usuwając średnik na początku wiersza.

Jak przekonamy się w kolejnych podrozdziałach, konfiguracja Samby, czyli edycja pliku `/etc/samba/smb.conf`, to w rzeczywistości całkiem łatwe zadanie, choć wydaje się niezbyt eleganckie w porównaniu z pozostałymi narzędziami konfiguracyjnymi oferowanymi przez Ubuntu.

Informacje prezentowane w tym podrozdziale będą miały zastosowanie niezależnie od tego, czy w posiadanej przez czytelnika wersji Linuksa istnieje odpowiednie narzędzie do zarządzania serwerem Samba (warto w tym celu sprawdzić zawartość menu *System/Administracja*). Dodatkowo warto posiadać nieco głębszą wiedzę na temat wewnętrznych mechanizmów działania Samby oraz sposobu organizacji jej pliku konfiguracyjnego.



Istnieją narzędzia do konfiguracji Samby za pośrednictwem przeglądarki WWW, takie jak *swat* (ang. *Samba Web Administration Tool*) czy *webmin* (narzędzie konfiguracyjne obsługiwane za pomocą przeglądarki WWW, zawierające moduły do zarządzania różnymi mechanizmami systemowymi) i można je znaleźć w repozytoriach Ubuntu. Narzędzia te wymagają specjalnej konfiguracji przed rozpoczęciem pracy. Zamiast poświęcać sporą ilość czasu na wyjaśnianie zasady ich działania, zdecydowałem się omówić klasyczną konfigurację Samby w oczekiwaniu na narzędzia w pełni zintegrowane ze środowiskiem GNOME/Ubuntu. W końcu odpowiednie narzędzie dla środowiska KDE już istnieje, zatem GNOME nie może (a raczej nie powinno) pozostawać w tyle.

W kilku kolejnych punktach skupię się na ważnych sekcjach pliku konfiguracyjnego Samby, które należy zmodyfikować, dostosowując do własnego środowiska. Nie będę omawiać każdej sekcji pliku konfiguracyjnego, ponieważ nie warto streszczać obszernych informacji zawartych w materiałach zamieszczonych w pakietach *samba-doc* i *samba-doc-pdf* a także książce na temat Samby. W pozostałej części rozdziału skupię się na opisie technik rozwiązywania konkretnych problemów tak szybko, jak to tylko możliwe.

Identyfikacja grupy roboczej lub domeny

Plik konfiguracyjny Samby jest podzielony na wiele sekcji identyfikowanych nazwami ujętymi w nawiasy kwadratowe. Wpisy dotyczące konfiguracji sieci, w której ma działać serwer Samby, znajdują się w sekcji `[global]` pliku `/etc/samba/smb.conf`.

Najważniejszy wpis, jaki należy ustawić, aby system Ubuntu mógł bez problemu wymieniać się danymi z systemami Windows, to nazwa grupy roboczej, identyfikująca grupę roboczą lub domenę systemu Windows, do której ma należeć serwer Samba. W moim przykładowym systemie mam następujący wpis:

```
[global]
workgroup = WVH
```

Ten wpis identyfikuje moją maszynę jako należącą do grupy roboczej Windows o nazwie WVH. W tej przykładowej, niewielkiej sieci domowej, mamy do czynienia z grupą roboczą, ale nie ma to większego znaczenia — najważniejsze, że serwer Samby może być członkiem już istniejącej grupy roboczej lub domeny Windows, może też być głównym kontrolerem domeny definiującym domenę, do której będą należały inne komputery.



Jeśli serwer Samba ma działać jako główny kontroler domeny (ang. *primary domain controller*, PDC) dla domeny Windows, należy zmodyfikować opcję `domain master` na wartość `yes` oraz upewnić się, że ta opcja nie jest oznaczona jako komentarz. W moim przypadku opcja ta ma następującą wartość:

```
domain master = auto
```

To oznacza, że serwer Samba będzie głównym kontrolerem domeny, w przypadku gdy w sieci nie występuje inny główny kontroler domeny.

W tym rozdziale omawiam sposób wykorzystywania serwera Samba w ramach grupy roboczej, co jest typowym zastosowaniem serwera Samba w środowisku domowym lub małej firmie (SOHO). Informacje, w jaki sposób skonfigurować kompletną domenę Windows, można znaleźć w dokumentacji (zainstalowanej w ramach pakietu *samba-doc* w podrozdziale „Instalacja serwera Samba i pakietów pomocniczych”), polecam też kopię książki *Samba 3 By Example*, którą można znaleźć w formacie HTML w katalogu `/usr/share/doc/samba-doc/htmldocs/Samba3-ByExample`.

W moim przykładzie przedstawiam niewielką grupę roboczą, warto zatem aktywować wpisy instruujące Sambę, aby do wyszukiwania hostów w sieci nie wykorzystywała protokołu WINS (ang. *Windows NetBIOS name service*, omówionego wcześniej w tym rozdziale), tylko wbudowany systemowy mechanizm *Domain Name Server*. W tym celu należy zmodyfikować następujące opcje:

```
wins support = no
dns proxy = yes
name resolve order = lmhosts host wins bcast
```

Ostatnia opcja informuje serwer Samba, że danych o hostach powinien szukać najpierw w lokalnej konfiguracji *lmhosts*, następnie sprawdzić w pliku */etc/hosts* w systemie serwera Samby, później sprawdzić bazę WINS (która po poprzedniej konfiguracji jest przekierowana do serwera DNS), a na końcu przeszukać sieć, wysyłając żądanie na adres rozgłoszeniowy. Ta kombinacja jest w zupełności wystarczająca dla wszystkich konfiguracji serwera Samba, z jakimi miałem okazję pracować.

I to już wszystkie modyfikacje pliku konfiguracyjnego serwera Samba, jakich należy dokonać, aby otrzymać działający serwer udostępniający pliki wskazanej grupie roboczej lub domenę. Nie było tak źle, prawda?

Konfiguracja uwierzytelniania Samby

W konfiguracji grupy roboczej, wykorzystywanej w naszym przykładzie, Samba jest już skonfigurowana w sposób umożliwiający jej uwierzytelnianie członków grupy roboczej. Aby jednak uzyskać dostęp do zasobów udostępnianych przez Sambę, musisz mieć możliwość uwierzytelnienia. Samba posiada własną bazę uwierzytelniającą; aby dodać nowego użytkownika do tej bazy, należy posłużyć się poleceniem *smbpasswd*. Gdy do systemu dodawane są informacje na temat użytkownika, trzeba zastosować opcję *-a*, po której wpisuje się nazwę nowego użytkownika. Polecenie *smbpasswd* poprosi o podanie hasła oraz jego powtórzenie (aby upewnić się, że nie wystąpił błąd). Oto przykład wywołania programu *smbpasswd*:

```
$ sudo smbpasswd -a jkowiński
Password:
New SMB password:
Retype new SMB password:
```

Pierwsze żądanie hasła pochodzi z programu *sudo*, w tym przypadku chodzi o podanie hasła użytkownika wywołującego polecenie w celu nadania mu uprawnień potrzebnych do zmodyfikowania haseł użytkowników Samby. Kolejne dwa żądania podania hasła dotyczą już nowo tworzonego użytkownika Samby (*wvh*).



Domyślne ustawienie zabezpieczeń serwera Samba definiuje opcja *security = user*, co wymusza konieczność istnienia kont użytkowników Samby również w systemowej bazie użytkowników */etc/passwd*. Inaczej przy wywołaniu polecenia *smbpasswd* serwer Samba wypisze komunikat o błędzie. Informacje na temat modeli bezpieczeństwa i ich działania można znaleźć w pliku */usr/share/doc/samba-doc/html/docs/ServerType.html*, dostępnym w pakiecie *samba-doc*.

Udostępnianie drukarek i katalogów domowych z użyciem Samby

Po skonfigurowaniu grupy roboczej i utworzeniu użytkownika należy wybrać zasób systemu Ubuntu, który ma być udostępniany systemom Windows przyłączającym się do naszego serwera Samba. Najczęściej w tym kontekście występują drukarki i katalogi domowe użytkowników.

Za udostępnianie drukarek odpowiadają następujące opcje w sekcji `[global]` pliku konfiguracyjnego Samby (należy upewnić się, że nie są zakomentowane, tzn. nie są poprzedzone np. znakiem średnika):

```
load printers = yes
printing = cups
printcap name = cups
```

W dalszej części pliku konfiguracyjnego w sekcjach `[printers]` i `[print$]` znajdują się szczegółowe opcje definiujące sposób interakcji systemu z serwerem Samba. Nie ma konieczności modyfikowania tych ustawień, ale warto się z nimi zapoznać:

```
[printers]
comment = Wszystkie drukarki
browseable = no
path = /tmp
printable = yes
public = no
writable = no
create mode = 0700
[print$]
comment = Sterowniki drukarek
path = /var/lib/samba/printers
browseable = yes
read only = yes
guest ok = no
```

Sekcja `[printers]` określa sposób, w jaki serwer Samba ma obsługiwać żądania identyfikacji drukarek i nadchodzących żądań wydruku od klientów Windows, natomiast sekcja `[print$]` definiuje tradycyjny zasób systemów Windows o nazwie *print\$*, który służy do udostępniania sterowników drukarek klientom systemu Windows, dzięki czemu systemy klienckie mogą automatycznie zainstalować drukarkę na etapie dodawania jej do systemu.

Najczęściej udostępnianym zasobem, oprócz drukarek, są katalogi domowe użytkowników w Ubuntu. Dzięki Sambie, mogą oni w prosty sposób uzyskać do nich dostęp wprost z systemu Windows. Daje to także możliwość tworzenia kopii plików systemu Windows na serwerze, na którym pracuje Samba – pozwala to na przykład na wykorzystanie mechanizmów tworzenia kopii zapasowych dostępnych w Linuksie do stworzenia centralnej kopii danych ze wszystkich komputerów firmowych (więcej informacji na temat mechanizmów wykonywania kopii zapasowych w Ubuntu można znaleźć w rozdziale 22).

Na szczęście, plik konfiguracyjny Samby jest domyślnie skonfigurowany do obsługi katalogów domowych pod warunkiem, że nazwy użytkowników systemu Windows i Linux są identyczne. Za takie działanie serwera Samba odpowiada następująca sekcja jego pliku konfiguracyjnego:


```
[homes]
comment = Katalogi domowe
browseable = no
writable = no
create mask = 0700
directory mask = 0700
```

Jak widać na tym przykładzie pliku konfiguracyjnego, katalogi domowe są udostępniane domyślnie, ale tylko w trybie do odczytu. To mało wygodna opcja, zatem warto zmodyfikować to ustawienie w następujący sposób:

```
writable = yes
```

Teraz użytkownicy mogą odczytywać i zapisywać pliki w swoich katalogach domowych.



Jeśli użytkownicy systemu Windows i Linux mają różne loginy, można je ze sobą powiązać, tworząc dodatkowe wpisy w pliku `/etc/samba/smbusers`. Wpisy w tym pliku muszą być dopasowane do wzorca `UnixLogin = WindowsLogin`. Przykładowo w celu powiązania użytkownika `wvh` systemu Linux z użytkownikiem `bill.vonhagen` systemu Windows należy w pliku `/etc/samba/smbusers` umieścić następujący wpis:

```
wvh = bill.vonhagen
```

Jeśli nazwa użytkownika Windows zawiera spacje, należy ująć ją w cudzysłowy.

W praktyce proponuję używać tych samych nazw użytkowników dla Ubuntu i Windowsa. Gdyby jednak nie było to możliwe, powyższy plik pozwoli rozwiązać problem odmiennych konwencji stosowanych do nadawania nazw użytkownikom w systemie Windows i Ubuntu.

Weryfikacja pliku konfiguracyjnego Samby

Po dokonaniu zmian w konfiguracji Samby należy sprawdzić, czy wprowadzone modyfikacje nie naruszają reguł składni pliku konfiguracyjnego. Jeśli plik konfiguracyjny Samby zawiera nieprawidłowe pozycje, podczas uruchomienia serwera pojawi się komunikat o błędzie i Samba nie zostanie uruchomiona. Takie zachowanie może okazać się bardzo stresujące, dlatego w pakiecie Samby znajduje się narzędzie sprawdzające poprawność konfiguracji i wskazujące miejsce wystąpienia wszystkich znalezionych błędów. Narzędzie to nosi nazwę `testparm`; uruchomione bez parametrów dokona sprawdzenia składni domyślnego pliku konfiguracyjnego Samby, czyli `/etc/samba/smb.conf`. Oto wynik przykładowego wywołania programu `testparm`:

```
$ testparm
Load smb config files from /etc/samba/smb.conf
Processing section "[homes]"
Processing section "[printers]"
Processing section "[print$]"
Loaded services file OK.
WARNING: passdb expand explicit = yes is deprecated
Server role: ROLE_STANDALONE
Press enter to see a dump of your service definitions
```

Jak widać w powyższym przykładzie, po wypisaniu informacji ogólnych, program `testparm` prosi użytkownika o naciśnięcie klawisza *Enter*, po czym wypisuje szczegółowe informacje na temat usług skonfigurowanych w naszej Sambie. Oto przykład tego typu szczegółowych informacji:

```
[global]
    workgroup = WVH
    server string = %h server (Samba, Ubuntu)
    obey pam restrictions = Yes
    passdb backend = tdbsam, guest
    syslog = 0
    log file = /var/log/samba/log.%m
    max log size = 1000
    name resolve order = lmhosts host wins bcast
    printcap name = cups
    panic action = /usr/share/samba/panic-action %d
    invalid users = root
    printing = cups
    print command =
    lpq command = %p
    lprm command =

[homes]
    comment = Katalogi domowe
    read only = No
    create mask = 0700
    directory mask = 0700
    browseable = No

[printers]
    comment = Wszystkie drukarki
    path = /tmp
    create mask = 0700
    printable = Yes
    browseable = No

[print$]
    comment = Sterowniki drukarek
    path = /var/lib/samba/printers
```

Wypisane informacje mogą być inne, zależy to od wersji Samby zainstalowanej w Ubuntu oraz zmian wprowadzonych w pliku konfiguracyjnym.

sprawdzanie dostępności i usług oferowanych przez Sambę

W pakiecie aplikacji Samba można znaleźć program *smbclient*, będący programem klienckim obsługiwany z wiersza poleceń, służącym do łączenia się z serwerem Samby i sprawdzania działających na nim usług. Narzędzie to może być również użyte do przetestowania poprawności działania serwera Samby oraz pozyskania listy oferowanych udziałów.

Po wprowadzeniu modyfikacji w konfiguracji serwera Samby (i zweryfikowaniu jej poprawności) należy ponownie uruchomić serwer. W tym celu należy wywołać następujące polecenie:

```
$ /etc/init.d/samba restart
```

To polecenie zamyka działający aktualnie serwer Samba i uruchamia go ponownie, co w praktyce wymusza ponowny odczyt konfiguracji i zastosowanie wprowadzonych modyfikacji.

Po wykonaniu tego polecenia można „odpytać” serwer Samba korzystając z aplikacji *smbclient*. W wywołaniu należy podać opcję *-L*, która spowoduje wypisanie listy dostępnych zasobów, nazwę hosta oraz opcję *-U%*, której użycie sprawi, że wypisane zostaną nazwy domyślnego konta, na jakie logują się anonimowi klienci. Oto przykładowy wynik działania tego polecenia:

```
$ smbclient -L ulaptop -U%
Domain=[WVH] OS=[Unix] Server=[Samba 3.0.22]

Sharename      Type      Comment
-----
ADMIN$         IPC       IPC Service (ulaptop server (Samba, Ubuntu))
IPC$           IPC       IPC Service (ulaptop server (Samba, Ubuntu))
print$         Disk      Sterowniki drukarek

Domain=[WVH] OS=[Unix] Server=[Samba 3.0.22]

Server          Comment
-----
ULAPTOP         ulaptop server (Samba, Ubuntu)

Workgroup       Master
-----
WVH             ULAPTOP
```

W pierwszej części można znaleźć listę zasobów dostępnych dla istniejącego użytkownika, który w tym przypadku jest użytkownikiem anonimowym (nieuwierzytelnionym). W drugiej części znajdują się informacje na temat serwera Samba oraz jego grupy roboczej lub domeny, której jest członkiem.

