Debian Reference

Osamu Aoki <osamu\#at\#debian.org>
Koordynator tłumaczenia: Bartosz Feński aka fEnIo <fenio@o2.pl>
'Autorzy' na 259 stronie

CVS, czwartek, 18 styczeń 2007, 11:53:26 UTC

Streszczenie

Debian Reference (`http://qref.sourceforge.net/`) z założenia stanowić ma **podręcznik poinstalacyjny** Debiana, będąc jednocześnie szerokim omówieniem tego systemu. Dokument opisuje wiele aspektów administracji, ilustrując je przykładowymi poleceniami. Zawarte w nim samouczki, porady i inne informacje dotyczą tematów takich jak: podstawowe założenia systemu Debian, wskazówki instalacyjne, zarządzanie pakietami, jądro Linuksa, dostrajanie systemu, konfiguracja bramy sieciowej, edytory tekstu, CVS, programowanie oraz GnuPG (dla zwykłych użytkowników).

Prawa autorskie

Copyright © 2001–2005 by Osamu Aoki <osamu#at#debian.org>.
Copyright (Rozdział 2) © 1996–2001 by Software in the Public Interest.

Ten dokument może zostać użyty zgodnie z warunkami GNU General Public License w wersji 2 lub wyższej. (`http://www.gnu.org/copyleft/gpl.html`)

Udziela się zgody na dystrybuowanie wiernych kopii tego dokumentu pod warunkiem opatrzenia wszystkich kopii informacją o prawach autorskich a także niniejszym pozwoleniem.

Udziela się zgody na tworzenie i dystrybucję zmodyfikowanych wersji tego dokumentu na warunkach takich, jak w przypadku wiernych kopii, pod warunkiem, że całe zmodyfikowane dzieło rozprowadzane jest na zasadach identycznych do zawartych w niniejszym pozwoleniu.

Udziela się zgody na kopiowanie i dystrybucję tłumaczeń tego dokumentu w innym języku, zgodnie z powyższymi warunkami dotyczącymi modyfikowanych wersji dokumentu, z tym wyjątkiem, iż tłumaczenie niniejszego pozwolenia musi zostać zatwierdzone przez Free Software Foundation lub pozostać w brzmieniu oryginalnym.

Powyższe tłumaczenie **nie** zostało zatwierdzone przez Free Software Foundation i w związku z tym **nie** jest w żadnym stopniu wiążące. Ma jedynie charakter informacyjny, a w przypadku redystrybucji lub modyfikacji dokumentu należy stosować się do oryginału zamieszczonego poniżej.

Permission is granted to make and distribute verbatim copies of this document provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this document under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this document into another language, under the above conditions for modified versions, except that this permission notice may be included in translations approved by the Free Software Foundation instead of in the original English.

Spis treści

Rozdział 1

Wstęp

Celem Debian Reference (`http://qref.sourceforge.net/`) jest dostarczenie szerokiego przeglądu systemu Debian w postaci **instrukcji poinstalacyjnej**. Docelowym czytelnikiem jest osoba, która jest skłonna do czytania skryptów powłoki systemowej (ang. shell). Oczekuję od czytelnika posiadania podstawowych umiejętności poruszania się w systemach uniksopodobnych.

Podjąłem świadomą decyzję **nie wyjaśniania** wszystkiego w szczegółach, jeśli są one do odnalezienia na stronach **podręcznika systemowego**, stronach systemu pomocy **info** lub w dokumentach **Jak to zrobić** (HOWTO). Zamiast szczegółowych wyjaśnień, starałem się raczej udzielić praktycznych porad dostarczając **dokładnych zestawów poleceń** w głównym tekście lub umieszczonych w katalogu `http://www.debian.org/doc/manuals/debian-reference/examples/` **plikach z przykładami**. Musisz zrozumieć zawartość przykładów zanim wykorzystasz je w prawdziwym systemie. Musisz również pamiętać, że w Twoim systemie mogą występować niewielkie różnice w tych poleceniach.

Wiele zamieszczonych informacji składa się z przypomnień lub odsyłaczy do pozycji z listy 'Zasoby' na 251 stronie. Jest to częściowo spowodowane tym, że ten dokument to **krótkie wprowadzenie**.

Krótkie i proste (KISS=Keep it short and simple) to moja główna zasada.

W sytuacji zagrożenia dla systemu, po pomoc udaj się natychmiast do rozdziału 'Polecenia zapewniające przetrwanie w Debianie' na 90 stronie.

1.1 Oficjalny dokument

Najnowszy oficjalny dokument znajduje się w archiwum Debiana, w pakiecie `debian-reference-en` oraz pod adresem `http://www.debian.org/doc/manuals/debian-reference/`.

Natomiast najnowszą wersję rozwojową możesz znaleźć pod adresem `http://qref.sourceforge.net/Debian/`. Strona domowa projektu to `http://qref.sourceforge.`

`net/`. Możesz tam znaleźć ten dokument w wersji tekstowej, HTML, PDF, SGML oraz PostScript.

1.2 Konwencje w dokumencie

Dokument o nazwie: „Debian Reference" zawiera informacje przedstawione w postaci krótkich poleceń powłoki systemowej Bash. Użyto następujących konwencji:

```
# polecenie wykonane z konta administratora
$ polecenie wykonane z konta użytkownika
```

Jeśli chcesz dowiedzieć się więcej o powłoce systemowej Bash, zerknij do 'Bash – interaktywna powłoka zgodna z **GNU**' na 228 stronie.

Odsyłam również do:
- **Podręcznika systemowego**: `bash(1)`.
- strony **GNU TEXINFO**: `info libc`.
- **książki**: *The C Programming Language*.
- **strony**: `http://www.debian.org/doc/manuals/debian-reference/`.
- dostępnego w systemie **pliku**: `/usr/share/doc/Debian/reference/`.

W dokumencie używane są skróty **LDP**, **DDP** i oznaczają:
- **LDP**: Linux Documentation Project (Projekt Dokumentacji Linuksa) (`http://www.tldp.org/`)
- **DDP**: Debian Documentation Project (Projekt Dokumentacji Debiana) (`http://www.debian.org/doc/`)

Przedstawiono tutaj tylko odwołania do LDP w postaci odsyłaczy URL, ale mogą być one pozyskane jako pakiet i zainstalowane do `/usr/share/doc/HOWTO/`. Zobacz także 'Zasoby' na 251 stronie.

1.3 Przykładowe skrypty

Przykładowe skrypty są dostępne w podkatalogach z przykładami (`http://www.debian.org/doc/manuals/debian-reference/examples/`) zainstalowanych w `/usr/share/doc/Debian/reference/`; początkowy znak dla ukrytych plików to : „.". W nazwach plików został on przerobiony na podkreślenie: „_". Dodatkowe rozszerzenia są dodawane w przypadku kilku alternatywnych wersji skryptów.

1.4 Podstawowe ustawienia

Jeśli system został zainstalowany z minimalnym zestawem pakietów, upewnij się, że wykonane zostało polecenie:

```
# aptitude install mc less ssh vim kernel-package \
                  manpages-dev doc-debian doc-linux-text \
                  debian-policy developers-reference maint-guide \
                  apt-howto harden-doc debian-reference \
                  libpam-doc glibc-doc samba-doc exim4-doc-html
```

1.5 Podstawowe informacje o dystrybucjach Debiana

Debian występuje w 3 „smakach”:

- stabilny (`stable`): Dobry do używania na serwerze produkcyjnym. Nudny w przypadku stacji roboczej (SR). Zobacz 'Dystrybucja `stabilna`' na 6 stronie.
- testowy (`testing`): Niezłe wyjście na stację roboczą. Zobacz 'Dystrybucja `testowa`' na 7 stronie.
- niestabilny (`unstable`): Nigdy nie wykorzystuj go „na ślepo”. Zobacz 'Dystrybucja `niestabilna`' na 7 stronie.

Zasubskrybuj przynajmniej kluczową listę `debian-devel-announce@lists.debian.org` informującą o uaktualnieniach Debiana.

W marcu 2002 roku te trzy wersje odpowiadały kodowym nazwom: Potato (jakość produkcyjna), Woody (testy, bardzo stabilna, gdyż nadchodziło kolejne wydanie) oraz Sid (testy alfa). W sierpniu 2002, już po wydaniu wersji Woody, odpowiednie wersje (stabilna, testowa, niestabilna) oznaczają: Woody (jakość produkcyjna), Sarge (testy, będzie w takim stanie jeszcze przez jakiś czas) oraz Sid (jak zwykle testy alfa). Gdy pakiety z wersji `niestabilnej` nie mają zgłoszonych żadnych krytycznych błędów (RC) po tygodniu, są automatycznie przenoszone do wersji `testowej`. Zobacz: 'Archiwa Debiana' na 5 stronie.

Są dwa sposoby na używanie najnowszych wersji oprogramowania.
- 'Podstawy zarządzania pakietami Debiana' na 84 stronie (głównie na stacje robocze)
- 'Przeniesienie pakietu do systemu `stabilnego`' na 98 stronie (głównie na serwery)

Używanie wersji `testowej` Debiana niesie ze sobą zagrożenie w postaci wolnego pojawiania się uaktualnień związanych z bezpieczeństwem. Zachowaj więc ostrożność.

Jeśli mieszasz smaki Debiana, używając części `testowej` lub `niestabilnej` w wydaniu `stabilnym`, możesz nieumyślnie zainstalować kluczowe pakiety w wersji, która może zawierać błędy. Również zachowaj ostrożność.

Używanie `testowych` lub `niestabilnych` wersji pociąga za sobą zwiększone ryzyko natknięcia się na istotny błąd. Nad ryzykiem tym można zapanować instalując dodatkowo wersję stabilną lub korzystając z programu „`chroot`” (opisano to w '`chroot`' na 134 stronie. Ostatni sposób umożliwia uruchamianie różnych smaków równocześnie, na różnych konsolach.

Po wyjaśnieniu kilku elementarnych sposobów dystrybucji Debiana w 'Debian – Podstawy' na sąsiedniej stronie, zaprezentuję kilka podstawowych informacji ułatwiających posługiwanie się najnowszym oprogramowaniem oraz wykorzystywanie wersji `testowej` i `niestabilnej`. Niecierpliwi powinni natychmiast zajrzeć do 'Polecenia zapewniające przetrwanie w Debianie' na 90 stronie. Miłego uaktualniania!

Rozdział 2

Debian – Podstawy

Ten rozdział dostarcza podstawowych informacji o systemie Debian dla osób nie będących deweloperami. Sprawdzone i oficjalne wiadomości należy czerpać z:

- Debian Policy Manual
- Debian Developer's Reference
- Debian New Maintainers' Guide

wymienionych w 'Zasoby' na 251 stronie.

Jeżeli szukasz mniej szczegółowych „jak-to-zrobić" („how-to"), przejdź bezpośrednio do 'Zarządzanie pakietami Debiana' na 83 stronie lub innych zasobów.

Niniejszy rozdział zawiera dokumenty wzięte z „Debian FAQ", poddane gruntownej reorganizacji dla ułatwienia startu początkującemu administratorowi systemu Debian.

2.1 Archiwa Debiana

2.1.1 Struktura katalogów

Pakiety z oprogramowaniem systemu Debian dostępne są poprzez FTP lub HTTP z katalogów znajdujących się na serwerze lustrzanym Debiana (`http://www.debian.org/mirror/`).

Na każdym z takich serwerów w katalogu `debian` można znaleźć następujące podkatalogi:

`dists/`: Ten katalog zawiera „dystrybucje", a niegdyś był to kanoniczny sposób uzyskiwania dostępu do aktualnych pakietów w stabilnych i niestabilnych wydaniach Debiana. Wciąż można tam znaleźć trochę starych pakietów, pliki `Contents-*.gz`, i pliki `Packages.gz`.

`pool/`: Nowe miejsce umieszczenia pakietów należących do wszystkich końcowych i testowych wydań Debiana.

`tools/`: DOSowe programy użytkowe, umożliwiające tworzenie dyskietek startowych, partycjonowanie dysków, kompresję i dekompresję plików oraz uruchomienie Linuksa.

`doc/`: Podstawowa dokumentacja Debiana, np. FAQ, instrukcje dotyczące zgłaszania błędów, itp.

`indices/`: Pliki Maintainers i override.

`project/`: przeważnie materiały dla deweloperów takie, jak:

- **`project/experimental/`:** Pakiety i narzędzia, które jeszcze nie zostały ukończone i znajdują się w fazie alfa. Nie należy ich używać ponieważ mogą się okazać niebezpieczne i szkodliwe nawet dla najbardziej doświadczonych.
- **`project/orphaned/`:** Pakiety osierocone przez dotychczasowych opiekunów i wykreślone z dystrybucji.

2.1.2 Dystrybucje Debiana

W katalogu `dists` normalnie znajdują się informacje o trzech dystrybucjach Debiana. Katalogi z nimi (i same dystrybucje) noszą nazwy `stable` (dystrybucja stabilna), `testing` (dystrybucja testowa) i `unstable` (dystrybucja niestabilna). Czasami też występuje tam dystrybucja `frozen` („zamrożona"). Katalog każdej dystrybucji stanowi symboliczne dowiązanie do rzeczywistego katalogu o odpowiednim kryptonimie w katalogu `dists`.

2.1.3 Dystrybucja `stabilna`

Dane pakietów należących do dystrybucji `stabilnej`, Debian Sarge (3.1r0), są zapisywane do katalogu `stable` (dowiązanie symboliczne do `Sarge/`):

- `stable/main/`: Ten katalog zawiera pakiety oficjalnie uznawane za najbardziej aktualne wydanie systemu Debian.

  Wszystkie z tych pakietów są zgodne z Wytycznymi Debiana dotyczącymi Wolnego Oprogramowania (`http://www.debian.org/social_contract#guidelines`) (dokument dostępny również jako `/usr/share/doc/debian/social-contract.txt` po zainstalowaniu pakietu `debian-doc`) i można ich swobodnie używać, a także rozpowszechniać.

- `stable/non-free/`: Ten katalog zawiera informacje o pakietach, których rozpowszechnianie zostało ograniczone przez wymagania stawiane dystrybutorowi, które mówią o zwróceniu szczególnej uwagi, na kwestie praw autorskich danego programu.

  Na przykład, licencje niektórych pakietów zabraniają komercyjnego rozpowszechniania. Inne znowuż mogą być redystrybuowane, ale stanowią shareware, a nie wolne oprogramowanie. Zanim włączy się którykolwiek z tych pakietów do jakiejś redystrybucji (np. na CD-ROMie), należy przestudiować jego licencję i prawdopodobnie przeprowadzić odpowiednie negocjacje.

- `stable/contrib/`: Ten katalog zawiera informacje o pakietach wolnych w rozumieniu DFSG (Debian Free Software Guidelines) i podlegających **swobodnemu rozpowszechnianiu**, ale w jakiś sposób zależnych od pakietu, który swobodnemu rozpowszechnianiu **nie podlega** i z tej przyczyny jest dostępny w sekcji non-free.

Wyżej wymienione katalogi zawierają informacje o pakietach. Same pakiety zaś mieszczą się w katalogu `pool` ('Katalog `pool`' na 9 stronie).

Aktualny stan błędów w dystrybucji `stabilnej` podaje strona WWW Problemy ze stabilną dystrybucją (`http://ftp-master.debian.org/testing/stable_probs.html`) (po angielsku) .

2.1.4 Dystrybucja `testowa`

Informacja o pakietach zawartych w dystrybucji `testowej`, czyli Debian Etch, jest zapisywana w katalogu `testing` (dowiązanie symboliczne do `Etch/`) po tym, jak przejdą testowanie wstępne w dystrybucji `niestabilnej`. Pakiety, których dotyczą informacje zapisane w katalogu `testing`, są umieszczane w katalogu `pool` ('Katalog `pool`' na 9 stronie). Oczywiście, w katalogu `testing` istnieją również podkatalogi `main`, `contrib` oraz `non-free`, pełniące takie same funkcje, jak ich odpowiedniki w `stable`.

Pakiety w dystrybucji `testowej` muszą działać na wszystkich architekturach, na których zostaną zbudowane i nie mogą posiadać zależności uniemożliwiających instalację; muszą mieć również mniej błędów o priorytecie release-critical (o znaczeniu krytycznym dla wydania), niż wersje znajdujące się w `unstable`. W ten sposób można mieć nadzieję, że dystrybucja `testowa` zawsze jest dystrybucją bliską wydania. Więcej informacji o mechanizmach testowania znajduje się na `http://www.debian.org/devel/testing` (po angielsku).

Najświeższe informacje o stanie dystrybucji `testowej` zawierają następujące strony WWW:

- update excuses (wymówki aktualizacyjne) (`http://ftp-master.debian.org/testing/update_excuses.html`)
- testing problems (problemy w dystrybucji testowej) (`http://ftp-master.debian.org/testing/testing_probs.html`)
- release-critical bugs (błędy o znaczeniu krytycznym dla wydania) (`http://bugs.debian.org/release-critical/`)
- base system bugs (błędy w systemie podstawowym) (`http://bugs.qa.debian.org/cgi-bin/base.cgi`)
- bugs in standard and task packages (błędy w pakietach kategorii standard i task) (`http://bugs.qa.debian.org/cgi-bin/standard.cgi`)
- other bugs and bug-squashing party notes (inne błędy i uwagi z sesji tępienia pluskiew) (`http://bugs.qa.debian.org/`)

2.1.5 Dystrybucja `niestabilna`

Informacje o pakietach wchodzących w skład dystrybucji `niestabilnej`, zawsze nazywanej „Sid", są zapisywane do katalogu `unstable` (dowiązanie symboliczne do `sid/`) po umieszczeniu tych pakietów w archiwum Debiana i pozostają tu, aż zostaną przeniesione do `testing/`. Odpowiadające wpisom pakiety umieszcza się w katalogu `pool` ('Katalog `pool`' na sąsiedniej stronie). Również istnieją tu podkatalogi `main`, `contrib` i `non-free`, spełniające te same zadania, co w katalogu `stable/`.

Dystrybucja `niestabilna` jest obrazem systemu w najnowszym stadium jego rozwoju. Nic nie stoi na przeszkodzie w używaniu i testowaniu tych pakietów, trzeba jednak uważać, bo mogą być jeszcze nie w pełni dopracowane. Zaletą korzystania z dystrybucji `niestabilnej` jest to, że jest się zawsze „na czasie" ze wszystkimi nowinkami w projekcie Debian, ale jeżeli coś nawali, to licz na siebie, a nie na pomoc.

Na stronie WWW Problemy z dystrybucją niestabilną (`http://ftp-master.debian.org/testing/unstable_probs.html`) (po angielsku) można się zapoznać z aktualnym stanem błędów w dystrybucji `niestabilnej`.

2.1.6 Dystrybucja `frozen` (`zamrożona`)

Gdy dystrybucja `testowa` osiągnie wystarczającą dojrzałość, zostaje zamrożona, co oznacza, że nie przyjmuje się do niej już żadnego nowego oprogramowania z wyjątkiem koniecznych poprawek (bugfixes). W katalogu `dists/` tworzy się nowy podkatalog dla dystrybucji testing, dowiązany do nowego kryptonimu. Dystrybucja zamrożona przechodzi przez kilkumiesięczny okres próbny składający się na przemian z aktualizacji i z okresów głębokiego zamrożenia nazywanych „cyklami testowymi".

Utrzymujemy rejestr błędów w dystrybucji zamrożonej, które mogą opóźnić wydanie pakietu lub doprowadzić do wstrzymania wydania całej dystrybucji. Gdy liczba błędów spadnie do maksymalnie akceptowalnej wartości, dystrybucja zamrożona staje się dystrybucją stabilną (stable), zostaje wydana, a dotychczasowe wydanie stabilne staje się przestarzałe (obsolete) i zostaje przeniesione do archiwum.

2.1.7 Kryptonimy dystrybucji Debiana

Nazwy rzeczywistych podkatalogów w katalogu `dists`, na przykład `Sarge` i `Etch`, są tylko „kryptonimami". Kiedy dystrybucja systemu Debian znajduje się jeszcze w etapie rozwojowym, nie posiada numeru wersji, a zamiast niego kryptonim. Kryptonimy zastosowano w celu ułatwienia tworzenia archiwów lustrzanych (mirroring) dystrybucji systemu Debian (gdyby rzeczywisty katalog taki, jak `unstable` nagle zmienił swoją nazwę na `stable/`, niejeden musiałby niepotrzebnie pobierać ponownie masę oprogramowania).

Aktualnie, `stable/` jest dowiązaniem symbolicznym do `Sarge/`, a `testing/` dowiązaniem do `Etch/`. Oznacza to, że `Sarge` jest aktualną dystrybucją stabilną, a `Etch` testową.

`unstable/` jest już na zawsze dowiązaniem symbolicznym do `sid/`, ponieważ Sid jest zawsze dystrybucją niestabilną.

2.1.8 Kryptonimy używane w przeszłości

Inne kryptonimy, których już wcześniej używano, to: „Buzz" dla wydania 1.1, „Rex" dla wydania 1.2, „Bo" dla wydania 1.3.x, „Hamm" dla wydania 2.0, „Slink" dla wydania 2.1, „Potato" dla wydania 2.2, „Woody" dla wydania 3.0 oraz „Sarge" dla wydania 3.1.

2.1.9 Źródło kryptonimów

Jak dotychczas wybierano postaci z filmu *Toy Story* produkcji Pixar.

- **Buzz** (Buzz Lightyear) był kosmonautą,
- **Rex** był tyranozaurem,
- **Bo** (Bo Peep) była dziewczynką, która opiekowała się owieczką,
- **Hamm** była świnką-skarbonką,
- **Slink** (Slinky Dog) był psem-zabawką,
- **Potato** był oczywiście Panem Ziemniakiem,
- **Woody** był kowbojem,
- **Sarge** był żołnierzem z zielonego plastiku,
- **Etch** (Etch-a-Sketch) był tablicą,
- **Sid** był chłopcem psującym zabawki.

2.1.10 Katalog `pool`

Dawniej pakiety przechowywano w podkatalogu katalogu `dists`, którego nazwa odpowiadała dystrybucji, w skład której wchodziły. Okazało się jednak, że wywoływało to różne problemy, jak np. duże obciążenie serwerów lustrzanych, gdy dokonywano większych zmian.

Obecnie pakiety znajdują się w dużej „puli" („pool"), której struktura jest utworzona na podstawie nazw pakietów źródłowych. Dla ułatwienia zarządzania czymś takim, pula - pool jest podzielona według sekcji (`main`, `contrib` i `non-free`) i według pierwszej litery nazwy pakietu źródłowego. Katalogi te zawierają pewną liczbę plików: pakiety binarne dla każdej architektury (platformy sprzętowej) oraz pakiety źródłowe, z których te pierwsze zostały wygenerowane.

Miejsce, gdzie znajduje się jakiś pakiet, można określić wykonując polecenie `apt-cache showsrc` *`nazwa_pakietu`* i znajdując w jego wyjściu wiersz zaczynający się od „Directory:". Na przykład pakiety serwera http `apache` znajdują się w `pool/main/a/apache/`. Pakietów `lib*` jest bardzo dużo, więc są traktowane szczególnie: na przykład pakiety `libpaper` są przechowywane w katalogu `pool/main/libp/libpaper/`.

Podkatalogi katalogu `dists` są w dalszym ciągu używane do przechowywania plików indeksowych używanych przez programy w rodzaju `apt`. Również, w czasie pisania niniejszego dokumentu starsze dystrybucje nie były przestawione na używanie katalogu pool, więc można zobaczyć takie nazwy dystrybucji, jak `potato` czy `woody` w wierszach zaczynających się od „Directory:" (przytoczone powyżej polecenie apt-cache).

Nie jest to powód do zmartwień, ponieważ nowy `apt` i prawdopodobnie również starszy `dpkg-ftp` (zobacz 'Sposoby aktualizacji systemu Debian' na 20 stronie) radzą sobie z taką strukturą bez problemów. Więcej informacji można znaleźć w RFC: implementation of package pools (`http://lists.debian.org/debian-devel-announce/2000/debian-devel-announce-200010/msg00007.html`) (po angielsku).

2.1.11 Nota historyczna o dystrybucji Sid

Kiedy dzisiejszy Sid jeszcze nie istniał, organizacja sieciowych archiwów Debiana miała jedną dużą wadę: kiedy dokładano nową architekturę do bieżącej dystrybucji `unstable`, pakiety zrobione dla niej mogły być wydane dopiero wtedy, gdy ta dystrybucja stawała się nową dystrybucją `stable`. Dla wielu architektur nie dochodziło do tego i trzeba było przenosić odpowiadające im katalogi, gdy dochodziło do wydania dystrybucji. Było to niepraktyczne, ponieważ przenoszenie katalogów silnie obciążało łącza.

Administratorzy archiwów sieciowych przez kilka lat obchodzili ten problem, umieszczając binaria dla architektur jeszcze nie wydanych w specjalnym katalogu o nazwie `sid`. Dla architektur jeszcze nie wydanych, tworzono w chwili wydania dowiązanie z aktualnego katalogu `stable` do `sid` i od tej pory tworzono je w drzewie `unstable`, jak zwykle. Takie rozwiązanie było trochę mylące dla użytkowników.

Z nadejściem katalogu „pool" (zobacz 'Katalog `pool`' na poprzedniej stronie) w trakcie powstawania dystrybucji Woody, zaczęto zapisywać pakiety binarne w lokalizacji kanonicznej w tymże katalogu, niezależnie od dystrybucji, więc wydanie dystrybucji przestało być związane z poddawaniem serwerów lustrzanych dużym obciążeniom (natomiast mamy do czynienia z dość sporymi, rozłożonymi w czasie obciążeniami w trakcie całego procesu rozwijania dystrybucji).

2.1.12 Pakiety umieszczone w `incoming/`

Pakiety umieszczane w archiwum trafiają najpierw do `http://incoming.debian.org/`, po sprawdzeniu autentyczności pochodzenia od jednego z deweloperów (w wypadku tzw. Non-Maintainer Upload – NMU – pakiety trafiają do podkatalogu `DELAYED`). Raz dziennie pakiety przenosi się z `incoming/` do `unstable/`.

W nagłych wypadkach można instalować pakiety z `incoming/`, zanim jeszcze trafią do `unstable`.

2.1.13 Odzyskiwanie starszego pakietu

Podczas gdy najnowsze dystrybucje Debiana trzyma się w podkatalogach katalogu `debian`, na każdym z serwerów wymienionych na Stronie serwerów lustrzanych Debiana (`http://www.debian.org/mirror/`), archiwa starszych dystrybucji (np. Slink) znajdują się na `http://archive.debian.org/` lub w podkatalogach katalogu `debian-archive` na każdym serwerze lustrzanym Debiana.

Starsze pakiety z `testing` i `unstable` można znaleźć na `http://snapshot.debian.net/`.

2.1.14 Podział na architektury

W obrębie każdego z głównych drzew katalogów (`dists/stable/main`, `dists/stable/contrib`, `dists/stable/non-free`, `dists/unstable/main/`, itd.), informacja o pakietach binarnych znajduje się w podkatalogach o nazwach wskazujących na platformę sprzętową (architekturę), dla jakiej zostały skompilowane.

- `binary-all/`, dla pakietów niezależnych od architektury (dotyczy to np. skryptów Perla lub czystej dokumentacji).
- `binary-`*`platform`*`/`, dla pakietów przeznaczonych dla konkretnej architektury.

Warto zauważyć, że pakiety binarne dla dystrybucji `testing` i `unstable` nie są już przechowywane w tych katalogach, ale w katalogu `pool`. Pliki indeksowe (`Packages` i `Packages.gz`) jednak, dla zachowania kompatybilności z wcześniejszymi rozwiązaniami, w dalszym ciągu przebywają tam, gdzie były.

Aby poznać faktyczny zestaw wspieranych platform sprzętowych, należy zapoznać się z Informacjami Wydawniczymi dla danej dystrybucji. Można je odnaleźć na stronach zajmujących się Uwagami Wydawniczymi dla stable (`http://www.debian.org/releases/stable/releasenotes`) i testing (`http://www.debian.org/releases/testing/releasenotes`).

2.1.15 Kod źródłowy

Debian posiada również kod źródłowy każdego ze swoich składników. Co więcej, warunki licencji większości programów w systemie zawierają **wymóg** dystrybucji kodu źródłowego wraz z programem, lub przynajmniej zadeklarowania gotowości dostarczenia kodu źródłowego wraz z programem.

Normalnie kod źródłowy jest rozpowszechniany za pośrednictwem katalogów `source`, istniejących równolegle do wszystkich katalogów charakterystycznych dla poszczególnych architektur, a obecnie w katalogu `pool` directory (zobacz 'Katalog `pool`' na 9 stronie). Aby pobrać z archiwum kod źródłowy bez konieczności zaznajamiania się ze strukturą archiwum Debiana należy wykonać polecenie podobne do tego: `apt-get source `*`mojanazwapakietu`*.

Niektóre pakiety, na przykład `pine`, są dostępne wyłącznie w postaci źródeł wskutek ograniczeń licencyjnych. (Niedawno pojawił się pakiet `pine-tracker` ułatwiający instalację Pine). Procedury opisane w 'Przeniesienie pakietu do systemu `stabilnego`' na 98 stronie i 'Pakietowanie' na 245 stronie opisują sposoby tworzenia pakietów samemu.

Dla pakietów z katalogów `contrib` i `non-free`, które oficjalnie nie stanowią części systemu Debian, kod źródłowy może być niedostępny.

2.2 System zarządzania pakietami w Debianie

2.2.1 Przegląd pakietów Debiana

W ogólności pakiety zawierają wszystkie pliki niezbędne do zaimplementowania zestawu odpowiednich poleceń lub właściwości. Są dwa typy pakietów Debiana:

- **Pakiety binarne**, zawierające pliki wykonywalne, pliki konfiguracyjne, strony podręcznika systemowego `man` lub `info`, informacje o prawach autorskich i pozostałą dokumentację. Są one rozpowszechniane w charakterystycznym dla Debiana formacie archiwum (zobacz 'Format pakietów Debiana' na sąsiedniej stronie); zwykle można je odróżnić od innych plików po tym, że ich nazwa kończy się na `.deb`. [1] Pakiety binarne można rozpakowywać z pomocą programu `dpkg`; ze szczegółami można się zapoznać czytając stronę podręcznika systemowego poświęconą dpkg.
- **Pakiety źródłowe**, składające się z pliku `.dsc` opisującego pakiet źródłowy (włącznie z nazwami plików składowych pakietu), pliku `.orig.tar.gz` zawierającego oryginalny, niezmodyfikowany kod źródłowy spakowany programem tar i skompresowany programem gzip oraz zwykle pliku `.diff.gz` zawierającego charakterystyczne dla Debiana zmiany w stosunku do oryginalnego źródła. Do pakowania i rozpakowywania archiwów źródłowych Debiana używa się programu użytkowego `dpkg-source`; szczegóły są dostępne po zapoznaniu się z poświęconą mu stroną podręcznika systemowego `man`.

Instalacja oprogramowania przez system pakietów posługuje się pojęciem „zależności" („dependencies"), troskliwie określonych przez opiekunów poszczególnych pakietów. Te zależności są wyszczególnione w pliku `control` wchodzącym w skład każdego pakietu. Na przykład, pakiet zawierający kompilator GNU C (`gcc`) jest zależny od zawierającego konsolidator (linker) i asembler pakietu `binutils`. Jeżeli użytkownik usiłuje zainstalować `gcc` nie zainstalowawszy uprzednio `binutils`, system zarządzania pakietami (dpkg) drukuje komunikat mówiący, że trzeba zainstalować `binutils`, a następnie zatrzymuje instalację `gcc` (uparty użytkownik może jednak zmienić to zachowanie; zobacz `dpkg(8)`). Z dodatkowymi szczegółami można zapoznać się w 'Zależności między pakietami' na 16 stronie.

Zawartych w Debianie narzędzi obsługujących pakiety można używać do:

- manipulowania i zarządzania pakietami lub ich częściami składowymi,

---

[1] Projekt `debian-installer` wprowadza pakiety o nazwach kończących się na `.udeb`. W skrócie, jest to format micro-`.deb`, który niekoniecznie zgadza się dokładnie z polityką Debiana, nie posiada zawartości takiej jak dokumentacja i jest wykorzystywany wyłącznie przez `debian-installer`, nowe instalator Debiana, tworzony dla wydania Sarge. Format plików `.deb` oraz `.udeb` jest identyczny. Program `udpkg` używany do obsługi pakietów `.udeb` posiada ograniczone możliwości w stosunku do standardowego `dpkg` i obsługuje mniej zależności pomiędzy pakietami. Różnica w nazwie wynika z tego, że osoby utrzymujące pakiety Debiana w formacie `.deb` nie były by zadowolone z pojawienia się pakietów nie spełniających założeń. Wybrano więc inną nazwę w celu podkreślenia tych różnic oraz zabezpieczenia przed sytuacją, w której użytkownicy mohą próbować instalować takie pakiety w działającym systemie. Pakiety `.udeb` są używane na startowym dysku umieszczonym w pamięci wyłącznie podczas podstawowej instalacji, w celu utworzenia bardzo okrojonego systemu.

- ułatwienia użytkownikowi podziału na części pakietów, które trzeba umieścić na nośnikach o ograniczonej wielkości (np. na dyskietkach),
- pomocy deweloperom w konstruowaniu archiwów pakietów, oraz
- pomocy użytkownikom w instalacji pakietów przechowywanych na sieciowych serwerach z archiwami Debiana.

2.2.2 Format pakietów Debiana

„Pakiet” Debiana (zwany też archiwum Debiana - nie mylić z umieszczoną na serwerze całą dystrybucją!) zawiera pliki wykonywalne, biblioteki oraz dokumentację związaną z konkretnym programem lub zestawem w jakiś sposób powiązanych ze sobą programów. Z reguły nazwa pakietu Debiana kończy się sufiksem `.deb`.

Budowę wewnętrzną pakietów binarnych w tym formacie opisuje podręcznik systemowy `deb(5)`. Z uwagi na możliwość zachodzenia zmian w specyfikacji formatu (z jednego wydania systemu Debian na kolejne jego wydanie), przy manipulacji plikami `.deb` należy zawsze korzystać z `dpkg-deb(1)`.

Wszystkie pakiety Debiana (Sarge i wcześniejsze dystrybucje), można obrabiać używając standardowych poleceń systemu Unix: `ar`, i `tar`, nawet, gdy polecenia `dpkg` są niedostępne.

2.2.3 Zasady nadawania nazw pakietom Debiana

Nazwy plików zawierających pakiety w Debianie przestrzegają następujących zasad:

```
foo_NumerWersji-NumerRewizjiDebiana.deb
```

gdzie *foo* stanowi nazwę pakietu. Dla sprawdzenia: mając dany plik `.deb` można określić nazwę zawartego w nim pakietu w jeden z następujących sposobów:

- przejrzeć plik „Packages” w katalogu, w którym jest przechowywany na jednym z serwerów Debiana. Plik ten zawiera sekcje opisujące każdy pakiet. Pierwszy wiersz każdej sekcji zawiera oficjalną nazwę pakietu (po „Package: ”).
- użyć polecenia `dpkg --info foo_VVV-RRR.deb` (gdzie *VVV* i *RRR* są odpowiednio wersją i rewizją pakietu *foo*). Polecenie wyświetla wśród innych rzeczy nazwę pakietu zawartego w pliku poddanym sprawdzeniu.

Składnik *VVV* jest numerem wersji, nadanym przez programistę zajmującego się danym programem/pakietem poza systemem Debian. Numery wersji nie są ustalane przez żadne normy, dlatego też mogą one mieć najrozmaitsze formaty, jak np. „19990513” czy „1.3.8pre1”.

Składnik *RRR* jest numerem rewizji w Debianie i jest nadawany przez opiekuna danego pakietu lub przez indywidualnego użytkownika, jeśli ten zechce zbudować pakiet samodzielnie.

Numer ten odpowiada poziomowi rewizji pakietu w systemie; z tego względu nowy poziom rewizji zwykle oznacza zmiany w plikach opisujących budowę i instalację pakietu: `debian/rules` (Debian makefile), `debian/control` (Debian control file), skrypcie instalacyjnym i deinstalacyjnym `debian/p*`, lub plikach konfiguracyjnych związanych z pakietem.

2.2.4 Ochrona lokalnych plików konfiguracyjnych

Mechanizm „conffiles" istniejący w Debianie pozwala na otoczenie ochroną plików konfiguracyjnych w systemie. Pliki te, zazwyczaj umieszczone w katalogu `/etc`, są wymienione w plikach o nazwach kończących się na `conffiles` wchodzących w skład systemu pakietów. Mechanizm ten gwarantuje, że systemowe pliki konfiguracyjne nie będą nadpisywane podczas instalacji nowej wersji pakietu.

Jeśli możliwa jest konfiguracja systemu bez dokonywania zmian w plikach należących do różnych pakietów, dobrym pomysłem jest powstrzymanie się od ich modyfikacji nawet jeśli są to pliki „conffiles". Ułatwi to i przyspieszy instalację nowych wersji pakietów.

Aby dowiedzieć się, jakie dokładnie pliki zostaną zachowane podczas instalacji nowej wersji (upgrade'u) pakietu, należy uruchomić polecenie:

```
dpkg --status pakiet
```

i szukać „Conffiles:".

Szczegółów na temat zawartości plików `conffiles` dostarcza dokument Debian Policy Manual w rozdziale 11.7 (zobacz 'Zasoby' na 251 stronie).

2.2.5 Skrypty instalacyjne i deinstalacyjne

Są to skrypty automatycznie uruchamiane przed instalacją i po instalacji pakietu. Wraz z plikiem o nazwie `control` wchodzą w skład sekcji „control" każdego pakietu w Debianie.

Są to:

preinst Ten skrypt jest wykonywany zanim pakiet zostanie rozpakowany z pliku `.deb`. Wiele skryptów „preinst" zatrzymuje działanie usług świadczonych przez pakiety, których nową wersję właśnie instalujemy, aż do czasu zakończenia procesu instalacji lub aktualizacji (tzn. do chwili bezbłędnego wykonania skryptu „postinst").

postinst Ten skrypt z zasady wykonuje wszelkie operacje konfiguracyjne wymagane do prawidłowej pracy pakietu po jego rozpakowaniu z pliku `.deb`. Skrypty „postinst" często wymagają wprowadzenia pewnych informacji przez użytkownika i/lub ostrzegają go, że w wypadku akceptacji wartości domyślnych powinien pamiętać o późniejszej rekonfiguracji pakietu w miarę potrzeb. Wiele ze skryptów „postinst" wykonuje następnie wszelkie polecenia konieczne do uruchomienia lub ponownego uruchomienia danej usługi po instalacji lub aktualizacji pakietu.

prerm Z reguły, skrypt ten zatrzymuje wszystkie demony związane z pakietem, a jest wykonywany przed usunięciem plików, które zawierał ten pakiet.

postrm Ten skrypt z reguły modyfikuje dowiązania lub inne pliki związane z pakietem i/lub usuwa pliki przezeń utworzone (zobacz również 'Pakiety wirtualne' na następnej stronie).

Aktualnie wszystkie pliki sterujące pakietu (te z sekcji „control") można znaleźć w katalogu `/var/lib/dpkg/info`. Pliki związane z pakietem `foo` mają nazwy zaczynające się od „foo" i kończące na „preinst", „postinst", itd. Plik `foo.list` w tym katalogu wymienia wszystkie pliki zainstalowane podczas instalacji pakietu `foo`. (Położenie tych plików jest wewnętrzną sprawą programu `dpkg` i może być zmienione).

2.2.6 Priorytety pakietów

Każdemu pakietowi w Debianie opiekunowie dystrybucji przypisali pewien **priorytet**, mający spełniać pomocnicze funkcje względem systemu zarządzania pakietami. Istnieją następujące rodzaje priorytetów:

- Pakiety **Wymagane** (**Required**), są konieczne do właściwego działania systemu.

  W ich skład wchodzą wszystkie narzędzia konieczne do naprawy uszkodzeń systemu. Nie wolno ich usuwać, bo system może ulec totalnej zapaści, która uniemożliwi nawet użycie `dpkg` do próby jego odtworzenia. Systemy zawierające wyłacznie pakiety o priorytecie Required najprawdopodobniej nie nadają się do wielu rzeczy, ale posiadają funkcjonalność wystarczającą do ich uruchomienia i instalacji dodatkowego oprogramowania.

- Pakiety **Ważne** (**Important**), powinny być zainstalowane na każdym systemie uniksopodobnym.

  Inne pakiety, bez których system nie będzie dobrze działał, lub nie będzie użyteczny, będą posiadać również ten priorytet. **Nie należą** do nich Emacs, X11, TeX czy inne duże aplikacje. Pakiety o priorytecie **Important** tworzą zaledwie gołą infrastrukturę.

- Pakiety **Standardowe** (**Standard**). Stanowią standard na każdym systemie linuksowym, tworząc nieduży, ale niezbyt ograniczony system pracujący w trybie tekstowym.

  Pakiety o tym priorytecie zainstalują się domyślnie, jeżeli użytkownik nie wybierze nic ponadto. Grupa **Standard** zawiera niewiele dużych aplikacji, ale zawiera Emacs (jest on bardziej elementem infrastruktury, niż aplikacją) i rozsądny wybór z TeX-a i LaTeX-a (to, czego można używać bez X).

- Pakiety **Opcjonalne** (**Optional**) są pakietami, których instalacja może się okazać rozsądnym wyborem nawet wtedy, gdy się nie zna ich na wskroś i kiedy nie ma się jakichś szczególnych wymagań.

  W skład tej grupy wchodzą X11, pełna dystrybucja TeX-a i mnóstwo aplikacji.

- Pakiety **Ekstra** (**Extra**) są w konflikcie z innymi pakietami o wyższym priorytecie, mają małą użyteczność dla nieobeznanych z nimi, albo mają szczególne wymagania, które nie pozwalają im wejść do grupy pakietów Opcjonalnych.

Zwróć uwagę na różnice pomiędzy „Priority: required", „Section: base" i „Essential: yes" w opisie pakietu. „Section: base" oznacza, że ten pakiet jest instalowany w nowym systemie przed czymkolwiek innym. Większość pakietów z „Section: base" posiada „"Priority: required„ lub przynajmniej "Priority: important„ i wiele z nich jest oznaczona jako "Essential: yes„. "Essential: yes„ oznacza, że aby taki pakiet usunąć należy użyć dodatkowych parametrów do polecenia tak by wymusić to usuwanie korzystając z `dpkg`. Na przykład `libc6`, `mawk` i `makedev` posiadają w ustawieniach "Priority: required„ i "Section: base„ ale nie "Essential: yes„.

2.2.7 Pakiety wirtualne

Pakiet wirtualny nosi nazwę odnoszącą się do dowolnego z grupy pakietów, posiadających zbliżoną funkcjonalność. Na przykład zarówno `tin` jak i `trn` są klientami grup dyskusyjnych i każdy z nich powinien spełniać wymagania innego programu, wymagającego czytnika news do działania. O obu z nich mówi się więc, że dostarczają „wirtualny pakiet" o nazwie `news-reader`.

Podobnie, `exim` i `sendmail` są programami transportującymi pocztę (mail transport agent). Mówi się więc o nich, że dostarczają wirtualny pakiet o nazwie `mail-transport-agent`. Jeżeli jeden z nich zostanie zainstalowany, każdy program zależący od instalacji mail transport agent zostanie usatysfakcjonowany dzięki istnieniu tego pakietu wirtualnego.

Debian posiada również mechanizm umożliwiający administratorowi wyznaczenie preferowanego pakietu w razie, gdy kilka zainstalowanych pakietów dostarcza ten sam „pakiet wirtualny". Odpowiednim poleceniem jest `update-alternatives`, które opisano w 'Alternatywne polecenia' na 101 stronie.

2.2.8 Zależności między pakietami

W systemie zarządzania pakietami w Debianie istnieje kategoria „zależności" między pakietami zaprojektowanych tak, aby w prosty sposób, (za pomocą pojedynczej flagi) ukazać poziom niezależności funkcjonowania programu A od istnienia w danym systemie programu B:

- Pakiet A **zależy** (**depends** on) od Pakietu B, jeżeli B musi być bezwarunkowo zainstalowany, aby można było uruchomić A. W niektórych przypadkach A nie tylko zależy od B, ale od jego określonej wersji. W takich przypadkach zależność wersji jest dolną granicą, co należy rozumieć, że A zależy od dowolnej wersji B nowszej od wersji podanej.
- Pakiet A **zaleca** (**recommends**) Pakiet B, jeżeli opiekun pakietu jest zdania, że większość użytkowników nie zechciałaby skorzystać z A bez posiadania możliwości oferowanych przez B.

- Pakiet A **sugeruje** (**suggests**) Pakiet B, jeżeli B zawiera pliki mające związek z funkcjonalnością A (zwykle zwiększające ją).

- Package A **jest w konflikcie** (**conflicts**) z Pakietem B wtedy, kiedy A nie będzie działać, jeżeli B jest zainstalowany w systemie. Konflikty zachodzą przeważnie wtedy, gdy A zawiera pliki, które mają pod jakimś względem przewagę nad plikami należącymi do B. Stan „conflicts" często występuje wspólnie z „replaces".

- Pakiet A **zastępuje** (**replaces**) Pakiet B wtedy, gdy pliki zainstalowane przez B ulegają usunięciu i (w niektórych wypadkach) nadpisaniu przez pliki należące do A.

- Pakiet A **dostarcza** (**provides**) Pakiet B wtedy, gdy wszystkie pliki i cała funkcjonalność pakietu B zawierają się w A. Daje to użytkownikom mniejszych dysków możliwość zainstalowania tylko tej części pakietu A, której naprawdę potrzebują.

Bardziej szczegółowe informacje o wykorzystaniu każdego z powyższych pojęć zawierają dokumenty *Packaging Manual* i *Policy Manual*.

Dobrze wiedzieć, że `dselect` umożliwia bardziej subtelną kontrolę nad pakietami **zalecanymi** i **sugerowanymi**, niż `apt-get`, który po prostu pobiera wszystkie pakiety oznaczone **zależy** a zostawia w spokoju **rekomendowane** i **sugerowane**. W nowoczesnym wydaniu obydwa programy są „nakładką" na APT.

2.2.9 Znaczenie „pre-depends"

Pojęcie „zależność wstępna" („pre-depend") stanowi szczególny rodzaj zależności. W wypadku zwyczajnego pakietu `dpkg` rozpakuje plik pakietu (plik `.deb`) niezależnie od tego, czy pliki, od których dany pakiet zależy, są już w systemie. Samo rozpakowanie polega na tym, że `dpkg` wyciąga z pliku archiwum pliki przeznaczone do instalacji w systemie użytkownika i umieszcza je we właściwych miejscach w tym systemie. Jeżeli pakiet właśnie instalowany **zależy** od obecności w systemie użytkownika jakichś innych pakietów, `dpkg` odmówi dokończenia instalacji (co przejawiłoby się wykonaniem akcji „configure") aż do czasu zainstalowania tych pakietów.

Istnieje jednak trochę pakietów, których `dpkg` nie będzie chciał nawet rozpakować, dopóki nie zostaną spełnione pewne zależności. Mówimy, że takie pakiety „pre-depend" (zależą wstępnie) od obecności w systemie jakichś innych pakietów. Mechanizm ten został wprowadzony w projekcie Debian w celu ułatwienia bezpiecznej instalacji nowych wersji pakietów w dobie przejścia z formatu `a.out` na `ELF`, gdy nawet **kolejność** rozpakowywania pakietów była krytyczna. Są też inne sytuacje związane z dużymi aktualizacjami systemu, w których omawiany mechanizm okazuje się użyteczny, np. dla pakietów o priorytecie „Wymagane" zależnych od libc.

Bardziej szczegółowe informacje można znaleźć w dokumencie Packaging Manual.

2.2.10 Status pakietu

Status pakietu może przybierać formę: „nieznany” („unknown”), „zainstalować” („install”), „usunąć częściowo” („remove”), „usunąć całkowicie” („purge”) lub „zatrzymać” („hold”). Te znaczniki pokazują, co użytkownik chciałby uczynić z danym pakietem (przez dokonanie wyboru w sekcji „Select” programu `dselect` lub przez bezpośrednie wywołanie `dpkg`).

Ich znaczenie jest następujące:

- **unknown** - użytkownik nigdy nie zdradził, czy w ogóle chce tego pakietu
- **install** - użytkownik chce, aby pakiet (lub jego nowsza wersja) został zainstalowany.
- **remove** - użytkownik chce, aby pakiet został usunięty, ale nie chce usuwać jego plików konfiguracyjnych.
- **purge** - użytkownik chce całkowitego usunięcia pakietu, z plikami konfiguracyjnymi włącznie.
- **hold** - użytkownik nie chce, aby pakiet był ruszany, tzn. chce zachować bieżącą wersję pakietu w bieżącym stanie, wszystko jedno jakim.

2.2.11 Zapobieganie instalacji nowych wersji pakietów

Są dwa mechanizmy chroniące pakiety przed instalacją nowej wersji, jeden z nich oparty jest na `dpkg`, a drugi od dystrybucji Woody, na `APT`.

Wykorzystując ten pierwszy z nich, należy najpierw wyeksportować listę zawierającą nazwy zainstalowanych pakietów i status każdego z nich:

```
dpkg --get-selections \* > selections.txt
```

Następnie należy dokonać edycji powstałego pliku *`selections.txt`*, zmieniając odpowiednio wiersz zawierający nazwę pakietu, który chcemy chronić (np. `libc6`), z czegoś takiego:

```
libc6                      install
```

na coś takiego:

```
libc6                      hold
```

Plik należy zapisać na dysku i wprowadzić go do bazy danych `dpkg` wykonując polecenie:

```
dpkg --set-selections < selections.txt
```

Jeżeli dobrze znamy nazwę pakietu, który chcemy chronić, możemy po prostu wykonać:

```
echo libc6 hold | dpkg --set-selections
```

Dla każdego potraktowanego w ten sposób pakietu oznacza to, że będzie on chroniony przed zainstalowaniem swojej nowszej wersji.

Taki sam wynik można uzyskać korzystając z programu `dselect`. Wystarczy w tym wypadku wejść do sekcji [S]elect, odnaleźć pakiet, który chcemy chronić przed zmianą i nacisnąć klawisz „=" lub „H". Zmiana statusu będzie obowiązywać od chwili opuszczenia sekcji [S]elect.

Wiodący w dystrybucji Woody, system APT posiada nowy, alternatywny mechanizm ochrony pakietów podczas ich pobierania z repozytorium, oparty na `Pin-Priority`. Więcej szczegółów na ten temat dostarczy `apt_preferences(5)`, wraz z `http://www.debian.org/doc/manuals/apt-howto/` lub pakietem `apt-howto`; 'Przegląd pliku `/etc/apt/preferences`' na 89 stronie również zawiera krótkie wyjaśnienie.

2.2.12 Pakiety źródłowe

Pakiety źródłowe są przechowywane w katalogu o nazwie `source`, można je pobierać klasycznymi metodami, można też użyć polecenia

```
apt-get source foo
```

do ich pobrania (o konfiguracji programu APT do tej operacji szerzej traktuje podręcznik systemowy: `apt-get(8)`).

2.2.13 Tworzenie pakietów binarnych ze źródłowych

Aby skompilować pliki źródłowe dla pakietu o nazwie *`foo`*, będzie trzeba użyć wszystkich, z następujących plików: *`foo_*`*`.dsc`, *`foo_*`*`.tar.gz` oraz *`foo_*.diff`*`.gz`. (ciekawa rzecz: dla rodzimych pakietów Debiana nie ma plików `.diff.gz`).

Jeśli pobraliśmy już wszystkie potrzebne pliki i mamy zainstalowany pakiet `dpkg-dev`, polecenie

```
$ dpkg-source -x foo_version-revision.dsc
```

rozpakuje pakiet do katalogu o nazwie *`foo-version`*.

Aby utworzyć pakiet binarny, należy wydać następujące polecenie (tutaj podane w wersji dla zwykłego użytkownika):

```
$ cd foo-version
$ su -c "apt-get update ; apt-get install fakeroot"
$ dpkg-buildpackage -rfakeroot -us -uc
```

Następnie,

```
$ su -c "dpkg -i ../foo_version-revision_arch.deb"
```

aby zainstalować świeżo utworzony pakiet. Więcej informacji - 'Przeniesienie pakietu do systemu `stabilnego`' na 98 stronie.

2.2.14 Tworzenie nowych pakietów Debiana

Szczegółowych informacji dotyczących tworzenia nowych pakietów dostarczy lektura *New Maintainers' Guide*, dostępnego jako pakiet `maint-guide`, lub pod adresem `http://www.debian.org/doc/manuals/maint-guide/`.

2.3 Aktualizacja systemu Debian

Jednym ze strategicznych celów Debiana jest dostarczenie spójnych metod umożliwiających bezpieczną instalację nowego oprogramowania; dokładamy starań, aby proces instalacji nowego wydania na poprzednim przebiegał jak najbardziej gładko. Pakiety będą informować użytkownika o ważnych zdarzeniach podczas instalacji, a często będą proponować rozwiązanie problemu.

Należy przeczytać Uwagi Wydawnicze (Release Notes), dokument opisujący szczegóły instalacji nowej wersji (upgrade'u) poszczególnych dystrybucji, dostarczany na każdej płytce z systemem Debian i dostępny na stronach WWW pod adresem `http://www.debian.org/releases/stable/releasenotes` lub `http://www.debian.org/releases/testing/releasenotes`.

Praktyczny poradnik traktujący o instalacji nowej dystrybucji na poprzednią jest zawarty w 'Zarządzanie pakietami Debiana' na 83 stronie. Ten rozdział opisuje podstawowe szczegóły.

2.3.1 Sposoby aktualizacji systemu Debian

Zawsze można użyć anonimowego FTP lub programu `wget`, aby dostać się do sieciowego archiwum Debiana, przeszukać katalogi, znaleźć żądany plik, pobrać go i wreszcie zainstalować przy pomocy `dpkg` (`dpkg` instaluje nowe pakiety na właściwym miejscu, również na działającym systemie). Czasami jednak nowa wersja jednego pakietu wymaga instalacji nowej wersji innego pakietu, co może prowadzić do uniemożliwienia instalacji żądanego pakietu do czasu zainstalowania tego drugiego, wymaganego.

Dla wielu osób takie ręczne podejście jest zbyt czasochłonne, ze względu na fakt, że Debian tak szybko ewoluuje - co tydzień dochodzi tuzin lub więcej nowych pakietów. Tuż przed dużymi wydaniami ta liczba jeszcze rośnie. Żeby sobie dać radę z taką lawiną, wiele osób preferuje używanie zautomatyzowanego oprogramowania. Do tego celu powstało kilka wyspecjalizowanych narzędzi do zarządzania pakietami.

2.3.2 Przegląd narzędzi do zarządzania pakietami

System zarządzania pakietami w Debianie posiada dwa cele: manipulacja samymi pakietami oraz pobieranie plików wraz z pakietami z archiwum pakietów. Pierwsze zadanie jest wykonywane przez `dpkg`, drugie - przez APT i `dselect`

2.3.3 `dpkg`

Jest to główny program do manipulacji plikami z pakietami. Pełny opis można znaleźć w `dpkg(8)`.

`dpkg` występuje w towarzystwie kilku prostych programów pomocniczych:

- `dpkg-deb`: Manipulacja plikami `.deb`. `dpkg-deb(1)`
- `dpkg-ftp`: Starszy program do pobierania plików z pakietami. `dpkg-ftp(1)`
- `dpkg-mountable`: Starszy program do pobierania plików z pakietami. `dpkg-mountable(1)`
- `dpkg-split`: Dzieli duży pakiet na mniejsze pliki. `dpkg-split(1)`

`dpkg-ftp` i `dpkg-mountable` zostały zastąpione przez system APT.

2.3.4 APT

APT (skrót od Advanced Packaging Tool - Zaawansowane Narzędzie Pakietujące) jest zaawansowanym interfejsem do debianowego systemu zarządzania pakietami składającym się z kilku programów, których nazwy z reguły zaczynają się od „apt-". `apt-get`, `apt-cache` i `apt-cdrom` są działającymi w środowisku znakowym narzędziami do obsługi pakietów. Są również używane jako „back end" (program wykonujący właściwą pracę pod osłoną interfejsu ułatwiającego użytkownikowi obsługę) dla innych narzędzi takich, jak `dselect` i `aptitude`.

Więcej informacji można uzyskać, instalując pakiet `apt` i czytając `apt-get(8)`, `apt-cache(8)`, `apt-cdrom(8)`, `apt.conf(5)`, `sources.list(5)`, `apt_preferences(5)` (Woody) oraz `/usr/share/doc/apt/guide.html/index.html`.

Innym źródłem informacji może być również APT HOWTO (`http://www.debian.org/doc/manuals/apt-howto/`), dostępny po zainstalowaniu pakietu `apt-howto` jako `/usr/share/doc/Debian/apt-howto/`.

`apt-get upgrade` i `apt-get dist-upgrade` pobierają tylko pakiety wymienione w polach „Depends:" i ignorują wszystkie pakiety wymienione w polach „Recommends:" i „Suggests:". Jeżeli się tego nie lubi, używa się `dselect`.

2.3.5 `dselect`

`dselect` jest zaopatrzonym w menu, interfejsem systemu zarządzania pakietami w Debianie. Jest szczególnie użyteczny przy okazji pierwszych instalacji i większych aktualizacji. Zobacz '`dselect`' na 85 stronie.

Więcej informacji zawiera dokument `/usr/share/doc/install-doc/dselect-beginner.en.html` z pakietu `install-doc` lub dselect Documentation for Beginners (Dokumentacja dselect dla początkujących) (`http://www.debian.org/releases/woody/i386/dselect-beginner`).

2.3.6 Aktualizacja działającego systemu

Kernel i system plików używane w Debianie umożliwiają zastępowanie jednych plików drugimi nawet wtedy, gdy są one właśnie używane.

Dostarczamy również program o nazwie `start-stop-daemon`, używany do uruchamiania demonów (pracujących w tle programów użytkowych) przy starcie systemu i do ich zatrzymywania podczas zmiany trybu pracy kernela (np. z trybu wieloużytkownikowego na jednoużytkownikowy lub na „halt"). Tego samego programu używają skrypty instalacyjne, gdy instalowany jest nowy pakiet zawierający demony - do ich zatrzymywania i uruchamiania w miarę potrzeb.

Nawiasem mówiąc, Debian nie wymaga, aby system poddawany aktualizacji pracował w trybie jednoużytkownikowym.

2.3.7 Pobrane i chwilowo zapisane na dysku pliki `.deb`

Jeżeli ręcznie pobrałeś pliki pakietów na dysk (co nie jest absolutnie konieczne, wystarczy zapoznać się z wyżej zamieszczonym opisem `dpkg-ftp` lub APT), to po ich zainstalowaniu możesz usunąć ze swojego systemu pliki `.deb`.

W wypadku użycia programu APT, pakiety są zapisywane w katalogu `/var/cache/apt/archives/`. Można je skasować po zainstalowaniu (`apt-get clean`) albo skopiować do katalogu `/var/cache/apt/archives/` na innej maszynie, aby nie ściągać ich kolejny raz przy powtórnych instalacjach.

2.3.8 Rejestracja zmian w pakietach

`dpkg` rejestruje pakiety, które rozpakowano, skonfigurowano, usunięto częściowo lub całkowicie, ale (przynajmniej obecnie) nie przechowuje rejestru tego, co się działo na konsoli w czasie poddawania pakietów tym działaniom.

Najprostszym sposobem obejścia tego problemu jest uruchamianie sesji `dpkg`, `dselect`, `apt-get` itd. przy pomocy programu `script` (`script(1)`).

2.4 Proces ładowania systemu w Debianie

2.4.1 Program `init`

Jak wszystkie Uniksy, Debian ładuje się do pamięci wykonując program `init`. W pliku konfiguracyjnym programu `init` (`/etc/inittab`) jest zapisane, że w pierwszej kolejności ma być wykonany skrypt `/etc/init.d/rcS`. Uruchamia on wszystkie skrypty znajdujące się w katalogu `/etc/rcS.d/` poprzez nowe podprocesy lub ich kopie, zależnie od rozszerzenia nazwy pliku, wykonując inicjalizację systemu, w skład której wchodzi sprawdzanie i montowanie systemów plików, ładowanie modułów, uruchamianie usług sieciowych, ustawianie zegara i in. Następnie, dla kompatybilności z innymi systemami, uruchamia skrypty umieszczone w katalogu `/etc/rc.boot/` (z wyjątkiem tych, których nazwy zawierają „."). Skrypty umieszczone w tym katalogu są zwykle zarezerwowane do wyłącznego użytku administratora i używanie ich w pakietach nie jest pochwalane. Więcej informacji można znaleźć w Podręczniku Polityki Debiana w 'Inicjalizacja systemu' na 139 stronie i System run levels and init.d scripts (`http://www.debian.org/doc/debian-policy/ch-opersys#s-sysvinit`).

2.4.2 Poziomy startu (Runlevels)

Po załadowaniu systemu, `init` wykonuje wszystkie skrypty startowe w katalogu określonym przez domyślny poziom startu (default runlevel, wpis `id` w pliku `/etc/inittab`). Jak większość Uniksów kompatybilnych z System V, Linux ma 7 poziomów startu:

- 0 (zatrzymanie systemu),
- 1 (tryb jednoużytkownikowy),
- 2 do 5 (różne tryby wieloużytkownikowe) oraz
- 6 (restart czyli przeładowanie systemu).

W Debianie ustawia się `id=2`, co oznacza, że domyślny poziom startu po wejściu w tryb wieloużytkownikowy wynosi 2, a uruchomieniu podlegają skrypty znajdujące się w katalogu `/etc/rc2.d/`.

W rzeczywistości skrypty w każdym z katalogów `/etc/rcN.d/` są tylko symbolicznymi dowiązaniami (symlinkami) do skryptów w `/etc/init.d/`. Ich **nazwy** natomiast dobiera się tak, aby odzwierciedlały **sposób**, w jaki zostaną uruchomione skrypty znajdujące się w `/etc/init.d/`. W szczególności, przed wejściem na którykolwiek poziom startu uruchomione zostają wszystkie skrypty o nazwach zaczynających się na „K"; są to skrypty wyłączające usługi. Następnie uruchomione zostają skrypty o nazwach zaczynających się na „S", które są skryptami uruchamiającymi usługi. Dwucyfrowa liczba występująca po „K" lub „S" określa kolejność, w jakiej skrypty zostaną uruchomione. Skrypty z mniejszymi liczbami są uruchamiane w pierwszej kolejności.

To wszystko działa, ponieważ skrypty w `/etc/init.d/` pobierają argument, którego wartością może być „start", „stop", „reload", „restart" lub „force-reload" i wykonują zadanie określone przez ten właśnie argument. Skryptów tych można używać również po załadowaniu systemu, sterując w ten sposób różnymi procesami.

Na przykład, (z argumentem „reload") polecenie

```
# /etc/init.d/exim4 reload
```

wysyła demonowi programu exim4 polecenie powtórnego wczytania pliku konfiguracyjnego.

2.4.3 Modyfikacje procesu ładowania

Debian nie korzysta z pochodzącego z BSD katalogu `rc.local` w celu dostosowywania procesu ładowania do jakichś szczególnych życzeń użytkownika; zamiast tego oferuje następujący mechanizm.

Załóżmy, że system powinien wykonać skrypt `foo` przy starcie lub podczas wchodzenia na któryś z poziomów startu. Administrator powinien wtedy:

1 Umieścić skrypt `foo` w katalogu `/etc/init.d/`.

2 Uruchomić występujące w Debianie polecenie `update-rc.d` z odpowiednimi argumentami, ustawiając w ten sposób dowiązania między wymienionymi w wierszu poleceń plikami w katalogach `rc?.d` a `/etc/init.d/foo`, gdzie *?* jest liczbą od 0 do 6 odpowiadającą jednemu z poziomów startu (runlevel) Systemu V.

3 Przeładować system.

Polecenie `update-rc.d` ustawi dowiązania między plikami w katalogach `rc?.d` a skryptem w `/etc/init.d/`. Nazwa każdego z dowiązań będzie się zaczynać od litery „K” lub „S”, po której wystąpi liczba oraz nazwa skryptu. Gdy system osiągnie poziom startu *N*, skrypty z `/etc/init.d/` posiadające w katalogu `/etc/rcN.d/` dowiązania o nazwach zaczynających się na „K” są wykonywane z argumentem `stop`, następnie są wykonywane skrypty, nazwy odniesień do których zaczynają się na „S”, przyjmując za argument `start`.

Można, na przykład, spowodować uruchomienie skryptu `foo` w toku sekwencji startowej umieszczając go w `/etc/init.d/` i instalując dowiązania poleceniem `update-rc.d foo defaults 19`. Argument `defaults` odnosi się do domyślnych poziomów startu (od 2 do 5). Argument `19` gwarantuje, że `foo` zostanie uruchomiony przed którymkolwiek skryptem zawierającym liczbę 20 lub większą.

2.5 Wsparcie dla różnorodności

Debian oferuje wiele sposobów spełniania życzeń administratora systemu bez groźby uszkodzenia systemu.

- `dpkg-divert`, zobacz 'Polecenie `dpkg-divert`' na 100 stronie.
- `equivs`, zobacz 'Pakiet `equivs`' na 101 stronie.
- `update-alternatives`, zobacz 'Alternatywne polecenia' na 101 stronie.

- `make-kpkg` może zadowolić wymagania stawiane przez wiele bootloaderów. Zobacz `make-kpkg(1)` i 'Standardowa metoda Debiana' na 103 stronie.

Wszystkie pliki w podkatalogach katalogu `/usr/local/` należą do administratora systemu i Debian ich nawet nie tknie. Większość (lub wszystkie) plików w podkatalogach `/etc` to pliki konfiguracyjne i Debian nie nadpisze ich w trakcie instalacji nowszych wersji pakietów, chyba, że administrator wyraźnie sobie tego zażyczy.

2.6 Internacjonalizacja

System Debian jest zinternacjonalizowany: obsługuje wyświetlanie i wprowadzanie znaków w wielu językach, tak na konsoli tekstowej, jak i w Xach. Wiele dokumentów, stron podręcznika systemowego `man` i komunikatów systemu przetłumaczono i tłumaczy się na coraz większą liczbę języków. W trakcie instalacji, Debian zachęca użytkownika do wybrania języka instalacji (czasem nawet lokalnego dialektu).

Jeżeli zainstalowany przez Ciebie system nie obsługuje wszystkich właściwości językowych, których potrzebujesz, lub jeżeli chcesz zmienić język lub zainstalować inną klawiaturę, która obsługiwałaby Twój język, zapoznaj się z 'Lokalizacja' na 173 stronie.

2.7 Debian i kernel

Zobacz 'Jądro systemu Linux w Debianie' na 103 stronie.

2.7.1 Kompilacja jądra ze źródeł innych, niż debianowe

Trzeba zrozumieć obowiązujące w Debianie zasady postępowania odnośnie plików nagłówkowych.

Biblioteki C w Debianie buduje się z wykorzystaniem najnowszych **stabilnych** wydań plików nagłówkowych **jądra** (**kernel**) (Na przykład, Debian-1.2 używał plików nagłówkowych jądra w wersji 5.4.13).

Taka praktyka jest w kontraście ze wszystkimi pakietami zawierającymi pliki źródłowe jądra, które wykorzystują również nowsze wersje plików nagłówkowych. Pliki nagłówkowe jądra rozpowszechniane z jego źródłami są umieszczone w katalogu `/usr/include/linux /include/`.

W razie potrzeby kompilacji programu wymagającego plików nagłówkowych kernela w wersji nowszej, niż zawarte w pakiecie `libc6-dev`, trzeba do linii poleceń w trakcie kompilacji dodać `-I/usr/src/linux/include/`. Coś takiego w pewnym momencie przydarzyło się na przykład pakietowi demona automountera (`amd`). Gdy nowsze jądro zmieniło trochę funkcji obsługujących NFS, `amd` musiał zostać o tym powiadomiony, co wymagało dołączenia najnowszej wersji plików nagłówkowych jądra.

2.7.2 Narzędzia do tworzenia jądra

Użytkowników, którzy chcą (albo muszą) utworzyć własne pakiety kernela, zachęcamy do instalacji pakietu `kernel-package`. Zawiera on skrypt umożliwiający zbudowanie pakietu zawierającego jądro, umożliwiając w ten sposób utworzenie Debianowego pakietu zawierającego już skompilowane jądro przez wydanie polecenia

```
# make-kpkg kernel_image
```

w katalogu `/usr/src/linux/`. Bardziej szczegółowych informacji udostępnia polecenie

```
# make-kpkg --help
```

oraz strona podręcznika systemowego `make-kpkg(1)` i 'Jądro systemu Linux w Debianie' na 103 stronie.

Użytkownicy chcący skompilować najnowszy kernel (albo po prostu dowolny, wymarzony) muszą samodzielnie pobrać kod źródłowy ze swojego ulubionego archiwum sieciowego, o ile nie jest dostępny pakiet kernel-source-*version*, gdzie *version* oznacza wersję kernela. Skrypt startowy `initrd` wymaga specjalnej łaty na kernel o nazwie `initrd`; zobacz `http://bugs.debian.org/149236`.

Szczegółową instrukcję użytkowania pakietu `kernel-package` można znaleźć w pliku `/usr/doc/kernel-package/README.gz`.

2.7.3 Specjalne wyposażenie do obsługi modułów jądra

Pakiet `modconf` zawiera skrypt `/usr/sbin/modconf`, którego można używać w celu wprowadzania własnych modyfikacji do konfiguracji modułów. Skrypt ten wyświetla menu, wypytując użytkownika o szczegóły dotyczące ładowanych sterowników urządzeń w jego systemie. Na podstawie udzielonych odpowiedzi przeprowadzana jest modyfikacja pliku `/etc/modules.conf` (wyszczególniającego aliasy i inne argumenty, których należy użyć w połączeniu z różnymi modułami), w powiązaniu z plikami w `/etc/modutils/` i plikiem `/etc/modules` (zawierającym spis modułów, które trzeba załadować przy starcie systemu).

Podobnie jak źródła kernela zostały wyposażone w plik `Configure.help`, ułatwiający konfigurację kerneli własnej produkcji, tak i pakiet `modconf` daje do dyspozycji serię plików pomocy (w `/usr/share/modconf/`), dostarczających szczegółowych informacji na temat odpowiednich argumentów dla każdego modułu. Przykłady można znaleźć w 'Zmodularyzowane jądro 2.4' na 105 stronie.

2.7.4 Usuwanie starego pakietu z jądrem

Skrypt `kernel-image-`*`NNN`*`.prerm` sprawdza, czy kernel, który jest aktualnie załadowany, nie jest tym samym, który usiłujemy odinstalować. Można więc niechciane pakiety kerneli usuwać używając polecenia

```
# dpkg --purge --force-remove-essential kernel-image-NNN
```

(Oczywiście, w miejsce *NNN* trzeba wpisać wersję i podwersję usuwanego kernela).

Rozdział 3

Wskazówki dotyczące instalacji Debiana

Oficjalna dokumentacja znajduje się na `http://www.debian.org/releases/stable/`, i `http://www.debian.org/releases/stable/installmanual`.

Natomiast wersje rozwojowe na `http://www.debian.org/releases/testing/`, i `http://www.debian.org/releases/testing/installmanual` (ciągle nad tym pracujemy).

Pomimo, że ten rozdział został napisany podczas gdy używano instalatora Potato, większość zawartości została zaktualizowana do instalatora z Woodiego, ale są one dość podobne. Ponieważ Sarge będzie używał zupełnie nowego instalatora, używaj tego rozdziału jako wprowadzenie.

3.1 Główne wskazówki dotyczące instalacji systemu Linux

Nie zapomnij sprawdzić `http://www.debian.org/CD/netinst/` jeśli szukasz skromniejszego obrazu CD instalatora systemu Debian.

Używanie wersji `testing` lub `unstable` Debiana zwiększa ryzyko poważnych błędów. Ryzyko może zostać ograniczone poprzez używanie wielorozruchowego schematu z bardziej stabilną wersją, lub stosując `chroot` tak jak jest to opisane w '`chroot`' na 134 stronie. Ten drugi sposób pozwoli na używanie różnych wersji Debiana jednocześnie na różnych konsolach.

3.1.1 Podstawy zgodności sprzętowych

Linux współpracuje z większością sprzętu PC i może być zainstalowany prawie na każdej maszynie. Dla mnie było to równie łatwe jak instalacja Windowsa 95/98/Me. Lista wspieranego sprzętu stale się powiększa.

Jeśli masz laptopa, sprawdź Linux on Laptops (`http://www.linux-laptop.net/`) w celu uzyskania wskazówek dotyczących modelu i marki.

