

Uwagi do wydania Debian GNU/Linux 4.0 (“etch”), Alpha

Josip Rodin, Bob Hilliard, Adam Di Carlo, Anne Bezemer, Rob Bradford, Frans Pop
(obecny), Andreas Barth (obecny), Javier Fernández-Sanguino Peña (obecny), Steve
Langasek (obecny)

<debian-doc@lists.debian.org>

\$Id: release-notes.en.sgml,v 1.312 2007-08-16 22:24:38 jseidel Exp \$

Spis treści

1	Wprowadzenie	1
1.1	Zgłaszanie błędów do tego dokumentu	1
1.2	Przesyłanie raportów z aktualizacji	2
1.3	Źródła dla tego dokumentu	2
2	Co nowego w Debian GNU/Linux 4.0	3
2.1	Co nowego w dystrybucji?	4
2.1.1	Zarządzanie pakietami	5
2.1.2	debian-volatile jest teraz oficjalnym serwisem	5
2.2	Udoskonalenia systemu	6
2.3	Główne zmiany dotyczące jądra	7
2.3.1	Zmiany w pakietach jądra	7
2.3.2	Nowe narzędzia do generowania initrds	8
2.3.3	Dynamiczne zarządzanie plikami /dev i wykrywanie sprzętu	8
3	System instalacyjny	9
3.1	Co nowego w instalatorze?	9
3.1.1	Główne zmiany	10
3.1.2	Zautomatyzowana instalacja	11
3.2	Badanie popularności	12
4	Upgrade z poprzednich wydań	13
4.1	Przygotowanie aktualizacji	13
4.1.1	Zrób kopię wszelkich danych i informacji konfiguracyjnych	13
4.1.2	Poinformuj użytkowników zawczasu	14

4.1.3	Przygotowanie do odzyskiwania	14
4.1.4	Przygotuj bezpieczne środowisko dla aktualizacji	15
4.1.5	Brak wsparcia dla jądra 2.2	15
4.2	Sprawdzenie statusu systemu	15
4.2.1	Przejrzyj akcje przewidywane przez menadżera pakietów	16
4.2.2	Wyłączenie pinningu APT	16
4.2.3	Sprawdzenie statusu pakietów	16
4.2.4	Źródła nieoficjalne i backporty	17
4.3	Ręczne usunięcie znaczników pakietów	18
4.4	Przygotowanie źródeł dla APT	18
4.4.1	Dodanie do APT źródeł internetowych	18
4.4.2	Dodanie źródeł APT dla lokalnego mirrora	19
4.4.3	Dodanie źródeł APT z CD-ROM'u lub DVD	19
4.5	Aktualizacja pakietów	20
4.5.1	Nagranie sesji	20
4.5.2	Aktualizacja listy pakietów	21
4.5.3	Upewnij się, że jest dość miejsca na dysku	21
4.5.4	Wstępny upgrade systemu	22
4.5.5	Upgrade jądra	24
4.5.6	Upgrade reszty systemu	25
4.5.7	Pobranie podpisów pakietów	25
4.5.8	Możliwe problemy podczas upgrade'u	26
4.6	Aktualizacja jądra i jego pakietów	27
4.6.1	Instalacja metapakietu jądra	27
4.6.2	Aktualizacja wewnątrz serii 2.6	28
4.6.3	Upgrade z jądra 2.4	28
4.6.4	Przeorganizowanie nazw urządzeń	29
4.6.5	Problemy z bootowaniem	30
4.7	Zadania przed restartem komputera	30
4.7.1	Konwersja z devfs	30
4.7.2	Aktualizacja mdadm	30

4.8	Przygotowanie do nowej edycji	31
4.9	Deprecated packages	31
4.10	Przestarzałe pakiety	32
4.10.1	Ślepe pakiety	32
5	Kwestie, o których warto wiedzieć mając etch	35
5.1	Potencjalne problemy	35
5.1.1	Problemy z urządzeniami odnoszącymi się do udev	35
5.1.2	Niektóre aplikacje nie będą pracować z jądrem 2.4	35
5.1.3	Właściwe miejsca sieciowe nie mogą zostać osiągnięte przez TCP	36
5.1.4	Spowolnione update'y plików indeksowych APT	36
5.1.5	Inicjalizacja sieci asynchronicznej może mieć nieprzewidziany przebieg	36
5.1.6	Trudności z używaniem bezpiecznych sieci bezprzewodowych WPA	36
5.1.7	Problemy z plikami o nazwach zawierających znaki nie-ASCII	37
5.1.8	Sound stops working	37
5.2	Upgrade do jądra 2.6	37
5.2.1	Konfiguracja klawiatury	38
5.2.2	Konfiguracja myszy	38
5.2.3	Konfiguracja dźwięku	38
5.3	Przejście z XFree86 na X.Org	39
5.4	Brak wsparcia dla 8-bitowych wyświetlaczy w wielu aplikacjach	39
5.5	Upgrade z exim do exim4	40
5.6	Aktualizacja apache2	40
5.7	Upgrade Zope i Plone	41
5.8	Rozszerzanie symboli (wildcard expansion - globbing) w GNU tar	41
5.9	NIS i Network Manager	42
5.10	Przestarzałe niezabezpieczone konfiguracje php	42
5.11	Status bezpieczeństwa produktów Mozilla	42
5.12	Środowisko KDE	43
5.13	Zmiany w środowisku GNOME	43
5.14	Domyślny edytor	43
5.15	Message of the day	44
5.16	Not default support for unicode in emacs21*	44