Nieźle. W następnym kroku warto sprawdzić, co zobaczy uwierzytelniony użytkownik. Wykorzystam przykładowego użytkownika *wvh*, którego utworzyłem wcześniej. Tym razem polecenie będzie miało następującą postać:

```
wvh@ulaptop:~$ smbclient -L ulaptop -Uwvh
Password:
Domain=[ULAPTOP] OS=[Unix] Server=[Samba 3.0.22]

Sharename      Type      Comment
-----
ADMIN$         IPC       IPC Service (ulaptop server (Samba, Ubuntu))
IPC$           IPC       IPC Service (ulaptop server (Samba, Ubuntu))
print$         Disk      Sterowniki drukarek
wvh            Disk      Katalogi domowe

Domain=[ULAPTOP] OS=[Unix] Server=[Samba 3.0.22]

Server          Comment
-----

Workgroup       Master
-----
WVH             ULAPTOP
```

Należy zauważyć, że narzędzie *smblclient* w pierwszej kolejności poprosiło o podanie hasła na serwerze Samba, ponieważ został zastosowany rzeczywisty login użytkownika. Wyświetlone informacje różnią się od informacji dostępnych dla anonimowego użytkownika. Najważniejsza różnica polega na tym, że uwierzytelniony użytkownik ma dostęp do swojego katalogu domowego na serwerze Samby, co zostało pokazane w sekcji *Katalogi domowe*, w sekcji *Sharename* wyniku działania programu *smblclient*.

Gratulacje! Wiemy już, że serwer Samba działa i jest prawidłowo skonfigurowany, pozwalając użytkownikom uzyskać dostęp do ich własnych zasobów; nadszedł czas, aby dodać zasoby systemowe, z których będą mogli korzystać wszyscy uwierzytelnieni użytkownicy. Na szczęście, o czym przekonamy się w następnym punkcie, można tego dokonać przy pomocy graficznego narzędzia konfiguracyjnego.

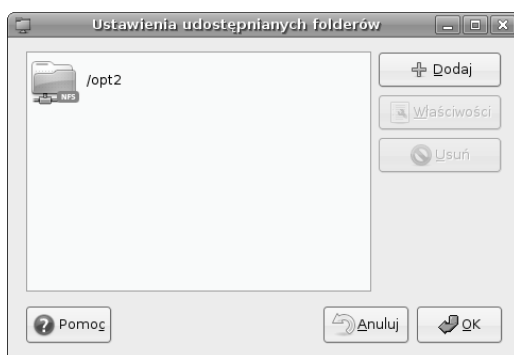
Wykorzystanie folderów udostępnionych

Choć Ubuntu nie posiada graficznych narzędzi do konfiguracji serwera Samba, to jednak oferuje proste w obsłudze narzędzie graficzne (*shares-admin*) do konfiguracji katalogów udostępnianych, które mają być serwowane przez Sambę. Ci, którzy udostępniali już zasoby przy pomocy NFS powinni znać to narzędzie — twórcy byli na tyle przebiegli, że za pomocą jednego narzędzia można skonfigurować zasoby udostępniane przez Sambę i NFS.

Narzędzie to uruchamia się z menu systemowego *System/Administracja/Foldery współdzielone*. Po wpisaniu hasła użytkownika zostanie otwarte okno przedstawione na rysunku 32.2, które pokazuje zasoby zdefiniowane w systemie, w tym przypadku znajduje się tam zasób */opt2* udostępniony za pomocą serwera NFS.

Rysunek 32.2.

Narzędzie Foldery współdzielone

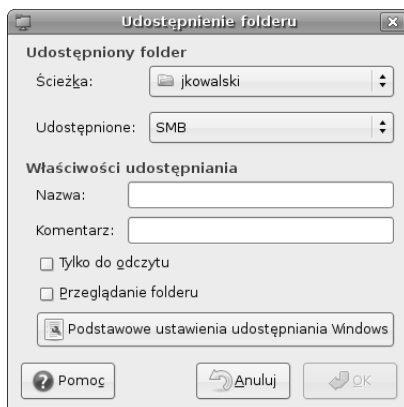


Aby zdefiniować katalog udostępniony przy pomocy serwera Samba, należy kliknąć przycisk *Dodaj*, co spowoduje otwarcie okna przedstawionego na rysunku 32.3.

Jak widać na rysunku 32.3., okno jest już wypełnione podstawowymi informacjami niezbędnymi do zdefiniowania zasobu SMB. Domyślnym katalogiem udostępnianym za pomocą tego narzędzia jest katalog domowy użytkownika. Nie jest to szczególnie interesujące, ponieważ Samba potrafi udostępniać te katalogi automatycznie. Dlatego w tym

Rysunek 32.3.

Definiowanie nowego
udostępnionego
folderu

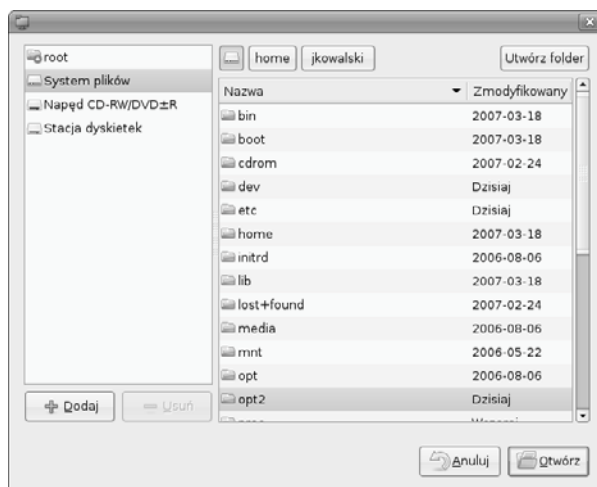


przykładzie posłużę się katalogiem zawierającym moją kolekcję muzyki (która, co muszę podkreślić, została legalnie zgrana z zakupionych przeze mnie płyt CD oraz z muzyki dostępnej legalnie za darmo w internecie).

Aby określić inny udostępniany katalog, należy kliknąć listę rozwijaną *Ścieżka* i wybrać opcję *Inne*, co spowoduje wyświetlenie okna wyboru katalogu, przedstawionego na rysunku 32.4.

Rysunek 32.4.

Wybór katalogu
do udostępnienia

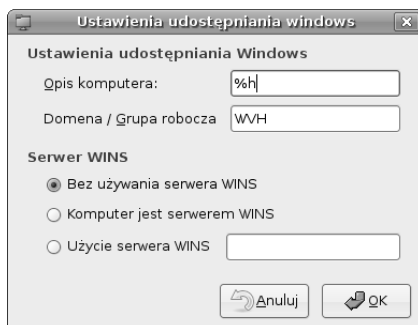


W tym oknie trzeba przejrzeć system plików w poszukiwaniu katalogu, który ma być udostępniony, w moim przykładzie jest to katalog */opt2*. Aby wybrać zaznaczony katalog, należy kliknąć przycisk *Otwórz*, co spowoduje powrót do okna przedstawionego na rysunku 32.3, tym razem jednak w polu *Ścieżka* będzie wprowadzona nazwa wybranego katalogu.

Następnie należy zidentyfikować ustawienia udostępniania oraz mechanizm wykorzystywany do uzyskania dostępu (zamontowania) tego zasobu. Aby zweryfikować te ustawienia, należy kliknąć przycisk *Podstawowe ustawienia udostępniania Windows*, co spowoduje wyświetlenie okna przedstawionego na rysunku 32.5.

Rysunek 32.5.

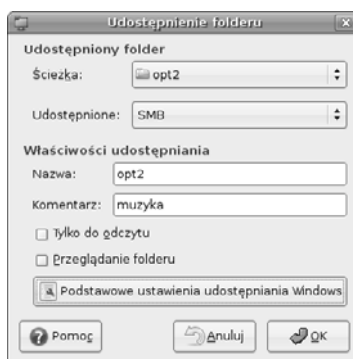
Ustawienia
udostępniania
Windows dla
udostępnianego
katalogu



To okno zawiera ustawienia udostępniania tego zasobu przez serwer Samba: grupą roboczą zasobu będzie *WVH*, do lokalizacji serwera nie będzie wykorzystywany mechanizm WINS (wystarczy plik hosts lub serwer DNS). Należy kliknąć *OK*, aby powrócić do okna przedstawionego na rysunku 32.3. Warto wprowadzić ogólne informacje na temat zasobu (pola *Nazwa* i *Komentarz*). W tym oknie można również ustawić opcję *Tylko do odczytu*, dzięki czemu definiowany zasób nie będzie „zapisywalny” dla zdalnych użytkowników. Jeśli to pole nie jest zaznaczone, każdy prawidłowo uwierzytelniony użytkownik będzie miał możliwość tworzenia i usuwania plików i katalogów w tym zasobie. Można też zaznaczyć opcję *Przeglądanie folderu*, dzięki której folder pojawi się na liście zasobów udostępnianych przez nasz serwer. Na rysunku 32.6 przedstawiam okno konfiguracji zasobu wypełnione podstawowymi informacjami.

Rysunek 32.6.

Uzupełnianie
informacji
o zasobie SMB



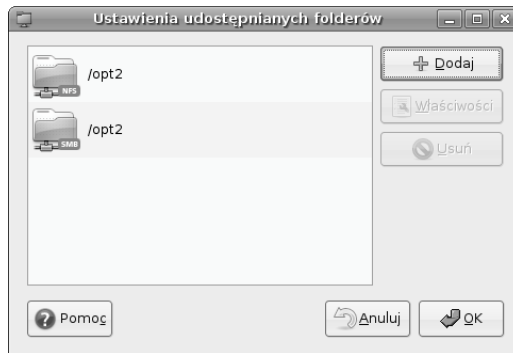
Aby kontynuować, należy kliknąć przycisk *OK*, co spowoduje ponowne wyświetlenie okna przedstawionego na rysunku 32.2, tym razem zawierającego nowo utworzony zasób. Nowo utworzony zasób *SMB* wraz z utworzonym wcześniej zasobem *NFS* przedstawiam na rysunku 32.7.

Prawie gotowe! Zdefiniowane zasoby można modyfikować, w tym celu wystarczy zaznaczyć zasób w oknie Ustawienia udostępnionych folderów i kliknąć przycisk *Właściwości*. Aby uaktywnić folder udostępniany, po jego zdefiniowaniu należy kliknąć przycisk *OK*, co spowoduje zamknięcie okna narzędzia *Foldery współdzielone*.

Jak można się było domyślić, narzędzie *Foldery współdzielone* w rzeczywistości dokonuje modyfikacji pliku konfiguracyjnego serwera Samba tak samo, jak zdefiniowanie zasobu

Rysunek 32.7.

Współdzielone
katalogi NFS i Samby
w narzędziu Foldery
współdzielone



NFS powodowało dopisanie pozycji w pliku */etc/exports*, definiującym zasoby serwera NFS. Zmiany zostały wprowadzone w pliku */etc/samba/smb.conf*, i w naszym przykładzie mają następującą postać:

```
[opt2]
path = /opt2
comment = Music
available = yes
browseable = no
public = yes
writable = yes
```

Dopisanie tych opcji do standardowego pliku konfiguracyjnego Samby spowoduje, że nowo zdefiniowane zasoby będą dostępne, gdy tylko zostanie uruchomiony serwer Samba.

Dalsze informacje o Sambie

Nie powinno dziwić, że sieć WWW jest doskonałym źródłem informacji na temat Samby. Tym bardziej zaskakujący jest sam projekt Samba, zawierający doskonałą dokumentację oraz wersję online książki na temat aktualnego wydania Samby (wersji 3.). Jak sugerowałem wcześniej, w Ubuntu warto zainstalować pakiety *samba-doc* i *samba-doc-pdf* zawierające kopie offline wspomnianej dokumentacji oraz książki. Dalsze informacje można znaleźć w następujących miejscach:

- ♦ **<http://www.samba.org>**: główna strona WWW projektu Samba, gdzie można znaleźć najświeższe porady, triki i kod źródłowy,
- ♦ ***/usr/share/doc/samba-doc/htmldocs/Samba3-HOWTO***: katalog zawierający dokument HOWTO w formacie HTML, omawiający sposób instalacji i konfiguracji Samby w wersji 3., można tam znaleźć odpowiedzi na często zadawane pytania (należy otworzyć w przeglądarce WWW zapisany w tym katalogu plik *index.html*); wersja HTML tego dokumentu jest dostępna w pakiecie *samba-doc*,
- ♦ ***/usr/share/doc/samba-doc/htmldocs/Samba3-ByExample***: katalog zawierający wersję HTML doskonałej, przeznaczonej dla praktyków książki o Sambie 3. Została ona napisana przez Johna Tepstrę, jednego z liderów projektu Samba, i wyjaśnia proces instalacji oraz konfiguracji Samby w wersji 3. w pełnym spektrum zastosowań sieciowych, od niewielkich sieci domowych i małych firm (SOHO),

czyli w zakresie omawianym w niniejszym rozdziale, aż po środowiska korporacyjne złożone z tysięcy użytkowników (należy otworzyć w przeglądarce WWW zapisany w tym katalogu plik *index.html*). Wersja HTML tego dokumentu jest dostępna w pakiecie *samba-doc*. Warto również kupić drukowaną kopię książki Johna — jest wyczerpującym i kompletnym źródłem informacji, a dodatkowo nie wymaga prądu.

Podsumowanie

W tym rozdziale wyjaśniam, w jaki sposób zainstalować i skonfigurować serwer Samba, za którego pomocą system Ubuntu może odczytywać dane zapisane na zdalnych systemach Windows i vice versa. Samba jest prawdopodobnie najlepszym przykładem wartości, jaką stanowi oprogramowanie open source dla nowoczesnych systemów informatycznych, dzięki niej współpraca między systemami uniksowymi a systemami Microsoft Windows jest możliwa w sposób praktycznie bezproblemowy, a to dzięki fantastycznej kombinacji inżynierii i doskonałych praktyk programistycznych.

Dodatki

Zawartość dysku CD

Ten dodatek zawiera informacje o zawartości dysku CD dołączonego do tej książki. Znajdują się tu następujące informacje:

- ♦ wymagania sprzętowe;
- ♦ wykorzystanie dysku CD w systemach Linux, Windows i Macintosh;
- ♦ zawartość dysku;
- ♦ rozwiązywanie problemów.

Wymagania sprzętowe

Przed wykorzystaniem dysku należy upewnić się, że posiadany komputer spełnia oczekiwane parametry opisane w tym podrozdziale. Jeśli komputer będzie posiadał słabsze parametry, mogą wystąpić problemy z wykorzystaniem zawartości dołączonego dysku CD.

Wymagania dotyczące systemów Windows 9x, Windows 2000, Windows NT4 (SP 4 lub nowszy), Windows Me, oraz Windows XP:

- ♦ komputer klasy PC z procesorem Pentium 120 MHz lub szybszym;
- ♦ co najmniej 96 MB pamięci RAM, dla optymalnej wydajności zaleca się co najmniej 128 MB RAM;
- ♦ karta sieciowa Ethernet lub modem o prędkości co najmniej 28 800 b/s;
- ♦ napęd CD-ROM.



Wymagania dla systemu Windows różnią się od wymagań dla Linuksa, ponieważ powyższe wymagania dotyczą jedynie wykorzystania oprogramowania dla systemu Windows dostępnego na dysku CD, a wymagania dla Linuksa dotyczą instalacji i wykorzystania systemu Linux dostępnego na dołączonym dysku CD.

Wymagania dotyczące Linuksa:

- ♦ komputer klasy PC z procesorem Pentium 90 MHz lub szybszym; dla optymalnej wydajności zaleca się Pentium III, Pentium 4 lub odpowiednik AMD pracujący z częstotliwością co najmniej 700 MHz;
- ♦ co najmniej 96 MB pamięci RAM, dla optymalnej wydajności zaleca się co najmniej 256 MB RAM;
- ♦ karta sieciowa Ethernet lub modem o prędkości co najmniej 28 800 b/s;
- ♦ napęd CD-ROM;
- ♦ co najmniej 3 GB wolnego miejsca na dysku twardym (w przypadku instalacji Ubuntu Linux na dysku).

Wymagania dotyczące systemów Macintosh:

- ♦ komputer Apple z procesorem G3 lub szybszym z systemem operacyjnym Mac OS 9.0 lub nowszym;
- ♦ co najmniej 96 MB pamięci RAM, dla optymalnej wydajności zaleca się co najmniej 256 MB RAM;
- ♦ co najmniej 3 GB wolnego miejsca na dysku twardym (w przypadku instalacji Ubuntu Linux na dysku).



W dalszej części dodatku znajdują się informacje niezbędne do określenia, czy z daną konfiguracją Apple Macintosh można wykorzystywać dołączony dysk CD, czy należy wyposażyć się w specjalną wersję Ubuntu CD.

Wykorzystanie dysku CD w Windows

Aby zainstalować w systemie Windows programy dostępne na tym dysku CD, należy wykonać następujące czynności:

1. Włożyć dysk CD do napędu CD-ROM.
Uruchomi się program instalacyjny pozwalający na instalację poszczególnych programów dla systemu Windows.
2. Aby zainstalować dowolny z nich, należy kliknąć przycisk *Install* obok nazwy programu. Więcej informacji na temat instalacji tych pakietów można znaleźć w rozdziale 2.

Jeśli w systemie nie jest aktywna funkcja *autorun* (automatyczne uruchamianie programów z dysku CD) lub jeśli nie pojawi się okno instalatora, należy wykonać następujące czynności:

1. Wywołać funkcję *Start/Uruchom*.
2. W oknie uruchamiania programu wpisać polecenie *d:\start.exe*, gdzie *d* to litera odpowiadająca napędowi CD-ROM. To uruchomi instalator opisany powyżej.
3. Wybrać oprogramowanie do instalacji zgodnie z opisem w rozdziale 2.