Moja rekomendacja dla sprzętu PC to:

- SCSI zamiast IDE do pracy, IDE/ATAPI HD do użytku prywatnego.
- IDE/ATAPI CD-ROM (lub CD-RW).
- PCI zamiast ISA, zwłaszcza dla kart sieciowych (NIC).
- Użyć układu NIC. Dobre są Tulip dla PCI, NE2000 dla ISA.
- Unikać PCMCIA (notebook) przy pierwszej instalacji.
- Żadnej myszki, klawiatury itp. na USB, chyba, że lubisz wyzwania :)

Jeśli masz wolny komputer to dobrym pomysłem jest podłączenie dysku do szybszej maszyny na czas instalacji.

3.1.2 Określanie informacji o sprzęcie i układach płyty głównej.

Podczas instalacji będziesz musiał wybrać rodzaj swojego sprzętu lub układ płyty głównej. Czasami nie jest łatwo znaleźć takie informacje. Proponuję wówczas:

1 Rozkręć swój komputer i zajrzyj do środka.
2 Spisz numery większych układów na karcie graficznej i sieciowej, chip blisko portu szeregowego i IDE.
3 Spisz nazwy kart PCI i ISA, patrząc na ich tylną stronę.

3.1.3 Określanie informacji o sprzęcie za pomocą Debiana;

Następujące polecenia powinny Cię naprowadzić na rodzaj sprzętu i jego konfigurację.

```
$ lspci -v |pager
$ pager /proc/pci
$ pager /proc/interrupts
$ pager /proc/ioports
$ pager /proc/bus/usb/devices
```

Polecenia te mogą być wydawane podczas instalacji. W celu włączenia konsoli wciśnij Alt-F2.

Typowa lista przerwań:

- IRQ0: timer output (8254)
- IRQ1: kontroler klawiatury
- IRQ2: kaskadowe połączenie z IRQ8–IRQ15 w PC-AT
- IRQ3: drugi port szeregowy (io-port=0x2F8) (`/dev/ttyS1`)
- IRQ4: drugi port szeregowy (io-port=0x3F8) (`/dev/ttyS0`)
- IRQ5: wolne (bylo to przerwanie HD w komputerach klasy XT)
- IRQ6: kontroler stacji dyskietek (io-port=0x3F0) (`/dev/fd0`, `/dev/fd1`)
- IRQ7: port równoległy (io-port=0x378) (`/dev/lp0`)
- IRQ8: rtc

- IRQ9: przerwanie programowe (int 0x0A), przekierowanie do IRQ2
- IRQ10: wolne
- IRQ11: wolne
- IRQ12: mysz PS/2
- IRQ13: wolne (był kiedyś koprocesor 80287)
- IRQ14: pierwszy kontroler IDE (`/dev/hda`, `/dev/hdb`)
- IRQ15: drugi kontroler IDE (`/dev/hdc`, `/dev/hdd`)

Urządzenia podłączone do portów USB są wyszczególnione w `/proc/bus/usb/devices` jako `Cls=`*`nn`*:

- Cls=00 : nieużywane
- Cls=01 : Audio (głośniki itp.)
- Cls=02 : Communication (MODEM, NIC, . . .)
- Cls=03 : HID (klawiatura, mysz, joystick itp)
- Cls=07 : Printer (drukarka)
- Cls=08 : Mass storage (pamięć masowa: FDD, napęd CD/DVD, HDD, Flash, . . .)
- Cls=09 : Hub (USB hub)
- Cls=255 : specyficzny dla dostawcy

Jeśli klasa urządzenia jest różna od 255 to system je obsłuży.

3.1.4 Określanie informacji o sprzęcie za pomocą innego systemu operacyjnego:

Informacje o sprzęcie można również uzyskać, używając innego systemu operacyjnego.

Zainstaluj inną komercyjną dystrybucję Linuksa. Wykrywanie sprzętu jest w nich jak na razie dużo lepsze niż w Debianie. Ta sytuacja powinna się jednak zmienić wraz z `debian-installer` wprowadzonym w dystrybucji Sarge.

Zainstaluj Windows. Konfiguracja sprzętowa może być ustalona poprzez kliknięcie prawym klawiszem myszy na ikonę 'Mój komputer' w celu otworzenia właściwości, a następnie 'Menedżera urządzeń'. Spisz wszystkie informacje takie jak IRQ, I/O adresy portów i DMA. Niektóre stare karty ISA muszą być instalowane pod DOSem.

3.1.5 Mity o Lilo

„Lilo jest ograniczone do 1024 cylindrów". NIEPRAWDA !

Nowsze `lilo` używane po Debianie Potato posiada wsparcie dla lba32. Jeśli BIOS na Twojej płycie głównej obsługuje lba32 to `lilo` powinno być w stanie ładować spoza 1024 cylindra.

Tylko upewnij się, że istnieje wpis „lba32" w pliku `lilo.conf` jeśli posiadasz stary odpowiednik. Zobacz `/usr/share/doc/lilo/Manual.txt.gz`

3.1.6 GRUB

Nowy program rozruchowy `grub` z projektu GNU Hurd może być zainstalowany na Debianie Woody:

```
# apt-get update
# apt-get install grub-doc
# mc /usr/share/doc/grub-doc/html/
... przeczytaj zawartość
# apt-get install grub
# pager /usr/share/doc/grub/README.Debian.gz
... to też :)
```

By zmienić menu GRUB wyedytuj `/boot/grub/menu.lst`. Zobacz 'Ustawianie parametrów uruchomieniowych GRUB' na 114 stronie, żeby się dowiedzieć jak to zrobić, ponieważ program ten różni się w konfiguracji od `lilo`.

3.1.7 Wybór zestawu dyskietek startowych

Dla Potato i zwykłej instalacji dla domowego komputera lubię używać zestawu dyskietek IDEPCI. Dla Woodiego preferuję zestaw dyskietek bf2.4. Obie używają pakietu `boot-floppies` do tworzenia dyskietek startowych.

Jeśli posiadasz kartę sieciową PCMCIA, będziesz musiał użyć standardowego zestawu dyskietek (większa ilość, ale wszystkie moduły sterowników są dostępne) i skonfigurować kartę podczas ustawiania PCMCIA; nie próbuj konfigurować karty podczas standardowej konfiguracji sieci.

Dla specyficznych systemów może wystąpić potrzeba stworzenia własnej dyskietki ratunkowej. Można tego dokonać podmieniając obraz jądra o nazwie „linux" na standardowej dyskietce ratunkowej Debiana, na inny skompilowany wcześniej na innej maszynie. Szczegóły są opisane w pliku `readme.txt`, który znajduje się na wspomnianej dyskietce. Dyskietka ratunkowa korzysta z systemu plików MS-DOS, tak więc można użyć dowolnego systemu operacyjnego by ją odczytać i zmodyfikować. To powinno uprościć życie ludziom posiadającym specyficzne karty sieciowe itp.

Dla Sarge zalecane jest używanie `debian-installer` i/lub `pgi` do tworzenia dyskietek rozruchowych.

3.1.8 Instalacja

Stosuj się do oficjalnych instrukcji przedstawionych na `http://www.debian.org/releases/stable/installmanual` lub `http://www.debian.org/releases/testing/installmanual` (ciągle nad tym pracujemy).

Jeśli instalujesz system używając `boot-floppies` w `testowym` wydaniu Debiana możesz być zmuszony do włączenia konsoli wciskając Alt-F2 i ręcznej edycji pliku `/etc/sources.list` w celu zmiany „stable" na „testing" by określić źródła APT.

Zamierzam umieścić `lilo` na partycji `/dev/hda3`, podczas instalowania `mbr` na dysku `/dev/hda`. To zmniejszy ryzyko nadpisania informacji startowych.

Podczas instalacji wybieram:

- MD5 passwords „tak”
- shadow passwords „tak”
- Instalacja „advanced” (dselect **) i wybieram
  - Wyłączyć emacs (jeśli zaznaczony), nvi, tex, telnet, talk(d);
  - Załączyć mc, vim oraz nano-tiny lub elvis-tiny. Zobacz ‘`dselect`’ na 85 stronie. Nawet jeśli jesteś przyzwyczajony do Emacsa unikaj go teraz i korzystaj z nano podczas instalacji. Na razie nie instaluj dużych pakietów takich jak TeX (miało to miejsce w Potato). Zobacz ‘Edytory awaryjne’ na 214 stronie by dowiedzieć się o instalacji nano-tiny lub elvis-tiny.
- Na wszelkie pytania podczas konfiguracji pakietów odpowiadam „y”.
- `exim`: wybieram 2 dla maszyn ponieważ wysyłam pocztę poprzez serwer SMTP mojego dostawcy Internetu.

Więcej informacji na temat dselect znajdziesz tu - ‘`dselect`’ na 85 stronie.

3.1.9 Nazwy hostów i adresy IP używane w sieci LAN

Przykład konfiguracji sieci LAN (C podsieć: 192.168.1.0/24):

```
Internet
   |
   +--- Zewnętrzny ISP zapewnia usługę POP (dostęp poprzez fetchmail)
   |
  Punkt dostępowy zapewnia usługi DHCP i SMTP
   |                      :
  Modem kablowy       (Dialup)
   |                      :
Bramka sieciowa, zewnętrzny interfejs: eth0 (numer IP przydzielony przez DHCP)
 używa starego notebooka PC (IBM Thinkpad, 486 DX2 50 MHz, 20MB RAM)
 uruchomione jądro Linux 2.4 z obsługą systemu plików ext3.
 uruchomiony pakiet "‘ipmasq"’ (z poprawkami NAT i firewall)
 uruchomiony pakiet "‘dhcp-client"’ ustawiony dla eth0 (podmienia ustawienia DNS)
 uruchomiony pakiet "‘dhcp"’ skonfigurowany dla eth1
 uruchomiony "‘exim"’ jako smarthost (tryb 2)
 uruchomiony "‘fetchmail"’ z długimi odstępami czasu (fallback)
 uruchomiony "‘bind"’ jako buforujący serwer nazw z sieci do Internetu
                    jako autorytatywny serwer nazw dla sieci wewnętrznej
 uruchomione "‘ssh"’ na portach 22 i 8080 (możliwość połączenia skądkolwiek)
 uruchomiony "‘squid"’ jako buforujący serwer dla archiwów Debiana (APT)
Bramka sieciowa, wewnętrzny interfejs: eth1 (IP = 192.168.1.1, stały)
                         |
         +--- LAN Switch (100base T) ---+
         |                              |
Kilka klientów ze stałymi IP     Kilka klientów DHCP
(IP = 192.168.1.2-127, stałe)    (IP = 192.168.1.128-200, dynamiczne)
```

Zobacz 'Konfiguracja sieci' na 185 stronie oraz 'Budowanie bramy sieciowej' na 208 stronie by uzyskać więcej szczegółów na temat konfiguracji sieci oraz bramki sieciowej.

3.1.10 Konta użytkowników

By mieć poczucie spójności na różnych maszynach, na początku zakładam kilka takich samych kont.

Zawsze pierwsze konto które zakładam nosi nazwę „admin" lub podobną i kieruję na nie wszystkie listy roota. Konto to przydzielam do grupy `adm` (zobacz '„Dlaczego GNU `su` nie obsługuje grupy `wheel`"' na 142 stronie), która pozwala na przyznanie wielu przywilejów administratora poprzez `su` za pomocą PAM lub polecenia `sudo`. Szczegóły znajdziesz w 'Dodawanie konta użytkownika' na 51 stronie.

3.1.11 Tworzenie systemów plików

Partycje twardego dysku

Wolę zakładać osobne partycje dla poszczególnych katalogów z drzewa „/" by uniknąć utraty danych w przypadku awarii systemu itp.

```
/          == (/ + /boot + /bin + /sbin)
           == 50MB+
/tmp       == 100MB+
/var       == 100MB+
/home      == 100MB+
/usr       == 700MB+ z Xami
/usr/local == 100MB
```

Rozmiar katalogu `/usr` zależy od miejsca jakie zajmują aplikacje i dokumentacja X Window. `/usr` może mieć rozmiar 300MB jeśli ktoś używa tylko trybu tekstowego, ale 2GB–3GB nie jest niczym szczególnym podczas korzystania z wielu aplikacji Gnome. Gdy katalog `/usr` będzie zajmował zbyt dużo miejsca można przenieść pliki z `/usr/share/` na inną partycje. Przy jądrach 2.4 katalog root może potrzebować ponad 200MB

Obecny stan mojej maszyny wygląda następująco (wydruk uzyskasz wprowadzając polecenie `df -h`):

```
System plików         rozm. użyte  dost. %uż. zamont. na
/dev/hda3             300M  106M  179M  38% /
/dev/hda7             100M   12M   82M  13% /home
/dev/hda8             596M   53M  513M  10% /var
/dev/hda6             100M  834k   94M   1% /var/lib/cvs
/dev/hda9             596M  222M  343M  40% /usr
/dev/hda10            596M  130M  436M  23% /var/cache/apt/archives
/dev/hda11            1.5G  204M  1.2G  14% /var/spool/squid
```

(Duży obszar zarezerwowany dla `/var/spool/squid` przeznaczony jest na serwer pośredniczący dla pobierania pakietów.)

Wprowadzając polecenie `fdisk -l` uzyskujemy informacje o wszystkich partycjach na twardym dysku:

```
# fdisk -l /dev/hda # komentarz

/dev/hda1             1        41    309928+   6  FAT16 # DOS
/dev/hda2            42        84    325080   83  Linux # (nie używana)
/dev/hda3   *        85       126    317520   83  Linux # Główna
/dev/hda4           127       629   3802680    5  Extended
/dev/hda5           127       143    128488+  82  Linux swap
/dev/hda6           144       157    105808+  83  Linux
/dev/hda7           158       171    105808+  83  Linux
/dev/hda8           172       253    619888+  83  Linux
/dev/hda9           254       335    619888+  83  Linux
/dev/hda10          336       417    619888+  83  Linux
/dev/hda11          418       629   1602688+  83  Linux
```

Istnieje kilka nieużywanych partycji. Są one założone z myślą o innych systemach operacyjnych, bądź przenoszeniu danych z obciążonych części dysku.

Montowanie systemów plików.

Poprawne montowanie powyższych systemów plików jest realizowane następującym plikiem `/etc/fstab`:

```
# /etc/fstab: statyczna informacja o systemach plików.
#
# (system plikow)  (punkt montowania)  (typ)    (opcje) (dump) (pass)
/dev/hda3       /               ext2    defaults,errors=remount-ro 0 1
/dev/hda5       none            swap    sw                       0 0
proc            /proc           proc    defaults                 0 0
/dev/fd0        /floppy         auto    defaults,user,noauto     0 0
/dev/cdrom      /cdrom          iso9660 defaults,ro,user,noauto  0 0
#
# trzymaj partycje osobno
/dev/hda7       /home           ext2    defaults                 0 2
/dev/hda8       /var            ext2    defaults                 0 2
/dev/hda6       /var/lib/cvs    ext2    defaults                 0 2
# noatime przyspieszy odczyt plików
/dev/hda9       /usr            ext2    defaults,noatime         0 2
/dev/hda10      /var/cache/apt/archives ext2    defaults         0 2
```

```
# bardzo duża partycja dla serwera pośredniczacego (proxy)
/dev/hda11      /var/spool/squid ext2   rw                      0 2

# zapasowa partycja rozruchowa DOS
/dev/hda1       /mnt/dos         vfat   rw,noauto               0 0
# zapasowa partycja rozruchowa Linux (nie zrobiona)
/dev/hda2       /mnt/linux       ext2   rw,noauto               0 0
#
# punkty montowań nfs
mickey:/        /mnt/mickey      nfs    ro,noauto,intr          0 0
goofy:/         /mnt/goofy       nfs    ro,noauto,intr          0 0
# minnie:/ /mnt/minnie smbfs ro,soft,intr,credentials={filename} 0 2
```

Przy NFS, używam opcji `noauto,intr` w połączeniu z domyślną opcją `hard`. Tym sposobem, istnieje możliwość wciskając kombinację Ctrl-C, uratowania się z sytuacji gdy nie mamy połączenia i proces montowania zawiesi się.

W przypadku maszyny z systemem Windows podłączonej poprzez Sambę (smbfs), opcje `rw,auto,soft,intr` powinny być odpowiednie. Zobacz 'Konfiguracja Samby' na 42 stronie.

Dla stacji dyskietek użycie opcji `noauto,rw,sync,user,exec` zapobiega uszkodzeniu plików przy przypadkowym wyjęciu dyskietki przed jej odmontowaniem. Dzieje się to jednak kosztem wolniejszego zapisu.

Montowanie autofs

Wskazówki do automatycznego montowania:

- Załaduj moduł `vfat` by `/etc/auto.misc` zawierało `-fstype=auto`:
  ```
  # modprobe vfat # przed próbą dostępu do dyskietki
   ... lub by zautomatyzować tę czynność,
  # echo "vfat" >> /etc/modules
   ... następnie uruchom ponownie system.
  ```
- Ustaw `/etc/auto.misc` następująco:
  ```
  floppy -fstype=auto,sync,nodev,nosuid,gid=100,umask=000 :/dev/fd0
   ... gdzie gid=100 to "‘users"’.
  ```
- Stwórz dowiązania `cdrom` i `floppy` w `/home/`*`user`*, które wskazywać będą na `/var/autofs/misc/cdrom` i `/var/autofs/misc/floppy`.
- Przydziel użytkownika *user* do grupy „users".

Montowanie NFS

Zewnętrzny serwer Linux NFS (goofy) znajduje się za ścianą ogniową (bramką sieciową). Zasady bezpieczeństwa w mojej sieci w stosunku do niego są mniej restrykcyjne ponieważ jestem jego jedynym użytkownikiem. By umożliwić dostęp do NFS, trzeba po jego stronie dodać do `/etc/exports` następujące wpisy:

```
# /etc/exports: the access control list for filesystems which may be
#               exported to NFS clients.  See exports(5).
/       (rw,no_root_squash)
```

Chcąc uruchomić serwer NFS potrzebne są powyższe wpisy oraz oczywiście instalacja i uruchomienie klienta i serwer NFS.

Dla uproszczenia zazwyczaj tworzę pojedynczą partycję o rozmiarze 2GB na eksperymentalne instalacje Linuksa. Opcjonalnie współdzielę pomiędzy tymi instalacjami partycję wymiany oraz `/tmp`. Schematy zawierające wiele partycji są również przydatne w takich instalacjach. Jeśli natomiast potrzeba nam prostego systemu w trybie konsoli, w zupełności powinno wystarczyć 500MB.

3.1.12 Wytyczne dotyczące pamięci DRAM

Bezwzględne wytyczne dotyczące DRAM.

```
  4MB:  Absolutne minimum by jądro Linuksa mogło funkcjonować.
 16MB:  Minimum dla rozsądnej pracy w konsoli.
 32MB:  Minimum dla prostego systemu X.
 64MB:  Minimum dla systemu X z GNOME/KDE.
128MB:  Komfortowa praca w systemie X z GNOME/KDE.
256MB (lub więcej):  Dlaczego nie jeśli jesteś w stanie je zdobyć?  DRAM jest tani
```

Możesz podczas rozruchu systemu użyć opcji `mem=4m` (lub w lilo `append=”mem=4m„`) by sprawdzić jak zachowa się system z 4MB pamięci operacyjnej. Jeśli posiadasz stary BIOS i więcej niż 64MB pamięci to również musisz użyć odpowiedniego parametru w lilo.

3.1.13 Przestrzeń wymiany

Używam następujących wytycznych dla przestrzeni wymiany:

- Każda partycja wymiany jest < 128MB (jeśli używasz jąder 2.0), < 2GB (dla nowszych jąder)
- Całość = albo (1 do 2 razy rozmiar zainstalowanego ramu) albo (128MB do 2GB) jako punkt wyjścia
- Rozdziel je na osobne napędy i zamontuj wszystkie z opcjami `sw,pri=1` w `/etc/fstab`. Powoduje to, że jądro obsługuje przestrzeń wymiany jako paskowany RAID co gwarantuje maksymalną wydajność.
- Jeśli to możliwe używaj centralnej części dysku do stworzenia partycji wymiany.

Nawet jeśli nie potrzebujesz, to i tak pożądane jest ustawienie partycji wymiany (128MB) po to by system zwolnił zanim zawiesi się z powodu programu z wyciekiem pamięci.

3.2 Konfiguracja Basha

Modyfikuję skrypty startowe powłoki znajdujące się w systemie tak by spełniały moje zachcianki:

```
/etc/bash.bashrc        Zastępuję własnym
/etc/profile            Pozostawiam bez zmian(\w -> \W)
/etc/skel/.bashrc       Zastępuję własnym
/etc/skel/.profile      Zastępuję własnym
/etc/skel/.bash_profile Zastępuje własnym
~/.bashrc               Zastępuję własnym dla każdego konta
~/.profile              Zastępuję własnym dla każdego konta
~/.bash_profile         Zastępuję własnym dla każdego konta
```

Zobacz szczegóły moich przykładowych skryptów (`http://www.debian.org/doc/manuals/debian-reference/examples/`). Lubię mieć przejrzysty system więc ustawiam `umask` na 002 lub 022.

`PATH` jest ustawiany poprzez następujące pliki konfiguracyjne w tej kolejności:

```
/etc/login.defs  - zanim powłoka ustawi zmienną PATH
/etc/profile     (może wywoływać /etc/bash.bashrc)
~/.bash_profile  (może wywoływać ~/.bashrc)
```

3.3 Konfiguracja myszy

3.3.1 Myszy PS/2

W przypadku płyty głównej ATX i myszy podłączonej przez PS/2, przepływ sygnału powinien być następujący:

```
mysz -> /dev/psaux -> gpm -> /dev/gpmdata = /dev/mouse -> X
```

W tym przypadku utworzone zostało dowiązanie symboliczne `/dev/mouse` wskazujące na `/dev/gpmdata` po to by niektóre narzędzia konfiguracyjne były szczęśliwe i by rekonfiguracja była prostsza. (Jeśli na przykład zrezygnujesz z używania `gpm` wystarczy, że po jego usunięciu zmienisz dowiązanie symboliczne `/dev/mouse` tak by wskazywało na `/dev/psaux`).

Taki przepływ sygnału pozwala na wyłączenie klawiatury oraz myszki oraz po ponownym ich podłączeniu uruchamiając `gpm` umożliwia jeszcze raz je zainicjalizować. Serwer X pozostanie uruchomiony!

Protokół sygnału przepływającego pomiędzy wyjściem `gpm` a wejściem X może być zaimplementowany na dwa sposoby. Albo jako „ms3" (używa protokołu dla szeregowych 3 przyciskowych myszek Microsoft) albo jako „raw" (używa tego samego protokołu co podłączona myszka). Wybór ten determinuje późniejszy wybór protokołu używanego w konfiguracji X.

By zademonstrować konfigurację, za przykład posłuży mi 3-przyciskowa mysz firmy Logitech (tradycyjna mysz stylu Unix) podłączana do portu PS/2.

Jeśli twoja karta graficzna nie jest wspierana przez nowy X4 i musisz używać starego X3 (niektóre 64 bitowe karty ATI), skonfiguruj `/etc/X11/XF86Config` zamiast `/etc/X11/XF86Config-4` w następujący sposób podczas instalacji pakietów X3.

Wariant ms3

```
/etc/gpm.conf            | /etc/X11/XF86Config-4
=========================+======================================
device=/dev/psaux        | Section "InputDevice"
responsiveness=          |  Identifier "Configured Mouse"
repeat_type=ms3          |  Driver     "mouse"
type=autops2             |  Option     "CorePointer"
append=""                |  Option     "Device"   "/dev/mouse"
sample_rate=             |  Option     "Protocol" "IntelliMouse"
                         | EndSection
```

Jeśli używasz takiego podejścia, konfiguracja myszy odbywa się poprzez edycję `gpm.conf`, natomiast konfiguracja X zostaje niezmieniona. Zobacz przykłady moich skryptów konfiguracyjnych (`http://www.debian.org/doc/manuals/debian-reference/examples/`).

Wariant raw

```
/etc/gpm.conf            | /etc/X11/XF86Config-4
=========================+======================================
device=/dev/psaux        | Section "InputDevice"
responsiveness=          |  Identifier "Configured Mouse"
repeat_type=raw          |  Driver     "mouse"
type=autops2             |  Option     "CorePointer"
append=""                |  Option     "Device"   "/dev/mouse"
sample_rate=             |  Option     "Protocol" "MouseManPlusPS/2"
                         | EndSection
```

Jeśli używasz takiego podejścia, konfiguracja myszy odbywa się poprzez edycję `gpm.conf` oraz konfigurację Xów.

Jak dodać inną myszkę

Urządzenie `gpm` typu `autops2` powinno rozpoznać większość myszek PS/2 dostępnych na rynku. Niestety nie zawsze działa i nie jest dostępne w poprzednikach wersji Woody. Spróbuj w takich przypadkach użyć `ps2` lub `imps2` w pliku `gpm.conf` zamiast `autops2`. By dowiedzieć się więcej o typach myszy `gpm` wpisz :`gpm -t help`. Zobacz `gpm(8)`.

Jeśli używasz 2-przyciskowej myszy PS/2, ustaw protokół X włączając opcje `Emulate3Buttons`. Różnice między protokołami 2-przyciskowej i 3-przyciskowej myszy są automatycznie rozpoznawane i ustawiane przez `gpm` po naciśnięciu środkowego przycisku.

Dla protokołu X z 'Wariant raw' na poprzedniej stronie lub bez `gpm` użyj:

- `IntelliMouse`: szeregowy port myszy (`gpm` repeater with „ms3")
- `PS/2`: port myszy PS/2 (zawsze to testuj na początku)
- `IMPS/2`: inny port PS/2 (2, 3, lub mysz z kółkiem, lepiej)
- `MouseManPlusPS/2`: port myszy Logitech PS/2
- `...`

Zobacz więcej na Mouse Support in XFree86 (`http://www.xfree86.org/current/mouse.html`).

Typowe myszy Microsoft z kółkiem najlepiej pracują z:

```
/etc/gpm.conf              | /etc/X11/XF86Config-4
===========================+=======================================
device=/dev/psaux          | Section "InputDevice"
responsiveness=            |  Identifier "Configured Mouse"
repeat_type=raw            |  Driver     "mouse"
type=autops2               |  Option     "CorePointer"
append=""                  |  Option     "Device"   "/dev/mouse"
sample_rate=               |  Option     "Protocol" "IMPS/2"
                           |  Option     "Buttons" "5"
                           |  Option  "ZAxisMapping" "4 5"
                           | EndSection
```

Dla nowszych laptopów Toshiba włączenie `gpm` przed PCMCIA w skryptach inicjalizujących System-V może pomóc przy zapobieganiu zawieszenia systemu. Dziwne, ale to prawda.

3.3.2 Mysz na USB

Upewnij się, że masz wkompilowane w jądro lub jako moduły wszystkie niezbędne funkcje:

- W części "Input core support„:
  - "Input core support„ (CONFIG_INPUT, `input.o`),
  - "Mouse support„ (CONFIG_INPUT_MOUSEDEV, `mousedev.o`),
- W części "USB support„:
  - "Support for USB„ (CONFIG_USB, `usbcore.o`),
  - "Preliminary USB device filesystem„ (CONFIG_USB_DEVICEFS),
  - "UHCI„ lub "OHCI„ (CONFIG_USB_UHCI || CONFIG_USB_UHCI_ALT || CONFIG_USB_OHCI, `usb-uhci.o` || `uhci.o` || `usb-ohci.o`),
  - "USB Human Interface Device (full HID) support„ (CONFIG_USB_HID, `hid.o`) i
  - "HID input layer support„ (CONFIG_USB_HIDINPUT)

Małymi literami wymienione są nazwy modułów.

Jeśli używasz devfs to stwórz wpis w `/dev/input/mice` zawierający major 13 i minor 63 w następujący sposób:

```
# cd /dev
# mkdir input
# mknod input/mice c 13 63
```

Dla typowej myszki z kółkiem na **USB** konfiguracja powinna wyglądać następująco:

```
/etc/gpm.conf             | /etc/X11/XF86Config-4
==========================+========================================
device=/dev/input/mice    | Section "InputDevice"
responsiveness=           |  Identifier "Generic Mouse"
repeat_type=raw           |  Driver     "mouse"
type=autops2              |  Option     "SendCoreEvents" "true"
append=""                 |  Option     "Device"   "/dev/input/mice"
sample_rate=              |  Option     "Protocol" "IMPS/2"
                          |  Option     "Buttons" "5"
                          |  Option  "ZAxisMapping" "4 5"
                          | EndSection
```

By dowiedzieć się więcej odwiedź Linux USB Project (`http://www.linux-usb.org/`)

3.3.3 Touchpad

Pomimo, że touchpad w laptopach emuluje standardowo 2-przyciskową mysz PS/2, pakiet `tpconfig` pozwala na pełną kontrolę nad urządzeniem. Przykładowo ustawienie `OPTIONS="--tapmode=0„` w `/etc/default/tpconfig` wyłączy zachowanie „click by tap". Ustaw `/etc/gpm.conf` następująco, by używać jednocześnie touchpada i myszy na porcie USB pod konsolą:

```
device=/dev/psaux
responsiveness=
repeat_type=ms3
type=autops2
append="-M -m /dev/input/mice -t autops2"
sample_rate=
```

3.4 Konfiguracja NFS

Konfiguracja NFS odbywa się poprzez przez ustawienie `/etc/exports`.

```
apt-get install nfs-kernel-server
# echo "/ *.nazwa-domeny-dla-lokalnych(rw,no_root_squash,nohide)" \
        >> /etc/exports
```

Szczegóły znajdziesz w moich przykładowych skryptach (`http://www.debian.org/doc/manuals/debian-reference/examples/`).

3.5 Konfiguracja Samby

Szczegółowe informacje:
- `http://www.samba.org/`
- pakiet `samba-doc`

Ustawienie serwera Samba w trybie „share" jest dużo prostsze ponieważ tworzy współdzielone napędy typu WindowsForWorkgroups. Ale tryb „user" jest preferowaną metodą.

Samba może zostać skonfigurowana poprzez `debconf` lub `vi`:

```
# dpkg-reconfigure --priority=low samba # w Woody
# vi /etc/samba/smb.conf
```

Szczegóły znajdziesz w moich przykładowych skryptach (`http://www.debian.org/doc/manuals/debian-reference/examples/`).

Dodawanie nowego użytkownika do pliku `smbpasswd` może być wykonane poprzez `smbpasswd`:

```
$ su -c "smbpasswd -a username"
```

Pamiętaj o użyciu zaszyfrowanych haseł dla optymalnej zgodności.

Ustaw `os level` zgodnie z następującymi odpowiednikami systemu (im większa liczba, tym większy priorytet jako serwer)

```
0:      Samba ze swobodną pozycją (nigdy nie będzie przeglądarką nadrzędną)
1:      WfW 3.1, Win95, Win98, Win/Me?
16:     Win NT WS 3.51
17:     Win NT WS 4.0
32:     Win NT SVR 3.51
33:     Win NT SVR 4.0
255:    Samba z potężną mocą
```

Upewnij się, że użytkownicy są członkami grupy do której przynależy katalog umożliwiający współdzielony dostęp, oraz że ścieżka do tego katalogu posiada odpowiednio ustawione uprawnienia bitu wykonywalności.

3.6 Konfiguracja drukarki

Tradycyjną metodą jest `lpr/lpd`. Istnieje nowy system drukowania CUPS™ (Common UNIX Printing System). Inną możliwością jest PDQ. Zobacz Linux Printing HOWTO (`http://www.tldp.org/HOWTO/Printing-HOWTO.html`) w celu uzyskania dodatkowych informacji.

3.6.1 `lpr/lpd`

Dla programów kolejkujących typu `lpr/lpd` (`lpr`, `lprng` i `gnulpr`) podłączonych do drukarek PostScript lub tekstowych ustaw `/etc/printcap` następująco (podstawy):

```
lp|alias:\
        :sd=/var/spool/lpd/lp:\
        :mx#0:\
        :sh:\
        :lp=/dev/lp0:
```

Znaczenie powyższych linii:
- Nagłówek: *lp* – nazwa kolejki, *alias* = alias
- mx#0 – maksymalny rozmiar pliku nieograniczony
- sh – wstrzymywanie drukowania sekwencji nagłówków stron
- lp=/dev/lp0 – lokalne urządzenie drukujące, lub port@komputer dla zdalnych

To dobra konfiguracja jeśli posiadasz drukarkę PostScript. Jest to również dobre ustawienie do drukowania z maszyny Windows poprzez Sambę, dla każdej drukarki obsługiwanej przez Windows (nie jest obsługiwana komunikacja dwukierunkowa). Musisz wybrać odpowiednie ustawienia drukarki na maszynie z Windows.

Jeśli nie masz drukarki PostScript powinieneś ustawić filtrowanie za pomocą `gs`. Istnieje wiele narzędzi automatycznie konfigurujących `/etc/printcap`. Każda z tych kombinacji jest właściwa:
- `gnulpr`, (`lpr-ppd`) i `printtool` — Właśnie tego używam.
- `lpr` i `apsfilter`
- `lpr` i `magicfilter`
- `lprng` i `lprngtool`
- `lprng` i `apsfilter`
- `lprng` i `magicfilter`

W przypadku uruchamiania narzędzia konfiguracyjnego wykorzystującego graficzny interfejs użytkownika, jak na przykład `printtool`, zobacz najpierw 'Zdobywanie uprawnień administratora w X' na 160 stronie aby dowiedzieć się jak uzyskać przywileje administratora. Kolejki wydruku utworzone przez `printtool` używają programu `gs` i występują jako drukarki PostScript. Dlatego, gdy z nich korzystasz, używaj sterowników PostScriptowych. W systemie Windows „Apple LaserWriter" jest standardowym przykładem takiej drukarki..

3.6.2 CUPS™

Zainstaluj Common UNIX Printing System (lub CUPS™):

```
# apt-get install cupsys cupsys-bsd cupsys-client cupsys-driver-gimpprint
# apt-get install foomatic-db-engine foomatic-db-hpijs
# apt-get install foomatic-filters-ppds foomatic-gui
```

Następnie skonfiguruj system używając przeglądarki internetowej:

```
$ mybrowser http://localhost:631
```

Przykład w jaki sposób dodać drukarkę do listy dostępnych:

- na głównej stronie kliknij „Printers”, a następnie „Add Printer”,
- wpisz użytkownika „root” i podaj jego hasło,
- postępuj według instrukcji by dodać drukarkę
- wróć do strony „Printers” i kliknij „Configure Printer”, następnie
- ustaw rozmiar papieru, rozdzielczość i inne parametry.

Więcej informacji znajdziesz na `http://localhost:631/documentation.html` i `http://www.cups.org/cups-help.html`.

3.7 Inne wskazówki instalacyjne

3.7.1 Doinstaluj jeszcze trochę pakietów po wstępnej instalacji

Jeśli wszystko przebiegło bez problemów to posiadasz już mały, funkcjonalny system Debian. Czas na instalację większych pakietów.

- Uruchom `tasksel`. Zobacz ‘Instalowanie *zadań* programem `tasksel` lub `aptitude`’ na 84 stronie.

  Możesz wybrać następujące elementy jeśli są Ci potrzebne:
  - End-user – X window system
  - Development – C i C++
  - Development – Python
  - Development – Tcl/Tk
  - Miscellaneous – środowisko TeX/LaTeX

  - Dla innych wolę używać `tasksel` jako przewodnika po komponentach wyszczególnionych w <Task Info> i instalować je programem `dselect`.

- Uruchom `dselect`.

  Pierwszą rzeczą jaką możesz zrobić jest wybór Twojego ulubionego edytora i innych programów, które są Ci potrzebne. Możesz zainstalować wiele odmian Emacsa na raz. Zobacz '`dselect`' na 85 stronie i 'Popularne edytory' na 213 stronie.

  Możesz również zastąpić niektóre domyślne pakiety takimi z większymi możliwościami.
  - ...

- ...

Zazwyczaj edytuję `/etc/inittab` w celu łatwego wyłączania komputera.

```
...
# Co zrobić gdy wciśnięte są klawisze CTRL-ALT-DEL
ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -h now
...
```

3.7.2 Moduły

Moduły dla sterowników urządzeń są konfigurowane podczas podstawowej instalacji. `modconf` oferuje możliwość konfiguracji modułów przy pomocy systemu menu. Jest on bardzo użyteczny, gdy jakieś moduły zostały opuszczone podczas instalacji nowego jądra.

Nazwy wszystkich wstępnie ładowanych modułów są wyszczególnione w pliku `/etc/modules`. Używam również `lsmod` i `depmod` by sterować nimi ręcznie.

Dla jądra 2.4 dodaj wpis w pliku `/etc/modules` pozwalający obsłużyć IP masquerading (FTP, itp.). Zobacz 'Zmodularyzowane jądro 2.4' na 105 stronie, a dokładniej 'Obsługa sieci' na 106 stronie.

3.7.3 Podstawy instalacji CD-RW

Wyedytuj następujące pliki:

```
/etc/lilo.conf  (dodaj append="hdc=ide-scsi ignore=hdc",
                 uruchom lilo w celu aktywacji)
/dev/cdrom      (symlink # cd /dev; ln -sf scd0 cdrom)
/etc/modules    (dodaj "ide-scsi" i "sg". Jeśli będzie to konieczne to również
"sr".)
```

Zobacz 'Nagrywarki CD' na 145 stronie by dowiedzieć się więcej.

3.7.4 Automatyczne wyłączanie komputera oraz komputery z większą ilością pamięci

Zmodyfikuj `/etc/lilo.conf` wg instrukcji poniżej aby uwzględnić obsługę większego rozmiaru pamięci (dla jąder 2.2) oraz automatycznego wyłączania (dla APM):

```
append="mem=128M apm=on apm=power-off noapic"
```

Teraz uruchom `lilo` aby zainstalować te ustawienia. `apm=power-off` jest wymagane dla jąder z obsługą wielu procesorów, a `noapic` jest potrzebne by unikać kłopotów z moim wadliwym sprzętem. Tego samego można dokonać wpisując podane opcje bezpośrednio podczas ukazania się znaku zachęty programu uruchomieniowego. Zobacz 'Inne triki stosowane podczas startu systemu' na 114 stronie.

Jeśli APM jest kompilowany jako moduł, jak to ma miejsce w domyślnym jądrze Debiana 2.4, uruchom `insmod apm power_off=1` po wystartowaniu systemu lub wpisz odpowiednie opcje do `/etc/modules`, na przykład tak:

```
# echo "apm power_off=1" >>/etc/modules
```

Kompilując obsługę ACPI można osiągnąć to samo z nowszymi jądrami i wygląda na to, że jest to bardziej przyjazne dla wieloprocesorowych maszyn (wymaga to jednak nowszej płyty głównej). Na jądrach z serii 2.4 i nowszych płytach głównych większe ilości pamięci powinny być wykrywane prawidłowo.

```
CONFIG_PM=y
CONFIG_ACPI=y
...
CONFIG_ACPI_BUSMGR=m
CONFIG_ACPI_SYS=m
```

i dodaj w tej kolejności poniższe linie do `/etc/modules`:

```
ospm_busmgr
ospm_system
```

Lub przekompiluj jądro z tymi opcjami ustawionymi na „y". W żadnym z przypadków użycia ACPI, nie są potrzebne żadne parametry podczas uruchamiania.

3.7.5 Dziwne kłopoty z dostępem do witryn WWW.

W nowszych jądrach Linuksa standardowo włączone jest ECN. Może to powodować problemy z dostępem do niektórych witryn WWW. By sprawdzić status ECN:

```
# cat /proc/sys/net/ipv4/tcp_ecn
 ... lub
# sysctl net.ipv4.tcp_ecn
```

Chcąc go wyłączyć, użyj:

```
# echo "0" > /proc/sys/net/ipv4/tcp_ecn
 ... lub
# sysctl -w net.ipv4.tcp_ecn=0
```

W celu wyłączenia ECN podczas każdego restartu, wyedytuj plik `/etc/sysctl.conf` i dodaj:

```
net.ipv4.tcp_ecn = 0
```

3.7.6 Konfiguracja połączenia modemowego (Dialup PPP)

Zainstaluj pakiet `pppconfig` w celu skonfigurowania połączenia z Internetem za pomocą modemu.

```
# apt-get install pppconfig
# pppconfig
 ... stosuj się do wskazówek by skonfigurować połączenie modemowe
# adduser nazwa_użytkownika dip
 ... pozwól użytkownikowi >nazwa_użytkownika na dostęp do modemu
```

Połączenie modemowe może być zainicjowane przez użytkownika (>*nazwa_użytkownika*):

```
$ pon nazwa_ISP   # rozpoczęcie dostępu poprzez modem do Twojego ISP
 ... możesz cieszyć się Internetem
$ poff nazwa_ISP # zatrzymuje dostęp, nazwa_ISP dowolna.
```

Więcej informacji znajdziesz na `/usr/share/doc/ppp/README.Debian.gz`.

Do skonfigurowania połączenia modemowego może być też użyty pakiet `wvdial`. Zauważ, że jest dobrze znany błąd `http://bugs.debian.org/82095`, który czasem powoduje, że nie-administrator nie może się wdzwonić.

Wszystkie te programy używają demona `pppd`, który uruchamia programy znalezione w `/etc/ppp/ip-up.d/` lub `/etc/ppp/ip-down.d/` po tym gdy się połączy lub rozłączy. Jest to używane do pobierania i wysyłania poczty.

3.7.7 Inne pliki konfiguracyjne do dostrojenia w `/etc/`

Możesz chcieć dodać plik `/etc/cron.deny`, brakujący po podstawowej instalacji Debiana (skopiuj `/etc/at.deny`)

Rozdział 4

Debian - samouczek

Ten rozdział przedstawia podstawowe informacje na temat systemu Debian i jest przeznaczony dla zupełnie początkujących użytkowników (newbies). Jeśli kiedykolwiek korzystałeś już z systemu uniksopodobnego, prawdopodobnie znasz wszystkie omówione w tym rozdziale informacje, ale możesz przeczytać go dla odświeżenia swojej wiedzy.

4.1 Zaczynamy

Po zainstalowaniu systemu Debian na swoim komputerze, musisz nauczyć się kilku rzeczy, aby go używać. Pozwól, że damy ci szybki kurs.

4.1.1 Logowanie do powłoki jako root

Po uruchomieniu systemu (reboot), zostanie przedstawiony Tobie graficzny lub tekstowy ekran logowania (zależnie od tego, który wybrałeś podczas wstępnego wyboru pakietów). Dla uproszczenia, jeśli pojawił się graficzny ekran logowania, naciśnij Ctrl-Alt-F1, [1] aby uzyskać tekstowy ekran logowania.

Przypuśćmy, że nazwa twojego hosta to *foo*. Wówczas twój znak zachęty wygląda tak:

```
foo login:
```

Wpisz `root`, naciśnij klawisz Enter, a następnie wpisz hasło, które wybrałeś podczas procesu instalacji. W systemie Debian, zgodnie z tradycją systemów Unix, wielkość liter w haśle ma znaczenie. Po podaniu hasła system uruchomi się z informacją powitalną i przedstawi ci znak zachęty linii poleceń użytkownika root, czekając na wprowadzenie poleceń. [2]

---

[1] Klawisze lewy-Ctrl, lewy-Alt i klawisz F1 należy przycisnąć jednocześnie.

[2] Zauważ, że jeśli edytowałeś informację powitalną w `/etc/motd`, poniższe komunikaty mogą wyglądać inaczej.

```
foo login: root
Password:
Last login: Sun Oct 26 19:04:09 2003 on tty3
Linux foo 2.4.22-1-686 #6 Sat Oct 4 14:09:08 EST 2003 i686 GNU/Linux

Most of the programs included with the Debian GNU/Linux system are
freely redistributable; the exact distribution terms for each program
are described in the individual files in /usr/share/doc/*/copyright

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

root@foo:root#
```

Od tej chwili masz możliwość administracji systemem z linii poleceń użytkownika root. Konto root jest także nazywane kontem superużytkownika (superuser) lub uprzywilejowanego użytkownika. Z tego konta masz możliwość robienia w systemie wszystkiego:

- odczytu, zapisu oraz usuwania każdego pliku w systemie, niezależnie od praw dostępu do tego pliku
- zmiany właściciela pliku oraz praw dostępu do każdego pliku w systemie
- ustawiania haseł dla nieuprzywilejowanych użytkowników w systemie
- logowania się na dowolne konto w systemie bez znajomości jego hasła

Współdzielenie dostępu do konta roota przez posiadanie wspólnego hasła jest fatalnym pomysłem. Użycie programów takich jak `sudo(8)` jest właściwym sposobem na współdzielenie przywilejów administratora.

Zauważ też, że logowanie się najpierw na konto nieuprzywilejowanego użytkownika, a dopiero z niego na konto z przywilejami administracyjnymi jest uważane za dobry zwyczaj uniksowy. Jeśli zajdzie potrzeba, używaj komend `sudo`, `super` lub `su -c`, aby uzyskać ograniczone przywileje roota. Zobacz 'Bezpieczniejsza praca – `sudo`' na 143 stronie. [3]

4.1.2 Tworzenie minimlnego środowiska dla początkującego

Uważam, że nauka systemu operacyjnego komputera przypomina nieco naukę języka obcego. Mimo, że książki są pomocne, to nic nie zastąpi samodzielnej praktyki. Musisz zacząć działać samodzielnie, przy pomocy ułatwiających to narzędzi. W związku z tym, dobrym pomysłem jest zainstalowanie kilku dodatkowych pakietów, takich jak `mc`, `vim`, `linx`, `doc-linux-text` oraz `debian-policy`. [4]

```
# apt-get update
```

---

[3] Muszę przyznać, że kiedyś używałem konta superużytkownika (konto root) częściej, niż potrzebowałem, głównie z powodu wygody i mojego niechlujstwa.

[4] Dobrym pomysłem może być także instalacja `gpm`, `emacs21` oraz `doc-linux-html`. Zobacz 'Konfiguracja myszy' na 38 stronie i 'Edytory' na 213 stronie.

```
 ...
# apt-get install mc vim lynx doc-linux-text debian-policy
 ...
```

Jeśli pakiety są już zainstalowane w systemie, zostaną dodane.

4.1.3 Dodawanie konta użytkownika

Podczas instalacji, zwykle tworzy się konto użytkownika nieuprzywilejowanego, służące do odbioru poczty elektronicznej wysyłanej do użytkownika root. [5] Ponieważ nie chcesz używać tego specjalnego konta do działalności treningowej, powienieneś utworzyć kolejne konto użytkownika.

Zakładając, że chcesz, aby nazwa nowego użytkownika była *`penguin`*, wpisanie:

```
root@foo:root# adduser penguin
... odpowiedz na wszystkie pytania
```

utworzy to konto. [6] Zanim przejdziemy dalej, nauczmy się paru rzeczy.

4.1.4 Przełączanie między konsolami wirtualnymi

W domyślnej instalacji systemu Debian, dostępnych jest sześć niezależnych pseudo-terminali, czyli można używać konsoli tekstowej na sześciu przełączalnych terminalach VT-100. Przełączać się między nimi można przez jednoczesne naciśnięcie klawisza Lewy-Alt i jednego z klawiszy F1–F6. Każdy pseudo-terminal pozwala na niezależne logowanie się na konto systemowe. Środowisko wieloużytkownikowe to wspaniała cecha Uniksa, po poznaniu której ciężko pracować inaczej.

Jeśli przypadkowo naciśniesz Alt-F7 podczas pracy na systemie z uruchomionym środowiskiem graficznym X Window i konsola przełączy się w tryb graficzny, naciśnięcie Ctrl-Alt-F1 pozwoli na przywrócenie dostępu do konsoli tekstowej. Po prostu spróbuj przełączać się na różne konsole i powracać do wyjściowej, aby przywyknąć do środowiska z wieloma pseudoterminalami.

4.1.5 Zamykanie systemu

Tak jak każdy współczesny OS, który cache'uje operacje na plikach w pamięci, także Debian wymaga właściwej procedury zamykania systemu w celu zachowania integralności plików,

---

[5] Zwykle nazywam te konto `admin`, ale może ono mieć dowolną nazwę.

[6] Możesz zechcieć dodać użytkownika *`penguin`* do grupy `adm`, aby umożliwić mu odczyt plików z logami w `/var/log/`. Zobacz `passwd(5)`, `group(5)`, `shadow(5)`, `group(5)`, `vipw(8)` oraz `vigr(8)`. W celu otrzymania informacji o oficjalnym znaczeniu użytkowników i grup, zobacz bieżącą wersję dokumentu Users and Groups (`/usr/share/doc/base-passwd/users-and-groups.html`).

zanim zasilanie może zostać bezpiecznie wyłączone. Używaj następującej komendy z konta root do zamknięcia systemu:

```
# shutdown -h now
```

Tak postępuje się w przypadku trybu wieluużytkownikowego. Jeśli pracujesz w trybie pojedynczego użytkownika (single-user), używaj komendy:

```
# poweroff -i -f
```

Możesz także nacisnąć Ctrl-Alt-Delete aby zamknąć system. [7]

Zaczekaj na pojawienie się napisu "System halted„ i dopiero wówczas wyłącz zasilanie. Jeśli funkcje APM lub ACPI zostały poprawnie włączone w BIOSie i Linuksie, system wyłączy się samodzielnie. Doczytaj szczegóły w 'Automatyczne wyłączanie komputera oraz komputery z większą ilością pamięci' na 46 stronie.

4.1.6 Do rzeczy

Teraz jesteś gotów rozpocząć zabawę z systemem Debian, przy czym dopóki używasz nieuprzywilejowanego konta *`penguin`*, nie musisz obawiać się, że coś zepsujesz w systemie. [8]

Zaloguj się na konto *`penguin`*. Jeśli jesteś zalogowany jako root, naciśnij Ctrl-D [9] aby zamknąć powłokę root i powrócić do znaku zachęty logowania. Wprowadź nowoutworzoną nazwę użytkownika *`penguin`* i przynależne jej hasło. [10] Zostanie ci przedstawiony następujący znak zachęty.

```
penguin@foo:penguin$
```

Od teraz, poniższy przykładowy zapis będzie używany dla uproszczenia:

- `#` : znak zachęty w koncie root
- `$` : znak zachęty nieuprzywilejowanego użytkownika

Teraz rozpoczniesz naukę systemu Debian, najpierw sposobu prostego 'Midnight Commander (MC)' na sąsiedniej stronie, a następnie sposobu właściwego 'Praca w środowisku uniksopodobnym' na 56 stronie.

---

[7] Klawisz lewy-Ctrl, klawisz lewy-Alt oraz klawisz Delete naciśnięte jednocześnie w konsoli. Przy domyślnych ustawieniach powoduje to reboot systemu. Aby wykonywane było zamknięcie systemu (`shutdown`) z opcją `-h`, należy zmodyfikować `/etc/inittab` zgodnie z wytycznymi z 'Doinstaluj jeszcze trochę pakietów po wstępnej instalacji' na 44 stronie.

[8] Dzieje się tak za sprawą faktu, że system Debian jest skonfigurowany z właściwymi prawami dostępu od razu po domyślnej instalacji. Uniemożliwia to nieuprzywijelowanym użytkownikom uszkodzenie systemu. Oczywiście, nadal mogą istnieć dziury, które mogą zostać wykorzystane do dokonania nadużyć, ale ci, którzy mają powody martwić się o nie, nie powinni czytać tego dokumentu, a Securing Debian Manual (`http://www.debian.org/doc/manuals/securing-debian-howto/`).

[9] Lewy klawisz Ctrl i klawisz d są naciskane jednocześnie. Nie ma potrzeby naciskania klawisza Shift pomimo, że te znaki kontrolne są oznaczone w odniesieniach wielką literą, jako "control D„.

[10] Jeśli wpiszesz tutaj `root` zamiast *`penguin`* oraz podasz przynależne hasło użytkownika `root`, ponownie otrzymasz dostęp do konta `root`. Ta procedura będzie konieczna do odzyskania dostępu do konta `root`.

4.2 Midnight Commander (MC)

Midnight Commander (MC) jest uniwersalnym narzędziem konsoli Linuksa oraz innych środowisk terminalowych. Umożliwia ono użytkownikowi zdobycie doświadczenia w wykonywania poleceń systemowych przy użyciu menu, które są poręczniejsze i łatwiejsze w nauce od standardowych poleceń systemu Unix.

Użyj tego narzędzia by przejrzeć strukturę systemu Debian. To najlepszy sposóbnauki. Przy pomocy klwiszy kursorów oraz klasiwsza Enter zajrzyj w kilka kluczowych miejsc:

- `/etc` oraz podkatalogi.
- `/var/log` oraz podkatalogi.
- `/usr/share/doc` oraz podkatalogi.
- `/sbin` and `/bin`

4.2.1 Konfigurowanie MC

Jeśli chcesz aby przy opuszczaniu programu MC zmieniał on aktualny katalog, musisz zmodyfikować plik `~/.bashrc` (lub `/etc/bash.bashrc`, wywoływany z `.bashrc`), tak jak dokładnie opisano to w jego podręczniku, `mc(1)`, przy opcji `-P`. [11]

4.2.2 Uruchamianie MC

```
$ mc
```

MC zajmuje się wszystkimi operacjami na plikach, wykonywanymi z jego menu, wymagając minimalnego wysiłku od użytkownika. Naciśnij po prostu klawisz F1 aby uzyskać ekran pomocy. Możesz poruszać się po całym programie używając klawiszy kursorów oraz klawiszy funkcyjnych. [12]

4.2.3 Menedżer plików w MC

Domyślne ustawienie to dwa okna zawierające listy plików. Kolejnym użytecznym trybem jest ustawienie prawego okna w tryb „informacja”, co pozwala zobaczyć prawa dostępu itp. Poniżej są niektóre ważniejsze skróty klawiszowe. Z użyciem uruchomionego demona `gpm` można też używać myszy. (Upewnij się, że naciskasz klawisz Shift w celu uzyskania normalnego działania Kopiuj i Wklej w MC.)

- F1: Menu pomocy
- F3: Wewnętrzny podgląd pliku
- F4: Wewnętrzny edytor
- F9: Uaktywnienie rozwijanego menu

---

[11] Jeśli nie rozumiesz o czym tutaj mówię, możesz tych zmian dokonać później.

[12] W terminalach takich jak `kon` lub `kterm` występują pewne problemy ze znakami graficznymi języka japońskiego. Dodanie `-a` do wywołania MC może je rozwiązać.

- F10: Wyjście z Midnight Commander
- Tab: Przełączanie pomiędzy dwoma oknami
- Insert: Wybranie pliku do operacji na wielu plikach, np. dla kopiowania
- `Del`: Usunięcie pliku (Bądź ostrożny—ustaw MC w tryb bezpiecznego usuwania.)
- Klawisze kursora: Oczywiste

4.2.4 Triki linii poleceń w MC

- Każde polecenie `cd` zmienia katalog pokazany na wybranym ekranie.
- Ctrl-Enter lub Alt-Enter wkleja nazwę pliku do linii poleceń. Używaj tego łącznie z poleceniem `cp` lub `mv` z edycją w powłoce.
- Alt-Tab pokaże dostępne polecenia z linii poleceń.
- Możesz określić katalogi startowe dla obu okien podając je jako parametry przy wywoływaniu MC; na przykład `mc /etc /root`.
- `Esc` + *klawisz numeryczny* == `F`*n* (np. Esc + 1 = F1, itd.; Esc + 0 = F10)
- Klawisz Esc == klawisz Alt (= Meta, M-); np. wpisz Esc + c by otrzymać Alt-C

4.2.5 Edytor w MC

Wewnętrzny edytor ma interesujący schemat kopiuj-i-wklej. Naciśnięcie F3 oznacza początek wyboru, kolejne wciśnięcie F3 oznacza koniec wyboru i podświetla zaznaczony obszar. Wtedy możesz ruszać kursorem. Gdy naciśniesz F6, zaznaczony obszar zostanie przeniesiony do miejsca, w którym jest kursor. Gdy przyciśniesz F5, zaznaczony obszar zostanie skopiowany i wklejony w miejsce, w którym jest kursor. F2 zapisuje plik. F10 powoduje zakończenie MC. Klawisze kursora zwykle działają intuicyjnie.

Edytor może być wywołany dla pliku:

```
$ mc -e plik_do_edycji
$ mcedit plik_do_edycji
```

Nie jest to edytor wielo-okienkowy, ale możesz osiągnąć podobny efekt używając wielu konsoli linuksowych. Aby kopiować pomiędzy oknami, użyj klawiszy Alt-F*n* do przełączania konsol wirtualnych, a do wklejania fragmentu pliku do innego pliku, użyj „Plik->Wstaw plik" lub „Plik->Kopiuj do pliku".

Wewnętrzny edytor może być zastąpiony przez dowolnie wybrany zewnętrzny.

Wiele programów używa także zmiennej środowiskowej `EDITOR` lub `VISUAL` by określić, którego edytora użyć. Jeśli nie odpowiada Ci `vim`, ustaw go na `mcedit` przez dodanie poniższych linii do `~/.bashrc`:

```
...
export EDITOR=mcedit
export VISUAL=mcedit
...
```

Zalecam jednak ustawienie ich na `vim`, jeśli to możliwe. Przywyknięcie do poleceń `vim` jest dobrą rzeczą, od kiedy edytor Vi jest nieodłączną cechą świata Linuksa/Uniksa. [13]

4.2.6 Podgląd w MC

Bardzo sprytne narzędzie. Jest świetny do wyszukiwania słów w dokumentach. Zawsze używam go do przeglądania plików w katalogu `/usr/share/doc`. To najszybszy sposób na przeglądanie mas informacji o Linuksie. Podgląd można uruchomić bezpośrednio w taki sposób:

```
$ mc -v nazwa_pliku_do_podglądu
```

4.2.7 Własności auto-startu programu MC

Wciśnij Enter na pliku i właściwy program zajmie się jego zawartością. Jest to bardzo wygodna cecha MC.

```
plik wykonywalny:  Wykonanie pliku wykonywalnego
plik man, html:    Przesłanie zawartości pliku przez przeglądarkę
plik tar.gz, plik deb: Przeglądanie zawartości pliku jako podkatalogu
```

By te przeglądarki/wirtualne pliki poprawnie działały, przeglądane pliki nie mogą mieć atrybutu wykonywalności. Zmień ich status używając polecenia `chmod` lub przez menu plików w MC.

4.2.8 Wirtualny system plików FTP w programie MC

MC może służyć do dostępu do plików przez Internet przy użyciu FTP. Wejdź do menu wciskając F9, następnie wpisz 'p', by aktywować wirtualny system plików FTP. Wpisz URL w formie: `nazwa_użytkownika:hasło@nazwa_hosta.nazwa_domeny`, co spowoduje pojawienie się katalogu zdalnego, wyglądającego jak lokalny.

Wypróbuj `http.us.debian.org/debian` jak adres URL i przejrzyj archiwum plików systemu Debian. Przeczytaj 'Archiwa Debiana' na 5 stronie by dowiedzieć się więcej o strukturze katalogów.

---

[13] Aktualnie, `vi` oazr `nvi` to programy, które znajdziesz wszędzie. Wybrałem `vim` dla początkujących gdyż oferuje on pomoc pod klawiszem F1 będąc zarazem podobnym i potężniejszym narzędziem od samego `vi`.

4.3 Praca w środowisku uniksopodobnym

Chociaż MC pozwala Ci zrobić niemal wszystko, jest bardzo ważną rzeczą aby nauczyć się używać narzędzi linii poleceń uruchamianych z powłoki aby zaznajomić się ze środowiskiem uniksopodobnym. [14]

4.3.1 Specjalne sekwencje klawiszy

W środowisku uniksopodobnym, istnieje kilka sekwencji klawiszy które mają znaczenie specjalne. [15]

- Ctrl-U: Usuwa linię przed kursorem.
- Ctrl-H: Usuwa znak przed kursorem.
- Ctrl-D: Kończy standardowe wejście (zamyke powłokę jeśli używasz powłoki)
- Ctrl-C: Kończy działający program.
- Ctrl-Z: Tymczasowo zatrzymuje program (przenosi program w tło, zobacz '`polecenie &`' na 63 stronie)
- Ctrl-S: Wstrzymuje wyjście na ekran. [16]
- Ctrl-Q: Przywraca wyjście na ekran.

Domyślna powłoka `bash`, posiada zdolność edycji-historii oraz oraz możliwość dopełniania poleceń przy pomocy klawisza tab, by wspomóc interaktywne używanie.

- strzałka w górę: Zaczyna przeszukiwać historię poleceń.
- Ctrl-R: Uruchamia przeszukiwanie historii poleceń.
- TAB: Dopełnia wprowadzoną nazwę pliku w linii poleceń.
- Ctrl-V TAB: Wprowadza TAB bez próby rozwinięcia go do dopasowania.

Inne ważne sekwencje klawiszy do zapamiętania:

- Ctrl-Alt-Del: Ponownie uruchom/zatrzymaj system, zobacz 'Doinstaluj jeszcze trochę pakietów po wstępnej instalacji' na 44 stronie.
- Przeciągnięcie-lewym-klawiszem myszy: Zaznacza i kopiuje do schowka.
- Kliknięcie środkowym klawiszem myszy: Wkleja ze schowka w miejsce kursora.
- Meta-klawisz (terminologia Emacs) jest tradycyjnie powiązany z klawiszem lewy Alt. Niektóre systemy mogą być skonfigurowane tak, że Meta-klawisz odpowiada klawiszowi Windows.

Aby móc używać myszki w znakowej konsoli systemu Linux, musisz mieć uruchomionego demona `gpm`. [17] Zajrzyj do 'Konfiguracja myszy' na 38 stronie.

---

[14] W tym rozdzialem, powłoką idomyślną jest `bash`. Po więcej informacji o różnych powłokach zajrzyj do 'Powłoka' na 227 stronie.

[15] W normalnej znakowej konsoli systemu Linux, tylko klawisze Lewy-Ctrl i Lewy-Alt mają oczekiwane funkcje.

[16] Możesz wyłączyć tę opcję terminala używając `stty(1)`.

[17] Wśrodowisku X Window, myszka funkcjonuje w ten sam sposób w programie Xterm.

4.3.2 Podstawowe komendy uniksowe

Nauczymy się teraz podstawowych komend uniksowych. [18] Wypróbuj działanie wszystkich poniższych poleceń używając do tego konta zwykłego użytkownika `penguin`:

- `pwd`
  - Wyświetla nazwę katalogu, w którym się aktualnie znajdujesz.
- `whoami`
  - Wyświetla Twoją nazwę użytkownika.
- `file foo`
  - Wyświetla typ pliku *foo*.
- `type -p nazwa_polecenia`
  - Wyświetla położenie pliku polecenia `nazwa_polecenia`.
  - `which nazwa_polecenia` robi to samo.
- `type nazwa_polecenia`
  - Wyświetla informacje o poleceniu `nazwa_polecenia`.
- `apropos słowo-klucz`
  - Wyszukuje polecenia związazane ze `słowem-kluczowym`.
  - `man -k słowo-klucz` robi to samo.
- `whatis nazwa_polecenia`
  - Wyświetla jednolinijkowe wyjaśnienie o działaniu polecenia `nazwa_polecenia`.
- `man -a nazwa_polecenia`
  - Wyświetla krótkie wyjaśnienie o działaniu poleceniu `nazwa_polecenia`. (styl uniksowy)
- `info nazwa_polecenia`
  - Wyświetla raczej obszerne wyjaśnienie o działaniu polecenia `nazwa_polecenia`. (styl GNU)
- `ls`
  - Listuje zawartość katalogu. (nie wyświetla plików oraz katalogów rozpoczynających się od kropki) [19]
- `ls -a`
  - Listuje zawartość katalogu (wszystkie pliki i katalogi)
- `ls -A`
  - Listuje zawartość katalogu. (wszystkie pliki i katalogi, za wyjątkiem ″`..`„ oraz ″`.`„)
- `ls -la`
  - Listuje całą zawartość wraz ze szczegółowymi informacjami. Przeczytaj 'Koncepcja systemu plików w systemie Debian' na 69 stronie.
- `ls -d`
  - Listuje podkatalogi znajdujące się w bieżącym katalogu.
- `lsof foo`
  - Wyświetla programy korzystające z pliku `foo`.

---

[18] Używam tutaj słowa ″Unix„ w jego ogólnym znaczeniu. Każdy system operacyjny będący klonem systemu Unix oferuje zestaw narzędzi będących odpowiednikami tych uniksowych. System Debian nie jest wyjątkiem. Nie martw się, jeśli jakieś polecenia nie działają zgodnie z Twoimi oczekiwaniami. Poniższe przykłady nie zależą od kolejności ich wykonywania.

[19] system Unix tradycyjnie ukryw pliki, których nazwy rozpoczynają się od ″`.`„. Są to zazwyczaj pliki konfiguracyjne oraz zawierające ustawienia preferencji uzytkownika.

- `mkdir` *`foo`*
  - Tworzy nowy katalog o nazwie *`foo`*, w aktualnym katalogu.
- `rmdir` *`foo`*
  - Usuwa katalog *`foo`* znajdujący się w aktualnym katalogu.
- `cd` *`foo`*
  - Zmienia aktualny katalog na katalog *`foo`* w bieżącym katalogu lub w katalogu wymienionym w zmiennej `CDPATH`. Przeczytaj o poleceniu `cd` w `builtins(7)`.
- `cd /`
  - Zmienia bieżący katalog na katalog główny.
- `cd`
  - Zmienia katalog bieżacy na katalog domowy uzytkownika.
- `cd /`*`foo`*
  - Zmienia katalog bieżący na ścieżke bezwzględna `/`*`foo`*.
- `cd ..`
  - Zmienia katalog na nadrzędny.
- `cd ~`*`foo`*
  - Zmienia katalog na katalog domowy użytkownika *`foo`*.
- `cd -`
  - Zmienia katalog na poprzedni.
- `</etc/motd pager`
  - Wyświetla zawartość pliku `/etc/motd` używając domyślnego pager'a. Przeczytaj '`polecenie <` *`foo`*' na 64 stronie. [20]
- `touch` *`junkfile`*
  - Tworzy pusty plik o nazwie *`junkfile`*.
- `cp` *`foo bar`*
  - Kopiuje istniejący plik *`foo`* do nowego pliku *`bar`*.
- `rm` *`junkfile`*
  - Usuwa plik *`junkfile`*.
- `mv` *`foo bar`*
  - Zmienia nazwę istniejącego pliku *`foo`* na *`bar`*.
- `mv` *`foo bar/baz`*
  - Przenosi istniejący plik *`foo`* do nowego miejsca o nazwie *`bar/baz`*. Katalog *`bar/`* musi istnieć.
- `chmod 600` *`foo`*
  - Usuwa z listy praw dostępu do istniejącego pliku *`foo`* prawa odczytu oraz zapisu dla innych użytkowników. (brak praw wykonywania dla wszystkich)
- `chmod 644` *`foo`*
  - Zmienia prawa dostępu do istniejącego pliku *`foo`* na umożliwiające odczyt ale zabraniające zapisu do pliku dla innych użytkowników. (brak prawa wykonywania dla wszystkich)
- `chmod 755` *`foo`*
  - Zmienia prawa dostępu do istniejącego pliku *`foo`* na prawo oidczytu lecz brak prawa zapisu. (prawo wykonywania dla wszystkich)

---

[20] Domyślnym pager'em systemu Debian jest `more`, który nie posiada opcji przewijania. Po zainstalowaniu pakietu `less` używając polecenia `apt-get install less`, program `less` staje się domyślnym pager'em posiadającym opcję przewijania przy pomocy klawiszy kursorów.

- `top`
    - Wyświetla informacje o procesach na pełnym ekranie. Wciśnij ″q„ by zakończyć.
- `ps aux | pager`
    - Wyświetla informacje o wszystkich działających procesach używając na wyjściu stylu BSD. Zobacz ‘`polecenie1 | polecenie2`’ na 63 stronie.
- `ps -ef | pager`
    - Wyświetla informacje o wszystkich działającyh procesach używając na wyjściu uniksowego stylu system-V.
- `ps aux | grep -e ″[e]xim„`
    - Wyświetla wszystkie procesy `exim`. Więcej na temat wyrażeń regularnych możesz dowiedzieć się ze strony podręcznika `grep(1)` wprowadzając `man grep`.
- `ps axf | pager`
    - Wyświetla informacje o wszystkich działającyh procesach używając na wyjściu stylu tworzenia grafiki przy pomocy znaków ASCII.
- `kill 1234`
    - Ubija proces identyfikowany poprzez jego numer ID: *1234*. Przeczytaj ‘Usuwanie procesów’ na 119 stronie.
- `grep -e ″pattern„ *.html`
    - Wyszukuje i wyświetla wszystkie wystąpienia ″*wzorca*„ w plikach `.html` znajdujących się w bieżącym katalogu.
- `gzip foo`
    - Kompresuje plik `foo` i tworzy `foo.gz` używając kodowania Lempel-Ziv (LZ77).
- `gunzip foo.gz`
    - Dekompresuje plik `foo.gz` i tworzy `foo`.
- `bzip2 foo`
    - Kompresuje plik `foo` i tworzy `foo.bz2` używając algorytmu kompresji tekstu poprzez sortowanie blokowe Burrows-Wheeler i kodowanie metodą Huffman'a. (lepsza kompresja niż przy użyciu `gzip`)
- `bunzip2 foo.bz2`
    - Dekompresuje plik `foo.bz2` i tworzy `foo`.
- `tar -xvvf foo.tar`
    - Wydobywa pliki z archiwum `foo.tar`.
- `tar -xvvzf foo.tar.gz`
    - Wydobywa pliki z archiwum `foo.tar.gz`
- `tar -xvvf --bzip2 foo.tar.bz2`
    - Wydobywa pliki z archiwum `foo.tar.bz2`. [21]
- `tar -cvvf foo.tar bar/`
    - Archiwizuje zawartość folderu `bar/` w pliku archiwum `foo.tar`.
- `tar -cvvzf foo.tar.gz bar/`
    - Archiwizuje zawartość folderu `bar/` w skompresowanym archiwum `foo.tar.gz`.
- `tar -cvvf --bzip2 foo.tar.bz2 bar/`
    - Archiwizuje zawartość folderu `bar/` w archiwum `foo.tar.bz2`. [22]
- `zcat README.gz | pager`

---

[21] Opcja `--bzip2` jest tutaj używana zamiast nowej krótszej `-j` by zapewnić poprawną pracę ze starszą wersją `tar` w Potato.

[22] Opcja `--bzip2` używana jest tutaj dla zachowania kompatybilności.

  - Wyświetla zawartość skompresowanego pliku `README.gz` używając domyślnego pager'a.
- `zcat README.gz > foo`
  - Tworzy plik o zdekompresowanej `foo` zawartości pliku `README.gz`.
- `zcat README.gz >> foo`
  - Dołącza zdekompresowaną zawartość pliku `README.gz` na koniec pliku `foo`. (Jeśli ten plik nie istnieje, zostanie najpierw utworzony)
- `find . -name` *`pattern`*
  - Wyszukuje nazwy plików pasujące do *`wzorca`*. (wolniejsz)
- `locate -d .` *`pattern`*
  - Wyszukuje nazwy plików pasujące do *`wzorca`* powłoki. (szybsze, przy użyciu regularnie aktualizowanej bazy plików)

W ramach treningu przejrzyj dostępne katalogi by zaznajomić się z systemem. Jeśli masz jakiekolwiek pytania, przeczytaj obowiązkowo stronę podręcznika systemowego. Dla przykładu zacznij od poniższych poleceń:

```
$ man man
$ man bash
$ man ls
```

Jest już także odpowiednia pora by uruchomić `vim` i wcisnąć klawisz F1. Powinieneś przeczytać przynajmniej pierwsze 35 linijek. Następnie przesuń kursor nad |`tutor`| i naciśnij Ctrl-] by rozpocząć kurs. Przeczytaj 'Edytory' na 213 stronie by nauczyć się więcej o edytorach.

Zauważ, że wiele polecen uniksopodobnych włączając w to niektóre GNU oraz BSD będą wyświetlały zwięzłą informację jeśli wywołasz je w jeden z poniższych sposobów:

```
$ commandname --help
$ commandname -h
```

Jako ćwiczenie, wypróbuj również przykłady znajdujące się w 'Debianowe porady' na 111 stronie.

4.3.3 Wykonywanie poleceń

Teraz masz już jakieś pojęcie jak używać systemu Debian. Zajrzyjmy więc głębiej by poznać mechanizm wykonywania poleceń w systemie Debian. [23]

4.3.4 Proste polecenie

Proste polecenie jest sekwencją:

---

[23] W tym miejscu opis został przygotowany dla zupełnie początkujących. Po dokładniejsze wyjaśnienia zajrzyj do `bash(1)`.

1 przypisań zmiennych (opcjonalne)
2 nazwy polecenia
3 argumentów (opcjonalne)
4 przekierowań (opcjonalne: > , >> , < , << , etc.)
5 operatorów kontrolujących (opcjonalnie: `&&` , || ; <newline> , `;` , `&` , `(` , `)`)

Bardziej skomplikowane polecenia wraz ze znakami cytowania i podmianą wyrażeń możesz obejrzeć w 'Przetwarzanie linii poleceń' na 231 stronie.

