

Debian Quick Reference

Osamu Aoki <osamu\#at\#debian.org>

Koordynator tłumaczenia: Bartosz Feński aka fEnIo <fenio@o2.pl>
'Autorzy' na **27** stronie

CVS, czwartek, 18 styczeń 2007, 11:54:36 UTC

Streszczenie

Debian Quick Reference (<http://qref.sourceforge.net/>) ma być krótkim wprowadzeniem do systemu Debian, tzw. **quick reference**. Jest to wyciąg z Debian Reference (<http://qref.sourceforge.net/>).

Prawa autorskie

Copyright © 2001–2005 by Osamu Aoki <osamu#@#debian.org>.

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

1	Wstęp	1
1.1	Konwencje w dokumencie	1
1.2	Podstawowe informacje o dystrybucjach Debiana	1
2	Uaktualnianie dystrybucji do wersji testowej	3
2.1	Przejsie do APT z Woodiego	3
2.2	Przygotowanie do przejścia (z wersji stabilnej na testową)	3
2.3	Aktualizacja systemu	4
2.3.1	Najlepiej do aktualizacji jest użyć programu <code>dselect</code>	5
3	Zarządzanie pakietami Debiana	7
3.1	Wprowadzenie	7
3.1.1	Podstawowe narzędzia	7
3.1.2	Wygodne narzędzia	8
3.2	Podstawy zarządzania pakietami Debiana	8
3.2.1	Instalowanie <i>zadań</i> programem <code>tasksel</code> lub <code>aptitude</code>	8
3.2.2	Konfiguracja systemu APT	9
3.2.3	<code>dselect</code>	9
3.2.4	<code>aptitude</code>	10
3.2.5	Polecenia <code>apt-cache</code> i <code>apt-get</code>	10
3.2.6	Podążanie za smakiem dystrybucji Debian	12
3.2.7	Obniżenie wersji wszystkich pakietów do <code>stable</code>	12
3.2.8	Przegląd pliku <code>/etc/apt/preferences</code>	12
3.3	Polecenia zapewniające przetrwanie w Debianie	13

3.3.1	Sprawdzanie błędów w Debianie i poszukiwanie pomocy	14
3.3.2	Problemy przy aktualizacji APT	14
3.3.3	Ratowanie przy użyciu <code>dpkg</code>	15
3.3.4	Przywracanie danych wyboru pakietów	16
3.3.5	Przywracanie systemu po awarii <code>/var</code>	16
3.3.6	Instalowanie pakietu w niestartującym systemie	16
3.3.7	Co robić, gdy polecenie <code>dpkg</code> nie działa?	17
3.4	Polecenia prowadzące do nirwany w Debianie	17
3.4.1	Informacje o pliku	17
3.4.2	Informacje o pakiecie	18
3.4.3	Niepilnowana instalacja przy użyciu APT	19
3.4.4	Ponowna konfiguracja zainstalowanych pakietów	19
3.4.5	Usuwanie i czyszczenie pakietów	20
3.4.6	Wstrzymywanie starszych pakietów	20
3.4.7	Mieszany system - <code>stable/testing/unstable</code>	20
3.4.8	Wyczyść buforowane pliki pakietów	21
3.4.9	Zapisanie/kopiowanie konfiguracji systemu	21
3.4.10	Przeniesienie pakietu do systemu <code>stabilnego</code>	21
3.4.11	Lokalne archiwum pakietów	22
3.4.12	Zmiana lub instalacja obcego pakietu binarnego	23
3.4.13	Sprawdzanie poprawności zainstalowanych plików	23
3.5	Inne osobliwości Debiana	24
3.5.1	Polecenie <code>dpkg-divert</code>	24
3.5.2	Pakiet <code>equivs</code>	24
3.5.3	Alternatywne polecenia	24
3.5.4	System-V <code>init</code> i poziomy startu (<code>runlevels</code>)	25
3.5.5	Wyłączone usługi demonów	26
A	Dodatek	27
A.1	Autorzy	27
A.2	Gwarancje	29
A.3	Pomoc	29

Rozdział 1

Wstęp

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

1.1 Konwencje w dokumencie

Dokument o nazwie: „Debian Quick Reference” zawiera informacje przedstawione w postaci krótkich poleceń powłoki systemowej Bash.

Odsyłam również do:

- **Podręcznika systemowego:** `bash(1)`.
- strony **GNU TEXINFO:** `info libc`.

1.2 Podstawowe informacje o dystrybucjach Debiana

Debian występuje w 3 „smakach”:

- **stabilny** (`stable`): Dobry do używania na serwerze produkcyjnym. Nudny w przypadku stacji roboczej (SR).
- **testowy** (`testing`): Niezłe wyjście na stację roboczą.
- **niestabilny** (`unstable`): Nigdy nie wykorzystuj go „na ślepo”.

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.

Są dwa sposoby na używanie najnowszych wersji oprogramowania.