Wykorzystanie CD w Linuksie

Dysk CD dołączony do tej książki jest przeznaczony dla 32-bitowych systemów. Aby zainstalować lub przetestować Ubuntu Linux na 64-bitowej wersji procesora x86, należy pozyskać Ubuntu CD dla systemów 64-bitowych. Informacje dotyczące pobierania i wypalania dysków CD z systemem Ubuntu Linux w wersjach dla systemów 64-bitowych można znaleźć w rozdziale 2.

Aby przetestować lub zainstalować Ubuntu Linux z załączonego dysku CD, należy wykonać następujące działania:

1. Włożyć dysk CD do napędu CD-ROM.
2. Ponownie uruchomić komputer, wybierając rozruch z dysku CD-ROM. Jeśli komputer nie jest skonfigurowany odpowiednio do rozruchu z napędu CD-ROM, należy odpowiednio zmodyfikować ustawienia BIOS-u, ustawiając sprawdzanie napędu CD-ROM przed dyskiem twardym w sekwencji rozruchowej komputera.
3. Po pojawieniu się menu startowego należy nacisnąć klawisz *Enter* lub odczekać 30 sekund, co spowoduje uruchomienie zapisanego na CD systemu Ubuntu Linux.
4. Po uruchomieniu systemu można z nim normalnie pracować, testując jego funkcje, można też zainstalować go na dysku twardym zgodnie z opisem z rozdziału 2.
5. Aby usunąć dysk CD z napędu, należy wybrać funkcję *System/Zakończ* i w oknie zamykania systemu kliknąć przycisk *Wyłącz komputer*. W odpowiednim momencie wysunie się tacka napędu, a na ekranie pojawi się informacja o tym, aby wyjąć dysk CD i nacisnąć *Enter*.

Wykorzystanie CD w Mac OS

W komputerach Apple opartych na procesorach Intel Ubuntu Linux można zainstalować jako jedyny system, wykorzystując instrukcje z rozdziału 2. Instalacja Linuksa jako alternatywnego systemu (z wyborem w menu rozruchowym) jest możliwa jedynie w przypadku zastosowania oprogramowania Boot Camp firmy Apple, jednak tego zagadnienia nie będziemy omawiać w książce.

Dysk CD dołączony do tej książki jest przeznaczony jedynie dla 32-bitowych systemów x86 i nie będzie działał w systemach Mac OS opartych na architekturze PPC. Aby zainstalować Ubuntu Linux na tego typu systemach, należy zaopatrzyć się w dysk CD systemu Ubuntu Linux dla architektury PPC. Informacje dotyczące pobierania i wypalania dysków CD z systemem Ubuntu Linux w wersjach dla systemów PPC można znaleźć w rozdziale 2.



System Ubuntu można z łatwością zainstalować na komputerze Apple, wykorzystując instrukcje z rozdziału 2., ale wówczas będzie to jedyny system operacyjny w komputerze. Aby zainstalować Ubuntu obok systemu Mac OS, należy wygospodarować na dysku twardym partycję o rozmiarze co najmniej 3 GB. Jeśli system nie posiada partycji o odpowiednim rozmiarze, należy zastosować się do zaleceń z rozdziału 3. dotyczących zmiany konfiguracji partycji w systemie Mac OS.

Zawartość CD

Poniżej opisuję pokrótce zawartość dysku CD dołączonego do tej książki.

Aplikacje

Na dysku znajduje się przede wszystkim rozruchowa wersja systemu Ubuntu Linux opracowana przez Canonical, Ltd., którą można uruchomić lub zainstalować na komputerze. Na dysku znajduje się też kilka aplikacji na licencji open source, które można zainstalować w systemie Microsoft Windows:

- ♦ AbiWord to popularny procesor tekstu wchodzący w skład pakietu biurowego środowiska GNOME. Jest to wolna i darmowa alternatywa dla kosztownego, komercyjnego programu Microsoft Word.
- ♦ Gaim to komunikator internetowy stanowiący alternatywę dla komercyjnych rozwiązań, obsługujący wiele formatów komunikacji, jak AOL, Yahoo itp.
- ♦ Gimp to popularny program do tworzenia i obróbki grafiki, stanowiący wolną i darmową alternatywę dla kosztownych, komercyjnych programów, jak Adobe Photoshop. Wykorzystanie programu Gimp zostało omówione w rozdziale 12. „Używanie programu GIMP”.
- ♦ Mozilla Firefox to przeglądarka WWW na licencji open source, stanowiąca w systemie Windows wolną alternatywę dla komercyjnego programu Microsoft Internet Explorer. Podstawowe możliwości przeglądarki Firefox można rozszerzyć dzięki setkom wtyczek dostępnych za darmo w internecie. Wykorzystanie programu Firefox jest opisane w rozdziale 9.
- ♦ Mozilla Thunderbird to klient poczty elektronicznej (MUA) na licencji open source, zapewniający popularny mechanizm obsługi poczty w standardach POP/POP3 i IMAP.

Programy na licencji shareware to w pełni funkcjonalne wersje testowe płatnych programów komercyjnych. Jeśli ktoś zdecyduje się wykorzystywać tego typu program, powinien skontaktować się z jego autorem, uiścić odpowiednią opłatę i uzyskać pełną licencję na wykorzystanie, wersje rozszerzone programu oraz jego wsparcie techniczne.

Programy freeware to komercyjne gry, aplikacje i narzędzia dostępne do darmowego użytku osobistego. W przeciwieństwie do shareware te programy nie wymagają wnoszenia opłat, ale nie jest dostępne wsparcie techniczne.

Oprogramowanie GNU jest chronione jednolitą licencją o tej samej nazwie, której tekst jest dostępny w katalogu GNU w ramach instalacji.

Wersje testowe (*trial*), demonstracyjne (*demo*) lub ewaluacyjne to z reguły ograniczone wersje komercyjnych programów (udostępniane na określony czas lub pozbawione części funkcji, jak możliwość zapisu projektów). Niektóre ograniczone czasowo wersje testowe posiadają mechanizmy zabezpieczające przed zmianą daty systemowej. W ich przypadku próba zmiany daty systemowej najczęściej kończy się zablokowaniem działania programu.

Rozwiązywanie problemów

Dysk CD dołączony do tej książki zawiera pełną wersję systemu Ubuntu Linux, którą można uruchomić bez instalacji na dysku twardym lub zainstalować w komputerze. W przypadku problemów z instalacją Ubuntu warto skonsultować się z forum dyskusyjnym <http://www.ubuntuforums.org>.

W przypadku problemów z instalacją dostępnych na tym dysku CD aplikacji dla systemu Microsoft Windows należy spróbować jednego z poniższych rozwiązań:

- ◆ Tymczasowo wyłączyć zainstalowane oprogramowanie antywirusowe. Niektóre instalatory wykonują działania zbliżone do technik stosowanych przez wirusy komputerowe, co może spowodować, że oprogramowanie chroniące komputer podniesie fałszywy alarm lub zablokuje działanie instalatora. Po zainstalowaniu aplikacji należy oczywiście ponownie uaktywnić program antywirusowy.
- ◆ Zamknąć wszystkie uruchomione programy. Im większa liczba programów działa w systemie, tym mniejsza ilość pamięci jest dostępna dla innych programów. Instalatory najczęściej aktualizują pliki i programy; jeśli w czasie instalacji uruchomione są inne programy, działanie instalatora może zakończyć się niepowodzeniem.

Skorowidz

- %1, 213
- &, 213
- .bash_profile, 214, 215
- .bashrc, 214, 215
- .vimrc, 231
- .xinitrc, 180
- .xsession, 180
- /bin, 123
- /boot, 104
- /dev, 123
- /etc, 123
- /etc/exports, 891
- /etc/fstab, 129, 132
- /etc/group, 137, 658
- /etc/inittab, 591, 598
- /etc/pam.d, 659, 661
- /etc/passwd, 136
- /etc/skel, 215
- /etc/sudoers, 668
- /home, 104, 124
- /init, 589
- /lib, 123
- /linuxrc, 589
- /opt, 104, 124
- /proc, 123
- /sbin, 123
- /sys, 123
- /tmp, 104, 123
- /usr, 106, 124
- /usr/local, 106
- /var, 124
- /var/log, 124
- |, 212
- >, 212
- >>, 212
- 1000-BaseT, 752
- 100-BaseT, 752
- 10-Base2, 752
- 10-BaseT, 752
- 3d chess, 464

- 802.11, 774, 775
- 802.11a, 775
- 802.11b, 775
- 802.11g, 775
- 802.11n, 775

A

- AAC, 408, 410
- AbiWord, 77, 340
- ABR, 408
- Access Control Lists, 670
- access point, 764, 774
- access.conf, 666
- ACL, 670
 - aktywacja obsługi, 673
 - eiciel, 677
 - fstab, 673
 - instalacja poleceń, 672
 - narzędzia graficzne, 676
 - odczyt, 674
 - typy, 671
 - ustawianie, 675
 - wiersz poleceń, 674
- ad hoc, 774
- addgroup, 670
- adduser, 534, 651, 669
- admin, 669
- Administracja, 152
- Adobe PageMaker, 306
- AdobeInDesign, 306
- adres
 - e-mail, 254
 - IP, 752, 753
 - MAC, 834
 - URI, 801
 - URL, 793
- afpfs, 129
- After Step, 144

- AIFF, 408
- AisleRiot, 450
- Akcesoria, 150
- aktualizacja
 - oprogramowanie, 643
 - system, 625
- aktywacja
 - interfejs sieciowy, 757
 - obsługa ACL, 673
 - repozytoria, 608, 609
 - tryb persistent podczas uruchamiania jądra systemu, 75
- aktywatory aplikacji, 148, 165, 166
 - tworzenie, 167, 197
- ALAC, 408, 410
- alias, 217
- aliasy, 217
- alien, 644, 645
- Almquist, 194
- ALSA, 414
- Alternate CD, 43
- Alternate Install CD, 93, 114
 - instalacja serwera, 117
 - instalacja systemu w trybie tekstowym, 115
 - opcje instalacyjne, 114
 - rozdzielczość monitora, 116
 - tryb OEM, 116
- alternatywne repozytoria, 608
- Amanda, 693
- AMD, 42
- analiza procesu startowego, 592
- anonimowy serwer FTP, 493
- anti-aliasing, 384
- AOL, 282
- Apache, 113, 791, 794
 - access.log, 803
 - dzienniki systemowe, 803
 - error.log, 803
 - FollowSymlinks, 803
 - Indexes, 802
 - instalacja, 795, 797
 - instalacja z wiersza poleceń, 796
 - konfiguracja serwera, 799
 - MultiViews, 802
 - plik konfiguracyjny, 800
 - pliki serwera, 798
 - poziomy diagnostyczne, 804
 - uruchamianie serwera, 801
 - usuwanie problemów, 803
- Apache 2, 575, 798
- apache2ctl, 798
- aparat cyfrowy, 514, 517
 - gnome-volume-thumg-manager, 517
 - gthumb, 518
 - kopiowanie zdjęć, 518
- APE, 408
- aplety, 145, 148
- Aplikacje, 150
 - dodawanie programów do menu, 154
- Apple Lossless Audio Codec, 408
- Applixware, 341
- apt.conf, 614
- apt-cache, 620
 - policy, 616
 - search, 620
 - show, 621
- apt-file list, 624
- apt-file search, 623
- apt-get, 102, 354, 602, 603, 624, 625
 - aktualizacja systemu, 625
 - build-dep, 627
 - dist-upgrade, 625
 - install, 624
 - inteligentna aktualizacja systemu, 625
 - obsługa zależności kompilacyjnych, 627
 - pobieranie kodów źródłowych pakietów, 626
 - remove, 624, 632
 - source, 626
 - update, 613, 616
 - upgrade, 625
- aptitude, 102, 354, 602, 603, 620, 628
 - Conflicts, 631
 - Depends, 631
 - Enhances, 631
 - instalacja pakietów, 630
 - interfejs użytkownika, 629
 - Pre-Depends, 631
 - Recommends, 631
 - remove, 632
 - search, 620
 - show, 621
 - Suggests, 631
 - zalety stosowania, 632
- apt-key, 612, 613
- architektura systemowa, 42
- archiwa, 685
- archiwalne kopie zapasowe, 685
- archiwizacja, 696
- archiwizator taśmowy, 692
- argumenty polecenia, 187
- arkusz kalkulacyjny, 343, 344
 - Calc, 345, 361
 - formatowanie komórek, 353
 - formuły, 346
 - Gnumeric, 345, 353
 - komórka, 344
 - kopiowanie komórek, 348
 - określanie zakresu komórek, 347
 - opis pola, 351

sumowanie, 346, 350
 wyświetlanie obliczonej wartości, 348
 zakres komórek, 346
 zaznaczanie komórek, 352
 zaznaczanie zakresu, 349
 zmiana szerokości kolumny, 351
 ARPANET, 749, 847
 ASCII, 219
 ash, 194
 ATA, 730
 ATAPI, 730
 Ataxx, 454
 atlantik, 467
 audio, 408
 audio CD, 425
 Audio Interchange File Format, 408
 automatyczna konfiguracja sieci, 760
 automatyczne montowanie systemu plików
 podczas uruchamiania komputera, 132
 przenośne nośniki danych, 135
 automatyczne wykrywanie urządzeń, 727
 rozwiązywanie problemów, 728
 automount, 880
 Avancer Audio Coding, 408
 Average Bit Rate, 408
 azureus, 511

B

Backup Administrator, 690
 Backup Exec, 690
 BackupEDGE, 690
 BackupPC, 693, 700
 backuppc, 703
 config.pl, 701
 definiowanie kopii zapasowej, 705
 Direct restore, 712
 Download Tar archive, 713
 Download Zip, 713
 hosts, 712
 hosty, 704
 identyfikacja hostów, 704
 instalacja, 700, 702
 konfiguracja, 703
 mechanizmy wykonywania kopii, 701
 odtworzenie danych, 712
 rsyncd, 705
 SMB, 708
 transporty, 701
 uruchamianie kopii zapasowych, 709
 wybór hosta, 710
 Bacula, 694
 Base, 314
 base station, 774

bash, 191, 194, 207
 aliasy, 217
 historia poleceń, 208
 kontrola zadań, 213
 pliki konfiguracyjne, 214
 potoki, 211
 przekierowanie danych, 211
 uzupełnianie nazw plików, 209
 wyrażenia wieloznaczne, 210
 zmienné środowiskowe, 215
 bash.bashrc, 214
 BBS, 45
 BearShare, 505
 bezpieczeństwo, 749, 763
 hasła, 765
 konta użytkowników, 764
 pliki dziennika systemowego, 765
 sieć, 750
 sieć bezprzewodowa, 776
 stan zabezpieczeń, 765
 WEP, 776
 WPA, 776
 zapora sieciowa, 765
 zbędne usługi, 764
 bezpieczne połączenia, 473
 białe listy, 826
 Bible, 397
 biblioteki readline, 208
 BIND, 753, 847, 848, 849, 866
 \$TTL, 854
 adres IP serwera poczty, 858
 aliasy nazw hostów, 858
 CNAME, 858
 definicja strefy DNS, 854
 IN, 857
 instalacja, 851
 instalacja pliku odwzorowania odwrotnego, 860
 instalacja pliku strefy, 860
 klient nieświadczący usług DNS, 851
 MX, 857
 named.conf, 852, 853
 named.conf.local, 853, 860
 nazwy kanoniczne, 858
 nsswitch.conf, 852
 odwzorowania odwrotne DNS, 859
 plik danych dla zapytań odwrotnych, 854
 plik odwzorowań odwrotnych, 859
 plik strefy, 857
 pliki konfiguracyjne, 852
 ponowne uruchamianie serwera, 861
 PTR, 859
 resolv.conf, 852
 rozwiązywanie problemów, 863
 serwer podrzędny, 851

BIND

- serwer podstawowy, 851
- serwer wyłącznie buforujący, 851
- SOA, 854, 855
- strefy, 854
- testowanie serwera, 861
- tworzenie definicji strefy DNS, 854
- tworzenie pliku odwzorowań
 - odwrotnych DNS, 859
- tworzenie pliku stref, 857
- zapytania odwrotne, 854

bind9, 861

BIOS, 82, 584, 896

bit rate, 408

bitmapy, 384

bittornado-gui, 511

BitTorrent, 504, 509, 512

- klient, 511
- wyszukiwanie torrentów, 511

Biuro, 150

Black Box, 144

Blackjack, 451

blogi, 45

blokada producenta, 33

blokada wyskakujących okienek, 288

Bluetooth, 773

BMP, 384

BNC, 752

Bombich Software Carbon Copy Cloner, 85

Boot Chart, 592

Boot Device, 733

Boot Monitor, 584

BOOTP, 834

Bourne-Again, 194

BPS, 408

brama, 752

BSD, 220

BSS, 775

build-essential, 546

bwhois, 763

C

Calc, 314, 343, 345, 361

- Atrybuty komórek, 367
- format danych, 366
- formaty plików, 368
- funkcje, 364
- import arkuszy, 368
- instalacja, 361
- Konwerter dokumentów, 369
- menu, 363
- menu kontekstowe, 364
- nagłówki kolumn i wierszy, 364