4.3.5 Wykonywanie polecenia oraz zmienna środowiskowa

Typowe wywołanie polecenia używa sekwencji powłoki tak jak poniżej: [24]

```
$ date
Sun Oct 26 08:17:20 CET 2003
$ LC_ALL=fr_FR date
dim oct 26 08:17:39 CET 2003
```

Powyżej, program `date` jest wykonywany jako zadanie w tle. Zmienna środowiskowa `LC_ALL` jest:

- nieustawiona (domyślne zachowanie systemu, tak jak i w przypadku `C`) dla pierwszego polecenia
- ustawiona na `fr_FR` (locale francuskie) dla drugiego polecenia

Większość wykonywanych poleceń nie posiada poprzedzających definicji zmiennej środowiskowej. Dla przykładu powyższego, alternatywnym przykłądem jest:

```
$ LC_ALL=fr_FR
$ date
dim oct 26 08:17:39 CET 2003
```

Jak pewnie zdążyłeś zauważyć, na dane wyjściowe polecenia oddziaływuje zmienna środowiskowa w taki sposób, by otrzymać je po francusku. Jeśli chcesz aby zmienna środowiskowa była dziedziczona przez podprocesy (na przykład podczas działania skryptu powłoki), musisz "wyeksportować„ tę zmienną w taki oto sposób:

```
$ export LC_ALL
```

4.3.6 Ścieżka przeszukiwań poleceń

Kiedy wpisujesz polecenie w powłoce, wyszukuje ona odpowiedniego programu zaglądając do katalogów, które umieszczone są w zmiennej środowiskowej `PATH`. Wartość zmiennej środowiskowej `PATH` zwana jest również ścieżką przeszukiwań powłoki.

---

[24] By otrzymać poniższe wyjście, musisz mieć zainstalowane francuskie locale, zobacz 'Locales' na 174 stronie. To nie jest zbyt istotne dla tego przewodnika i zostało pokazane tylko by uświadomić Ci ewentualne efekty.

W domyślnej instalacji systemu Debian, zmienna środowiskowa `PATH` konta użytkownika może nie zawierać ścieżki `/sbin`. Więc, jeśli chcesz uruchamiać takie polecenia jak `ifconfig` z katalogu `/sbin`, musisz zmienić zmienną środowiskową `PATH` tak by zawierała tę ścieżkę. Zmienna środowiskowa `PATH` jest zazwyczaj inicjalizowana przez plik `~/.bash_profile`, przeczytaj 'Konfiguracja Basha' na 38 stronie.

4.3.7 Opcje linii poleceń

Niektóre polecenia przyjmują argumenty. Te zaczynające się od – oraz od `--` zwane są opcjami i kontrolują one zachowanie się polecenia.

```
$ date
Mon Oct 27 23:02:09 CET 2003
$ date -R
Mon, 27 Oct 2003 23:02:40 +0100
```

Tutaj, argument linii poleceń `-R` zmienia zachowanie polecenia `date` tak, by wyjściowiy ciąg znaków zgodny był z dokumentem RFC-2822.

4.3.8 Wzorce powłoki

Bardzo często zdarza się, że potrzebujemy by polecenie działało z grupą plików, a nie chcemy podawać ich wszystkich ręcznie. Dopełnianie nazw plików przez **wzorce powłoki** jest do tego celu w zupełności wystarczające.

- `*`
  - Zastępuje grupę złożoną z zera lub więcej znaków.
  - Wyjątkiem od tej reguły są pliki zaczynające się od "`.`„.
- `?`
  - Zastępuje dokładnie jeden znak.
- `[...]`
  - Zastępuje dokładnie jeden znak z grupy znajdującej się w nawiasach kwadratowych.
- `[a-z]`
  - Zastępuje dokładnie jeden znak z grupy znaków od `a` do `z`.
- `[^...]`
  - Zastępuje dokładnie jeden znak inny niż te znajdujące się w nawiasach (za wyjątkiem "`^`„).

Dla przykładu, wykonaj poniższe przykłady i przemyśl wyniki jakie dają:

```
$ mkdir junk; cd junk; $ touch 1.txt 2.txt 3.c 4.h .5.txt
$ echo *.txt
1.txt 2.txt
$ echo *
```

```
1.txt 2.txt 3.c 4.h
$ echo *.[hc]
3.c 4.h
$ echo .*
. .. .5.txt
$ echo .[^.]*
.5.txt
$ echo [^1-3]*
4.h
$ cd ..; rmdir junk
```

4.3.9 Wartość zwracana przez polecenia

Każde polecenie kończąc swoje wykonywanie zwraca wartość zwaną statusem zakończenia.

- zwracana wartość = 0 jeśli zakończyło się pomyślnie.
- zwracana wartość = różna od zera, jeśli polecenie zakończyło się z błędem.

Wartość zwrócona przez polecenie jest dostępna zaraz po wykonaniu polecenia w zmiennej środowiskowej `$?`.

Zauważ, że jeśli status zakończenia używany jest w kontekście logicznym powłoki, **powodzenie** jest traktowane jako logiczna **PRAWDA**. Jest to czasem mylące gdyż **powodzenie** dotyczy wartości **zero**.

Przeczytaj 'Polecenia warunkowe powłoki' na 230 stronie.

4.3.10 Właściwa sekwencja poleceń

Spróbuj zapamiętać następujące idiomy poleceń. Zajrzyj do 'Parametry powłoki' na 229 stronie, 'Przekierowywanie powłoki' na 230 stronie, 'Polecenia warunkowe powłoki' na 230 stronie, oraz 'Przetwarzanie linii poleceń' na 231 stronie po przeczytaniu tychże idiomów.

`polecenie &`

`Polecenie` zostanie wykonane w **tle**. Zadania w tle pozwalają użytkownikowi uruchamiać wiele programów w pojedynczej powłoce.

Zarządzanie procesami znajdującymi się w tle dokonywane jest przy pomocy poleceń wbudowanych powłoki: `jobs`, `fg`, `bg` i `stop`. Przeczytaj części "SIGNALS„, "JOB CONTROL„ i "SHELL BUILTIN COMMAND„ w podręczniku systemowym `bash(1)`. [25]

`polecenie1 | polecenie2`

Tutaj, na standardowe wejście `polecenia2` zostaje przekierowane standardowe wyjście `polecenia1`. Oba polecenia mogą działać **równolegle**. Zwane jest to **potokiem**.

---

[25] System Debian jest systemem wielozadaniowym.

`polecenie1 ; polecenie2`

Tutaj, `polecenie1` oraz `polecenie2` wykonywane są **sekwencyjnie**.

`polecenie1 && polecenie2`

Wykonywane jest `polecenie1`. Jeśli zakończy się pomyślnie to wykonywane jest `polecenie2`. Wykonywanie poleceń ma tu charakter **sekwencyjny**. Na końcu zwracany jest status pomyślnego zakończenia, jeśli oba: `polecenie1` **i** `polecenie2` zakończyły się pomyślnie.

`polecenie1 || polecenie2`

Wykonywane jest `polecenie1`. Jeśli nie wykona się ono pomyślnie, wykonywane jest `polecenie2`. Tu także charakter wykonywania poleceń jest **sekwencyjny**. Status zakończenia jest zwracany pomyślny jeśli `polecenie1` **lub** `polecenie2` zakończyło się pomyślnie.

`polecenie >` *`foo`*

Przekierowuje standardowe wyjście z `polecenia` do pliku *`foo`*. (powoduje nadpisanie pliku)

`polecenie >>` *`foo`*

Przekierowuje standardowe wyjście `polecenia` do pliku *`foo`*. (powoduje dopisanie do pliku)

`polecenie >` *`foo`* `2>&1`

Przekierowuje standardowe wyjście oraz wyjście błędów `polecenia` do pliku *`foo`*.

`polecenie <` *`foo`*

Przekierowuje standardowe wejście `polecenia` do pliku *`foo`*. Sprawdź:

```
$ </etc/motd pager
 ... (the greetings)
$ pager </etc/motd
 ... (the greetings)
$ pager /etc/motd
 ... (the greetings)
$ cat /etc/motd | pager
 ... (the greetings)
```

Pomimo, że wszystkie powyższe składnie robią dokładnie to samo, ostatni przykład uruchamia dodatkowe polecenie `cat` i przez to marnuje zasoby systemowe bez powodu.

4.3.11 Aliasy poleceń

Możesz ustawić sobie alias do często wykonywanych poleceń. Na przykład:

```
$ alias la='ls -la'
```

Teraz, `la` działa jak `ls -la` lecz zapisywane jest krócej. Polecenie to powoduje wylistowanie wszystkich plików w katalogu w długim (szczegółowym) formacie.

Możesz określić dokładną ścieżkę lub zidentyfikować polecenie używając `type`. Przykładowo:

```
$ type ls
ls is hashed (/bin/ls)
$ type la
la is aliased to `ls -la'
$ type echo
echo is a shell builtin
$ type file
file is /usr/bin/file
```

W powyższym powłoka ostatnio poszukiwała `ls` podczas gdy nie robiła tego w przypadku `file`, stąd wniosek, że powłoka posiada wewnętrzny bufor w którym przetrzymywana jest lokalizacja najczęściej używanych poleceń (`ls`).

4.4 Uniksopodobne przetwarzanie tekstów

Istnieje kilka standardowych narzędzi do przetwarzania tekstów, które są bardzo często używane w systemach uniksopodobnych.

- Bez użycia wyrażenia regularnego :
  - `head` wyświetla poczśtek pliku.
  - `tail` wyświetla końcowa część pliku.
  - `sort` sortuje linie pliku tekstowego.
  - `uniq` usuwa zduplikowane linie z posortowanego pliku.
  - `tr` zamienia lub kasuje znaki.
  - `diff` porównuje pliki linia po linii.
- Z użyciem podstawowych wyrażeń regularnych (BRE):
  - `grep` dopasowuje tekst do wzorca.
  - `ed` jest prymitywnym edytorem linii poleceń.
  - `sed` jest edytorem strumieni.
  - `vi` jest edytorem ekranowym.

  - `emacs` jest edytorem ekranowym.
- Z użyciem rozszerzonych wyrażeń regularnych (ERE):
  - `egrep` dopasowuje tekst do wzorca.
  - `awk` narzędzie do prostego przetwarzania tekstów. Zajrzyj do 'Awk' na 232 stronie.
  - `perl` potężne narzędzie do przetwarzania tekstów. Zajrzyj do 'Perl' na 233 stronie.

Zajrzyj do'Zastępowanie przy pomocy wyrażeń regularnych' na 124 stronie, 'Kawałki kodu przekierowywania poleceń' na 126 stronie, oraz 'Szaleństwo krótkich skryptów Perla' na 128 stronie Po kilka przykładowych skryptów.

4.4.1 Wyrażenia regularne

Wyrażenia regularne używane są w wielu narzędziach do przetwarzania tekstu. Są one analogią do wzorców powłoki (patrz 'Wzorce powłoki' na 62 stronie) lecz są zarówno bardziej skomplikowane jak i dają o wiele większe możliwości.

Wyrażenie regularne opisuje pasujący wzorzec i jest zbudowane ze znaków tekstowych oraz **metaznaków**. Metaznak jest znakiem ale traktowanym w specjalny sposób. Dwa główne style, BRE oraz ERE, zależne od używanych narzędzi, opisane są w 'Uniksopodobne przetwarzanie tekstów' na poprzedniej stronie.

W stylu ERE, **metaznaki** zawierają ″`\ . [ ] ^ $ * + ? ( ) { } |`„. Wyrażenie regularne oznacza:

- `c`
  - Dopasowuje nie-metaznaki ″`c`„.
- `\c`
  - Dopasowuje znaki liter ″`c`„.
- `.`
  - Zastępuje każdy znak włączając w to znak nowej linii.
- `^`
  - Zastępuje ″początek„ ciągu znaków.
- `$`
  - Zastępuje ″koniec„ ciągu znaków.
- `\<`
  - Zastępuje ″początek„ słowa.
- `\>`
  - Zastępuje ″koniec„ słowa.
- `[abc...]`
  - Ta lista zastępuje każdy ze znaków ″`abc...`„.
- `[^abc...]`
  - Ta zanegowana lista zastępuje każdy ze znaków inny niż ″`abc...`„.
- `r*`
  - Zastępuje zero lub więcej wyrażeń regularnych identyfikowanych przez ″`r`„.
- `r+`
  - Zastępuje jeden lub więcej wyrażeń regularnych identyfikowanych przez ″`r`„.
- `r?`
  - Zastępuje zero lub jedno wyrażenie regularne identyfikowane przez ″`r`„.

- `r1|r2`
  - Zastępuje jedno wyrażenie regularne identyfikowane przez ″`r1`„ lub ″`r2`„.
- `(r1|r2)`
  - Zastępuje jedno z wyrażeń regularnych identyfikowaych przez ″`r1`„ lub ″`r2`„ i traktuje je jako wyrażenie regularne objęte **nawiasami kwadratowymi**.

W stylu BRE **metaznaki** ″`+ ?  ( ) { } |`„ tracą swoje specjalne znaczenie; w zamian należy poprzedzić je znakiem backslash ″`\+ \?  \( \) \{ \} \|`„. Więc konstrukcja grupująca `(r1|r2)` musi być zapisana jako `\(r1|r2\)` w BRE. `Emacs`, wykorzystujący głównie BRE, traktuje ″`+ ?  -`„ jako **metaznaki**. Z tego powodu nie ma potrzeby ich poprzedzania. Sprawdź rozdział 'Wyrażenia zastępujące' na tej samej stronie by dowiedzieć się jak została użyta konstrukcja grupująca.

Dla przykładu, `grep` może zostać użyty do przeszukiwania tekstu z użyciem wyrażeń regularnych:

```
$ egrep 'GNU.*LICENSE|Yoyodyne' /usr/share/common-licenses/GPL
                    GNU GENERAL PUBLIC LICENSE
                    GNU GENERAL PUBLIC LICENSE
  Yoyodyne, Inc., hereby disclaims all copyright interest in the program
```

4.4.2 Wyrażenia zastępujące

W wyrażeniach zastępujących następujące znaki mają specjalne znaczenie:

- `&`
  - Reprezentuje dopasowanie wyrażenia regularnego. (w programie `emacs` używaj `\&`)
- `\`*n*
  - Reprezentuje *n*-te dopasowanie wyrażenia regularnego (**ujętego w nawias**).

W Perlu dla ciągu zastępującego używany jest `$`*n* zamiast `\`*n*, a także `&` nie ma specjalnego znaczenia.

Przykładowo:

```
$ echo zzz1abc2efg3hij4 | \
  sed -e 's/\(1[a-z]*\)[0-9]*\(.*\)$/=&=/'
zzz=1abc2efg3hij4=
$ echo zzz1abc2efg3hij4 | \
  sed -e 's/\(1[a-z]*\)[0-9]*\(.*\)$/\2===\1/'
zzzefg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
  perl -pe 's/(1[a-z]*)[0-9]*(.*)$/$2===$1/'
```

```
zzzefg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
  perl -pe ’s/(1[a-z]*)[0-9]*(.*)$/=&=/’
zzz=&=
```

Zwróć tutaj szczególną uwagę na styl wyrażenia regularnego objętego **nawiasami kwadratowymi** i w jaki sposób ciągi znaków pasujące do wzorca w procesie zastępowania ich wystąpień w tekście, przez różne narzędzia.

Te wyrażenia regularne mogą być takżę użyte do poruszania kursorem oraz do zastępowania tekstu w edytorach.

Przeczytaj prosze wszystkie związane z tym strony podręcznika systemowego, by lepiej zapoznać się z tymi poleceniami.

4.5 Uniksopodobny system plików

W GNU/Linux oraz innych uniksopodobnych systemach operacyjnych **pliki** zorganizowane są w **katalogi**. [26] Wszystkie **pliki** i **katalogi** są zgrupowane w jednym wielkim drzewie, hierarchii plików, której korzeniem (root) jest `/`.

Te pliki i katalogi mogą być rozmieszczone na wielu urządzeniach. Polecenie `mount(8)` pozwala dołączyć pewne systemy plików z podłączonych urządzeń do tego wielkiego drzewa plików. Odwrotnynm z kolei działaniem zajmuje się polecenie `unmount(8)`, które pozwala z powrotem odłączyć dany system plików.

4.5.1 Podstawy uniksowego systemu plików

Oto podstawy:

- Wielkość liter w nazwach plików ma znaczenie. To znaczy, że `MOJPLIK` oraz `MOjPlik` są **całkiem** innymi plikami.
- Katalog główny zwany jest krótko `/`. Nie myl jednak katalogu głównego (root) z użytkownikiem ″root„ . Zobacz ‘Logowanie do powłoki jako root’ na 49 stronie.
- Każdy katalog ma swoją nazwę, która może zawierać wszystkie litery i i symbole **za wyjątkiem** `/`. [27] Katalog główny (root) jest wyjątkiem; jego nazwa to `/` (wymawiane „slesz” lub „katalog główny”) i nie może ona zostać zmieniona.
- Każdy plik lub katalog oznaczony jest poprzez **w pełni określającą nazwę pliku**, **absolutną nazwę pliku** lub **ścieżkę**, podając sekwencję katalogów, przez które należy przejść by dotrzeć do pliku. Te trzy terminy są synonimami. Wszystkie absolutne nazwy plików

---

[26] W niektórych systemach **katalogi** zwane są **folderami**.

[27] Mimo tego, że **możesz** używać prawie wszystkich liter i symboli w nazwie pliku, w praktyce jest to zły pomysł. Lepiej unikać wszelkich znaków, które często mają specjalne znaczenie w linii poleceń, włączając w to spacje, znak tabulacji, znak nowej linii oraz inne: `{ } ( ) [ ] ’ ‘ ″ \ / > < | ; !  # & ^ * % @ $`. Jeśli potrzebujesz rozdzielić słowa w nazwie pliku, dobrym wyborem jest użycie kropki, myślnika lub znaku podkreślenia. Możesz także napisać każde słowo z dużej litery. `NaPrzykładTak`.

rozpoczynają się od katalogu `/` i pomiędzy każdym katalogiem lub nazwą pliku znajduje się `/`. Pierwszy `/` jest nazwą katalogu, ale kolejne są po prostu separatorami by rozróżnić części nazwy pliku. Słowa tutaj używane mogą być mylące. Weźmy następujący przykłąd:

`/usr/share/keytables/us.map.gz`

To jest w pełni określająca nazwa pliku; niektórzy nazywają ją **ścieżką**. Ludzie czasem nazywają nazwą pliku samo `us.map.gz`. [28]

- Katalog główny (root) ma kilka gałęzi, takich jak `/etc` ii `/usr`. Te podkatalogi także posiadają swoje gałęzie (podkatalogi), na przykład `/etc/init.d` i `/usr/local`. Cała ta struktura zwana jest **drzewem katalogów**. Możesz myśleć o w pełni określającej nazwie pliku jak o drodze od podstawy drzewa (`/`), aż do konkretnej gałęzi (pliku). Nie raz też usłyszysz jak inni mówią o drzewie katalogów jakby to było drzewo **rodzinne**, gdzie podkatalogi mają swoich **rodziców**, a ścieżka to dokładne pochodzenie pliku. Istnieją także ścieżki względne, które rozpoczynają się całkiem gdzie indziej niż katalog główny (root). Powinieneś zapamiętać, że `../` odnosi się do katalogu nadrzędnego.
- Nie ma tutaj katalogu, który odnosiłby się do fizycznego urządzenia takiego jak dysk twardy. Jest to zasadnicza różnica w kontekście takich systemó jak CP/M, DOS oraz Windows, gdzie wszystkie ścieżki rozpoczynają się od nazwy urządzenia; na przykład `C:\`. Patrz 'Koncepcja systemu plików w systemie Debian' na bieżącej stronie.

Najlepsze rady dotyczące hierarchii plików zebrane sostały pod adresem Filesystem Hierarchy Standard (`/usr/share/doc/debian-policy/fhs/fhs.txt.gz`). Jako początkujący powinieneć pamiętać następujące fakty:

- `/`
  - Znak `/` reprezentuje katalog główny (root).
- `/etc`
  - To miejsce gdzie przechowywane są pliki konfiguracyjne systemu.
- `/var/log`
  - To miejsce przechowywania logów systemowych.
- `/home`
  - To katalog zawierający katalogi domowe zwykłych użytkowników.

4.5.2 Koncepcja systemu plików w systemie Debian

Zgodnie z tradycją uniksową, system Debian udostępnia system plików, w którym fizyczne dane na dyskach twardych i innych nośnikach, oraz interakcja pomiędzy sprzętem takim jak ekran konsoli i zdalnymi konsolami szeregowymi, zostały ujednolicone.

Każdy plik, katalog, nazwany potok lub urządzenie fizyczne w systemie Debian posiada strukturę danych zwaną **węzłem**, która opisuje związane z nim atrybuty takie jak: właściciel, grupa do której należy, ostatni czas dostępu, itd. Dokładna definicja `struktury iwęzła` w systemie Debian GNU/Linux znajduje się w `/usr/include/linux/fs.h`.

Ta ujednolicona reprezentacja jednostek fizycznych to potężne narzędzie, gdyż pozwala na

---

[28] Słowo **ścieżka** jest również używane w całkiem innym znaczeniu. Patrz 'Ścieżka przeszukiwań poleceń' na 61 stronie. Odpowiednie znaczenie jest przeważnie zawarte w kontekście wypowiedzi.

stosowanie tych samych poleceń do tych samych operacji na całkowicie różnych od siebie urządzeniach.

Wszystkie Twoje pliki mogą znajdować się na jednym dysku — lub możesz mieć 20 dysków, niektóre z nich podłączone do innych komputerów gdzieś w sieci. Nie będziesz mógł tego rozpoznać po samym wyglądzie drzewa katalogów, a niemal wszystkie polecenia działają tak samo i bez znaczenia jest na jakim urządzeniu fizycznym tak na prawdę, znajdują się Twoje pliki.

4.5.3 Prawa dostępu do plików i katalogów

Prawa dostępu do plików i katalogów są definiowane oddzielnie dla następujących trzech kategorii użytkowników, których dotyczą:

- **Właściciel** pliku (u),
- inni użytkownicy w **grupie**, do której należy plik (g), oraz dla
- wszystkich **pozostałych** użytkowników (o).

Dla pliku każde prawo pozwala odpowiednio na:

- **odczyt** (r): odczyt zawartości pliku,
- **zapis** (w): modyfikację zawartości pliku, oraz
- **wykonanie** (x): uruchomienie pliku jako polecenia.

Dla katalogu każde prawo pozwala odpowiednio na:

- **odczyt** (r): wyświetlenie zawartości katalogu,
- **zapis** (w): dodawanie i usuwanie plików w katalogu, oraz
- **wykonanie** (x): dostęp do plików w katalogu.

Tutaj prawo **wykonania** dla katalogu oznacza nie tylko prawo do odczytu plików w nim zawartych, ale także na przeglądanie atrybutów takich jak rozmiar i czas modyfikacji.

Aby otrzymać informacje o uprawnieniach (oraz inne) plików i katalogów, używa się polecenia `ls`. Zobacz `ls(1)`. Gdy `ls` jest wywołane z parametrem `-l`, wyświetla następujące informacje w następującej kolejności:

- **Typ pliku** (pierwsza litera)
  - `-`: zwykły plik
  - `d:` katalog
  - `l:` dowiązanie symboliczne (symlink)
  - `c:` urządzenie znakowe
  - `b:` urządzenie blokowe
  - `p:` nazwany potok
  - `s:` gniazdo
- **prawa** dostępu do pliku (kolejnych dziewięć znaków, po trzy kolejno dla: właściciela, grupy, oraz pozostałych)
- **liczba dowiązań** (hardlink) do pliku
- nazwa **użytkownika** będącego właścicielem pliku
- nazwa **grupy**, do której należy plik
- **rozmiar** pliku w znakach (bajtach)
- **data i czas** modyfikacji pliku (mtime)

- **nazwa** pliku.

Aby zmienić właściciela pliku, używa się z konta root polecenia `chown`. Aby zmienić grupę pliku, używa się polecenia `chgrp` z konta właściciela pliku lub konta root. Aby zmienić prawa dostępu do plików czy katalogów, używa się polecenia `chmod`, także z konta użytkownika lub konta root. Podstawowa składnia pozwalająca na manipulowanie plikiem `foo` to:

```
# chown nowy_wlasciciel foo
# chgrp nowa_grupa foo
# chmod  [ugoa][+-=][rwx][,...] foo
```

Zobacz `chown(1)`, `chgrp(1)`, oraz `chmod(1)` po więcej szczegółów.

Na przykład w celu uczynienia użytkownika *foo* właścicielem drzewa katalogów, a grupy *bar* współwłaścicielem, wykonaj następujące polecenia z konta root:

```
# cd /some/location/
# chown -R foo:bar
# chmod -R ug+rwX,o=rX
```

Istnieją jeszcze trzy specjalne bity uprawnień:

- **ustawia ID użytkownika** (s lub S zamiast x użytkownika),
- **ustawia ID grupy** (s lub S zamiast x grupy), oraz
- **bit lepkości** (sticky bit) (t lub T zamiast x pozostałych).

Na wyjściu polecenia `ls -l` dużymi literami oznaczane są bity, na których miejscu bit wykonywalności nie był ustawiony.

Ustawienie **ID użytkownika** dla pliku wykonywalnego pozwala użytkownikowi na wykonanie tego pliku z ID właściciela pliku (na przykład **root**). Podobnie, ustawienie **ID grupy** dla pliku wykonywalnego, pozwala użytkownikowi na wykonanie pliku z ID grupy pliku (na przykład **root**). Ponieważ może to powodować ryzyko naruszenia bezpieczeństwa, włączanie tych opcji wymaga szczególnej ostrożności.

Ustawienie **ID grupy** dla katalogu uaktywnia schemat tworzenia plików w stylu BSD, w którym wszystkie pliki tworzone w tym katalogu należą do **grupy**, mającej do niego uprawnienia.

Ustawienie **bitu lepkości** dla katalogu zapobiega usuwaniu z niego plików przez użytkowników, którzy nie są właścicielami tych pliku. W celu zabezpieczenia zawartości pliku w katalogach zapisywalnych, takich jak `/tmp` czy w katalogach, w których prawo zapisu ma grupa, potrzeba nie tylko, by było wyłączone prawo **zapisu** dla pliku, ale także należy ustawić **sticky bit** dla katalogu. W innym przypadku plik może zostać usunięty, a następnie w jego miejsce może zostać utworzony nowy plik, o takiej samej nazwie przez dowolnego użytkownika, który ma prawo zapisu do tego katalogu.

Poniżej kilka interesujących przykładów praw do pliku.

```
$ ls -l /etc/passwd /etc/shadow /dev/ppp /usr/sbin/pppd
```

```
crw-rw----    1 root     dip      108,   0 Jan 18 13:32 /dev/ppp
-rw-r--r--    1 root     root         1051 Jan 26 08:29 /etc/passwd
-rw-r-----    1 root     shadow        746 Jan 26 08:29 /etc/shadow
-rwsr-xr--    1 root     dip        234504 Nov 24 03:58 /usr/sbin/pppd
$ ls -ld /tmp /var/tmp /usr/local /var/mail /usr/src
drwxrwxrwt    4 root     root         4096 Feb  9 16:35 /tmp
drwxrwsr-x   10 root     staff        4096 Jan 18 13:31 /usr/local
drwxrwsr-x    3 root     src          4096 Jan 19 08:36 /usr/src
drwxrwsr-x    2 root     mail         4096 Feb  2 22:19 /var/mail
drwxrwxrwt    3 root     root         4096 Jan 25 02:48 /var/tmp
```

Istnieje alternatywna metoda numeryczna do opisu praw pliku w `chmod(1)` poleceniach. Metoda ta używa trzech lub czterech cyfr o znaczeniu ósemkowym. Oznaczają one odpowiednio:

- 1-sza cyfra opcjonalna: suma opcji **ustawione ID użytkownika** (=4), **ustawione ID grupy** (=2), oraz **bit lepkości** (=1)
- 2-ga cyfra: suma praw **odczytu** (=4), **zapisu** (=2), oraz **wykonywania** (=1) dla **użytkownika**
- 3-cia cyfra: to samo ale dla **grupy**
- 4-ta cyfra: to samo ale dla **innych**

Brzmi to dość skomplikowanie ale jest całkiem proste. Jeśli spojrzysz na pierwsze kilka kolumn (2-10) będących wynikiem polecenia `ls -l` i odczytasz ich znaczenie binarnie (podstawa=2) reprezentację praw do pliku (gdzie „-" to „0" a „rwx" to „1"), wartość takiego wyrażenia numerycznego powinna mieć dla Ciebie poprawny sens jako reprezentacja ósemkowa (podstawa=8) praw dostępu do pliku. [29] Spójrz na ten przykład:

```
$ touch foo bar
$ chmod u=rw,go=r foo
$ chmod 644 bar
$ ls -l foo bar
-rw-r--r--    1 penguin  penguin  0 Nov  3 23:30  foo
-rw-r--r--    1 penguin  penguin  0 Nov  3 23:30  bar
```

Domyślna maska praw dostępu do pliku ustawiana jest przy użyciu wbudowanego polecenia powłoki `umask`. Zobacz `builtins(7)`.

4.5.4 Znaczniki czasu

Dla plików GNU/Linux istnieją trzy typy znaczników czasu:

- **mtime**: czas modyfikacji (`ls -l`),
- **ctime**: czas zmiany statusu (`ls -lc`), oraz
- **atime**: czas ostatniego dostępu (`ls -lu`).

Zauważ, że **ctime** nie jest czasem utworzenia pliku.

---

[29] Oczywiśce metoda ta sprawdza się tylko w przypadku 3 cyfr.

- Nadpisanie pliku zmieni wszystkie **mtime**, **ctime** i **atime** pliku.
- Zmiana praw lub właściciela pliku zmieni **ctime** oraz **atime** pliku.
- Odczyt pliku zmieni jego **atime**.

Zauważ, że nawet zwykły odczyt z pliku w systemie Debian powoduje normalnie operację zapisu w celu aktualizacji informacji **atime** w **inode**. Montowanie systemu plików z opcją `noatime` pozwoli systemowi na ominięcie tej operacji i zaowocuje szybszym dostępem do pliku przy odczycie. Zobacz `mount(8)`.

Użyj polecenia `touch(1)` aby zmienić znacznik czasu istniejącego pliku.

4.5.5 Dowiązania (links)

Istnieją dwie metody na związanie pliku *foo* z innym plikiem *bar*.

- **dowiązanie** (hardlink) jest to inna nazwa dla istniejącego pliku (`ln foo bar`),
- **dowiązanie symboliczne** lub „symlink" jest specjalnym plikiem, który wskazuje na inny poprzez nazwę (`ln -s foo bar`).

Spójrz na poniższe przykłady zmian w dowiązaniach, oraz na subtelne różnice w rezultacie polecenia `rm`.

```
$ echo "Oryginalna zawartość" > foo
$ ls -l foo
-rw-r--r--    1 osamu    osamu           4 Feb  9 22:26 foo
$ ln foo bar     # dowiązanie (hardlink)
$ ln -s foo baz  # dowiązanie symboliczne
$ ls -l foo bar baz
-rw-r--r--    2 osamu    osamu           4 Feb  9 22:26 bar
lrwxrwxrwx    1 osamu    osamu           3 Feb  9 22:28 baz -> foo
-rw-r--r--    2 osamu    osamu           4 Feb  9 22:26 foo
$ rm foo
$ echo "Nowa zawartość" > foo
$ cat bar
Oryginalna zawartość
$ cat baz
Nowa zawartość
```

Dowiązanie symboliczne zawsze ma, jak widać w przykładzie powyżej prawa dostępu „rwxrwxrwx", a jego faktyczne prawa dostępu są takie jak wskazywanego pliku.

Katalog `.` jest dowiązaniem do katalogu, w którym sam się pojawia, zatem liczba dowiązań do nowego katalogu rozpoczyna się od 2. Katalog `..` jest dowiązaniem do katalogu nadrzędnego, zatem liczba dowiązań do katalogu rośnie wraz z liczbą nowych podkatalogów.

4.5.6 Nazwane potoki (FIFOs)

Naznway potok jest plikiem, który działa jak potok. Wrzucasz coś do pliku i dostajesz coś z niego na zewnątrz. Zwane jest to FIFO, lub First-In-First-Out (Pierwsze-Weszlo-Pierwsze-

Wyszło): pierwsza rzecz jaką włożysz do takiego pliku, jest pierwszą rzeczą którą możesz z niego odebrać.

Jeśli zapisujesz do nazwanego potoku, proces który tego dokonuje, nie zakończy się dopóki dane, które zostały do niego zapisane, nie zostaną odczytane. Jeśli czytasz z nazwanego potoku, proces który dokonuje odczytu czeka z zakończeniem, dopóki tylko jest coś do odczytania. Nazwany potok ma zawsze zerową wielkość — nie przechowuje żadnych danych, po prostu łączy dwa procesy tak jak | w powłoce. Jednakże, to iż taki potok ma swoją nazwę, powoduje, że dwa procesy nie muszą być w jednej linii poleceń a nawet, mogą być uruchamiane przez różnych użytkowników.

Spróbuj sam wykonując poniższy przykład:

```
$ cd; mkfifo mypipe
$ echo "hello" >mypipe & # put into background
[1] 5952
$ ls -l mypipe
prw-r--r--    1 penguin penguin  0 2003-11-06 23:18 mypipe
$ cat mypipe
hello
[1]+  Done                    echo hello >mypipe
$ ls mypipe
prw-r--r--    1 penguin penguin  0 2003-11-06 23:20 mypipe
$ rm mypipe
```

4.5.7 Gniazda

Gniazdo jest podobne do nazwanego potoku (FIFO) i pozwala procesom na wymianę informacji. W przypadku gniazda jednak, procesy nie muszą być uruchomione w tym samym czasie, a nawet nie nie muszą być procesami potomnymi swoich przodków. To jest punkt styku ich komunikacji międzyprocesowej. Informacje mogą być wymieniane przez sieć pomiędzy dwoma różnymi komputerami.

4.5.8 Pliki urządzeń

Plik urządzenia odnosi się do fizycznego lub wirtualnego urządzenia, takiego jak dysk twardy, karta graficzna, ekran czy klawiatura, w Twoim systemie. Przykładem urządzenia wirtualnego jest konsola, reprezentowana przez `/dev/console`.

Istnieją dwa typy urządzeń:

- **urządzenie znakowe**
  - Dostęp do takiego urządzenia odbywa się znak po znaku, to znaczy, że najmniejsza porcja danych jaka może być zapisana lub odczytana z urządzenia to jeden znak (bajt).
- **urządzenie blokowe**

- Dostęp do takiego urządzenia następuje poprzez zczytania lub zapisanie większej ilości danych, zwanych blokami, które zawierają pewną ilość znaków. Dobrym przykładem jest tutaj Twój dysk twardy.

Możesz odczytywać oraz zapisywać do plików urządzeń, lecz mogą one zawierać również dane binarne o całkowicie niezrozumiałym zapisie dla człowieka. Zapisywanie danych do takich plików jest czasami przydatne przy badaniu problemów tyczących się łączenia sprzętu. Dla przykładu możesz przesłać plik tekstowy do urządzenia drukarki `/dev/lp0` lub wysłać odpowiednie polecenie modemowe na odpowiedni port szeregowy `/dev/ttyS0`. Jednak robiąc to nieostrożnie można spowodować poważne uszkodzenie, więc postępuj z rozwagą.

`/dev/null` i inne

`/dev/null` jest specjalnym plikiem urządzenia, które porzuca wszystko co do niego wrzucisz. Jeśli czegoś nie potrzebujesz, wrzuć to do `/dev/null`. Jest to taka dziura bez dna. Jeśli spróbujesz coś przeczytać z `/dev/null`, dostaniesz natychmiast znak końca pliku.

`/dev/zero` działa podobnie, z tym, że próba odczytu zwraca znak `\0` (to nie to samo zero co „0" w kodzie ASCII). Patrz 'Sztuczne pliki' na 133 stronie.

Numer węzłowy urządzenia

Numer węzłowy urządzenia wyświetlany jet jako wynik polecenia `ls`:

```
$ ls -l /dev/hda /dev/ttyS0 /dev/zero
brw-rw----    1 root     disk       3,   0 Mar 14  2002 /dev/hda
crw-rw----    1 root     dialout    4,  64 Nov 15 09:51 /dev/ttyS0
crw-rw-rw-    1 root     root       1,   5 Aug 31 03:03 /dev/zero
```

Tutaj,

- `/dev/hda` posiada pierwszorzędny numer urządzenia 3 oraz pomoniczy numer urządzenia 0. Ustawione jest też prawo odczytu/zapisu dla użytkowników należacych do grupy `disk`,
- `/dev/ttyS0` posiada pierwszorzędny numer urządzenia 4 oraz drugorzędny numer urządzenia 64. Prawa odczytu/zapisu ustawione są dla użytkowników należących do grupy `dialout` i
- `/dev/zero` posiada pierwszorzędny numer urządzenia 1 oraz drugorzędny numer urządzenia 5. Prawa odczytu/zapisu ustawione są dla wszystkich.

W starszych systemach, proces instalacyjny tworzył węzły urządzeń przy użyciu polecenia `/sbin/MAKEDEV`. Patrz `MAKEDEV(8)`.

W systemach nowszych, system plików `/dev` jest automatycznie zapełniany przez urządzenie systemu plików podobne do `/proc`.

4.5.9 System plików `/proc`

System plików `/proc` jest tak naprawdę pseudo-systemem plików, który zawiera informacje o systemie oraz uruchomionych w nim procesach.

Ludzie często nie wiedzą co robić, gdy zauważają dość duży plik w `/proc/kcore`. Jest to mniej więcej kopia pamięci Twojego komputera. Używana jest do diagnozowania jądra. Obecnie ten plik nie istnieje, więc nie martw się o jego wielkość.

Patrz 'Dostrajanie jądra przy pomocy systemu plików proc' na 109 stronie and `proc(5)`.

4.6 System X Window

Patrz 'X' na 149 stronie.

4.6.1 Uruchamianie systemu X Window

System X Window może być uruchamiany automatycznie z podobnym do `xdm` graficznym demonem logowania lub poprzez wpisanie w konsoli: console.

```
$ exec startx
```

4.6.2 Menu systemu X Window

Od kiedy środowisko X potrafi pomieścić wiele menedżerów okien, zmienił się znacząco interfejs użytkownika. Zapamiętaj, że zawsze po naciśnięciu prawego guzika myszy w głównym oknie menedżera, ujrzysz rozwijalne menu.

- Aby uruchomić powłokę, wybierz z menu program Xterm:
  - „XShells" –> „XTerm".
- Aby przeglądać strony internetowe w trybie graficznym, wybierz Mozillę:
  - „Apps" –> „Net" –> „Mozilla Navigator".
- Aby przeglądać pliki PDF w trybie graficznym, wybierz Xpdf:
  - „Apps" –> „Viewers" –> „Xpdf".

Jeśli nie możesz znaleźć któregoś programu w menu, zainstaluj odpowiednie pakiety. Patrz 'Podstawy zarządzania pakietami Debiana' na 84 stronie.

4.6.3 Skróty klawiszowe w systemie X Window

Poniżej lista ważniejszych sekwencji klawiszy, przydatnych podczas pracy w systemie X Window.

- Ctrl-Alt-F1 do F6: Przełącza na inne pseudo-terminale (podobnie do systemów X window, DOSEMU, itp.)

- Alt-F7: Przełącza z powrotem do systemu X Window
- Ctrl-Alt-minus: Zmienia rozdzielczość w systemie X Window (minus odnosi się do klawisza na klawiaturze numerycznej)
- Ctrl-Alt-plus: Zmienia rozdzielczość w przeciwną stronę w systemie X Window (plus odnosi sie do klawisza na klawiaturze numerycznej)
- Ctrl-Alt-Backspace: Kończy działanie X Serwera
- Alt-X, Alt-C, Alt-V: Standardowe kombinacje klawiszy: wycinanie, kopiowanie oraz wklejanie działające tak jak w systemach Windows/Mac. W niektórych programach powyższe kombinacje z klawiszem Alt zostały zastąpione odpowiednimi kombinacjami z klawiszem Ctrl, na przykłąd w Netscape Composer..

4.7 Dalsza nauka

Na tym etapie wiedzy, polecam Ci zaznajomienie się z kluczowymi książkowymi przewodnikami ze strony The Linux Documentation Project: Guides (`http://www.tldp.org/guides.html`):

- „The Linux System Administrators' Guide”,
  - Opisuje wszelkie aspekty dotyczące utrzymania systemu na chodzie, włączając w to konta użytkownikóa, kopie zapasowe oraz konfigurowanie systemu.
  - pakiet: `sysadmin-guide`
  - plik: `/usr/share/doc/sysadmin-guide/html/index.html`
  - strona: `http://www.tldp.org/LDP/sag/index.html`
- „The Linux Network Administrator's Guide, Second Edition”,
  - Jest to podręcznik administracji siecią w środowisku Linux.
  - pakiet: `(not available)`
  - plik: `(notapplicable)`
  - strona: `http://www.tldp.org/LDP/nag2/index.html`
- „Linux: Rute User's Tutorial and Exposition”
  - A nice online and hardcover book covering GNU/Linux system administration.
  - By Paul Sheer
  - Published by Prentice Hall
  - Package: `rutebook` (from `non-free`)
  - File: `/usr/share/doc/rutebook/`

Zobacz 'Rozwiązywanie problemów' na 251 stronie po więcej materiałów do nauki.

Rozdział 5

Uaktualnianie dystrybucji do wersji `testowej`

Oficjalne informacje o wydaniach uaktualnień są umieszczone na stronie `http://www.debian.org/releases/stable/releasenotes` oraz `http://www.debian.org/releases/testing/releasenotes` (w przygotowaniu).

Aktualizacja systemu do wersji `testowej/niestabilnej` przebiega według następującego schematu:

- aktualizacja systemu APT do wersji Woody, jeśli twój system to Potato, ponieważ stara wersja APT nie posiada cech opisanych w `apt_preferences(5)` w Woodym.
- zmiana pliku `/etc/apt/sources.list` i `/etc/apt/preferences`, by odnosiły się do repozytoriów „testing". Możesz też dodać, jeśli tylko chcesz, odniesienia do „unstable".
- aktualizacja listy pakietów oraz instalacja dowolnego, dającego się zaktualizować pakietu.

5.1 Przejście do APT z Woodiego

Aktualizacja sieciowa systemu APT i kilka pakietów podstawowych do wersji Woody, jeśli nadal używasz Potato, może być wykonana jak poniżej, po dodaniu źródła wersji `stabilnej` do pliku `/etc/apt/sources.list`.

```
# apt-get update
# apt-get install libc6 perl libdb2 debconf
# apt-get install apt apt-utils dselect dpkg
```

5.2 Przygotowanie do przejścia (z wersji `stabilnej` na `testową`)

Przejście na dystrybucję `testową` Debiana ma pewien skutek uboczny - trzeba dłużej czekać na uaktualnienia bezpieczeństwa. Ostrzeżono Cię - wybór należy do Ciebie.

Aktualizacja sieciowa do wersji `testowej` może przebiegać w następujący sposób (uruchom skrypt go-woody (`http://www.debian.org/doc/manuals/debian-reference/examples/`), by wykonać poniższe czynności jednym poleceniem):

Wyczyść istniejący plik `sources.list`:

```
# cd /etc/apt
# cp -f sources.list sources.old
# :>sources.list
```

Pobierz czystą listę repozytoriów dla wersji `stabilnej`:

```
# cd /
# apt-setup noprobe
 ... wybierz repozytoria dostępne przez HTTP lub FTP
```

Dodaj sekcję wersji `testowej` do nowej listy. Linie `deb-src` są zakomentowane.

```
# cd /etc/apt
# grep -e "^deb " sources.list  >sources.deb
# grep -e "^deb-" sources.list  >sources.src
# sed -e "s/stable/testing/"  sources.deb \
    >>sources.list
# sed -e "s/stable/testing/" sources.src | \
    sed -e "s/^deb-/#deb-/"  >>sources.list

# apt-get update
# apt-get install apt apt-utils
# cat >preferences <<EOF
 Package: *
 Pin: release a=testing
 Pin-Priority: 600

 Package: *
 Pin: release a=unstable
 Pin-Priority: 50

 EOF
```

Ewentualnie możesz dodać odnośniki do archiwów wersji `niestabilnej`.

```
# sed -e "s/stable/unstable/" sources.deb \
     >>sources.list
# sed -e "s/stable/unstable/" sources.src | \
     sed -e "s/^deb-/#deb-/"  >>sources.list
```

Zobacz 'Podstawy zarządzania pakietami Debiana' na 84 stronie, gdzie opisana jest sztuka dopieszczania plików `/etc/apt/sources.list` i `/etc/apt/preferences`.

Możesz teraz uaktualnić pakiety, używając jednego ze sposobów opisanych poniżej.

5.3 Aktualizacja systemu

Po prawidłowym ustawieniu plików `/etc/apt/sources.list` i `/etc/apt/preferences`, system jest gotowy do przejścia na wersję `testową`. Przeczytaj rozdział 'Zarządzanie pakietami Debiana' na 83 stronie, by zapoznać się z podstawami, oraz 'Problemy przy aktualizacji APT' na 90 stronie, jeśli będziesz miał problemy.

5.3.1 Najlepiej do aktualizacji jest użyć programu `dselect`

Jeśli system posiada wiele pakietów mających w nazwach `-dev`, itp., poniższa metoda z użyciem `dselect` jest zalecana do właściwej kontroli pakietów.

```
# dselect update # rób to zawsze przed aktualizacją
# dselect select # wybierz pakiety oznaczone "‘suggests"’ i "‘recommends"’
```

Zaraz po uruchomieniu `dselect` wybierze aktualnie zainstalowane pakiety. Może Cię zapytać o dodatkowe pakiety, których nazwy weźmie z pól `depends`, `suggests` oraz `recommends`. Jeśli nie chcesz dodawać żadnych pakietów, wciśnij `Q`, by opuścić program `dselect`.

```
# dselect install
```

Będziesz jeszcze musiał odpowiedzieć na pytania stawiane przez konfiguratory pakietów podczas tego procesu, więc przygotuj swoje notatki i poświęć tej części trochę czasu. Zobacz też '`dselect`' na 85 stronie.

Używaj `dselect`. **To zawsze działa :).**

5.3.2 Niezalecany sposób aktualizacji przy użyciu `apt-get`

Użycie `apt-get`, które jest opisane poniżej, ma dużo większe możliwości, ale *nie* jest zalecane do aktualizacji systemu. Jeśli musisz aktualizować bez użycia `dselect`, rozważ skorzystanie z `aptitude` oraz z innych programów.

Jeśli w systemie znajduje się niewiele pakietów lub w archiwach Debiana nie zaszło dużo zmian, poniższe polecenia mogą być wystarczające.

```
# apt-get update # rób to zawsze przed aktualizacją
... by uaktualnić system z zależnościami ("‘depends"’):
# apt-get upgrade
 ... by uaktualnić całą dystrybucję wraz z zależnościami:
# apt-get -u dist-upgrade
 ... lub by uaktualnić i zastosować dotychczasowe ustawienia dselect (nowe, lepsze
# apt-get -u dselect-upgrade         # użyj ustawień dselect
```

Ponieważ ten sposób aktualizacji używa `apt-get`, wybieranie pakietów *zalecanych* i *sugerowanych* jest ograniczone. Zobacz ‘Zależności między pakietami’ na 16 stronie.

Rozdział 6

Zarządzanie pakietami Debiana

This chapter is based on older version of English original text. Please check the English version too.

By zmniejszyć obciążenie sieci na serwerach archiwów Debiana, upewnij się, że skonfigurowałeś lokalnego pośrednika HTTP (np. `squid`) dla pakietów ściąganych przez APT oraz, jeśli jest taka potrzeba, zmienną środowiskową `http_proxy` lub ustaw wartość `http` w pliku `/etc/apt/apt.conf`. To znacząco zwiększy wydajność uaktualnień przez sieć, szczególnie wtedy, gdy posiadasz wiele komputerów z Debianem w sieci lokalnej.

Pomimo tego, że możliwości przyszpilania w `apt_preferences(5)` są wręcz nieograniczone, nie rozwiązuje ono wszystkich problemów z zależnościami, ponieważ zazwyczaj wymagają one najnowszych wersji innych, podstawowych programów.

Użycie metody opisanej w '`chroot`' na 134 stronie jest pożądane do równoczesnego zabezpieczenia zarówno stabilności systemu jak i dostępu do najnowszych wersji oprogramowania.

Informacje opisane w tym rozdziale dotyczą systemu Woody, ale wiele z nich tyczy się również systemu Potato (z wyjątkiem `apt_preferences(5)` i tematów związanych z `/etc/apt/preferences`).

6.1 Wprowadzenie

Jeśli czytanie dokumentacji dla deweloperów jest dla ciebie zbyt ciężkim zajęciem, przeczytaj poniższy rozdział i zacznij się cieszyć prawdziwą mocą Debiana w wersji `testing/unstable` :-).

6.1.1 Podstawowe narzędzia

```
dselect   - narzędzie do zarządzania pakietami z menu
dpkg      - instaluje pakiet (z pliku)
apt-get   - instaluje pakiet (z archiwum, CLI APT)
```

```
tasksel   - instaluje zadania (zestawy pakietów)
aptitude  - instaluje pakiet (pakiet i zadanie, ncurses APT)
deity     - alternatywny interfejs ncurses dla APT
synaptic, gsynaptic - alternatywne graficzne interfejsy dla APT
```

Powyższe narzędzia nie działają na tym samym poziomie. `dselect` działa nad APT (polecenie `apt-get`) oraz `dpkg`.

APT używa plików `/var/lib/lists/*` do śledzenia dostępnych pakietów, podczas gdy `dpkg` używa pliku `/var/lib/dpkg/available`. Jeśli zainstalowałeś pakiety używając bezpośrednio `apt-get` lub podobnego programu (jak np. `aptitude`) upewnij się, że uaktualniłeś plik `/var/lib/dpkg/available` korzystając z opcji `[U]pdate` w menu `dselect` lub za pomocą polecenia „`dselect update`" przed wywołaniem `dselect select`, `tasksel` lub `dpkg -l`.

Jeśli chodzi o zależności między pakietami, `apt-get` automatycznie ściąga pakiety z jego **zależnościami** (**depends**), ale nie rusza pakietów podanych w polach **recommends** (zaleca) i **suggests** (proponuje), podczas gdy `dselect` pozwala wybierać między zalecanymi lub proponowanymi pakietami. `aptitude` pozwala na automatyczne ściągnięcie wszystkich pakietów z pól **depends**, **recommends** oraz **suggests**. Zajrzyj również do rozdziału 'Zależności między pakietami' na 16 stronie.

6.1.2 Wygodne narzędzia

```
apt-cache          - przeglądanie lokalnej kopii archiwum pakietów
dpkg-reconfigure   - ponowna konfiguracja już zainstalowanego pakietu
                    (jeśli korzysta z debconf)
dpkg-source        - obsługa pakietów źródłowych
dpkg-buildpackage - automatyzacja budowania pakietu
...
```

6.2 Podstawy zarządzania pakietami Debiana

Możesz instalować zestawy pakietów nazywane *zadaniami* lub instalować pojedyncze pakiety albo uaktualniać system przy użyciu narzędzi do zarządzania pakietami, które są opisane poniżej. Zajrzyj również na strony 'Wskazówki dotyczące instalacji Debiana' na 29 stronie, 'Uaktualnianie dystrybucji do wersji `testowej`' na 79 stronie oraz 'Edytory awaryjne' na 214 stronie.

6.2.1 Instalowanie *zadań* programem `tasksel` lub `aptitude`

`tasksel` to **Debian Task Installer** (instalator zadań Debiana), który podczas instalacji systemu jest dostępny jako jedna z „`prostszych`" opcji.

Jeśli chcesz zainstalować całe środowisko lub zestaw logicznie ze sobą powiązanych programów, które wymaga wielu pakietów, to ta droga jest jedną z najlepszych. Upewnij się, że wykonujesz polecenia w następujący sposób:

```
# dselect update
# tasksel
```

`aptitude` również daje dostęp do *zadań*. Pozwala on jednak nie tylko na ich wybór, ale również na rezygnację z instalacji poszczególnych pakietów z danego *zadania*.

6.2.2 Konfiguracja systemu APT

Jeśli spróbujesz poniższego opisu na mieszane środowisko, możesz doświadczyć pewnych konfliktów z zależnościami. Dobrym zwyczajem jest nie mieszanie różnych dystrybucji. Poniższy rozdział jest dla ludzi, którzy licząc się z ryzykiem chcą poeksperymentować.

Dla selektywnej aktualizacji podczas trzymania się ciągle dystrybucji `testowej`, system APT (>Woody) musi być skonfigurowany jak w 'Przejście do APT z Woodiego' na 79 stronie, by móc korzystać z opcji `apt_preferences(5)`.

Najpierw dodaj źródła dla `stable`, `testing` i `unstable` do pliku `/etc/apt/sources.list`, a następnie zmodyfikuj plik `/etc/apt/preferences` tak, by ustawić poprawne wartości Pin-Priority. [1]

```
Package: *
Pin: release a=stable
Pin-Priority: 500

Package: *
Pin: release a=testing
Pin-Priority: 600

Package: *
Pin: release a=unstable
Pin-Priority: 50
```

6.2.3 `dselect`

Zaraz po uruchomieniu `dselect` automatycznie wybiera pakiety oznaczone jako wymagane (required), ważne (important) oraz standardowe (standard). W Potato, niektóre duże objętościowo programy takie jak teTeX czy Emacs były również wybierane tutaj i najlepiej było je po

---

[1] Zostawiam poniższy opis dla spójności z innymi częściami. Jeśli naprawdę chcesz śledzić `testing` lub `unstable`, powinieneś usunąć wszelkie odsyłacze do `stable` w `/etc/apt/sources.list` i `/etc/apt/preferences`. Tak jest ponieważ `testing` jest ustawiana jako kopia `stable`.

prostu omijać (przez ręczne odznaczenie - wpisując '_') przy wstępnej instalacji. W Woodym, powyższe pakiety zostały przeniesione do kategorii „Optional".

`dselect` w pewnym sensie posiada dziwny interfejs użytkownika. Dostępne są cztery dwuznaczne polecenia (wielkość liter ma znaczenie):

Klawisz	Działanie
Q	Wyjście. Potwierdź obecny wybór i wyjdź (zastąp zależności)
R	Powrót! Nie chciałem tego zrobić.
D	Cholera! Zwisa mi co o tym myśli dselect. Po prostu wykonaj to!
U	Ustaw wszystko na stan "`sugerowane"'

Za pomocą `D` i `Q` możesz dokonać konfliktującego wyboru na własne ryzyko. Używaj tych poleceń z rozwagą. Obecnie `dselect` jest dojrzałym narzędziem, które daje Ci całkiem dużą kontrolę nad wyborem pomiędzy pakietami **sugerowanymi** a **rekomendowanymi**.

Jeśli chcesz zmniejszyć ilość pojawiających się ostrzeżeń i komunikatów, to dodaj opcję „expert" do pliku `/etc/dpkg/dselect.cfg`. Jeżeli posiadasz powolny komputer, to możesz uruchamiać `dselect` na innej, szybszej maszynie do wyszukiwania pakietów, a później możesz skorzystać z polecenia `apt-get install`, by je zainstalować.

`dselect` nie daje dostępu do pakietów, które nie są preferowane przez Pin-Priority.

6.2.4 `aptitude`

`aptitude` jest nowym programem do instalowania pakietów podobnym do `dselect`. Możesz go używać jako zamiennika polecenia tekstowego `apt-get`. Zobacz też `aptitude(1)`.

Wydawanie poleceń `aptitude` odbywa się za pośrednictwem poszczególnych klawiszy.

Klawisz	Działanie
F10	Menu
?	Ściągawka z klawiszologii
u	Uaktualnij informacje o archiwum pakietów
g	Ściągnij i zainstaluj wybrane pakiety
q	Wyjdź z danego ekranu i zapamiętaj zmiany
x	Wyjdź z danego ekranu i porzuć zmiany
Enter	Pokaż informacje o pakiecie

`aptitude` umożliwia ci automatyczne ściągnięcie wszystkich pakietów bazując na polach **depends**, **recommends** oraz **suggests**. Możesz zmienić to zachowanie wybierając z menu `F10 -> Opcje -> Zależności`.

`aptitude` daje Ci dostęp do wszystkich dostępnych wersji pakietów.

6.2.5 Polecenia `apt-cache` i `apt-get`

Podczas śledzenia dystrybucji `testing` jak zostało to opisane w powyższym przykładzie, możemy zarządzać pakietami przy użyciu poniższych poleceń:

- `apt-get -u upgrade`

  Uaktualnia wszystkie pakiety w systemie, wybierając zależności z dystrybucji `testing`.

- `apt-get -u dist-upgrade`

  Uaktualnia wszystkie pakiety w systemie, wybierając i rozwiązując zależności z dystrybucji `testing`.

- `apt-get -u dselect-upgrade`

  Uaktualnia wszystkie pakiety w systemie korzystając z wyboru dokonanego przez `dselect`.

- `apt-get -u install` *`pakiet`*

  Instaluje *pakiet* wraz z zależnościami z dystrybucji `testing`.

- `apt-get -u install` *`pakiet`*`/unstable`

  Instaluje *pakiet* z `unstable`, ale zależności pobiera z `testing`.

- `apt-get -u install -t unstable` *`pakiet`*

  Instaluje *pakiet* z `unstable` wraz z zależnościami z `unstable`, poprzez ustawienie Pin-Priority `unstable` na 990.

- `apt-cache policy` *`foo bar ...`*

  Sprawdza stan pakietów *foo bar ...*.

- `apt-cache show` *`foo bar ...`* `| less`

  Wyświetla informacje o pakietach *foo bar ...*.

- `apt-get install` *`foo=2.2.4-1`*

  Instaluje wybraną wersję *2.2.4-1* pakietu *foo*.

- `apt-get -u install` *`foo bar-`*

  Instaluje pakiet *foo* i usuwa pakiet *bar*

- `apt-get remove` *`bar`*

  Usuwa pakiet *bar*, ale zachowuje zmienione pliki konfiguracyjne.

- `apt-get remove --purge` *`bar`*

  Usuwa pakiet *bar* wraz ze wszystkimi jego plikami konfiguracyjnymi.

W powyższych przykładach opcja `-u` wymusza, by `apt-get` pokazał listę pakietów do aktualizacji i poczekał na dalsze decyzje użytkownika przez przystąpieniem do dalszych czynności. Poniższe działania ustawiają opcję `-u` jako domyślną:

```
$ cat >> /etc/apt/apt.conf << .
// Zawsze pokazuj aktualizowane pakiety (-u)
APT::Get::Show-Upgraded "true";
.
```

Użyj opcji `-s`, by zasymulować aktualizację.

6.2.6 Podążanie za smakiem dystrybucji Debian

W zależności od wyboru smaku dystrybucji, możesz zmienić przykładowy plik `/etc/apt/preference` z 'Konfiguracja systemu APT' na 85 stronie tak, by odpowiadał twoim potrzebom:

```
śledzenie stable:            zmień Pin-Priority testing na 50
śledzenie testing:           pozostaw bez zmian
śledzenie testing(unstable): zmień Pin-Priority unstable na 500
śledzenie unstable(testing): zmień Pin-Priority unstable na 800
```

Zasadą przy ustawianiu wartości Pin-Priority powinno być przewijanie wartości z góry na dół tabeli wraz z upływem czasu, od chwili po wydaniu dystrybucji, do czasu zamrażania przed wydaniem kolejnym.

Uwaga: śledzenie dystrybucji `testing` ma jedną wadę - poprawki bezpieczeństwa pojawiają się dosyć późno.

Jeśli mieszasz dystrybucje Debiana, `testing` z `stable` lub `unstable` ze `stable`, prawdopodobnie będziesz często nieświadomie pobierać kluczowe pakiety z `testing` lub `unstable`, które nie zawsze są najwyższej jakości. Miej to na uwadze.

Przykłady pliku `/etc/apt/preferences`, dzięki którym pobierane są kluczowe pakiety w dojrzałych wersjach, natomiast pakiety dodatkowe mogą być w wersjach niestabilnych, znajdują się w katalogu z przykładami (`http://www.debian.org/doc/manuals/debian-reference/examples/`) jako pliki `preferences.testing` i `preferences.unstable`. Z drugiej strony, `preferences.stable` wymusza, by wszystkie wersje pakietów zostały zmniejszone do wersji stabilnych.

6.2.7 Obniżenie wersji wszystkich pakietów do `stable`

By zmniejszyć wersję wszystkich pakietów do wydania `stable`, ustaw plik `/etc/apt/preferences` jak poniżej:

```
Package: *
Pin: release a=stable
Pin-Priority: 1001
```

a następnie uruchom „`apt-get upgrade`”, które wymusi zmniejszenie wersji z powodu Pin-Priority > 1000. Pamiętaj jednak, że podczas tej operacji mogą wystąpić problemy z zależnościami.

6.2.8 Przegląd pliku `/etc/apt/preferences`

W przykładowym pliku `/etc/apt/preferences` przedstawionym w ‘Konfiguracja systemu APT’ na 85 stronie, pierwsza linia, `Package: *`, oznacza, że ta regułka odnosi się do wszystkich pakietów. Możesz użyć konkretnej nazwy pakietu zamiast gwiazdki, by ustawić Pin-Priority dla danego pakietu.

Następna linia, `Pin: release a=stable`, oznacza, że `apt-get` pobiera plik `Packages.gz` z archiwum zawierającego plik `Release`, w którym znajduje się linia `Archive: stable`.

Ostatnia linia, `Pin-Priority: 500`, ustala Pin-Priority na 500.

Jeśli jest wiele pakietów o tej samej nazwie, to instalowany jest ten z najwyższą wartością Pin-Priority.

Poniżej znajdują się oznaczenia Pin-Priority:

- 1001 i więcej: priorytety zmniejszające wersję pakietu.

  Zmniejszanie wersji pakietu jest dozwolone dla tego przedziału Pin-Priority.
- 100 do 1000: normalne priorytety.

  Zmniejszanie wersji pakietu jest niedozwolone. Kilka kluczowych wartości Pin-Priority:
  - 990: priorytet ustawiony przez opcję `--target-release` lub `-t` programu `apt-get(8)`.
  - 500: priorytet dla wszystkich domyślnych plików pakietów.
  - 100: priorytet dla obecnie zainstalowanych pakietów.
- 0 do 99: nieautomatyczne priorytety. (są one używane tylko, jeśli pakiet nie jest zainstalowany i nie ma żadnej dostępnej wersji)
- mniej niż 0: wersja nigdy nie zostanie wybrana.

Ten sam efekt działania jak przy `--target-release` może być osiągnięty przez ustawienie pliku `/etc/apt/apt.conf` jak poniżej:

```
# echo 'APT::Default-Release "testing";' >> /etc/apt/apt.conf
```

Opcja lini poleceń `--target-release` oraz ustawienia w pliku `/etc/apt/apt.conf` zastępują ustawienia pliku `/etc/apt/preferences`. Miej to na uwadze podczas zabawy z plikiem `/etc/apt/preferences`.

6.3 Polecenia zapewniające przetrwanie w Debianie

Posiadając tę wiedzę, możesz cieszyć się **nieśmiertelnymi** aktualizacjami :-).

6.3.1 Sprawdzanie błędów w Debianie i poszukiwanie pomocy

Jeśli masz problemy z konkretnym pakietem, pamiętaj by najpierw sprawdzić poniższe strony zanim będziesz szukać pomocy lub zgłaszać błąd. (`lynx`, `links` oraz `w3m` działają tak samo dobrze):

```
$ lynx http://bugs.debian.org/
$ lynx http://bugs.debian.org/nazwa pakietu   # jeśli znasz nazwę pakietu
$ lynx http://bugs.debian.org/numer błędu     # jeśli znasz numer błędu
```

Szukaj z Google (http://www.google.com/) dołączając ciąg „site:debian.org".

Jeśli masz wątpliwości, to przeczytaj porządny podręcznik. Ustaw zmienną `CDPATH` jak poniżej:

```
export CDPATH=.:/usr/local:/usr/share/doc
```

i wpisz

```
$ cd nazwa pakietu
$ pager README.Debian # jeśli istnieje
$ mc
```

Więcej pomocnych zasobów podanych jest w rozdziale 'Rozwiązywanie problemów' na 251 stronie.

6.3.2 Problemy przy aktualizacji APT

Podczas aktualizacji w dystrybucji `niestabilnej/testowej` mogą wystąpić problemy z zależnościami między pakietami tak, jak to zostało opisane w rozdziale 'Aktualizacja systemu' na 81 stronie. W większości przypadków jest to spowodowane tym, że aktualizowany pakiet posiada nowe zależności, które nie są jeszcze spełnione. W takim przypadku, problem możesz rozwiązać używając

```
# apt-get dist-upgrade
```

Jeśli to nie działa, powtarzaj jedno z poniższych poleceń dopóki problem sam się nie rozwiąże:

```
# apt-get upgrade -f         # dalej aktualizuj nawet jeśli wystąpiły błędy
... lub
# apt-get dist-upgrade -f    # dalej aktualizuj całą dystrybucję nawet jeśli
                               wystąpiły błędy
```

Niektóre naprawdę zepsute skrypty aktualizujące mogą powodować trwałe problemy. Najlepiej w takim przypadku przyjrzeć się skryptom `/var/lib/dpkg/info/`*`nazwapakietu`*`.{post-,pre-}{install,removal}` dla danego pakietu, a następnie uruchomić:

```
# dpkg --configure -a    # konfiguruje wszystkie częściowo zainstalowane pakiety
```

Jeśli skrypt mówi o brakującym pliku konfiguracyjnym, poszukaj odpowiedniego pliku w katalogu `/etc`. Jeśli istnieje szukany plik z rozszerzeniem `.new` (lub podobnym), zmień nazwę tego pliku (za pomocą `mv`) tak, by usunąć rozszerzenie.

Problemy z zależnościami mogą też występować podczas instalacji w dystrybucji `unstable/testing`. Są jednak sposoby na obejście zależności.

```
# apt-get install -f pakiet # pomiń zepsute zależności
```

Innym sposobem na wyjście z tej sytuacji jest użycie pakietu `equivs`. Zobacz `/usr/share/doc/equivs/README.Debian` jak również ‘Pakiet `equivs`’ na 101 stronie.

6.3.3 Ratowanie przy użyciu `dpkg`

Doraźny ratunek zepsutego `dselect` (APT) może być przeprowadzony na całkowicie zepsutym systemie przy użyciu tylko `dpkg` bez APT:

```
# cd /var/cache/apt/archives
# dpkg -i libc6* libdb2* perl*
# dpkg -i apt* dpkg* debconf*
# dpkg -i *  # aż nie pojawi się żaden błąd
```

Jeśli brakuje jakiegoś pakietu, ściągnij go z serwerów lustrzanych (`http://www.debian.org/mirror/`) przy pomocy:

```
# mc            # użyj "‘FTP link"’ wskazując serwer FTP Debiana
```

Od niedawna, właściwe pakiety na serwerach HTTP/FTP mogą nie znajdować się w katalogu `/dist`, ale raczej w nowym `/pool`. (Zobacz też 'Katalog `pool`' na 9 stronie.)

Ściągnięty plik zainstaluj jak poniżej:

```
# dpkg -i /var/cache/apt/archives/plikpakietu.deb
```

W przypadku niespełnionych zależności, napraw je lub użyj poniższych poleceń:

```
# dpkg --ignore-depends=pakiet1,... -i plikpakietu.deb
# dpkg --force-depends -i plikpakietu.deb
# dpkg --force-depends --purge pakiet
# dpkg --force-confmiss -i plik pakietu.deb # zainstaluj brakujące
pliki konfiguracyjne
```

6.3.4 Przywracanie danych wyboru pakietów

Jeśli z jakiekogoś powodu plik `/var/lib/dpkg/status` zostanie uszkodzony, Debian straci dane wyboru pakietów i przez to stanie się nie do końca sprawny. Poszukaj starego pliku `/var/lib/dpkg/status` w `/var/lib/dpkg/status-old` lub `/var/backups /dpkg.status.*`.

Umieszczanie `/var/backups/` na osobnej partycji jest dobrym pomysłem, ponieważ katalog ten zawiera wiele ważnych dla systemu danych.

Jeśli stary plik `/var/lib/dpkg/status` nie jest dostępny, nadal możesz przywrócić informacje z katalogów w `/usr/share/doc/`.

```
# ls /usr/share/doc | \
  grep -v [A-Z] | \
  grep -v '^texmf$' | \
  grep -v '^debian$' | \
  awk '{print $1 " install"}' | \
  dpkg --set-selections
# dselect --expert # przeinstaluj system, odznacz, jeśli trzeba
```

6.3.5 Przywracanie systemu po awarii `/var`

Ponieważ katalog `/var` zawiera regularnie uaktualniane dane takie, jak poczta, jest on podatny na wszelkiego rodzaju uszkodzenia. Oddzielenie tego katalogu na innej, niezależnej partycji zmniejsza ryzyko wystąpienia problemów. Jeśli zdarzy się katastrofa, wtedy możesz przebudować katalog `/var`, by przywrócić sprawność systemu.

Najpierw zdobądź szkielet zawartości `/var` z minimalnego, działającego systemu Debian w tej samej lub starszej wersji, np. z `var.tar.gz` `(http://people.debian.org/~osamu/pub/)`, i umieść go w katalogu głównym zepsutego systemu. Następnie wykonaj poniżesze kroki:

```
# cd /
# mv var var-old      # jeśli jakieś przydatne dane pozostały
# tar xvzf var.tar.gz # użyj szkieletu Woody
# aptitude            # lub dselect
```

To powinno postawić system na nogi. Możesz przyspieszyć przywracanie wyborów pakietów używając sposobu opisanego w rozdziale 'Przywracanie danych wyboru pakietów' na sąsiedniej stronie. ([FIXME]: Ta procedura wymaga więcej testów do potwierdzenia poprawności.)

6.3.6 Instalowanie pakietu w niestartującym systemie

Uruchom Linuksa używając płyty lub dyskietki ratunkowej Debiana lub alternatywnej partycji w przypadku tzw. multi-bootu. Zobacz 'Uruchamianie systemu' na 111 stronie. Zamontuj partycję główną niestartującego systemu w `/target` i skorzystaj z `dpkg` w trybie instalacji w środowisku chroot.

```
# dpkg --root /target -i plikpakietu.deb
```

Następnie skonfiguruj pakiet i popraw wszelkie problemy.

Tak przy okazji - jeśli to `lilo` uniemożliwia start systemu, możesz go uruchomić używając standardowego dysku ratunkowego Debiana. W lini zachęty programu startującego wpisz (zakładając, że twoją główną partycją jest `/dev/hda12` i chcesz uruchomić system na poziomie startu (runlevel) 3):

```
boot: rescue root=/dev/hda12 3
```

W ten sposób wystartujesz prawie w pełni funkcjonalny system z jądrem ładowanym z dyskietki (mogą jednak wystąpić pewne problemy spowodowane brakiem niektórych funkcji jądra lub modułów).

6.3.7 Co robić, gdy polecenie `dpkg` nie działa?

Niedziałający `dpkg` może uniemożliwić instalację jakiegokolwiek pliku `.deb`. Poniższa procedura pomoże Ci naprawić tą sytuację (w pierwszej linijce zastąp „links" nazwą twojej ulubionej przeglądarki internetowej):

```
$ links http://http.us.debian.org/debian/pool/main/d/dpkg/
  ... ściągnij dobry plik dpkg_wersja_arch.deb
$ su
password: *****
# ar x dpkg_version_arch.deb
# mv data.tar.gz /data.tar.gz
# cd /
# tar xzfv data.tar.gz
```

Dla architektury `i386` jako adresu możesz użyć `http://packages.debian.org/dpkg`.