- 'Podstawy zarządzania pakietami Debiana' na 8 stronie (głównie na stacje robocze)
- 'Przeniesienie pakietu do systemu stabilnego' na 21 stronie (głównie na serwery)

Rozdział 2

Uaktualnianie dystrybucji do wersji testowej

2.1 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
```

2.2 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 8 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.

2.3 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 7 stronie, by zapoznać się z podstawami, oraz ‘Problemy przy aktualizacji APT’ na 14 stronie, jeśli będziesz miał problemy.

2.3.1 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 konfigurator pakietów podczas tego procesu, więc przygotuj swoje notatki i poświęć tej części trochę czasu. Zobacz też `'dselect'` na [9](#) stronie.

Używaj `dselect`. **To zawsze działa** :). Jeśli musisz aktualizować bez użycia `dselect`, rozważ skorzytanie z `aptitude` oraz z innych programów.

Rozdział 3

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 ściągniętych 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 `chroot` 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`).

3.1 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` :-).

3.1.1 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**.

3.1.2 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
...
```

3.2 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.

3.2.1 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*.

3.2.2 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 'Przejdźcie do APT z Woodiego' na 3 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
```

3.2.3 `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 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):

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

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.

3.2.4 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
?	Ściągnawka 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.

3.2.5 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.

3.2.6 Podążanie za smakiem dystrybucji Debian

W zależności od wyboru smaku dystrybucji, możesz zmienić przykładowy plik `/etc/apt/preferences` z 'Konfiguracja systemu APT' na 9 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.

3.2.7 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.

3.2.8 Przegląd pliku `/etc/apt/preferences`

W przykładowym pliku `/etc/apt/preferences` przedstawionym w 'Konfiguracja systemu APT' na 9 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 linii 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`.

3.3 Polecenia zapewniające przetrwanie w Debianie

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

3.3.1 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
```

3.3.2 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 4 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 przyrzeć się skryptom `/var/lib/dpkg/info/nazwapakietu.{post-,pre-}{install,removal}` dla danego pakietu, a następnie uruchomić:

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

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`.

3.3.3 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`.

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
```

3.3.4 Przywracanie danych wyboru pakietów

Jeśli z jakiegokolwiek 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
```

3.3.5 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 Debiana 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 poniższe 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 bieżącej stronie. ([FIXME]: Ta procedura wymaga więcej testów do potwierdzenia poprawności.)

3.3.6 Instalowanie pakietu w niestartującym systemie

Uruchom Linuksa używając płyty lub dyskietki ratunkowej Debiana lub alternatywnej partycji w przypadku tzw. multi-bootu. 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 linii 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).

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

Niedziałający `dpkg` może uniemożliwić instalację jakiegokolwiek pliku `.deb`. Poniższa procedura pomoże Ci naprawić tę 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`.

3.4 Polecenia prowadzące do nirwany w Debianie

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

3.4.1 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
```

3.4.2 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
$ apt-cache policy pakiet # informacje o priorytecie i dystrybucji paki
$ apt-cache show -a pakiet # pokazuje opis pakietu ze wszystkich dystryb
$ 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 lin
$ dpkg -L pakiet ... # pokazuje listę plików zainstalowanych przez
```

`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*
```

3.4.3 Niepilnowana instalacja przy użyciu APT

Żeby móc instalować pakiety bez potrzeby pilnowania przebiegu instalacji, dodaj poniższą 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.

3.4.4 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
...
```

3.4.5 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 ...
```

3.4.6 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$"
```

3.4.7 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
```

3.4.8 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
```

3.4.9 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` znajdują 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
```

3.4.10 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 linii.

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 pakietów 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 `root`.

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
```

3.4.11 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 -l 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`.

3.4.12 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.

3.4.13 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).

3.5 Inne osobliwości Debiana

3.5.1 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 .

```
# dpkg-divert [--add] filename # dodaj "`dywersję`"  
# dpkg-divert --remove filename # usuń "`dywersję`"
```

Zazwyczaj nie należy stosować `dpkg-divert`, gdy nie jest to absolutnie potrzebne.

3.5.2 Pakiet `equivs`

Jeśli własnoręcznie kompilujesz program, najlepiej jest uczynić go prawdziwym, lokalnym pakietem zdebianaizowanym (*.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.
```

3.5.3 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`.

`/bin/sh` jest bezpośrednim dowiązaniem do `/bin/bash` lub `/bin/dash`. Bezpieczniej jest użyć `/bin/bash`, by zachować kompatybilność ze skryptami skazonymi 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`.

3.5.4 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 administratorowi 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 linii 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`.

3.5.5 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 jakiegokolwiek wątpliwości (odnośnie Exima, DHCP, ...). Sprawdź również 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. Przekazywanie X przez SSH również jest wyłączone.

Dodatek A

Dodatek

A.1 Autorzy

Debian Quick 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)
 - Réналd 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 Ścierański
 - 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>

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 jakiegokolwiek 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.