- określanie rodzaju danych, 366
- pakiety lokalizacyjne, 362
- paski narzędziowe, 363
- rodzaj danych, 366
- uruchamianie programu, 363
- Waluta, 366

Canonical Ltd., 50

cardctl, 744

- ident, 744

Cat5, 751

CBR, 408

cd, 200

CD, 407, 408

CD Database, 409

CDDA, 409

CDDDB, 409

CDE, 146

CD-R, 409

CD-RW, 409

cgoan, 465

chac1, 674, 675

chgrp, 206

Chicken of the VNC, 480

chkrootkit, 765

chmod, 204, 205, 207

chown, 206

CIFS, 128, 500, 897

Citrix Metaframe, 489

clamav, 813

ClamAV, 825, 829

CNAME, 858

Common Desktop Environment, 146

Common Internet File System, 129

Common Unix Printing System, 868

Compact Disc, 408

Compact Disc Digital Audio, 409

CompactFlash, 125, 542

Concurrent Versioning System, 571

conduit, 522

CONFIG_INITRAMFS_SOURCE, 590

Constant Bit Rate, 408

cp, 201, 697

- kopia katalogu lokalnego, 697

cpio, 588, 691

cracker, 750

csh, 194

C-Shell, 194

CSLIP, 751

CSS, 425

CUPS, 868, 877

- access_log, 875

- drukarki, 872

- dzienniki, 875

- error_log, 875

integracja podsystemu wydruków Mac OS X, 872
 integracja podsystemu wydruków Windows, 871
 interfejs WWW, 870
 konfiguracja zdalnych hostów, 869
 obsługa wstępnie sformatowanych
 wydruków, 877
 odblokowanie dostępu do funkcji
 interfejsu administracyjnego, 876
 opcje administracyjne, 870
 page_log, 875
 usuwanie problemów z udostępnianymi
 drukarkami, 874
 CVS, 570, 571
 cyfrowa kamera wideo, 515
 cyfrowe odtwarzacze audio, 530, 541
 Cyrix, 42
 czarne listy, 826
 czas instalacji, 43
 Cztery w rzędzie, 458

D

DAE, 409
 dash, 194
 DAT, 409
 dcm, 529
 dd, 442, 691
 ddns-update-style, 840
 DEB, 627, 644
 debfoster, 647
 Debian, 39, 613
 Debian Almquist, 194
 Debian GNU/Linux, 23
 deborphan, 647
 debugger, 563
 default-lease-time, 840
 definiowanie dostępu do plików, 670
 defragmentacja dysku, 85
 Mac OS X, 86
 Microsoft Windows, 85
 Defragmentator dysku, 85
 demon SSH, 476
 Desktop CD, 43, 53
 dostęp do dysku twardego, 69
 instalacja programów dla Windows, 76
 instalacja Ubuntu, 56
 komputer 64-bitowy, 54
 kopiowanie plików do innego komputera
 dostępnego w sieci, 75
 PPC, 54
 tryb persistence, 73
 uruchamianie systemu z płyty, 55
 desktop publishing, 791
 df, 203, 503, 738

dhcp, 838
 DHCP, 106, 753, 833, 834
 broadcast, 834
 discover, 834
 dzierżawa adresu IP, 835, 836
 gdhcpd, 841
 instalacja serwera, 836
 klient, 834
 konfiguracja serwera, 838
 narzędzia graficzne, 841
 okres dzierżawy, 836
 problemy, 844
 protokół, 834
 rozwiązywanie problemów, 844
 serwer, 834, 836
 tryb ad-hoc, 844
 tryb interim, 844
 tworzenie plików konfiguracyjnych serwera, 838
 ustawienia serwera, 840
 zakres adresów IP, 842
 zarządzanie serwerem, 837, 841
 żądania, 834
 dhcp3, 841
 dhcpd3-server, 836, 844
 dialout, 520
 dig, 862
 Digital Audio Extraction, 409
 Digital Audio Tape, 409
 Digital Millenium Copyright Act, 409
 Digital Rights Management, 409
 Digital Signal Processing, 409
 Digital Video Disc, 409
 Disks Manager, 70
 diskutil, 90, 91
 DISPLAY, 144
 DistroWatch, 51
 dithering, 384
 DMCA, 409
 dmesg, 739
 DNS, 753, 756, 833, 847, 848, 866, 897
 A, 858
 adres IP serwera poczty, 858
 BIND, 849
 CNAME, 858
 hosts, 849
 IN, 857
 klient, 849
 MX, 857
 nsswitch.conf, 849
 pamięć podręczna, 850
 PTR, 859
 resolv.conf, 850
 resolver, 849
 serwer, 849

DNS

serwer najwyższego poziomu, 850

SOA, 855

strefy, 850

TTL, 850

zone, 850

dnsmasq, 838

Dodaj do panelu, 165

Dodaj/Usuń, 151, 155

dodawanie

drukarki, 718

dyski, 729

napędy CD/DVD, 729

napędy EIDE/ATA, 730

napędy SATA, 731

napędy SCSI, 732

pakiety, 633

repozytoria, 610

sprzęt, 717

dokumentacja, 48

dokumenty, 305, 306

języki znaczników, 306

LaTeX, 309, 312

OpenDocument, 315

RTF, 329

skład tekstu, 332

TeX, 307

tworzenie, 320

domenowe nazwy hostów, 849

domyślne uprawnienia, 139

domyślne ustawienia dźwięków, 414

dostęp do dysku twardego z systemu Desktop CD,
69

partycje Microsoft Windows, 69

partycje PPC Macintosh, 72

dostęp do katalogów NFS, 502

dostęp do zasobów systemów Windows, 499

Dostępność aktualizacji, 611

dostosowywanie

menu, 153

mysz, 159

panel, 162

pulpit, 159

system dźwiękowy, 414

właściwości paneli, 164

dowiązanie, 189

DPI, 386

dpkg, 424, 602, 603, 616

eksploracja systemu, 616

--info, 646

-l, 616

-L, 621

lista pakietów dostępnych do instalacji, 618

lista pakietów zainstalowanych w systemie, 616

odczyt informacji o zawartości pakietu, 621

określanie pakietu, do którego należy istniejący
plik, 622określenie pakietu, w którym znajduje się
brakujący plik, 623

--search, 622

Draw, 314, 397

instalacja, 397

linijki, 399

menu, 399

menu kontekstowe, 400

paski narzędzi, 399

Style i formatowanie, 400

uruchamianie programu, 398

DRM, 409, 531

drukarka, 718

sterownik, 721

typ połączenia, 720

USB, 718

właściwości, 722

właściwości wykrywania, 719

Drukarki w systemie Ubuntu, 720

drukowanie, 868

CUPS, 868

IPP, 868

LPD, 868

system wydruku, 868

dselect, 603, 628

DSL, 752

DSP, 409

dump, 687, 691

DVD, 407, 409, 440

DVI, 308

dvilj, 311

dvilj2p, 311

dvilj4, 311

dvilj6, 311

dvipdf, 311

dvips, 311

dvired, 311

dynamicmaps.cf, 819

dynamiczny adres IP, 753

dysk twardy, 125

Dyski, 736

formatowanie dysków, 737

informacje o partycjach, 737

dystrybucja Linuksa, 23, 34

dystrybuowanie kopii, 509

działanie w tle, 213

dzienniki systemowe, 124

Dźwięk i obraz, 151

dźwięki, 414

poziom głośności, 415

E

- e2label, 75
- eboard, 464
- echo, 215, 216
- eclipse, 554
- Eclipse, 552, 554
 - Build Automatically, 562
 - Build Project, 562
 - Console, 560
 - debugger, 563
 - dostosowywanie skrótów klawiszowych, 562
 - editor, 558
 - Editor, 559
 - edycja pliku Makefile, 569
 - feature, 558
 - funkcjonalność, 558
 - GDB Debugger, 563
 - import projektów, 564
 - instalacja, 554
 - interfejs użytkownika, 557
 - Managed Make C Project, 556, 560
 - Navigator, 559
 - nazwa projektu, 559
 - Outline, 560
 - perspective, 558
 - perspektywa, 558
 - pierwsze uruchomienie, 555
 - plugin, 558
 - pop3check, 565
 - projekty, 556
 - przestrzeń robocza, 558
 - skróty klawiszowe, 561
 - Standard Make C Project, 557
 - środowisko robocze, 558
 - tworzenie aplikacji w języku C, 555
 - tworzenie pliku źródłowego, 560
 - tworzenie projektu, 556
 - uruchamianie programu, 562
 - view, 558
 - widok, 558
 - workbench, 558
 - workspace, 558
 - wtyczka, 558
- eclipse-cdt, 554
- eclipse-ecj-gcj, 554
- eclipse-pydev, 554
- ed, 247
- editor, 558
- eDonkey2000, 504
- Edubuntu, 35
- edycja
 - grafika wektorowa, 401
 - tekst, 314
 - znaczniki ID3, 531
- edytor, 558
- edytor ACL, 676
- Edytor konta użytkownika, 652, 653
- Edytor menu Alacarte, 156
- edytor tekstowy, 220, 247
 - ed, 247
 - emacs, 221, 233
 - EMACS, 221
 - ex, 247
 - gedit, 246
 - jedit, 248
 - modalny, 222
 - nano, 247
 - niemodalny, 222
 - nvi, 221
 - pico, 248
 - TECO, 220
 - THE, 248
 - vi, 220, 221
 - vim, 221
- efekt schodków, 386
- eiciel, 673, 676
- EIDE, 730
- EIDE/ATA, 729, 730
- eject, 541
- eksploracja systemu plików, 175
- EM64T, 42
- emacs, 156, 209, 221, 233
 - bufory, 242
 - definiowanie funkcji, 245
 - dostosowywanie edytora, 243
 - ekran startowy, 239
 - funkcje, 245
 - kopiowanie, 242
 - modyfikacja ustawień trybów działania, 244
 - polecenia, 236
 - powiązania plików, 244
 - przemieszczanie kursora, 240
 - skróty klawiaturowe, 240
 - tryb fundamentalny, 244
 - uruchamianie, 238
 - ustawianie zmiennych, 244
 - wiele okien, 242
 - wklejanie, 242
 - wycinanie, 242
 - wyłączanie, 238
 - wyszukiwanie tekstu, 242
 - zastępowanie tekstu, 242
 - zmiana powiązań plików, 244
 - zmiana przypisania klawiszy, 243
 - zmiennne, 244
- EMACS, 221
- EMC Insignia Retrospect, 85
- EQ, 409

ESS, 775
ESSID, 778
esvn, 572
eth0, 762
Ethernet, 751, 752, 760
Evolution, 251
 adres e-mail, 254
 adresowanie wiadomości, 267
 automatyczne sprawdzanie
 niechcianej poczty, 278
 dopasowanie wiadomości do filtra, 277
 filtrowanie wiadomości, 258, 274
 filtry, 258, 275
 foldery, 270
 foldery wirtualne, 272
 foldery wyszukiwania, 272
 format HTML, 268
 formaty wiadomości, 267
 IMAP, 255
 konto pocztowe, 254
 kreator ustawień, 253
 MTA, 256
 Niechciana, 265
 niechciana poczta, 278
 Nowy, 264
 odbieranie wiadomości, 263
 Odebrane, 271
 Odpowiedz, 264
 Odpowiedz wszystkim, 264
 odzyskiwanie usuniętych wiadomości, 270
 opcje poczty wychodzącej, 260
 opcje wyświetlania wiadomości
 przychodzących, 269
 POP, 255
 Pożądana, 265
 Prześlij, 264
 Search Folders, 273
 serwer poczty przychodzącej, 256
 serwer poczty wychodzącej, 259
 SMTP, 259
 SpamAssassin, 278
 stosowanie filtrów, 275
 strefa czasowa, 261
 Szkice, 271
 Śmieć, 271
 Śmietnik, 271
 tworzenie folderu, 272
 tworzenie wiadomości, 267
 uruchamianie programu, 252
 Usuń, 265
 uwierzytelnianie, 257
 vfoldery, 272
 wiadomości, 263
 wiele kont, 262

Wychodzące, 271
Wydrukuj, 264
Wysłane, 271
wysyłanie wiadomości, 263, 266
wyszukiwanie, 272
Wyślij/Odbierz, 264
wyświetlanie wiadomości, 265
źródła informacji, 279

EWMH, 182
ex, 247
Examples, 67
Exim, 809
export, 216
exportfs, 889, 890
ext2, 126
ext3, 62, 127

F

faks, 322
Fast SCSI-2, 732
FastTrack, 504
FAT32, 69
fc, 208
fdisk, 73, 129, 739, 740, 741
Fedora Core, 586
FFP, 410
filtrowanie
 MAC, 776
 pakiety, 765, 766
finger, 763
Firebird, 282
Firefox, 281, 282
 baza dodatków, 300
 blokada wyskakujących okienek, 288
 ciasteczka, 292
 dane formularzy, 292
 Dodaj do zakładek, 295
 dodawanie motywów, 302
 dynamiczne zakładki, 288, 297
 elementy okna, 283
 główne okno, 284
 hasła, 292
 historia, 292
 historia pobierania, 292
 instalacja rozszerzeń, 299
 interfejs, 283
 Java, 291
 JavaScript, 290
 karty, 284, 287
 konfiguracja, 289
 kopia zapasowa zakładek, 297
 Menedżer zakładek, 296
 menu, 284

- menu kontekstowe, 286
- motywy, 299, 302
- mysz, 285
- obrazki, 291
- pamięć podręczna, 292
- panel boczny, 287
- panel historii, 287
- pasek narzędzi nawigacyjnych, 284
- pasek stanu, 284
- pasek zakładek, 284
- pobieranie rozszerzeń, 300
- Preferencje programu Firefox, 289
- Prywatność, 292
- przeglądanie w kartach, 287
- rozszerzenia, 299
- RSS, 288, 297
- strona domowa, 289
- tworzenie dynamicznych zakładek, 297
- tworzenie zakładek, 295
- uruchamianie programu, 283
- wyskakujące okienka, 290
- wyszukiwarka, 284
- zakładki, 294
- zarządzanie zakładkami, 296
- FireWire, 515
- FLAC, 428
- FLAC Footprint, 410
- flash USB, 74, 728
- FLOSS, 51
- Fluxbox, 144
- FOAF, 45
- Foldery współdzielone, 886, 906, 908
- fora internetowe, 45
- formatowanie
 - dysk, 736, 740
 - zewnętrzny nośnik pamięci, 74
- formatowanie plików DVI, 311
- formaty graficzne, 395
- formuły, 346
- Free Software Foundation, 32, 34
- Free Standards Group Open
 - Internationalization Initiative, 36
- FreeCell, 452
- Freedb, 410
- Free-java-sdk, 553
- Freenet, 504
- FreeNX, 488, 608
- Fridge, 48
- fruit, 463
- fsck, 127
- FSF, 34
- fstab, 129, 132, 673
- ftp, 496

- FTP, 76, 492, 493, 792
 - anonimowy serwer, 493
 - logowanie, 494
 - narzędzia graficzne, 499
 - narzędzia linii poleceń, 496
 - połączenie z serwerami, 493
 - przesyłanie plików, 497
 - put, 497
- full backup, 685
- fundacja Ubuntu, 38
- funkcjonalność, 558
- FVWM, 144

G

- g++, 547
- G3, 42
- Gaim, 77
- gateway, 752, 764
- gcc, 69, 546, 547
 - kompilacja, 549
- GCC, 545, 546, 548
- GCC C, 548
- gdb, 563
- gdhcpd, 841
- gedit, 246
- genext2fs, 590
- Genie-Soft Backup Manager, 85
- getfacl, 674, 676
- gFTP, 499
- GID, 654, 671
- GIF, 384
- GIMP, 33, 77, 387
 - dokowanie okien, 390
 - dostosowywanie rozmiaru obrazu
 - do wymiarów strony internetowej, 396
 - formaty graficzne, 395
 - główne okno narzędziowe, 389
 - konwersja obrazów, 394
 - odłączanie okien, 390
 - Opcje narzędzia, 390
 - operacje, 391
 - pedzle, 389
 - Skaluj obraz, 395
 - uruchamianie programu, 388
 - warstwy, 389
 - Zapisz jako, 394
 - zmniejszanie obrazów, 394
 - zrzut ekranu, 391
- GIMP Toolkit, 361
- gksudo, 668
- głębia koloru, 384
- głośność, 415
- główny kontroler domeny, 900