6.4 Polecenia prowadzące do nirwany w Debianie

Te polecenia uchronią człowieka od wiecznego cierpienia w piekle aktualizacji i pozwolą osiągnąć mu stan **nirwany**. :-)

6.4.1 Informacje o pliku

Aby odnaleźć nazwę pakietu, do którego należy dany plik wpisz:

```
$ dpkg {-S|--search} wzorzec # szukaj wg. wzorca w zainstalowanych pakietach
$ wget http://ftp.us.debian.org/debian/dists/sarge/Contents-i386.gz
$ zgrep -e pattern Contents-i386.gz
                             # znajdź wg. wzoru nazwę plików w archiwum Debiana
```

Możesz też użyć specjalnie do tego celu przeznaczonych narzędzi:

```
# apt-get install dlocate
$ dlocate nazwa_pliku          # szybsza wersja dpkg -L oraz dpkg -S
...
# apt-get install auto-apt   # narzędzie instalujące pakiet na życzenie
# auto-apt update            # tworzenie pliku bazy dla auto-apt
$ auto-apt search wzorzec
                             # szukaj wg. wzorca we wszystkich pakietach
```

6.4.2 Informacje o pakiecie

APT pozwala na przeszukiwanie i wyświetlanie informacji pobranych z archiwum pakietów. Upewnij się, że APT pobiera prawidłowe archiwa (plik `/etc/apt/sources.list`). Jeśli chcesz zobaczyć jaki jest stosunek pakietów z `testing/unstable` do tych zainstalowanych, użyj `apt-cache policy`.

```
# apt-get   check              # aktualizuje cache i sprawdza błędne pakiety
$ apt-cache search  wzorzec     # szuka pakietu wg. podanego wzorca (szuka w opisa
$ apt-cache policy  pakiet      # informacje o priorytecie i dystrybucji pakietu
$ apt-cache show -a pakiet      # pokazuje opis pakietu ze wszystkich dystrybucji
$ apt-cache showsrc pakiet      # pokazuje opis pakietu źródłowego
$ apt-cache showpkg pakiet      # informacje o pakiecie dla odpluskiwania
# dpkg  --audit|-C             # szuka częściowo zainstalowanych pakietów
$ dpkg {-s|--status} pakiet ... # pokazuje opis zainstalowanego pakietu
$ dpkg -l pakiet ...            # stan zainstalowanego pakietu (po jednej lini na
$ dpkg -L pakiet ...            # pokazuje listę plików zainstalowanych przez paki
```

`apt-cache showsrc` jest nieudokumentowane w Woodym, ale działa :).

Możesz też znaleźć informacje o pakiecie w (ja używam `mc` do przeglądania poniższych):

```
/var/lib/apt/lists/*
/var/lib/dpkg/available
```

Porównanie poniższych plików da Ci informacje o tym, co dokładnie działo się w ciągu ostatnich kilku sesji instalacyjnych.

```
/var/lib/dpkg/status
/var/backups/dpkg.status*
```

6.4.3 Niepilnowana instalacja przy użyciu APT

Żeby móc instalować pakiety bez potrzeby pilnowania przebiegu instalacji, dodaj poniższa linię do pliku `/etc/apt/apt.conf`:

```
Dpkg::Options {"--force-confold";}
```

Jest to równoważne uruchomieniu `apt-get -q -y` *`pakiet`*. Korzystaj z tej opcji bardzo ostrożnie, ponieważ sprawia ona, że automatycznie zostaje wybrana odpowiedź „tak” na każde pytanie, co może sprawić Ci wiele problemów. Przeczytaj `apt.conf(5)` oraz `dpkg(1)`.

Możesz skonfigurować dowolny pakiet później za pomocą techniki opisanej w rozdziale ‘Ponowna konfiguracja zainstalowanych pakietów’ na tej samej stronie.

6.4.4 Ponowna konfiguracja zainstalowanych pakietów

Poniższe polecenia ponownie konfigurują dowolny, już zainstalowany pakiet.

```
# dpkg-reconfigure --priority=medium pakiet [...]
# dpkg-reconfigure --all   # przekonfiguruj wszystkie pakiety
# dpkg-reconfigure locales # wygeneruj dodatkowe locale
# dpkg-reconfigure --p=low xserver-xfree86 # przekonfiguruj serwer X
```

Przekonfiguruj `debconf`, jeśli chcesz zmienić na stałe typ zapytań `debconf`.

Niektóre programy posiadają również własne skrypty konfiguracyjne.

```
apt-setup     - tworzy /etc/apt/sources.list
install-mbr   - instaluje menedżera Master Boot Record
tzconfig      - ustawia lokalną strefę czasową
gpmconfig     - konfiguruje demona myszy gpm
eximconfig    - konfiguruje Exim (MTA)
texconfig     - konfiguruje teTeX
apacheconfig  - konfiguruje Apache (httpd)
cvsconfig     - konfiguruje CVS
sndconfig     - konfiguruje system dźwięku
...
update-alternatives - ustawia domyślne polecenia, np. vim jako vi
update-rc.d         - zarządza skryptami startowymi System-V
update-menus        - system menu Debiana
...
```

6.4.5 Usuwanie i czyszczenie pakietów

Usunięcie pakietu pozostawiając jego konfigurację:

```
# apt-get remove pakiet ...
# dpkg  --remove pakiet ...
```

Usunięcie pakietu wraz z jego konfiguracją:

```
# apt-get remove --purge pakiet ...
# dpkg    --purge        pakiet ...
```

6.4.6 Wstrzymywanie starszych pakietów

Przykładowe wstrzymanie pakietów `libc6` oraz `libc6-dev` dla programu `dselect` oraz dla `apt-get -u upgrade` *`pakiet`* odbywa się następująco:

```
# echo -e "libc6 hold\nlibc6-dev hold" | dpkg --set-selections
```

Stan „hold" nie powstrzyma polecenia `apt-get -u install` *`pakiet`*. By wstrzymać pakiet przez wymuszanie automatycznego zmniejszenia wersji dla poleceń `apt-get -u upgrade` *`pakiet`* lub `apt-get -u dist-upgrade`, dodaj poniższe do pliku `/etc/apt/preferences`:

```
Package: libc6
Pin: release a=stable
Pin-Priority: 2000
```

Tutaj pole `Package:` nie może przyjmować wpisów takich jak `libc6*`. Jeśli musisz trzymać zsynchronizowane wersje wszystkich pakietów binarnych powiązanych z pakietem źródłowym `glibc` musisz je oddzielnie wypisać.

Poniższe polecenie wyświetli listę pakietów wstrzymanych:

```
dpkg --get-selections "*"|grep -e "hold$"
```

6.4.7 Mieszany system - `stable/testing/unstable`

`apt-show-versions` pokaże Ci listę dostępnych wersji pakietu wg. dystrybucji, z której pochodzi.

```
$ apt-show-versions | fgrep /testing | wc
... jak dużo posiadasz pakietów z testing
$ apt-show-versions -u
... lista pakietów do uaktualnienia
$ apt-get install `apt-show-versions -u -b | fgrep /unstable`
... aktualizacja wszystkich niestabilnych pakietów do ich najnowszych wersji
```

6.4.8 Wyczyść buforowane pliki pakietów

Instalacja pakietów przy użyciu APT pozostawia pliki pakietów w katalogu `/var/cache/apt/archives`, które muszą być usuwane by zwolnić miejsce.

```
# apt-get autoclean # usuwa tylko zbędne pliki pakietów
# apt-get clean     # usuwa wszystkie przetrzymywane pliki pakietów
```

6.4.9 Zapisanie/kopiowanie konfiguracji systemu

Poniższe polecenie tworzy lokalną kopię wyborów pakietów:

```
# debconf-get-selections > debconfsel.txt
# dpkg --get-selections "*" > myselections   # lub \* zamiast "*"
```

„`*`" sprawia, że w pliku *`myselections`* znajdą się również pozycje odnoszące się do pakietów ze stanem „purge".

Możesz skopiować ten plik na inny komputer i zainstalować go za pomocą poniższych poleceń:

```
# dselect update
# debconf-set-selections < debconfsel.txt
# dpkg --set-selections < myselections
# apt-get -u dselect-upgrade    # lub dselect install
```

6.4.10 Przeniesienie pakietu do systemu `stabilnego`

W przypadku częściowych uaktualnień systemu `stable`, pożądane jest przebudowywanie pakietów w jego własnym środowisku korzystając z pakietu źródłowego. Pozwala to uniknąć masowych uaktualnień spowodowanych ich zależnościami. Przede wszystkim dodaj poniższe linie do pliku `/etc/apt/sources.list`:

```
deb-src http://http.us.debian.org/debian testing \
 main contrib non-free
deb-src http://http.us.debian.org/debian unstable \
 main contrib non-free
```

W tym przykładzie każdy wpis `deb-src` został rozbity na dwie linie z powodu ograniczonego rozmiaru papieru podczas drukowania. Sam wpis w `sources.list` powinien składać się z jednej lini.

Przyszła kolej na ściągnięcie źródeł i zbudowanie lokalnego pakietu:

```
$ apt-get update # uaktualnij listę ze źródłami pakietów
$ apt-get source pakiet
$ dpkg-source -x pakiet.dsc
$ cd pakiet-wersja
  ... przejrzyj wymagane pakiety (Build-depends w pliku .dsc)
      i również je zainstaluj. Potrzebujesz także pakietu
      'fakeroot'.

$ dpkg-buildpackage -rfakeroot

  ...lub (bez sygnatury)
$ dpkg-buildpackage -rfakeroot -us -uc # jeśli trzeba, użyj później "`debsign"'

  ...Następnie zainstaluj pakiet
$ su -c "dpkg -i plikpakietu.deb"
```

Zazwyczaj należy też zainstalować kilka pakietow z końcówką „-dev”, by spełnić zależności pakietów. `debsign` znajduje się w pakiecie `devscripts`. `auto-apt` może pomóc przy spełnianiu tych zależności. Użycie `fakeroot` pozwala uniknąć niepotrzebnego korzystania z konta roota.

W Woodym, sprawy związane z zależnościami mogą być ułatwione. Przykładowo, by skompilować występujący tylko w formie źródłowej pakiet `pine`:

```
# apt-get build-dep pine
# apt-get source -b pine
```

6.4.11 Lokalne archiwum pakietów

Aby utworzyć zgodne z systemem APT i `dselect` lokalne archiwum pakietów, należy utworzyć plik `Packages`, a same pliki pakietów muszą być umieszczone w specjalnym drzewie katalogów.

Lokalne archiwum pakietów `deb` podobne do oficjalnego archiwum Debiana tworzy się w następujący sposób:

```
# apt-get install dpkg-dev
# cd /usr/local
# install -d pool # tutaj umieszczane są pakiety
# install -d dists/unstable/main/binary-i386
# ls -1 pool | sed 's/_.*$/ priority section/' | uniq > override
# editor override # zmień priority oraz section
# dpkg-scanpackages pool override /usr/local/ \
   > dists/unstable/main/binary-i386/Packages
# cat > dists/unstable/main/Release << EOF
Archive: unstable
Version: 3.0
Component: main
Origin: Local
Label: Local
Architecture: i386
EOF
# echo "deb file:/usr/local unstable main" \
   >> /etc/apt/sources.list
```

Można też użyć szybszego sposobu:

```
# apt-get install dpkg-dev
# mkdir /usr/local/debian
# mv /ścieżka/do/pakietu.deb /usr/local/debian
# dpkg-scanpackages /usr/local/debian /dev/null | \
  gzip - > /usr/local/debian/Packages.gz
#  echo "deb file:/usr/local/debian ./" >> /etc/apt/sources.list
```

Zdalny dostęp do utworzonych archiwów jest możliwy za pośrednictwem HTTP lub FTP. Należy wówczas odpowiednio ustawić wpisy w pliku `/etc/apt/sources.list`.

6.4.12 Zmiana lub instalacja obcego pakietu binarnego

`alien` pozwala na konwersję pakietów binarnych z formatów `rpm` Redhata, `slp` Stampede, `tgz` Slackware oraz `pkg` Solarisa na format pakietu `deb` Debiana. Jeśli chcesz użyć pakietu z innej dystrybucji Linuksa, możesz użyć programu `alien` do zmiany go na twój preferowany format i późniejszej instalacji. `alien` obsługuje również pakiety LSB.

6.4.13 Sprawdzanie poprawności zainstalowanych plików

`debsums` sprawdza poprawność zainstalowanych plików pakietów na podstawie sum kontrolnych MD5. Niektóre z pakietów nie posiadają sum MD5. W takim przypadku tymczasowym rozwiązaniem dla administratorów jest:

```
# cat >>/etc/apt/apt.conf.d/90debsums
DPkg::Post-Install-Pkgs {"xargs /usr/bin/debsums -sg";};
^D
```

wg. pomysłu Joerga Wendlanda <`joergland@debian.org`> (nieprzetestowane).

6.4.14 Zoptymalizowane `sources.list`

Wymyślne sposoby optymalizacji pliku `sources.list` nie spowodowały u mnie (w USA), znaczącej poprawy. Ręcznie wybrałem najbliższe serwery przy użyciu `apt-setup`.

`apt-spy` automatycznie tworzy `sources.list`, kierując się opóźnieniami i przepustowością. `netselect-apt` tworzy bardziej kompletny `sources.list`, ale używa gorszej metody do wybrania najlepszego serwera lustrzanego (porównanie czasów odpowiedzi na pingi).

```
# apt-get install apt-spy
# cd /etc/apt ; mv sources.list sources.list.org
# apt-spy -d testing -l sources.apt
```

6.5 Inne osobliwości Debiana

6.5.1 Polecenie `dpkg-divert`

Objazdy (diversions) są metodą zmuszenia `dpkg` w **zmienionej**, w stosunku do domyślnej lokalizacji. **Objazdy** mogą być używane przez skrypty pakietu Debiana do przeniesienia pliku, gdy ten powoduje konflikt. Administrator systemu może też użyć dywersji do zastąpienia pliku konfiguracyjnego pakietu lub gdy pewne pliki (nieoznaczone jako **conffiles**) muszą być zachowane przez `dpkg`, podczas instalacji nowej wersji pakietu je zawierającego (zobacz 'Ochrona lokalnych plików konfiguracyjnych' na 14 stronie).

```
# dpkg-divert [--add]  filename # dodaj "‘dywersję"’
# dpkg-divert --remove filename # usuń "‘dywersję"’
```

Zazwyczaj nie należy stosować `dpkg-divert`, gdy nie jest to absolutnie potrzebne.

6.5.2 Pakiet `equivs`

Jeśli własnoręcznie kompilujesz program, najlepiej jest uczynić go prawdziwym, lokalnym pakietem zdebianizowanym (`*.deb`). W ostateczności użyj `equivs`.

```
Package: equivs
Priority: extra
Section: admin
Description: Circumventing Debian package dependencies
 This is a dummy package which can be used to create Debian
 packages, which only contain dependency information.
```

6.5.3 Alternatywne polecenia

Użyj programu `update-alternatives`, jeśli chcesz, by polecenie `vi` uruchamiało program `vim`:

```
# update-alternatives --display vi
...
# update-alternatives --config vi
  Selection    Command
-----------------------------------------------
      1        /usr/bin/elvis-tiny
      2        /usr/bin/vim
*+    3        /usr/bin/nvi

Enter to keep the default[*], or type selection number: 2
```

Poszczególne pozycje w systemie alternatyw Debiana są trzymane jako dowiązania symboliczne w katalogu `/etc/alternatives`.

Jeśli chcesz ustawić swoje ulubione środowisko graficzne X, użyj `update-alternatives` na `/usr/bin/x-session-manager` oraz `/usr/bin/x-window-manager`. Szczegóły w 'Personalizowane sesje X.' na 155 stronie.

`/bin/sh` jest bezpośrednim dowiązaniem do `/bin/bash` lub `/bin/dash`. Bezpieczniej jest użyć `/bin/bash`, by zachować kompatybilność ze skryptami skażonymi starą składnią Basha. Lepiej jednak zdyscyplinować się do używania `/bin/dash`, w celu wymuszenia zgodności z normą POSIX. `/bin/dash`. Aktualizacja jądra Linuksa do wersji 2.4 skłania się do użycia `/bin/dash`.

6.5.4 System-V `init` i poziomy startu (runlevels)

W pliku `/etc/inittab` można ustawić domyślny poziom startu.

W przeciwieństwie do innych dystrybucji, Debian w pełni pozostawia zarządzanie poziomami startu administratori systemu. Zarządzanie skryptami `init` w stylu System-V w Debianie powinno odbywać się za pośrednictwem skryptu `update-rc.d`.

Poniżej znajduje się przykład polecenia, które tak ustawi skrypty startowe (skrypty `init`), by skrypt `/etc/init.d/`*`nazwa`* był uruchamiany na poziomach startu (runlevel) 1,2,3, a zatrzymywany na poziomach 4 i 5 z priorytetem równym 20 (normalnym).

```
# update-rc.d nazwa start 20 1 2 3 . stop 20 4 5 .
```

Usunięcie dowiązań symbolicznych z pozostawieniem skryptu w `init.d`:

```
# update-rc.d -f nazwa remove
```

Jeśli chodzi o edycję poziomów startu, nigdy nie używam narzędzi do tego celu. Poszczególne elementy zmieniam ręcznie za pomocą `mv` w lini poleceń `mc`, kopiując dowiązania za pomocą Alt-Enter. Przykład:

```
# mv S99xdm K99xdm # wyłącz xdm (X display manager)
```

Również wyłączam demony przez wstawienie `exit 0` na początku skryptu w `init.d`. Ot, takie szybkie rozwiązanie. W końcu są to pliki oznaczone jako `conffile`.

6.5.5 Wyłączone usługi demonów

Bezpieczeństwo systemu jest bardzo ważną rzeczą w Debianie, który zakłada, że administrator jest osobą kompetentną. Dlatego czasami łatwość użycia stoi na drugim miejscu, a wiele usług demonów domyślnie ma ustawiony najwyższy poziom bezpieczeństwa z włączonymi tylko nielicznymi usługami (lub nawet żadnymi).

Uruchom `ps aux` lub przejrzyj zawartość `/etc/init.d/*` oraz `/etc/inetd.conf`, jeśli masz jakiekolwiek wątpliwości (odnośnie Exima, DHCP, . . .). Sprawdź również opisany w ‘Ograniczenia dostępu poprzez PAM’ na 141 stronie plik `/etc/hosts.deny`. Pomocne może się też okazać polecenie `pidof` (patrz `pidof(8)`).

W ostatnich wersjach Debiana, X11 domyślnie nie zezwala na (zdalne) połączenia TCP/IP. Patrz też ‘Używanie X’ów poprzez TCP/IP’ na 158 stronie. Przekazywanie X przez SSH również jest wyłączone. Patrz ‘Zdalne połączenie z X – `ssh`’ na 159 stronie.

Rozdział 7

Jądro systemu Linux w Debianie

Debian posiada swoją własną metodę kompilacji jądra i powiązanych z nim modułów. Zobacz również 'Debian i kernel' na 25 stronie.

7.1 Kompilacja jądra

Użycie `gcc`, `binutils` i `modutils` z Debiana `unstable` może pomóc przy kompilacji najnowszych wydań jądra systemu Linux. Zobacz `/usr/share/doc/kernel-package/README.gz`, szczególnie koniec, aby uzyskać oficjalne informacje.

Od zawsze kompilacja jądra była trudnym tematem, który nawet najbardziej zaawansowanych developerów może wprowadzić w zakłopotanie:

Manoj Srivastava napisał: `--initrd` wymaga specjalnej łatki dla Debiana: cramfs.

Herbert Xu napisał: Nie, wcale nie, wszystko co musisz zrobić aby użyć innego systemu plików niż CRAMFS, to ustawienie MKIMAGE w `/etc/mkinitrd/mkinitrd.conf`.

Bądź ostrożny i zawsze polegaj na `/usr/share/doc/kernel-package/README.gz` napisanym przez Manoj i Kent'a. Upewnij się, że zdobyłeś najnowszą wersję pakietu unstable `kernel-package`, jeśli chcesz kompilować najnowszą wersję jądra.

`initrd` nie jest wymagane dla jądra kompilowanego tylko dla jednej maszyny. Używam tego od czasu, kiedy chcę aby moje jądro było niemal takie samo, jak to dostarczane przez pakiety kernel-image. Jeśli używasz `initrd`, powinieneś przeczytać `mkinitrd(8)` i `mkinitrd.conf(5)`. Zobacz również `http://bugs.debian.org/149236`.

7.1.1 Standardowa metoda Debiana

Śledź uważnie raporty o błędach w `kernel-package`, `gcc`, `binutils` i `modutils`. W miarę możliwości używaj ich nowych wersji.

Kompilacja spersonalizowanego jądra ze źródeł pod Debianem wymaga specjalnej uwagi. Użyj nowego `--append_to_version` wraz z `make-kpkg`, aby zbudować obrazy jądra wielokrotnego użytku.

```
# apt-get install debhelper modutils kernel-package libncurses5-dev
# apt-get install kernel-source-2.4.18    # użyj najnowszej wersji
# apt-get install fakeroot
# vi /etc/kernel-pkg.conf                # wpisz nazwisko i e-mail
$ cd /usr/src                            # stwórz katalog
$ tar --bzip2 -xvf kernel-source-2.4.18.tar.bz2
$ cd kernel-source-2.4.18                 # jeżeli są to źródła Twojego jądra
$ cp /boot/config-2.4.18-386 .config      # ustaw obecny plik konfiguracyjny jako
$ make menuconfig                        # wybierz potrzebne Ci opcje
$ make-kpkg clean                        # musi być uruchomione (per: man make-kpk
$ fakeroot make-kpkg --append_to_version -486 --initrd \
        --revision=rev.01 kernel_image \
        modules_image # modules_image używa się dla pcmcia-cs* itd.
$ cd ..
# dpkg -i kernel-image*.deb pcmcia-cs*.deb # instalacja
```

`make-kpkg kernel_image` właściwie wykonuje `make oldconfig` i `make dep`. Nie używaj `--initrd` jeżeli initrd nie jest używane.

Jeżeli ktoś chce używać modułów z pcmcia-cs lub bez pcmcia, powinien wybrać „General setup —>", następnie „PCMCIA/CardBus support —>" przy `make menuconfig` i ustawić konfigurację na „< > PCMCIA/CardBus support" (np. odznaczyć pole).

Na maszynie z SMP (Simetric Multi processing), ustaw CONCURRENCY_LEVEL zgodnie z `kernel-pkg.conf(5)`.

7.1.2 Metoda klasyczna

Pobierz źródła z:

- Linux: `http://www.kernel.org/`
- pcmcia-cs: `http://pcmcia-cs.sourceforge.net/`

lub użyj równoważnych źródeł z Debiana i postępuj następująco:

```
# cd /usr/src
# tar xfvz linux-whatever.tar.gz
# rm -rf linux
# ln -s linux-whatever linux
# tar xfvz pcmcia-cs-whatever.tar.gz
# ln -s pcmcia-cs-whatever pcmcia
# cd linux
# make menuconfig
... konfiguracja jądra ...
# make dep
# make bzImage
... edycja lilo / grub ...
```

```
... przeniesienie /usr/src/linux/arch/i386/boot/bzImage do  /boot ...
... uruchomienie /sbin/lilo lub równoważnego polecenia dla grub
# make modules; make modules_install
# cd ../pcmcia
# make config
# make all
# make install
... dodanie niezbędnych nazw modułów do /etc/modules
# shutdown -r now
... wystartowanie systemu z nowym jądrem ...
```

7.1.3 Pliki nagłówkowe jądra

Większość „zwykłych" programów nie potrzebuje plików nagłówkowych jądra, a nawet może się popsuć jeśli użyjesz ich bezpośrednio; zamiast tego powinny być skompilowane z plikami nagłówkowymi, z którymi **zbudowano/skompilowano `glibc`**, które są wersjami z `/usr/include/linux` i `/usr/include/asm` systemu Debian.

Nie twórz więc dowiązań symbolicznych do katalogów w `/usr/src/linux` z `/usr/include/linux` i `/usr/include/asm`, jak jest to sugerowane w pewnych przestarzałych dokumentach.

Jeżeli szczególnie **potrzebujesz** plików nagłówkowych jądra dla aplikacji zależnych od jądra, zmień plik/pliki Makefile, tak aby ich ścieżki do plików dołączanych (include) wskazywały na *`katalog-specjalnych-plików-nagłówkowych-jądra`*`/include/linux` i *`katalog-specjalnych-plików-nagłókowych-jądra`*`/include/asm`.

7.2 Zmodularyzowane jądro 2.4

Nowe jądra 2.4 w Debianie, dostarczone przez `kernel-image-2.4.`*`NN`* są bardzo zmodularyzowane. Musisz się upewnić czy moduły te są aktywne, aby jądro działało tak, jak zamierzałeś.

W następnej części, jako szybkie rozwiązania, prezentuję wiele przykładów pliku `/etc/modules`. Jednakże, lepszą metodą rozwiązywania problemów z modułami jest przypisanie ich, jako nazw zastępczych (aliasów) dla urządzeń, w plikach w `/etc/modutils/`. Niektóre moduły mogą być automatycznie aktywowane przez programy rozpoznające sprzęt, takie jak `discover`. Więcej na ten temat tutaj: 'Wykrywanie sprzętu dla X' na 151 stronie.

Zobacz `Documentation/*.txt` w źródłach Linuksa aby zdobyć szczegółowe informacje.

7.2.1 PCMCIA

`/etc/modules` musi zawierać następujące wpisy, aby urządzenia PCMCIA mogły działać:

```
# sterownik ISA PnP
isa-pnp
# Niskopoziomowy sterownik PCMCIA
# yenta_socket # w moim przypadku nie jest potrzebny
```

O resztę zadbają skrypty PCMCIA (z pakietu `pcmcia-cs`), `depmod` i `kmod`. Wydaje mi się, że potrzebowałem `isa-pnp` ponieważ mój laptop jest starym ISA-PCMCIA. Nowsze laptopy z CardBus/PCMCIA być może tego nie wymagają.

Zdanie wspaniałomyślnego Miquela van Smoorenburga `<miquels@cistron.nl>`:

„Po prostu wyrzuciłem całość pcmcia z laptopa w pracy, włącznie z cardmgr itd. Następnie zainstalowałem jądro 2.4 ze wsparciem dla cardbus, i nowy pakiet `hotplug` z woody'ego.

Jeśli masz 32-bitowe karty, nie potrzebujesz pakietu pcmcia ; 2.4 ma wbudowaną obsługę tych kart. Zwykły sterownik tulip powinien pracować dobrze z Twoją kartą dlink.

—Mike."

Zobacz: Linux PCMCIA HOWTO (`http://www.tldp.org/HOWTO/PCMCIA-HOWTO.html`) oraz 'Konfiguracja sieci i PCMCIA' na 206 stronie.

7.2.2 SCSI

[NIE SPRAWDZANE] `/etc/modules` musi zawierać następujące wpisy, aby urządzenia SCSI mogły działać:

```
# rdzeń SCSI
scsi_mod
# ogólny sterownik SCSI
sg
# dysk SCSI
sd_mod
# wszystkie inne niezbędne moduły
...
```

`depmod` zajmie się niektórymi z powyższych modułów.

7.2.3 Obsługa sieci

`/etc/modules` musi zawierać nastepujące wpisy dla działania dodatkowych funkcji sieciowych:

```
# net/ipv-4
ip_gre
```

```
ipip

# net/ipv-4/netfilter
# iptable (in order)
ip_tables
ip_conntrack
ip_conntrack_ftp
iptable_nat
iptable_filter
iptable_mangle
#
ip_nat_ftp
ip_queue
#
ipt_LOG
ipt_MARK
ipt_MASQUERADE
ipt_MIRROR
ipt_REDIRECT
ipt_REJECT
ipt_TCPMSS
ipt_TOS
ipt_limit
ipt_mac
ipt_mark
ipt_multiport
ipt_owner
ipt_state
ipt_tcpmss
ipt_tos
ipt_unclean
#
#ipchains
#ipfwadm
```

Powyższe wpisy mogą nie być zoptymalizowane. `depmod` zajmie się cześcią z wyżej wymienionych modułów.

7.2.4 System plików EXT3 (> 2.4.17)

Umożliwienie księgowania systemu plików z EXT3 FS przy użyciu Debianowego pakietu prekompilowanego obrazu jądra (> 2.4.17) wymaga następujących kroków:

```
# cd /etc; mv fstab fstab.old
```

```
# sed ’s/ext2/ext3,ext2/g’ <fstab.old >fstab
# vi /etc/fstab
... ustaw typ głównego systemu plików na "‘auto"’ zamiast "‘ext3,ext2"’
# cd /etc/mkinitrd
# echo jbd >>modules
# echo ext3 >>modules
# echo ext2 >>modules
# cd /
# apt-get update; apt-get install kernel-image-2.4.17-686-smp
... zainstaluj najnowsze jądro i ustaw bootowanie (uruchom tu lilo)
# tune2fs -j -i 0 /dev/hda1
# tune2fs -j -i 0 /dev/hda2
... i tak dla wszystkich systemów plików EXT2 konwertowanych na EXT3
# shutdown -r now
```

Teraz księgowanie EXT3 już działa. Używając `ext3,ext2` jako wpis „typ” w `fstab` zapewniasz bezpieczny powrót do EXT2, jeżeli jądro nie wspiera EXT3 dla partycji innych niż główna.

Jeżeli wcześniej zainstalowałeś sobie jądro 2.4 i nie chcesz go przeinstalowywać, wykonaj powyższe kroki aż do poleceń `apt-get`, następnie:

```
# mkinitrd -o /boot/initrd.img-2.4.17-686-smp /lib/modules/2.4.17-686-smp
# lilo
# tune2fs -j -i 0 /dev/hda1
# tune2fs -j -i 0 /dev/hda2
... dla wszystkich, którzy przekonwertowali EXT2 FS do EXT3
# shutdown -r now
```

Teraz księgowanie EXT3 już działa.

Jeżeli `/etc/mkinitrd/modules` nie zostało ustawione, podczas uruchomienia `mkinitrd` i chciałbyś dodać jakieś moduły w czasie startu to :

```
... przy zgłoszeniu initrd, aby uzyskać powłokę (5 sek.), wciśnij RETURN
# insmod jbd
# insmod ext3 # modprobe ext3 zajmie się wszystkim
# insmod ext2
# ^D
... kontynuuj uruchamianie.
```

Na ekranie uruchamiania pojawia się (`dmesg`), „cramfs: wrong magic”, ale jest to uznawane za nieszkodliwe. Ten problem został rozwiązany w Sarge (2002/10). Zobacz `http://bugs.debian.org/135537` i EXT3 File System mini-HOWTO (`http://www.zip.com.au/~akpm/linux/ext3/ext3-usage.html`) lub `/usr/share/doc/HOWTO/en-txt/mini/extra/ext3-mini-HOWTO.gz` w celu uzyskania dalszych informacji.

W niektórych systemach donoszone jest o doświadczaniu poważnych blokad jądra, gdy używane jest EXT3, ale ja nie miałem problemów (przynajmniej z 2.4.17).

7.2.5 Wsparcie układu Realtek RTL-8139 w jądrze 2.4

Z nieznanych przyczyn, wsparcie dla modułu RTL-8139 nie nazywa się już rtl8139, tylko 8139too. Po prostu wyedytuj swój `/etc/modules` by uwzględnić tę zmianę przy zmianie jądra 2.2 na 2.4.

7.2.6 Wsparcie dla portów równoległych

Dla `kernel-image-2.4.*`, wsparcie portu równoległego jest udostępniane jako moduł. Uaktywniasz je przez:

```
# modprobe lp
# echo lp >> /etc/modules
```

Zobacz `Documentation/parport.txt` w źródłach Linuksa.

7.3 Dostrajanie jądra przy pomocy systemu plików proc

Zachowanie jądra systemu Linux może być zmieniane „w locie" przy pomocy systemu plików proc.

Podstawowych informacji o zmianach parametrów jądra przez system plików `/proc` szukaj w źródłach Linuksa w `Documentation/sysctl/*`.

Zobacz kilka przykładów dla modyfikacji parametrów jądra w `/etc/init.d/networking` i 'Dziwne kłopoty z dostępem do witryn WWW.' na 46 stronie.

By dowiedzieć się jak ustawić konfigurację jądra poprzez system plików `/proc` podczas uruchamiania zobacz do `sysctl.conf(5)`. Zazwyczaj wykorzystywany jest do tego skrypt `/etc/init.d/procps.sh` uruchamiany z `/etc/rcS.d/S30procps.sh`.

7.3.1 Za dużo otwartych plików

Jądro Linuksa może zgłaszać problem „Za dużo otwartych plików". Spowodowane jest to małą domyślną wartością (8096) dla `file-max`. Aby naprawić ten problem, wywołaj następujące polecenie jako root:

```
# echo "65536"  > /proc/sys/fs/file-max  # dla jąder 2.2 i 2.4
# echo "131072" > /proc/sys/fs/inode-max # tylko dla jąder 2.2
```

lub by zmienić to na stałe umieść następujące wpisy w `/etc/sysctl.conf`:

```
file-max=65536   # dla jąder 2.2 i 2.4
inode-max=131072 # tylko dla jąder 2.2
```

7.3.2 Przedziały czasu między operacjami dyskowymi

Możesz zmienić przedziały czasu między operacjami dyskowymi przy pomocy systemu plików proc. Następująco skrócimy je z domyślnych pięciu sekund do sekundy.

```
# echo "40 0 0 0 100 30000 60 0 0"  > /proc/sys/vm/bdflush
```

Może to trochę negatywnie wpływać na osiągi pliku wejścia/wyjścia. Ale zabezpiecza to zawartość pliku, z wyjątkiem ostatniej sekundy, co i tak jest krótszym czasem, niż domyślne pięć sekund. Działa to nawet dla systemu plików z księgowaniem.

7.3.3 Stare, powolne maszyny z małą ilością pamięci

Dla niektórych starych systemów z małą ilością pamięci, użyteczne wciąż może być uruchomienie, przy pomocy systemu plików proc, przydzielania nieosiągalnej w danym momencie pamięci:

```
# echo 1 > /proc/sys/vm/overcommit_memory
```

Rozdział 8

Debianowe porady

8.1 Uruchamianie systemu

Zajrzyj do LDP BootPrompt-HOWTO (`http://www.tldp.org/HOWTO/BootPrompt-HOWTO.html`) w celu uzyskania szczegółowych informacji na temat boot prompt.

8.1.1 „Zapomniałem hasła roota!" (1)

Możliwe jest uruchomienie systemu i zalogowanie się na konto administratora (root) bez znajomości hasła, jeżeli tylko posiadamy dostęp do lokalnej konsoli (zakładając, że nie zostało założone hasło na BIOS lub program ładujący system taki jak `lilo`).

Poniżej przedstawiono procedurę postępowania w takim przypadku, która nie wymaga posiadania dodatkowych narzędzi, jak choćby dyskietek czy płyt z systemem, ani modyfikacji ustawień BIOSu. Zakładamy, że „Linux" to etykieta podstawowego jądra systemu w domyślnej instalacji Debiana.

Gdy tylko pojawi się ekran startowy `lilo` oraz napis `boot:` (być może musisz przytrzymać klawisz shift podczas uruchamiania `lilo` aby zapobiec automatycznemu startowi systemu) wpisz:

```
boot: Linux init=/bin/sh
```

To wymusza załadowanie jądra oraz uruchomienie `/bin/sh` w miejsce standardowego programu `init`. Dzięki temu uzyskujesz przywileje administratora oraz dostęp do konta root. Ponieważ system plików / jest zamontowany w trybie „tylko do odczytu", a inne nie są zamontowane w ogóle, trzeba wykonać kilka operacji, aby uzyskać w pełni funkcjonujący system.

```
init-2.03# mount -n -o remount,rw /
```

```
init-2.03# mount -avt nonfs,noproc,nosmbfs
init-2.03# cd /etc
init-2.03# vi passwd
init-2.03# vi shadow
```

(Jeżeli drugie pole danych zapisanych w `/etc/passwd` zawiera tylko literkę „x" dla każdego użytkownika, oznacza to, że Twój system używa osobnego pliku do przechowywania haseł i musisz edytować `/etc/shadow`.) Aby wyłączyć hasło dla konta root, wykasuj całą zawartość pola odpowiedzialnego za hasło (jest to drugie pole w pliku `/etc/passwd` lub `/etc/shadow`, jeżeli system go używa). Teraz możesz przeładować system i zalogować się na konto root bez hasła. Podczas inicjowania na poziomie startu równym 1 („runlevel 1"), Debian (od wersji Potato) wymaga hasła, czego niektóre starsze dystrybucje nie czynią.

Dobrym pomysłem jest mieć jakiś prosty edytor w `/bin/` na wypadek, gdyby `/usr/` było niedostępne (zobacz także 'Edytory awaryjne' na 214 stronie).

Możesz także rozważyć instalację pakietu `sash`. Jeśli system przestanie się uruchamiać, wykonaj:

```
boot: Linux init=/bin/sash
```

`sash` jest interaktywnym zamiennikiem dla `sh`, nawet gdy `/bin/sh` jest niezdatny do użytku. `sash` jest łączony statycznie i zawiera wbudowane wiele typowych funkcji (wpisz „help" w linii poleceń, aby poznać jego możliwości).

8.1.2 „Zapomniałem hasła roota!" (2)

Uruchom system z zestawu dyskietek ratunkowych lub z płyty CD. Załóżmy, że */dev/hda3* to oryginalna, główna partycja (root). Poniższe polecenia pozwolą na równie łatwe, jak w poprzednim punkcie, wyedytowanie plików z użytkownikami i hasłami.

```
# mkdir fixit
# mount /dev/hda3 fixit
# cd fixit/etc
# vi shadow
# vi passwd
```

Przewagą tej metody nad poprzednią jest to, że nie trzeba znać hasła do `lilo` (jeżeli jest takowe uaktywnione). Aczkolwiek wymaga to ingerencji w ustawienia BIOSu i włączenie możliwości startu systemu z CD czy też dyskietek.

8.1.3 System się nie uruchamia

To żaden problem, nawet jeśli zaniedbano przygotowawania dystkietki startowej. Jeśli `lilo` jest uszkodzone, użyj płyty startowej z zestawu płyt instalacyjnych Debiana (lub utwórz dyskietkę startową z obrazów zawartych na tejże płycie) i uruchom system przy jej pomocy.

Zakładamy, że główna partycja Twojej instalacji Linuksa to *`/dev/hda12`* oraz że chcesz ustawić 3 poziom startu.

Gdy pojawi się komunikat startowy, wpisz:

```
boot: rescue root=/dev/hda12 3
```

Po chwili otrzymujesz prawie w pełni funkcjonalny system wykorzystujący jądro z płytki/dyskietki startowej. Mogą jednak wystąpić pewne problemy wynikające z braku pewnych możliwości lub modułów w dystrybucyjnym jądrze.

Jeśli masz uszkodzony system zainteresuj się też rozdziałem 'Instalowanie pakietu w niestartującym systemie' na 93 stronie.

Jeżeli potrzebujesz specjalnie dostosowanej dyskietki startowej, przejrzyj plik `readme.txt` umieszczony na dysku ratunkowym.

8.1.4 Jak wyłączyć automatyczne uruchamianie Xów przy starcie?!

Ujarzmianie wersji `unstable/sid` może być zabawne, ale zawierający błędy `xdm`, `gdm`, `kdm` oraz `wdm` starujący przy uruchamianiu systemu może Ci dopiec.

Na początek zdobądź dostęp do konta administratora (root) wpisując poniższe polecenie podczas startu `lilo`:

```
boot: Linux vga=normal s
```

Zakładamy, że *Linux* to etykieta jądra, które zazwyczaj uruchamiasz. Natomiast „vga=normal" sprawi, że `lilo` wystartuje w normalnym trybie VGA. Ostatni parametr – „s" (lub „S") sprawia, że `init` uruchamia system w trybie pojedynczego użytkownika. Teraz wystarczy wpisać hasło administratora (konto root).

Istnieje kilka sposobów na wyłączenie wszystkich demonów systemu X:

- uruchom `update-rc.d ?dm stop 99 1 2 3 4 5 6 .`
- wstaw „exit 0" na początek wszystkich plików `/etc/init.d/?dm`.
- zmień nazwę wszystkich plików `/etc/rc2.d/S99?dm` na `/etc/rc2.d/K99?dm`.
- usuń wszystkie pliki `/etc/rc2.d/S99?dm`.
- uruchom `:>/etc/X11/default-display-manager`

W przykładach `rc2.d` oznacza aktualny poziom startu (runlevel) ustalony w `/etc/inittab`, zaś `?dm` oznacza, że powinieneś uruchomić każdy z programów: `xdm`, `gdm`, `kdm` czy też `wdm`.

Tylko pierwsze z rozwiązań wymienionych na liście, jest „jedynie słuszną drogą” w systemie Debian. Ostatni ze sposobów jest łatwy, ale wymaga ponownego ustawienia menedżera ekranu używając później `dpkg-reconfigure`. Pozostałe to różne metody na wyłączenie demonów.

Po tych czynnościach nadal możesz uruchomić środowisko graficzne poleceniem `startx` wydanym na dowolnej konsoli tekstowej.

8.1.5 Inne triki stosowane podczas startu systemu

System może być uruchamiany w wybranym poziomie startu (runlevel) oraz z określoną przez użytkownika konfiguracją. Wystarczy wykorzystać udostępnianą przez `lilo` możliwość podania parametrów przy uruchamianiu systemu. Szczegóły znajdziesz w BootPrompt-HOWTO (`http://www.tldp.org/HOWTO/BootPrompt-HOWTO.html`) (LDP).

Jeżeli chcesz uzyskać 4 poziom startu, spróbuj poniższego:

```
boot: Linux 4
```

Jeśli natomiast chcesz uzyskać tryb pojedynczego użytkownika (single-user), a znasz hasło administratora, możesz użyć:

```
boot: Linux S
boot: Linux 1
boot: Linux -s
```

W przypadku, gdy chcesz uruchomić system tak, aby wykorzystywał mniej pamięci, niż jest w rzeczywistości w komputerze (powiedzmy 48MB, podczas gdy komputer ma 64MB), użyj:

```
boot: Linux mem=48M
```

Upewnij się, że nie podajesz wartości przewyższającej dostępną pamięć, gdyż wtedy jądro może ulec katastrofie! Jeżeli posiadamy więcej niż 64MB (przykładowo 128MB), do czasu aż podamy `mem=128M` podczas startu lub w `/etc/lilo.conf`, starsze jądra i/lub płyty główne ze starszym BIOSem mogą nie używać pamięci powyżej 64MB.

8.1.6 Ustawianie parametrów uruchomieniowych GRUB

`GRUB` jest to nowy zarządca startu systemu wywodzący się z projektu GNU Hurd. Jest on bardziej elastyczny niż `lilo`, jednakże występują tu drobne różnice w zakresie manipulowania parametrami startu systemu.

```
grub> find /vmlinuz
grub> root (hd0,0)
grub> kernel /vmlinuz root=/dev/hda1
grub> initrd /initrd
grub> boot
```

Musisz być świadom nazw urządzeń pochodzących z Hurda:

```
HURD/GRUB           Linux               MSDOS/Windows
 (fd0)               /dev/fd0            A:
 (hd0,0)             /dev/hda1           C: (zazwyczaj)
 (hd0,3)             /dev/hda4           F: (zazwyczaj)
 (hd1,3)             /dev/hdb4           ?
```

Zajrzyj do `/usr/share/doc/grub/README.Debian.gz` oraz `/usr/share/doc/grub-doc/html/` po dokładniejsze informacje.

8.2 Rejestrowanie działań

8.2.1 Rejestrowanie poleceń powłoki

Administracja systemem pociąga za sobą o wiele więcej drobiazgowych zadań w Uniksie niż w zwykłych środowiskach komputerów osobistych. Upewnij się, że znasz przynajmniej podstawowe sposoby konfiguracji, na wypadek gdyby zaszła potrzeba odratowania systemu z problemów. Bazujące na graficznym interfejsie narzędzia konfiguracyjne wyglądają przyjemnie i wygodnie, jednakże zazwyczaj nie nadają się do użycia w stanie zagrożenia.

Tak więc, rejestrowanie działań poleceniami powłoki jest dobrym zwyczajem, szczególnie podczas pracy jako administrator (root).

Emacs: Użyj M-x `shell` aby rozpocząć nagrywanie do bufora oraz C-x C-w aby zapisać bufor na dysk.

Powłoka: Użyj polecenia `screen` oraz „^A H” jak opisano w ‘Zmiana konsoli przy pomocy `screena`’ na 130 stronie lub skorzystaj z `script`.

```
$ script nagranie
Script started, file is nagranie
$ ...
  ... Rób cokolwiek ...
  ... Na zakończenie naciśnij Ctrl+D ...
Script done, file is nagranie
$ col -bx <nagranie >raport
$ vi raport
```

Poniższe polecenie może być użyte zamiast polecenia `script`:

```
$ bash -i 2>&1 | tee nagranie
```

8.2.2 Rejestrowanie obrazu w Xach.

Jeżeli chcesz zarejestrować graficzny obraz aplikacji w środowisku X-Window, możesz użyć programu `gimp` (GUI). Potrafi on przechwytywać określone okna z osobna lub cały ekran. Innym rozwiązaniem jest zastosowanie programów takich jak: `xwd` (z pakietu `xbase-clients`), `import` (z pakietu `imagemagick`) oraz `scrot` (z pakietu `scrot`).

8.3 Kopiowanie oraz archiwizacja całych katalogów

8.3.1 Podstawowe polecenia do kopiowania całych podkatalogów

Jeżeli chcesz przearanżować strukturę położenia plików, skorzystaj na przykład z:

```
Typowa metoda:
# cp -a /początkowy/katalog /docelowy/katalog # wymaga narzędzia GNU cp
# (cd /początkowy/katalog && tar cf - . ) | \
        (cd /docelowy/katalog && tar xvfp - )
Jeżeli twarde dowiązania są wymagane, przydatna okaże się metoda:
# cd /ścieżka/do/starego/katalogu
# find . -depth -print0 | afio -p -xv -0a /punkt/montowania/nowego/katalogu
Jeśli cel jest na odległej maszynie:
# (cd /początkowy/katalog && tar cf - . ) | \
        ssh użytkownik@komputer.odległy (cd /docelowy/katalog && tar xvfp - )
Jeżeli nie ma dowiązań do plików, wystarczy:
# scp -pr użytkownik1@jeden.komputer:/początkowy/katalog \
          użytkownik2@drugi.komputer:/docelowy/katalog
```

W przykładach `scp` <==> `rcp` oraz `ssh` <==> `rsh`.

Powyższe sposoby kopiowania całych katalogów zostały zaprezentowane przez Manoja Srivastavę `<srivasta@debian.org>` na liście dyskusyjnej debian-user@lists.debian.org.

8.3.2 `cp`

Tradycyjne narzędzie `cp` nie było dobrym kandydatem do takich zadań, gdyż nie wyłuskiwało symbolicznych dowiązań ani nie chroniło dowiązań twardych. Kolejną rzeczą wymagającą rozważenia były tzw. rzadkie (sparse) pliki (pliki z dziurami).

GNU `cp` przezwyciężyło te ograniczenia, jednakże na systemach niezgodnych z GNU, `cp` może wciąż sprawiać problemy. Możesz również nie móc tworzyć małych, przenośnych archiwów używając `cp`.

```
% cp -a . newdir
```

8.3.3 `tar`

Tar przezwycięża kilka z problemów, które `cp` ma z symbolicznymi dowiązaniami. Jednakże, jakkolwiek `cpio` obsługuje specjalne pliki, to konwencjonalny `tar` nie.

W przypadku wielu dowiązań (hardlink) do jednego pliku, `tar` umieszcza na taśmie tylko jedną kopię; `cpio` umieszcza jedną kopię dla każdego dowiązania.

Polecenie `tar` zmieniło swoją opcję używaną dla plików `.bz2` pomiędzy Potato a Woody, więc używaj w skryptach `--bzip2` zamiast ich krótkiej formy `-I` (Potato) czy też `-j` (Woody).

8.3.4 `pax`

Nowe, wypasione narzędzie POSIX (IEEE Std 1003.2-1992, strony 380–388 (rozdział 4.48) oraz strony 936–940 (rozdział E.4.48)) o nazwie Portable ArchiveInterchange. `pax` potrafi czytać, zapisywać i wyświetlać zawartość pliku archiwum, jak również kopiować hierarchie katalogów. Operacje narzędzi `pax` są niezależne od określonego rodzaju archiwum i obsługują szeroką gamę różnorodnych formatów.

Implementacje programu `pax` są ciągle nowe i siermiężne.

```
# apt-get install pax
$ pax -rw -p e . nowy_katalog
 lub
$ find . -depth  | pax -rw -p e  nowy_katalog
```

8.3.5 `cpio`

`cpio` kopiuje pliki do lub z archiwum programu `cpio` albo programu `tar`. Archiwum może być plikiem na dysku, taśmą magnetyczną czy też potokiem.

```
$ find . -depth -print0 | cpio --null --sparse -pvd new-dir
```

8.3.6 `afio`

`afio` jest lepszym wyborem w zakresie obsługi archiwów w formacie `cpio`. Jest zdecydowanie szybszy niż `cpio`, udostępnia więcej opcji obsługi taśmy magnetycznej oraz zachowuje się o wiele wdzięczniej w przypadku uszkodzenia danych wejściowych. Obsługuje wieloczęściowe archiwa podczas interaktywnych operacji. `afio` może tworzyć kompresowane archiwa, które są o wiele bezpieczniejsze niż kompresowane produkty `tara` lub `cpio`. `afio` jest używane jako „archiwizujący silnik" w skryptach do tworzenia kopii zapasowych.

```
$ find . -depth -print0 | afio -px -0a new-dir
```

8.4 Kopia różnicowa oraz synchronizacja danych

Kopie różnicowe oraz synchronizacja danych mogą być wykonane różnymi metodami:

- `rcs`: kopia oraz historia, tylko tekst
- `rdiff-backup`: kopia oraz historia. symboliczne dowiązania .
- `pdumpfs`: kopia oraz historia w granicach systemu plików. symlink OK
- `rsync`: jednokierunkowa synchronizacja
- `unison`: dwukierunkowa synchronizacja
- `cvs`: wielotorowa synchronizacja z serwerem, kopia oraz historia, tylko tekst, dojrzałe rozwiązanie. Zobacz też tutaj: 'Concurrent Versions System (CVS)' na 219 stronie.
- `arch`: wielotorowa synchronizacja z serwerem, kopia oraz historia, brak takich rzeczy jak „roboczy katalog".
- `subversion`: wielotorowa synchronizacja z serwerem, kopia oraz historia, Apache.

Połączenie jednej z tych metod z archiwizacją opisaną tutaj: 'Kopiowanie oraz archiwizacja całych katalogów' na 116 stronie oraz automatycznym wykonywaniem zadań o określonym czasie ('Harmonogram działań (`cron`, `at`)' na 130 stronie) pozwoli stworzyć miły system kopii bezpieczeństwa.

Poniżej opiszę trzy łatwe w użyciu narzędzia.

8.4.1 Kopia różnicowa z użyciem rdiff

`rdiff-backup` pozwala przyjemnie i prosto utworzyć kopię bezpieczeństwa wraz z historią zmian, obsługując dowolne typy plików, włączając w to dowiązania symboliczne. Aby zarchiwizować `~/` do `/mnt/kopia`, zastosuj:

```
$ rdiff-backup --include ~/tmp/keep --exclude ~/tmp  ~/ /mnt/kopia
```

Aby przywrócić dane sprzed trzech dni do `~/stare`, użyj:

```
$ rdiff-backup -r 3D /mnt/kopia ~/stare
```

Poczytaj też podręcznik systemowy: `rdiff-backup(1)`.

8.4.2 Codzienna kopia z użyciem **`pdumpfs`**

`pdumpfs` to prosty system tworzenia codziennych kopii zapasowych, zbliony do `dumpfs`. Możesz korzystać z dostępu do wcześniejszych kopii w dowolnym momencie. Spróbuj uruchomić kopię bezpieczeństwa swojego katalogu domowego z wykorzystaniem `pdumpfs` oraz `crona`!

`pdumpfs` tworzy archiwum `YYYY/MM/DD` w docelowym katalogu. Przy pierwszym uruchomieniu `pdumpfs` wszystkie źródłowe pliki są kopiowane do docelowego katalogu. Za kolejnymi uruchomieniami kopiowane są jedynie te pliki, które zostały zmodyfikowane lub utworzone od ostatniego uruchomienia programu `pdumpfs`, natomiast pozostające bez zmian są przedstawiane w postaci dowiązania do ich odpowiedników we wcześniejszych kopiach celem zaoszczędzenia miejsca na dysku.

```
$ pdumpfs kat_źródłowy kat_docelowy [nazwa docelowa]
```

Zobacz także: `pdumpfs(8)`.

8.4.3 Regularne kopie różnicowe wykonywane z użyciem RCS

`Changetrack` regularnie rejestruje zmiany w tekstowych plikach konfiguracyjnych. Zobacz także: `changetrack(1)`.

```
# apt-get install changetrack
# vi changetrack.conf
```

8.5 Przywracanie po zawieszeniu systemu

8.5.1 Usuwanie procesów

Uruchom `top` aby zobaczyć, które procesy zużywają największą ilość zasobów. Naciśnij „P" aby posortować wg zużycia mocy procesora, „M" wg zużycia pamięci oraz „k" aby usunąć proces. Alternatywnie, możesz użyć `ps aux | less` (sposób BSD) lub `ps -efH | less` (sposób System-V). Ta druga składnia pokazuje identyfikatory procesów nadrzędnych. (`PPID`), co pozwala na ubicie zombie (niefunkcjonujących procesów potomnych).

Użyj polecenia `kill` aby usunąć proces (lub wysłać do niego sygnał) według ID procesu. `killall` działa analogicznie, ale możliwe jest podanie nazwy procesu. Najczęściej używane sygnały:

```
 1: HUP,  uruchom ponownie demona
15: TERM, normalne zakończenie pracy
 9: KILL, bezkompromisowe usunięcie
```

8.5.2 Alt-SysRq

Jądro pozwala się zabezpieczyć przed wadliwym działaniem systemu za pomocą opcji „Magic SysRq key" podawanej przy jego kompilacji. Wciśnięcie na architekturze i386 Alt-SysRq, a za nim jeden z klawiszy `r 0 k e i s u b` czyni cuda.

Klawisz 'r' przywróci stan klawiatury po takich niespodziankach jak zawieszenie się serwera X. Zmiana poziomu logowania na konsoli na '0' zmniejsza ilość wyświetlanych komunikatów o błędach. sa'k' (z ang. system attention key) zabija wszystkie procesy na danej konsoli wirtualnej. 'e' zabija wszystkie procesy na danym terminalu poza procesem init. 'i' zabija wszystkie procesy poza procesem init.

'S'ync, 'u'mount i re'b'oot służą do wychodzenia z najcięższych kłopotów.

W momencie powstawania tego dokumentu, domyślne jądro dystrybucyjne Debiana nie było kompilowane z tą opcją, musisz więc skompilować je samodzielnie, jeżeli chcesz ją aktywować. Szczegółowe instrukcje znajdziesz w pliku: `/usr/share/doc/kernel-doc-`*`version`*`/Documentation/sysrq.txt.gz` lub `/usr/src/`*`kernel-version`*`/Documentation/sysrq.txt.gz`.

8.6 Warte zapamiętania polecenia

8.6.1 Przeglądarka treści tekstowych

`less` jest domyślną przeglądarką zawartości plików. Klawisz „h" przywołuje pomoc. Program potrafi o wiele więcej niż `more`. `less` może być „podrasowany" przez wykonanie polecenia `eval $(lesspipe)` lub `eval $(lessfile)` w skryptach startowych powłoki. Zobacz więcej: `/usr/share/doc/lessf/LESSOPEN`. Opcja `-R` umożliwia wykorzystywanie nieobrobionego wyjścia oraz włącza sekwencje ANSI sterujące kolorami. Zobacz także: `less(1)`.

`w3m` może być użyteczny jako alternatywna przeglądarka w przypadku niektórych systemów kodowania (np. EUC).

8.6.2 Wolna pamięć

`free` oraz `top` stanowią dobry sposób uzyskania informacji na temat pamięci. Nie zwracaj uwagi na wartość „used" w linii „Mem:", ale skorzystaj z liczby umieszczonej poniżej (w przykładzie jest to 38792).

```
$ free -k # dla komputera z pamięcią o rozmiarze 256MB
             total       used       free     shared    buffers cached
Mem:        257136     230456      26680      45736     116136 75528
-/+ buffers/cache:      38792     218344
Swap:       264996          0     264996
```

Ścisły rozmiar fizycznej pamięci może być sprawdzony poleceniem `grep '^Memory' /var/log/dmesg`, które w przypadku z przykładu powyżej daje komunikat: „Memory: 256984k/262144k available (1652k kernel code, 412k reserved, 2944k data, 152k init)".

```
Total         = 262144k = 256M (1k=1024, 1M=1024k)
Free to dmesg = 256984k = Total - kernel - reserved - data - init
Free to shell = 257136k = Total - kernel - reserved - data
```

Około 5MB jest nieużyteczne dla systemu, ponieważ jądro tego używa.

8.6.3 Ustawianie czasu (BIOS)

```
# date MMDDhhmmCCYY
# hwclock --utc --systohc
# hwclock --show
```

Powyższe polecenia ustawiają sprzętowy zegar na wartość: `MM/DD hh:mm, CCYY`. Czas jest pokazywany zgodnie z lokalną strefą czasową, jednak sprzętowy zegar jest ustawiany według czasu uniwersalnego (UTC).

8.6.4 Ustawianie czasu (NTP)

Zaglądnij też do: Managing Accurate Date and Time HOWTO (`http://www.tldp.org/HOWTO/TimePrecision-HOWTO/index.html`).

Ustawianie czasu w przypadku stałego łącza z Internetem

Ustaw czas systemu automatycznie według odległego serwera:

```
# ntpdate serwer
```

Dobrym rozwiązanie jest umieszczenie tego w pliku `/etc/cron.daily/` jeżeli Twój system ma stałe łącze z Internetem, a dokładniej, z wskazanym serwerem.

Ustawianie czasu przy czasowych połączeniach z siecią

Użyj pakietu `chrony`.

8.6.5 Jak kontrolować funkcje konsoli, takie jak wygaszacz ekranu

W celu wyłączenia wygaszacza ekranu w konsoli tekstowej, należy wykonać:

```
# setterm -powersave off
```

Możesz uruchomić konsolę kon2 (kanji) używając:

```
# kon -SaveTime 0
```

Mając uruchomieny X Window System:

```
# xset s off
 lub
# xset -dpms
 lub
# xscreensaver-command -prefs
```

Przeczytaj odpowiednie strony podręcznika systemowego na temat obsługi innych funkcji konsoli. Zobacz także `stty(1)`. Znajdziesz tam nieco na temat sprawdzania oraz zmiany ustawień linii komend.

8.6.6 Przeszukiwanie administracyjnych baz danych

Glibc umożliwia wykorzystanie: `getent(1)` do wyszukiwania wpisów w administracyjnych bazach danych, takich jak passwd, group, hosts, services, protocols czy networks.

```
getent baza_danych [klucz ...]
```

8.6.7 Wyłączenia dźwięków (brzęczyk)

Najprościej odłączyć głośniczek ;-). Albo, dla basha:

```
echo "set bell-style none">> ~/.inputrc
```

8.6.8 Komunikaty o błędach na ekranie konsoli

Aby wyłączyć komunikaty o błędach wysyłane na konsolę tekstową, zaglądnij do `/etc/init.d/klogd`. Ustaw `KLOGD=„-c` *3”* w tym skrypcie oraz uruchom `/etc/init.d/klogd restart`. Inna metoda, to uruchomienie `dmesg -n`*3*.

Poziomy raportowania błędów są następujące:

- 0: KERN_EMERG, system jest bezużyteczny
- 1: KERN_ALERT, reakcja musi być natychmiastowa
- 2: KERN_CRIT, krytyczna sytuacja (critical conditions)
- 3: KERN_ERR, błędna sytuacja (error conditions)
- 4: KERN_WARNING, ostrzeżenie (warning conditions)
- 5: KERN_NOTICE, normalna, choć ważna sytuacja (significant condition)
- 6: KERN_INFO, informacyjne
- 7: KERN_DEBUG, komunikaty odpluskwiania

Jeśli denerwuje Cię jeden, specyficzny, bezużyteczny komunikat o błędzie, zastanów się nad zastosowaniem trywialnej łatki na jądro, takiej jak `shutup-abit-bp6` (dostępna w podkatalogu z przykładami (`http://www.debian.org/doc/manuals/debian-reference/examples/`)).

Możesz także zaglądnąć do `/etc/syslog.conf`; sprawdź czy jakieś komunikaty nie są wysyłane do urządzenia powiązanego z konsolą.

8.6.9 Ustawienie właściwego typu konsoli

Ekran konsoli w systemach uniksowych jest zazwyczaj obsługiwany przez bilioteki (n)curses. Zapewniają one niezależne od rodzaju terminala metody obsługi ekranu. Sprawdź: `ncurses(3X)` oraz `terminfo(5)`.

W systemie Debian, jest nieco wstępnie zdefiniowanych wartości:

```
$ toe | less                    # wszystkie wartości
$ toe /etc/terminfo/ | less    # wartości dające się modyfikować przez użytkowników
```

Wyeksportuj Twój wybór jako zmienną środowiskową `TERM`.

Jeżeli wartość terminfo dla `xterm` nie działa właściwie z xtermami nie pochodzącymi z Debiana, zmień rodzaj terminala z „xterm” na jedną z wersji posiadających ograniczenia funkcjonalności, jak np.„xterm-r6” podczas zdalnego logowania do Debiana. Zobacz także: `/usr/share/doc/libncurses5/FAQ`. „dumb” jest najniższą wspólną definicją dla terminfo.

8.6.10 Przywracanie konsoli do rozsądnego stanu

Jeśli ekran stanie się nieczytelny po wykonaniu polecenia `cat` *`jakiś-plik-binarny`* (możesz nie widzieć poleceń, jakie wpisujesz), zastosuj:

```
$ reset
```

8.6.11 Konwersja plików ze stylu dosowego na uniksowy

Przekształcanie dosowego pliku tekstowego (koniec-linii = `^M^J`) na format uniksowy (koniec-linii = `^J`):

```
# apt-get install sysutils
$ dos2unix plik_dosowy
```

8.6.12 Zastępowanie przy pomocy wyrażeń regularnych

Zamiana wszystkich wystąpień *WYRAŻENIE_POCZĄTKOWE* na *WYRAŻENIE_KOŃCOWE* we wszystkich plikach *PLIKI ...*:

```
$ perl -i -p -e 's/WYRAŻENIE_POCZĄTKOWE/WYRAŻENIE_KOŃCOWE/g;' PLIKI ...
```

`-i` służy do „in-place editing”, `-p` wymusza przetwarzanie w pętli, każdy plik z podanych jako *PLIKI ...* Jeśli zastępowanie jest złożone, możesz zabezpieczyć się przed błędami używając parametru `-i.bak` zamiast `-i`; to zachowa oryginalne pliki dodając do ich nazwy `.bak`.

8.6.13 Edytowanie plików przez skrypt

Poniższy skrypt usunie linie 5–10 oraz 16–20.

```
#!/bin/bash
ed $1 <<EOF
16,20d
5,10d
w
q
EOF
```

Polecenia `ed` są takie same jak polecenia `vi` w trybie wydawania poleceń. Modyfikacja pliku od tyłu ułatwia jego edycję w skryptach.

8.6.14 Odnajdywanie różnic i scalanie uaktualnień w pliku źródłowym

Poniższa procedura odnajduje różnice pomiędzy dwoma plikami i tworzy zunifikowane pliki „diff”: *plik.łatka0* lub *plik.łatka1*:

```
$ diff -u plik.stary plik.nowy1 > plik.łatka0
$ diff -u stary/plik nowy1/plik > plik.łatka1
```

Pliki „diff”, nazywane także łatkami (patch) są używane do przesyłania uaktualnień kodów źródłowych programów. Po odebraniu takiego pliku, można go zaaplikować na posiadany stary plik używając poleceń:

```
$ patch -p0 plik.stary < plik.łatka0
$ patch -p1 plik.stary < plik.łatka1
```

Jeśli masz wszystkie trzy wersje kodów źródłowych, możesz je połączyć używając narzędzia `diff3`:

```
$ diff3 -m plik.mój plik.stary plik.twój > plik
```

8.6.15 Przekształcanie dużego pliku w kilka małych

```
$ split -b 650m plik   # podział pliku na kawałki 650 MB
$ cat x* > dużyplik     # łączenie plików w jeden duży
```

8.6.16 Wyłuskiwanie danych z tekstowych tabel

Rozpatrzymy plik nazwany `DPL` w którym zachowywane są wszystkie nazwiska przywódców projektu Debian oraz daty rozpoczęcia ich działalności. Dane oddzielone są przerwami.

```
Ian     Murdock   August  1993
Bruce   Perens    April   1996
Ian     Jackson   January 1998
Wichert Akkerman  January 1999
Ben     Collins   April   2001
Bdale   Garbee    April   2002
Martin  Michlmayr March   2003
```

Najczęściej używanym narzędziem do takich zadań jest Awk.

```
$ awk '{ print $3 }' <DPL                   # Miesiące
August
April
January
January
April
April
March
$ awk '($1=="Ian") { print }' <DPL          # przywódca zwany Ian
Ian     Murdock   August  1993
Ian     Jackson   January 1998
$ awk '($2=="Perens") { print $3,$4 }' <DPL # Kiedy rozpoczął Perens
April 1996
```

Powłoka taka jak Bash również może zostać użyta do obrabiania tego typu plików:

```
$ while read first last month year; do
    echo $month
  done <DPL
... taki sam efekt jak w pierwszym przykładzie z Awk
```

W tym przypadku wbudowane polecenie `read` używa znaków z $IFS (wewnętrzne separatory pól) by podzielić linie na słowa.

Jeśli zmienisz IFS na „:” będziesz mógł bezproblemowo z powłoką obrabiać plik `/etc/passwd`:

```
$ oldIFS="$IFS"   # zapisuje starą wartość
$ IFS=":"
$ while read user password uid gid rest_of_line; do
    if [ "$user" = "osamu" ]; then
      echo "$user's ID is $uid"
    fi
  done < /etc/passwd
osamu's ID is 1001
$ IFS="$oldIFS"   # przywraca starą wartość
```

(Jeśli chcesz użyć Awk do tego samego skorzystaj z `FS=„:”` by ustawić separator pól.)

IFS jest również używane przez powłokę do rozdzielania wyników dopełniania parametrów, podmian poleceń i dopełniania arytmetycznego. To nie dotyczy wyrazów w podwójnym lub pojedynczym cudzysłowiu. Domyślną wartością IFS jest kombinacja następujących znaków <spacja>, <tabulator> i <znak nowej linii>.

Uważaj używając trików z IFS w powłoce. Dziwne rzeczy się mogą dziać kiedy powłoka zinterpretuje część skryptu jako swoje **wejście**.

```
$ IFS=":,"                        # użyj ":" i "," jako IFS
$ echo IFS=$IFS,   IFS="$IFS"     # echo jest wbudowane w Basha
IFS=  , IFS=:,
$ date -R                         # zwykłe wyjście polecenia
Sat, 23 Aug 2003 08:30:15 +0200
$ echo $(date -R)                 # pod-powłoka --> wejście do głównej powłoki
Sat  23 Aug 2003 08 30 36 +0200
$ unset IFS                       # przywrócenie domyślnej wartości IFS
$ echo $(date -R)
Sat, 23 Aug 2003 08:30:50 +0200
```

8.6.17 Kawałki kodu przekierowywania poleceń

Poniższe skrypty wspólnie tworzą piękną całość jeśli są połączone potokiem (ang. pipe).

```
find /usr | egrep -v "/usr/var|/usr/tmp|/usr/local"
                     # znajduje wszystkie pliki w /usr z wyłączeniem pewnych
plików
xargs -n 1 polecenie  # uruchamia polecenia dla każdego z elementu ze standardoweg
xargs -n 1 echo |    # dzieli elementy oddzielone białymi znakami na linie
xargs echo      |    # łączy wszystkie linie w jedną
grep -e wzorzec|     # wyciąga linie zawierające dany wzorzec
cut -d: -f3 -|       # wyciąga trzecie pole oddzielone znakiem : (np. plik passwd)
awk ’{ print $3 }’ | # wyciąga trzecie pole oddzielone białymi znakami
awk -F’\t’ ’{ print $3 }’ |        # wyciąga trzecie pole oddzielone tabulatorem
col -bx |            # usuwa backspace i rozwija tabulacje na spacje
expand -|            # rozwija tabulacje
sort -u|             # sortuje i usuwa powtórzenia

tr ’\n’ ’ ’|         # usuwa znaki LF (wysuń wiersz)
tr ’\r’ ''|          # usuwa znaki CR (powrót karetki)
tr ’A-Z’ ’a-z’|      # przekształca wielkie litery w małe
sed ’s/^/# /’|       # zmienia każdą linię na komentarz (zaczyna się od #)
sed ’s/\.ext//g’|     # usuwa .ext
sed  -n -e 2p|       # wypisuje drugą linię
head -n 2 -|         # wypisuje pierwsze 2 linie
tail -n 2 -|         # wypisuje ostatnie 2 linie
```

8.6.18 Skrypty działające w pętli na każdym pliku

Poniższe sposoby pętli działającej na plikach z rozszerzeniem `*.ext` zapewniają prawidłową obsługę dziwnych nazw plików na przykład ze spacjami:

- Pętla w powłoce:

```
for x in *.ext; do
  if test -f "$x"; then
    command "$x"
  fi
done
```

- `find` w kombinacji z `xargs`:

```
find . -type f -maxdepth 1 -name ’*.ext’ -print0 | \
 xargs -0 -n 1 command
```

- `find` z opcją `-exec` w poleceniu:

```
find . -type f -maxdepth 1 -name ’*.ext’ \
 -exec command ’{}’ \;
```

- `find` z opcją `-exec` i krótkim skryptem powłoki:

```
find . -type f -maxdepth 1 -name '*.ext' \
 -exec sh -c "command '{}' && echo 'successful'" \;
```

8.6.19 Szaleństwo krótkich skryptów Perla

Jakkolwiek każdy skrypt Awk może zostać automatycznie przekształcony do postaci perlowej z użyciem `a2p(1)`, to jednolinijkowe skrypty Awk najwygodniej przekształcać do jednolinijkowych skryptów Perla ręcznie. Przykładowo:

```
awk '($2=="1957") { print $3 }' |
```

może być zapisany jak poniżej:

```
perl -ne '@f=split; if ($f[1] eq "1957") { print "$f[2]\n"}' |
perl -ne 'if ((@f=split)[1] eq "1957") { print "$f[2]\n"}' |
perl -ne '@f=split; print $f[2] if ( $f[1]==1957 )' |
perl -lane 'print $F[2] if $F[1] eq "1957"' |
```

Teraz usuwamy wszystkie białe znaki oraz wykorzystujemy przewagę automatycznej konwersji pomiędzy liczbami a ciągami tekstowymi:

```
perl -lane 'print$F[2]if$F[1]eq+1957' |
```

Zobacz też do: `perlrun(1)`. Wiele równie, lub bardziej szalonych skrytów Perla znajdziesz tutaj: `http://perlgolf.sourceforge.net`.

8.6.20 Pobieranie tekstu oraz archiwów list dyskusyjnych ze stron WWW

Poniższe polecenia zczytają stronę WWW do pliku tekstowego. Przydatne na przykład do skopiowania konfiguracji ze strony WWW.

```
$ lynx -dump http://www.odległa-strona.pl/plik.html >plik.txt
```

`links` i `w3m` mogą być również analogicznie użyte, z dokładnością do niewielkich różnic.

Jeśli to jest archiwum listy dyskusyjnej, użyj `munpack` aby uzyskać zawartość mime.

8.6.21 Estetyczne wydruki stron WWW

Poniższe polecenia wydrukują stronę WWW do drukarki/pliku PostScript:

```
$ apt-get install html2ps
$ html2ps URL | lpr
```

Zobacz także '`lpr/lpd`' na 43 stronie oraz sprawdź pakiety: `a2ps` oraz `mpage` służące do tworzenia PostScriptu.

8.6.22 Estetyczne wydruki stron podręcznika systemowego

Poniższe polecenia wydrukują stronę podręcznika systemowego do drukarki/pliku PostScript:

```
$ man -Tps jakaś-strona-podręcznika | lpr
$ man -Tps jakaś-strona-podręcznika | mpage -2 | lpr
```

8.6.23 Łączenie dwóch plików w formacie PostScript lub PDF

Możesz połączyć pliki w formacie PostScript lub PDF:

```
$ gs -q -dNOPAUSE -dBATCH -sDEVICE=pswrite \
  -sOutputFile=bla.ps -f foo1.ps foo2.ps
$ gs -q -dNOPAUSE -dBATCH -sDEVICE=pdfwrite \
  -sOutputFile=bla.pdf -f foo1.pdf foo2.pdf
```

8.6.24 Czas trwania

Wyświetlanie czasu trwania danego polecenia.

```
# time jakieś-polecenie >/dev/null
real    0m0.035s       # czas, jaki upłynął na zegarku ściennym (czas rzeczywisty)
user    0m0.000s       # czas spędzony w trybie użytkownika
sys     0m0.020s       # czas spędzony w trybie jądra
```

8.6.25 Polecenie `nice`

Użyj polecenia `nice` (z pakietu GNU `shellutils`) by ustawić podczas startu „przyjazność" programu w stosunku do innych. `renice` (z `bsdutils`) lub `top` może zmienić tą wartość. Wartość równa 19 oznacza najniższy priorytet procesu. Wartość ujemna jest uważana za „nieprzyjazną" - wartość -20 oznacza wysoki priorytet procesu. Tylko superużytkownik może ustawiać ujemne wartości nice.

```
# nice  -19 top # bardzo "‘miły"’ dla innych procesów
# nice --20 cdrecord -v -eject speed=2 dev=0,0 disk.img # bardzo szybki
```

Czasami ekstremalne wartości nice sprawiają więcej kłopotu niż pożytku, dlatego używaj tego polecenia bardzo uważnie.

8.6.26 Harmonogram działań (`cron`, `at`)

Użyj pakietu `cron` lub `at` aby zaplanować zadania w systemie GNU/Linux. Zobacz: `at(1)`, `crontab(5)`,`crontab(8)`.

Wykonaj polecenie `crontab -e` edytujące zawartość spisu zadań („crontab”). Przykładowa zawartość takiego pliku:

```
# użyj /bin/sh do wykonywania poleceń, niezależnie od zapisu w /etc/passwd
SHELL=/bin/sh
# prześlij standardowe wyjście poleceń do ‘pawel’, niezależnie od tego,
# czyj zestaw poleceń jest wykonywany
MAILTO=pawel
# Min Godzina DzieńMiesiąca Miesiąc DzieńTygodnia polecenie
# (DzieńTygodnia i DzieńMiesiąca są połączone spójnikiem LUB)
# wykonaj o 00:05 każdego dnia
5  0  *  * *   $HOME/bin/daily.job >> $HOME/tmp/out 2>&1
# wykonaj o 14:15 pierwszego dnia każdego miesiąca -- wynik wysylany do Pawła
15 14 1  * *   $HOME/bin/monthly
# wykonaj o 22:00 każdego dnia tygodnia (1-5), wkurzamy Joe.
# % to nowa linia, ostatni % to cc:
0 22 *   * 1-5 mail -s "Jest 10 wieczór" joe%Joe,%%Czy wiesz, gdzie są twoje dziec
23 */2 1 2 *   echo "wykonane 23 minuty po godzinie 0, 2, 4 ..., dnia 1 lutego"
5  4 *   * sun echo "wykonane 04:05 każdej niedzieli"
# Wykonaj o 03:40 w pierwszy poniedziałek miesiąca
40 3 1-7 * *   [ "$(date +%a)" == "Mon" ] && command -args
```

Uruchom polecenie `at` aby wykonać jednorazowe zadanie:

```
$ echo 'polecenie -argumenty'| at 3:40 monday
```

8.6.27 Zmiana konsoli przy pomocy `screena`

Program `screen` umożliwia uruchomienie **wielu** pozornych terminali, każdy ze swoją własną powłoką, **na pojedynczym** fizycznym terminalu lub emulatorze terminala. Nawet jeśli używasz wirtualnych konsoli Linuksa lub wielu x-terminali, program `screen` jest wart zainteresowania z uwagi na bogaty zbiór możliwości, takich jak np.:

- przewijany dostęp do historii poleceń,
- kopiowanie i wklejanie,
- zapisywanie stanu wyjścia (logowanie),
- digraph entry oraz
- możliwość **odłączenia** sesji całego programu `screen` od terminala i ponowne jej przyłączenie później. Pozwala to na zdalne uruchomienie czasochłonnego zadania, rozłączenie i powrócenie do niego, gdy już się wykona.

Scenariusz zdalnego dostępu

Jeśli często podłączasz się do systemu ze zdalnego terminala lub korzystając z programu emulującego terminal w trybie VT100, `screen` uczyni Twoje życie łatwiejszym :-) dzięki opcji **detach**.

1 jesteś połączony przez łącze wdzwaniane i masz uruchomiony program `screen` ze sporą ilością sesji zawierających różne programy
2 nagle musisz porzucić terminal, ale nie chcesz stracić swojej dotychczasowej pracy po rozłączeniu
3 wpisujesz więc `Ctrl+A d` aby **odłączyć** sesje od terminala, a następnie przerywasz podłączenie do systemu. (lub szybciej: wpisując `Ctrl+A DD` zmuszając program `screen` jednocześnie do odłączenia sesji i zakończenia połączenia)
4 gdy ponownie przyłączysz się do systemu, wpisz polecenie `screen -r`, a `screen` magicznie **przyłączy się ponownie** i wszystkie sesje zostaną przywrócone.

Podstawowe polecenia `screena`

Gdy uruchomisz program `screen`, wszystkie naciskane klawisze są wysyłane do aktualnej sesji, za wyjątkiem kombinacji wywołującej system poleceń `screena`. Domyślnie jest to `Ctrl+A`. Wszystkie polecenia programu `screen` są wpisywane przy użyciu `Ctrl+A` a następnie pojedynczych klawiszy. Użyteczne polecenia:

`^A ?`	pomoc do programu
`^A c`	utwórz nową sesję i przełącz się do niej
`^A n`	następna sesja
`^A p`	poprzednia sesja
`^A` *0*	idź do sesji numer *0*
`^A w`	wyświetl listę sesji
`^A a`	Wyślij Ctrl+A do aktualnej sesji (normalne Ctrl+A jest przechwytywane)
`^A h`	zapisz kopię aktualnej sesji do pliku
`^A H`	rozpocznij/zakończ przechwytywanie aktualnej sesji do pliku
`^A ^X`	zabezpiecz terminal (ochrona hasłem)
`^A d`	odłącz sesje od terminala
`^A DD`	odłącz sesje od terminala, a terminal od systemu

To tylko niewielki podzbiór poleceń programu `screen`. Jeśli istnieje coś, co chciał(a)byś wykonać przy użyciu `screena` zapewne jest to możliwe!

Zobacz także: `screen(1)`.

Backspace i/lub Ctrl+H w sesji `screena`

Jeśli zauważysz, że backspace `i/lib` Ctrl+H nie działają dobrze podczas działania programu `screen`, wyedytuj plik `/etc/screenrc`, znajdź linię:

```
bindkey -k kb stuff "\177"
```

i zakomentuj ją (np. dodaj „#" jako pierwszy znak tej linii).

Zastępca `screena` w X Window System

Sprawdź `xmove`. Poczytaj `xmove(1)`.

8.6.28 Podstawy testowania sieci

Zainstaluj pakiety: `netkit-ping`, `traceroute`, `dnsutils`, `ipchains` (dla jądra 2.2), `iptables` (dla jądra 2.4), oraz `net-tools` i wykonaj:

```
$ ping www.google.com       # sprawdź połączenie z www.google.com
$ traceroute www.google.com # sprawdź trasę pakietów
$ ifconfig                  # sprawdź ustawienia komputera
$ route -n                  # sprawdź ustawienia tras pakietów
$ dig [@serwer-dns.pl] host.dom [{a|mx|any}] |less
      # sprawdź wpisy DNS dla host.dom zapisane w dns-server.com
      # typy rekorów to: {a|mx|dowolne}
$ ipchains  -L -n |less     # sprawdź filtry pakietów (jądro 2.2)
$ iptables -L -n |less      # check packet filter (jądro 2.4)
$ netstat -a                # znajdź wszystkie otwarte porty
$ netstat -l --inet         # znajdź nasłuchujące porty
$ netstat -ln --tcp         # znajdź nasłuchujące porty TCP (numerycznie)
```

8.6.29 Wypychanie przesyłek z lokalnej kolejki

Aby wypchnąć listy z lokalnej kolejki, możesz użyć:

```
# exim -q    # wypchnij czekające listy
# exim -qf   # wypchnij wszystkie listy
# exim -qff  # wypchnij nawet zamrożone przesyłki
```

`-qff` może być użyteczną opcją w skrypcie `/etc/ppp/ip-up.d/exim`.

8.6.30 Usuwanie zamrożonych przesyłek z lokalnej kolejki

Aby usunąć zamrożone przesyłki z lokalnej kolejki, wysyłając jednocześnie komunikat o błędzie, wykonaj:

```
# exim -Mg `mailq | grep frozen | awk '{ print $3 }'`
```

8.6.31 Ponownie dostarczanie zawartości mboksa

Jeśli Twój katalog domowy został zapełniony i procmailowi nie udało się dokonać swego dzieła, musisz ręcznie przenieść pocztę do posortowanych skrzynek w Twoim katalogu domowym z pliku `/var/mail/`*`nazwa_użytkownika`*. Po zwolnieniu miejsca w katalogu domowym wykonaj poniższe czynności:

```
# /etc/init.d/exim stop
# formail -s procmail </var/mail/nazwa_użytkownika
# /etc/init.d/exim start
```

8.6.32 Usuwanie zawartości plików

W celu usunięcia zawartości plików takich jak dzienniki systemowe, nie używaj polecenia `rm` do usunięcia pliku i nie twórz pustego od nowa, ponieważ pomiędzy tymi operacjami, jakaś część systemu może chcieć dokonać zapisków do takiego pliku. Poniżej pokazano sposób na bezpieczne usuwanie zawartości pliku:

```
$ :>plik-do-wyczyszczenia
```

8.6.33 Sztuczne pliki

Poniższe polecenia utworzą sztuczne lub puste pliki:

```
$ dd if=/dev/zero    of=nazwa_pliku bs=1k count=5 # 5KB samych zer
$ dd if=/dev/urandom of=nazwa_pliku bs=1M count=7 # 7MB losowej zawartości
$ touch nazwa_pliku #  pusty plik, jeśli nie istnieje
```

Przykładowo, poniższe polecenie, wykonane z powłoki dyskietki startowej systemu Debian usunie całkowicie zawartość dysku twardego `/dev/hda`:

```
# dd if=/dev/urandom of=/dev/hda ; dd if=/dev/zero of=/dev/hda
```

8.6.34 `chroot`

Program `chroot`, `chroot(8)`, pozwala na uruchomienie różnych instancji środowiska GNU/Linux na jednym systemie bez ponownego uruchomiania go.

Można też uruchamiać zasobożerne programy takie jak `apt-get` czy `dselect` w środowisku chroot na szybkim komputerze mając zamontowane na wolniejszym komputerze zasób NFS na prawach r/w, który to zasób jest głównym punktem montowania na wolniejszym komputerze.

Uruchamianie innej dystrybucji Debiana przy użyciu `chroot`

Debianowe środowisko chroot można łatwo stworzyć przy pomocy `debootstrap` w Woodym. Na przykład jeśli chcesz utworzyć chroota z Sidem w katalogu */sid-root* mając szybki dostęp do Internetu:

```
main # cd / ; mkdir /sid-root
main # debootstrap woody /sid-root http://ftp.debian.org/debian/
... obserwuj jak program ściąga całą podstawę systemu
main # echo "proc-sid /sid-root/proc proc none 0 0" >> /etc/fstab
main # mount proc-sid /sid-root/proc -t proc
main # cp /etc/hosts /sid-root/etc/hosts
main # chroot /sid-root /bin/bash
chroot # cd /dev; /sbin/MAKEDEV generic ; cd -
chroot # apt-setup # ustawianie /etc/apt/sources.list
chroot # vi /etc/apt/sources.list # wskaż źródła na unstable
chroot # dselect  # możesz też użyć aptitude, zainstaluj mc i vima :-)
```

W tej chwili powinieneś posiadać w pełni funkcjonalny system Debian, w którym możesz mieszać bez strachu, że coś popsujesz w Twojej głównej dystrybucji.

Ta sztuczka z `debootstrap` może też być użyta do instalcji Debiana bez użycia dysku instalacyjnego Debiana lecz przy wykorzystaniu innej dystrybucji GNU/Linux. Zobacz też `http://www.debian.org/releases/stable/i386/apcs04`.

Ustawianie loginu dla `chroot`

Wpisanie `chroot /sid-root /bin/bash` jest proste, ale zostawia to wiele zmiennych środowiskowych walających się po pamięci, a które nie są potrzebne lub nawet są niepożądane. Dużo lepszym sposobem jest uruchomienie kolejnego procesu login na oddzielnym terminalu wirtualnym, z którego możesz się bezpośrednio logować do systemu chroot.

Ponieważ domyślnie w systemie Debian terminale `tty1` do `tty6` są przeznaczone na konsole Linuksa, a `tty7` na sesję X Window System, użyjemy konsoli `tty8` jako konsoi chroot. Po utworzeniu środowiska chroot tak jak to zostało opisane w 'Uruchamianie innej dystrybucji Debiana przy użyciu `chroot`' na bieżącej stronie, wpisz będąc rootem w głównym systemie:

```
main # echo "8:23:respawn:/usr/sbin/chroot /sid-root "\
       "/sbin/getty 38400 tty8"  >> /etc/inittab
main # init q    # przeładowanie init
```

Ustawianie Xów do działania w `chroot`

Chcesz bezpiecznie uruchamiać najnowsze Xy i GNOME w środowisku chroot? Jest to możliwe! Poniższy przykład uruchomi GDM na wirtualnym terminalu `vt9`.

Najpierw zainstaluj system chroot wg. opisu w 'Uruchamianie innej dystrybucji Debiana przy użyciu `chroot`' na sąsiedniej stronie. Z głównego systemu skopiuj kluczowe pliki konfiguracyjne do systemu chroot.

```
main # cp /etc/X11/XF86Config-4 /sid-root/etc/X11/XF86Config-4
main # chroot /sid-root # lub użyj konsoli chroot
chroot # cd /dev; /sbin/MAKEDEV generic ; cd -
chroot # apt-get install gdm gnome x-window-system
chroot # vi /etc/gdm/gdm.conf # wykonaj s/vt7/vt9/ w sekcji [servers]
chroot # /etc/init.d/gdm start
```

Plik `/etc/gdm/gdm.conf` został tak zmieniony, by pierwszą konsolą była konsola `vt9`, a nie `vt7`.

Teraz możesz bez problemu przełączać się między funkcjonalnymi środowiskami X systemu głównego i systemu chroot przez proste przełączanie wirtualnych terminali Linuksa, np. za pomocą kombinacji klawiszy Ctrl-Alt-F7 i Ctrl-Alt-F9. Miłej zabawy!

[FIXME] Dodaj komentarz i dowiązanie do skryptu init z chrootowanego gdm.

Uruchamianie innych dystrybucji w środowisku `chroot`

Środowisko chroot dla innej dystrybucji można utworzyć całkiem prosto. Instalujesz system na oddzielnej partycji korzystając z instalatora danej dystrybucji, i wykonujesz poniższe polecenia (przy założeniu, że główna partycja innej dystrybucji to *`/dev/hda9`*):

```
main # cd / ; mkdir /other-dist
main # mount -t ext3 /dev/hda9 /other-dist
main # chroot /other-dist /bin/bash
```

Następnie postępuj tak jak opisano to w 'Uruchamianie innej dystrybucji Debiana przy użyciu `chroot`' na poprzedniej stronie, 'Ustawianie loginu dla `chroot`' na sąsiedniej stronie oraz 'Ustawianie Xów do działania w `chroot`' na bieżącej stronie.

Budowanie pakietu w środowisku `chroot`

`pbuilder` jest bardziej wyspecjalizowanym narzędziem przygotowującycm środowisko chroot i budującym w nim pakiet. Jest to idealny system sprawdzający, czy zależności budowania są poprawne oraz do upewnienia się, że niepotrzebne i niepoprawne zależności budowania nie znajdą się w pakiecie wynikowym.

8.6.35 Jak weryfikować dowiązania (hardlink)

Możesz sprawdzić czy dwa pliki są tym samym plikiem z dwoma dowiązaniami:

```
$ ls -li plik_1 plik_2
```

8.6.36 Montowanie obrazów systemów plików

Jeżeli `plik.img` zawiera obraz zawartości twardego dysku, zaś oryginalny twardy dysk ma konfigurację dającą: *xxxx* = (bajtów/sektor) * (sektorów/cylinder), wtedy możesz taki obraz zamontować do katalogu `/mnt` poleceniem:

```
# mount -o loop,offset=xxxx plik.img /mnt
```

Zauważ, że większość twardych dysków ma 512 bajtów/sektor.

8.6.37 Samba

Przykładowy dostęp do plików serwowanych przez Windows:

- montowanie zasobów Windows do Linuksa:
```
# mount -t smbfs -o username=moja_nazwa,uid=mój_uid,gid=mój_gid \
        //serwer/dzielony_zasób /mnt/smb
```
- inny sposób na montowanie zasobów Windows do Linuksa:
```
# smbmount //serwer/dzielony_zasób /mnt/smb \
        -o "username=moja_nazwa,uid=mój_uid,gid=mój_gid"
```
- lista udostępnianych zasobów na wskazanym serwerze:
```
# smbclient -L 192.168.1.2
```

Otoczenie sieciowe Samby może być sprawdzane spod GNU/Linuksa:

```
# smbclient -N -L adres_ip_twojego_komputera | less
# nmblookup -T "*"
```

8.6.38 Narzędzia do obsługi obcych systemów plików

Obsługa wielu obcych systemów plików jest wspierana obecnie przez linuksowe jądro, a więc mogą być one dostępne w prosty sposób, poprzez zamontowanie urządzeń je zawierających. Dla niektórych systemów plików są dostępne także specjalne narzędzia umożliwiające dostęp do nich bez montowania urządzeń. Są one dostępne jako programy pracujące w przestrzeni użytkownika, a zatem wsparcie ze strony jądra nie jest potrzebne.

- `mtools`: system plików MSDOS (MS-DOS, Windows)
- `cpmtools`: system plików CP/M
- `hfsutils`: system plików HFS (native Macintosh)
- `hfsplus`: system plików HFS+ (modern Macintosh)

Do tworzenia i weryfikacji systemu plików MS-DOS FAT użyteczne może być narzędzie: `dosfstools`.

8.7 Typowe błędy

Poniżej przedstawiamy kilka przykładów niebezpiecznych działań. Niebezpieczeństwo jest jeszcze większe, jeśli używasz konta uprzywilejowanego (`root`).

8.7.1 `rm -rf .*`

W poleceniu „`rm -rf .*`”, „`.*`” jest rozwijane w taki sposób, że zawiera także katalogi „`.`” oraz „`..`”. Jeśli więc posiadasz uprawnienia zapisu do katalogu nadrzędnego, to polecenie to usunie także wszystkie katalogi znajdujące się **obok** aktualnego katalogu.

- „`rm -rf .`” : usuwa wszystko w głąb bieżącego katalogu oraz również ten katalog.
- „`rm -rf *`” : usuwa wszystkie pliki i katalogi, poza rozpoczynającymi się znakiem kropki leżące w bieżącym katalogu.
- „`rm -rf .[^.]*`” : usuwa wszystkie pliki i katalogi, których nazwy rozpoczynają się znakiem kropki
- „`rm -rf .*`” : usuwa wszystko znadujące się w nadrzędnym katalogu oraz sam nadrzędny katalog.

8.7.2 `rm /etc/passwd`

Utrata kilku ważnych plików, takich jak `/etc/passwd` na skutek głupiej pomyłki bywa problematyczne. Debian wykonuje regularne kopie tych plików w `/var/backup/`. Kiedy przywrócisz te pliki, musisz ręcznie ustawić odpowiednie prawa.

```
# cp /var/backup/passwd /etc/passwd
# chmod 644 /etc/passwd
```

Przeczytaj także 'Przywracanie danych wyboru pakietów' na 92 stronie.

Rozdział 9

Dostrajanie systemu Debian

Ten rozdział opisuje tylko podstawy konfiguracji systemu, z użyciem narzędzi linii poleceń. Powinieneś najpierw przeczytać 'Wskazówki dotyczące instalacji Debiana' na 29 stronie.

Jeśli chcesz się dowiedzieć czegoś o bezpieczeństwie, przeczytaj Securing Debian Manual (`http://www.debian.org/doc/manuals/securing-debian-howto/`), który znajduje się w pakiecie `harden-doc`.

9.1 Inicjalizacja systemu

Debian używa systemu skryptów inicjalizacyjnych System V. Dobrym wprowadzeniem jest lektura 'Program `init`' na 23 stronie.

9.1.1 Dostosowywanie skryptów startowych do własnych potrzeb.

Najprostszym sposobem na kontrolowanie zachowania systemu inicjalizującego jest zmiana przypisań zmiennych środowiskowych w pliku nazwanym tak jak skrypt, a znajdującym się w katalogu `/etc/default/`. [1] Na przykład, `/etc/default/hotplug` może zostać użyty do kontrolowania pracy `/etc/init.d/hotplug`. `/etc/default/rcS` może zostać wykorzystany do dostrojenia ustawień dla `motd`, `sulogin`, itp. podczas startu systemu.

Jeśli zmieniając te zmienne nie możesz otrzymać pożądanego rezultatu, możesz zmodyfikować skrypt inicjalizujący; to również pliki konfiguracyjne.

---

[1] Pliki w `/etc/default/` zawierają **tylko** przypisania zmiennych środowiskowych. Każdy plik jest wchłaniany przez skrypt inicjalizujący odpowiedni dla niego w taki sposób, że te przypisania nadpisują jakiekolwiek domyślne ustawienia zmiennych środowiskowych w skrypcie. Wybór tego katalogu jest typowy (`http://lists.debian.org/debian-devel/2003/debian-devel-200308/msg02114.html`) dla systemu Debian. W gruncie rzeczy jest to odpowiednik `/etc/sysconfig` znanego z systemu RedHat i innych dystrybucji.

9.1.2 Logowanie komunikatów systemowych

Zachowanie programu logującego może być modyfikowane poprzez `/etc/syslog.conf`. Sprawdź pakiet `colorize` jeśli szukasz programu do kolorowania logów systemowych. Zobacz także `syslogd(8)` oraz `syslog.conf(5)`.

9.1.3 Optymalizacja sprzętu

Istnieje kilka ustawień sprzętowych, których optymalizację Debian pozostawia w gestii administratora.

- `hdparm`
  - Optymalizacja dostępu dysku twardego. Bardzo skuteczna.
  - Niebezpieczne. Przeczytaj `hdparm(8)` przed użyciem.
  - `hdparm -tT` *`/dev/hda`*, aby sprawdzić szybkość dostępu do dysku.
  - `hdparm -q -c3 -d1 -u1 -m16` *`/dev/hda`*, aby przyspieszyć działanie nowoczesnego dysku IDE. (Może być niebezpieczne.)

- `setcd`
  - Optymalizacja dostępu napędu płyt kompaktowych.
  - `setcd -x` *`2`*, aby zmniejszyć prędkość do 2x.
  - Zobacz `setcd(1)`.

- `setserial`
  - Zbiór narzędzi do konfiguracji portu szeregowego.

- `scsitools`
  - Zbiór narzędzi do konfiguracji urządzeń SCSI.

- `memtest86`
  - Zbiór narzędzi do testowania pamięci RAM.

- `hwtools`
  - Zbiór narzędzi do niskopoziomowej konfiguracji sprzętu.
    - `irqtune`: zmienia priorytet IRQ, aby zapewnić go tym urządzeniom, które wymagają wyższego priorytetu i szybkiego obsłużenia (np. porty szeregowe, modemy). Możliwe jest nawet trzykrotne zwiększenie przepustowości modemu/portu szeregowego.
    - `scanport`: skanuje adresy I/O od 0x100 do 0x3ff w celu znalezienia zainstalowanych urządzeń ISA.
    - `inb`: narzędzie odczytujące porty I/O i wyświetlające otrzymane wyniki w formacie binarnym i szesnastkowym.

- `schedutils`
  - Narzędzie do zmiany ustawień programu szeregującego jądra Linuksa (scheduler).
  - Zawiera `taskset`, `irqset`, `lsrt`, oraz `rt`.

  - W połączeniu z `nice` i `renice` (nie będącymi częścią tego pakietu), pozwalają na pełną kontrolę parametrów programu szeregującego.

Montowanie systemu plików z opcją `noatime` jest także bardzo skutecznym rozwiązaniem w przyspieszaniu odczytu plików. Zobacz `fstab(5)` i `mount(8)`.

Niektóre urządzenia mogą być konfigurowane bezpośrednio przez jądro Linuksa, poprzez system plików proc. Zobacz 'Dostrajanie jądra przy pomocy systemu plików proc' na 109 stronie.

W Debianie istnieje wiele narzędzi przeznaczonych dla konkretnego sprzętu. Wiele z nich związanych jest z laptopami. Oto lista interesujących pakietów dostępnych w Debianie:

- `tpconfig` - Program do konfiguracji urządzeń typu touchpad
- `apmd` - Konfiguracja zaawansowanego zarządzania energią (APM)
- `acpi` - Wyświetla informacje o urządzeniach z ACPI
- `acpid` - Narzędzie konfiguracji ACPI
- `lphdisk` - Przygotowuje partycję hibernacyjną dla Phoenix NoteBIOS
- `sleepd` - Usypia laptopa, gdy jest nieużywany
- `noflushd` - Pozwala bezczynnym dyskom na wolniejszy obrót
- `big-cursor` - Większy kursor w X-ach
- `acme` - Obsługa „multimedialnych przycisków" w laptopie
- `tpctl` - Narzędzie konfiguracji IBM ThinkPad
- `mwavem` - Obsługa modemów Mwave/ACP
- `toshset` - Umożliwia dostęp do większości ustawień sprzętowych laptopów Toshiba
- `toshutils` - Narzędzia dla laptopów Toshiba
- `sjog` - Program do obsługi „Jog Dial" na laptopach Sony Vaio
- `spicctrl` - Program do ustawiania parametrów ekranu LCD w laptopach Sony Vaio

ACPI jest nowszym systemem zarządzania energią niż APM.

Niektóre z wymienionych pakietów wymagają specjalnych modułów jądra. W większości wypadków są one dołączone do najnowszych źródeł jądra. Jednak w razie problemów będziesz musiał zainstalować najnowszą łatkę dla jądra.

9.2 Ograniczenia dostępu

9.2.1 Ograniczenia dostępu poprzez PAM

PAM (Pluggable Authentication Modules) pozwala kontrolować w jaki sposób użytkownicy logują się do systemu.

```
/etc/pam.d/*              # pliki konfiguracyjne PAM
/etc/pam.d/login          # plik konfiguracyjny PAM dla programu login
/etc/security/*           # ustawienia modułów PAM
/etc/securetty            # terminale, na których może pracować root (login)
/etc/login.defs           # ustawienia programu login (login)
```

Zmień zawartość `/etc/pam.d/login` tak, jak poniżej, jeśli chcesz aby dostęp do terminala (również konsoli) nie był warunkowany hasłem. Używaj na własną odpowiedzialność.

```
#auth       required   pam_unix.so nullok
auth       required   pam_permit.so
```

Podobne sztuczki mogą być zastosowane dla `xdm`, `gdm`, . . . , aby uzyskać analogiczną funkcjonalność w X-ach.

Z drugiej strony, jeśli chcesz wymusić na użytkownikach politykę wyboru bezpiecznych haseł, zainstaluj `cracklib2` i ustaw parametry w pliku `/etc/pam.d/passwd` następująco:

```
password required       pam_cracklib.so retry=3 minlen=6 difok=3
```

Możesz także, podczas aktywacji konta, ustawić hasło jednokrotnego użytku, co zmusi użytkownika do jego zmiany przy pierwszym logowaniu. Aby uzyskać taki efekt, użyj polecenia `passwd` z opcją `-e`. Zobacz `passwd(1)`.

Maksymalna ilość procesów może być ustalona poprzez `ulimit -u 1000` w powłoce Bash albo poprzez edycję `/etc/security/limits.conf`, elementu PAM. Inne limity, jak np `core` mogą być ustawione w sposób analogiczny. Początkowa wartość zmiennej `PATH` może być ustawiona w pliku `/etc/login.defs` zanim jeszcze zostaną uruchomione skrypty startowe danej powłoki.

Dokumentacja dla PAM jest dostępna w pakiecie `libpam-doc`.*Linux-PAM System Administrator's Guide* zawiera informacje dotyczące konfiguracji PAM, dostępnych modułów itp. Dokumentacja zawiera także *The Linux-PAM Application Developers' Guide* oraz *The Linux-PAM Module Writers' Guide.*

9.2.2 „Dlaczego GNU `su` nie obsługuje grupy `wheel`”

Jest to słynne zdanie w dokumentacji `info su` , napisane przez Richarda M. Stallmana. Ale nie obawiaj się, obecne `su` w Debianie używa PAM, więc można ograniczyć możliwość używania `su` do dowolnej grupy używając `pam_wheel.so` w pliku `/etc/pam.d/su`. Poniższy przykład ilustruje ustawienie grupy `adm` analogicznie do `wheel` z systemów BSD oraz umożliwia używanie `su` bez podawania hasła dla członków tej grupy.

```
# konfiguracja anty-RMS w /etc/pam.d/su
auth       required   pam_wheel.so group=adm

# Członkowie grupy wheel mogą wywoływać su bez podawania hasła
auth       sufficient pam_wheel.so trust group=adm
```

9.2.3 Przeznaczenie standardowych grup

Oto kilka interesujących grup:

- `root` grupa jest domyślnym odpowiednikiem grupy wheel dla `su`, jeśli `pam_wheel.so` jest użyte bez podania argumentu `group=`.
- `adm` członkowie grupy mogą czytać logi systemowe.
- `cdrom` grupa może zostać użyta, aby nadać użytkownikom prawa dostępu do napędu CD-ROM.
- `floppy` grupa może zostać użyta, aby nadać użytkownikom prawa dostępu do napędu dyskietek.
- `audio` grupa może zostać użyta, aby nadać użytkownikom prawa dostępu do urządzeń audio.
- `src` grupa jest właścicielem kodu źródłowego umieszczonego w `/usr/src`. Może być użyta, aby nadać użytkownikom możliwość zarządzania kodem źródłowym systemu.
- `staff` grupa służąca do nauki administracji systemem.

Aby uzyskać kompletną listę zobacz część FAQ w Securing Debian Manual (`http://www.debian.org/doc/manuals/securing-debian-howto/`), który w Woodym może być także znaleziony w pakiecie `harden-doc`. Także nowe `base-passwd` (>3.4.6) zawierają wiarygodną listę: `/usr/share/doc/base-passwd/users-and-groups.html`.

9.2.4 Bezpieczniejsza praca – `sudo`

Używam `sudo` do ochrony przed własną głupotą. Osobiście uważam, że używanie `sudo` jest lepszą alternatywą niż ciągła praca w systemie na koncie użytkownika root.

Zainstaluj `sudo` i skonfiguruj go poprzez edycję `/etc/sudoers(http://www.debian.org/doc/manuals/debian-reference/examples/)`. Zapoznaj się także z możliwościami `sudo` w pliku `/usr/share/doc/sudo/OPTIONS`.

Przykładowa konfiguracja zapewnia członkom grupy staff uprawnienia do wykonywania dowolnych poleceń jako root poprzez `sudo`, natomiast członkom grupy src prawo do wykonywania jedynie niektórych poleceń z takimi uprawnieniami.

Zaletą `sudo` jest to, że do zalogowania potrzebuje jedynie podania hasła użytkownika, a dodatkowo monitoruje aktywność użytkownika. Jest to dobry sposób, aby dać pewną władzę młodym administratorom. Na przykład:

```
$ sudo chown -R myself:mygrp .
```

Oczywiście, jeżeli znasz hasło root-a (jak większość domowych użytkowników), dowolne polecenie może zostać wydane jako root z Twojego konta użytkownika:

```
$ su -c "shutdown -h now"
Password:
```

(Wiem, że powinienem ograniczyć przywileje `sudo` konta admin, ale ponieważ jest to mój domowy serwer, jeszcze się tym nie przejmowałem.)

Inny program umożliwiający zwykłym użytkownikom uruchamianie poleceń z uprawnieniami root-a możesz znaleźć w pakiecie `super`.

9.2.5 Ograniczenia dostępu do usług

Internetowy *super-serwer*, `inetd`, jest uruchamiany podczas startu systemu z `/etc/rc2.d/S20inetd` (dla RUNLEVEL=2), który jest dowiązaniem symbolicznym do pliku `/etc/init.d/inetd`. W skrócie, `inetd` umożliwia uruchomienie jednego demona, który w razie potrzeby uruchamia inne, zmniejszając w ten sposób obciążenie systemu.

Gdy `inetd` przyjmuje zgłoszenie, sprawdza protokół i nazwę żądanej usługi przeszukując bazę danych w plikach `/etc/protocols` oraz `/etc/services`. Następnie, wyszukuje odpowiedzialną za obsługę zgłoszenia usługę w `/etc/inetd.conf` jeśli jest to zwyczajna usługa internetowa, natomiast w `/etc/rpc.conf` jeśli jest to usługa Sun-RPC.

Mając na względzie bezpieczeństwo, wyłącz zbędne usługi w pliku `/etc/inetd.conf`. Usługi Sun-RPC są wykorzystywane przez NFS oraz inne programy oparte na RPC.

Czasami `inetd` nie przekazuje kontroli do żądanej usługi bezpośrednio, lecz uruchamia demona TCP/IP `tcpd` przekazując mu nazwę usługi jako argument. W tym wypadku `tcpd` uruchomi żądaną usługę dopiero po zgłoszeniu tego do logów systemowych oraz zanalizowaniu plików `/etc/host.deny` i `/etc/host.allow`.

Jeśli masz problemy ze zdalnym dostępem w aktualnej dystrybucji Debian, skomentuj zapis „ALL: PARANOID” w `/etc/host.deny`, jeśli taki zapis się tam znajduje.

Aby uzyskać bardziej szczegółowe informacje zobacz `inetd(8)`, `inetd.conf(5)`, `protocols(5)`,`services(5)`,`tcpd(8)`,`hosts_access(5)`,i `hosts_options(5)`.

Aby uzyskać informacje o Sun-RPC zobacz `rpcinfo(8)`,`portmap(8)`,i `/usr/share/doc/portmap/portmapper.txt.gz`.

9.2.6 Centralne uwierzytelnianie – LDAP Użyj Lightweight Directory Access Protocol (LDAP).

Zasoby:

- OpenLDAP (`http://www.openldap.org/`)
- OpenLDAP Admin Guide w pakiecie `openldap-guide`
- LDP: LDAP Linux HOWTO (`http://www.tldp.org/HOWTO/LDAP-HOWTO/index.html`)
- LDP: LDAP Implementation HOWTO (`http://www.tldp.org/HOWTO/LDAP-Implementation-HOWTO/index.html`)
- OpenLDAP, extensive use reports (`http://portal.aphroland.org/~aphro/ldap-docs/ldap.html`)

- Open LDAP with Courier IMAP and Postfix (`http://alinux.washcoll.edu/docs/plc/postfix-courier-howto.html`)

9.3 Nagrywarki CD

Nagrywarki CD z interfejsem ATAPI/IDE stały się w ostatnim czasie bardzo popularne. Są miłym rozwiązaniem archiwizowania danych i wykonywania kopii zapasowych w sytuacji, gdy wystarcza < 640MB pojemności nośnika. Aby uzyskać najbardziej wiarygodne informacje, zobacz LDP CD-Writing-HOWTO (`http://www.tldp.org/HOWTO/CD-Writing-HOWTO.html`).

9.3.1 Wstęp

Po pierwsze, każde zakłócenie danych wysłanych do nagrywarki może spowodować nieodwracalne uszkodzenie płyty CD. Zaopatrz się w nagrywarkę z jak największym rozmiarem bufora. Jeśli pieniądze nie są problemem, zamiast wersji ATAPI/IDE zdecyduj się na wersję SCSI. Jeśli możesz wybierać pomiędzy IDE, zdecyduj się na wersję podłączaną do slotu PCI (na płycie głównej) zamiast do slotu ISA (jak karta SB16 itp.).

W jądrach Linuksa 2.2 i 2.4 nagrywarka IDE musi być uzupełniona sterownikami IDE-SCSI, zamiast zwykłego sterownika IDE dla napędów CD. Również ogólny sterownik SCSI (SCSI generic driver) musi zostać uaktywniony. Są na to dwa sposoby, zakładając że jądra rozpowszechniane były z aktualnymi dystrybucjami (jak te z marca 2001).

9.3.2 Sposób 1: moduły jądra + `lilo`

Dodaj poniższe linie do pliku `/etc/lilo.conf` jeśli używasz standardowego jądra rozpowszechnianego z systemem Debian. Jeśli użytych jest kilka opcji, pozostaw między nimi znak spacji.

```
append="hdx=ide-scsi ignore=hdx"
```

W tym przykładzie nagrywarka jest dostępna poprzez sterownik ide-scsi, a oznaczony parametrem `hdx`, gdzie *x* oznacza jedno z poniższych:

```
hda          master na pierwszym porcie IDE
hdb          slave na pierwszym porcie IDE
hdc          master na drugim porcie IDE
hdd          slave na drugim porcie IDE
hde ... hdh  napęd na zewnętrznym porcie IDE lub ATA66/100 IDE
```

Wpisz następujące polecenia jako root po zakończeniu konfiguracji.

```
# lilo
# shutdown -r now
```

9.3.3 Sposób 2: ponowna kompilacja jądra

Debian używa `make-kpkg` do tworzenia jądra. Użyj różnych `--append_to_version` z `make-kpkg` aby zbudować wiele obrazów jądra. Zobacz 'Jądro systemu Linux w Debianie' na 103 stronie.

Użyj następujących ustawień podczas `make menuconfig`:

- bzImage
- Wyłącz sterownik IDE CD, (niekoniecznie, ale to prostsze rozwiązanie)
- Dołącz do jądra obsługę ide-scsi i sg lub ustaw je jako moduły

9.3.4 Czynności po konfiguracji

Wsparcie jądra dla nagrywarki może być uaktywnione podczas startu systemu poprzez następujące polecenia:

```
# echo ide-scsi >>/etc/modules
# echo sg       >>/etc/modules
# cd /dev; ln -sf scd0 cdrom
```

Ręczna inicjalizacja może być wykonana przez:

```
# modprobe ide-scsi
# modprobe sg
```

Po restarcie możesz sprawdzić swoją konfigurację:

```
$ dmesg|less
# apt-get install cdrecord
# cdrecord -scanbus
```

[Per Warren Dodge] Czasami może wystąpić konflikt między `ide-scsi` i `ide-cd`, jeśli masz zarówno CD-ROM jak i napęd CD-R/RW. Możesz wtedy spróbować dodać poniższą linię do swojego `/etc/modutils/aliases`, uruchomić `update-modules` i zrestartować system.

```
pre-install      ide-scsi      modprobe ide-cd
```

Polecenie to nakazuje załadowanie sterownika IDE przed `ide-scsi`. Sterownik IDE `ide-cd` przejmuje kontrolę nad urządzeniem ATAPI CD-ROM — wszystkim, co nie zostało mu nakazane do **zignorowania**. W ten sposób `ide-scsi` przejmuje kontrolę tylko nad zignorowanymi urządzeniami.

9.3.5 Obrazy startowe płyt CD

Aby nagrać katalog `target-directory` jako `cd-image.raw` (startowy, Joliet z TRANS.TBL; jeśli płyta nie ma być bootowalna, usuń opcje `-b` i `-c`), włóż dyskietkę startową do stacji dyskietek i:

```
# dd if=/dev/fd0 target-directory/boot.img
# mkisofs -r -V volume_id -b boot.img -c bootcatalog
-J -T \ -o cd-image.raw target_directory/
```

Śmiesznym kruczkiem może być zrobienie startowego CD-ROM z systemem DOS. Jeśli obraz zwyczajnej dyskietki startowej jest we wspomnianym pliku *boot.img*, przy restarcie płyta zadziała jak gdyby dosowa dyskietka startowa była w napędzie dyskietek. Może być to jeszcze ciekawsze, jeśli wykorzystasz freeDOS.

Zawartość obrazu tej płyty może być sprawdzona poprzez zamontowanie jej jako urządzenie loop.

```
# mount -t iso9660 -o ro,loop cd-image.raw /cdrom
# cd /cdrom
# mc
# umount /cdrom
```

9.3.6 Wypalanie płyty (R, RW):

Najpierw porównajmy (zakładając podwójną prędkość napędu) z:

```
# nice --10 cdrecord -dummy speed=2 dev=0,0 disk.img
```

Następnie, jeśli wszystko poszło OK, wypalimy płytę CD-R:

```
# nice --10 cdrecord -v -eject speed=2 dev=0,0 disk.img
```

Albo CD-RW:

```
# nice --10 cdrecord -v -eject blank=fast speed=2 dev=0,0 disk.img
```

Niektóre napędy CD-RW działają lepiej z:

```
# nice --10 cdrecord -v blank=all speed=2 dev=0,0 disk.img
```

i następnie:

```
# nice --10 cdrecord -v -eject speed=2 dev=0,0 disk.img
```

Te dwa kroki są niezbędne, aby zapobiec pokrywaniu się faz zerowania płyty i wypalania. Wartość argumentów dla `nice` może także wymagać pewnej regulacji.

9.3.7 Wykonanie obrazu z posiadanej płyty CD

Niektóre płyty CD-R i komercyjne CD zawierają specjalne, błędne sektory, które uniemożliwiają ich kopiowanie przez `dd` (płyta Windows 98 jest jedną z nich). Pakiet `cdrecord` udostępnia polecenie `readcd`. Użyj go do skopiowania zawartości płyty do pliku. Jeśli to plik z danymi, zamontuj go i wykonaj `df` aby uzyskać jego rzeczywisty rozmiar. Podziel liczbę bloków przez 2 (1 blok = 1024 bajty) aby uzyskać ilość sektorów na CD (1 sektor = 2048 bajty). Uruchom `readcd` z niezbędnymi opcjami i użyj obrazu płyty do wypalenia następnej.

```
# readcd dev=target,lun,scsibusno # select function 11
```

Zwykle można ustawić wszystkie trzy parametry jako 0. Ilość sektorów podana przez `readcd` jest zazwyczaj zbyt duża, użyj podanej powyżej metody aby uzyskać dokładniejsze wyniki.

Powinno zostać zauważone, że użycie `dd` z płytami CD-ROM stwarza pewne problemy. Pierwsze uruchomienie `dd` spowoduje komunikat błędu i może wytworzyć krótszy obraz dysku, utraciwszy końcową jego część. Drugie uruchomienie `dd` może utworzyć większy obraz płyty ze „śmieciami" dołączonymi na końcu, jeśli rozmiar danych nie został podany. Tylko drugie uruchomienie `dd` wraz z podaniem poprawnego rozmiaru danych, bez wyciągania CD, po otrzymaniu komunikatu błędu wydaje się działać poprawnie. Jeśli na przykład rozmiar obrazu płyty wyświetlony przez `df` wynosi `46301184` bloki, użyj dwukrotnie następującego polecenia aby otrzymać prawidłowy obraz (jest to informacja, uzyskana przeze mnie empirycznie):

```
# dd if=/dev/cdrom of=cd.img bs=2048 count=$((46301184/2))
```

9.3.8 Obrazy Debian CD

Aby otrzymać najnowsze informacje na temat płyt z Debianem, odwiedź stronę Debian CD (`http://www.debian.org/CD/`).

Jeśli posiadasz szybkie połączenie internetowe, pomyśl o instalacji poprzez sieć, używając:

- kilku obrazów dyskietek (`http://www.debian.org/distrib/floppyinst`).
- małego startowego obrazu CD (`http://www.debian.org/CD/netinst/`).

Jeśli nie posiadasz szybkiego połączenia z Internetem, pomyśl o zakupie płyt CD od dystrybutora CD (`http://www.debian.org/CD/vendors/`).

Proszę, nie zajmuj niepotrzebnie łącza ściągając obraz standardowego CD, chyba że jesteś testerem płyt (nawet nową metodą jigdo).

Jedną z niewielu, godną uwagi płytą jest KNOPPIX - Live Linux Filesystem On CD (`http://www.knopper.net/knoppix/index-en.html`). Z tą płytą uruchomisz w pełni działającego Debiana bez potrzeby instalowania go na twardym dysku.

9.3.9 Wykonywanie kopii systemu na płytach CD-R

Aby skopiować kluczowe pliki konfiguracyjne i dane na płyty CD-R, użyj przykładowego skryptu backup `backup(examples/)`. Zobacz także 'Kopiowanie oraz archiwizacja całych katalogów' na 116 stronie i 'Kopia różnicowa oraz synchronizacja danych' na 118 stronie.

9.3.10 Kopiowanie CD z muzyką

Nie testowane przeze mnie:

```
# apt-get install cdrecord cdparanoia
# cdparanoia -s -B
# cdrecord dev=0,0,0 speed=2 -v -dao -eject defpregap=1 -audio *.wav
```

albo,

```
# apt-get install cdrdao #disk at once
# cdrdao read-cd --device /dev/cdrom --paranoia-mode 3 my_cd # read cd
# cdrdao write --device /dev/cdrom --speed 8 my_cd    # wypal nowy CD
```

`cdrdao` wykonuje prawdziwą kopię (żadnych przerw, itp. . .)

9.4 X

X Window System jest zapewnione dzięki Xfree86 (`http://www.xfree86.org/`). Istnieją dwie główne wersje serwera X dostępne dla Debiana: XFree86 Version 3.3 (XF3) i XFree86 Version serii 4.x (XF4), obie oparte o specyfikację X11R6 wg. X.Org (`http://www.x.org/`).

Podstawowe informacje nt. X możesz uzyskać z `X(7)`, LDP XWindow-User-HOWTO (`http://www.tldp.org/HOWTO/XWindow-User-HOWTO.html`), oraz Remote X Apps mini-HOWTO (`http://www.tldp.org/HOWTO/mini/Remote-X-Apps.html`). Informacje specyficzne dla Debiana, przeznaczone dla jego użytkowników, przeczytaj w `/usr/share/doc/xfree86-common/FAQ.gz` dostarczanym w pakiecie `xfree86-common`. Zawiera interesujące uwagi na temat przypisywania różnych funkcji do klawiszy autorstwa Brandena Robinsona.

'X serwer' na 151 stronie program na lokalnym hoście, zajmujący się wyświetlaniem X-ów na monitorze użytkownika, jest także odpowiedzialny za obsługę poleceń z klawiatury i myszy.

'Klienty X' na 154 stronie program na (zdalnym lub lokalnym) hoście, który uruchamia oprogramowanie zgodne z X.

Zauważ, że jest to odwrócenie rozumienia słowa serwer i klient przyjętego w innych kontekstach.

Istnieje kilka dróg, aby „X serwer" (strona wyświetlająca) przyjął zdalne połączenie od „klienta X" (strona oprogramowania).

- Wykorzystanie `xhost`
  - mechanizm listy hostów (bardzo niebezpieczny).
  - niezaszyfrowany protokół (narażony na podsłuchiwanie).
  - Nie używaj go, jeśli możesz.
  - Zobacz 'Zdalne połączenie z X – `xhost`' na 158 stronie i `xhost(1x)`.
- Wykorzystanie **`xauth`**
  - mechanizm cookies (niebezpieczny, ale lepszy niż `xhost`).
  - niezaszyfrowany protokół (narażony na podsłuchiwanie).
  - używaj go jedynie dla połączeń lokalnych, gdyż nie obciąża procesora tak jak `ssh -X`.
  - Zobacz 'Zdobywanie uprawnień administratora w X' na 160 stronie i `xauth(1x)`.
- Wykorzystanie `xdm`, `wdm`, `gdm`, `kdm`, . . .
  - mechanizm cookies (równie niebezpieczny jak `xauth`).
  - Zobacz `xdm(1x)` i `Xsecurity(7)`, aby zdobyć podstawowe informacje nt. kontroli dostępu do X.
  - Zobacz `wdm(1x)`, `gdm(8)`, i `kdm.options(5)` dla dalszych informacji, jeśli te programy są zainstalowane.
  - Zobacz 'System-V `init` i poziomy startu (runlevels)' na 101 stronie aby zorientować się jak wyłączyć `xdm`, by logować się w trybie tekstowym bez usuwania pakietu `xdm`.
- Wykorzystanie **`ssh -X`**
  - mechanizm podłączania portów poprzez secure shell (**bezpieczne**).
  - zaszyfrowany protokół (strata zasobów, gdy używane lokalnie).
  - używaj w połączeniach zdalnych.
  - Zobacz 'Zdalne połączenie z X – `ssh`' na 159 stronie.

Wszystkie metody połączeń zdalnych, za wyjątkiem wykorzystywania `ssh`, wymagają włączonych połączeń TCP/IP w X serwerze. Zobacz 'Używanie X'ów poprzez TCP/IP' na 158 stronie.

9.4.1 Pakiety związane z X

Istnieje kilka (meta)pakietów, które zapewniają prostą instalację X Window System w Woodym.

`x-window-system-core` Ten metapakiet dostarcza kluczowe składniki dla pojedynczej stacji roboczej, na której uruchomiony ma być X Window System. Dostarcza biblioteki, X serwer (`xserver-xfree86`), zestaw czcionek, jak również kilka podstawowych klientów X i programów użytkowych.

`x-window-system` Ten metapakiet dostarcza praktycznie wszystkich składników X Window System rozwijanych przez XFree86 Project, jak również zbiór popularnych kiedyś dodatkowych programów. (Zauważ, że ten pakiet zależy od `x-window-system-core`, `twm` oraz `xdm`, tak więc w przypadku jego instalacji `x-window-system-core` nie musi zostać oddzielnie wyznaczony do zainstalowania)

`xserver-common-v3` Pliki i programy użytkowe wspólne dla serwerów X XFree86 3.x (XF3)

`xserver-*` Dodano pakiety serwera XF3, aby obsłużyć sprzęt z różnych powodów nie obsługiwany przez nowy serwer XF4 (`xserver-xfree86`). Niektóre stare karty ATI mach64 nie są obsługiwane przez XF4, zaś inne karty graficzne zawieszają się w wersji XF4 dostępnej wraz z Woodym, itp. (Aby obejrzeć dostępne pakiety wykonaj `apt-cache search xserver- | less`. Wszystkie z tych serwerów XF3 zależą od pakietu `xserver-common-v3`.)

W większości przypadków `x-window-system` jest odpowiednim pakietem do zainstalowania. (Jeśli chcesz logować się w trybie tekstowym, upewnij się że wyłączyłeś usługę `xdm` tak, jak opisano to w części 'Jak wyłączyć automatyczne uruchamianie Xów przy starcie?!' na 113 stronie.)

9.4.2 Wykrywanie sprzętu dla X

Aby wykryć swój sprzęt podczas konfiguracji X-ów, zainstaluj następujące pakiety przed instalacją X Window System:

- `discover` – narzędzie identyfikacji sprzętu.
- `mdetect` – autodetekcja myszy.
- `read-edid` – gromadzenie informacji sprzętowej o monitorach VESA PnP.

9.4.3 X serwer

Zobacz `XFree86(1x)` aby uzyskać informację o X serwerze.

Uruchom X serwer z lokalnej konsoli:

```
$ startx -- :<display> vtXX
e.g.:
$ startx -- :1 vt8 -bpp 16
... uruchom na vt8 podłączonym do localhost:1 w trybie 16 bpp
```

Argumenty podane po `--` są przekazywane X serwerowi.

Jeśli używasz skryptu `~/.xserverrc` do konfigurowania uruchomienia X, pamiętaj aby wykonać `exec` na prawidłowym X serwerze. W przeciwnym wypadku może on wolno się uruchamiać i zamykać. Na przykład:

```
#!/bin/sh
exec /usr/bin/X11/X -dpi 100 -nolisten tcp
```

Konfiguracja serwera X (wersja 4)

Aby skonfigurować (lub przekonfigurować) serwer XF4,

```
# dpkg-reconfigure --priority=low xserver-common
# dpkg-reconfigure --priority=low xserver-xfree86
```

co utworzy plik `/etc/X11/XF86Config-4`, po czym skonfiguruj X-y używając skryptu `dexconf`.

Konfiguracja serwera X (wersja 3)

Aby skonfigurować (lub przekonfigurować) serwer XF3 na przykład dla karty ATI mach64,

```
# dpkg-reconfigure --priority=low xserver-common-v3
# dpkg-reconfigure --priority=low xserver-mach64
```

co utworzy plik `/etc/X11/XF86Config`, po czym skonfiguruj X-y używając skryptu `xf86config-v3`.

Ręczna konfiguracja serwera X,

Dla Woodiego modyfikując ustawienia w `/etc/X11/XF86Config-4` **nie zmieniaj zawartości pliku konfiguracyjnego pomiędzy liniami**:

```
### BEGIN DEBCONF SECTION
[ciach]
### END DEBCONF SECTION
```

Zamiast tego **dodaj własne opcje przed tym tekstem**. Na przykład, aby użyć innej karty graficznej dodaj podobne linie na *początku* pliku:

```
Section "Device"
  Identifier        "Moje Urzadzenie"
  Driver            "ati"
  Option            "NoAccel"
EndSection

Section "Screen"
  Identifier  "Mój Ekran"
  Device      "Moje Urządzenie"
  Monitor     "Generic Monitor"
  DefaultDepth 24
  Subsection "Display"
    Depth       8
    Modes       "1280x960" "1152x864" "1024x768" "800x600" "640x480"
  EndSubsection
  Subsection "Display"
    Depth       16
    Modes       "1280x960" "1152x864" "1024x768" "800x600" "640x480"
  EndSubsection
  Subsection "Display"
    Depth       24
    Modes       "1280x960" "1152x864" "1024x768" "800x600" "640x480"
  EndSubsection
EndSection

Section "ServerLayout"
    Identifier        "Custom"
    Screen            "Mój Ekran"
    InputDevice       "Generic Keyboard" "CoreKeyboard"
    InputDevice       "Configured Mouse" "CorePointer"
EndSection
```

Dla Sarge (`testing` w momencie pisania tych słów), jeśli chcesz wraz z aktualizacją zatrzymać własne modyfikacje pliku `/etc/X11/XF86Config`, uruchom następujące polecenia jako root:

```
# cp /etc/X11/XF86Config-4 /etc/X11/XF86Config-4.custom
# md5sum /etc/X11/XF86Config-4 > /var/lib/xfree86/XF86Config-4.md5sum
# dpkg-reconfigure xserver-xfree86
```

By poprawić **wygląd czcionek**, musisz przeedytować plik `/etc/X11/XF86Config-4` tak jak opisano to w 'Czcionki TrueType dla X' na 162 stronie.

Sprawdź również inne części konfiguracji X. Złe ustawienia monitora mogą być istotniejszą przyczyną bólu głowy niż złe czcionki, więc upewnij się, że odświeżanie jest ustawione na tak wysokie jakie potrafi obsłużyć monitor (85 Hz jest wspaniałe, 75 Hz jest OK, a 60 Hz to tragedia).

9.4.4 Klienty X

Większość klientów X może być uruchomiona poleceniem podobnym do:

```
client $ xterm -geometry 80x24+30+200 -fn 6x10 -display hostname:0 &
```

Dodatkowe argumenty oznaczają:

- `-geometry` *SZER*x*WYS*+*XOFF*+*YOFF*: początkowy rozmiar i pozycję okna programu.
- `-fn` *NAZWAFONTU*: czcionkę służącą do wyświetlania tekstu. *NAZWAFONTU* może przyjmować wartość:
  - a14: Normalny rozmiar
  - a24: Duży rozmiar czcionki
  - … (sprawdź dostępne czcionki za pomocą `xlsfont`.)
- `-display` *displayname*: nazwa X serwera, który ma zostać użyty. *displayname* może przyjmować wartość:
  - *hostname*`:`*D*`.`*S* oznacza ekran *S* na monitorze *D* komputera *hostname*; X serwer dla tego monitora nasłuchuje na porcie TCP 6000+D
  - *host*`/unix:`*D*`.`*S* oznacza ekran *S* na monitorze *D* hosta *host*; X-serwer dla tego monitora nasłuchuje na porcie typu UNIX. `/tmp/.X11-unix/XD` (więc jest dostępny tylko z *host*).
  - `:`*D*`.`*S* jest równoważne do *host*`/unix:`*D*`.`*S*, gdzie *host* to nazwa lokalnego hosta.

Domyślna wartość zmiennej *displayname* dla klientów X (oprogramowanie), może być ustawiona poprzez zmienną środowiskową DISPLAY. Można to osiągnąć przez np. uruchomienie następujących skryptów przed klientem X:

```
$ export DISPLAY=:0
   # Domyślne ustawienie, lokalna maszyna, pierwszy screen
$ export DISPLAY=hostname.fulldomain.name:0.2
$ export DISPLAY=localhost:0
```

Start Xów może być dostosowany do własnych potrzeb za pomocą `~/.xinitrc`. Np:

```
xrdb -load $HOME/.Xresources
xsetroot -solid gray &
xclock -g 50x50-0+0 -bw 0 &
xload -g 50x50-50+0 -bw 0 &
xterm -g 80x24+0+0 &
xterm -g 80x24+0-0 &
twm
```

Tak jak opisano to w 'Personalizowane sesje X.' na sąsiedniej stronie, zastępuje to wszystko, co dzieje się podczas uruchomienia `Xsession` przez `startx`. Używaj `~/.xsession`, a z powyższego sposobu korzystaj jedynie w ostateczności. Zobacz `xsetroot(1x)`,`xset(1x)` i 'Zasoby X' na 159 stronie.

9.4.5 Sesje X

Sesja X (X-serwer + klient X) może być rozpoczęta poprzez:

- `startx`: skrypt obudowujący `xinit`, służący do uruchomienia sesji z terminala tekstowego. Jeśli plik `~/.xinitrc` nie istnieje, `/etc/X11/Xsession` jest wywoływany przez `/etc/X11/xinit/xinitrc`.
- `xdm`, `gdm`, `kdm`, lub `wdm`: demon uruchamiający X-serwer i klientów, umożliwiający logowanie w GUI. `/etc/X11/Xsession` jest uruchamiany bezpośrednio.

Tryb tekstowy może być udostępniony, jak to opisano w 'Jak wyłączyć automatyczne uruchamianie Xów przy starcie?!' na 113 stronie.

Personalizowane sesje X.

Domyślny skrypt startowy `/etc/X11/Xsession` jest kombinacją `/etc/X11/Xsession.d/50xfree86-common_determine-startup` oraz `/etc/X11/Xsession.d/99xfree86-common_start`.

Na przebieg wykonania `/etc/X11/Xsession` wpływają ustawienia z `/etc/X11/Xsession.options`. Jego głównym zadaniem jest uruchomienie za pomocą `exec` pierwszego istniejącego programu znajdującego się na następującej liście:

1. `~/.xsession` lub `~/.Xsession`, jeśli istnieje.
2. `/usr/bin/x-session-manager`, jeśli istnieje.
3. `/usr/bin/x-window-manager`, jeśli istnieje.
4. `/usr/bin/x-terminal-emulator`, jeśli istnieje.

Dokładne znaczenie tych poleceń jest uzależnione od alternatywnych ustawień systemu Debian, opisanych w 'Alternatywne polecenia' na 101 stronie. Na przykład:

```
# update-alternatives --config x-session-manager
... lub
# update-alternatives --config x-window-manager
```

Aby wybrać domyślny menedżer okien X, jednocześnie zachowując GNOME i KDE, zamień plik `/etc/X11/Xsession.d/50xfree86-common_determine-startup` z plikiem dostarczonym w drugim raporcie o błędzie na `http://bugs.debian.org/168347` (mam nadzieję, że zostanie to szybko naprawione) i wyedytuj `/etc/X11/Xsession.options` jak następuje:

```
# /etc/X11/Xsession.options
#
# opcje konfiguracyjne dla /etc/X11/Xsession
# Wyjaśnienie dostępnych opcji znajdziesz w Xsession.options(5)
# Default enabled
allow-failsafe
allow-user-resources
```

```
allow-user-xsession
use-ssh-agent
# Domyślnie wyłączone (włącz odkomentowując)
do-not-use-x-session-manager
#do-not-use-x-window-manager
```

Bez wspomnianych powyżej zmian w systemie pakiety `gnome-session` i `kdebase` zapewniają funkcjonalność menedżera sesji. Usunięcie ich, umożliwia ustawienie jako domyślnego innego menedżera okien.

W systemie, gdzie `/etc/X11/Xsession.options` zawiera linię `allow-user-xsession` bez poprzedzających jej znaków, każdy użytkownik, który stworzy `~/.xsession` lub `~/.Xsession` będzie mógł dostosować działanie `/etc/X11/Xsession`.

Ostatnie polecenie w pliku `~/.xsession` powinno przyjmować formę: `exec` *`some-window/session-manager`*, aby uruchomić ulubiony menedżer sesji.

Dobry przykład skryptu `~/.xsession` jest podany w `/usr/share/doc/xfree86-common/examples/xsession.gz`.

Używam tego skryptu do wybierania menedżera okien, dostępu do ekranu i obsługiwanego języka dla poszczególnych użytkowników. Przeczytaj także 'Startowanie sesji X dla użytkownika.' na tej samej stronie, 'Zdobywanie uprawnień administratora w X' na 160 stronie i 'Przykład wielojęzykowego systemu X window' na 178 stronie.

Jeśli chcesz by kilka klientów X startowało automatycznie, zobacz przykłady z 'Klienty X' na 154 stronie i wywołuj je z `~/.xsession` zamiast `~/.xinitrc`.

Dodatkowe zasoby X dla poszczególnych użytkowników są przechowywane w `~/.Xresources`. Przeczytaj 'Zasoby X' na 159 stronie.

Dostosowania mapy klawiatury i funkcji klawiszy myszy również można dokonać w skryptach startowych użytkownika. Przeczytaj 'Mapy klawiatury i mapowanie myszy w X' na 160 stronie.

Startowanie sesji X dla użytkownika.

Podążając za wytycznymi, opisanymi w 'Personalizowane sesje X.' na poprzedniej stronie, wybrany przez użytkownika menedżer sesji/okien może być uaktywniony poprzez zainstalowanie odpowiedniego pakietu i odpowiednie zmiany pod koniec pliku `~/.xsession`. (Osobiście lubię `blackbox/fluxbox` z powodu ich prostego stylu i szybkości.):

- domyślny menedżer sesji X
  - Przeczytaj 'Alternatywne polecenia' na 101 stronie
  - `exec /usr/bin/x-session-manager`

- domyślny menedżer okien X
  - Przeczytaj 'Alternatywne polecenia' na 101 stronie.

  - `exec /usr/bin/x-window-manager`

- Menadżer sesji GNOME (przeładowany)
  - Zainstaluj pakiet: `gnome-session`
  - `exec /usr/bin/gnome-session`

- Menadżer sesji KDE (przeładowany)
  - Zainstaluj pakiet: `kdebase` (lub `kdebase3` dla KDE3)
  - `exec /usr/bin/kde2`

- Blackbox window manager (lekki, szybki).
  - Zainstaluj pakiet: `blackbox`
  - `exec /usr/bin/blackbox`

- Fluxbox window manager (lekki, nowy blackbox).
  - Zainstaluj pakiet: `fluxbox`
  - `exec /usr/bin/fluxbox`

- Xfce window manager (podobny do Mac OS-X, SUN CDE).
  - Zainstaluj pakiet: `xfce`
  - `exec /usr/bin/xfwm`

- IceWM window manager (mały, alternatywa dla GNOME)
  - Zainstaluj pakiet: `icewm`
  - `exec /usr/bin/X11/icewm`

- FVWM2 virtual window manager (mały, podobny do Win95)
  - Zainstaluj pakiet: `fvwm`
  - `exec /usr/bin/fvwm2`

- Windowmaker window manager (trochę jak NexT)
  - Zainstaluj pakiet: `wmaker`
  - `exec /usr/bin/wmaker`

- Enlightenment window manager (przeładowany)
  - Zainstaluj pakiet: `enlightenment`
  - `exec /usr/bin/enlightenment`

Przeczytaj Window Managers for X (`http://www.xwinman.org`).

Ustawianie KDE i GNOME

Aby otrzymać w pełni funkcjonalne środowiska KDE lub GNOME, przydatne są poniższe metapakiety:

- KDE: zainstaluj pakiet `kde`
- GNOME: zainstaluj pakiet `gnome`

Instalacja tych pakietów za pomocą narzędzi korzystających ze zmiennych `recommends`, takich jak `dselect` i `aptitude`, zaopatrzy Cię w większy wybór oprogramowania niż instalacja tylko za pomocą `apt-get`.

Jeśli chcesz logować się w trybie tekstowym wyłącz menedżera wyświetlania X, takiego jak `kdm`, `gdm`, i `wdm`, które mogą zostać zainstalowane poprzez zależności, jak opisano to w 'Jak wyłączyć automatyczne uruchamianie Xów przy starcie?!' na 113 stronie.

Jeśli GNOME ma się uruchamiać domyślnie, zamiast KDE, upewnij się jak właściwie skonfigurować `x-session-manager`, co opisano w 'Alternatywne polecenia' na 101 stronie.

9.4.6 Używanie X'ów poprzez TCP/IP

Ponieważ zdalne połączenie TCP/IP bez szyfrowania można stosunkowo łatwo podsłuchiwać, domyślne ustawienia X w aktualnej wersji systemu Debian nie udostępnia gniazda TCP/IP. Rozważ użycie `ssh`, jeśli zależy Ci na zdalnym połączeniu (zobacz 'Zdalne połączenie z X – `ssh`' na następnej stronie).

Metoda opisana tutaj nie jest zalecana, chyba że stosowana jest w bardzo bezpiecznym środowisku, za dobrym firewallem i tylko wtedy jeśli wszyscy użytkownicy są godni zaufania. Użyj tego polecenia, aby sprawdzić obecne ustawienia X serwera dotyczące gniazd TCP/IP:

```
# find /etc/X11 -type f -print0 | xargs -0 grep nolisten
/etc/X11/xinit/xserverrc:exec /usr/bin/X11/X -dpi 100 -nolisten tcp
```

Usuń `-nolisten` aby przywrócić nasłuch poprzez TCP/IP.

9.4.7 Zdalne połączenie z X – `xhost`

`xhost` umożliwia dostęp wyznaczonym hostom. Jest to bardzo niebezpieczne. Poniższa linia wyłącza sprawdzanie hostów i pozwala na połączenie skądkolwiek, jeśli dozwolone są połączenia TCP/IP (przeczytaj 'Używanie X'ów poprzez TCP/IP' na tej samej stronie):

```
$ xhost +
```

Możesz ponownie uaktywnić sprawdzanie hostów poprzez:

```
$ xhost -
```

`xhost` nie rozróżnia użytkowników na zdalnych maszynach. Oprócz tego adresy hostów mogą być oszukane (spoofing).

Powinno unikać się tej metody, szczególnie jeśli jesteś podłączony do sieci niegodnej zaufania (np. PPP dialup). Przeczytaj `xhost(1x)`.

9.4.8 Zdalne połączenie z X – `ssh`

`ssh` zapewnia bezpieczny sposób połączenia lokalnego X serwera z programem uruchomionym na zdalnej maszynie.

- Ustaw pozycje `X11Forwarding` i `AllowTcpForwarding` na `yes` w pliku `/etc/ssh/sshd_config` zdalnego hosta, jeśli chcesz uniknąć podawania tych opcji w lini poleceń.
- Uruchom X serwer na zdalnej maszynie.
- Otwórz `xterm` na lokalnym komputerze.
- Uruchom `ssh`, aby ustanowić połączenie ze stroną zdalną.

```
localname @ localhost $ ssh -q -X -l loginname remotehost.domain
Password:
.....
```

- Uruchom aplikacje X po stronie zdalnej.

```
loginname @ remotehost $ gimp &
```

Metoda ta umożliwia wyświetlanie wyjścia zdalnego klienta X, jak gdyby komunikował się on lokalnie poprzez gniazda UNIX.

9.4.9 Emulator terminala X – `xterm`

Dowiedz się wszystkiego o `xterm`: `http://dickey.his.com/xterm/xterm.faq.html`.

9.4.10 Zasoby X

Dużo starszych programów dla X, jak `xterm`, używa bazy danych zasobów X do konfiguracji własnego wyglądu. Plik `~/.Xresources` jest używany do przechowywania specyfikacji zasobów użytkownika. Jego zawartość jest automatycznie dołączana do domyślnych zasobów X podczas logowania. Zasoby dla całego systemu przechowywane są w `/etc/X11/Xresources/*` i domyślne ustawienia aplikacji w `/etc/X11/app-defaults/*`. Użyj tych ustawień jako wyjściowych.

Oto parę przydatnych ustawień dla twojego pliku `~/.Xresources`:

```
! Ustaw czcionkę do bardziej czytelnego rozmiaru 9x15
XTerm*font: 9x15
```

```
! Wyświetl pasek przewijania
XTerm*scrollBar: true

! Ustaw rozmiar bufora na 1000 linii
XTerm*saveLines: 1000

! Duży ekran kterm
KTerm*VT100*fontList: -*-fixed-medium-r-normal--24-*,\
 -*-gothic-medium-r-normal--24-*,\
 -*-mincho-medium-r-normal--24-*
```

Aby zmiany te weszły w życie natychmiast, dodaj je do bazy danych zasobów poleceniem:

```
xrdb -merge ~/.Xresources
```

Zobacz `xrdb(1x)`.

9.4.11 Mapy klawiatury i mapowanie myszy w X

Program `xmodmap` jest używany do edycji i wyświetlania mapę modyfikacji klawiatury i tablicę klawiszy, które są używane przez aplikacje klienckie do zamiany zdarzeń klawiaturowych na funkcje w X.

```
$ xmodmap -pm
 ... wyświetla aktualną mapę modyfikacji
$ xmodmap -pk | pager
 ... wyświetla aktualną tablicę mapy klawiszy
$ xmodmap -e "pointer = 3 2 1" # ustawia myszkę dla leworęcznych.
$ xmodmap ~/.xmodmaprc # ustawia klawiaturę korzystając z ~/.xmodmaprc
```

Jest to najczęściej ładowane ze skryptu startowego sesji użytkownika, `~/.xsession`.

By zobaczyć kody klawiszy (`keycode`), użyj `xev` i naciskaj klawisze. By znaleźć znaczenie klawiszy `keysym`, spójrz do definicji z pliku `/usr/include/X11/keysymdef.h`. Wszystkie definicje nazywane są jako `XK_` poprzedzające nazwy funkcji.

Zobacz `xmodmap(1x)`.

9.4.12 Zdobywanie uprawnień administratora w X

Jeśli program z GUI musi być uruchomiony z uprawnieniami administratora użyj następującej procedury.**Nigdy nie uruchamiaj X serwera bezpośrednio z konta roota**, aby uniknąć problemów z bezpieczeństwem.

Uruchom X serwer jako zwykły użytkownik i otwórz okno programu `xterm`. Potem:

```
$ XAUTHORITY=$HOME/.Xauthority
$ export XAUTHORITY
$ su root
Password:*****
# printtool &
```

Używając tego sposobu by zyskać prawa użytkownika innego niż root, upewnij się, że ~ `/.Xauthority` jest dostępny do odczytu dla grupy tego użytkownika.

Aby zautomatyzować te sekwencje poleceń, utwórz plik `~/.xsession` na koncie użytkownika zawierający linie:

```
# To sprawia, że X-y działają, kiedy za pomocą su staję się root'em.
if [ -z "$XAUTHORITY" ]; then
        XAUTHORITY=$HOME/.Xauthority
        export XAUTHORITY
fi
unset XSTARTUP
# Jeśli jakieś menedżery okien/sesji są wybrane, odkomentuj następujące
# linie i dostosuj je do swoich potrzeb.
#XSTARTUP=/usr/bin/blackbox
# To uruchomi program x-window/session-manager
if [ -z "$XSTARTUP" ]; then
  if [ -x /usr/bin/x-session-manager ]; then
    XSTARTUP=x-session-manager
  elif [ -x /usr/bin/x-window-manager ]; then
    XSTARTUP=x-window-manager
  elif [ -x /usr/bin/x-terminal-emulator ]; then
    XSTARTUP=x-terminal-emulator
  fi
fi
# Uruchom automatycznie wybrany menedżer okien/sesji X
exec $XSTARTUP
```

Następnie uruchom `su` (nie `su -`) w oknie programu `xterm`. Teraz programy uruchamiane z tego `xterma` będą wyświetlać się temu użytkownikowi, podczas gdy pracować będą z prawami administratora. Działa to tak długo, jak długo domyślny plik `/etc/X11/Xsession` jest wykonywany. Jeśli użytkownik zmienił swoje środowisko edytując `~/.xinit` lub ~ `/.xsession`, to wymieniona zmienna środowiskowa `XAUTHORITY` musi być w nich odpowiednio zmieniona.

Opcjonalnie, `sudo` może być wykorzystane dla zautomatyzowania tej sekwencji poleceń:

```
$ sudo xterm
... lub
$ sudo -H -s
```

W tym wypadku plik `/root/.bashrc` powinien zawierać:

```
if [ $SUDO_USER ]; then
    sudo -H -u $SUDO_USER xauth extract - $DISPLAY | xauth merge -
fi
```

Działa to poprawnie nawet gdy katalog domowy użytkownika jest na NFS, ponieważ root nie odczytuje jego `.Xauthority`.

Jest kilka pakietów przeznaczonych specjalnie dla tego celu: `kdesu`, `gksu`, `gksudo`, `gnome-sudo`, i `xsu`. Parę innych metod może być użytych do osiągnięcia podobnych rezultatów: utworzenie dowiązania symbolicznego z pliku `/root/.Xauthority` do użytkownika, użycie skryptu sux (`http://fgouget.free.fr/sux/sux-readme.shtml`); lub wpisanie „`xauth merge ~`*`USER_RUNNING_X`*`/.Xauthority`” w plikach startowych administratora.

Przeczytaj więcej na debian-devel mailing list (`http://lists.debian.org/debian-devel/2002/debian-devel-200207/msg00259.html`).