- główny sektor rozruchowy, 83
- GNOME, 35, 37, 143, 145, 147
 - aktywatory aplikacji, 148, 166
 - aplety, 148
 - Aplikacje, 148, 150
 - dolny panel, 148
 - dostosowywanie menu, 153
 - dostosowywanie myszy, 159
 - dostosowywanie paneli, 162
 - dostosowywanie pulpitu, 159
 - dostosowywanie właściwości paneli, 164
 - ekran logowania, 179
 - górny panel, 147
 - gry, 450
 - klawiatura ekranowa, 172
 - Kosz, 148
 - menedżer plików, 174
 - menu, 150
 - menu kontekstowe, 153
 - menu kontrolne okna, 149
 - menu Sesje, 179
 - Metacity, 149
 - Miejsca, 148, 151
 - motywy, 147, 170
 - Nautilus, 174
 - nenu na panelach, 150
 - okno aplikacji, 149
 - panele, 147, 162, 163
 - powiększanie ekranu, 172
 - przełączanie motywów, 170
 - przestrzeń robocza, 148
 - pulpit, 147
 - rozdzielczość wyświetlania, 161
 - skróty klawiaturowe, 173
 - System, 148, 152
 - technologie wspierające, 171
 - tematy, 171
 - terminal, 195
 - tło pulpitu, 169
 - wskaźnik myszy, 160
 - wygaszacz ekranu, 167
 - Zmiana sesji, 180
- GNOME On-Screen Keyboard, 171
- GNOME Partition Editor, 87
- gnome-app-install, 154, 156
- gnome-btdownload, 509
 - pobieranie pliku, 511
- gnome-chess, 463
- gnome-lokkit, 768, 769
- gnome-mag, 171
- gnome-panel, 147, 162
 - menu kontekstowe, 163
- gnome-pilot, 521
- gnome-session, 182
- Gnometris, 458
- gnome-volume-manager, 135
- gnome-volume-manager-gthumb, 515
- gnome-volume-thumg-manager, 517
- gnome-wm, 183
- gnopernicus, 171
- GNU, 32, 34
- GNU Compiler Collection, 548
- GNU Compiler Suite, 548
- GNU Image Manipulation Program, 33
- GNU Lesser Public License, 315
- gnugo, 465
- Gnumeric, 345, 353
 - filtry, 360
 - format danych, 357
 - Format komórek, 358
 - formaty plików, 360
 - funkcje, 355
 - import arkuszy, 359
 - menu, 354
 - menu kontekstowe, 355
 - nagłówki kolumn i wierszy, 355
 - określanie rodzaju danych, 357
 - paski narzędziowe, 354
 - pomoc, 356
 - rodzaj danych, 357
 - uruchamianie programu, 354
 - Waluta, 357, 359
- Gnutella, 504, 505
- go, 465
- gok, 171
- gpg, 615
- gpilot, 522
 - typy danych, 523
- graficzny interfejs użytkownika, 122
- grafika, 383
 - anti-aliasing, 384
 - bitmapy, 384
 - BMP, 384
 - dithering, 384
 - Draw, 397
 - efekt schodków, 386
 - formaty bezstratne, 385
 - formaty stratne, 385
 - GIF, 384
 - GIMP, 387
 - głębia koloru, 384
 - Inkscape, 401
 - JPEG, 385
 - mapy bitowe, 384
 - metadane, 385
 - monochromatyczna, 386
 - PCL, 386
 - piksele, 386

PNG, 386
 postrzępienie, 385
 PostScript, 386
 półtony, 385
 prepress, 386
 rastrowa, 386
 rozdzielczość, 386
 skala szarości, 385
 TIFF, 387
 warstwy, 385
 wektorowa, 387, 401
 Grafika, 151
 greylisting, 826
 group, 137, 655, 658
 group.conf, 666
 groupadd, 651
 groupdel, 651
 groupmod, 651
 groups, 534
 GRUB, 44, 65, 103, 132, 584
 konfiguracja, 587
 menu rozruchowe, 586
 modyfikacja konfiguracji, 734
 grub-doc, 587
 grupy, 136, 649, 650
 admin, 669
 ID, 650
 tworzenie, 650, 656
 zarządzanie, 658
 gry, 449
 3d chess, 464
 AisleRiot, 450
 Ataxx, 454
 atlantik, 467
 Blackjack, 451
 cgoban, 465
 Cztery w rzędzie, 458
 eboard, 464
 FreeCell, 452
 GNOME, 450
 gnome-chess, 463
 Gnometriss, 458
 gnugo, 465
 go, 465
 Iagno, 455
 karty, 450
 Klocki, 456
 knights, 464
 kości, 453
 Kulki, 454
 Mahjongg, 456
 MAME, 459
 MESS, 459

Miny, 457
 Monopoly, 467
 Nibbles, 459
 pasjanse, 450
 planszowe, 453
 Roboty, 460
 Same GNOME, 461
 Scrabble, 470
 szachy, 462
 Tali, 453
 Tetravex, 457
 wideo, 457
 Gry, 151
 GSS-API, 884
 GStreamer, 422
 instalacja, 422
 kodeki Microsoft Windows, 424
 odtwieranie płyt DVD, 425
 wtyczki, 422
 gthumb, 518
 GTK, 361
 gtkatlantic, 467
 gtk-gnutella, 505, 506
 instalacja, 505
 pobieranie pliku, 508
 gtkpod, 532, 534
 GUI, 122
 gunzip, 590
 gzip, 696

H

haker, 750
 HAP, 774
 Hard Disk Drives, 733
 Hardware Access Point, 774
 hasła, 102, 765
 HFS, 90
 HFS+, 72, 86
 historia poleceń, 208
 history, 208
 home, 122
 hopach, 763
 host, 763, 862
 hosts, 849
 hot band, 86
 Hot-File-Adaptive-Clustering, 86
 HotSync, 522
 HTML, 792, 793
 htpasswd, 703
 HTTP, 792
 httpd, 794
 hub, 752

I

- i18n, 36
- i386, 42
- Iagno, 455
- IBSS, 774
- ICCCM 2.0, 182
- IceWM, 181, 182
- ICEwm, 144
- ID3, 410, 531
- IDE, 551, 730
- identyfikacja
 - napędy, 738
 - partycje, 738
 - procesy, 213
- IEEE, 671
- IETF, 897
- ifconfig, 745, 762, 779
- IGS, 465
- IMAP, 252, 255, 258, 808, 824
- IMAP4rev1, 252
- Impress, 371
 - import prezentacji, 380
 - instalacja, 372
 - Konwerter dokumentów, 380
 - kreator prezentacji, 373
 - menu, 375
 - menu kontekstowe, 377
 - pakiety językowe, 373
 - panel edycji, 377
 - paski narzędziowe, 377
 - ramka slajdów, 377
 - ramka zadań, 377
 - slajd tytułowy, 379
 - szczegóły prezentacji, 375
 - tło, 374
 - tworzenie prezentacji, 377
 - uruchamianie programu, 373
- IN, 857
- inetd, 824
- info, 192, 193, 209
- Informacje o GNOME, 152
- informacje o karcie dźwiękowej, 418
- informacje o pakiecie, 621
- inicjujący dysk RAM, 588, 590
- inicjujący ramdysk, 587
- inicjujący system plików, 588
- inicjujący system plików RAM, 588, 590
- init, 589
- initial filesystem, 588
- initial RAM disk, 587, 588
- initial RAM filesystem, 588
- initng, 598
- initramfs, 588
- initrd, 587, 588, 590
- initscripts, 591
- inittab, 591, 598
- Inkscape, 401
 - instalacja, 402
 - linijki, 404
 - menu, 404
 - menu kontekstowe, 404
 - paski narzędziowe, 404
 - uruchamianie programu, 402
- instalacja
 - Apache, 795
 - kodeki Microsoft Windows, 424
 - oprogramowanie, 628, 638
 - programy dla Windows, 76
 - serwer BIND, 851
 - serwer DHCP, 836
 - serwer DNS, 851
 - serwer LAMP, 113
 - serwer NFS, 884
 - serwer Samba, 898
 - zaporą sieciową, 765
- instalacja serwera Ubuntu, 96
 - czas systemowy, 101
 - GRUB, 103
 - hasła, 102
 - język, 96
 - katalogi, 104
 - konto domyślnego użytkownika, 101
 - lokalizacja geograficzna, 97
 - nazwa komputera, 98
 - nazwa użytkownika, 101
 - partycjonowanie dysków, 98, 100
 - pierwsze uruchomienie systemu, 113
 - rozwiązywanie problemów, 105
 - strefa czasowa, 100
 - tryb podwójnego rozruchu, 99
 - układ klawiatury, 97
 - układ partycji, 104
- instalacja Ubuntu, 53, 81
 - informacje o użytkowniku, 60
 - język, 58
 - partycjonowanie dysku, 62
 - plyta Desktop CD, 56
 - tryb OEM, 116
 - tryb podwójnego rozruchu, 82
 - tryb tekstowy, 115
 - układ klawiatury, 59
- Intel, 42
- inteligentna aktualizacja systemu, 625
- interfejs sieciowy, 754
 - aktywacja, 757
 - DNS, 756

- Ethernet, 754
- konfiguracja, 754, 756
- sieci bezprzewodowe, 755
- internacjonalizacja, 36, 41
- internet, 752
- Internet, 151
- Internet Information Server, 282
- Internet Printing Protocol, 868
- Internet Relay Chat, 47
- internetowe radio, 431, 433
- interpreter linii poleceń, 185
- ipchains, 767
- ip-forwarding, 840
- iPod, 515, 530
 - bezpieczne odłączanie, 541
 - dodawanie katalogów z plikami MP3, 538
 - dodawanie plików audio, 536
 - DRM, 531
 - gtkpod, 532, 534
 - listy odtwarzania, 538
 - obsługa, 532
 - podłączanie, 533
 - punkt montowania, 536
 - zapisywanie zmian, 541
- IPP, 868
- iptables, 765, 766
- IPX, 896
- IRC, 47
- ISO, 44, 410, 444
- iTunes, 531
- iwconfig, 780
- iwlist, 780

J

- jakość, 40
- Java Development Kit, 553
- jądro, 34
 - ładowanie, 587
- jedit, 248
- język, 58
- język znaczników, 306
- JPEG, 384, 385, 394

K

- K Desktop Environment, 146
- K3b, 442
 - instalacja, 443
 - kopiowanie płyt DVD, 445
 - nagrywanie obrazów ISO, 444
- kalendarz wydań Ubuntu, 37
- kamera wideo, 514
- kanały RSS, 297

- kandydat do wydania, 40
- karta dźwiękowa, 417
 - informacje, 418
- karty, 450
- karty pamięci, 517, 542
 - CompactFlash, 542
 - SD, 542
- karty PCI, 745
- karty PCMCIA, 744
 - interfejs sieciowy, 745
 - zarządzania, 744
- katalogi, 104, 122, 123
 - główny, 122
 - tworzenie, 178
 - uprawnienia, 138
 - usuwanie, 204
- KDE, 145, 146, 179
- kdesvn, 572
- KDevelop, 552
- KDevelop3, 552
- Kerberos, 137, 474
- kernel, 34
- kill, 213
- killer app, 791
- klawiatura ekranowa, 172
- Klocki, 456
- Klondike, 451
- knights, 464
- kod źródłowy, 33
- kodek, 410
 - instalacja, 424
 - Shorten, 412
- kolejność sprawdzania urządzeń, 733
- komercyjne wsparcie techniczne, 49
- komórka, 344
- kompilacja, 549
- kompilator, 546
 - GCC, 548
- komputerowy skład tekstu, 332
 - ramka, 332
- komunikacja sieciowa, 752
- koncentrator, 752
- konfiguracja
 - automatyczne wykrywanie urządzeń, 727
 - interfejs sieci bezprzewodowej, 777
 - interfejs sieciowy, 754
 - modem, 759
 - napędy CD i DVD, 420
 - obsługa nośników pamięci, 516
 - połączenia modemowe, 757
 - połączenia PPP, 758
 - połączenia z publicznym serwerem FTP, 493
 - serwer Apache, 799
 - serwer DHCP, 833

konfiguracja

- serwer DNS, 847
- serwer NFS, 879
- serwer poczty, 807
- serwer Postfix, 819
- serwer Samba, 895, 899
- serwer WWW, 791
- serwer wydruków, 867
- sieć, 749
- sprzęt sieciowy, 754
- urządzenia dźwiękowe, 414
- urządzenia Palm OS, 521
- urządzenia PDA, 516
- zapora sieciowa, 767, 769

Konqueror, 281

konto pocztowe, 254

kontrola zadań, 213

konwersja

- obrazy, 394
- pakiety z innych formatów, 644
- pliki audio, 436
- pliki MP3, 422

kopie zapasowe, 681

- Amanda, 693
- archiwalne, 684, 685
- BackupPC, 693, 700
- Bacula, 694
- cpio, 691
- dd, 691
- dump, 687, 691
- komercyjne rozwiązania, 690
- lokalne, 690
- lokalne nośniki wymienne, 695
- migawkowe, 684
- nośniki, 683
- odtworzenie, 690, 712
- oprogramowanie, 85, 689, 690
- pełne, 685
- planowanie strategii, 687
- przyrostowe, 684
- restore, 691
- rsync, 694, 698
- sieciowe, 693
- strategia wykonywania, 687
- tar, 692, 696
- testowanie, 688
- typy, 684
- urządzenia, 688
- weryfikacja, 688
- wybór plików, 688
- wykonywanie, 690, 695
- zastosowanie, 682

kopiowanie

- katalogi, 176
- pliki, 176, 200

pliki do innego komputera dostępnego w sieci, 75

plyty DVD, 442, 445

Kosz, 148

koszty, 33

kraker, 750

kronika, 90

kronikowanie, 90, 127

ksh, 194

ktorrent, 511

Kubuntu, 35

Kulki, 454

kwm, 144, 145

KWord, 341

L

LAMP, 96

LAN, 774

latex, 310

LaTeX, 305, 309

narzędzia graficzne, 311

źródła informacji, 314

LDAP, 137, 252

less, 192

LGPL, 315

Li18nux, 36

libapache2-svn, 572

libmultisync-plugin-all, 526

LiLo, 132, 584

LimeWire, 505

limits.conf, 666

Line Printer Daemon, 868

linia poleceń, *Patrz* wiersz poleceń

Linux, 23, 32, 121

Linux for people, 40

Linux Standard Base, 124

Linux Terminal Server, 489

Lisp, 553

lista pakietów dostępnych do instalacji, 618

lista pakietów zainstalowanych w systemie, 616

listy dyskusyjne, 47

live CD, 53, 75

Logical Volume Management, 44

login, 659

logo Kubuntu, 68

logowanie, 114, 662

logowanie na zdalnych systemach, 474

lokalizacja, 36

lokalizacja geograficzna, 58

lokalizowanie dysków, 736

lokalny system plików, 126

Lokkit, 767

aktywacja zapory sieciowej, 772

hosty sieci lokalnej, 770

- instalacja, 767
- konfiguracja zapory sieciowej, 769
- opcje DHCP, 771
- otwieranie dostępu do usług, 771
- wybór typu zapory, 769