9.4.13 Czcionki TrueType dla X

Standardowy `xfs` z XFree86-4 poprawnie współpracuje z czcionkami TrueType. Jeśli natomiast korzystasz z XFree86-3 musisz zainstalować osobny pakiet, taki jak `xfs-xtt`.

Musisz upewnić się, że wszystkie programy, które mają korzystać z czcionek TrueType zostały połączone z libXft lub libfreetype (prawdopobnie nie musisz się o to martwić, jeśli korzystasz z prekompilowanych pakietów .deb).

Na początku przygotuj obsługę czcionek:

- Zainstaluj pakiety `x-ttcidfont-conf` i `defoma`. To pozwoli zautomatyzować tworzenie plików `fonts.scale` i `fonts.dir`.

  ```
  # apt-get install x-ttcidfont-conf
  ```

- Zmień część `Section „Files”` w pliku `/etc/X11/XF86Config-4` tak jak podano poniżej:

  ```
  Section "Files"
      FontPath  "/var/lib/defoma/x-ttcidfont-conf.d/dirs/TrueType"
      FontPath  "/usr/share/fonts/truetype"
      FontPath  "/usr/lib/X11/fonts/CID"
      FontPath  "/usr/lib/X11/fonts/Speedo"
      FontPath  "/usr/lib/X11/fonts/misc"
      FontPath  "/usr/lib/X11/fonts/cyrillic"
      FontPath  "/usr/lib/X11/fonts/100dpi:unscaled"
      FontPath  "/usr/lib/X11/fonts/75dpi:unscaled"
      FontPath  "/usr/lib/X11/fonts/Type1"
  EndSection
  ```

  Pierwsza linia ustawi XFree86 by korzystało z czcionek TrueType, które zainstalujesz z pakietów systemu Debian. Wpisz z Type1 jest przesunięty na dół, gdyż XFree86 wyświetla je dość słabo. Opcja `:unscaled` dla czcionek bitmapowych powinna być zbędna w nowszych XF4, ale zostawiłem ją dla pewności. By zachować własne modyfikacje `/etc/X11/XF86Config-4` postępuj zgodnie z instrukcjami z 'Ręczna konfiguracja serwera X,' na 152 stronie.

Następnie zainstaluj pakiety z fontami (DFSG):

- Zachodnie czcionki TrueType:
  - `ttf-bitstream-vera`: A set of high-quality TrueType fonts created by Bitstream, Inc. [2]
  - `ttf-freefont`: A set of free high-quality TrueType fonts covering the UCS character set.
  - `ttf-thryomanes`: A TrueType Unicode font covering Latin, Greek, Cyrillic and IPA.
- Czcionki azjatyckie:
  - `tfm-arphic-bsmi00lp`: Chinese Arphic „AR PL Mingti2L Big5" TrueType font TeX font metric data
  - `tfm-arphic-bkai00mp`: Chinese Arphic „AR PL KaitiM Big5" TrueType font TeX font metric data
  - `tfm-arphic-gbsn00lp`: Chinese Arphic „AR PL SungtiL GB" TrueType font TeX font metric data
  - `tfm-arphic-gkai00mp`: Chinese Arphic „AR PL KaitiM GB" TrueType font TeX font metric data
  - `ttf-baekmuk`: Korean Baekmuk series TrueType fonts
  - `hbf-jfs56`: Chinese Jianti Fangsong 56x56 bitmap font (GB2312) for CJK
  - `hbf-cns40-b5`: Chinese Fanti Song 40x40 bitmap font (Big5) for CJK
  - `hbf-kanji48`: Japanese Kanji 48x48 bitmap font (JIS X-0208) for CJK

Ponieważ wybór **Wolnych** czcionek jest dosyć ograniczony, dobrym rozwiązaniem może być równoczesna instalacja ich komercyjnych odpowiedników. Aby uczynić ich instalację łatwiejszą, stworzono kilka pakietów:

- `ttf-commercial`
- `msttcorefonts (>1.1.0)` [3]

Będziesz miał naprawdę duży wybór czcionek, jeśli połączysz użycie swoich **Wolnych** czcionek z innymi, komercyjnymi.

Wszystkie te pakiety czcionek powinny działać bez żadnych dodatkowych poleceń i powinny być dostępne dla wszystkich programów X, które używają czcionek systemowych. W ich skład wchodzą Xterm, Emacs i większość aplikacji spoza KDE i GNOME.

Teraz uruchom `xfontsel` i wybierz jakąkolwiek czcionkę TrueType z menu fndry. W menu „fmly" powinieneś zobaczyć wiele aktywnych wpisów.

Dla KDE2.2 i GNOME1.4 (z libgdxft0, które jest pewnym obejściem by czcionki w GTK 1.2 były antyaliasowane), potrzeba ustawienia dodatkowo Xft1. Xft1 jest bardzo przestarzałe i praktycznie potrzebne tylko dla GNOME1.4 i KDE2.2. W pliku `/etc/X11/XftConfig` dodaj następującą linię

---

[2] Dostępne od dystrybucji Sarge.

[3] Pakiet w Woodym nie działa z powodu zmiany strony internetowej Microsoftu w sierpniu 2002 roku. Użyj wersji z Sarge, nawet gdy używasz Woodiego.

```
        dir "/var/lib/defoma/x-ttcidfont-conf.d/dirs/TrueType"
```

przed innymi katalogami. [4]

Dla GNOME2 i KDE3 (wszystkie wydania po Sarge), musisz ustawić `fontconfig`, który jest wykorzystywany przez Xft2 do odnajdywania czcionek. [5] Nie powinno być potrzeby instalacji żadnych dodatków do tego, ponieważ wszystkie pakiety używające `fontconfig` już od niego zależą (pośrednio).

Na początek zajrzyj do `/etc/fonts/fonts.conf`. Powinna tam być linia taka jak poniżej. Jeśli jej nie ma, otwórz `/etc/fonts/local.conf` i dodaj

```
        <dir>/var/lib/defoma/x-ttcidfont-conf.d/dirs/TrueType</dir>
```

zaraz po linii <fontconfig>.

`Fontconfig` powinien wykryć to od razu i „fc-list" powinien wyświetlać nowe czcionki. Kolejną przyjemną możliwością `fontconfig`, jest fakt, że możesz umieścić czcionki w `~/.fonts/` i wszystkie programy powinny od razu mieć do nich dostęp.

Jeśli ręcznie doinstalujesz nowy zestaw czcionek bez używania pakietu z systemu Debian, uruchom

```
   # xset fp rehash
```

by XFree86 przeglądnęło zawartość katalogu ponownie i dodało nowe czcionki.

9.4.14 Przeglądarka WWW w X

Jest kilka graficznych przeglądarek stron WWW w Woodym:

- `mozilla` The Mozilla (nowa)
- `galeon` przeglądarka oparta o Mozillę z Gnome UI (nowa)
- `konqueror` przeglądarka dla KDE
- `dillo` przeglądarka w GTK
- `amaya-gtk` przeglądarka wzorcowa W3C
- `amaya-lesstif` przeglądarka wzorcowa W3C
- `netscape-...` (dużo, stare)
- `communicator-...` (dużo, stare)
- ...

---

[4] Nie posiadam żadnego oprogramowania korzystającego z xft1 więc nie jestem pewien czy potrzeba zrestartować X by te zmiany odniosły skutek. Pamiętam, że bodajże „xftcache" powinien uaktualnić bufor Xft1, ale byłoby dobrze jakby ktoś to potwierdził.

[5] `Fontconfig` **nie** istnieje w Woodym.

Wersja pakietu `mozilla` musi odpowiadać wersji wymaganej przez pakiet `galeon`. Chociaż różnią się one interfejsem użytkownika, to korzystają z tego samego mechanizmu renderowania stron HTML Gecko.

Wtyczki dla przeglądarek takich, jak `mozilla` i `galeon` mogą być uaktywnione poprzez ręczną instalację odpowiednich plików „`*.so`" w katalogu z wtyczkami i ponowne uruchomienie przeglądarki.

Popularne wtyczki:
- wtyczka Javy: zaistaluj „J2SE" z `http://java.sun.com`.
- wtyczka Flash: zainstaluj „Macromedia Flash Player 5" z `http://www.macromedia.com/software/flashplayer/`.
- `freewrl`: przeglądarka VRML i wtyczka dla Netscape'a
- ...

9.5 SSH

SSH (Secure SHell) zapewnia bezpieczny sposób połączeń internetowych. Wersja SSH stanowiąca wolne oprogramowanie nazywa się OpenSSH i jest dostępna w systemie Debian jako pakiet `ssh`.

9.5.1 Podstawy SSH

Najpierw zainstaluj serwer i klienta OpenSSH.

```
# apt-get update && apt-get install ssh
```

Jeśli serwer OpenSSH ma być uruchamiany, to nie może istnieć plik `/etc/ssh/sshd_not_to_be_run`.

SSH zapewnia dwa protokoły identyfikacji:

- Pierwszy protokół SSH:
  - Jedynie Potato obsługuje ten protokół.
  - dostępne metody identyfikacji:
    - RSAAuthentication: oparta na kluczu RSA użytkownika
    - RhostsAuthentication: oparta na .rhosts (niebezpieczna, wyłączona)
    - RhostsRSAAuthentication: oparta na kombinacji .rhosts i kluczy RSA (wyłączona)
    - ChallengeResponseAuthentication: oparta o mechanizm RSA challenge-response
    - PasswordAuthentication: oparta na hasłach

- Drugi protokół SSH:

  - podstawowy dla systemów post-Woody
  - dostępne metody identyfikacji:
    - PubkeyAuthentication: oparta o klucz publiczny
    - HostbasedAuthentication: oparta o `.rhosts` lub `/etc/hosts.equiv`, w połączeniu z metodą klucza publicznego (wyłączona).
    - ChallengeResponseAuthentication: oparta o mechanizm challenge-response
    - PasswordAuthentication: oparta na hasłach

Jeśli używasz Woodiego lub systemu innego niż Debian, to zwróć uwagę na te różnice.

Przeczytaj `/usr/share/doc/ssh/README.Debian.gz`, `ssh(1)`, `sshd(8)`, `ssh-agent(1)`,i `ssh-keygen(1)` dla dalszych informacji.

Główne pliki konfiguracyjne to:

- `/etc/ssh/ssh_config`: Ustawienia domyślne klienta SSH. Przeczytaj `ssh(1)`. Ważne uwagi, pozycje:
  - `Host`: Ogranicza następujące po nim deklaracje (aż do następnego wystąpienia Host) jedynie do komputerów pasujących do wzorca podanego po danym słowie kluczowym.
  - `Protocol`: Określa wersje protokołu SSH. Domyślnie to „2,1”.
  - `PreferredAuthentications`: Określa metodę identyfikacji klienta SSH2. Domyślna wartość to „hostbased,publickkey,keyboard-interactive,password”.
  - `PasswordAuthentication`: Jeśli chcesz logować się używając hasła, upewnij się, że pole to nie jest ustawione na `no`.
  - `ForwardX11`: Domyślnie wyłączone. Nadrzędna wobec tego ustawienia jest opcja wiersza poleceń „`-X`”.

- `/etc/ssh/sshd_config`: Domyślne ustawienia serwera SSH. Przejrzyj też `sshd(8)`. Pozycje godne uwagi:
  - `ListenAddress`: określa lokalny adres, na którym `sshd` powinien nasłuchiwać. Dozwolona jest więcej niż jedna opcja.
  - `AllowTcpForwarding`: Domyślnie wyłączone.
  - `X11Forwarding`: Domyślnie wyłączone.

- `$HOME/.ssh/authorized_keys`: lista domyślnych kluczy publicznych, które zostały użyte przez klientów do połączenia na dane konto na tym komputerze. Zobacz `ssh-keygen(1)`.

- `$HOME/.ssh/identity`: Zobacz `ssh-add(1)` oraz `ssh-agent(1)`.

Rozpoczęcie połączenia przez klienta `ssh`.

```
$ ssh username@hostname.domain.ext
$ ssh -1 username@hostname.domain.ext # Wymusza używanie SSH w wersji 1
$ ssh -1 -o RSAAuthentication=no -l username foo.host
```

```
    # wymusza używanie haseł SSH1
$ ssh -o PreferredAuthentications=password -l username foo.host
    # wumusza używanie haseł SSH2
```

Z punktu widzenia użytkownika `ssh` działa jako sprytniejszy i bezpieczniejszy odpowiednik programu `telnet` (nie zarzuci nas np. ^]).

9.5.2 Przekierowywanie portów dla tuneli SMTP/POP3

Aby utworzyć potok łączący port 25 komputera *zdalny-serwer* i port 4025 na komputerze lokalnym, oraz potok łączący port 110 komputera *zdalny-serwer* z portem 4110 lokalnej maszyny, poprzez `ssh`, wykonaj na lokalnej maszynie:

```
# ssh -q -L 4025:zdalny-serwer:25 4110:zdalny-serwer:110 \
           użytkownik@zdalny-serwer
```

Jest to bezpieczny sposób na nawiązywanie połączenia z serwerami SMTP/POP3 w Internecie. Ustaw `AllowTcpForwarding` na `yes` w pliku `/etc/ssh/sshd_config` zdalnego komputera.

9.5.3 Połączenia z mniejszą ilością haseł – RSA

Można uniknąc konieczności pamiętania hasła dla każdego używanego zdalnego komputera używając RSAAuthentication (protokół SSH1) lub PubkeyAuthentication (protokół SSH2).

Na zdalnym systemie ustaw odpowiednio pozycje „RSAAuthentication yes" lub „PubkeyAuthentication yes", w `/etc/ssh/sshd_config`.

Następnie wygeneruj klucze identyfikacyjne lokalnie i zainstaluj klucz publiczny na zdalnym systemie:

```
$ ssh-keygen          # RSAAuthentication: klucz RSA1 dla SSH1
$ cat .ssh/identity.pub   | ssh użytkownik1@zdalny \
        "cat - >>.ssh/authorized_keys"
...
$ ssh-keygen -t rsa   # PubkeyAuthentication: Klucz RSA dla SSH2
$ cat .ssh/id_rsa.pub | ssh użytkownik1@zdalny \
        "cat - >>.ssh/authorized_keys"
...
$ ssh-keygen -t dsa   # PubkeyAuthentication: klucz DSA dla SSH2
$ cat .ssh/id_dsa.pub | ssh użytkownik1@zdalny \
        "cat - >>.ssh/authorized_keys"
```

Można później zmienić hasło za pomocą polecenia „`ssh-keygen -p`". Sprawdź ustawienia testując połączenie. W razie nieprawidłowości skorzystaj z polecenia „`ssh -v`".

Możesz dodać opcje do pozycji w `authorized_keys`, aby ograniczyć uprawnienia hostów i użyć odpowiedniego polecenia. Po dalsze informacje sięgnij do `sshd(8)`.

Zauważ, że SSH2 obsługuje `HostbasedAuthentication`. Aby uaktywnić ten mechanizm, musisz zmienić wartość zmiennej `HostbasedAuthentication` na `yes` zarówno w pliku `/etc/ssh/sshd_config` po stronie serwera, jak i w plikach `/etc/ssh/ssh_config` lub `$HOME/.ssh/config` na komputerze klienta.

9.5.4 Radzenie sobie z innymi klientami SSH

Istnieje parę klientów SSH na platformy inne niż Unix.

Windows puTTY (`http://www.chiark.greenend.org.uk/~sgtatham/putty/`) (GPL)

Windows (cygwin) SSH w cygwin (`http://www.cygwin.com/`) (GPL)

Macintosh Classic macSSH (`http://www.macssh.com/`) (GPL) [Zauważ, że Mac OS X zawiera OpenSSH; użyj `ssh` w aplikacji Terminala]

Zobacz także SourceForge.net, site documentation (`http://www.sourceforge.net/docman/?group_id=1`), „6. CVS Instructions".

9.5.5 Ustawianie `ssh-agent`

Bezpieczniej jest chronić swój klucz SSH hasłem. Jeśli nie zostało ono ustawione, użyj polecenia `ssh-keygen -p` aby to zmienić.

Umieść swój klucz publiczny (np `~/.ssh/id_rsa.pub`) w pliku `~/.ssh/authorized_keys` na zdalnym hoście, używając połączenia opartego o hasła, tak jak to opisano w 'Połączenia z mniejszą ilością haseł – RSA' na poprzedniej stronie.

```
$ ssh-agent bash # lub uruchom zamiast tego  zsh/tcsh/pdksh
$ ssh-add ~/.ssh/id_rsa
Enter passphrase for /home/osamu/.ssh/id_rsa:
Identity added: /home/osamu/.ssh/id_rsa (/home/osamu/.ssh/id_rsa)
$ scp foo user@remote.host:foo
 ... nie potrzebne tu jest moje hasło :-)
$^D
 ... przerywanie sesji ssh-agent
```

Dla X-serwera, standardowe skrypty startowe systemu Debian uruchamiają `ssh-agent` jako proces macierzysty. Musisz więc jedynie uruchomić jednokrotnie `ssh-add`.

Dla dalszych informacji przeczytaj `ssh-agent(1)` i `ssh-add(1)`.

9.5.6 Rozwiązywanie problemów z SSH

Jeśli masz problemy, sprawdź prawa dostępu do plików konfiguracyjnych i uruchom `ssh` z opcją „`-v`”.

Użyj parametru „`-P`” jeśli jesteś rootem i masz problem z firewallem; pozwala to uniknąć korzystania z portów 1–1023.

Jeśli zdalne połączenie `ssh` nagle przestaje działać, może być to spowodowane działaniami administratora, a najbardziej, prawdopodobną zmianą w `host_key`. Po upewnieniu się, że nie jest to jednak sprytna próba włamania poprzez podszywanie się za zdalnej maszyny, możesz odzyskać połączenie usuwając pozycje `host_key` z `$HOME/.ssh/known_hosts` na lokalnej maszynie.

9.6 Poczta

Konfiguracja poczty dzieli się na 3 kategorie:

- mail transfer agent (program przesyłający pocztę - MTA): `exim`, `postfix`, `sendmail`, `qmail`, `ssmtp`, `nullmailer`,…
- pocztowe programy użytkowe: `procmail`, `fetchmail`, `mailx`,…
- mail user agent (klient poczty - MUA): `mutt`, `emacs+gnus`,…

9.6.1 Agenci transportu poczty (MTA)

Aby mieć w pełni wyposażony MTA użyj `exim`. Zasoby:
- pakiety `exim-doc` i `exim-doc-html`
- `http://www.exim.org/`

Jeśli troszczysz się o bezpieczeństwo, to jedyną rozsądną alternatywą jest `postfix`. `sendmail` i `qmail`, są również dostępne jako pakiety systemu Debian, ale ich używanie nie jest polecane.

Jeśli nie potrzebujesz „relay capability” MTA, jak to jest w przypadku np. laptopów, możesz rozważyć użycie rozwiązań mniejszego kalibru:
- `ssmtp`: wymaga połączenia SMTP i obsługuje aliasy, lub
- `nullmailer`: potrafi kolejkować, ale nie obsługuje aliasów.

Obecnie uważam, że `exim` jest bardziej odpowiedni nawet dla mojej stacji roboczej, będącej właśnie laptopem.

Musisz usunąć `exim` aby zainstalować `w/w` pakiety, gdyż zachodzi pomiędzy nimi konflikt:

```
# dpkg -P --force-depends exim
# apt-get install nullmailer        # lub ssmtp
```

Smarthost

Jeśli używasz `exima` na komputerze, który jest podłączony poprzez jakieś łącze konsumenckie, upewnij się, że wysyłasz pocztę przez tak zwany smarthost oferowany przez Twojego dostawcę lub innych. [6] Jest kilka dobrych powodów by tak robić:

- by upewnić się, że list doszedł, ponieważ serwer Twojego dostawcy najczęściej posiada lepsze połączenie.
- by zapobiec przesyłania wiadomości bezpośrednio z **dynamicznego adresu IP**, który prawdopodobnie zostanie zablokowany przez listy spamerskie połączeń wdzwanianych.
- by zaoszczędzić przepustowość połączenia lokalnego w przypadku wysyłania wiadomości do wielu odbiorców.

Jedynymi dopuszczalnymi wyjątkami są:

- sytuacje awaryjne, jeśli usługa SMTP Twojego dostawcy nie działa.
- eksperymenty w celach edukacyjnych.
- posiadanie komputera z profesjonalnym hostingiem.

Podstawowa konfiguracja Exima

Aby używać `exim` jako MTA, ustaw odpowiednie opcje w:

```
/etc/exim/exim.conf      "‘eximconfig"’  tworzenie i edycja
/etc/inetd.conf          zakomentuj smtp by używać serwera exim w trybie demon
/etc/email-addresses     Dodawanie listy fałszywych adresów
sprawdź filtry, używając exim -brw, -bf, -bF, -bV, ... etc.
```

Wyłapywanie poczty dla nieistniejących adresów w Eximie

W `/etc/exim/exim.conf` (Woody lub później), w części DIRECTORS, pod koniec (za 'lokalny użytkownik: przekierowanie') dodaj przekierowanie pasujące do wszystkich adresów, które poprzednie przekierowania nie mogły objąć (przez Miquela van Smoorenburga):

```
catchall:
  driver = smartuser
    new_address = webmaster@mydomain.com
```

Jeśli chce się mieć bardziej szczegółowy opis każdej wirtualnej domeny itp. dodaj następujące linie pod koniec pliku `/etc/exim/exim.conf` (nie sprawdzone).

---

[6] Musisz przestrzegać tej reguły dla każdego komputera podłączonego przez łącze wdzwaniane, DSL, telewizję kablową czy sieć lokalną z routerem. Nawet jeśli posiadasz publiczne IP od swojego dostawcy, nadal powinieneś przestrzegać tej reguły. Większość stacji i serwerów domowych podpada pod tę kategorię.

```
*@yourdomain.com ${lookup{$1}lsearch*{/etc/email-addresses} \
        {$value}fail} T
```

Następnie umieść pozycje „*” w pliku `/etc/email-addresses`.

Zamiana adresów dla poczty wychodzącej poczty w Eximie

Selektywna zamiana adresów dla wychodzącej poczty, aby otrzymać poprawnie nagłówki „From: ”, może być osiągnięta w programie `exim` dzięki odpowiedniej konfiguracji pod koniec pliku `/etc/exim/exim.conf`:

```
*@komp1.gdzieś.dyndns.org \
  "${if eq {${lookup{$1}lsearch{/etc/passwd}{1}{0}}} {1}  \
   {$0}{$1@gdzieś.dyndns.org}}"  frFs
```

Ta regułka zamienia wszystkie adresy pasujące do `*@komp1.gdzieś.dyndns.org`.

1 Najpierw sprawdza w pliku `/etc/password` czy część lokalna ($1) jest użytkownikiem czy nie.

2 Jeśli to lokalny użytkownik, przepisuje adres w takiej postaci w jakiej występował.

3 Jeśli nie, podmienia nazwę domeny.

Konfigurowanie autoryzacji SMTP w Eximie

Niektórzy usługodawcy SMTP, jak np. yahoo.com, wymagają autoryzacji SMTP. Skonfiguruj `/etc/exim/exim.conf` następująco:

```
remote_smtp:
  driver = smtp
  authenticate_hosts = smtp.mail.yahoo.com
...

smarthost:
  driver = domainlist
  transport = remote_smtp
  route_list = "* smtp.mail.yahoo.com bydns_a"
...

plain:
  driver = plaintext
  public_name = PLAIN
  client_send = "^cmatheson3^this_is_my_password"
```

Nie zapomnij o dwóch znakach cudzysłowia w ostatniej lini

9.6.2 Pobieranie poczty – Fetchmail

`fetchmail` jest uruchamiany w trybie demona, aby pobrać pocztę z konta POP3 u ISP do lokalnego systemu pocztowego. Konfiguracja:

```
/etc/init.d/fetchmail
/etc/rc?.d/???fetchmail uruchom update-rc.d fetchmail default priority 30
/etc/fetchmailrc        plik konfiguracyjny (chown 600, własność fetchmaila)
```

Uruchomienie `fetchmaila` jako demona wywoływanego ze skryptu `init.d` w Potato może być kłopotliwe (w Woodym jest już inaczej). Zobacz przykładowe skrypty (`http://www.debian.org/doc/manuals/debian-reference/examples/`).

Jeśli nagłówki poczty elektronicznej są zanieczyszczone znakiem ^M z winy Twojego ISP, dodaj „stripcr" do opcji w pliku `$HOME/.fetchmailrc`:

```
options fetchall no keep stripcr
```

9.6.3 Przetwarzanie poczty – Procmail

`procmail` spełnia rolę lokalnego MDA oraz programu filtrującego. Należy stworzyć plik `$HOME/.procmailrc` dla każdego konta, które z niego korzysta. Przykłady _procmailrc (`http://www.debian.org/doc/manuals/debian-reference/examples/`)

9.6.4 Odczytywanie poczty – Mutt

Użyj `mutt` jako czytnika poczty (mail user agent (MUA)) w połączeniu z `vim`. Dostosuj do własnych potrzeb edytując plik`~/.muttrc`; na przykład:

```
# używaj trybu wizualnego i "‘gq"’ do formatowania cytatów
set editor="vim -c ’set tw=72 et ft=mail’"
#
# usuwanie nagłówków wzięte z podręcznika (Sven’s Draconian header weeding)
#
ignore *
unignore from: date subject to cc
unignore user-agent x-mailer
hdr_order from subject to cc date user-agent x-mailer
auto_view application/msword
....
```

Użyj następujących wpisów w plikach `/etc/mailcap` lub `$HOME/.mailcap`, aby przeglądać załączniki w formatach HTML i MS Word.

```
text/html; lynx -force_html %s; needsterminal;
application/msword; /usr/bin/antiword '%s'; copiousoutput;
description="Microsoft Word Text"; nametemplate=%s.doc
```

9.7 Lokalizacja

Debian jest projektem międzynarodowym, obsługującym rosnącą ilość języków itp. Ta część wymieni kilka obszarów różnic językowych, które Debian obecnie obsługuje. Następne zaś skupią się na **lokalizacji**, procesie uzyskania wejścia i wyjścia programów w wybranym języku, jak również stosowania wybranych konwencji drukowania daty, liczb i symboli monetarnych, jak również innych różnic, jakie mogą występować na skutek stosowania systemu Debian w Twoim regionie.

9.7.1 Podstawy lokalizacji

Jest kilka aspektów lokalizowania systemu Debian i obsługi języków narodowych.

Lokalizacja klawiatury

Debian jest rozpowszechniany z blisko dwoma tuzinami map klawiatur. W Woodym możesz zmienić ustawienia klawiatury poprzez:

- `dpkg-reconfigure --priority=low console-data # console`
- `dpkg-reconfigure --priority=low xserver-xfree86 # XF4`
- `dpkg-reconfigure --priority=low xserver-common-v3 # XF3`

Lokalizacja danych

Szeroka gama oprogramowania dostępnego jako pakiety Debiana wspiera obsługę znaków graficznych różnych od standardu US-ASCII poprzez zmienną środowiskową LC_CTYPE, udostępnianą przez mechanizm **locale** w glibc

- 8-bit clean: praktycznie wszystkie programy
- inne zestawy znaków Latin (np. ISO-8859-1 lub ISO-8859-2): większość programów
- wielobajtowe języki, jak Chiński, Japoński czy Koreański: wiele nowych programów

Lokalizacja wyświetlania

X-y mogą wyświetlać każde kodowanie, włączając UTF-8, i obsługują wszystkie czcionki. Zawierają się w tym nie tylko wszystkie czcionki 8-bitowe, ale także 16-bitowe - np. chińska, japońska czy koreańska. Obsługa wielobajtowych zestawów znaków przy wprowadzaniu danych jest zapewniona przez mechanizm 'Alternatywne metody wejścia X' na 181 stronie.

Zobacz 'Przykład wielojęzykowego systemu X window' na 178 stronie i 'Obsługa UTF-8 w emulatorach terminali X' na 182 stronie.

Wyświetlanie japońskiego kodu EUC jest także możliwe w konsoli graficznej, poprzez pakiet `kon2`. Jest także alternatywny program, używający konsoli framebuffera, pakiet `jfbterm`. W tych środowiskach sposób zapewnienia wejścia kompatybilnego z językiem japońskim musi być zapewniony przez programy użytkowe. Użyj pakietu `egg` dla Emacsa lub `jvim` dla Vima.

Instalacja czcionek innych niż Unicode dla X pomoże wyświetlać dokumenty z jakimkolwiek kodowaniem. Więc nie przejmuj się zbytnio kodowaniem czcionek.

Lokalizacja komunikatów i dokumentacji

Dostępne są tłumaczenia wielu komunikatów i dokumentów wyświetlanych w systemie Debian takich, jak komunikaty błędów, standardowe wyjście programów, menu oraz strony podręcznika systemowego. Obecnie istnieją strony podręcznika w języku niemieckim, hiszpańskim, fińskim, francuskim, węgierskim, włoskim, japońskim, koreańskim, polskim, portugalskim, chińskim oraz rosyjskim - wszystkie one są dostępne w pakietach `manpages-`*`LANG`* (gdzie *LANG* jest dwuliterowym kodem ISO kraju. Użyj `apt-cache search manpages-|less` aby otrzymać listę dostępnych pakietów.)

Aby korzystać z przetłumaczonych stron podręcznika, użytkownik musi ustawić zmienną środowiskową LC_MESSAGES na odpowiednią wartość. W przypadku języka włoskiego LC_MESSAGES musi zostać ustawione na `it`. Wtedy `man` będzie szukał włoskiego tłumaczenia w pliku `/usr/share/man/it/`.

9.7.2 Locales

Debian oferuje mechanizm **locale**. Pozwala on programom na dostosowanie wyjścia i możliwości w zależności od ustawień systemowych takich, jak używany zestaw znaków, format wyświetlania daty i godziny, symbol waluty itp. Na przykład, zakładając występowanie zarówno angielskich (amerykańskich) jak i niemieckich locales w Twoim systemie, komunikat błędu może być wyświetlany w wielu językach:

```
$ LANG="en_US" cat foo
cat: foo: No such file or directory
$ LANG="de_DE" cat foo
cat: foo: Datei oder Verzeichnis nicht gefunden
```

Glibc oferuje tą funkcjonalność dla programów jako bibliotekę. Zobacz `locale(7)`.

9.7.3 Wprowadzenie do locales

Pełny opis locale składa się z trzech części: `xx_YY.ZZZZ`.

- **`xx`**: kod języka w ISO 639 (małe litery)
- **`YY`**: kod kraju w ISO 3166 (wielkie litery)
- **`ZZZZ`**: zestaw kodów, np. ustawienia znaków lub identyfikator kodowania.

Kody języków i krajów możesz znaleźć w opisie w `info gettext`.

Zwróć uwagę, że część opisująca kody może zostać wewnętrznie znormalizowana poprzez usunięcie wszystkich znaków – i zamianę liter na małe w celu większej kompatybilności, Typowe zestawy kodów to:

- **UTF-8**: Unikod (Unicode) dla wszystkich regionów, głównie w 1-3 oktetach (de facto nowy standard)
- **ISO-8859-1**: zachodnia Europa (de facto stary standard)
- **ISO-8859-2**: wschodnia Europa (bośniacki, chorwacki, czeski, węgierski, polski, rumuński, serbski, słowacki, słoweński)
- **ISO-8859-3**: maltański
- **ISO-8859-5**: macedoński i serbski
- **ISO-8859-6**: arabski
- **ISO-8859-7**: grecki
- **ISO-8859-8**: hebrajski
- **ISO-8859-9**: turecki
- **ISO-8859-11**: tajski (=TIS-620)
- **ISO-8859-13**: łotewski, litewski, maori
- **ISO-8859-14**: walijski
- **ISO-8859-15**: zachodnia Europa z obsługą euro
- **KOI8-R**: rosyjski
- **KOI8-U**: ukraiński
- **CP1250**: czeski, węgierski, polski (MS Windows)
- **CP1251**: bułgarski, białoruski (MS Windows)
- **eucJP**: japoński w stylu Unix (=ujis)
- **eucKR**: koreański w stylu Unix
- **GB2312**: uproszczony chiński w stylu Unix (=GB, =eucCN) dla zh_CN
- **Big5**: tradycyjny chiński dla zh_TW
- **sjis**: japoński w stylu Microsoft (Shift-JIS)

Znaczenie podstawowych systemów kodowania:

- **ASCII**: 7 bitów (0-0x7f)
- **ISO-8859-?**: 8 bitów (0-0xff)
- **ISO-10646-1**: Universal Character Set (UCS) (31 bitów, 0-0x7fffffff)
- **UCS-2**: Pierwsze 16 bitów UCS jako proste 2 oktety (Unicode: 0-0xffff)
- **UCS-4**: UCS jako proste 4 oktety (UCS: 0-0x7fffffff)
- **UTF-8**: UCS zakodowany w 1-6 oktetach (głównie w 3 oktetach)
- **ISO-2022**: 7 bitów (0-0xff) z sekwencjami escape. ISO-2022-JP jest najpopularniejszym kodowaniem dla japońskich wiadomości.
- **EUC**: 8 bitów + 16 bitów kombinacji (0-0xff), styl Unix
- **Shift-JIS**: 8 bitów + 16 bitów kombinacji (0-0xff), styl Microsoft.

ISO-8859-?, EUC, ISO-10646-1, UCS-2, UCS-4 i UTF-8 dla znaków 7 bitowych współdzielą te same kody wraz z ASCII. EUC i Shift-JIS używa znaków wysokich bitów (0x80-0xff) dla zaznaczenia, że część kodowania jest 16 bitowa. UTF-8 również używa znaków wysokich bitów

(0x80-0xff) by zaznaczyć sekwencje znaków spoza 7 bitów i jest to najbardziej powszechny system kodowania znaków spoza ASCII.

Zwróć uwagę na różnicę w kolejności bajtów w implementacjach Unikodu:

- **Standard UCS-2, UCS-4**: big endian
- **Microsoft UCS-2, UCS-4**: little endian dla ix86 (zależne od komputera)

Więcej możesz dowiedzieć się we Wprowadzeniu do i18n (`http://www.debian.org/doc/manuals/intro-i18n/`).

9.7.4 Aktywacja wsparcia locale

Debian **nie** jest instalowany z wszystkimi możliwymi prekompilowanymi wersjami locales . Sprawdź plik `/usr/lib/locale`, aby zobaczyć, które locale (poza domyślnym „C"), są obecne w Twoim systemie. Jeśli potrzebujesz takiego, który nie jest dostępny masz dwie możliwości:

- Dodaj pożądane locale do pliku `/etc/locale.gen`, następnie uruchom `locale-gen` jako root aby je skompilować. Przeczytaj `locale-gen(8)` i strony wymienione w jego części „SEE ALSO".
- Uruchom `dpkg-reconfigure locales`, aby ponownie skonfigurować pakiet `locales`. A jeśli nie jest jeszcze zainstalowany, to instalacja wywoła interfejs programu debconf, aby pozwolić Ci wybrać potrzebne locale i skompilować bazę danych.

9.7.5 Uaktywnienie określonego locale.

Następujące zmienne środowiskowe są używane, aby dostarczyć ustawień locale dla programów:

1 LANGUAGE: Ta zmienna składa się z oddzielonej dwukropkami listy locale w porządku ważności. Używaj wyłącznie, jeśli POSIX locale jest ustawione do wartości innej niż „C" [w Woodym; wersja Potato jest zawsze ważniejsza od POSIX locale]. (rozszerzenie GNU)

2 LC_ALL: Jeśli nie przyjmuje wartości NULL, wartość ta jest użyta dla categories. (POSIX.1) Zwykle „" (null).

3 LC_*: Jeśli nie przyjmuje wartości NULL, wartość ta jest użyta dla odpowiedniej kategorii (POSIX.1). Zwykle „C".

Dostępne zmienne LC_*:

- LC_CTYPE: Konwersja czcionki i wielkości liter.
- LC_COLLATE: Porządek sortowania.
- LC_TIME: Format wyświetlania daty i godziny.
- LC_NUMERIC: Wyświetlanie liczb nie związanych z walutą
- LC_MONETARY: Formaty walutowe.
- LC_MESSAGES: Format wiadomości informacyjnych, diagnostycznych oraz określających interakcje programu.
- LC_PAPER: Rozmiar papieru.

- LC_NAME: Format nazw.
- LC_ADDRESS: Format wyświetlania adresu i lokalizacji.
- LC_TELEPHONE: Format wyświetlania numeru telefonu.
- LC_MEASUREMENT: Jednostki miary (metryczna lub inna)
- LC_IDENTIFICATION: Metadata o ustawieniach locale.

4 LANG: Jeśli nie przyjmuje wartości NULL i LC_ALL jest niezdefiniowane, wartość wszystkich niezdefiniowanych kategorii LC_* jest ustawiana na tą wartość. (POSIX.1) Zwykle „C".

Zauważ, że niektóre programy (np. Netscape 4) ignorują ustawienia LC_*.

Program `locale` może wyświetlać ustawienia locale i dostępne locale; przeczytaj `locale(1)`. (Zapamiętaj: `locale -a` podaje wszystkie locale, o których istnieniu wie; *nie* oznacza to, że wszystkie są dostępne! Zobacz 'Aktywacja wsparcia locale' na sąsiedniej stronie.)

9.7.6 Format daty ISO 8601

Locale obsługuje międzynarodowy standard formatu daty `yyyy-mm-dd` (ISO 8601) poprzez locale `en_DK`, — Angielski w Danii, co jest żartem :-) Wydaje się działać jedynie w trybie tekstowym w `ls`.

9.7.7 Przykład dla US (ISO-8859-2)

Dodaj poniższe linie do pliku `~/.bash_profile`:

```
LC_CTYPE=en_US.ISO-8859-1
export LC_CTYPE
```

9.7.8 Przykład dla Francji, z walutą Euro (ISO-8859-15)

Dodaj poniższe linie do pliku `~/.bash_profile`:

```
LANG=fr_FR@euro
export LANG
LC_CTYPE=fr_FR@euro
export LC_CTYPE
```

Skonfiguruj klawiaturę jako francuskie „AZERTY", jak to opisano w 'Lokalizacja klawiatury' na 173 stronie i dodaj francuskie strony podręcznika systemowego instalując `manpages-fr`. Prawy Alt w US to Alt-Gr w Europie. Naciśnięcie tych klawiszy razem z innymi tworzy liczne znaki diakrytyczne oraz znaki specjalne. Np Alt-Gr+E odpowiada za znak Euro.

Większość języków zachodnio-europejskich może być skonfigurowana podobnie.

Zobacz Debian Euro HOWTO (`http://www.debian.org/doc/manuals/debian-euro-support/`), jak dodać obsługę nowej waluty Euro oraz Utiliser et configurer Debian pour le français (`http://www.debian.org/doc/manuals/fr/debian-fr-howto/`) po więcej informacji w języku francuskim.

9.7.9 Przykład wielojęzykowego systemu X window

Ustawmy wielojęzykowy system X window, który będzie jednocześnie obsługiwał japoński, angielski, niemiecki i francuski z kodowaniami EUC, UTF-8 i ISO-8859-1 w różnych konsolach.

Pokażę sposób dostosowania z użyciem systemu menu Debiana. Szczegóły tego systemu możesz znaleźć w `/usr/share/doc/menu/html/index.html`. W przykładzie stworzę również skrót do przeglądarki `mozilla`. [7]

- dodaj obsługę japońskiego ja_JP.eucJP oraz reszty potrzebnych locale metodą opisaną w 'Lokalizacja' na 173 stronie (dla wszystkich).
- zainstaluj system konwersji oraz słownik Kana-to-Kanji (dla japońskiego):
  - `canna` – lokalny serwer (licencja „free beer"), lub
  - `freewnn-jserver` – sieciowy serwer (Public Domain)
- zainstaluj mechanizmy japońskiego wejścia (dla japońskiego):
  - `kinput2-canna`– dla X-ów, lub
  - `kinput2-canna-wnn`– dla X-ów, i
  - `egg`– działa bezpośrednio z Emacsem nawet w trybie tekstowym (opcjonalnie).
- terminale odpowiednio przystosowane (dla wszystkich):
  - `xterm` – X (dla ISO-8859-1 i UTF-8),
  - `kterm` – X (dla japońskiego EUC) i
  - `mlterm` – X (wielojęzykowy).
- dodaj wszystkie wymagane czcionki (dla wszystkich)
- stwórz plik `~/.xsession` dla ustawień zależnych od użytkownika tak jak to opisano w 'Personalizowane sesje X.' na 155 stronie (dla wszystkich):

```
#!/bin/sh
# To sprawia, że X-y działają gdy przeloguję się na roota za pomocą su.
if [ -z "$XAUTHORITY" ]; then
        XAUTHORITY=$HOME/.Xauthority
        export XAUTHORITY
```

---

[7] W tym przykładzie zostały wykorzystane dwa obejścia błędów sytępujących w menedżerze okien `blackbox` w roku 2003. Używam `sh -c` w poleceniu. Również wpis `~/.menu/*` jest nieużywany, a w zamian wykorzystane zostało `/etc/menu/*`.

```
fi

# Ustawianie odpowiedniego środowiska przez system menu Debiana.
# Resetowanie locale
unset LC_CTYPE LC_NUMERIC LC_TIME LC_COLLATE LC_MONETARY LC_MESSAGES
unset LC_PAPER LC_NAME LC_ADDRESS LC_TELEPHONE LC_MEASUREMENT
unset LC_IDENTIFICATION LC_ALL LANG LANGUAGE PAGER
# ustawienie domyślnego locale w X
LANG=C
# eksportowanie locale
export LC_CTYPE LC_NUMERIC LC_TIME LC_COLLATE LC_MONETARY LC_MESSAGES
export LC_PAPER LC_NAME LC_ADDRESS LC_TELEPHONE LC_MEASUREMENT
export LC_IDENTIFICATION LC_ALL LANG LANGUAGE PAGER
###
# uaktywnianie japońskiej metody wejściowej z kinput2
kinput2 &
XMODIFIERS="@im=kinput2"
export XMODIFIERS
# A może blackbox window manager (leciutki)
exec /usr/bin/blackbox
#exec xfwm
#exec wmaker
```

- ustaw locale dla konsoli w `~/.bash_profile` (dla wszystkich)
- jeśli istnieją to usuń ustawienia locale z `~/.bashrc` (dla wszystkich)
- utwórz kilka plików w `/etc/menu/` (dla wszystkich).
  - `/etc/menu/xterm-local`: (dodaj nowe wpisy do menu) [8]

```
?package(xterm):\
 needs=x11\
 section=XShells\
 longtitle="XTerm: terminal emulator (en_US.ISO-8859-1)"\
 title="XTerm (en_US.ISO-8859-1)"\
 command="sh -c 'LC_ALL=en_US.ISO-8859-1 xterm'"
?package(xterm):\
 needs=x11\
 section=XShells\
 longtitle="XTerm: terminal emulator (de_DE.ISO-8859-1)"\
 title="XTerm (de_DE.ISO-8859-1)"\
 command="sh -c 'LC_ALL=de_DE.ISO-8859-1 xterm -T xterm-de'"
?package(xterm):\
 needs=x11\
 section=XShells\
```

---

[8] Użyj nazwy pliku, która nie konfliktuje z żadną nazwą pakiety.

```
 longtitle="XTerm: terminal emulator for X with Unicode support (Japan
 title="UXTerm (ja_JP.UTF-8)"\
 command="sh -c 'LC_ALL=ja_JP.UTF-8 uxterm'"
```

  - `/etc/menu/kterm`: (nadpisanie domyślnych ustawień) [9]

```
?package(kterm):\
 needs="x11"\
 section="XShells"\
 command="sh -c 'LC_ALL=ja_JP.eucJP PAGER=w3m /usr/X11R6/bin/kterm -xi
 title="Kanji Terminal"
?package(kterm):\
 needs="x11"\
 section="XShells"\
 command="sh -c 'LANG=ja_JP.eucJP \
      LC_MESSAGES=en_US.ISO-8859-1 PAGER=w3m /usr/X11R6/bin/kterm -xim
      title="Kanji Terminal (bilingal)"
```

  - `/etc/menu/mozilla-local`: (dodaj nowy skrót) [10]

```
?package(mozilla-browser):needs="x11" section="/" \
 title=" Mozilla Navigator" command="mozilla-1.5" hints="Web browsers"
  icon=/usr/share/pixmaps/mozilla.xpm
```

  - uruchom `update-menus` z konta administratora.

- dodaj następujące linie do `~/.muttrc` (dla japońskiego):

```
# Obsługa UTF-8 nie jest popularna w powszechnym japońskim środowisku EMACS
# 7 bitowe kodowanie iso-2002-jp jest prostsze dla wszystkich.
# domyślna kolejność kodowań = us-ascii --> iso-8859-1 --> iso-2022-jp
set send_charset="us-ascii:iso-8859-1:iso-2022-jp"
set allow_8bit=no
```

- uaktywnij XIM `kinput2` dla aplikacji w X-ach (dla japońskiego)
  - dodaj `*inputMethod: kinput2` i `KTerm*VT100*OpenIm: true` do Twojego pliku `~/.Xresources` (wygląda na to, że Debian załatwia to jakoś automatycznie).
  - Niektóre programy, jak np. `mlterm` pozwalają także na ustawianie `*inputMethod:` i innych informacji dynamicznie, podczas działania (naciśnij **Ctrl-MouseButton-3** w `mlterm`).

- uruchom X poprzez `startx` lub menedżer wyświetlania (`xdm`, `gdm`, `kdm`, `wdm`, …) (dla wszystkich).

- uruchom program kompatybilne z językiem japońskim takie, jak VIM6, (x)emacs21, mc-4.5, mutt-1.4, … w `kterm` (dla japońskiego). (Emacs wydaje się najpopularniejszy, aczkolwiek osobiście go nie używam).

---

[9] Użyj nazwy pliku konfliktującej z nazwą pakietu.

[10] Ukośnik w `section=„/”` uaktywnia wpis w menu głównym, a początkowa spacja w `title=„ Mozilla Navigator”` sprawia, że wpis znajduje się na górze listy.

- naciśnij **Shift+Space** aby włączyć/wyłączyć japoński tryb wejścia (dla japońskiego).
- przeczytaj przetłumaczone strony podręcznika systemowego wprowadzając polecenie w zlokalizowanej konsoli (dla wszystkich).

Obsługa innych języków CJK opisana jest w następnych rozdziałach oraz pod adresem SuSE pages for CJK (`http://www.suse.de/~mfabian/suse-cjk/suse-cjk.html`).

9.7.10 Alternatywne metody wejścia X

Jest wiele innych pakietów oferujących alternatywne metody wejścia dla X:

Język	LC_CTYPE	XIM serwer	XMODIFIERS	klawisz startowy
japoński	ja_JP*	kinput2	"@im=kinput2"	Shift-Space
koreański	ko_KR*	ami	"@im=Ami"	Shift-Space
chiński(T)	zh_TW.Big5	xcin	"@im=xcin-zh_TW.big5"	Ctrl-Space
chiński(S)	zh_CN.GB2312	xcin	"@im=xcin-zh_CN.GB2312"	Ctrl-Space

Japońska metoda wejściowa `kinput2` jest dostarczana przez pakiety `kinput2-canna-wnn`, `kinput2-canna` i `kinput2-wnn`. By były użyteczne wymagają serwerów słownikowych takich jak `canna` oraz `freewnn-jserver`.

9.7.11 Emulatory terminali X

Jest wiele konsoli X wspierających zwykłe kodowanie 8-bitowe jeśli odpowiednie pakiety z czcionkami są zainstalowane.

- `xterm` – The X terminal emulator
- `gnome-terminal` – `xterm` dla Gnome
- `konsole` – `xterm` dla KDE
- `rxvt` – VT102 terminal (lżejszy)
- `aterm` – VT102 dla Afterstep WM
- `eterm` – VT102 dla Enlightment WM
- `wterm` – VT102 dla WindowMaker WM

Wielobajtowe kodowania dla konsol X wspiera `xterm` poprzez kodowanie UTF-8 ('Obsługa UTF-8 w emulatorach terminali X' na następnej stronie). Nad wsparciem dla innych tradycyjnych kodowań trwają prace (2003). Następujące pakiety oferują wsparcie dla tradycyjnych kodowań:

- `aterm-ml` – wielojęzykowy
- `kterm` – wielojęzykowy (japoński, . . .)
- `rxvt-ml` – wielojęzykowy
- `wterm-ml` – wielojęzykowy
- `cxterm-big5` – chiński (Trad., Big5)
- `cxterm-gb` – chiński (Simp., GB)

- `cxterm-ks` – chiński (KS)
- `cxterm-jis` – japoński
- `hanterm-classic` – koreański (Hangul)
- `hanterm-xf` – koreański (Hangul)
- `hztty` – chiński (GB, Big5, zW/HZ)

Dla `kterm` (i prawdopodobnie innych), możesz uaktywnić XIM poprzez menu klikając środkowy klawisz myszy.

9.7.12 Obsługa UTF-8 w emulatorach terminali X

Dla XFree86 4.x obsługa UTF-8 w emulatorze terminala X dostarczana jest przez program `uxterm` z pakietu `xterm`. Uaktywnia wsparcie dla wszystkich języków. Jest on swego rodzaju nakładką nad `xterm(1)`, która wywołuje go z odpowiednio ustawionym zasobem X „UXTerm".

Przykładowo by uaktywnić wyświetlanie ładnych, dużych znaków angielskich, rosyjskich, japońskich, chińskich oraz koreańskich, po zainstalowaniu odpowiednich czcionek dodaj do swojego `~/.Xresources` następujące linie:

```
! set large font
UXTerm*font: -misc-fixed-medium-r-normal-*-18-120-100-100-c-90-iso10646-1
! Use XIM for Japanese
*inputMethod: kinput2
```

Następnie tak jak to opisano w 'Zasoby X' na 159 stronie, uruchom `xrdb -merge ~/.Xresources` by uaktualnić swoje zasoby X.

Większość popularnych programów konsolowych takich jak `vim`, `mutt` i `emacs` zostały ostatnio dostosowane do kodowania UTF-8 (Woody-Sarge). Jednakże program `mc` nadal nie jest zgodny z UTF-8, a tylko czysty 8-bitowo. Jeśli edytujesz 7-bitową część ASCII pliku o nieznanym lub mieszanym kodowaniu, to bezpieczniej jest używać oprogramowania niewspierającego locale.

Zobacz The Unicode HOWTO (`http://www.tldp.org/HOWTO/Unicode-HOWTO.html`).

9.7.13 Przykład UTF-8 w terminalu framebuffer

Wsparcie UTF-8 na terminalu FB jest zapewnione przez `bterm` używany w `debian-installer`.

9.7.14 Poza locales

Kiedy po raz pierwszy starasz się skonfigurować **narodowe środowisko językowe** zastanów się nad użyciem `tasksel` lub `aptitude`, aby dowiedzieć się jakie pakiety są instalowane, wybierając odpowiadające im funkcje lokalizacyjne. Wybór pakietów jest przydatny nawet w systemie wielojęzycznym. Jeśli napotkasz konflikty podczas instalacji z Twoim ostrożnie skonfigurowanym systemem, unikaj instalowania pakietu powodującego konflikt. Możliwe, że będziesz musiał użyć `update-alternative` aby przywrócić oryginalny stan dla niektórych poleceń, gdyż instalowane mogły mieć wyższy priorytet od istniejących.

Ważne nowe programy używają glibc 2.2 i przeważnie są umiędzynarodowione. Więc specjalnie zlokalizowane wersje, jak `jvim` dla VIM, mogą nie być potrzebne, gdyż wymagana funkcjonalność jest oferowana przez wersję 6.0 `vima` w X-ach. Chociaż z tym różnie bywa. Np. `jvim` posiada wersję skompilowaną z bezpośrednim wsparciem dla japońskiej metody wejściowej (`canna`) nawet w terminalu tekstowym i lepiej rozwiązuje niektóre problemy specyficzne dla japońskiego, więc wciąż możesz na niego reflektować :-)

Programy mogą potrzebować konfiguracji wykraczającej poza `locale` aby zapewnić komfortowe środowisko pracy. Pakiet `language-env` i jego polecenie `set-language-env` w istotny sposób ułatwiają ten proces.

Zobacz także dokument o internacjonalizacji, Introduction to i18n (`http://www.debian.org/doc/manuals/intro-i18n/`). Przeznaczony jest głównie dla programistów, ale przydatny może być również dla administratorów.

Rozdział 10

Konfiguracja sieci

Ten rozdział skupia się na administracji siecią w sytemie Debian. Ogólne wprowadzenie do sieci w GNU/Linuksie możesz przeczytać w Net-HOWTO (`http://www.tldp.org/HOWTO/Net-HOWTO/index.html`).

By system Debian mógł korzystać z internetu, jego interfejsy sieciowe muszą być obsługiwane przez jądro oraz poprawnie skonfigurowane.

Pierwszym wymaganiem jest obsługa przez jądro urządzenia sieciowego takiego jak karta sieciowa, karta Wi-Fi czy modem. By uzyskać tę obsługę, możliwe, że będziesz musiał przekompilować jądro lub załadować moduły tak jak to opisano w 'Jądro systemu Linux w Debianie' na 103 stronie.

Konfiguracja sieciowych urządzeń jest opisana poniżej. Informacje z tego rozdziału zostały zaktualizowane dla Sarge. Spora część nie dotyczy wcześniejszych wersji.

10.1 Podstawy sieciowe

Komputer z systemem Debian może posiadać wiele interfejsów, każdy z innym adresem IP. Interfejsy mogą być różnego typu, włączając:

- Loopback: `lo`
- Ethernet: `eth0,eth1,...`
- Wi-Fi: `wifi0,wifi1,...` [1]
- Token Ring: `tr0,tr1,...`
- PPP: `ppp0,ppp1,...`

Istnieje jeszcze wiele innych urządzeń sieciowych, włączając SLIP, PLIP (szeregowa i równoległa linia IP), urządzenia „shaper” służące do kontroli przepustowości na innych interfejsach, frame relay, AX.25, X.25, ARCnet, i LocalTalk.

---

[1] Zauważ, że interfejs Wi-Fi jest tak właściwie aliasem na interfejs Ethernet, który umożliwia dostęp do parametrów konfiguracyjnych specyficznych dla Wi-Fi. Parametry te są kontrolowane programem `iwconfig`.

Każdy interfejs sieciowy podłączony bezpośrednio do internetu (albo sieci bazującej na IP) jest identyfikowany za pomocą unikalnego 32 bitowego adresu IP. [2] Adres IP może być podzielony na część, która określa sieć oraz część określająca komputer. Jeśli weźmiesz adres IP, ustawisz na 1 wszystkie bity, które są częścią adresu sieci i na 0 wszystkie bity, które są częścią adresu komputera, to otrzymasz maskę sieciową danej sieci.

Dawniej, sieci IP były pogrupowane w klasy, których część sieciowa adresu była długa na 8, 16 lub 24 bity. [3]

	Adresy IP	maska sieci	długość
Klasa A	1.0.0.0 - 126.255.255.255	255.0.0.0	= /8
Klasa B	128.0.0.0 - 191.255.255.255	255.255.0.0	= /16
Klasa C	192.0.0.0 - 223.255.255.255	255.255.255.0	= /24

Adresy IP spoza tych zakresów są używane do specjalnych celów.

W każdej klasie istnieją zakresy adresów zarezerwowanych do użycia w sieciach lokalnych (LAN). Istnieje gwarancja, że te adresy nie będą w konflikcie z innymi adresami w internecie. (Z tego samego powodu, jeśli jeden z takich adresów jest przydzielony do komputera, ten komputer nie może mieć bezpośredniego dostępu do internetu, a tylko poprzez bramkę, która pracuje jako serwer pośredniczący dla usług lub wykonuje translację adresów (NAT)). W poniższej tabeli są wyszczególnione te zakresy adresów wraz z ich liczbą w każdej klasie.

	adres sieci	długość	ile
Klasa A	10.x.x.x	/8	1
Klasa B	172.16.x.x - 172.31.x.x	/16	16
Klasa C	192.168.0.x - 192.168.255.x	/24	256

Pierwszy adres w każdej sieci jest równocześnie adresem tej sieci. Ostatni adres jest adresem rozgłoszeniowym sieci (broadcast). [4] Wszystkie inne adresy mogą być przydzielane komputerom w sieci. Najczęściej pierwszy lub ostatni adres jest przydzielany bramce internetowej danej sieci.

Tablica routingu zawiera informacje jądra na temat tego jak przesyłać pakiety na ich miejsce przeznaczenia. Poniżej znajduje się przykładowa tablica routingu z komputera znajdującego się w sieci lokalnej o adresie 192.168.50.x/24. Komputer 192.168.50.1 (również w tej sieci) jest routerem dla sieci korporacyjnej 172.20.x.x/16, a komputer 192.168.50.254 (również w tej sieci) jest routerem dla internetu.

```
# route
```

---

[2] Jest to prawdą gdy używana jest 4 wersja protokołu IP. W IPv6 adresy są 128 bitowe. Zobacz `http://www.ipv6.org/`.

[3] Taki system był mało elastyczny i powodował marnotrastwo wielu adresów IP, więc dzisiaj sieci IPv4 są dzielone z adresami sieciowymi o różnych długościach.

[4] Adres sieci można otrzymać poprzez mnożenie logiczne jakiegokolwiek adresu z sieci z maską sieci. Adres rozgłoszeniowy można otrzymać przez sumowanie logiczne adresu sieci z dopełnioną jedynkami maską sieci.

```
Kernel IP routing table
Destination    Gateway          Genmask         Flags Metric Ref Use Iface
127.0.0.0      *                255.0.0.0       U     0      0     2 lo
192.168.50.0   *                255.255.255.0   U     0      0   137 eth0
172.20.0.0     192.168.50.1     255.255.0.0     UG    1      0     7 eth0
default        192.168.50.254   0.0.0.0         UG    1      0    36 eth0
```

- Pierwsza linia po nagłówkach mówi, że ruch przeznaczony dla sieci `127.x.x.x` będzie przesyłany poprzez `lo`, interfejs zwrotny.
- Druga linia mówi, że ruch przeznaczony dla komputerów z sieci lokalnej będzie przesyłany przez `eth0`.
- Trzecia linia mówi, że ruch przeznaczony dla sieci korporacyjnej będzie przesyłany przez bramkę `192.168.50.1` również przez `eth0`.
- Czwarta linia mówi, że ruch przeznaczony dla internetu będzie kierowany przez bramkę `192.168.50.254` również przez `eth0`.

Adresy IP w tablicy mogą pojawia się również jako nazwy, które są pobierane z pliku `/etc/networks` lub poprzez mechanizm rozwiązujący nazwy (C Library resolver).

Dodatkowo, oprócz routingu, jądro może wykonywać translację adresów, ograniczanie przepustowości czy filtrowanie.

Zobacz Net-HOWTO (`http://www.tldp.org/HOWTO/Net-HOWTO/index.html`) i other networking HOWTOs (`http://www.tldp.org/HOWTO/Networking-Overview-HOWTO.html`) po więcej informacji.

10.2 Zaawansowana konfiguracja sieci

Tradycyjne zaawansowane narzędzia do konfiguracji sieci w GNU/Linuksie to programy `ifconfig` i `route` znajdujące się w pakiecie `net-tools`. Te narzędzia zostały oficjalnie zastąpione przez `ip` z pakietu `iproute`. Program `ip` działa z Linuksem 2.2 lub nowszymi i jest bardziej dostosowany. Jednakże, stare narzędzia nadal działają i są znane większości użytkowników.

10.2.1 Zaawansowana konfiguracja sieci – `ifconfig` i `route`

Poniżej ilustracja jak zmienić adres IP interfejsu `eth0` z `192.168.0.3` na `192.168.0.111` oraz ustawić `eth0` by przesyłane były przez niego dane do sieci `10.0.0.0` poprzez `192.168.0.1`. Zaczynamy od uruchomienia `ifconfig` i `route` bez żadnych argumentów by wyświetlić aktualny stan wszystkich interfejsów oraz routing.

```
# ifconfig
eth0 Link encap:Ethernet  HWaddr 08:00:46:7A:02:B0
     inet addr:192.168.0.3  Bcast:192.168.0.255  Mask:255.255.255.0
     UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

```
          RX packets:23363 errors:0 dropped:0 overruns:0 frame:0
          TX packets:21798 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:13479541 (12.8 MiB)  TX bytes:20262643 (19.3 MiB)
          Interrupt:9

     lo   Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:230172 errors:0 dropped:0 overruns:0 frame:0
          TX packets:230172 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:22685256 (21.6 MiB)  TX bytes:22685256 (21.6 MiB)
     # route
     Kernel IP routing table
     Destination  Gateway      Genmask          Flags Metric Ref Use Iface
     192.168.0.0  *            255.255.0.0      U     0      0     0 eth0
     default      192.168.0.1  255.255.255.255  UG    0      0     0 eth0
```

Na początek wyłączamy interfejs.

```
# ifconfig eth0 inet down
# ifconfig
lo   Link encap:Local Loopback
  ... (brak pozycji eth0)
# route
  ... (brak wpisów w tabeli routingu)
```

Następnie przywracamy go z powrotem z nowym adresem IP i nowym routingiem.

```
# ifconfig eth0 inet up 192.168.0.111 \
           netmask 255.255.0.0 broadcast 192.168.255.255
# route add -net 10.0.0.0 netmask 255.0.0.0 gw 192.168.0.1 dev eth0
```

Rezultat:

```
# ifconfig
eth0 Link encap:Ethernet  HWaddr 08:00:46:7A:02:B0
     inet addr:192.168.0.111  Bcast:192.168.255.255  Mask:255.255.0.0
     UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
     ...

lo   Link encap:Local Loopback
     inet addr:127.0.0.1  Mask:255.0.0.0
```

```
    ...
# route
Kernel IP routing table
Destination  Gateway      Genmask      Flags Metric Ref Use Iface
192.168.0.0  *            255.255.0.0  U     0      0     0 eth0
10.0.0.0     192.168.0.1  255.0.0.0    UG    0      0     0 eth0
```

Więcej informacji uzyskasz w `ifconfig(8)` i `route(8)`.

10.2.2 Zaawansowana konfiguracja sieci – `ip`

Odpowiedniki poprzednich poleceń z wykorzystaniem `ip` zamiast `ifconfig` i `route` są:

- `ip link show`
- `ip route list`
- `ip link set eth0 down`
- `ip addr del dev eth0 local 192.168.0.3`
- `ip addr add dev eth0 local 192.168.0.111/16 broadcast 192.168.255.255`
- `ip link set eth0 up`
- `ip route add dev eth0 to 10.0.0.0/8 src 192.168.0.111 via 192.168.0.1`

Program `ip` wyświetla składnię swoich poleceń gdy jest uruchamiany z argumentem `help`. Przykładowo `ip link help` wyświetli:

```
Usage: ip link set DEVICE { up | down | arp { on | off } |
                            dynamic { on | off } |
                            multicast { on | off } | txqueuelen PACKETS |
                            name NEWNAME |
                            address LLADDR | broadcast LLADDR |
                            mtu MTU }
       ip link show [ DEVICE ]
```

Zobacz również `ip(8)`.

10.2.3 Konfiguracja interfejsu Wi-Fi

Dla interfejsów Wi-Fi wraz z `ifconfig` lub `ip` używany jest program `iwconfig` z pakietu `wireless-tools`.

Zobacz `iwconfig(8)`.

10.2.4 Konfiguracja interfejsu PPP

Jeśli korzystasz z internetu poprzez modem podpięty do wdzwanianej linii telefonicznej wtedy połączenie jest negocjowane przy pomocy protokołu Point-to-Point (PPP). Takie połączenia są dostępne jako interfejsy sieciowe `ppp0`, `ppp1` i tak dalej.

Interfejs PPP jest zarządzany przez demona PPP `pppd` z pakietu `ppp`. Z tego powodu dla użytkownika, konfigurować interfejs PPP znaczy konfigurację `pppd`.

Ręczna konfiguracja `pppd`

By zestawić połączenie sieciowe, port komunikacyjny (zazwyczaj port szeregowy) musi zostać otwarty, polecenia muszą zostać przesłane do urządzenia łączącego się (zazwyczaj modemu), numer telefoniczny musi zostać wybrany, tożsamość musi zostać zweryfikowana przez odległy demon PPP, interfejs PPP musi zostać utworzony i wtedy tablice routingu muszą zostać odpowiednio zmodyfikowane by ruch był wysyłany przez to połączenie. `pppd` może wykonać to wszystko, a w związku z tym posiada bardzo długą listę dostępnych opcji. Opcje te zostały opisane w `pppd(8)`.

W systemie Debian globalne opcje ustawia się w pliku `/etc/ppp/options`. Opcje specyficzne dla użytkownika można ustawić w `~/.ppprc`. Opcje zależne od portu komunikacyjnego są przechowywane w `/etc/ppp/options.`*`nazwa_portu`*. Przykładowo, załóżmy, że masz dwa modemy — wbudowany modem Lucent LT dostępny przez /dev/LT-modem i zewnętrzny modem dostępny przez /dev/ttyS0. Stwórz dwa poniższe pliki z opcjami.

```
# cat > /etc/ppp/options.LT-modem <<EOF
115200
init "/usr/sbin/chat -f /etc/chatscripts/setup-LT-modem"
EOF
# cat > /etc/ppp/options.ttyS0 <<EOF
115200
init "/usr/sbin/chat -f /etc/chatscripts/setup-ttyS0"
EOF
```

To odnosi się do następujących skryptów chat. Najpierw `/etc/chatscripts/setup-LT-modem`.

```
ABORT ERROR
"' ATZ
OK 'ATW2X2 S7=70 S11=55'
OK AT
```

A teraz `/etc/chatscripts/setup-ttyS0`.

```
ABORT ERROR
"' ATZ
OK 'ATL1M1Q0V1W2X4&C1&D2 S6=4 S7=70 S11=55 S95=63 S109=1 +FCLASS=0'
OK AT
```

Zawartość tych plików oczywiście zależy od Twojego sprzętu.

Opcje mogą również zostać przekazane do `pppd` jako argumenty.

W systemie Debian `pppd` jest zazwyczaj uruchamiany używając polecenia `pon`. Kiedy `pon` zostaje użyty, jego pierwszym argumentem jest nazwa pliku z opcjami znajdującego się w `/etc/ppp/peers/`, który będzie odczytany przez `pppd`. [5] To miejsce gdzie ustawiasz opcje specyficzne na przykład dla konkretnego połączenia — konkretnego dostawcy internetu (ISP).

Załóżmy przykłado że dojeżdżasz pomiędzy Amsterdamem i Den Haag. W każdym mieście masz dostęp do dwóch dostawców internetu — Planet i KPN. Najpierw utwórz podstawowy plik z opcjami dla każdego dostawcy.

```
# cat > /etc/ppp/peers/KPN <<EOF
remotename KPN
noauth
user kpn
noipdefault
ipparam KPN
EOF
# cat > /etc/ppp/peers/Planet <<EOF
remotename Planet
auth
user user3579@planet.nl
noipdefault
mru 1000
mtu 1000
ipparam Planet
EOF
```

Te pliki ustawiają różne opcje pomiędzy dostawcami. Wspólne opcje mogą być umieszczone w `/etc/ppp/options` lub w jednym z plików związanym z konkretnym interfejsem.

Teraz stwórz pliki z opcjami dla każdego dostawcy w każdym mieście. W naszym przykładzie jedyną różnicą w połączeniach do dostawców w różnych lokalizacjach będzie wymagany skrypt chat. (Skrypt chat jest różny ponieważ lokalny dostęp do numeru telefonicznego jest inny.)

```
# cat > /etc/ppp/peers/KPN-Amsterdam <<EOF
connect "/usr/sbin/chat -v -f /etc/chatscripts/KPN-Amsterdam"
```

---

[5] Ta opcja jest załączona używając opcji `call`.

```
file /etc/ppp/peers/KPN
EOF
# cat > /etc/ppp/peers/KPN-DenHaag <<EOF
connect "/usr/sbin/chat -v -f /etc/chatscripts/KPN-DenHaag"
file /etc/ppp/peers/KPN
EOF
# cat > /etc/ppp/peers/Planet-Amsterdam <<EOF
connect "/usr/sbin/chat -v -f /etc/chatscripts/Planet-Amsterdam"
file /etc/ppp/peers/Planet
EOF
# cat > /etc/ppp/peers/Planet-DenHaag <<EOF
connect "/usr/sbin/chat -v -f /etc/chatscripts/Planet-DenHaag"
file /etc/ppp/peers/Planet
EOF
```

Każda dyrektywa `file` załącza jeden z plików z opcjami pokazanych wcześniej. Dyrektywa `connect` określa polecenie z którego skorzysta `pppd` do nawiązania połączenia. Zazwyczaj używa się programu `chat` do tego odpowiednio dostosowując skrypt chat do dostawcy. Poniżej skrypt chat dla Den Haag; skrypt dla Amsterdamu może być podobny z wyjątkiem numeru telefonu lub może się znacznie różnić jeśli tam dostawca oferuje usługi poprzez inną firmę.

```
# cat > /etc/chatscripts/KPN-DenHaag <<EOF
ABORT BUSY
ABORT 'NO CARRIER'
ABORT VOICE
ABORT 'NO DIALTONE'
ABORT 'NO DIAL TONE'
ABORT 'NO ANSWER'
ABORT ERROR
OK-AT-OK ATDT 0676012321
CONNECT
EOF
# cat > /etc/chatscripts/Planet-DenHaag <<EOF
ABORT BUSY
ABORT 'NO CARRIER'
ABORT VOICE
ABORT 'NO DIALTONE'
ABORT 'NO DIAL TONE'
ABORT 'NO ANSWER'
ABORT ERROR
OK-AT-OK ATDT 0676002505
CONNECT
EOF
```

By móc połączyć się z tymi dostawcami potrzebujesz nazwę klienta oraz hasło któe `pppd` może wykorzystać przy połączeniu. Te informacje przechowywane są albo w `/etc/ppp`

`/pap-secrets` (jeśli używany jest protokół PAP) albo w `/etc/ppp/chap-secrets` (jeśli używany jest protokół CHAP). Mimo, że CHAP jest bardziej bezpieczny to PAP jest częściej używany. Ponieważ te pliki zawierają poufne informacje, grupa oraz inni nie powinni mieć uprawnień do czytania czy zapisywania ich. Format tych plików opisany jest w `pppd(8)`. Hasło (trzecie pole) jest wybierane na podstawie nazwy klienta (pierwsze pole) i/lub nazwy serwera (drugie pole). Podczas łączenia się z dostawcami generalnie nie jest znana nazwa serwera więc przesyłana jest w zamian nazwa klienta; jest to wykonywane w powyższych plikach w liniach `user` w `peers/KPN` i `peers/Planet`.

```
# nazwa klienta     nazwa serwera  hasło
kpn                 *              kpn
user3579@planet.nl  *              myfavoritepet
```

Konfiguracja `pppd` używając `pppconfig`

Szybkim sposobem na konfigurację `pppd` jest użycie programu `pppconfig` z pakietu o tej samej nazwie. `pppconfig` ustawia powyższe pliki po zadaniu użytkownikowi pytań w przyjaznym interfejsie.

Jeśli zamierzasz używać `resolvconf` upewnij się, że wybrałeś „None” na ekranie „Configure Nameservers”.

Konfiguracja interfejsu PPP używając `wvdial`

Innym sposobem na używanie `pppd` jest uruchamianie go z `wvdial` który znajduje się w pakiecie `wvdial`. Zamiast uruchamiać `pppd` wraz z `chat` do negocjacji i nawiązania połączenia, `wvdial` sam wdzwania się i negocjuje połączenie a następnie uruchamia `pppd` do reszty zadań. Podając tylko numer telefonu, nazwę użytkownika i hasło `wvdial` radzi sobie z nawiązaniem połączenia w większości przypadków.

10.3 Przydzielanie nazwy komputerowi

10.3.1 Nazwa komputera

System Debian czasem potrzebuje identyfikacji poprzez nazwę. Do tych celów **hostname** jest zarządzany przez jądro.

Skrypt startowy `/etc/init.d/hostname.sh` ustawia nazwę komputera podczas uruchamiania systemu (używając polecenia `hostname`) i przetrzymuje ją w `/etc/hostname`. Ten plik powinien zawierać **tylko** nazwę komputera, a nie jego pełną nazwę domenową.

By wyświetlić aktualną nazwę komputera uruchom `hostname` bez argumentów.

10.3.2 Nazwa pocztowa (mailname)

mailname danego komputera jest nazwą, której programy związane z pocztą elektroniczną używają do identyfikacji komputera. Plik `/etc/mailname` zawiera tę nazwę zakończoną nową linią. Nazwa pocztowa jest zazwyczaj jedną z pełnych domenowych nazw komputera (fully qualified domain name). Zobacz `mailname(5)`.

To co odbiorca wiadomości email wysłanej przez Twój system Debian widzi przy nagłówku `From:` zależy od tego jak serwer oraz klient pocztowy są skonfigurowane. Załóżmy, że lokalny użytkownik *`foo`* wysyła wiadomość z komputera o nazwie pocztowej *`mojkomputer.dom`*. W nagłówku `From:` wiadomości wychodzącej będzie:

- „`From:` *`foo`*`@`*`mojkomputer.dom`*" jeśli klient nie ma ustawionego nagłówka `From:`;
- „`From:` *`bar`*`@`*`mojkomputer.dom`*" jeśli klient ma ustawiony nagłówek „`From:` *`bar`*";
- „`From:` *`bar`*`@`*`bogus.dom`*" jeśli klient ma ustawiony nagłówek „`From:` *`bar`*`@`*`bogus.dom`*"

Nawet jeśli klient ma ustawiony nagłówek `From:`, serwer może dodać nagłówek „`Sender:`*`foo`*`@`*`herman.dom`*" tak by zaznaczyć prawdziwe źródło pochodzenia.

Oczywiście jeśli serwer podejmie próbę podmienienia adresu tak jak zostało to omówione w 'Wyłapywanie poczty dla nieistniejących adresów w Eximie' na 170 stronie i 'Zamiana adresów dla poczty wychodzącej poczty w Eximie' na 171 stronie, adres email widoczny dla odbiorcy może być zmieniony na cokolwiek.

10.4 Usługa nazw (DNS)

Komputery rozpoznawane są przez domenę tak samo jak przez adresy IP. DNS jest systemem typu klient-serwer, który zamienia nazwy na podstawie danych z serwerów nazw tak by nazwa była powiązana z adresem IP i innymi właściwościami komputera. Biblioteka GNU C `resolver(3)` może również poszukiwać adresów IP w plikach lub odwoływać się do Network Information Services (NIS).

By zobaczyć jaka nazwa domeny jest powiązana z lokalnym komputerem, użyj polecenia `hostname --fqdn`. Spowoduje to wyświetlenie pierwszej pełnej nazwy domenowej znalezionej dla lokalnego komputera przez program rozwiązujący nazwy. [6]

10.4.1 Resolver

Odszukiwaniem informacji o tym jaki adres IP jest powiązany z konkretną nazwą domeny zajmuje się resolver. Najczęściej używanym resolverem jest zbiór funkcji o tej nazwie (`resolver(3)`) w bibliotece GNU C. Innym jest FireDNS resolver z pakietu `libfiredns`.

To w jaki sposób resolver LIBC rozwiązuje nazwy jest zdefiniowane w lini `hosts` pliku konfiguracyjnego `/etc/nsswitch.conf`. Ta linia zawiera usługi, które powinny być użyte do

---

[6] Technicznie, jest to FQDN zwracany przez `gethostbyname(2)` dla komputera zwracanego przez `gethostname(2)`.

rozwiązania nazwy: np. `dns`, `files`, `nis`, `nisplus`. [7] Zobacz `nsswitch.conf(5)`. Dopóki używana jest usługa `files`, zachowanie resolvera definiuje również plik konfiguracyjny `/etc/hosts`. Zobacz `hosts(5)`.

Wszystkie wymienione pliki są statyczne i mogą być edytowane przy pomocy ulubionego edytora.

Dopóiki używana jest usługa `dns`, zachowanie resolvera definiuje również plik konfiguracyjny `/etc/resolv.conf`. Zobacz `resolv.conf(5)`. Jedną z najważniejszych funkcji `resolv.conf` jest lista adresów IP serwerów nazw z którymi będzie nawiązywany kontakt w celu rozwiązania nazwy. Ta lista często zależy od środowiska sieciowego, a ono może się zmieniać od czasu do czasu gdy komputer jest uruchomiony. Takie programy jak `pppd` i `dhclient` mają możliwość manipulowania plikiem `resolv.conf`, poprzez dodawanie i usuwanie linii, ale te funkcje nie zawsze działają poprawnie i konfliktują ze sobą. Pakiet `resolvconf` radzi sobie z tym problemem lepiej poprzez standard aktualizowania tego pliku. Zobacz 'Zarządzanie informacjami o serwerach nazw – `resolvconf`' na bieżącej stronie.

10.4.2 Zarządzanie informacjami o serwerach nazw – `resolvconf`

Pakiet `resolvconf` dostarcza standard do dynamicznego zarządzania informacją o dostępnych serwerach nazw. Rozwiązuje dawny problem zarządzania dynamiczną listą serwerów nazw dla resolvera i buforu DNS. Resolvconf ustawia się jako pośrednik między programami, które kontrolują interfejsy sieciowe i dostarczają informacji o serwerach nazw, a programami, które tych informacji potrzebują.

`resolvconf` jest zaprojektowany by działać, bez potrzeby jakiejkolwiek ręcznej konfiguracji. Jednakże pakiet ten jest raczej nowy i może wymagać ręcznej interwencji by działał poprawnie. To będzie z pewnością wymagane jeśli kiedykolwiek instalowałeś własne pakiety, które aktualizują `/etc/resolv.conf`: wtedy będziesz musiał wyłączyć te aktualizacje.

Szczegóły w `/usr/share/doc/resolvconf/README.gz`.

10.4.3 Buforowanie rozwiązanych nazw – `dnsmasq`

Jeżeli Twój serwer nazw nie musi być autorytatywnym dla domeny to lepiej włączyć buforowanie rozwiązywania nazw na przykład przy pomocy `dnsmasq`. Współpracuje dobrze z `resolvconf`.

10.4.4 Buforowanie rozwiązanych nazw – `bind`

Możesz również użyć `bind9` lub `binda` jako lokalny bufor rozwiązanych nazw. Również dobrze współpracuje z `resolvconf`, ale w momencie pisania tego (październik 2003), trzeba

---

[7] Przyjmuje się, że to w jaki sposób resolver rozwiązuje nazwy jest również zdefiniowane w pliku konfiguracyjnym `/etc/host.conf`. Linia `order` tego pliku zawiera listę metod, które powinny zostać użyte do rozwiązania nazwy: `bind`, `hosts`, `nis`. Zobacz `host.conf(5)`. Uważam, że ta linia została zastąpiona przez `hosts` `nsswitch.conf`, ale nie jestem pewien.

ustawiać go ręcznie. Instrukcje w `/usr/share/doc/resolvconf/README.gz`.

10.4.5 Udostępnianie DNS – `bind`

Jeśli musisz udostępniać autorytatywnych informacji o nazwach dla domeny, potrzebujesz pełnego serwera nazw, takiego jak `named`, który znajduje się albo w pakiecie `bind9` albo `bind`. Dla nowych instalacji zalecany jest `bind9`.

By zainstalować `bind9`, zainstaluj te podstawowe pakiety: `bind9`; `dnsutils`. Jeśli chcesz, możesz doinstalować również te pakiety z narzędziami: `bind9-host`; `dns-browse`; `dnscvsutil`; `nslint`. Jeśli chcesz, możesz doinstalować również pakiet z dokumentacją: `bind9-doc`. Jeśli chcesz, możesz doinstalować również pakiety deweloperskie: `libbind-dev`; `libnet-dns-perl`. Jeśli konfigurujesz interfejsy przy użyciu DHCP, to przydatn może okazać się pakiet: `dhcp-dns`.

Zainstaluj `bind9` lub przekonfiguruj go przy pomocy `dpkg-reconfigure` by dokonać podstawowych ustawień. Konfiguracja dokonuje się poprzez edycję pliku `named.conf`. W systemie Debian ten plik znajduje się w `/etc/bind/` i jest używany głównie do definicji podstawowych stref DNS; `zawiera` dwa inne pliki: `named.conf.local`, używany do definicji lokalnych stref, oraz `named.conf.options`, używany do ustawień opcji. Ten drugi jest przetwarzany przez `resolvconf` by utworzyć `/var/run/bind/named.options`, który jest identyczny z oryginalnym poza opcją `forwarders`, która zawiera listę aktualnie dostępnych nielokalnych serwerów nazw. By z niego skorzystać, zmień linię `include` w `named.conf` tak by zawierała `/var/run/bind/named.options`. Zobacz 'Zarządzanie informacjami o serwerach nazw – `resolvconf`' na poprzedniej stronie.)

Pliki z danymi wprowadzone w `named.conf*` bez pełnej ścieżki będą przetrzymywane w `/var/cache/bind/`. To prawidłowe miejsce na przetrzymywanie plików generowanych przez `nameda`: na przykład, plików z danymi dla stref dla których jest serwerem podrzędnym. Statyczne pliki z danymi w `/etc/bind/` są i muszą być wpisane w `named.conf` wraz z pełnymi ścieżkami. Szczegóły w `/usr/share/doc/bind9/README.Debian.gz`.

10.5 Konfiguracja interfejsów sieciowych z wykorzystaniem DHCP

Zaawansowana konfiguracja interfejsów sieciowych może być zautomatyzowana przez Dynamic Host Configuration Protocol (DHCP). Twój firewall lub router lub dostawca internetu może udostępniać adresy IP i inne parametry w ten sposób.

By skorzystać z tych usług musisz zainstalować jeden z następujących pakietów:
- `dhcp3-client` (wersja 3, Internet Software Consortium)
- `dhcpcd` (Yoichi Hariguchi i Sergei Viznyuk)
- `pump` (Redhat)

`pump` jest prosty i popularny. `dhcp3-client` jest bardziej skomplikowany, ale również ma większe możliwości konfiguracyjne. [8]

---

[8] W październiku 2003 roku dostępny jest również pakiet `dhcp-client`. Zawiera on wersję 2 klienta DHCP

10.6 Konfiguracja sieci wyższego poziomu w systemie Debian

W celu ułatwienia konfiguracji sieci, system Debian dostarcza standardowe narzędzia konfiguracyjne wyższego poziomu, na które składają się programy `ifup` i `ifdown` oraz plik `/etc/network/interfaces`. [9] Jeśli zdecydujesz się na użycie `ifupdown` do zarządzania konfiguracją swojej sieci, **nie** powinieneś jednocześnie używać poleceń niższego poziomu. [10] `Ifupdown` został napisany z myślą, że będzie używany samodzielnie do konfiguracji i dekonfiguracji interfejsów sieciowych.

By zaktualizować konfigurację interfejsów:

```
# ifdown eth0
# editor /etc/network/interfaces  # dostosuj według potrzeb
# ifup eth0
```

Więcej informacji uzyskasz w `interfaces(5)`,`/usr/share/doc/ifupdown/examples/network-interfaces.gz` i `ifup(8)`.

10.6.1 Konfiguracja interfejsu ze statycznym adresem IP

Załóżmy, że chcesz skonfigurować interfejs Ethernet tak by miał adres `192.168.0.123`. Ten adres zaczyna się od `192.168.0`, tak więc musi znajdować się w sieci lokalnej (LAN). Załóżmy dodatkowo, że `192.168.0.1` jest adresem bramki sieciowej. Przeedytuj `/etc/network/interfaces` tak by zawierał linie podobne do poniższych:

```
iface eth0 inet static
        address 192.168.0.123
        netmask 255.255.255.0
        gateway 192.168.0.1
```

Jeśli posiadasz zainstalowany `resolvconf` możesz dodać linie określające informacje DNS. Przykłado:

```
iface eth0 inet static
        address 192.168.0.123
        netmask 255.255.255.0
        gateway 192.168.0.1
        dns-search extradomena.org
        dns-nameservers 195.238.2.21 195.238.2.22
```

ISC, który został zastąpiony przez wersję 3 z pakietu `dhcp3-client`. Opiekunowie zamierzają przechrzcić pewnego dnia `dhcp3-client` na `dhcp-client`, ale wygląda na to, że nie stanie się to przed wydaniem Sarge.

[9] Format pliku `/etc/network/interfaces` dla aktualnej wersji `ifupdown` jest niekompatybilny z formatem używanym we wcześniejszych wersjach pakietu w Potato. Skrypt poinstalacyjnych pakietu `ifupdown` powinien zaktualizować plik automatycznie, jeśli pojawi się taka konieczność.

[10] To oznacza również, że nie powinieneś używać innych narzędzi konfiguracyjnych wyższego poziomu, takich jak `whereami`, który korzysta z narzędzi niższego poziomu.

Po podniesieniu interfejsu, argumenty opcji `dns-search` i `dns-nameservers` są dostępna dla `resolvconf`, by mógł je włączyć do `resolv.conf`. Argument `extradomena.org` opcji `dns-search` odpowiada argumentowi opcji `search` w `resolv.conf(5)`. Argumenty `195.238.2.21` i `195.238.2.22` opcji `dns-nameservers` odpowiadają argumentom opcji `nameserver` w `resolv.conf(5)`. Inne rozpoznawane opcje to `dns-domain` i `dns-sortlist`. Zobacz 'Zarządzanie informacjami o serwerach nazw – `resolvconf`' na 195 stronie.

10.6.2 Konfiguracja interfejsu z użyciem DHCP

By skonfigurować interfejs z użyciem DHCP przeedytuj `/etc/network/interfaces` tak by zawierał linijkę podobną do:

```
iface eth0 inet dhcp
```

Aby to zadziałało musisz mieć zainstalowany jeden z klientów DHCP wspomnianych w 'Konfiguracja interfejsów sieciowych z wykorzystaniem DHCP' na 196 stronie.

10.6.3 Konfiguracja interfejsu bezprzewodowego Wi-Fi

Pakiet `wireless-tools` zawiera skrypt `/etc/network/if-pre-up.d/wireless-tools`, który pozwala na konfigurowanie sprzętu Wi-Fi (802.11a/b/g) zanim interfejs zostanie podniesiony. Konfiguracji dokonuje się używając programu `iwconfig`; zobacz `iwconfig(8)`. Dla każdego możliwego parametru polecenia `iwconfig` możesz dołączyć opcję o takiej samej nazwie w `/etc/network/interfaces` poprzedzając ją przedrostkiem „wireless-". Przykładowo by ustawić ESSID dla `eth0` na `myessid` i klucz szyfrujący na `123456789e` przed podniesieniem interfejsu `eth0` używając DHCP, przeedytuj `/etc/network/interfaces` tak by zawierał poniższe linie:

```
iface eth0 inet dhcp
        wireless-essid myessid
        wireless-key 123456789e
```

10.6.4 Konfiguracja interfejsu PPP

Programy `ifup` i `ifdown` używają `pon` i `poff` do dodawania i usuwania interfejsów PPP więc najpierw przeczytaj 'Konfiguracja interfejsu PPP' na 190 stronie.

Załóżmy, że ustawiłeś PPP by działało z połączeniem `mojisp`. Przeedytuj `/etc/network/interfaces` tak by zawierał poniższe linie:

```
iface ppp0 inet ppp
        provider mojisp
```

Z takim ustawieniem `ifup ppp0` uruchomi

```
pon mojisp
```

Niestety nie ma w tej chwili możliwości by przekazać dodatkowe opcje dla `pppd` w liniach dotyczących `ppp` znajdujących się `/etc/network/interfaces`. [11]

Nie jest w tej chwili możliwe użycie `ifupdown` do przeprowadzenia pomocniczej konfiguracji interfejsów PPP. Ponieważ `pon` istnieje zanim `pppd` skończy nawiązywanie połączenia, `ifup` uruchamia skrypty `up` zanim interfejs PPP jest gotowy do użycia. Dopóki ten błąd [12] nie zostanie poprawiony, wymagane jest konfigurowanie przez edycję plików `/etc/ppp/ip-up` lub `/etc/ppp/ip-up.d/`.

10.6.5 Konfiguracja interfejsu PPPoE

Wiele dostawców internetu (ISP) używa PPP do negocjacji połączenia nawet w momencie gdy maszyna klienta podłączona jest poprzez Ethernet lub/i sieć ATM. Realizuje się to używając PPP ponad Ethernetem (PPPoE), który jest techniką pakowaia strumieni PPP w ramkach Ethernetu. Załóżmy, że Twój dostawca nazywa się *`mojisp`*. Najpierw skonfigurujemy PPP i PPPoE do połączenia *`mojisp`*. Najprostszym sposobem osiągnięcia tego jest instalacja pakietu `pppoeconf` i uruchomienie w konsoli `pppoeconf`. Następnie przeedytuj `/etc/network/interfaces` tak by zawierało poniższe linie:

```
iface eth0 inet ppp
        provider mojisp
```

Czasami występują problemy z Maksymalną Jednostką Transmisji (MTU) w połączeniu z PPPoE przez Digital Subscriber Line (DSL). Szczegóły znajdziesz w DSL-HOWTO (`http://www.tldp.org/HOWTO/DSL-HOWTO/`).

Zauważ, że jeśli Twój modem zawiera router, wtedy modem/router obsługuje połączenie PPPoE samodzielnie i pojawia się w sieci jako zwykła bramka Ethernetowa.

10.6.6 Konfiguracja wielu interfejsów Ethernet

Załóżmy, że interfejs `eth0` jest podłączony do Internetu z adresem pobieranym poprzez DHCP, a `eth1` jest podłączony do sieci lokalnej ze statycznym adresem `192.168.1.1`. Zmodyfikuj `/etc/network/interfaces` by zawierało poniższe wpisy:

```
iface eth0 inet dhcp
```

---

[11] Zobacz błąd #196877 (`http://bugs.debian.org/196877`).

[12] Zobacz błąd #127786 (`http://bugs.debian.org/127786`).

```
iface eth1 inet static
 address 192.168.1.1
 netmask 255.255.255.0
```

Jeśli na tym komputerze uaktywnisz NAT, tak jak to opisano w 'Budowanie bramy sieciowej' na 208 stronie, będziesz mógł współdzielić połączenie internetowe ze wszystkimi komputerami z sieci.

10.6.7 Konfigurowanie interfejsów wirtualnych

Używając interfejsów wirtualnych możesz skonfigurować jedną kartę Ethernet by była interfejsem do wielu podsieci. Załóżmy, że Twój komputer jest w sieci 192.168.0.x/24. Teraz chcemy podłączyć go do Internetu używając tej samej karty i adresu publicznego dostarczanego przez DHCP. Zmodyfikuj `/etc/network/interfaces` tak by zawierał poniższe linie:

```
iface eth0 inet static
 address 192.168.0.1
 netmask 255.255.255.0
 network 192.168.0.0
 broadcast 192.168.0.255

iface eth0:0 inet dhcp
```

Interfejs `eth0:0` jest interfejsem wirtualnym. Jego rodzicem jest `eth0`.

10.7 Rekonfiguracja sieci

W poniższych rozważaniach ważne jest by czytelnik rozumiał różnicę pomiędzy **fizycznym**, a **logicznym** interfejsem. [13] Interfejs fizyczny jest tym co do tej pory nazywaliśmy „interfejsem”, a więc rzeczy określane jako `eth0`, `ppp1` i podobnie. Logiczny interfejs to zbiór wartości, które mogą być przydzielone do zmienny parametrów interfejsu fizycznego. Jeśli jest to dla Ciebie niezrozumiałe, zamieniaj podczas czytania wyrażenie „skonfigurowany jako interfejs logiczny `X`” na „skonfigurowany z profilem interfejsu `X`”.

Definicje `iface` w `/etc/network/interfaces` są definicjami logicznych interfejsów, a nie fizycznych. [14] Jeśli nie zamierzasz nigdy rekonfigurować interfejsów, to możesz zignorować ten fakt, bo fizyczny interfejs `X` domyślnie będzie skonfigurowany jako logiczny interfejs `X`.

Załóżmy jednakże, że Twój komputer to laptop, którego przewozisz między pracą i domem. Gdy podłączasz go do sieci korporacyjnej lub do sieci domowej, musisz odpowiedni skonfigurować `eth0`.

---

[13] Taka terminologia jest używana w dokumentacji pakietu `ifupdown`.

[14] Zauważ, że interfejsy wymienione w liniach `auto` są fizycznymi, a nie logicznymi interfejsami.

Najpierw zdefiniuj dwa logiczne interfejsy `dom` i `praca` (zamiast jednego `eth0` jak to robiliśmy wcześniej), które opisują jak interfejsy powinny być skonfigurowane dla odpowiednich sieci.

```
iface dom inet static
 address 192.168.0.123
 netmask 255.255.255.0
 gateway 192.168.0.1

iface praca inet static
 address 81.201.3.123
 netmask 255.255.0.0
 gateway 81.201.1.1
```

Wtedy fizyczny interfejs `eth0` może zostać podniesiony dla sieci domowej z użyciem odpowiedniej konfiguracji, używając w linii poleceń:

```
# ifup eth0=dom
```

By przekonfigurować `eth0` do użycia w pracy, wprowadź polecenia:

```
# ifdown eth0
# ifup eth0=praca
```

Zauważ, że przy tak napisanym pliku `interfaces` nie ma możliwości podniesienia `eth0` wpisując samo `ifup eth0`. Powodem jest fakt, że `ifup` domyślnie używa fizycznej nazwy interfejsu jako logicznej, a w tym momencie nie ma zdefiniowanych żadnego interfejsu logicznego `eth0`.

10.8 Magiczna rekonfiguracja sieci

Nazwy logicznych interfejsów mogą być „zmapowane" na inne nazwy podczas uruchamiania `ifup`. To jak nazwy będą mapowane może być uzależnione od różnych warunków. Zatem `ifup` może być tak skonfigurowany, że będzie podnosił dany interfejs fizyczny jako odpowiedni interfejs logiczny wybrany z predefiniowanych alternatyw.

Mapowanie nazw logicznych interfejsów odbywa się następująco:

- Jeśli w linii poleceń `ifup` nie podano nazwy logicznego interfejsu, nazwa fizycznego zostanie użyta jako wstępna nazwa.
- Jeśli nazwa logicznego interfejsu pasuje do wzorca użytego w części `mapping` to jest ona użyta do tworzenia nowej nazwy logicznego interfejsu. Odbywa się to rekursywnie aż do braku dopasowań.

- Jeśli ostatnia nazwa logicznego interfejsu jest etykietą defuuchu ubterfejsu w `/etc/network/interfaces`, to fizyczny interfejs jest podnoszony jako logiczny. W przeciwnym wypadku `ifup` wyświetla komunikat „Ignoring unknown interface” i kończy swoje działanie.

Składnia części `mapping` wygląda następująco:

```
mapping wzorzec
 script nazwa-skryptu
 [map dane skryptu]
```

Skrypt określony w części `mapping` jest uruchamiany z nazwą fizycznego interfejsu jako jego argument oraz z zawartością wszystkich następujących linii „map” podanych na wejście standardowe skryptu. Skrypt wypisuje rezultat mapowania na swoim standardowym wyjściu tuż przed zakończeniem.

Przykładowo poniższy wpis `mapping` sprawi, że `ifup` podniesie interfejs `eth0` jako interfejs logiczny `dom`.

```
mapping eth0
 script /usr/local/sbin/echo-home
```

gdzie `/usr/local/sbin/echo-home` jest:

```
#!/bin/sh
echo home
```

Ponieważ mapowanie odbywa się w skrypcie, istnieje możliwość wyboru logicznego interfejsu na podstawie różnych testów. Zacznijmy od przykładu. Załóżmy, że masz inne karty sieciowe w domu i w pracy. Katalog `/usr/share/doc/ifupdown/examples/` zawiera skrypt mapujący, który może być wykorzystany do wyboru logicznego interfejsu na podstawie adresu MAC. Najpierw zainstaluj skrypt w odpowiednim katalogu.

```
# install -m770 /usr/share/doc/ifupdown/examples/get-mac-address.sh \
/usr/local/sbin/
```

Następnie dodaj poniższy wpis do `/etc/network/interfaces`:

```
mapping eth0
 script /usr/local/sbin/get-mac-address.sh
 map 02:23:45:3C:45:3C dom
 map 00:A3:03:63:26:93 praca
```

Bardziej skomplikowane programy mapujące są opisane poniżej w ‘guessnet’ na następnej stronie, ‘ifupdown-roam’ na sąsiedniej stronie i ‘laptop-net’ na następnej stronie.

10.8.1 guessnet

Zainstaluj `guessnet` i dodaj poniższy wpis do `/etc/network/interfaces`:

```
mapping eth0
 script guessnet-ifupdown
 map dom
 map praca
```

Teraz przy wpisaniu `ifup eth0 guessnet` sprawdzi czy `eth0` może być podniesiony jako `dom` czy jako `praca`. By to osiągnąć `guessnet` używa informacji przechowywanej w definicjach logicznych interfejsów. Korzysta z żądań ARP by wykryć bramę sieciową albo z komputera opisanego w linii `test-peer`.

`guessnet` jest w ciągłym rozwoju.

10.8.2 ifupdown-roam

Pakiet `ifupdown-roam` zawiera program `ifscout`, którego możliwości mapowania są rozszerzeniem guessnet. `ifscout` wywołuje `guessnet` by odszukać komputery w bezprzewodowej sieci Ethernet, ale potrafi również wykryć serwery DHCP i bezprzewodowe punkty dostępu. `ifscout` jest skryptem powłoki więc może być dostosowany do Twoich potrzeb.

Aktualnie (październik 2003) `ifupdown-roam` nie jest częścią Debiana. Możliwe, że w chwili gdy to czytasz jest już w archiwach Debiana; w przeciwnym razie możesz go pobrać z jego gdzie znajdziesz również informacje na temat instalacji i użytkowania.

10.8.3 laptop-net

Pakiet `laptop-net` stosuje inne podejście do automatycznej rekonfiguracji sieci. Nie korzysta z logicznych interfejsów `ifupdown`, a zamiast tego ma swój system „schematów" i „profili". Mimo to laptop-net nadal używa `ifupdown` do konfiguracji fizycznych interfejsów. Więcej informacji znajdziesz w przystępnie napisanej dokumentacji z pakietu `laptop-net-doc`.

10.9 Wywoływanie rekonfiguracji sieci

Widzieliśmy już jak można przekonfigurować interfejsy. Rekonfiguracja musi się odbywać w odpowiednim momencie.

Tradycyjnie sieć jest konfigurowana podczas sekwencji uruchomieniowej przez skrypt inicjalizacyjny `/etc/rcS.d/S40networking` i jest rzadko zmieniana. Usługi wykorzystujące sieć są uruchamiane później. Podczas wyłączania systemu skrypty inicjalizujące są uruchamiane w odwrotnej kolejności.

Jednakże aktualnie można zauważyć trend w którym GNU i Linux zmierza do obsługi sprzętu i wyjątków, które zmieniają się dynamicznie. Na początku dodano obsługę dla wymienialnych kart PCMCIA; później został zaimplementowany mechanizm `hotplug`, który umożliwia wymianę większej ilości sprzętu podczas pracy komputera. Zawiera się w tym również sprzęt sieciowy. Zauważ, że usługi, które zależą od wymienialnego podczas pracy sprzętu muszą być uruchamiane po włożeniu owego sprzętu, a zatrzymywane po jego wyjęciu. To oznacza, że te usługi muszą być usunięte spod kontroli systemu inicjalizującego System V i obsługiwane przez `ifupdown`.

Załóżmy przykładowo, że usługa `foo` obsługiwana przez skrypt inicjalizacyjny `/etc/init.d/foo` zależy od dynamicznie rekonfigurowanego interfejsu sieciowego `eth0`.

- Najpier usuń `foo` spod kontroli systemu inicjalizującego:

  ```
  # update-rc.d -f foo remove
  ```

- Następnie umieść `foo` pod kontrolą `ifupdown` dodając w `/etc/network/interfaces` przy interfejsie `eth0` opcje `up` i `down`, które będą wywoływać skrypt inicjalizujący `foo`:

  ```
  iface eth0 inet dhcp
   up /etc/init.d/foo start
   down /etc/init.d/foo stop
  ```

10.9.1 Wywoływanie konfiguracji sieci podczas uruchamiania

Podczas uruchamiania systemu skrypt inicjalizujący `/etc/rcS.d/S40networking` uruchamia polecenie `ifup -a`. Podnosi to wszystkie fizyczne interfejsy wymienione w `/etc/network/iterfaces` z opcją `auto`.

Na dzień dzisiejszy w wielu przypadkach lepiej jest zarządzać konfiguracją sieci w sposób dynamiczny. W momencie gdy mechanizmy obsługujące dynamicznie zmieniający się sprzęt są uruchomione, można dużo łatwiej traktować sprzęt statyczny jakby był również dynamiczny. Uruchamianie można wtedy postrzegać jako kolejne zdarzenie hotplug (zmiany sprzętu). (Zobacz 'Wywoływanie konfiguracji sieci – `hotplug`' na tej samej stronie.)

Jednakże w większości przypadków będziesz chciał podnosić podczas uruchamiania przynajmniej interfejs zwrotny `lo`. W związku z tym upewnij się, że `/etc/network/interfaces` zawiera następujący wpis:

```
auto lo
face lo inet loopback
```

Możesz dopisać inne fizyczne interfejsy jeśli chcesz by również były podnoszone podczas uruchamiania systemu. **Nigdy** nie wpisuj interfejsów PCMCIA w części `auto`. `cardmgr` PCMCIA jest uruchamiany po uruchomieniu `/etc/rcS.d/S40networking`.

10.9.2 Wywoływanie konfiguracji sieci – `hotplug`

By otrzymać obsługę hotplug (urządzenia, które mogą być dokładane/usuwane podczas pracy komputera), zainstaluj pakiet `hotplug`.

Sprzęt sieciowy może być dołożony albo podczas uruchamiania albo już po nim gdy karta (np. karta PCMCIA) jest włożona do komputera albo narzędzie `discover` zostaje uruchomione i ładuje odpowiednie moduły sterowników. [15]

Kiedy jądro wykrywa nowe urządzenie, inicjalizuje sterownik, a następnie uruchamia program `hotplug` by je skonfigurować. Później gdy sprzęt jest usuwany, jądro uruchamia `hotplug` ponownie, ale z innymi zmiennymi środowiskowymi. W systemie Debian, podczas gdy wywoływany jest `hotplug`, uruchamia on skrypty z `/etc/hotplug/` i `/etc/hotplug.d/`. Szczegóły znajdziesz w `hotplug(8)`.

Nowo włożony sprzęt sieciowy jest konfigurowany przez skrypt `/etc/hotplug/net.agent`. [16] Załóżmy, że Twoja karta sieciowa PCMCIA została włożona, a co za tym idzie interfejs `eth0` jest dostępny do wykorzystania. `/etc/hotplug/net.agent` wykonuje następujące polecenie:

```
ifup eth0=hotplug
```

Jeśli nie dodasz definicji interfejsu logicznego lub mapowania o nazwie `hotplug` w `/etc/network/interfaces` to to polecenie nic nie zdziała. By sprawić, aby to polecenie skonfigurowało `eth0`, dodaj następujący wpis do `/etc/network/interfaces`:

```
mapping hotplug
 script echo
```

Tak jak wyjaśniono to w '**Re**konfiguracja sieci' na 200 stronie taki wpis zmapuje pokazane polecenie, więc odpowiada następującemu:

```
ifup eth0=eth0
```

(**Nie** używaj tego typu wpisów mapowania jeśli do kontroli interfejsu używasz `ifplugd` uruchamionego przez `hotplug`, tak jak to opisano w 'Wywoływanie konfiguracji sieci – `ifplugd`' na następnej stronie.)

Jeśli chcesz jedynie by `eth0` zostało podniesione, użyj `grep` zamiast `echo` jak w przykładzie poniżej:

```
mapping hotplug
 script grep
 map eth0
```

Więcej wskazówek znajdziesz w 'Magiczna rekonfiguracja sieci' na 201 stronie i `/usr/share/doc/hotplug/README.Debian`.

---

[15] Jeśli Twój system uruchamia `discover` w pliku `/etc/rcS.d/S36discover`, musisz przenieść go do `/etc/rcS.d/S40discover`. Zobacz błąd #208155 (`http://bugs.debian.org/208155`).

[16] Może być również konfigurowany przez jakikolwiek inny skrypt umieszczony w `/etc/hotplug.d/net/`.

10.9.3 Wywoływanie konfiguracji sieci – `ifplugd`

`ifplugd` podnosi interfejs w zależności od tego czy sprzęt jest podłączony do sieci czy nie. Może on wykryć podłączenie kabla Ethernet lub punkt dostępowy związany z interfejsem Wi-Fi. Gdy `ifplugd` zauważy, że stan połączenia się zmienił, uruchamia pośredni skrypt, który domyślnie uruchamia `ifup` lub `ifdown` dla danego interfejsu.

`ifplugd` dobrze współpracuje z `hotplug`. Po włożeniu karty, które powoduje możliwością skorzystania z interfejsu, `/etc/hotplug.d/net/ifplugd.hotplug` uruchamia instancję `ifplugd` dla tego interfejsu. Gdy `ifplugd` wykryje, że karta jest podłączona do sieci podnosi interfejs korzystając z `ifup`.

10.9.4 Wywoływanie konfiguracji sieci – `waproamd`

By powiązać punkt dostępowy z kartą Wi-Fi możliwe, że zaistnieje konieczność zaprogramowania jej z odpowiednim kluczem szyfrującym WEP. Jeśli do kontrolowania `ifup` korzystasz z `ifplugd` tak jak to opisano w 'Wywoływanie konfiguracji sieci – `ifplugd`' na tej samej stronie to oczywiste jest, że nie możesz ustawić klucza szyfrującego, bo `ifup` odpalany jest już po fakcie. Jednym z możliwych rozwiązań jest zaprogramowanie wszystkich wymaganych kluczy w nienaruszalnej pamięci karty Wi-Fi. Jeśli jednak przemieszczasz się pomiędzy wieloma sieciami, to Twoja karta Wi-Fi prawdopodobnie nie będzie w stanie przechowywać tak wiele kluczy.

Innym rozwiązaniem jest skorzystanie z `waproamd`, który ustawia klucz szyfrujący WEP w zależności od tego jaki punkt dostępowy zostanie wyszukany.

`waproamd` dobrze współpracuje z `hotplug`. Gdy karta zostanie włożona, przez co uzyskamy możliwość wykorzystania interfejsu, `/etc/hotplug.d/net/waproamd.hotplug` uruchamia instancję `waproamd` dla tego interfejsu. `waproamd` dobrze współpracuje również z `ifplugd`.

Sprawdź plik README z pakietu `waproamd` by dowiedzieć się więcej.

10.9.5 Konfiguracja sieci i PCMCIA

W poprzednich wydaniach systemu Debian standardowym sposobem konfiguracji kart sieciowych PCMCIA było wykorzystanie możliwości pakietu `pcmcia-cs`. Skrypty z `/etc/pcmcia/` są wykonywane przez `cardmgr`, daemon, który zarządza 16-bitowymi kartami PCMCIA, po ich włożeniu lub wyjęciu. Niektórzy używają `pcmcia-cs` w jego domyślnej konfiguracji, w której `/etc/pcmcia/network` wywołuje `ifup` po dodaniu interfejsu i `ifdown` po jego usunięciu. Inni używają specjalnego systemu wywołań niskopoziomowych poleceń konfiguracyjnych, które są zaimplementowane w `/etc/pcmcia/network` i uaktywnione gdy w `/etc/pcmcia/network.opts` odpowiednie opcje ustawione są na `y`. Ten drugi sposób pociąga za sobą wiele problemów. Działa tylko z 16-bitowymi kartami PCMCIA oraz duplikuje funkcje `ifupdown`. Z kolei mechanizm `pcmcia-cs` konfiguracji sieci jest uważany za przestarzały na rzecz `hotplug` opisanego w 'Wywoływanie konfiguracji sieci – `hotplug`' na 204 stronie.

By wyłączyć wszystkie mechanizmy konfiguracji interfejsów sieciowych w `pcmcia-cs`, dodaj następującą linię:

```
exit 0
```

na początku pliku `/etc/pcmcia/network`.

Zauważ, że nie ma nic złego w używaniu `cardmgr`. Po prostu nie chcemy by wywoływał programy konfiguracji sieci. Zobacz 'PCMCIA' na 105 stronie i Linux PCMCIA HOWTO (`http://www.tldp.org/HOWTO/PCMCIA-HOWTO.html`).

10.10 Konfiguracja usług sieciowych

Typowa konfiguracja usług sieciowych na komputerze/serwerze domowym zawiera:

- Internet *super-server* i daemon TCP/IP wrapper, zobacz 'Ograniczenia dostępu do usług' na 144 stronie.
  - `/etc/inetd.conf`
- `ssh`: OpenSSH bezpieczna powłoka, zobacz 'SSH' na 165 stronie.
  - `/etc/ssh/ssh_config`
  - `/etc/ssh/sshd_config`
- `exim`: agent transportu poczty, zobacz 'Nazwa pocztowa (mailname)' na 194 stronie.
  - `/etc/exim/exim.conf`
  - `/etc/mailname`
  - `/etc/aliases`
  - `/etc/email-addresses`
- `fetchmail`: daemon do pobierania wiadomości e-mail z konta POP3, zobacz 'Pobieranie poczty – Fetchmail' na 172 stronie.
  - `/etc/fetchmailrc`
- `procmail`: lokalna dystrybucja wiadomości e-mail oraz program filtrujący, zobacz 'Przetwarzanie poczty – Procmail' na 172 stronie.
  - `~/.procmailrc`
- Nazwa hosta i DNS (serwer pośredniczący, bufor, ...), zobacz 'Nazwa komputera' na 193 stronie i 'Usługa nazw (DNS)' na 194 stronie.
  - `/etc/host.conf`
  - `/etc/hostname`
  - `/etc/hosts`
  - `/etc/hosts.allow`
  - `/etc/hosts.deny`
  - `/etc/resolv.conf`
  - `/etc/bind/named.conf` (przeedytuj)
  - `/etc/bind/db.lan` (dodaj dla komputerów z sieci lokalnej)
  - `/etc/bind/db.`*`192.168.0`* (dodaj dla komputerów z sieci lokalnej, zwrotny DNS)
- DHCP, zobacz 'Konfiguracja interfejsów sieciowych z wykorzystaniem DHCP' na 196 stronie.

  - `/etc/dhcp3/dhclient.conf` (DHCP klient)
  - `/etc/default/dhcp3-server` (DHCP serwer)
  - `/etc/dhcp3/dhcpd.conf` (DHCP serwer)
- `cvs`: zobacz 'Concurrent Versions System (CVS)' na 219 stronie.
  - `/etc/cvs-cron.conf`
  - `/etc/cvs-pserver.conf`
- `nfs-kernel-server`: sieciowy system plików, zobacz 'Konfiguracja NFS' na 41 stronie. (dla systemów uniksowych)
  - `/etc/exports`
- `samba`: współdzielenie plików i drukarek w sieciach Windows, zobacz 'Konfiguracja Samby' na 42 stronie i 'Samba' na 136 stronie.
  - `/etc/samba/smb.conf`
- Daemon systemu drukowania, zobacz 'Konfiguracja drukarki' na 43 stronie.
  - `/etc/printcap` (dla lpr)
- `apache` i `apache2`: serwer WWW.
  - `/etc/apache/*`
  - `/etc/apache2/*`
- `squid`: serwer pośredniczący WWW.
  - `/etc/squid/*`

10.11 Rozwiązywanie problemów związanych z siecią

Jeśli doświadczysz problemów z siecią, wyniki następujących poleceń to pierwsze co musisz sprawdzić:

```
# ifconfig
# cat /proc/pci
# cat /proc/interrupts
# dmesg | more
```

Zobacz również podrozdziały 'Podstawy testowania sieci' na 132 stronie.

Jeśli masz problemy z konkretnymi stronami, zobacz 'Dziwne kłopoty z dostępem do witryn WWW.' na 46 stronie.

10.12 Budowanie bramy sieciowej

Komputer z systemem Debian może pełnić funkcję pełnej bramy sieciowej, która wykonuje zamianę adresów sieciowych (NAT, znany również jako maskarada), obsługuje pocztę, DHCP, buforowanie DNS, pośredniczenie HTTP, usługę CVS, udostępnianie NFS lub Samby. Zobacz w 'Nazwy hostów i adresy IP używane w sieci LAN' na 33 stronie przykłady takich ustawień.

10.12.1 Konfiguracja netfiltra

Projekt netfilter/iptables jest podsystemem ściany ogniowej dla Linuksa 2.4 i nowszych. Udaj się na stronę Netfilter (`http://www.netfilter.org/`), gdzie opisano i wyjaśniono wiele zagadnień dotyczących konfiguracji sieciowej.

Netfilter - podstawy

Netfilter przetwarza pakiety przy użyciu pięciu wbudowanych łańcuchów: PREROUTING, INPUT, FORWARD, OUTPUT i POSTROUTING.

```
                decyzja
                routingu
IN ------> PRE ---> ------> FORWARD -----> ----> POST -----> OUT
interfejs  ROUTING  \       śledzenie    /       ROUTING     interfejs
           DNAT     |       połączeń     ^       SNAT
           REDIRECT |                    |      MASQUERADE
                    v                    |
                  INPUT                OUTPUT
                    | filtrowanie        ^ filtrowanie,DNAT
                    v                    |
                    \--> Lokalny Proces--/
                            program przestrzeni użytkownika
```

Tablice Netfilter

Pakiety są przetwarzane w każdym łańcuchu w poszczególnych tablicach.

- filter (filtrowanie pakietów, domyślna)
  - INPUT (tylko dla pakietów wchodzących do hosta)
  - FORWARD (tylko dla pakietów przekazywanych przez hosta)
  - OUTPUT (tylko dla lokalnie generowanych pakietów).
- nat (network address translation - translacja adresów sieciowych)
  - PREROUTING (do zmiany zawartości pakietów podczas ich wejścia)
  - OUTPUT (do zmiany zawartości pakietów generowanych lokalnie, przed routingiem)
  - POSTROUTING (do zmiany zawartości pakietów, gdy wychodzą one z hosta)
- mangle (network address mangling, sprawne tylko w jądrach nowszych niż 2.4.18)
  - wszystkie 5 wbudowanych łańcuchów.

Cele Netfilter

Reguły ściany ogniowej mają wiele celów (miejsc przeznaczenia):

- 4 podstawowe cele:

  - ACCEPT przepuszcza pakiet (zezwala na dostęp).
  - DROP po cichu odrzuca pakiet (nie informuje o tym nikogo).
  - QUEUE przekazuje pakiet programowi działającemu w przestrzeni użytkownika (jeśli jest to obsługiwane przez jądro).
  - RETURN zatrzymuje przetwarzanie danego łańcucha i wznawia przetwarzanie w następnej regułce poprzedniego (z którego nastąpiło wywołanie) łańcucha.
- rozszerzone cele:
  - LOG włącza logowanie jądra.
  - REJECT zwraca informację o błędzie nadawcy i odrzuca pakiet.
  - SNAT zmienia adres źródłowy pakietu i jest używany tylko w łańcuchu POSTROUTING (tablica nat).
    `--to-source  ipaddr[-ipaddr][:port-port]`
  - MASQUERADE działa tak samo jak SNAT, ale jest przeznaczona dla połączeń z dynamicznie przypisywanym numerem IP (połączenie modemowe) (tablica nat).
    `--to-ports port[-port]`
  - DNAT zmienia adres docelowy pakietu, jest używany w łańcuchach PREROUTING i OUTPUT oraz w zdefiniowanych przez użytkownika łańcuchach, które są wywoływane z dwóch powyższych łańcuchów (tablica nat).
    `--to-destination ipaddr[-ipaddr][:port-port]`
  - REDIRECT zmienia adres docelowy tak, by pakiet otrzymał komputer, na którym działa ogniomurek (czyli ten komputer).
    `--to-ports port[-port]`

Polecenia Netfiltera

Podstawowe polecenia `iptables`:

```
iptables -N łańcuch                     # tworzy łańcuch

iptables -A łańcuch \                   # dodaje regułkę do łańcuch
         -t tablica \                   # w tablica (filter, nat, mangle)
         -p protokół \                  # tcp, udp, icmp lub all
         -s adres-źródłowy[/maska] \
         --sport port[:port] \          # port źródłowy, jeśli -p jest tcp lub udp
         -d adres-docelowy[/maska] \
         --dport port[:port] \          # port docelowy, jeśli -p jest tcp lub udp
         -j cel \                       # co zrobić, jeśli regułka pasuje
         -i interfejs-wejściowy \       # dla INPUT,  FORWARD, PREROUTING
         -o interfejs-wyjściowy         # dla FORWARD, OUTPUT, POSTROUTING
```

Zamiana adresów sieciowych (NAT)

Komputery w sieci lokalnej mogą mieć dostęp do Internetu przez bramkę sieciową, która zamienia adresy sieci na adresy używane w Internecie.

```
# apt-get install ipmasq
```

Zastosuj przykładowe reguły, by zwiększyć zabezpieczenia `ipmasq`. Przeczytaj `/usr/share/doc/ipmasq/examples/stronger/README`. W przypadku debianowego kernel-image-2.4 upewnij się, że załadowałeś odpowiednie moduły. Opis potrzebnej konfiguracji jest w 'Obsługa sieci' na 106 stronie.

W przypadku debianowego kernel-image-2.2, modyfikuj plik `Z92timeouts.rul` w katalogu `/etc/masq/rules` tak, jak pokazano poniżej, by zapewnić dłuższe połączenia ze zdalnymi serwerami (dobre dla dużych listów elektronicznych, itp.):

```
# tcp, tcp-fin, udp
# 2hr, 10 sec, 160 sec - domyślnie
# 1 day, 10 min, 10 min - przykład dłuższego czasu życia
$IPCHAINS -M -S 86400 600 600
```

Jeśli sieć jest osiągalna przez kartę PCMCIA, `ipmasq` musi być wystartowane z pliku `/etc/pcmcia/network.opts`. Przeczytaj `/usr/share/doc/ipmasq/ipmasq.txt.gz`.

Przekierowanie połączeń SMTP (2.4)

Załóżmy, że masz notebooka, który jest tak skonfigurowany, że używa różnych środowisk sieciowych i chcesz używać swojego agenta (serwer) poczty bez potrzeby ciągłej zmiany jego konfiguracji.

Dodając poniższą regułkę `iptables` na bramce sieciowej przekierujesz połączenia SMTP do komputera pełniącego funkcję bramki.

```
# iptables -t nat -A PREROUTING -s 192.168.1.0/24 -j REDIRECT \
          -p tcp --dport smtp --to-port 25 # smtp=25, INPUT is open
```

Jeśli oczekujesz dokładniejszych zestawów regułek, rozważ zainstalowanie pakietu `ipmasq` i dodanie M30redirect.def (`http://www.debian.org/doc/manuals/debian-reference/examples/`) do katalogu `/etc/ipmasq/rules/`.

10.12.2 Zarządzanie wieloma połączeniami z siecią

[FIXME] Routing rozszerzony (policy routing) (autor: Phil Brutsche `<pbrutsch@tux.creighton.edu>`): Zobacz iproute manual (`http://lartc.org/`), jeśli interesują Cię szczegóły. Kształtowanie ruchu (tc, ang. Traffic control) może być równie interesujące.

Środowisko:

```
eth0: 192.168.1.2/24; brama 192.168.1.1
eth1: 10.0.0.2/24; brama 10.0.0.1
Brak maskarady na tym komputerze.
```

Magiczne zaklęcia:

```
ip rule add from 192.168.1.2 lookup 1
ip rule add from 10.0.0.2 lookup 2
ip route add to default via 10.0.0.1 metric 0
ip route add to default via 192.168.1.1 metric 1
ip route add table 1 to 192.168.1.0/24 via eth0
ip route add table 1 to 10.0.0.2/24 via eth1
ip route add table 1 to default via 192.168.1.1
ip route add table 2 to 192.168.1.0/24 via eth0
ip route add table 2 to 10.0.0.2/24 via eth1
ip route add table 2 to default via 10.0.0.2
```

[FIXME] Nigdy tego nie robiłem. Jak ustawić połączenie wdzwaniane jako połączenie zapasowe szybkiego łącza, używając właściwości automatycznego łączenia na żądanie? Wyślij łatkę tutaj :)

Rozdział 11

Edytory

11.1 Popularne edytory

Linux oferuje wiele alternatyw dla konsolowych edytorów tekstu. Wśród nich:

- `vim`: Potężny i lekki edytor, wywodzący się z BSD. VI iMproved.
- `emacs`: Podstawowy i ciężki edytor, wywodzący się z projektu GNU. Autentyk RMS-a (Richarda M. Stallmana).
- `xemacs`: Emacs: Następne pokolenie, pochodzący od Lucid.
- `mcedit`: GNU edytor dla początkujących. Identyczny z wewnętrznym edytorem z `mc`. Zobacz 'Edytor w MC' na 54 stronie.
- `ae`: Domyślny mały edytor (Potato). Należy go unikać.
- `nano`: Domyślny mały edytor (Woody). Naśladuje `pico`.
- `joe`: Dla fanów czasów WordStara lub TurboPascala.
- `jed`: Szybki, w pełni wyposażony, obsługiwany przez menu, ze skrótami klawiszowymi Emacsa.
- `jove`: Bardzo mały edytor ze skrótami klawiszowymi Emacsa.
- `nvi`: Nowy vi. Błąd w błąd kompatybilny z oryginalnym vi.

Aby zmienić domyślny edytor, użyj `update-alternatives --config editor`. Wiele programów używa również zmiennych środowiskowych takich jak `EDITOR` lub `VISUAL` aby zdecydować, którego edytora użyć. Zobacz 'Edytor w MC' na 54 stronie.

Także kilka edytorów środowiska X Window jest godnych uwagi:

- `gvim`: Vim z interfejsem graficznym (Pakiety `vim` i `vim-gtk`).
- `emacs`: Jedyny prawdziwy Emacs (automatycznie wykrywa X Window).
- `xemacs`: Emacs następnej generacji (automatycznie wykrywa X Window).

Wywołania tych klientów X zawierają standardowe opcje, takie jak `-fn` *`a24`*, które ułatwiają życie starszym użytkownikom, takim jak ja :) Zobacz 'Klienty X' na 154 stronie.

11.2 Edytory awaryjne

Istnieje kilka edytorów umieszczonych w katalogu `/bin`. Jeden z nich powinien być zainstalowany do prostego edytowania plików, gdy `/usr` jest nieosiągalne.

- `elvis-tiny`: Minimalny edytor vi (`vi` aby uruchomić)
- `nano-tiny`: Minimalny edytor różny od vi (`nano-tiny` aby uruchomić)
- `ed`: Minimalny edytor (zawsze obecny, ale trudny w użyciu)

11.3 Emacs i Vim

11.3.1 Wskazówki do Vima

Przeczytaj dokument „VIM - main help file” wciskając <F1> podczas pracy programu.

<F1>	Pomoc
<esc>	Powrót do normalnego trybu
V	Tryb wizualny
i	Tryb wprowadzania
:	Polecenia linii poleceń
:set tw=72	Ustawia szerokość tekstu na 72
<F11>	Tryb wprowadzania (wklejania)
:r! date -R	Wstaw datę zgodną z RFC-822
q*a*	Zapisz sekwencję klawiszy w rejestrze *a*
q	Zatrzymaj zapisywanie sekwencji klawiszy
@*a*	Wykonaj sekwencję klawiszy z rejestru *a*
:edit *foo.txt*	Edytuj inny plik przez wczytanie *foo.txt*
:wnext	Zapisz bieżący plik i edytuj następny plik

`q` i `@` mogą być używane do zapisywania i odtwarzania prostych makr. Na przykład, aby utworzyć makro, które wstawia znaczniki pochyłości HTML wokół słowa pod kursorem, możesz wpisać `qii<i>^[ea</i>^[q` (gdzie `^[` jest klawiszem ESC). Gdy później ustawisz kursor na początku słowa i wpiszesz `@i`, zostaną dodane znaczniki <i> i </i>.

Zobacz również ‘Używanie GnuPG z Vimem’ na 249 stronie.

11.3.2 Wskazówki do Emacsa

<F1>	Pomoc
<F10>	Menu
C-u M-! date -R	Wstaw datę zgodną z RFC-822

11.3.3 Uruchamianie edytora

```
uruchom edytor:                            emacs plik  vim plik
uruchom w trybie zgodności z vi:                       vim -C
uruchom w trybie niezupełnie zgodnym z vi:             vim -N
uruchom z opcjami domyślnymi:              emacs -q    vim -N -u NONE
```

11.3.4 Podsumowanie poleceń edytorów (Emacs, Vim)

```
wyjście:                       C-x C-c         :qa /:wq /:xa /:q!
Powrót/tryb poleceń:           C-g             <esc>
Do tyłu (w lewo):              C-b             h
Do przodu (w prawo):           C-f             l
W dół:                         C-n             j
W górę:                        C-p             k
początek linii(^):             C-a             0
koniec linii($):               C-e             $
Polecenia wielokrotne:         C-u nnn polec   nnn polecenie
Polecenia wielokrotne:         M-cyfra polecenie
zapisz plik:                   C-x C-s         :w plik
początek bufora:               M-<             1G
koniec bufora:                 M->             G
przewiń 1 ekran do przodu:     C-v             ^F
przewiń 1/2 ekranu do przodu:                  ^D
przewiń 1 linię do przodu:                     ^E
przewiń 1 ekran do tyłu:       M-v             ^B
przewiń 1/2 ekranu do tyłu:                    ^U
przewiń 1 linię do tyłu:                       ^Y
przewiń inne okno:              M-C-v
skasuj w miejscu kursora:       C-d             x
skasuj od kurs. do końca linii: C-k             D
przeszukiwanie w przód:         C-s
przeszukiwanie wstecz:          C-r
wyszukaj w przód:               C-s enter       /
wyszukaj wstecz:                C-r enter       ?
szukanie wyrażeń regularnych:   M-C-s
szukanie wyrażeń reg. wstecz:   M-C-r
znajdź wyrażenie regularne:     M-C-s enter     /
znajdź wyrażenie reg. wstecz:   M-C-r enter     ?
Pomoc:                          C-h C-h         :help
Pomoc Apropos:                  C-h a
objaśnienia klawiszy:           C-h b           :help [klawisz]
Pomoc (Info):                   C-h i
Pomoc Główny tryb:              C-h m
Pomoc samouczek:                C-h t           :help howto
```

```
Cofnij (Undo):                  C-_             u
Anuluj cofnięcie (Redo):        C-f             ^R
Zaznacz pozycję kursora:        C-@             m{a-zA-Z}
Zmień zaznaczenie i pozycję:    C-x C-x
Idź do znacznika w bieżącym pliku:              '{a-z}
Idź do znacznika w dowolnym pliku:              '{A-Z}
kopiuj obszar:                  M-w             {w wizualnym}y
usuń obszar:                    C-w             {w wizualnym}d
Wklej i zachowaj bufor:         C-y
Wklej z bufora usuniętych:      M-y             p
zmień obszar na duże litery:    C-x C-u         {w wizualnym}U
zmień obszar na małe litery:    C-x C-l         {w wizualnym}u
Wstaw znak specjalny:           C-q numer ósemkowo/sekwencja klawiszy
                                                ^V dziesiętnie/sekwencja klawiszy
zastąp:                         M-x replace-string      :%s/aaa/bbb/g
zastąp wyrażenie regularne:     M-x replace-regexp      :%s/aaa/bbb/g
zastąp z potwierdzeniem:        M-%                     :%s/aaa/bbb/gc
zastąp z potwierdzeniem:        M-x query-replace
zastąp wyrażenie reg. z potwierdzeniem: M-x query-replace-regexp
Otwórz plik:                    C-x C-f         :r plik
Zapisz plik:                    C-x C-s         :w
Zapisz wszystkie bufory:        C-x s           :wa
Zapisz jako:                    C-x C-w plik    :w plik
zmień bufor:                    C-x b
wyświetl bufory:                C-x C-b         :buffers
Przełącz na tylko-do-odczytu:   C-x C-q         :set ro
zapytaj i usuń bufor:           C-x k
Podziel pionowo:                C-x 2           :split
Podziel poziomo:                C-x 3           :vsplit (wersja 6)
Przenieś do innego okna:        C-x o           ^Wp
Usuń to okno:                   C-x 0           :q
Skasuj inne okno (okna):        C-x 1           ^Wo
uruchom kompilację w tle:       M-x compile
usuń kompilację w tle:          M-x kill-compilation
uruchom make                                    :make Makefile
sprawdź informacje o błędach:   C-x`            :echo errmsg
uruchom powłokę logując wyjście: M-x shell      :!script -a tmp
...wyczyść znaki backspace, ...                 :!col -b <tmp >record
...zapisz/przywołaj zapis powłoki: C-x C-w record  :r record
uruchom powłokę:                M-! sh          :sh
uruchom polecenie:              M-! cmd         :!cmd
uruchom polecenie i wstaw:      C-u M-! cmd     :r!cmd
uruchom filtr:                  M-| plik        {wizualny}:w file
uruchom filtr i wstaw:          C-u M-| filter  {wizualny}:!filter
pokaż opcję:                                    :se[t] {opcja}?
przywróć wartość domyślną opcji                 :se[t] {opcja}&
```

<table>
<tr><td>ustaw opcję logiczną (boolean)</td><td>:se[t] no{opcja}</td></tr>
<tr><td>przełącz opcję logiczną (boolean)</td><td>:se[t] inv{option}</td></tr>
<tr><td>zawiń tekst na 72. kolumnie</td><td>:se tw=72</td></tr>
<tr><td>nie zawijaj</td><td>:se tw=0</td></tr>
<tr><td>automatycznie wcinaj</td><td>:se ai</td></tr>
<tr><td>rozwijaj tabulator</td><td>:se et</td></tr>
<tr><td>ustaw znaki komentarza (mail)</td><td>:se comments=n:>,n:\|</td></tr>
</table>

uruchom GDB	M-x gdb
opisz tryb GDB	C-h m
rusz się o jedną linię	M-s
następna linia	M-n
rusz się o jedną instrukcję	M-i
zakończ bieżącą ramkę stosu	C-c C-f
kontynuuj	M-c
podnieś ramki argumentów	M-u
upuść ramki argumentów	M-d
skopiuj ilość od punktu, wstaw na końcu:	C-x &
ustaw pułapkę	C-x SPC

11.3.5 Konfiguracja Vima

W celu użycia wszystkich funkcji Vima oraz podświetlania składni, należy dodać następujące linie do `~/.vimrc` lub `/etc/vimrc`:

```
set nocompatible
set nopaste
set pastetoggle=<f11>
syn on
```

Tryb wklejania umożliwia unikanie wpływu automatycznego wcinania przy operacjach wytnij-i-wklej w terminalu. Robi on więcej niż proste „:set noai”.

Jeżeli interesuje Cię integracja z GnuPG, zobacz ‘Używanie GnuPG z Vimem’ na 249 stronie.

11.3.6 Ctags

`apt-get install exuberant-ctags` i uruchom `ctags` na plikach źródłowych. Wpisz `:tag` *`nazwa_funkcji`* w Vimie, aby przeskoczyć do linii w której *nazwa_funkcji* się zaczyna. Znaczniki działają dla C, C++, Javy, Pythona i wielu innych języków.

Emacs posiada te same możliwości ctags.

11.3.7 Konwersja ekranu z podświetlaniem składni do kodu HTML

`so \$VIMRUNTIME/syntax/2html.vim` wywołane z linii poleceń Vima zamieni podświetlany tekst na kod HTML. Zapisz używając `:w plik.html` i `:q`. Użytecznie dla kodu w języku C itd.

11.3.8 Rozdzielanie ekranu w `vimie`

`vim` może edytować wiele plików w środowisku z wielokrotnie podzielonym ekranem. W celu uzyskania szczegółowych informacji, wpisz `:help usr_08.txt`.

Aby rozdzielić ekran między różne pliki, wpisz w lini poleceń vi:

```
:split inny-plik
:vsplit inny-plik
```

Lub w linii poleceń powłoki:

```
$ vi -o file1.txt file2.txt   # Podział poziomy
$ vi -O file1.txt file2.txt   # Podział pionowy
```

wywoła to vi podzielonego na wiele okien.

```
$ vimdiff file.txt~ file.txt         # sprawdza ostatnie zmiany w file.txt
$ vimdiff file.en.sgml file.fr.sgml  # sprawdza zamiany w przekładzie/tłumaczeniu
$ gvimdiff file.txt~ file.txt        # w środowisku X Window
```

spowoduje wyświetlenie ładnego widoku zmian między oryginałem i kopią zapasową. W przypadku SGML-a dopasowuje znaczniki, więc porównywanie tłumaczeń w tym trybie działa bardzo dobrze.

Specjalne ruchy kursora wywoływane przez Ctrl-W:

```
Ctrl-W +      zwiększa rozmiar okna
Ctrl+W -      zmniejsza rozmiar okna
Ctrl-W h      przejdź do okna po lewej
Ctrl-W j      przejdź do okna poniżej
Ctrl-W k      przejdź do okna powyżej
Ctrl-W l      przejdź do okna po prawej
...
```

Aby kontrolować przewijanie ekranu użyj następujących opcji:

```
:set scrollbind
:set noscrollbind
```

Rozdział 12

Systemy kontroli wersji

12.1 Concurrent Versions System (CVS)

Przeczytaj (np. przeglądarką `lynx`) pliki: `/usr/share/doc/cvs/html-cvsclient`, `/usr/share/doc/cvs/html-info`, `/usr/share/doc/cvsbook` lub wykonaj polecenia: `info cvs` oraz `man cvs` w celu uzyskania dokładnych informacji.

12.1.1 Instalacja serwera CVS

Poniższe ustawienia pozwalają na wprowadzanie zmian w repozytorium wyłącznie użytkownikom należącym do grupy „src”, a na administrację CVS wyłącznie członkowi grupy „staff”, redukując w ten sposób szanse zastrzelenia samego siebie.

```
# cd /var/lib; umask 002 ; mkdir cvs # [Woody] FSH
# apt-get install cvs cvs-doc cvsbook
# export CVSROOT=/var/lib/cvs
# cd $CVSROOT
# chown root:src . # "`staff"' dla większych ograniczeń rozpoczynania projektów
# chmod 3775 . # Jeśli w powyższym występuje "`staff"', użyj 2775
# cvs -d /var/lib/cvs init  # bezpieczniej jest określić -d tutaj!
# cd CVSROOT
# chown -R root:staff .
# chmod 2775 .
# touch val-tags
# chmod 664 history val-tags
# chown root:src history val-tags
```

12.1.2 Przykładowe sesje CVS

Poniżej znajdują się przykłady konfiguracji środowiska powłoki dla dostępu do repozytoriów CVS.

Anonimowy CVS (tylko pobieranie)

Zdalny dostęp tylko do odczytu:

```
$ export CVSROOT=:pserver:anonymous@cvs.sf.net:/cvsroot/qref
$ cvs login
$ cvs -z3 co qref
```

Użycie lokalnego serwera CVS

Dostęp lokalny z powłoki na tej samej maszynie:

```
$ export CVSROOT=/var/lib/cvs
```

Użycie odległego pservera CVS

Zdalny dostęp bez SSH (używając obsługi RSH w `cvs`):

```
$ export CVSROOT=:pserver:account@cvs.foobar.com:/var/lib/cvs
$ cvs login
```

Jest to metoda podatna na podsłuch.

Użycie zdalnego CVS przez `ssh`

Zdalny dostęp z SSH:

```
$ export CVSROOT=:ext:account@cvs.foobar.com:/var/lib/cvs
```

lub dla SourceForge:

```
$ export CVSROOT=:ext:account@cvs.sf.net:/cvsroot/qref
```

Możesz również użyć uwierzytelnienia RSA ('Połączenia z mniejszą ilością haseł – RSA' na 167 stronie), które eliminuje pytanie o hasło.

Tworzenie nowego archiwum CVS

Dla,

OBIEKT	WARTOŚĆ	ZNACZENIE
Drzewo źródeł:	*~/project-x*	Wszystkie kody źródłowe
Nazwa projektu:	*project-x*	Nazwa projektu
Źródło modułu:	*Main-branch*	Nazwa całego modułu (vendortag)
Etykieta wydania:	*Release-original*	Nazwa wydania (releasetag)

Wykonaj,

```
$ cd ~/project-x                  # przejdź do katalogu ze źródłami
 ... stwórz drzewo źródeł ...
$ cvs import -m "Start project-x" project-x Main-branch Release-original
$ cd ..; rm -R ~/project-x
```

Praca z CVS

Praca nad projektem *project-x* przy użyciu lokalnego repozytorium:

```
$ cd                             # przejście do przestrzeni roboczej
$ cvs co project-x               # pobranie źródeł z CVS-u
$ cd project-x
 ... wprowadzanie zmian w zawartości ...
$ cvs diff -u                    # podobne do diff -u repozytorium/ lokalne/
$ cvs up -C modyfikowany_plik    # cofnięcie zmian w pliku
$ cvs ci -m "Opisz zmiany"       # wysłanie modyfikacji do CVS
$ vi nowy_plik
$ cvs add nowy_plik
$ cvs ci -m "Dodano nowy_plik"
$ cvs up                         # pobranie najnowszych wersji z CVS
 ... by utworzyć wszystkie nowoutworzone katalogi w CVS użyj
 ... "cvs up -d -P" w zastępstwie.
 ... uważaj na linie zaczynające się od "‘C nazwa_pliku"’
 ... kod niezmodyfikowany przeniesiony jest do ‘.#nazwa_pliku.wersja’.
 ... szukaj "<<<<<<<" i ">>>>>>>" w nazwa_pliku.
$ cvs tag Release-1              # dodaj symboliczną nazwę wydania
 ... edytuj dalej ...
$ cvs tag -d Release-1           # usuń nazwę wydania
$ cvs ci -m "następne komentarze"
$ cvs tag Release-1              # ponownie dodaj nazwę wydania
$ cd                             # wróć do przestrzeni roboczej
$ cvs co -r Release-initial -d old project-x
```

```
 ... pobierz wersję oryginalną do katalogu old
$ cd old
$ cvs tag -b Release-initial-bugfixes # stwórz nazwę odgałęzienia (-b)
 ... możesz teraz pracować na starej wersji (Tag=sticky)
$ cvs update -d -P         # nie twórz pustych katalogów
 ... drzewo źródeł ma teraz trwałą (sticky) nazwę "‘Release-initial-bugfixes"’
 ... pracuj na tym odgałęzieniu
$ cvs up -d -P # zsynchronizuj z plikami modyfikowanymi przez innych w tej gałęzi
$ cvs ci -m "check into this branch"
$ cvs update -kk -A -d -P
 ... usuń trwałą nazwę i zapomnij zawartość
 ... aktualizuj z głównego pnia bez rozwijania słów kluczowych
$ cvs update -kk -d -P -j Release-initial-bugfixes
 ... włączenie z odgałęzienia Release-initial-bugfixes do drzewa
 ... głównego.  Napraw konflikty edytorem.
$ cvs ci -m "włączono Release-initial-bugfixes"
$ cd
$ tar -cvzf old-project-x.tar.gz old     # stwórz archiwum, -j dla bz2
$ cvs release -d old               # usuń lokalne źródła (opcjonalne)
```

Miłe opcje do zapamiętania (używaj jako pierwszych opcji dla `cvs`):

```
-n      pusty przebieg, żadnych efektów
-t      wyświetl komunikaty pokazujące etapy pracy cvs
```

Eksport plików z CVS

Aby pobrać najnowszą wersję z CVS użyj „tomorrow”:

```
$ cvs ex -D tomorrow nazwa_modułu
```

Zarządzanie CVS

Dodaj alias dla projektu (serwer lokalny):

```
$ su - admin           # członek grupy staff
$ export CVSROOT=/var/lib/cvs
$ cvs co CVSROOT/modules
$ cd CVSROOT
$ echo "px -a project-x" >>modules
$ cvs ci -m "px jest teraz aliasem project-x"
$ cvs release -d .
$ exit                 # Ctrl-D aby wyjść z su
```

```
$ cvs co -d project px
 ... pobierz project-x (alias:px) z CVS to katalogu project
$ cd project
 ... wprowadź zmiany w zawartości ...
```

0

12.1.3 Rozwiązywanie problemów z CVS

Uprawnienia plików w repozytorium

CVS nie nadpisuje aktualnych plików w repozytorium, tylko zastępuje je innymi. Z tego powodu, *pozwolenie na pisanie do katalogu z repozytorium* ma kluczowe znaczenie. Dla każdego nowego repozytorium, wykonaj następujące czynności aby zapewnić ten warunek.

```
# cd /var/lib/cvs
# chown -R root:src repozytorium
# chmod -R ug+rwX   repozytorium
# chmod    2775     repozytorium  # jeśli potrzebne, w podkatalogu
```

Bit wykonywalności

Bit wykonywalności pliku jest zachowany przy pobieraniu z CVS. Kiedy tylko zobaczysz problem z prawem do wykonywania w pobranych plikach, zmień uprawnienia pliku w repozytorium następującym poleceniem.

```
# chmod ugo-x nazwa_pliku
```

12.1.4 Polecenia CVS

Oto polecenia CVS wraz ze skrótami.

```
{add|ad|new} [-k kflag] [-m 'komunikat'] pliki...
{admin|adm|rcs} [opcje-rcs] pliki...
{annotate|ann} [opcje] [pliki...]
{checkout|co|get} [opcje] moduły...
{commit|ci|com}  [-lnR]  [-m  'podsumowanie_zmian'  |  -f  file] \
        [-r rewizja] [pliki...]
{diff|di|dif} [-kl] [opcje_rcsdiff] [[-r rev1 | -D data1] \
        [-r rev2 |  -D data2]] [pliki...]
{export|ex|exp} [-flNn] -r rev|-D data [-d katalog] [-k kflag] moduł...
{history|hi|his} [-report] [-flags] [-options args] [pliki...]
```

```
{import|im|imp} [-opcje] repozytorium vendortag releasetag...
{login|logon|lgn}
{log|lo|rlog} [-l] opcje-rlog [pliki...]
{rdiff|patch|pa} [-flags] [-V vn] [-r t|-D d [-r t2|-D d2]] moduły...
{release|re|rel} [-d] katalogi...
{remove|rm|delete} [-lR] [pliki...]
{rtag|rt|rfreeze} [-falnR]  [-b]  [-d]  [-r  tag  |  -D  data] \
         nazwa_symboliczna moduły...
{status|st|stat} [-lR] [-v] [pliki...]
{tag|ta|freeze} [-lR] [-F] [-b] [-d] [-r tag | -D data]  [-f] \
         nazwa_symboliczna [pliki...]
{update|up|upd} [-AdflPpR] [-d] [-r tag|-D data] pliki...
```

12.2 Subversion

Subversion jest systemem kontroli wersji nowej generacji, który tworzony był z myślą o zastąpieniu CVS. Według twórców jest on obecnie w stanie „alpha”, jednak prawdopodobnie jest wystarczająco stabilny dla większości użytkowników. W momencie powstawania tego tekstu, Subversion dostępny jest wyłącznie w `niestabilnej` gałęzi Debiana.

12.2.1 Instalacja serwera Subversion

Meta-pakiet `subversion-server` jest zależny od wymaganych do użycia systemu Subversion pakietów (`libapache2-dav-svn` oraz `subversion-tools`).

Tworzenie repozytorium

Aktualnie, pakiet `subversion` nie tworzy repozytorium, więc musi być ono utworzone ręcznie. Jednym z możliwych miejsc dla repozytorium jest `/var/local/repos`.

Utwórz katalog:

```
# mkdir -p /var/local/repos
```

Utwórz bazę danych repozytorium:

```
# svnadmin create /var/local/repos
```

Daj prawo do zapisu w repozytorium serwerowi WWW:

```
# chown -R www-data:www-data /var/local/repos
```

Konfiguracja Apache2

Aby umożliwić dostęp do repozytorium przez uwierzytelnienie użytkownika, dodaj (lub usuń komentarz) następujące wpisy do `/etc/apache2/mods-available/dav_svn.conf`:

```
<Location /repos>
  DAV svn
  SVNPath /var/local/repos
  AuthType Basic
  AuthName "Repozytorium Subversion"
  AuthUserFile /etc/subversion/passwd
  <LimitExcept GET PROPFIND OPTIONS REPORT>
    Require valid-user
  </LimitExcept>
</Location>
```

Następnie stwórz plik z hasłami użytkowników używając polecenia:

```
htpasswd2 -c /etc/subversion/passwd jakiś-użytkownik
```

Przeładuj Apache2, a Twoje nowe repozytorium Subversion będzie dostępne pod adresem http://*nazwa_hosta*/repos.

12.2.2 Przenoszenie repozytorium CVS do Subversion

12.2.3 Przykłady użycia Subversion

Poniższe części pokazują jak używać różnych poleceń w Subversion.

Tworzenie nowego archiwum

Aby stworzyć nowe archiwum Subversion, napisz poniższe polecenia:

```
$ cd ~/twój-projekt         # przejdź do katalogu źródeł
$ svn import http://localhost/repos twój-projekt \
  nazwa-projektu -m "początkowy import projektu"
```

Tworzy to katalog o nazwie *nazwa-projektu* w Twoim repozytorium. Katalog ten zawiera pliki Twojego projektu. Sprawdź pod adresem http://localhost/repos/ czy się tam znalazł.

Praca z Subversion

Praca z *project-y* przy użyciu Subversion:

```
$ cd                            # przejdź do przestrzeni roboczej
$ svn co http://localhost/repos/project-y  # pobierz źródła
$ cd project-y
 ... wykonaj pracę ...
$ svn diff                      # podobne do diff -u repozytorium/ lokalne/
$ svn revert modyfikowany_plik  # cofnij zmiany w pliku
$ svn ci -m "Opisz zmiany"      # wyślij pliki do repozytorium
$ vi dodany_plik
$ svn add dodany_plik
$ svn add nowy_katalog          # rekursywnie dodaj pliki z nowy_katalog
$ svn add -N new_dir2           # nierekursywnie dodaj nowy katalog
$ svn ci -m "Dodano nowy_plik, nowy_katalog, nowy_katalog2"
$ svn up                        # pobranie najnowszej wersji z repozytorium
$ svn log                       # pokazuje wszystkie wprowadzone zmiany
$ svn copy http://localhost/repos/project-y \
      http://localhost/repos/project-y-branch \
      -m "Tworzę własną gałąź project-y"  # rozgałęzienie project-y
$ svn copy http://localhost/repos/project-y \
      http://localhost/repos/proj-y_release1.0 \
      -m "project-y 1.0 release"    # dodanie tagu release
 ... zauważ, że odgałęzianie i tagowanie są tym samym. Jedyną różnicą jest to,
 ... że gałęzie można uaktualniać, a tagów nie.