LONE-TAR, 690

loopback, 761

LPD, 868

LPI, 386

ls, 187

- a, 188

- aF, 189

- al, 190

- F, 188

- l, 188

- ld, 138

LSB, 124, 644

lspci, 746

lsusb, 718

LTS, 38

LVM, 44

Lyx, 312

LZW, 384

Ł

ładowane moduły jądra, 718

ładowanie

- inicjujący dysk RAM

- lub system plików RAM, 588

- jądro, 587

- moduły PAM, 659

łączenie się z innymi systemami, 473

- rdesktop, 483, 484

- serwer usług terminalowych Windows, 482

- SSH, 474

- tsclient, 485

- VNC, 477

M

M4A, 410

MAC, 834

Mac OS X, 54, 144

macierze RAID, 683

Mahjongg, 456

mail, 819

Mail Delivery Agent, 808

Mail Transfer Agent, 808

Mail User Agent, 808

mailscanner, 813

MailScanner, 825, 829

mailx, 813

main.cf, 819

make, 546, 547, 550

makefile, 550

MAME, 459

man, 192

manifest Ubuntu, 36

mapy bitowe, 384

marshalling, 882

Master Boot Record, 44, 83

master.cf, 819

max-lease-time, 840

MBR, 44, 83, 584

MD5, 410

md5sum, 410

MDA, 254, 808

Menedżer aktualizacji, 604, 611, 643

Menedżer dysków, 70, 71

menedżer okien, 144, 178

- IceWM, 181, 182

- Metacity, 149

- zmiana, 180

Menedżer pakietów Synaptic, 604, 633

menedżer plików, 174

- Nautilus, 174

menedżer urządzeń, 747

menu, 150

- Aplikacje, 150

- dodawanie pozycji, 156

- dostosowywanie, 153

- kontekstowe, 153

- Miejsca, 151

- System, 152

MESS, 459

Metacity, 144, 145, 149

metadane, 385

Microsoft PowerPoint, 380

Microsoft Windows, 54, 144

Microsoft Windows CE, 525

Miejsca, 151, 493, 500

mieszanie repozytoriów Ubuntu i Debiana, 613

minimalne wymagania sprzętowe, 42

Miny, 457

mkfs, 740, 743

mkswap, 740, 744

modem, 757

moduły jądra, 718

moduły PAM, 659

modyfikacja sekwencji uruchamiania

- skryptów usług, 596

Monitor systemu, 166

Monopoly, 467

montowanie, 126, 131

- automatyczne podczas uruchamiania

- komputera, 132

- fstab, 132

- montowanie
 - opcje, 134
 - partycje Windows, 71
 - system plików, 129
 - wymienne nośniki, 728
- Montowanie wymiennych nośników po podłączeniu, 515
- Montowanie wymiennych nośników po włożeniu, 515
- Mosaic, 282
- Motion Picture Association of America, 411
- Motor, 552
- motywy, 145, 147, 170
 - przełączanie, 170
- mount, 73, 503, 674, 729
 - opcje, 134
- Mount Point, 64
- Mozilla Firefox, 77
- Mozilla Organization, 282
- Mozilla Thunderbird, 77
- MP3, 411, 422
- MP3 CD, 411
- MP4, 410
- MPAA, 411
- MPEG, 411
- MPEG-1, 413
- MPEG-1 Audio Layer 3, 411
- MPEG-4, 411
- MTA, 254, 256, 808
- MUA, 254, 808
- mulitsync, 524
- multi-homed systems, 755
- multimedia, 407
 - AAC, 408
 - ABR, 408
 - AIFF, 408
 - ALAC, 408
 - ALSA, 414
 - APE, 408
 - bit rate, 408
 - CBR, 408
 - CD, 408
 - CDDA, 409
 - CDDb, 409
 - CD-R, 409
 - CD-RW, 409
 - DAE, 409
 - DAT, 409
 - DMCA, 409
 - DRM, 409
 - DSP, 409
 - DVD, 409
 - EQ, 409
 - FFP, 410
 - formaty bezstratne, 410
 - formaty stratne, 410
 - Freedb, 410
 - GStreamer, 422
 - ID3, 410
 - internetowe radio, 431
 - ISO, 410
 - kodek, 410
 - konfiguracja urządzeń dźwiękowych, 414
 - konwersja plików audio, 436
 - M4A, 410
 - MD5, 410
 - MP3, 411
 - MP3 CD, 411
 - MP4, 410
 - MPAA, 411
 - MPEG, 411
 - MPEG-4, 411
 - nagrywanie płyt, 429
 - napędy CD/DVD, 420
 - normalizacja, 411
 - OGG, 411
 - P2P, 411
 - PCM, 411
 - płyty audio CD, 425
 - płyty DVD, 440
 - podcasty, 435
 - poziom głośności, 415
 - RealAudio, 411
 - redukcja szumów, 411
 - Rhythmbox, 431
 - RIAA, 412
 - ripowanie, 412
 - Serpentine, 429
 - SHN, 412
 - skórka, 412
 - Sound Juicer, 426, 427
 - strumieniowe przesyłanie danych, 412
 - SVCD, 412
 - tack-at-once, 412
 - Totem, 440
 - transkodowanie, 412
 - TTA, 412
 - VBR, 408, 413
 - VCD, 413
 - WAV, 413
 - Wavpack, 413
 - WMA, 413
 - WMV, 413
 - wypalanie, 413
 - zgrzywanie płyt audio CD, 427
 - zgrzywanie płyt DVD, 442
 - zaczyniki, 410
 - zrzucanie, 412
 - źródła dźwięku, 430

- multisync, 517, 525
 - biblioteki, 526
 - definiowanie par do synchronizacji, 527
 - instalacja, 526
 - pary synchronizacyjne, 527
 - synchronizacja danych, 529
 - wtyczki, 526, 528
- multiverse, 605, 607
- mv, 202
- MX, 857
- MySQL, 113
- mysz, 145
 - dostosowywanie działania, 159
 - przyciski, 146
 - wskaznik, 160

N

- nadawanie uprawnień, 655
 - uprawnienia domyślne, 139
- nagrywanie płyt, 429
 - obrazy ISO, 444
 - płyty DVD, 442
- named, 852
- named.conf, 852
- named-checkzone, 864
- nano, 247
- napędy
 - CD/DVD, 420
 - EIDE/ATA, 730
 - FireWire, 727
 - SATA, 731
 - SCSI, 732
 - USB, 727
- Napędy i nośniki wymienne, 727
- narzędzia do zdalnego łączenia, 488
- narzędzia programistyczne, 546
 - build-essential, 546
 - GCC, 546, 548
 - IDE, 551
 - kompilatory, 546
 - make, 546, 550
 - pliki makefile, 550
 - system kontroli wersji, 570
 - zintegrowane środowiska programistyczne, 551
- Narzędzia sieciowe, 761, 762
 - narzędzia diagnostyczne, 762
- nash, 194
- NAT, 753, 765, 766
- Nautilus, 159, 174
 - eksploracja systemu plików, 175
 - kopiowanie plików i katalogów, 176
 - menu kontekstowe, 175
 - operacje, 174

- przenoszenie plików i katalogów, 177
- Skopiuj, 176
- tworzenie katalogów, 178
- ustawienia uprawnień, 205
- Wklej, 176
- Wklej do folderu, 176
- zaznaczanie wielu plików i katalogów, 177
- zmiana nazwy pliku lub katalogu, 177
- nazwa komputera, 98
- NCSA, 282, 794
- NDIS Wrapper, 783, 786
- ndisgtk, 783, 784
- ndiswrapper, 782, 783, 784, 787
- netatalk, 129
- NetBEUI, 896
- NetBIOS, 896
- netfilter, 765, 766
- netstat, 763
- Network Backup, 690
- Network File System, 128
- Network Information System, 880
- Network Manager, 760, 788
- network-admin, 186, 754, 777
- NFS, 85, 128, 502, 879
 - /etc/exports, 891
 - automatyczne montowanie katalogów
 - domowych, 893
 - błędy komunikacji, 882
 - definiowanie eksportów, 891
 - definiowanie folderu udostępnionego, 886
 - definiowanie użytkowników, 888
 - exportfs, 890
 - Foldery współdzielone, 886, 889
 - instalacja serwera, 884
 - komunikacja RPC, 882
 - marshalling, 882
 - pakowanie, 882
 - portmapper, 882
 - protokoły komunikacji, 881
 - protokół, 880
 - rozpakowywanie, 882
 - RPC, 881
 - rpciod klienta, 883
 - serwer, 879, 882, 884
 - specyfikacja eksportu, 890
 - udostępnianie plików, 880
 - unmarshalling, 882
 - uwierzytelnianie, 891
 - użytkownicy, 891
 - wersje, 883
 - weryfikacja działania mechanizmu, 889
 - współdzielenie zbioru binariów między systemami, 881
 - wykorzystanie zasobów, 880
 - XDR, 882

NFS v3, 881
nfs-common, 885
nfs-kernel-server, 885
Nibbles, 459
niechciana poczta, 278
NIS, 137, 880
nmap, 763, 765
nm-applet, 788
NNGS, 465
NoMachine NX Server, 488
normalizacja, 411
normalize, 411
nośniki danych, 683
Notatki, 166
Novell Core Protocol, 129
Novell Netware, 129
nslookup, 763, 862
nsswitch.conf, 849, 852
NTFS, 69
nvi, 221

O

O Ubuntu, 152
obniżanie jakości pliku JPEG, 396
obróbka grafiki, 383
obsługa
 karty pamięci, 542
 multimedia, 608
 poczta elektroniczna, 251
 POP/POP3, 823
 skanery, 722
 urządzenia elektroniczne, 514
oczko, 451
odbieranie poczty elektronicznej, 263
odczyt informacji o pakiecie, 621
 zawartość pakietu, 621
odłączanie iPoda, 541
odtworzenie audio, 515, 530, 541
odtworzenie
 dane z kopii zapasowej, 712
 internetowe stacje radiowe, 433
 muzyka, 431
 pliki MP3, 422
 płyty CD, 426
 płyty DVD, 422, 425, 440
odzworowania odwrotne DNS, 859
odzworowanie nazw domenowych na adresy IP, 849
OEM, 27, 44, 116
OGG, 411
Ogg Vorbis, 411, 436
okno aplikacji GNOME, 149
oocalc, 363
oodraw, 398

oimpress, 373
oowriter, 315, 317
opcje poleceń, 187
open relay, 822
OpenDocument, 315
OpenOffice.org, 306
 instalacja, 315
 pakiety językowe, 316
 rozszerzenia plików dokumentów, 317
OpenOffice.org Calc, 345, 361
OpenOffice.org Draw, 397
OpenOffice.org Impress, 371
OpenOffice.org Writer, 314
OpenSSH, 476
openssh-server, 476
OPML, 45
oprogramowanie, 546, 601
 aktualizacja, 643
 instalacja, 628, 638
 konwersja pakietów z innych formatów, 644
 repozytoria, 604
 usuwanie, 624, 628, 641
 utrzymanie porządku w systemie, 647
optymalizacja procesu rozruchowego, 594
Original Equipment Manufacturers, 44
OSX VNC, 480
Otwarcie położenia, 173
otwarty pośrednik, 822

P

P2P, 411, 491, 503
 udostępnianie plików, 503
pakiet biurowy, 314, 340
pakiety, 605
 DEB, 627
 eksploracja systemu, 616
 informacje, 621
 instalacja, 638
 konwersja z innych formatów, 644
 odczyt informacji o zawartości, 621
 repozytoria, 604
 usuwanie, 641
 źródłowe, 548
pakiety sieciowe, 752
Palm, 516
Palm OS, 516, 517
palmtop, 516, 520, 522
 konfiguracja urządzeń, 521
 synchronizacja urządzeń, 521
PAM, 137, 659
 brak plików konfiguracyjnych, 666
 kolejność ładowania modułów, 659
 konfiguracja plików modułów, 666

- pliki konfiguracji, 661
- proces logowania w systemie, 662
- reguły sprawdzające siłę hasła, 665
- pam_env.conf, 666
- pam_env.so, 663
- pam_group.so, 663
- pam_lastlog.so, 664
- pam_limits.so, 664
- pam_mail.so, 664
- pam_motd.so, 664
- pam_nologin.so, 663
- pam_securetty.so, 663
- pam_unix.so, 664, 665
- Pamięć masowa, 515
- pamięć wirtualna, 62, 740
- panele, 162
 - aktywatory aplikacji, 166
 - dostosowywanie właściwości, 164
 - dostosowywanie zawartości, 165
- parametry startowe jądra, 76
- parted, 92
- Partition, 64
- partitions, 130
- partycje, 62, 104, 125, 129, 737
 - Microsoft Windows, 69
 - PPC Macintosh, 72
 - punkt montowania, 110
 - tworzenie, 64
 - wymiany, 62
- partycjonowanie dysku, 62, 736, 740
- pary synchronizacyjne, 527
- pasjanse, 450
- passwd, 136, 137, 655
- PATA, 730, 731
- PATH, 215, 216
- PC Speaker, 417
- PCI, 745
- PCL, 386
- PCM, 411, 417
- PCMCIA, 744
- PDA, 516
- PDC, 900
- PDF, 311, 405
- pdksh, 194
- PDP-10, 220
- peer-to-peer, 411, 491
- pełne kopie zapasowe, 685
- pen drive, 74
- Penguin Liberation Front, 609
- Perl, 113
- persistent, 75
- perspektywa, 558
- phalanx, 463
- Phoenix, 282
- pico, 248
- Pida, 552
- pierwsze uruchomienie systemu, 66, 113
- piksele, 386
- Pilot Applet, 521
- pilot-link, 521
- ping, 763
- pipe, 212
- PKG, 645
- Plam OS, 520
- Planet Ubuntu, 45
- Planet-pl, 45
- pliki, 122
 - ASCII, 219
 - dowiązanie, 189
 - DVI, 308
 - dźwiękowe, 431
 - kopiowanie, 176, 200
 - makefile, 550
 - PDF, 311
 - przenoszenie, 177, 200
 - tekstowe, 219
 - ukryte, 189
 - uprawnienia, 138, 670
 - usuwanie, 200
 - zmiana nazwy, 177, 200
 - zmiana uprawnień, 204
- pliki konfiguracyjne powłoki, 214
- plyta live CD, 53
- plyty audio CD, 425
- plyty DVD, 440
 - K3b, 442
 - kopiowanie, 442
 - nagrywanie, 442
 - odtwarzanie, 440
 - zgrywanie, 442
- plyty instalacyjne Ubuntu, 43, 93
 - Alternate CD, 43
 - Desktop CD, 43
 - obrazy ISO, 44
 - Server CD, 43
- PNG, 384, 386, 401, 593
- pobieranie kodów źródłowych pakietów, 626
- poczta elektroniczna, 251
 - Exim, 809
 - filtrowanie wiadomości, 274
 - formaty wiadomości, 267
 - greylisting, 826
 - IMAP, 252
 - konfiguracja mechanizmów antyspamowych, 825
 - konfiguracja mechanizmów antywirusowych, 825
 - MailScanner, 825
 - MDA, 254, 808
 - MTA, 254, 808

poczta elektroniczna

- MUA, 254, 808
- odbieranie wiadomości, 263
- POP, 252, 808
- Postfix, 809, 812
- postgrey, 826
- protokoły, 252
- Qmail, 810
- Qpopper, 823
- Sendmail, 810
- serwer, 807, 811
- SMTP, 252, 808
- spam, 825
- SpamAssassin, 825
- szare listy, 826
- wirusy, 825
- wysyłanie wiadomości, 263, 266

podcasty, 435

podłączanie iPoda, 533

podmenu, 150

podwójny rozruch, 65

point-to-point, 491, 492, 504, 749

polecenia, 199

- addgroup, 670
- adduser, 534, 651, 669
- alias, 217
- apt-get, 625
- aptitude, 620, 628
- argumenty, 187
- bwwhois, 763
- cardctl, 744
- cd, 122, 200
- chac1, 674, 675
- chgrp, 206
- chkrootkit, 765
- chmod, 204, 205, 207
- chown, 206
- cp, 201, 697
- cpio, 588, 691
- dd, 442, 691
- df, 203, 503, 738
- dig, 862
- diskutil, 90
- dmesg, 739
- dpkg, 424
- dvilj, 311
- dvilj2p, 311
- dvilj4, 311
- dvilj6, 311
- dvipdf, 311
- dvips, 311
- dvired, 311
- e2label, 75
- echo, 215, 216

eiciel, 676

- eject, 541
- evolution, 253
- export, 183, 216
- exportfs, 889, 890
- fc, 208
- fdisk, 739, 740, 741
- finger, 763
- firefox, 283
- ftp, 496
- getfacl, 674, 676
- gimp, 388
- gksudo, 668
- gpg, 615
- groupadd, 651
- groupdel, 651
- groupmod, 651
- groups, 138, 534
- history, 208
- host, 763, 862
- ifconfig, 745, 762, 779
- info, 192, 193
- informacje, 192
- inkscape, 402
- iptables, 765, 766
- iwconfig, 780
- iwlist, 780
- kill, 213
- latex, 310
- less, 192
- ls, 138, 187
- lspci, 746
- lsusb, 718
- mail, 819
- make, 546, 550
- man, 192
- mkfs, 740
- mkswap, 740, 744
- mount, 131
- mv, 202
- mysql, 113
- netfilter, 765, 766
- netstat, 763
- network-admin, 754
- newgrp, 138
- nmap, 763, 765
- nslookup, 763, 862
- oocalc, 363
- oodraw, 398
- ooinpress, 373
- oowriter, 315, 317
- opcje, 187
- parted, 91
- ping, 763

- ul>
- popd, 200
- potoki, 193
- printenv, 215
- pushd, 200
- pwd, 200
- rdesktop, 484
- restore, 691
- rm, 203
- rmdir, 204
- rpm, 646
- rsync, 698
- scp, 474
- set, 215
- setfacl, 674, 675
- setq, 244
- sftp, 474
- smbclient, 904
- smbpasswd, 901
- ssh, 474
- startx, 181
- strings, 838
- su, 139, 140, 667
- sudo, 139, 140, 206, 667, 668
- svn, 574
- svnadmin, 573
- testparm, 903
- tex, 308
- texmacs, 312, 313
- traceroute, 763
- umask, 139
- useradd, 651
- userdel, 651
- usermod, 651
- visudo, 668
- vncviewer, 481
- wget, 424
- whereis, 207
- whois, 763
- xninit, 181
- xterm, 196
- połączenia
 - modemowe, 757
 - PPP, 758
 - TCP/IP, 751
 - z innymi systemami, 473
 - z publicznym serwerem FTP, 493
- pomoc online, 47
- POP, 252, 255, 258, 808, 823, 824
- POP3, 808, 823, 824
- pop3check, 565, 567
- popd, 200
- poprawki bezpieczeństwa, 38
- port wychodzący, 752
- portmapper, 882
- poruszanie się po systemie plików, 200
- POSIX, 671
- postfix, 813
- Postfix, 809, 812
 - ClamAV, 827, 829
 - dynamicmaps.cf, 819
 - greylisting, 826
 - identyfikacja zaufanych hostów i domen, 822
 - instalacja, 812, 817
 - instalacja z wiersza poleceń, 814
 - Internet site, 813
 - Internet site using smarthost, 814
 - konfiguracja, 819
 - Local delivery only, 814
 - MailScanner, 825, 827, 829
 - main.cf, 819, 822
 - masquerade_domains, 822
 - master.cf, 819
 - modele konfiguracji, 813, 815
 - mydestination, 823
 - mynetworks, 822
 - nazwa serwera, 816
 - No configuration, 814
 - odrzućcie niechcianej poczty, 826
 - opcje konfiguracyjne, 820
 - otwarty pośrednik, 822
 - pakiety, 813
 - pliki konfiguracyjne, 819
 - polityka pośredniczenia, 822
 - postgrey, 826
 - przepisywanie adresów
 - w poczcie wychodzącej, 822
 - przyjmowanie poczty dla całej domeny, 823
 - Satellite system, 814
 - spam, 825
 - SpamAssassin, 827, 829
 - TLS, 820
 - wirtualne adresy e-mail, 819
 - wirusy, 825
- postfix-doc, 813
- postgrey, 813, 826
- postrzępienie, 385
- PostScript, 386
- potoki, 193, 211
- PowerPC, 42, 54
- powiększanie ekranu, 172
- powłoka, 191
 - ash, 194
 - bash, 191, 194
 - Bourna, 191
 - csh, 194
 - dash, 194
 - ksh, 194
 - nash, 194

powłoka

- pdksh, 194
- plik konfiguracyjny, 214
- potoki, 193
- sh, 194
- skrypty, 193
- tcsh, 194
- Terminal, 195
- Z, 194
- zsh, 194
- poziom głośności, 415
- poziomy pracy, 589, 590
- półtony, 385
- PPC, 54
- PPD-11, 220
- PPI, 386
- PPP, 529, 751, 754, 758
 - konfiguracja połączenia, 758
- Preferencje, 152
- Preferencje myszy, 160
- prepress, 386
- prezentacje, 343, 371
 - tworzenie, 377
- printenv, 215
- proces rozruchu, 82, 583
 - analiza procesu, 592
 - BIOS, 584
 - Boot Chart, 592
 - Boot Monitor, 584
 - inicjujący ramdysk, 587
 - inicjujący system plików, 588
 - inicjujący system plików RAM, 588
 - init, 589
 - initramfs, 588
 - initrd, 588, 590
 - inittab, 591
 - ładowanie inicjującego dysku RAM
 - lub systemu plików RAM, 588
 - ładowanie jądra, 587
 - MBR, 584
 - menu rozruchowe, 586
 - optymalizacja procesu, 594
 - poziomy pracy, 590
 - program rozruchowy, 584
 - skrypty startowe, 591
 - sysv-rc-conf, 594
 - Ubuntu, 584
 - Zarządca programów startowych, 594
- proces uruchamiania komputera, 82
- proces zamykania systemu, 598
- procesor tekstu, 305, 314, 340
- procesy, 213
 - init, 589
- profile, 214
- profile sieciowe, 760

Profile użytkowników, 653

program rozruchowy, 132, 584

GRUB, 585

LiLo, 585

programowanie, 546

projekt

Debian, 39

GNU, 34

Ubuntu, 31

protokoły

BOOTP, 834

DHCP, 753, 833

FTP, 493

HTTP, 792

IMAP, 258

IPX, 896

LPD, 868

NetBIOS, 896

NFS, 880

NIS, 137

POP, 258, 808

PPP, 754, 757

RDP, 483

RPC, 881

SMB, 500

SMTP, 252, 259, 808

SPX, 896

SSH, 474

TCP/IP, 751

UDP, 751

WEP, 776

WINS, 901

WPA, 776

przeciągnij i upuść, 145

przeglądanie sieci WWW, 281

przeglądarki WWW, 281, 792

przekierowanie danych, 211

przełączanie motywów, 170

przełącznik, 752

przenoszenie

katalogi, 177

pliki, 177, 200

przenośne napędy, 727

przestrzeń robocza, 148

przesyłanie plików, 491, 492

dostęp do zasobów systemów Windows, 499

FTP, 492

NFS, 502

Samba, 499

SMB, 500

przydzielanie adresów IP, 833

przyłączanie

karty PCI, 745

urządzenia peryferyjne, 717

przywracanie danych, 696
 ps, 549
 PTR, 859
 publikowanie dokumentów, 305
 pulpit, 57, 66, 143
 dostosowywanie, 159
 GNOME, 147
 zmiana tła, 169
 Pulse Code Modulation, 411
 punkt dostępowy, 764, 774
 punkt montowania, 71, 110, 125
 punkt rozszerzający, 775
 pushd, 200
 pwd, 200

Q

Qmail, 810
 Qpopper, 823
 QT, 332
 qtorrent, 511
 Quark XPress, 306

R

RAID, 44, 683
 ramka, 332
 rapidsvn, 572
 Raport pogodowy, 166
 raw device, 688
 rcp, 474
 rdesktop, 483, 484, 486
 RDF, 45
 RDP, 483
 readline, 208
 RealAudio, 411, 412
 RealOne, 412
 RealPlayer, 412
 RealVNC, 481
 RecoverEDGE, 690
 redukcja szumów, 411
 regulacja dźwięku, 417
 Regulacja głośności, 166, 417
 Reiser4, 127, 743
 ReiserFS, 127, 743
 rekord SOA, 855
 relase candidate, 40
 Remote Desktop Protocol, 483
 remote sync, 694
 renderowanie, 792
 repartycjonowanie dysku, 84
 GNOME Partition Editor, 87
 graficzne programy, 91
 Micromats TechTool Pro, 86

PPC Macintosh, 90
 Ubuntu Desktop CD, 87
 repozytorium, 604
 aktywacja, 609
 aktywacja elementów, 606
 alternatywne, 608
 backports, 606
 Debian, 614
 dodawanie, 610
 FreeNX, 608
 main, 605, 606
 mieszanie repozytoriów Ubuntu i Debiana, 613
 multiverse, 605, 607, 608
 niestandardowe, 610
 obsługa multimediów, 608
 Penguin Liberation Front, 609
 restricted, 605, 606
 sources.list, 607
 universe, 605, 608
 Ustawienia oprogramowania, 609
 Wine, 609
 resolv.conf, 850, 852
 resolver, 849
 restore, 691
 restricted, 605
 reverse map file, 859
 Rhythmbox, 431
 biblioteka utworów, 432
 importowanie plików, 432
 odtwarzanie internetowych stacji radiowych, 433
 podcasty, 435
 tworzenie list odtwarzania, 433
 RIAA, 412
 rio, 542
 rioutils, 542
 ripowanie, 412
 rm, 203
 rm -r, 204
 rmdir, 204
 Roboty, 460
 root, 139
 root filesystem, 585
 rootkity, 765
 ROSETA, 37
 router, 752
 ROX, 145
 rozdzielczość, 386
 monitor, 116
 wyświetlanie, 161
 Rozdzielczość ekranu, 161
 rozruch systemu, 583
 rozwiązywanie problemów
 serwer DNS, 863
 urządzenia, 746

RPC, 881
rpciod, 883
RPCSEC_GSS, 884
rpm, 646
RPM, 644, 645
RSA, 475
rsh, 474
RSS, 45, 288, 297
rsync, 694, 698, 701
 argumenty, 700
 wykonywanie kopii katalogu, 698
rsyncd, 701
rsyncd.conf, 699
RTF, 329
runlevel, 589

S

Samba, 129, 499, 895, 897
 definiowanie zasobów, 907
 domeny, 900
 Foldery udostępnione, 906
 Foldery współdzielone, 906
 główny kontroler domeny, 900
 grupy robocze, 900
 identyfikacja grupy roboczej lub domeny, 900
 instalacja serwera, 898
 konfiguracja serwera, 899
 konfiguracja uwierzytelniania, 901
 PDC, 900
 serwer, 895, 898
 SMB, 896
 smb.conf, 899, 900
 sprawdzanie dostępności, 904
 udostępnianie zasobów, 902
 uwierzytelnianie, 901
 weryfikacja pliku konfiguracyjnego, 903
 WINS, 901
Same GNOME, 461
SANE, 722, 726
SAP, 775
SATA, 729, 730, 731, 734
Sbcl, 553
SCC, 570
sep, 76, 474
Scrabble, 470
Scribus, 332
 Aktualizuj ramkę tekstową i zakończ, 336
 biuletyn, 337
 Dopasuj ramkę do obrazka, 337
 edycja ramek, 335
 Edytor artykułów, 334
 Edytuj tekst, 335
 Edytuj tekst w edytorze artykułów, 334

 Edytuj zawartość ramki, 334
 grafika, 337
 instalacja, 332
 łączenie ramek tekstowych, 338
 szablony, 333
 wybór szablonu, 334
script kiddies, 750
SCSI, 730, 732, 735
SD, 542
Sendmail, 259, 810
Serpentine, 429
Server CD, 43
Server Install CD, 93, 94
 instalacja serwera, 96
 menu instalacyjne, 94
Server Message Block, 128
serwer
 Apache, 791
 BIND, 847, 849
 DHCP, 753, 833, 834
 DNS, 756, 847
 FTP, 492
 LAMP, 113
 LDAP, 137
 NFS, 879, 884
 pocztowy, 807, 811
 Samba, 895, 898
 Subversion, 575
 usługi terminalowe Windows, 482
 vino, 478
 VNC, 477, 479
 WWW, 791, 794
 wydruków, 867, 869
set, 215
setfacl, 674, 675
setq, 244
sftp, 474
sh, 194
shadow, 655
Shared Folders, 76
shares-admin, 906
shell, 191
ShipIt, 44
SHN, 412
Shorten, 412
sieciowe kopie zapasowe, 693
sieciowe systemy plików, 128
sieć, 749, 751
 adres IP, 752
 bezpieczeństwo, 750, 764
 brama, 752
 diagnostyka, 762
 DNS, 756, 848
 dyskietkowa, 491

- Ethernet, 751
- filtrowanie pakietów, 766
- Gnutella, 505
- komunikacja, 752
- konfiguracja, 754
- lokalna, 751
- loopback, 761
- NAT, 753, 765, 766
- P2P, 491, 503
- pakiety, 752
- piesza, 491
- połączenia modemowe, 757
- przydzielanie adresów IP, 833
- punkt dostępowy, 764
- TCP/IP, 751
- testowanie, 761
- urządzenia, 752
- wiele konfiguracji, 760
- WWW, 792
- zaporą sieciową, 765
- Sieć, 760, 761
- sieć bezprzewodowa, 755, 773, 774
 - 802.11, 774
 - 802.11a, 775
 - 802.11b, 775
 - 802.11g, 775
 - 802.11n, 775
 - access point, 774
 - ad hoc, 774
 - bezpieczeństwo, 776
 - BSS, 775
 - ESS, 775
 - ESSID, 778
 - filtrowanie MAC, 776
 - HAP, 774
 - IBSS, 774
 - instalacja sterowników systemu Windows, 781, 784
 - iwconfig, 780
 - iwlist, 780
 - klucz WEP, 778
 - konfiguracja interfejsów, 777
 - managed, 774
 - narzędzia do konfiguracji, 780
 - ndiswrapper, 782, 783
 - Network Manager, 788
 - punkt dostępowy, 774
 - punkt rozszerzający, 775
 - SAP, 775
 - SSID, 776
 - stacja bazowa, 774
 - standardy, 775
 - sterowniki dla systemu Windows, 781
 - WEP, 776, 778
 - WPA, 776
 - zarządzana, 774
 - zarządzanie interfejsami, 788
- SISSL, 315
- skala szarości, 385
- skaner, 722
 - konfiguracja preferencji detekcji, 723
 - podgląd obszaru skanowania, 724
 - rozdzielczość skanowania, 725
- SANE, 722
 - tryb skanowania, 725
 - XSane Image Scanner, 722
- Skaner obrazów XSane, 723
- skanowanie, 724
- skanowanie portów, 763
- skład tekstu, 332
- skórka, 412
- skrótów klawiaturowe w GNOME, 173
- skrypty powłoki, 193
- skrypty startowe, 591
- SlashDot, 503
- SLIP, 751
- SLP, 645
- Smart backup, 690
- smartfon, 516, 520
- smb, 701
- SMB, 85, 128, 500, 896
- smb.conf, 899
- smbclient, 904
- smbpasswd, 901
- SMTP, 252, 259, 808
- sneaker net, 491
- SOA, 855
- software, 546
- Software Access Points, 775
- SOHO, 814, 900
- Sound Converter, 436
 - instalacja, 436
 - konwersja plików, 438
 - wybór plików, 438
- Sound Juicer, 426, 427
- source code control, 570
- sources.list, 106, 422, 607, 610, 643
- spam, 825
- SpamAssassin, 278, 825, 829
- specjalne uprawnienia, 139
- SPI, 386
- społeczność Ubuntu, 24, 41, 45
- sprawdzanie spójności systemu plików, 127
- Sprawdzanie ze słownikiem, 165
- sprzęt, 42, 717
- SPX, 896
- ssh, 474

- SSH, 126, 474
 - demon, 476
 - klucz RSA, 474
 - logowanie na zdalnych systemach, 474
 - OpenSSH, 476
 - uruchamianie demonu, 476
- sshd, 476
- SSID, 776
- stacja bazowa, 774
- StarOffice, 315, 370
- statyczny adres IP, 753
- stos, 200
- strefa czasowa, 59
- strefy DNS, 850
- strings, 838
- strona domowa, 289
- strona WWW, 791, 793
- strony man, 192
- strumieniowe przesyłanie danych, 412
- su, 667
- submenu, 150
- subversion, 572
- Subversion, 571
 - Apache 2, 576
 - dokonywanie zmian w repozytoriach, 579
 - import projektu do repozytorium, 574
 - instalacja, 571
 - konfiguracja repozytoriów, 573
 - konfiguracja serwera, 575
 - obsługa adresów typu svn, 578
 - pobieranie plików z repozytoriów, 578
 - serwer, 575
 - sprawdzanie plików, 575
 - Svnserve, 577
 - tworzenie repozytoriów, 573
 - WebDAV, 573
- subversion-helper-scripts, 572
- sudo, 206, 667, 668, 687
 - !fqdn, 669
 - !lecture, 669
 - konfiguracja polecenia, 667
 - naprawa uszkodzonego polecenia, 670
 - przydzielanie uprawnień administracyjnych, 669
 - tty_tickets, 669
- sudoers, 668
- suid, 687
- Sun UltraSPARC, 42
- Super Video CD, 412
- surfowanie, 281
- surowy dostęp do urządzenia, 688
- SVCD, 412
- SVG, 402, 405
- svn, 574
 - ci, 579
 - co, 578
 - import, 574
 - list, 575
- svnadmin create, 573
- Svnserve, 575, 577
- swap, 62
- switch, 752
- synaptic, 602, 603
 - Aktualizacja systemu, 636
 - dodawanie pakietów, 633
 - dokumentacja, 634
 - instalacja pakietów, 638
 - konfiguracja ustawień, 635
 - paneje, 633
 - Traktowanie rekomendowanych pakietów jako zależności, 635
 - usuwanie pakietów, 633, 641
 - Wykonywanie zmian w oknie terminala, 636
 - wyszukiwanie oprogramowania, 636
- Synaptic, 102
- SyncBackSE, 85
- synce-pstatus, 530
- synce-serial-config, 527, 529
- synchronizacja
 - urządzenia Palm OS, 521
 - urządzenia z systemem Microsoft Windows, 525
- Synchronizuj urządzenia Palm™
 - po podłączeniu, 517
- Synchronizuj urządzenia PocketPC
 - po podłączeniu, 517
- sysinit, 591
- syslog, 664
- System, 152
- system dźwiękowy, 414
- system kontroli wersji, 570
 - CVS, 570, 571
 - gałęzie, 570
 - konflikty między zmianami, 570
 - merging, 570
 - SCC, 570
 - Subversion, 571
- system operacyjny, 31, 34
- system plików, 124, 743
 - automatyczne montowanie podczas uruchamiania komputera, 132
- CIFS, 128
- ext2, 126
- ext3, 127
- lokalny, 126
- montowanie, 126, 129
- NFS, 128, 879
- Novell Netware, 129
- partycje, 125
- punkt montowania, 125
- Reiser4, 127
- ReiserFS, 127

- sieciowy, 128
- sprawdzanie spójności, 127
- standardowy, 126
- tworzenie, 743
- XFS, 127
- z kroniką, 126, 127
- system wielosieciowy, 755
- system wydruku, 868
- system X Window, 144
- SYSV R3, 194
- SysVInit, 591, 598
- sysv-rc-conf, 594
- szachy, 462
- szarych list, 826
- Szuflada, 165