 ... wprowadzaj zmiany w gałęzi ...

$ # połącz odgałęzienie z gałęzią główną
$ svn merge http://localhost/repos/project-y \
   http://localhost/repos/project-y-branch
$ svn co -r 4 http://localhost/repos/project-y # pobierz rewizję 4
```

Rozdział 13

Programowanie

Pamiętaj, by nie używać nazwy „test” do oznaczania pliku wynikowego, ponieważ `test` jest wbudowanym poleceniem powłoki.

13.1 Gdzie zacząć?

Odnośniki:

- Dokumentacja i przykłady w katalogu `/usr/share/doc/`*`pakiet`*
- Unix / Programming Information (`http://arioch.unomaha.edu/~jclark/#info`)
- *Linux Programming Bible* (John Goerzen/IDG books)

Wiele długich dokumentów info można otrzymać w postaci wydrukowanej z GNU (`http://www.gnu.org/`).

W następnych czterech rozdziałach znajdziesz przykładowe skrypty, każdy napisany w innym języku, które tworzą plik tekstowy zawierający informacje o koncie, które z kolei mogą być wstawione do pliku `/etc/passwd` za pomocą przetwarzania wsadowego np. programem `newusers`. Każdy ze skryptów oczekuje na wejściu pliku, którego linie są następującego formatu `imię nazwisko hasło`. (Katalogi domowe użytkowników nie będą tworzone przez te skrypty.)

13.2 Powłoka

Najlepszym sposobem na zrozumienie działania systemów uniksowych jest przeglądanie i czytanie skryptów powłoki. Poniżej podam pewne wskazówki i materiały przypominające programowanie w powłoce. Zobacz Pomyłki powłoki (`http://www.greenend.org.uk/rjk/2001/04/shell.html`) by uczyć się na błędach.

13.2.1 Bash – interaktywna powłoka zgodna z GNU

Odnośniki do informacji na temat Bash-a:

- `bash(1)`
- `info bash`
- LDP BASH Programming - Introduction HOWTO (`http://www.tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html`) (Programowanie w BASH-u - Wprowadzenie JTZ) - informacje dla początkujących
- `mc /usr/share/doc/bash/examples/ /usr/share/doc/bash/` (Zainstaluj najpierw pakiet `bash-doc`, aby zobaczyć przykłady.)
- *Learning the bash Shell*, 2nd edition (O'Reilly) - (Nauka powłoki bash - drugie wydanie)

Oto krótki, przykładowy program (tworzy on wpisy dt. kont na podstawie danych ze standardowego wejścia dla programu `newusers`):

```
#!/bin/bash
# (C) Osamu Aoki Sun Aug 26 16:53:55 UTC 2001 Public Domain
pid=1000;
while read n1 n2 n3 ; do
if [ ${n1:0:1} != "#" ]; then
let pid=$pid+1
echo ${n1}_${n2}:password:${pid}:${pid}:"`,/home/${n1}_${n2}:/bin/bash
fi
done
```

13.2.2 Powłoki zgodne z POSIX

W systemie Debian powłoki zgodne z POSIX znajdują się w kilku pakietach:

- `dash` (Sarge)
  - Priority: optional
  - Installed-Size: 176
  - Najmniejszy i jak dotąd najszybszy. – najlepszy dla początkowego rozruchu.
- `ash` (Woody)
  - Priority: optional
  - Installed-Size: 180
  - Mniejszy i o wiele szybszy – dobry dla początkowego rozruchu.
- `bash`
  - Essential: yes
  - Priority: required
  - Installed-Size: 580
  - Większy i z większymi możliwościami – napisanych zostało już wiele rozszerzeń.
- `pdksh`
  - Priority: optional
  - Installed-Size: 408
  - Kompletny klon powłoki AT&T ksh.

Jeśli piszesz skrypty powłoki, które mają być przenoszone na inne powłoki/systemy to najlepiej pisać je jako skrypty powłoki POSIX. Użyj `/bin/sh` dowiązanego do `ash` lub (`dash`) aby sprawdzić zgodność z POSIX. Unikaj pisania skryptów z naleciałościami **bashowymi** lub **zshsowymi**, które przypominają składnię `csh`. Na przykład, staraj się unikać:

- `if [ foo == bar ] ; then ...`
- `diff -u file.c{.orig,}`
- `mkdir /foo{bar,baz}`

Opisy powłoki w tym dokumencie dotyczą tylko powłok zgodnych ze strandardem POSIX, a więc nie dotyczą powłok typu `csh`, włączając `tcsh`.

13.2.3 Parametry powłoki

Kilka **parametrów specjalnych** do zapamiętania:

```
$0      = nazwa powłoki lub skryptu
$1      = pierwszy(1) argument powłoki
 ...
$9      = dziewiąty(9) argument powłoki
$#      = ilość parametrów pozycyjnych
"$*"    = "$1 $2 $3 $4 ... $n"
"$@"    = "$1" "$2" "$3" "$4" ... "$n"
$?      = kod zakończenia ostatnio wykonywanego polecenia
$$      = PID aktualnie wykonywanego skryptu powłoki
$!      = PID ostatnio rozpoczętego zadania w tle
```

Podstawowe **rozwinięcia parametrów** (parameter expansion) do zapamiętania:

Forma	Jeśli *parametr* jest ustawiony	Jeśli *parametr* nie jest ustawiony
`${`*parametr*`:-`*słowo*`}`	`$`*parametr*	*słowo*
`${`*parametr*`:+`*słowo*`}`	*słowo*	`null`
`${`*parametr*`:=`*słowo*`}`	`$`*parametr*	*słowo* (i wykonuje *parametr*=*słowo*)
`${`*parametr*`:?`*słowo*`}`	`$`*parametr*	(wyświetla *słowo* i kończy działanie)

Dwukropek (':') we wszystkich powyższych operatorach nie jest tak naprawdę potrzebny.

- Z ':' = operator sprawdza czy zmienna „istnieje” i „nie jest pusta”.
- Bez ':' = operator sprawdza tylko czy zmienna „istnieje”.

Podstawowe **podstawienia parametrów**(parameter substitutions) do zapamiętania:

Forma	Wynik
`${`*parametr*`%`*końcówka*`}`	Usunięcie najmniejszego wzorca *końcówka*
`${`*parametr*`%%`*końcówka*`}`	Usunięcie największego wzorca *końcówka*
`${`*parametr*`#`*przedrostek*`}`	Usunięcie najmniejszego wzorca *przedrostek*
`${`*parametr*`##`*przedrostek*`}`	Usunięcie największego wzorca *przedrostek*

13.2.4 Przekierowywanie powłoki

Podstawowe operatory **przekierowywania** do zapamiętania ([n] jest opcjonalną liczbą określającą kwalifikator pliku):

```
[n]> plik     Przekieruj standardowe wyjście (lub n) do pliku.
[n]>> plik    Dołącz standardowe wyjście (lub n) do pliku.
[n]< plik     Przekieruj standardowe wejście (lub n) z pliku.
[n1]>&n2      Przekieruj standardowe wyjście (lub n1) do n2.
> plik >&2    Przekieruj standardowe wyjście i strumień błędów do pliku.
| polecenie     Przekieruj standardowe wyjście na wejście
polecenia.
>&2 | polecenie Przekieruj standardowe wyjście i strumień błędów
na wejście polecenia.
```

Gdzie

- standardowe wejście (stdin): kwalifikator pliku = 0)
- standardowe wyjście (stdout): (kwalifikator pliku = 1)
- standardowe wyjście błędu (stderr): (kwalifikator pliku = 2)

Powłoka umożliwia otwarcie plików przez użycie wbudowanego polecenia `exec` i bezwzględnego kwalifikatora pliku.

```
$ echo Witaj >pliczek
$ exec 3<pliczek 4>pliczek2  # otwieranie plików
$ cat <&3 >&4        # przekierowanie stdin do 3, a stdout do 4
$ exec 3<&- 4>&-     # zamykanie plików
$ cat pliczek2
Witaj
```

Powyżej *n*`<&-` oraz *n*`>&-` oznacza zamknięcie kwalifikatora pliku o numerze *n*.

13.2.5 Polecenia warunkowe powłoki

Każde polecenie zwraca **kod wyjścia** (exit status), który może zostać użyty w wyrażeniu warunkowym:

- Sukces: 0 (Prawda)
- Błąd: 1–255 (Fałsz)

Zauważ, że użycie wartości 0 w celu wyrażenia „prawdy” różni się od zwykłej konwencji w innych obszarach przetwarzania. Poza tym '[' jest odpowiednikiem polecenia `test`, które traktuje argumenty aż do ']' jako wyrażenie warunkowe.

Podstawowe **idiomy warunkowe** (conditional idioms) do zapamiętania to:

```
polecenie && jeśli_sukces_uruchom_również_to_polecenie || true
polecenie || jeśli_nie_sukces_uruchom_w_zamian_to_polecenie
```

```
if [ wyrażenie_warunkowe ]; then
 jeśli_sukces_uruchom_to_polecenie
else
 jeśli_nie_sukces_uruchom_to_polecenie
fi
```

`||` `true` było potrzebne by upewnić się, że ten skrypt przypadkowo nie zakończy się w tej linii jeśli powłoka byłaby wywołana z flagą `-e`.

Operatory porównywania **plików** w wyrażeniach warunkowych to:

```
-e plik         Prawda, jeśli plik istnieje.
-d plik         Prawda, jeśli plik istnieje i jest katalogiem.
-f plik         Prawda, jeśli plik istnieje i jest normalnym plikiem.
-w plik         Prawda, jeśli plik istnieje i jest zapisywalny.
-x plik         Prawda, jeśli plik istnieje i jest wykonywalny.
plik1 -nt plik2 Prawda, jeśli plik1 jest nowszy niż plik2. (modyfikacja)
plik1 -ot plik2 Prawda, jeśli plik1 jest starszy niż plik2. (modyfikacja)
plik1 -ef plik2 Prawda, jeśli mają takie same numery urządzeń i iwęzłów.
```

Operatory porównywania **łańcuchów znaków** w wyrażeniach warunkowych:

```
     -z słowo    Prawda, jeśli długość słowo wynosi zero.
     -n słowo    Prawda, jeśli długość słowo jest różna od zera.
słowo1 == słowo2   Prawda, jeśli słowa są równe.
słowo1 = słowo2    Prawda, jeśli słowa są równe.
    (dla ścisłej zgodności z POSIX, zamiast "`=="' należy używać "`="')
słowo1 != słowo2   Prawda, jeśli słowa nie są równe.
słowo1 <  słowo2   Prawda, jeśli słowo1 sortuje się przed słowo2 (zleżne od lokali
słowo1 >  słowo2   Prawda, jeśli słowo1
sortuje się po słowo2 (zależne od lokalizacji (locale) ).
```

Operatory **arytmetycznego** porównywania liczb całkowitych

Wyrażenia warunkowe to: `-eq`, `-ne`, `-lt`, `-le`, `-gt`, i `-ge`.

13.2.6 Przetwarzanie linii poleceń

Powłoka przetwarza skrypt tak:

- rozbicie na **żetony** (tokens) przez meta-znaki: SPACJA, TAB, NOWA_LINIA, ;, (,), <, >, `|`, &
- sprawdzenie **słów kluczowych** (keyword), jeśli nie są pomiędzy „. . . ” lub ’. . . ’ (pętla)
- rozwinięcie **aliasów**, jeśli nie są pomiędzy „. . . ” lub ’. . . ’ (pętla)
- rozwinięcie **nawiasów**, `a{1,2}` -> `a1 a2`, jeśli nie są pomiędzy „. . . ” lub ’. . . ’
- rozwinięcie **tyld**, *~użytkownik* -> katalog domowy *użytkownika*, jeśli nie jest pomiędzy „. . . ” lub ’. . . ’

- rozwinięcie **parametrów**, $*PARAMETRÓW*, jeśli nie są pomiędzy „...”
- rozwinięcie **podstawienia polecenia**, $(*polecenie*), jeśli nie jest pomiędzy ’...’
- podział na **słowa** wg. $IFS, jeśli nie są pomiędzy „...” lub ’...’
- rozszerzenie *?[] na **ścieżki**, jeśli nie jest pomiędzy „...” lub ’...’
- znalezienie **polecenia**
  - funkcji
  - wbudowanego
  - pliku w $PATH
- pętla

Pojedyncze cudzysłowy pomiędzy podwójnymi cudzysłowami nie dają żadnego efektu.

Wykonanie `set -x` w powłoce lub też wywołanie powłoki z opcją `-x` zmusza powłokę do wypisywania wszystkich poleceń, jakie wykonuje. Może to być pomocne przy odpluskwianiu.

13.3 Awk

Miejsca, w których można znaleźć informacje o Awk:

- *Effective awk Programming*, 3rd edition (O’Reilly) - (Efektywne programowanie w awk, trzecie wydanie)
- *Sed & awk*, 2nd edition (O’Reilly) - (Sed i awk, drugie wydanie)
- `mawk(1)` i `gawk(1)`
- `info gawk`

Krótki przykład programu (tworzącego wpisy dla `newusers`):

```
#!/usr/bin/awk -f
# Skrypt tworzący plik przystosowany do użycia z poleceniem ’newusers’
# z pliku zawierającego ID użytkowników i hasła w formie:
# Imię Nazwisko hasło
# Copyright (c) KMSelf Sat Aug 25 20:47:38 PDT 2001
# Distributed under GNU GPL v 2, or at your option, any later version.
# This program is distributed WITHOUT ANY WARRANTY.

BEGIN {
    # Przydzielenie początkowych UID i GID
    if ( ARGC > 2 ) {
        startuid = ARGV[1]
        delete ARGV[1]
    }
    else {
        printf( "Wywołanie: newusers PoczatkoweUID plik\n" \
          "  gdzie:\n"
          "    PoczątkoweUID jest początkowym identyfikatorem dodawanego użytkowni
          "    plik jest plikiem wejścia w postaci:\n" \
          "      imię nazwisko hasło\n" \
```

```
        )
        exit
    }

    infile = ARGV[1]
    printf( "Początkowe UID: %s\n\n", startuid )
}

/^#/ { next }

{
    ++record
    first = $1
    last = $2
    passwd = $3
    user= substr( tolower( first ), 1, 1 ) tolower( last )
    uid = startuid + record - 1
    gid = uid
    printf( "%s:%s:%d:%d:%s %s"`/home/%s:/bin/bash\n",  \
        user, passwd, uid, gid, first, last, user \
        )
}
```

Dwa pakiety w systemie Debian dostarczają `awk` zgodnego z POSIX:
- `mawk`
    - Priority: required
    - Installed-Size: 228
    - Mniejszy i znacznie szybszy – dobry dla domyślnej instalacji
    - Zawiera ograniczenia ustalane podczas kompilacji
        - NF = 32767
        - sprintf buffer = 1020
- `gawk`
    - Priority: optional
    - Installed-Size: 1708
    - Większy i potężniejszy – posiada wiele rozszerzeń
        - System V Release 4 version of UNIX
        - Bell Labs awk
        - GNU-specific

13.4 Perl

To jest **prawdziwy** interpreter w systemach uniksowych.

Miejsca, w których można znaleźć informacje o Perl'u:
- `perl(1)`

- *Programming Perl*, 3rd edition (O'Reilly)

Krótki, przykładowy program (tworzy wpisy dla polecenia `newusers`):

```
#!/usr/bin/perl
# (C) Osamu Aoki Sun Aug 26 16:53:55 UTC 2001 Public Domain
$pid=1000;
while (<STDIN>) {
        if (/^#/) { next;}
        chop;
        $pid++;
        ($n1, $n2, $n3) = split / /;
        print $n1,"_",$n2,":", $n3, ":",$pid,
                  ":",$pid,""`,/home/",$n1,"_",$n2,":/bin/bash\n"
}
```

Instalacja modułu Perla *nazwa_modułu*:

```
# perl -MCPAN -e 'install nazwa_modułu'
```

13.5 Python

Jest to miły, zorientowany obiektowo interpreter.

Miejsca, w których można znaleźć informacje o Pythonie:

- `python(1)`
- *Learning Python* (O'Reilly).

Krótki, przykładowy program (tworzy wpisy dla polecenia `newusers`):

```
#! /usr/bin/env python
import sys, string

# (C) Osamu Aoki Sun Aug 26 16:53:55 UTC 2001 Public Domain
# Ported from awk script by KMSelf Sat Aug 25 20:47:38 PDT 2001
# This program is distributed WITHOUT ANY WARRANTY.

def usages():
    print \
"Sposób użycia:  ", sys.argv[0], " PoczątkoweUID [plik]\n" \
"\tPoczątkoweUID jest początkowym identyfikatorem dodawanego użytkownika.\n" \
"\tplik jest plikiem wejścia. Jeśli nie określony to standardowe wejście.\n\n" \
"Format pliku wejściowego:\n"\
"\timię nazwisko hasło\n"
                return 1
```

```
def parsefile(startuid):
    #
    # główne filtrowanie
    #
    uid = startuid
    while 1:
        line = infile.readline()
        if not line:
            break
        if line[0] == ’#’:
            continue
        (first, last, passwd) = string.split(string.lower(line))
        # powyższe rozsypuje się przy złej liczbie parametrów :-)
        user = first[0] + last
        gid = uid
        lineout = "%s:%s:%d:%d:%s %s"‘/home/%s:/bin/bash\n" %  \
            (user, passwd, uid, gid, first, last, user)
        sys.stdout.write(lineout)
        +uid

if __name__ == ’__main__’:
    if len(sys.argv) == 1:
        usages()
    else:
        uid = int(sys.argv[1])
        #print "# UID zaczyna się od: %d\n" % uid
        if len(sys.argv) > 1:
            infilename   = string.join(sys.argv[2:])
            infile = open(infilename, ’r’)
            #print "# Czytaj plik z: %s\n\n" % infilename
        else:
            infile = sys.stdin
        parsefile(uid)
```

13.6 Make

Miejsca, w których można znaleźć informacje o Make:

- `info make`
- `make(1)`
- *Managing Projects with make*, 2nd edition (O’Reilly)

Proste zmienne automatyczne:

Składnia reguł:

```
Cel: [ warunek wstępny ... ]
 [TAB]  polecenie1
 [TAB]  -polecenie2 # ignoruj błędy
 [TAB]  @polecenie3 # wstrzymaj echo
```

Gdzie `[TAB]` jest kodem TAB. Każda linia jest intrepretowana przez powłokę po podmianie zmiennych przez `make`. Użyj \ na końcu linii, by kontynuować skrypt. Użyj `$$`, by wprowadzić `$` dla zmiennych środowiskowych w skrypcie powłoki.

Uwikłane reguły dla *celu* oraz *warunki wstępne* mogą być, na przykład, zapisane jako:

```
%: %.c header.h
```

lub

```
%.o: %.c header.h
```

Tutaj *cel* zawiera znak `%` (dokładnie jeden z nich). Znak `%` może zastępować jakikolwiek niepusty podciąg aktualnych nazw plików danego celu. W *warunkach wstępnych* również można użyć znaku `%` by pokazać jak ich nazwy mają się do aktualnej nazwy celu.

Reguły przyrostkowe są **przestarzałym** sposobem definiowania reguł uwikłanych dla `make`. Są nadal obsługiwane w GNU `make` dla zgodności, ale używaj odpowiednich reguł wzorców, kiedykolwiek to możliwe:

```
stara reguła przyrostkowa --> nowa reguła wzorcowa
.c:                       --> %  : %.c
.c.o:                     --> %.o: %.c
```

Automatyczne zmienne dla reguł:

```
foo.o: new1.c new2.c old1.c new3.c
$@ == foo.o                        (cel)
$< == new1.c                       (pierwszy)
$? == new1.c new2.c new3.c         (nowsze)
$^ == new1.c new2.c old1.c new3.c  (wszystkie)
$* == `%' dopasowany rdzeń we wzorcu celu
```

Opis zmiennych:

```
foo1 := bar    # Jednorazowe rozszerzenie
foo2  = bar    # Rekursywne rozszerzenie
foo3 += bar    # Dopisanie
SRCS := $(wildcard *.c)
```

```
OBJS := $(foo:c=o)
OBJS := $(foo:%.c=%.o)
OBJS := $(patsubst %.c,%.o,$(foo))
DIRS  = $(dir katalog/plik.rozszerzenie) # Wydobywa "‘katalog"’
$(notdir NAZWY...), $(basename NAZWY...), $(suffix NAZWY...) ...
```

Wprowadź `make -p -f/dev/null`, by zobaczyć automatyczne reguły wewnętrzne.

13.7 C

Przygotowanie:

```
# apt-get install glibc-doc manpages-dev libc6-dev gcc
```

Miejsca, w których można znaleźć informacje o C:
- `info libc` (opis funkcji biblioteki C)
- `gcc(1)`
- `each_C_library_function_name(3)`
- Kernighan & Ritchie, *The C Programming Language*, 2nd edition (Prentice Hall).

13.7.1 Prosty program w C (gcc)

Prosty przykład kompilowania `przyklad.c` z biblioteką `libm` do pliku wykonywalnego `uruchom_przyklad`:

```
$ cat > przyklad.c << EOF
#include <stdio.h>
#include <math.h>
#include <string.h>

int main(int argc, char **argv, char **envp){
        double x;
        char y[11];
        x=sqrt(argc+7.5);
        strncpy(y, argv[0], 10); /* zapobiega przepełnieniu bufora */
        y[10] = ’\0’; /* wypełniamy, by upewnić się, że ciąg kończy się ’\0’ */
        printf("%5i, %5.3f, %10s, %10s\n", argc, x, y, argv[1]);
        return 0;
}
EOF
$ gcc -Wall -g -o uruchom_przyklad przyklad.c -lm
$ ./uruchom_przyklad
```

```
        1, 2.915, ./uruchom_,     (null)
$ ./uruchom_przyklad 1234567890qwerty
        2, 3.082, ./uruchom_, 1234567890qwerty
```

W przykładzie `-lm` jest potrzebne, by zlinkować bibliotekę `libm` dla `sqrt()`. Faktycznie biblioteka znajduje się w `/lib` pod nazwą `libm.so.6`, która jest dowiązaniem symbolicznym do `libm-2.1.3.so`.

Spójrz na ostatni parametr w wyjściowym tekście. Jest tam więcej niż dziesięć znaków mimo, że użyliśmy `%10s`.

Używanie funkcji ze wskaźnikami do operacji na pamięci bez sprawdzania obszarów granicznych, zwłaszcza takich jak `sprintf` lub `strcpy`, jest uznawane za przestarzałe, co ma zapobiegać błędom przepełnienia buforu. Zamiast nich należy używać `sprintf` oraz `strncpy`.

13.7.2 Odpluskwianie

Uruchamianie i testowanie programów przy pomocy `gdb`

Przygotowanie:

```
# apt-get install gdb
```

Miejsca, w których można znaleźć informacje o `gdb`:
- `info gdb` (samouczek)
- `gdb(1)`
- `http://www.unknownroad.com/rtfm/gdbtut/gdbtoc.html`

Do prześledzenia działania programu skompilowanego z opcją `-g` użyj programu `gdb`. Wiele poleceń ma swoje kilkuznakowe skróty. Dopełnianie tabulatorem działa tak samo jak w powłoce.

```
$ gdb program
(gdb) b 1                # ustaw punkt wstrzymania w linii 1
(gdb) run arg1 arg2 arg3 # uruchom program
(gdb) next               # następna linia
...
(gdb) step               # krok do przodu
...
(gdb) p parm             # wyświetl parm
...
(gdb) p parm=12          # ustaw wartość na 12
```

Jeśli chcesz się dowiedzieć, jak przeprowadzić odpluskwianie przy pomocy Emacsa, zerknij do 'Podsumowanie poleceń edytorów (Emacs, Vim)' na 215 stronie.

Sprawdzanie zależności od bibliotek

Aby znaleźć wymagane przez program biblioteki skorzystaj z `ldd`:

```
$ ldd /bin/ls
        librt.so.1 => /lib/librt.so.1 (0x4001e000)
        libc.so.6 => /lib/libc.so.6 (0x40030000)
        libpthread.so.0 => /lib/libpthread.so.0 (0x40153000)
        /lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

By `ls` mogło pracować w środowisku `chroot`, muszą się w nim znaleźć powyższe biblioteki.

Następujące polecenia mogą być również użyteczne:
- `strace`: śledzi wywołania systemowe oraz sygnały
- `ltrace`: śledzi wywołania biblioteczne

Odpluskwianie z użyciem narzędzi do wykrywania wycieków pamięci

Istnieje wiele narzędzi do wykrywania wycieków pamięci w Debianie:
- `njamd`
- `valgrind`
- `dmalloc`
- `electric-fence`
- `memprof`
- `memwatch` (niedostępny w pakiecie, pobierz z memwatch (`http://directory.fsf.org/devel/debug/memwatch.html`).)
- `mpatrol`
- `leaktracer`
- `libgc6`
- `Insure++` z Parasoft (`http://www.parasoft.com`). (non-free, komercyjnie za opłatą)

Sprawdź również Debugging Tools for Dynamic Storage Allocation and Memory Management (`http://www.cs.colorado.edu/homes/zorn/public_html/MallocDebug.html`).

13.7.3 Flex – lepszy Lex

`flex` jest szybkim generatorem analizatorów leksykalnych.

Miejsca, w których można znaleźć informacje o `flex`:
- `info flex` (samouczek)
- `flex(1)`

Musisz utworzyć własną funkcję `main()` i `yywrap()`, lub `program.l` powinien wyglądać tak, jak poniżej, by kompilował się bez biblioteki (`yywrap` jest makrem; `%option main` uaktywnia bezwarunkowo `%option noyywrap`):

```
%option main
%%
.|\n    ECHO ;
%%
```

Zamiast tego możesz skompilować program podając na końcu polecenia `cc` opcję `-lfl` (tak jak przy AT&T-Lex opcję `-ll`). Nie potrzeba wtedy używać `%option`.

13.7.4 Bison – lepszy Yacc

Kilka pakietów w Debianie dostarcza zgodny z Yacc generator analizatorów składniowych:

- `bison`: GNU LALR parser generator
- `byacc`: The Berkeley LALR parser generator
- `btyacc`: Backtracking parser generator oparty na `byacc`

Miejsca, w których można znaleźć informacje o `bison`:

- `info bison` (samouczek)
- `bison(1)`

Musisz utworzyć własne funkcje `main()` i `yyerror()`. `main()` wywołuje `yyparse()`, która z kolei wywołuje `yylex()`, zazwyczaj stworzoną przez FleX.

```
%%

%%
```

13.7.5 Autoconf

`autoconf` jest narzędziem tworzącym skrypty powłoki, które automatycznie konfigurują źródła oprogramowania, by dostosować je do wielu systemów uniksopodobnych, używając systemu budowania GNU.

`autoconf` tworzy skrypt konfigurujący `configure`, a `configure` automatycznie dostosowany `Makefile` używając wzorca `Makefile.in`.

Kompilacja i instalacja programu

Debian nie dotyka plików znajdujących się w `/usr/local` (zobacz 'Wsparcie dla różnorodności' na 24 stronie). W związku z tym, jeśli skompilujesz program ze źródeł oraz zainstalujesz go w `/usr/local` nie będzie on kolidował z Debianem.

```
$ cd src
$ ./configure --prefix=/usr/local
$ make
$ make install # to polecenie przekopiuje pliki do systemu
```

Odinstalowywanie programu

Jeśli nadal posiadasz źródła i używają one `autoconf/automake` oraz pamiętasz parametry z jakimi je konfigurowałeś spróbuj:

```
$ ./configure wszelkie-opcje-jakie-podałeś-przy-konfiguracji
# make uninstall
```

Zamiast tego, jeśli jesteś absolutnie pewien, że podczas instalacji pliki kopiowane są tylko do `/usr/local` i nie ma tam niczego ważnego, możesz usunąć jego zawartość wprowadzając:

```
# find /usr/local -type f -print0 | xargs -0 rm -f
```

Jeśli natomiast nie jesteś pewien gdzie zostały przekopiowane pliki podczas instalacji, rozważ zastosowanie `checkinstall`, który pozwala w łatwy sposób odinstalować programy.

13.8 Przygotowanie dokumentu

13.8.1 Skład tekstu `roff`

Tradycyjnie już, `roff` jest głównym systemem procesu tekstu w systemach Unix.

Zobacz `roff(7)`, `groff(7)`, `groff(1)`, `grotty(1)`, `troff(1)`, `groff_mdoc(7)`, `groff_man(7)`,`groff_ms(7)`,`groff_me(7)`,`groff_mm(7)` oraz `info groff`.

Istnieje dobre wprowadzenie do makr `-me`. Jeśli posiadasz `groff` (w wersji 1.18 lub nowszej), znajdź `/usr/share/doc/groff/meintro.me.gz` i wprowadź poniższe polecenie:

```
$ zcat /usr/share/doc/groff/meintro.me.gz | \
     groff -Tascii -me - | less -R
```

Kolejne polecenie utworzy zwykły plik tekstowy:

```
$ zcat /usr/share/doc/groff/meintro.me.gz | \
    GROFF_NO_SGR=1 groff -Tascii -me - | col -b -x > meintro.txt
```

Dla drukowania skorzystaj z wyjścia PostScript.

```
$ groff -Tps meintro.txt | lpr
$ groff -Tps meintro.txt | mpage -2 | lpr
```

13.8.2 SGML

Przygotowanie:

```
# apt-get install debiandoc-sgml debiandoc-sgml-doc
```

Miejsca, w których można znaleźć informacje o `debiandoc-sgml`:

- `/usr/share/doc/debiandoc-sgml-doc`
- `debiandoc-sgml(1)`
- *DocBook: The Definitive Guide (`/usr/share/doc/docbook-defguide/html/docbook.html`)*, by Walsh and Muellner, (O'Reilly) (pakiet `docbook-defguide`)

SGML umożliwia zarządzanie wieloma formatami dokumentu. Jednym z prostszych systemów SGML jest Debiandoc, który jest tutaj używany. Wymaga on pewnych konwersji plików tekstowych dla następujących znaków:

- „<” –> `<`
- „>” –> `>`
- „ ” –> ` ` (niełamana spacja)
- „&” –> `&`
- „%” –> `%`
- „©” –> `©`
- „–” –> `–`
- „—” –> `—`

W celu oznaczenia części jako niedrukowalny komentarz wprowadź:

```
<!-- A tu jest właśnie komentarz ... -->
```

Do oznaczenia miejsca, z którym mamy problemy (brak wiedzy, doświadczenia z danego zakresu) użyj poniższej formy:

```
<![ %FIXME; [ Nie wiem, co tu powinno być ... ]]>
```

W SGMLu *pierwsza definicja* entitki wygrywa. Przykład:

```
<!entity % qref "INCLUDE">
<![ %qref; [ <!entity param "Data 1"> ]]>
<!entity param "Data 2">
&param;
```

W dokumencie wynikowym będzie „Data 1”. Jeśli w pierwszej linii byłoby „IGNORE” zamiast „INCLUDE”, pojawiłoby się „Data 2” (druga linia jest wyrażeniem warunkowym). Również powtórzone zwroty, wyrażenia mogą być z góry oddzielone od kontekstu.

```
<!entity czyjtojest "mój">
Witaj  &czyjtojest; przyjacielu.
To jest &czyjtojest; podręcznik.
```

Efektem będzie:

```
Witaj mój przyjacielu.
To jest mój podręcznik.
```

Przyjrzyj się prostemu przykładowi SGML `sample.sgml`, znajdującego się w examples (`http://www.debian.org/doc/manuals/debian-reference/examples/`).

Niekiedy, gdy dokument SGML rozrasta się, TeX (używany jako silnik do przetwarzania tekstu) może zacząć generować błędy. Więcej informacji tutaj: 'TeX/LaTeX' na tej samej stronie.

13.8.3 TeX/LaTeX

Przygotowanie:

```
  # tasksel # select Miscellaneous  --> TeX/LaTeX environment
```

Odsyłacze poświęcone LaTeXowi:

- The teTeX HOWTO: The Linux-teTeX Local Guide (`http://www.tldp.org/HOWTO/TeTeX-HOWTO.html`)
- `tex(1)`
- `latex(1)`
- *The TeXbook*, by Donald E. Knuth, (Addison-Wesley) [1]
- *LaTeX - A Document Preparation System*, by Leslie Lamport, (Addison-Wesley)
- *The LaTeX Companion*, by Goossens, Mittelbach, Samarin, (Addison-Wesley)

TeX to najpotężniejsze środowisko składu dokumentów. Wiele procesorów SGML używa go do przetwarzania tekstu. Lyx dostarczany w pakietach `lyx`, `lyx-xforms` oraz `lyx-qt` oferuje przyjemne środowisko WYSIWYG dla LaTeXa, jednakże wiele osób wybiera Emacsa lub Vima do tworzenia źródeł (La)TeXowych.

Istnieje wiele dokumentacji i zasobów dostępnych w Sieci:

- teTeX - A Documentation Guide (`/usr/share/doc/texmf/newhelpindex.html`) (`tetex-doc` package)
- A Quick Introduction to LaTeX (`http://www.msu.edu/user/pfaffben/writings/`)

---

[1] TeXowe źródło tej książki jest dostępne tutaj: `ftp://ftp.dante.de/pub/tex/systems/knuth/tex/texbook.tex`. `ftp://ftp.dante.de/pub/tex/systems/knuth/lib/manmac.tex` zawiera większość potrzebnych makrodefinicji. Można przetworzyć ten dokument programem `tex` po zakomentowaniu linii od 7 do 10 i dodaniu `\input manmac \proofmodefalse`. Jakkolwiek silnie polecamy zakup tej książki (oraz innych autorstwa Donalda E. Knutha) zamiast korzystania z tej wersji, to jest ona wspaniałym przykładem źródła dokumentu TeXowego.

- A Simple Guide to Latex/Lyx (`http://www.stat.rice.edu/~helpdesk/howto/lyxguide.html`)
- Word Processing Using LaTeX (`http://www-h.eng.cam.ac.uk/help/tpl/textprocessing/latex_basic/latex_basic.html`)
- Local User Guide to teTeX/LaTeX (`http://supportweb.cs.bham.ac.uk/documentation/LaTeX/lguide/local-guide/local-guide.html`)

Gdy dokument rozrasta się, TeX może zacząć generować błędy. Żeby uporać się z tym problemem, musisz zwiększyć parametr pool size w `/etc/texmf/texmf.cnf` (lub, co bardziej zalecane, zmodyfikować `/etc/texmf/texfm.d/95NonPath` i uruchomić `update-texmf`).

13.8.4 Literat-programista

Zamiast pisania kodu zawierającego dokumentację, literat-programista pisze dokumentację zawierajacą kod. To zapewnia dobrą dokumentację dla programu.

Więcej na ten temat znajduje się tutaj: Literackie programowanie (`http://www.literateprogramming.com/`).

Noweb

Przygotowanie:

```
# apt-get install nowebm
```

Odsyłacze na temat Noweb:
- Noweb — A Simple, Extensible Tool for Literate Programming (`http://www.eecs.harvard.edu/~nr/noweb/`)
- `noweb(1)`

Jest to narzędzie w stylu WEB, które jest prostsze oraz zapewnia rozszerzalność i niezależność od języka. [2] Kiedy `noweb` zostanie użyty, wypisuje źródło programu do plików wyjściowych wskazanych w pliku noweb oraz tworzy pliki w formacie TeX składające się na dokumentację.

Pakiet `ifupdown` to świetny przykład.

```
$ apt-get source ifupdown
$ cd ifupdown*
$ make ifupdown.pdf ifupdown.ps
```

Doxygen

Przygotowanie:

---

[2] WEB nie ma nic **nic** wspólnego z World Wide Web. WEB (dla Pascala) oraz CWEB (dla C/C++) to tradycyjne narzędzia programowania literackiego.

```
    # apt-get install doxygen doxygen-doc doxygen-gui
```

Odsyłacze na temat Doxygen (tworzone przez `doxygen`!):

- Homepage (`http://www.doxygen.org/`)
- `/usr/share/doc/doxygen-doc/html/index.html`

Narzędzie potrafi tworzyć pliki HTML, RTF, podręcznika systemowego Unix, PostScript oraz PDF (z użyciem LaTeXa) stanowiące dokumentację do C++, C, Jawy, IDL oraz w pewnej mierze programów PHP i C#. Doxygen jest zgodny z JavaDoc (1.1), Qt-Doc oraz KDOC i został zaprojektowany specjalnie dla projektów wykorzystujących firmowane przez Troll Techa Qt (`http://www.trolltech.no/qt/`)rozwiązania. Program tworzy drzewo zależności, diagramy współpracy oraz graficzną hierarchię klas (nawet dla nieudokumentowanych programów). W efekcie powstaje dokumentacja podobna do dostarczanej z bibliotekami Qt.

13.9 Pakietowanie

Przygotowanie:

```
# apt-get install debian-policy developers-reference \
                  maint-guide dh-make debhelper
# apt-get install packaging-manual # dla Potato
```

Miejsca, w których można znaleźć informacje o pakietowaniu:

- 'System zarządzania pakietami w Debianie' na 12 stronie (podstawy)
- Debian New Maintainers' Guide (samouczek)
- `dh-make(1)`
- Debian Developer's Reference (najlepsza praktyka)
- Debian Policy Manual (autorytatywny)
- Packaging Manual (Potato)

13.9.1 Pakietowanie pojedyńczego pliku binarnego

Szybka i brzydka metoda pakietowania podesłana przez Joey Hessa.

```
# mkdir -p mypkg/usr/bin mypkg/DEBIAN
# cp binary mypkg/usr/bin
# cat > mypkg/DEBIAN/control
Package: mojpakiet
Version: 1
Architecture: i386
Maintainer: Joey Hess <joeyh@debian.org>
Description: mój mały pakiecik
 Nie spodziewaj się zbyt wiele.
^D
# dpkg-deb -b mypkg
```

13.9.2 Pakietowanie z użyciem narzędzi

W celu utworzenia podstawowego pakietu możesz skorzystać z `dh_make` dostępnego w pakiecie `dh-make`. Następnie postępuj według instrukcji z `dh-make(1)`. Ten sposób wykorzystuje `debhelper` w `debian/rules`.

Starszym sposobem jest użycie `deb-make` z pakietu `debmake`. Wtedy korzysta się jedynie z powłoki, zamiast z `debhelper`.

Przykładem wieloźródłowego pakietu może być „mc" (`dpkg-source -x mc_4.5.54.dsc`), korzystający z „sys-build.mk" Adama Heatha (<`doogie@debian.org`>), oraz „glibc" (`dpkg-source -x glibc_2.2.4-1.dsc`), który korzysta z innego systemu Joela Kleckera (<`espy@debian.org`>).

Rozdział 14

GnuPG

Odnośniki:

- `gpg(1)`.
- `/usr/share/doc/gnupg/README.gz`
- *GNU privacy handbook* w `/usr/share/doc/gnupg-doc/GNU_Privacy_Handbook/` (zainstaluj pakiet `gnupg-doc`)

14.1 Instalacja GnuPG

```
$ gpg --gen-key
            # tworzenie nowej pary kluczy
$ gpg --gen-revoke ID_użytk
            # tworzenie certyfikatu unieważnienia klucza dla ID_użytkownika
$ host -l pgp.net | grep www | less
            # znajdowanie serwerów kluczy
```

Jak na razie, dobrymi serwerami kluczy są:

```
keyserver wwwkeys.eu.pgp.net
keyserver wwwkeys.pgp.net
```

Musisz uważać, aby **nie tworzyć więcej, niż dwa podklucze**. Jeśli tak zrobisz, serwery kluczy w pgp.net **popsują** Twój klucz. Użyj nowszego `gnupg` (>1.2.1-2), aby obsłużyć te popsute podklucze. Zobacz `http://fortytwo.ch/gpg/subkeys`.

Miej również na uwadze, że w `$HOME/.gnupg/options` można ustawić tylko jeden serwer kluczy.

Niestety, następująca konstrukcja już nie działa:

```
keyserver search.keyserver.net
keyserver pgp.ai.mit.edu
```

14.2 Używanie GnuPG

Obsługa plików:

```
$ gpg  [opcje]  polecenie  [argumenty]
$ gpg {--armor|-a} {--sign|-s} plik
            # zapisanie podpisanego pliku do pliku tekstowego plik.asc
$ gpg --clearsign plik
            # złożenie podpisu z zachowaniem czytelności tekstu
$ gpg --clearsign --not-dash-escaped plik_łaty
            # podpisanie z zachowaniem czytelności pliku łaty
$ gpg --verify plik
            # sprawdzenie podpisu pliku
$ gpg -o plik.sig {-b|--detach-sig} plik
            # złożenie podpisu oddzielnego od dokumentu
$ gpg --verify plik.sig plik
            # sprawdzenie oddzielonego podpisu plik.sig
$ gpg -o plik_wynik {--recipient|-r} nazwa {--encrypt|-e} plik
            # zaszyfrowanie pliku kluczem publicznym użytkownika o danej nazwie
$ gpg -o plik_wynik {--symmetric|-c} plik
            # szyfrowanie algorytmem symetrycznym
$ gpg -o file --decrypt plik_szyfr
            # odszyfrowanie danych z plik_szyfr
```

14.3 Utrzymywanie GnuPG

Zarządzanie kluczami:

```
$ gpg --edit-key ID_użytk
            # "`help"' w celu uzyskania pomocy, tryb interaktywny
$ gpg -o plik --exports
            # eksport wszystkich kluczy do pliku
$ gpg --imports plik
            # import wszystkich kluczy z pliku
$ gpg --send-keys ID_użytk
            # eksport klucza o podanym ID_użytkownika do serwera kluczy
$ gpg --recv-keys ID_użytk
            # pobieranie klucza o podanym ID_użytkownika z serwera kluczy
$ gpg --list-keys ID_użytk
            # wypisanie kluczy o podanym ID_użytkownika
$ gpg --list-sigs ID_użytk
            # wypisanie podpisów kluczy o podanym ID_użytkownika
$ gpg --check-sigs ID_użytk
```

```
            # sprawdzenie podpisów na kluczach o podanym ID_użytkownika
$ gpg --fingerprint ID_użytk
            # sprawdzenie odcisku klucza o danym ID_użytkownika
$ gpg --list-sigs | grep '^sig' | grep '[User id not found]' \
  | awk '{print $2}' | sort -u | xargs gpg --recv-keys # get unknown keys
            # pobranie kluczy dla każdego nieznanego podpisu
```

Kod zaufania:

`-`	`Brak przypisanego poziomu zaufania / jeszcze nie obliczony.`
`e`	`Obliczanie poziomu zaufania nie powiodło się.`
`q`	`Zbyt mało informacji do obliczeń.`
`n`	`Nie ufaj temu kluczowi.`
`m`	`Częściowo zaufany.`
`f`	`W pełni zaufany.`
`u`	`Zaufanie bezgraniczne.`

Następujące polecenie eksportuje mój klucz *„A8061F32”* do wielu serwerów:

```
$ for xx in us es cz de dk uk ch net.uk earth.net.uk; \
$ do gpg --keyserver wwwkeys.$xx.pgp.net  --send-keys A8061F32; done
```

14.4 Używanie GnuPG z innymi programami

14.4.1 Używanie GnuPG z Muttem

Aby powstrzymać wolne GnuPG od automatycznego startowania, pozwalając jednocześnie na użycie go przez wpisanie ‘`S`’, dodaj następujący wpis do `~/.muttrc`:

```
macro index S ":toggle pgp_verify_sig\n"
set pgp_verify_sig=no
```

14.4.2 Używanie GnuPG z Vimem

Aby w sposób przezroczysty używać GnuPG, dodaj zawartość pliku `_vimrc` z katalogu z przykładami (`http://www.debian.org/doc/manuals/debian-reference/examples/`) do `~/.vimrc`.

Rozdział 15

Rozwiązywanie problemów

Poniższe zasoby dostarczają pomocy, rad i wsparcia dla użytkowników. Postaraj się najpierw użyć tych zasobów zanim zaczniesz płakać na listach dyskusyjnych :).

W swoim systemie masz dostep do obszernej dokumentacji za pomoca programow `dhelp`, `dwww` i dowolnej przegladarki WWW.

15.1 Zasoby

Poniższe zasoby są pomocne dla rozwiązywania problemów z Debianem i Linuksem w ogólności. Jeśli ich treść nie pokrywa się w pewnych miejscach, to zawsze bardziej ufaj zasobom oznaczonym jako podstawowe.

- Podręcznik instalacji (podstawowe)
  - Przeczytaj przed instalacją lub aktualizacją.
  - WWW: `http://www.debian.org/releases/stable/installmanual`
  - WWW: `http://www.debian.org/releases/testing/installmanual` (w przygotowaniu, czasami może nie działać)
  - Pakiet: `Not available in install-doc:  Bug#155374`
  - Plik: `DebianCDunder/doc/`

- Notki o wydaniu (podstawowe)
  - Koniecznie przeczytaj przed instalacją lub aktualizacją, nawet jeśli jesteś doświadczonym użytkownikiem.
  - WWW: `http://www.debian.org/releases/stable/releasenotes`
  - WWW: `http://www.debian.org/releases/testing/releasenotes` (w przygotowaniu, czasami może nie istnieć)
  - Pakiet: `Not available in install-doc:  Bug#155374`
  - Plik: `DebianCDunder/doc/`

- FAQ (dodatkowe)

  - Najczęściej zadawane pytania
  - WWW: `http://www.debian.org/doc/manuals/debian-faq/`
  - Pakiet: `doc-debian`
  - Plik: `/usr/share/doc/debian/FAQ/index.html`

- Debian Reference (dodatkowe)
  - Najbardziej szczegółowy, poinstalacyjny podręcznik użytkownika.
  - WWW: `http://www.debian.org/doc/manuals/debian-reference/`
  - Pakiet: `debian-reference-en`
  - Plik: `/usr/share/doc/Debian/reference/`

- APT HOWTO (dodatkowe)
  - Szczegółowy podręcznik użytkownika do systemu zarządzania pakietami Debiana (Woody).
  - WWW: `http://www.debian.org/doc/manuals/apt-howto/`
  - Pakiet: `apt-howto`
  - Plik: `/usr/share/doc/Debian/apt-howto/`

- Securing Debian Manual (dodatkowe)
  - Szczegółowy podręcznik użytkownika traktujący o zabezpieczaniu domyślnej instalacji Debiana (Woody).
  - WWW: `http://www.debian.org/doc/manuals/securing-debian-howto/`
  - Pakiet: `harden-doc`
  - Plik: `/usr/share/doc/harden-doc/html/securing-debian-howto/`

- `dselect` Dokumentacja dla początkujących (dodatkowe)
  - Podręcznik `dselecta`
  - WWW: `http://www.debian.org/releases/woody/i386/dselect-beginner`
  - Pakiet: `Not available in install-doc: Bug#155374`
  - Plik: `DebianCDunder/doc/`

- Debian Policy Manual (podstawowe)
  - Techniczne kulisy Debiana.
  - WWW: `http://www.debian.org/doc/debian-policy/`
  - Pakiet: `debian-policy`
  - Plik: `/usr/share/doc/debian-policy/`

- Debian Developer's Reference (podstawowe)
  - Podstawowa wiedza dla deweloperów.
  - Wszyscy powinniśmy przejrzeć ten dokument.
  - WWW: `http://www.debian.org/doc/manuals/developers-reference/`
  - Pakiet: `developers-reference`
  - Plik: `/usr/share/doc/developers-reference/`

- Debian New Maintainers' Guide (podstawowe)

  - Praktyczny podręcznik dla deweloperów.
  - Wprowadzenie do pakietowania dla wszystkich.
  - WWW: `http://www.debian.org/doc/manuals/maint-guide/`
  - Pakiet: `maint-guide`
  - Plik: `/usr/share/doc/maint-guide/`

- Wprowadzenie do pakietowania (Potato)
  - Pakiet `packaging-manual` w Potato. (przeniesiony do dodatków *Developer's Reference*)

- Strony podręcznika man w stylu Uniksa (podstawowe)
  - `man` *`nazwa-pakietu`*

- Strony podręcznika info w stylu GNU (podstawowe)
  - `info` *`nazwa-pakietu`*

- Dokumenty specyficzne dla poszczególnych pakietów (podstawowe)
  - Znajdziesz je w `/usr/share/doc/`*`nazwa-pakietu`*

- LDP: Linux Documentation Project (dodatkowe)
  - Ogólnolinuksowe HOWTO i mini-HOWTO
  - WWW: `http://www.tldp.org/`
  - Pakiet: `doc-linux-text`
  - Plik: `/usr/share/doc/HOWTO/`

- DDP: Debian Documentation Project (dodatkowe)
  - Podręczniki traktujące o Debianie
  - WWW: `http://www.debian.org/doc/`

- Debian Developers' Corner (dodatkowe)
  - Kluczowe informacje dla deweloperów Debiana
  - Raczej nie dla zwykłego użytkownika
  - WWW: `http://www.debian.org/devel/`

- Kod źródłowy (całkowicie podstawowe)
  - Nikt nie może temu zaprzeczyć :-)
  - Ściągnij kod źródłowy z 'Kod źródłowy' na 11 stronie

Poniższe zasoby traktują o Uniksie w ogólności. Zauważ, że jest wiele kluczowych różnic pomiędzy różnymi systemami Uniksowymi. Nazwy urządzeń i sposoby inicjalizacji systemu wymagają specjalnego zainteresowania.

- *The UNIX Programming Environment*
  - Książka opisująca działania Uniksa.
  - Autorzy: B. W. Kernighan i R. Pike,
  - Wydana przez Princeton Hall Software Series

- *Język ANSI C* (wyd. polskie)
    - Książka traktująca o ANSI C
    - Autorzy: B. W. Kernighan i D. M. Ritchie
    - Wydana przez Wydawnictwa Naukowo-Techniczne

- *UNIX Power Tools*
    - Książka traktująca o różnych sztuczkach w systemie Unix.
    - Autorzy: Jerry Peek, Tim O'Reilly i Mike Loukides
    - Wydana przez O'Reilly and Associates

- *Essential System Administration* (second edition)
    - Książka o administracji systemami Unix
    - Autor: Aeleen Frisch
    - Wydana przez O'Reilly and Associates

- Bell Labs: Computing Sciences Research
    - Bogate archiwum historii Uniksa
    - Główne: `http://cm.bell-labs.com/cm/cs/`
    - Wybrane raporty techniczne: `http://cm.bell-labs.com/cm/cs/cstr.html`
    - Kilka dokumentów: `http://cm.bell-labs.com/cm/cs/papers.html`

- Dostępne w sieci serwisy traktujące o Linuksie i Debianie
    - Debian Planet (`http://www.debianplanet.org/`)
    - debianHELP (`http://www.debianhelp.org/`)
    - Linux.com (`http://linux.com/`)
    - The Linux Home Page at Linux Online (`http://www.linux.org/`)
    - Red Hat (commercial Linux vender) (`http://www.redhat.com/`) (RPM, Sys-V init)
    - SuSE, Inc. (commercial Linux vender) (`http://www.suse.de/`) (RPM, Sys-V init)
    - Slackware (`http://www.slackware.com/`) (TGZ, BSD-style init)

- Dostępne w sieci zasoby traktujące o Uniksie
    - A UNIX Introductory Course from Ohio State University (`http://www-wks.acs.ohio-state.edu/unix_course/unix.html`)
    - UNIXhelp from The University of Edinburgh (`http://unixhelp.ed.ac.uk/`)
    - Unix / Programming Information (`http://arioch.unomaha.edu/~jclark/#info`)
    - comp.unix.questions FAQ (`http://www.faqs.org/faqs/unix-faq/faq/`)
    - comp.unix.user-friendly FAQ (`http://www.camelcity.com/~noel/usenet/cuuf-FAQ.htm`)
    - FreeBSD Documentation (`http://www.freebsd.org/docs.html`)
    - The FreeBSD Handbook (`http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/index.html`)
    - UNIX GUIDE (`http://ieee.uow.edu.au/documents/`)
    - The Unix Heritage Society (`http://www.tuhs.org/`)

- Strony domowe projektu Free Software

- GNU Project (`http://www.gnu.org/`)
- The Linux Documentation Project (`http://www.tldp.org/`)
- The Linux Kernel Archives (`http://www.linux.org/`)
- The XFree86 Project, Inc (`http://www.xfree86.org/`)
- GNOME (`http://www.gnome.org/`)
- K Desktop Environment (`http://www.kde.org/`)
- GNU software at Red Hat (`http://sources.redhat.com/`)
- Mozilla (`http://www.mozilla.org`)
- FreeBSD (`http://www.freebsd.org/`)
- OpenBSD (`http://www.openbsd.org/`)
- NetBSD (`http://www.netbsd.org/`)

15.2 Znajdowanie znaczenia słowa

Wiele słów używanych w Debianie to tajemniczy żargon lub akronimy. Poniższe polecenie pomoże Ci zrozumieć większość z nich:

```
$ dict wstaw-tu-dziwne-angielskie-słowo
```

15.3 Sprawdzanie popularności pakietu Debiana

W Debianie istnieje wiele pakietów i czasami trudno jest się zdecydować, który chcemy najpierw wypróbować. Zobacz Debian Popularity Contest Results (`http://popcon.debian.org/`), by dowiedzieć się, czego używają inni. Możesz również zainstalować pakiet `popularity-contest`, by mieć swój wkład w ten serwis.

15.4 System śledzenia błędów Debiana

Dystrybucja Debiana posiada system śledzenia błędów (BTS) (`http://bugs.debian.org/`), który zawiera ich listę zgłoszonych przez użytkowników i deweloperów. Każdy błąd ma przypisany swój numer i jest przechowywany, dopóki nie zostaje on oznaczony jako rozwiązany.

Zanim zgłosisz błąd, sprawdź czy ktoś inny tego nie zrobił. Lista aktualnych błędów jest dostępna na stronie BTS (`http://bugs.debian.org/`). Możliwe są również inne (`http://www.debian.org/Bugs/Access`) sposoby dotarcia do tych informacji . Zobacz także 'Sprawdzanie błędów w Debianie i poszukiwanie pomocy' na 90 stronie.

Sposób zgłaszania błędów jest opisany w `http://www.debian.org/Bugs/Reporting`.

15.5 Listy dyskusyjne

Aby być na bieżąco z dystrybucją Debiana, czytaj przynajmniej „debian-devel-announce" (angielska, tylko do odczytu, niewielki ruch).

Najbardziej interesujące użytkowników Debiana listy dyskusyjne to „debian-user" (angielska, otwarta dla każdego, wysoki ruch) oraz listy „debian-user-*język*" (dla innych języków).

Więcej informacji o tych listach i sposobach ich subskrypcji znajdziesz w `http://lists.debian.org/`. Przejrzyj archiwa w poszukiwaniu odpowiedzi na stawiane pytanie, zanim je wyślesz na listę. Stosuj się również do przyjętych zasad pisania na listy dyskusyjne.

Jeśli nie masz ochoty otrzymywać odpowiedzi także w postaci kopii na prywatny adres, użyj pola `Mail-Followup-To:` z nagłówka, które jest bardzo efektywnym ogranicznikiem.

Postępowanie takie jest to nieformalna konwencja list dyskusyjnych wyjaśniona w `http://cr.yp.to/proto/replyto.html`.

15.6 Internet Relay Chat (IRC)

IRC (Internet Relay Chat) jest sposobem na rozmowę z innymi ludźmi z całego świata w czasie rzeczywistym. Kanały IRC poświęcone Debianowi znajdują się w sieci IRC freenode (`http://www.freenode.info/`). By się połączyć potrzebujesz klienta IRC. Kilka z najbardziej popularnych klientów to XChat, BitchX, ircII, irssi, epic4, KSirc. Wszystkie te są dostępne w Debianie w postaci pakietów. Gdy już masz zainstalowanego klienta IRC, połącz się z serwerem. W większości klientów należy wydać polecenie:

```
/server irc.debian.org
```

Po połączeniu wejdź na kanał #debian wpisując

```
/join #debian
```

Aby opuścić kanał #debian, napisz:

```
/part #debian
```

Możesz także całkowicie zamknąć klienta IRC poprzez:

```
/quit
```

Aby wysłać prywatną wiadomość „Cześć Jasiu" do użytkownika *jasiu*, napisz:

```
/msg jasiu Cześć Jasiu
```

Pamiętaj, że wszystko, co jest wpisywane bez poprzedzającego znaku `/`, będzie wysyłane bezpośrednio jako wiadomość do kanału.

Uwaga: klienty takie jak XChat często posiadają różne graficzne interfejsy użytkownika do łączenia się z serwerem i wchodzenia na kanały.

15.7 Wyszukiwarki

Jest wiele wyszukiwarek, które pomagają w przeszukiwaniu dokumentacji Debiana:

- Debian WWW search site (`http://search.debian.org/`).
- Google (`http://www.google.com/`): w polu tekstowym dodaj na końcu „site:debian.org".
- Google Groups (`http://groups.google.com/`): wyszukiwarka grup dyskusyjnych. Dodaj do pola tekstowego „group:linux.debian.*".
- AltaVista (`http://www.altavista.com/`)

Szukając na przykład ciągu „cgi-perl", otrzymasz bardziej szczegółowe informacje o tym pakiecie, niż krótki opis z pliku control. Więcej rad na stronie 'Sprawdzanie błędów w Debianie i poszukiwanie pomocy' na 90 stronie.

15.8 Strony w sieci

Poniżej znajduje się lista kilku wybranych odnośników na różne okazje.

- IBM developerWorks: Linux (`http://www.ibm.com/developerworks/linux/`)
- Adrian Bunk's latest packages (back port to stable) (`http://www.fs.tum.de/~bunk/`)
- Linux on Laptops (`http://www.linux-laptop.net/`)
- Xterm FAQ (`http://dickey.his.com/xterm/xterm.faq.html`)
- EXT3 File System mini-HOWTO (`http://www.zip.com.au/~akpm/linux/ext3/ext3-usage.html`)
- Large File Support in Linux (`http://www.suse.de/~aj/linux_lfs.html`)
- Window Managers for X (`http://www.xwinman.org`)
- Linux USB Project (`http://www.linux-usb.org/`)

- SuSE pages for CJK (`http://www.suse.de/~mfabian/suse-cjk/suse-cjk.html`)
- LNX-BBC (Business-card-sized boot CD project) (`http://www.lnx-bbc.org/`)
- Linux info by Karsten Self (partitioning, backup, browsers. . .) (`http://kmself.home.netcom.com/Linux/index.html`)
- Backup info HOWTO by Alvin Oga (`http://www.Linux-Backup.net/`)
- Security info HOWTO by Alvin Oga (`http://www.Linux-Sec.net/`)
- Various UNOFFICIAL sources for APT (`http://www.apt-get.org/`)
- Laptop Ethernet Configuration (`http://www.orthogony.com/gjw/lap/lap-ether-intro.html`)

Dodatek A

Dodatek

A.1 Autorzy

Debian Reference został zapoczątkowany przez Osamu Aoki `<osamu\#at\#debian.org>` jako osobisty pamiętnik z instalacji, który w końcu został nazwany „Quick Reference…”. Wiele treści znajdujących się tutaj pochodzi z archiwum listy dyskusyjnej „debian-user”. Zacytowano również fragmenty dokumentów „Debian Installation Manual” i „Debian Release Notes”.

Zgodnie z sugestią Josipa Rodina, który jest bardzo aktywnym członkiem Debian Documentation Project (`http://www.debian.org/doc/ddp`) (DDP) i aktualnym opiekunem „The Debian FAQ”, nazwa tego dokumentu została zmieniona na „Debian Reference”, a zawartość powstała z wybranych części „The Debian FAQ”. Następnie dokument został przemianowany na „Debian Quick Reference”.

Dokument był edytowany, tłumaczony i rozwijany przez następujących członków zespołu QREF :

- Pierwowzory dokumentów dla oryginalnego „Quick Reference…”
  - Osamu Aoki `<osamu\#at\#debian.org>` (główny koordynator)
- Angielska korekta i dodatki
  - David Sewell `<dsewell\#at\#virginia.edu>` (koordynator tłumaczenia na język angielski)
  - Thomas Hood `<jdthood\#at\#yahoo.co.uk>` (rzeczy związane z siecią)
  - Brian Nelson `<nelson\#at\#bignachos.com>` (głównie rzeczy związane z X)
  - Daniel Webb `<webb\#at\#robust.colorado.edu>`
  - Jan Michael C Alonzo `<jmalonzo\#at\#spaceants.net>`
  - Daniel Webb `<webb\#at\#robust.colorado.edu>`
  - Poprawki od wszystkich tłumaczy
- Tłumaczenie na język francuski
  - Guillaume Erbs `<gerbs\#at\#free.fr>` (koordynator tłumaczenia na język francuski)
  - Rénald Casagraude `<rcasagraude\#at\#interfaces.fr>`
  - Jean-Pierre Delange `<adeimantos\#at\#free.fr>

  - Daniel Desages `<daniel\#at\#desages.com>`
- Tłumaczenie na język włoski
  - Davide Di Lazzaro `<mc0315\#at\#mclink.it>` (koordynator tłumaczenia na język włoski)
- Tłumaczenie na język portugalski (brazylijski)
  - Paulo Rogério Ormenese `<pormenese\#at\#uol.com.br>` (koordynator tłumaczenia na język portugalski)
  - Andre Luis Lopes `<andrelop\#at\#ig.com.br>`
  - Marcio Roberto Teixeira `<marciotex\#at\#pop.com.br>`
  - Rildo Taveira de Oliveira `<to_rei\#at\#yahoo.com>`
  - Raphael Bittencourt Simoes Costa `<raphael-bsc\#at\#bol.com.br>`
  - Gustavo Noronha Silva `<kov\#at\#debian.org>` (koordynator)
- Tłumaczenie na język hiszpański
  - Walter Echarri `<wecharri\#at\#infovia.com.ar>` (koordynator tłumaczenia na język hiszpański)
  - José Carreiro `<ffx\#at\#urbanet.ch>`
- Tłumaczenie na język niemiecki
  - Jens Seidel `<tux-master\#at\#web.de>` (koordynator tłumaczenia na język niemiecki)
  - Willi Dyck `<wdyck\#at\#gmx.net>`
  - Stefan Schröder `<stefan\#at\#fkp.uni-hannover.de>`
  - Agon S. Buchholz `<asb\#at\#kefk.net>`
- Tłumaczenie na język polski — następujące osoby z projektu PDDP (`http://debian.linux.org.pl`):
  - Marcin Andruszkiewicz
  - Mariusz Centka `<mariusz.centka\#at\#debian.linux.org.pl>`
  - Bartosz Feński (koordynator tłumaczenia na język polski) `<fenio\#at\#debian.linux.org.pl>`
  - Radosław Grzanka `<radekg\#at\#debian.linux.org.pl>`
  - Bartosz 'Xebord' Janowski
  - Jacek Lachowicz
  - Rafał Michaluk
  - Leonard Milcin, Jr.
  - Tomasz Z. Napierała `<zen\#at\#debian.linux.org.pl>`
  - Oskar Ostafin `<cx\#at\#debian.linux.org.pl>`
  - Tomasz Piękoś
  - Jacek Politowski
  - Mateusz Prichacz `<mateusz\#at\#debian.linux.org.pl>`
  - Marcin Rogowski
  - Paweł Różański
  - Mariusz Strzelecki
  - Krzysztof Ścierski
  - Przemysław Adam Śmiejek `<tristan\#at\#debian.linux.org.pl>`
  - Krzysztof Szynter
  - Mateusz Tryka `<uszek\#at\#debian.linux.org.pl>`
  - Cezary Uchto

  - Krzysztof Witkowski <`tjup\#at\#debian.linux.org.pl`>
  - Bartosz Zapałowski <`zapal\#at\#debian.linux.org.pl`>
- Tłumaczenie na język chiński (uproszczony)
  - Hao „Lyoo” LIU <`iamlyoo\#at\#163.net`>
  - Ming Hua <`minghua\#at\#rice.edu`>
  - Xiao Sheng Wen <`atzlinux\#at\#163.com`> (leader: zh-cn)
  - Haifeng Chen <`optical.dlz\#at\#gmail.com`>
  - Xie Yanbo <`xieyanbo\#at\#gmail.com`>
  - easthero <`easthero\#at\#gmail.com`>
- Tłumaczenie na język chiński (tradycyjny)
  - Asho Yeh <`asho\#at\#debian.org.tw`> (koordynator: zh-tw)
  - Tang Wei Ching <`wctang\#at\#csie.nctu.edu.tw`> (były koordynator: zh-tw)
- Tłumaczenie na język japoński
  - Shinichi Tsunoda <`tsuno\#at\#ngy.1st.ne.jp`> (koordynator: ja)
  - Osamu Aoki <`osamu\#at\#debian.org`>

QREF była skróconą formą pierwszego tytułu dokumentu, „Quick Reference…” teraz jest również nazwą projektu na qref.sourceforge.net.

Wiele stron podręcznika systemowego i info w systemie Debian zostało użytych jako pierwotne źródło informacji przy pisaniu tego dokumentu. W pewnej mierze Osamu Aoki rozważał uporządkowanie stylu i ich rozwinięcie, wiele ich części, głównie definicje poleceń, zostało użytych jako części zdań, po ich dostosowaniu do stylu i tematu tego dokumentu.

Większość 'Debian – Podstawy' na 5 stronie pochodzi z „The Debian FAQ” (Marzec 2002):
- 5. Archiwa FTP Debiana : `ftparchives.sgml` (cały rozdział)
- 6. Podstawy zarządzania systemem pakietów Debiana : `pkg_basics.sgml` (cały rozdział)
- 7. Narzędzia do zarządzania pakietami w Debianie : `pkgtools.sgml` (cały rozdział)
- 8. Aktualizowanie systemu Debian : `uptodate.sgml` (cały rozdział)
- 9. Debian i jądro systemu : `kernel.sgml` (cały rozdział)
- 10. Dostosowywanie systemu Debian do Twoich potrzeb : `customizing.sgml` (część rozdziału)

Powyższe części z „The Debian FAQ” zostały dołączone do tego dokumentu po znacznej reorganizacji, aby odzwierciedlić zmiany w systemie Debian. Obecnie obydwa dokumenty są aktualizowane równolegle.

Pierwotnie opiekunami „Debian FAQ”, byli jego twórcy — J. H. M. Dassen (Ray) i Chuck Stickelman. Autorami przepisanej wersji „Debian FAQ” byli Susan G. Kleinmann i Sven Rudolph. Następnym opiekunem był Santiago Vila. Aktualnie dokumentem tym zajmuje się Josip Rodin.

Informacje zawarte w „The Debian FAQ” pochodzą między innymi z/od:
- The Debian-1.1 release announcement, autor Bruce Perens (`http://www.perens.com/`).
- The Linux FAQ, autor Ian Jackson (`http://www.chiark.greenend.org.uk/~ijackson/`).

- Archiwów list dyskusyjnych Debiana (`http://lists.debian.org/`),
- Podręcznika dla programistów dpkg i The Debian Policy manual (zobacz 'Zasoby' na 251 stronie)
- wielu deweloperów, wolontariuszy i beta testerów, oraz
- szczątkowych zapisków z pamiętnika autora tego dokumentu. :-)

Pewne części „Samouczków Debiana" zostały zapożyczone z

- „Debian Tutorial" autostwa Havoca Penningtona, Olivera Elphicka, Ole Tetlie'a, Jamesa Treacy'go, Craiga Sawyera, i Ivana E. Moore'a II. (Ten dokument z kolei powstał na bazie „Linux User's Guide" autorstwa Larry'ego Greenfielda.)
- „Debian GNU/Linux: Guide to Installation and Usage" autorstwa Johna Goerzena i Ossamy Othmana.

Autorzy chcą podziękować wszystkim tym, dzięki którym możliwe stało się powstanie tego dokumentu.

A.2 Gwarancje

Ponieważ nie jestem ekspertem, nie zamierzam nawet sprawiać wrażenia, że wiem wszystko o Debianie, jak i Linuksie w ogólności. Moje rady odnośnie bezpieczeństwa mogą być w zasadzie dobre tylko do użytku domowego.

Ten dokument nie ma na celu zastąpienia żadnego innego przewodnika lub innej dokumentacji.

Autorzy tego dokumentu nie biorą jakiejkolwiek odpowiedzialności za powstałe szkody. Wszelkie znaki handlowe są własnością ich posiadaczy.

A.3 Pomoc

Komentarze i uzupełnienia do tego dokumentu są zawsze mile widziane. Jeśli masz jakieś sugestie odnośnie pakietu `debian-reference` lub jego odpowiedniego tłumaczenia, wyślij list na adres Debian BTS system (`http://bugs.debian.org/`). Możesz użyć także `reportbug`, który ułatwia wysyłanie dogłębnych zgłoszeń błędów. Oczywiście możesz również wysłać wiadomość w języku angielskim do Osamu Aoki (`http://people.debian.org/~osamu/`), na adres `<osamu\#at\#debian.org>`, lub do poszczególnych tłumaczy w odpowiednich językach.

Autor mówi: „Mimo, iż mieszkam w USA, język angielski nie jest dla mnie językiem ojczystym. Tak więc, jakiekolwiek poprawki gramatyczne są mile widziane."

Tłumacze na język Polan dodają: „Mimo, że mieszkamy w Polsce i język polski jest dla nas językiem ojczystym, to również, świadomi swojej niedoskonałości, przyjmiemy pomoc w postaci korekty uchybień językowych. Jeszcze bardziej zaś wdzięczni będziemy za korektę ewentualnych uchybień w procesie tłumaczenia.".

Najmilej widziane są łatki (wyniki działania programu diff) na wersję SGML, ale łatki na wersję tekstową również będą przyjmowane. Zajrzyj na stronę: 'Oficjalny dokument' na 1 stronie.

Oryginalne pliki SGML używane do stworzenia tego dokumentu dostępne są również przez CVS na: `:pserver:anonymous@cvs.sf.net/cvsroot/qref` lub `http://qref.sourceforge.net/Debian/qref.tar.gz`.

A.4 Format dokumentu

Dokument ten został napisany przy użyciu DebianDoc SGML DTD (przepisany z LinuxDoc SGML). System DebianDoc SGML pozwala na przetwarzanie plików pochodzących z jednego źródła na różne formaty, np. ten dokument może być przeglądany jako HTML, zwykły tekst, TeX DVI, PostScript, PDF oraz GNU info.

Narzędzia konwertujące DebianDoc SGML są dostępne w pakiecie `debiandoc-sgml`.

A.5 Labirynt Debiana

System Linux jest potężną platformą dla komputerów podłączonych do sieci. Jednakże opanowanie wszystkich jego możliwości wcale nie jest łatwe. Dobrym przykładem jest podłączenie i konfiguracja drukarki.

Istnieje kompletna i szczegółowa dokumentacja zwana „KODEM ŹRÓDŁOWYM”. Jest ona dokładna, lecz trudna do zrozumienia. Dostępne są również źródła informacji w postaci dokumentów HOWTO (HOWTO - Jak To Zrobić (JTZ)) i mini-HOWTO. Są prostsze, ale mają tendencję do opisu zbyt wielu detali, a co za tym idzie, zamglenia praktycznych informacji. Miewam czasem problemy ze znalezieniem pożądanej części długiego HOWTO, gdy muszę wywołać tylko kilka poleceń.

By móc odnaleźć się w gąszczu konfiguracji systemu, zacząłem zapisywać krótkie notatki w pliku tekstowym tak, by stanowiły dla mnie szybkie i proste źródło informacji. Plik ten z czasem zaczął rosnąć, a ja w międzyczasie nauczyłem się DebianDoc. Efektem tego jest właśnie *Debian Reference*.

A.6 Cytaty o Debianie

Tutaj znajduje się kilka interesujących cytatów z listy dyskusyjnej Debiana.

- „To jest Unix. Daje Ci wystarczająco dużo liny, abyś mógł się powiesić.” — Miquel van Smoorenburg `<miquels@cistron.nl>`
- „Unix **JEST** przyjazny dla użytkownika… Jest tylko wybredny w wyborze przyjaciół.” — Tollef Fog Heen `<tollef@add.no>`