Ś

- środowisko graficzne, 143
 - CDE, 146
 - GNOME, 145, 147
 - KDE, 145, 146
 - menedżer okien, 144, 178
 - mysz, 145
 - ROX, 145
 - X Window, 143
 - X.org, 144
 - Xfce, 145
 - zmiana menedżera okien, 180
- środowisko robocze, 558

T

- Tali, 453
- tar, 692, 696, 701
 - archiwizacja, 696
 - kompresja, 696
 - opcje, 696
 - przywracanie danych, 696
- TCP/IP, 494, 751
- tcsh, 194
- technologie wspierające, 171
- TECO, 220
- telnet, 473
- telnetd, 474
- tematy, 171
- TENEX C-Shell, 194
- TERM, 630
- termcap, 220
- Terminal, 73, 195
 - kodowanie znaków, 196
- testowanie
 - kopie zapasowe, 688
 - serwer DNS, 861

- sieć, 761
 - system dźwiękowy, 414
- Ubuntu, 66
- testparm, 903
- tetex-base, 307
- tetex-bin, 307
- tetex-extra, 307
- Tetravex, 457
- tex, 308
- TeX, 306, 307
 - DVI, 308
 - instalacja systemu, 307
 - narzędzia graficzne, 311
 - znaczniki, 307
 - źródła informacji, 314
- texmacs, 312, 313
- TGZ, 645
- THE, 248
- The Ubuntu Marketplace, 50
- Thunderbird, 282
- TIFF, 387
- TightVNC, 481
- tihtgvncserver, 479
- time.conf, 666
- tło pulpitu, 169
- tlumaczenie, 36
- Totem, 440
- traceroute, 763
- track-at-once, 412
- transkodowanie, 412
- True Audio codec, 412
- tryb persistence, 73
 - aktywacja podczas uruchamiania jądra, 75
- tryb podwójnego rozruchu, 65, 82
 - przygotowywanie systemu, 83
- tsclient, 483, 485
- TTA, 412
- TTL, 850, 855
- twm, 144
- tworzenie
 - aktywatory aplikacji, 167, 197
 - aplikacje w języku C, 555
 - grupy, 650, 656
 - katalogi, 178
 - oprogramowanie, 545
 - partycja wymiany, 740
 - partycje, 64
 - pliki makefile, 550
 - prezentacje, 371, 377
 - profile sieciowe, 761
 - system plików, 740, 743
 - użytkownicy, 650, 651
- typy kopii zapasowych, 684

U

- uaktualnienia systemu, 38
- Ubuntu, 23, 31, 35
 - architektury systemowe, 42
 - społeczność, 41
 - uaktualnienia, 38
 - wsparcie systemu, 38
 - wydania, 37, 40
 - wymagania instalacyjne, 41
 - zalety stosowania, 40
- Ubuntu 6.06 LTS, 24
- Ubuntu Announcement, 48
- Ubuntu Blog, 51
- Ubuntu Desktop CD, 54
- Ubuntu Development, 48
- Ubuntu live CD, 54
- Ubuntu Security Announcement, 48
- Ubuntu Server, 93
- Ubuntu User, 48
- Ubuntu Women, 48, 51
- UbuntuGuide, 51
- Ubuntux, 51
- UCB, 220
- UCE, 825
- udhcpd, 838
- udostępnianie
 - drukarki, 902
 - obrazy ISO, 605
- udostępnianie plików, 499, 880, 902
 - BitTorrent, 504, 509
 - gnome-btdownload, 509
 - gtk-gnutella, 505
 - NFS, 502
 - P2P, 503
 - połączenie z zasobami, 501
 - Samba, 499
 - sieć Microsoft Windows, 896
- UDP, 751
- UID, 654, 671, 677
- układ klawiatury, 59
- układ partycji, 104
- ukryte pliki, 189
- umask, 675
- universe, 605
- Universe, 179
- Unsolicited Commercial E-mail, 825
- update-manager, 603
- uprawnienia, 136, 649, 653, 670
 - domyślne, 139
 - katalogi, 138
 - nadawanie, 655
 - PAM, 137
 - passwd, 137
 - pliki, 138
 - specjalne, 139
 - sprawdzanie, 138
 - tryb zapisu, 204
 - zmiana, 204
- URI, 606, 793
- URL, 793, 897
- URN, 793
- uruchamianie
 - demon SSH, 476
 - komputer, 82
 - system po dodaniu nowych urządzeń, 733
 - system z płyty Desktop CD, 55
- uruchamianie Ubuntu, 65
 - GRUB, 65
 - pierwsze uruchomienie, 66
 - tryb podwójnego rozruchu, 65
- urządzenia
 - dźwiękowe, 414
 - loopback, 761
 - pamięci masowe, 125
 - PCI, 746
 - SCSI, 730
 - USB, 722
- Urządzenia, 746
 - zaawansowane informacje, 747
- urządzenia elektroniczne, 513
 - aparat cyfrowy, 514, 517
 - iPod, 515, 530
 - kamera wideo, 514
 - karty pamięci, 517
 - konfiguracja obsługi, 517
 - konfiguracja systemu, 514
 - odtwarzacz audio, 515, 530
 - Palmtop, 516, 520
 - PDA, 516
 - smartfon, 516, 520
- USB, 515, 718
- useradd, 651
- userdel, 651
- usermod, 651
- usługi terminalowe Windows, 482
- Ustawienia dźwięku, 415
- Ustawienia oprogramowania, 609
 - Aktualizacje internetowe, 611
 - Dodaj kanał, 610
 - dodawanie niestandardowego repozytorium, 610
 - dodawanie repozytorium, 610
 - Kanał, 610
 - niestandardowe repozytoria, 612
 - Uwierzytelnianie, 611
- Ustawienia tła pulpitu, 170
- Ustawienia wygaszacza ekranu, 168
- Ustawienie wymiennych napędów i nośników, 514

- usuwanie
 - katalogi, 204
 - konto użytkownika, 655
 - oprogramowanie, 624, 628, 633, 641
 - pliki, 200
- UUCP, 749
- uwierzytelnianie, 137, 659
 - PAM, 659
- uzupełnianie nazw plików, 209
- użyteczność, 41
- użytkownicy, 136, 649, 650
 - ID, 650
 - modyfikacja ustawień konta, 656
 - root, 139
 - tworzenie, 650, 651
 - uprawnienia, 653
 - usuwanie konta, 655
 - uwierzytelnianie, 659
- Użytkownicy i grupy, 650, 652, 656
 - edytor konta użytkownika, 652
 - Grupy, 657
 - modyfikacja ustawień konta, 656
 - Profile użytkowników, 653
 - tworzenie grup, 656
 - Tworzenie nowej grupy, 657
 - tworzenie użytkowników, 651
 - uprawnienia, 653
 - Ustawienia grupy, 658
 - usuwanie konto, 655
 - zarządzanie grupami, 658
 - zarządzanie użytkownikami, 654

V

- vanilla, 126
- Variable Bit Rate, 408, 413
- VAX, 229
- VBR, 408, 413
- VCD, 413
- vcnviewer, 481
- Vegas Strip, 452
- vendor lock, 33
- Veritas Backup Exec, 85
- vfat, 743
- vfoldery, 272
- vi, 220, 221
 - autocmd, 231
 - cofanie zmian, 229
 - dostosowywanie edytora, 231
 - graficzne wersje, 232
 - iab, 231
 - kopiowanie tekstu, 227
 - map, 231
 - pliki konfiguracyjne, 231

- przemieszczanie kursora, 225
- set, 231
- status kopiowania, 227
- tryb linii poleceń, 223
- tryb normalny, 222
- tryb wstawiania, 222
- uruchamianie, 223
- usuwanie tekstu, 226
- wiele okien, 229
- wstawianie tekstu, 224, 227
- wycinanie tekstu, 227
- wyłączanie, 223
- wyszukiwanie tekstu, 228
- zarządzanie wieloma oknami, 230
- zastępowanie tekstu, 228
- zmiana tekstu, 226
- Video CD, 413
- vim, 221, 232
- vimrc, 231
- vino, 478
- Virtual Network Computing, 477
- visudo, 668
- VNC, 477
 - Chicken of the VNC, 480
 - klient, 478, 480, 481
 - Mac OS X, 480
 - OSX VNC, 480
 - pobieranie programu, 478
 - serwer, 478, 479, 480, 481
 - serwer vino, 478
 - TightVNC, 481
 - vcnviewer, 481
 - Windows, 481
- vnc4server, 479
- vcnviewer, 483, 484
- VPN, 474
- VT100, 545
- vtwm, 144

W

- warstwy, 385
- WAV, 413, 428
- Wavpack, 413
- Web log, 45
- WebDAV, 571, 573
- WEP, 776, 778
- weryfikacja kopii zapasowych, 688
- wget, 424
- whereis, 207
- whois, 763
- wiadomości e-mail, 807
- Wide SCSI-3, 732
- wide-dhcpv6-server, 838

- wideo, 408
- widok, 558
- wiele konfiguracji sieciowych, 760
- wiersz poleceń, 185
 - aliasy, 217
 - bash, 207
 - historia poleceń, 208
 - kontrola zadań, 213
 - kopiowanie plików, 200
 - poruszanie się po systemie plików, 200
 - potoki, 211
 - powłoka, 191
 - procesy, 213
 - przekierowanie danych, 211
 - przenoszenie plików, 200
 - stos, 200
 - usuwanie plików, 200
 - uzupełnianie nazw plików, 209
 - wykonywanie komend, 187
 - wyrażenia wieloznaczne, 210
 - wyszukiwanie poleceń, 206
 - zastosowanie, 186
 - zmiana katalogu, 200
 - zmiana nazwy plików, 200
 - zmiana uprawnień, 204
 - zmiennie środowiskowe, 215
 - znak zachęty, 185
- wildcards, 210
- Window Maker, 144
- WINDOW_MANAGER, 183
- Windows, 609
- Windows CE, 516
- Windows Media Audio, 413
- Windows Media Video, 413
- Windows Pocket CE PDA, 520
- Windows Pocket Edition, 516, 525
- Windows Pocket PC, 520
- Windows Terminal Server, 485, 489
- Windows Virtual FAT, 743
- Wine, 609
- WINGS, 633
- WINS, 901
- wireless-tools, 780
- wirusy, 33
- wlan0, 779
- WMA, 413
- WMF, 405
- WMV, 413
- workspace, 558
- WPA, 776
- WPA1, 776
- WPA2, 776
- Writer, 306, 314
 - dane użytkownika, 320
 - Edytuj styl akapitu, 324
 - Formatowanie, 318
 - importowanie plików, 329
 - instalacja, 315
 - kodowanie znaków, 330
 - konwersja dokumentów, 330
 - Konwerter dokumentów, 329
 - Kreator faksu, 321
 - kreatory, 320
 - linijki, 318
 - menu, 318
 - menu kontekstowe, 318
 - modyfikowanie stylów, 324
 - OpenDocument, 315
 - paski narzędzi, 318
 - Projekt strony, 321
 - Standardowy, 318
 - stosowanie szablonów, 326
 - style, 324
 - Style i formatowanie, 318, 324
 - szablony, 322, 326
 - tworzenie dokumentów, 320
 - tworzenie dokumentu na podstawie szablonu, 328
 - tworzenie stylu, 325
 - tworzenie szablonów, 326
 - uruchamianie, 317
 - ustawienia strony, 327
 - wpisywanie danych osobistych, 320
 - wygląd dokumentów, 324
 - zapisywanie szablonu, 328
- wskaźnik myszy, 160
- wsparcie systemu, 38
- wsparcie techniczne, 44
 - blogi, 45
 - Canonical Ltd., 50
 - dokumentacja, 48
 - fora internetowe, 45
 - IRC, 47
 - komercyjne wsparcie, 49
 - listy dyskusyjne, 47
 - społeczność Ubuntu, 45
 - The Ubuntu Marketplace, 50
- wtyczka, 558
- WVH, 900
- WWW, 791, 792
- wydania Ubuntu, 37
- wygazacz ekranu, 167
- wykonywanie kopii zapasowych, 695
- wykonywanie poleceń
 - wymagających specjalnych, 139
- Wyłącz komputer, 583, 598

- wymagania instalacyjne, 41
 - architektury systemowe, 42
 - sprzęt, 42
- wymagania sprzętowe, 42
- wypalanie, 413
- wrażenia wieloznaczne, 210
- wysyłanie poczty elektronicznej, 263
- wyszukiwanie
 - adres IP, 764
 - oprogramowanie, 636
 - pakiety, 620
 - polecenia, 206
- wyświetlanie plików DVI, 311

X

- X Window, 143
- X Window XDMCP, 489
- X.org, 144
- X11, 143
- Xchat, 47
- XDMCP, 489
- XDR, 882
- XEmacs, 236
- Xerox PARC, 751
- Xfce, 35, 145
- XFree86, 144
- XFS, 127
- xinetd, 824
- xmame, 459
- XML, 315
- XMMS, 413
- xninit, 181
- XSane Image Scanner, 722
- xterm, 196, 197, 198, 199
 - Fonts, 199
 - Main Options, 198
 - opcje, 198
 - VT Options, 198
- Xubuntu, 35, 146
- xubuntu-desktop, 94

Z

- zabezpieczenia systemu, 763
- Zakończ, 583, 598
- zakres komórek, 346
- zależności kompilacyjne, 627
- zamykanie systemu, 583, 598
- zapora sieciowa, 765
 - filtrowanie pakietów, 765, 766
 - FORWARD, 767
 - INPUT, 767

- instalacja, 767
- ipchains, 767
- iptables, 765, 766
- konfiguracja, 767, 769
- Linux, 766
- Lokkit, 767
- łańcuchy, 767
- NAT, 765, 766, 767
- netfilter, 765, 766
- OUTPUT, 767
- POSTROUTING, 767
- PREROUTING, 767
- reguły, 767
- Zarządca programów startowych, 594
 - modyfikacja sekwencji uruchamiania skryptów usług, 596
- zarządzanie
 - dostęp do plików, 670
 - grupy, 650, 658
 - karty PCMCIA, 744
 - serwer DHCP, 837, 841
 - użytkownicy, 649, 650, 654
- zarządzanie pakietami, 601, 603, 613, 616
 - apt-get, 603
 - aptitude, 603
 - dpkg, 603
 - dselect, 603
 - repozytoria, 604
 - synaptic, 603
 - update-manager, 603
- zdalne połączenia, 473
- zewnętrzne dyski, 727
- zgrywanie płyt
 - audio CD, 427
 - DVD, 442
- zintegrowane środowiska programistyczne, 551
 - Eclipse, 552, 554
 - Free-java-sdk, 553
 - KDevelop3, 552
 - Motor, 552
 - Pida, 552
 - Sbcl, 553
- zmiana
 - domyślna karta dźwiękowa, 417
 - katalog, 200
 - menedżer okien, 180
 - nazwa katalogu, 177
 - nazwa napędu SATA, 734
 - nazwa pliku, 177, 200
 - poziom głośności dźwięku, 416
 - tło pulpitu, 169
 - uprawnienia, 204
- zmienne środowiskowe, 215
- zmniejszanie obrazów, 394

znaczniki

HTML, 793

ID3, 531

LaTeX, 309

TeX, 307

znak zachęty, 185

zrzucanie, 412

zrzut ekranu, 391

zsh, 194

Ż

źródła dźwięku, 430

Zawartość CD-ROM-u:

Na dołączonej płycie znajdziesz rozruchową wersję systemu Ubuntu Linux opracowaną przez Canonical, Ltd., którą można uruchomić lub zainstalować na komputerze, oraz następujące aplikacje: AbiWord, Gaim, Gimp, Mozilla Firefox, Mozilla Thunderbird.

Materiały do książki można znaleźć pod adresem:

ftp://ftp.helion.pl/przyklady/ubunbi_ebook.zip Rozmiar pliku: 681 MB

PROGRAM PARTNERSKI

— GRUPY HELION —

- 
- A top-down view of four hands of different skin tones holding four interlocking puzzle pieces. The pieces are arranged in a 2x2 square. The top-left piece is red, the top-right is dark blue, the bottom-left is light blue, and the bottom-right is a slightly darker shade of light blue. The hands are positioned at the corners of the square, with fingers gripping the edges of the puzzle pieces.
1. ZAREJESTRUJ SIĘ
 2. PREZENTUJ KSIĄŻKI
 3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>

GRUPA
Helion 

KOMPLEKSOWO SZKOLIMY NOWOCZESNY BIZNES



IT



BIZNES



PROJEKTY



PROCESY

NASZE SZKOLENIA SĄ PROWADZONE
ZGODNIE Z METODĄ

BLENDED LEARNING

modelem kształcenia, który łączy tradycyjne szkolenie
z dostępem do nowoczesnych narzędzi - wideokursów,
e-booków i audiobooków

T: 609 850 372 E: SZKOLENIA@HELION.PL
WWW.HELIONSZKOLENIA.PL