6	Więcej informacji o Debian GNU/Linux	45
6.1	Co czytać dalej	45
6.2	Pomoc	45
6.2.1	Listy dyskusyjne	45
6.2.2	Internet Relay Chat (IRC)	46
6.3	Zgłaszanie błędów	46
6.4	Rozwijanie Debiana	46
A	Zarządzanie systemem sarge	49
A.1	Aktualizacja systemu sarge	49
A.2	Sprawdzenie listy źródeł	49

Rozdział 1

Wprowadzenie

Najważniejszym celem niniejszych Uwag do Wydania jest poinformowanie użytkowników o najważniejszych zmianach wprowadzonych w tym wydaniu dystrybucji Debian GNU/Linux, sposobie bezpiecznego upgrade'u z poprzedniego do obecnego wydania i wreszcie o znanych potencjalnych problemach, które można napotkać podczas upgrade'u lub użytkowania wydania etch.

Jest niemożliwym wymienić wszystkich znanych problemów, dlatego wybrane zostały te, które występują często lub są najbardziej dokuczliwe.

Ostatnia wersja tego dokumentu jest zawsze dostępna pod adresem <http://www.debian.org/releases/stable/releasenotes>. Jeśli wersja, którą czytasz, jest starsza niż miesiąc¹, lepiej jest ściągnąć najnowszą wersję.

Prosimy zwrócić uwagę, że wspieramy i dokumentujemy upgrade z poprzedniego wydania Debiana (w tym przypadku sarge). Jeśli konieczne jest upgrade ze starszych wydań, sugerujemy najpierw przeczytanie uwag do poprzednich wydań i dokonanie aktualizacji do sarge.

1.1 Zgłaszanie błędów do tego dokumentu

Przystąpiliśmy do testowania wszystkich kroków upgrade'u opisanych w tym dokumencie, jak również spróbowaliśmy antycypować w problemach, z którymi mogą zetknąć się użytkownicy.

Pomimo to, jeśli sądzisz, że znalazłeś jakiś błąd w tej dokumentacji (nieprawidłowa informacja lub jej brak), prosimy o zgłoszenie błędu w systemie śledzenia błędów (<http://bugs.debian.org/>) do pakietu `release-notes`.

¹Nr wersji można zobaczyć na pierwszej stronie dokumentu PDF lub w stopce dokumentu HTML

1.2 Przesyłanie raportów z aktualizacji

Chętnie czytamy wszelkie informacje pochodzące od użytkowników odnoszące się do aktualizacji z sarge na etch. Jeśli chcesz podzielić się nimi, prosimy zgłosić błąd w systemie śledzenia błędów (<http://bugs.debian.org/>) do pakietu `upgrade-reports` z Twoimi wynikami. Prosimy kompresować załączniki (przy użyciu `gzip`).

W raporcie z `upgrade`'u prosimy dołączyć następujące informacje:

- Status bazy pakietów przed i po aktualizacji: dla `dpkg` dostępny jest w pliku `/var/lib/dpkg/status`, a dla `aptitude` w pliku `/var/lib/aptitude/pkgstates`. Powinieneś dokonać backupu przed aktualizacją, jak opisano w 'Zrób kopię wszelkich danych i informacji konfiguracyjnych' na stronie 13, możesz też znaleźć te informacje w `/var/backups`.
- Logi z sesji sporządzone za pomocą `script`, jak to opisano w 'Nagranie sesji' na stronie 20.
- Logi programu `aptitude` są w pliku `/var/log/aptitude`.

Uwaga: powinieneś zwrócić uwagę na niektóre informacje zawarte w logach, których nie chciałbyś ujawnić i usunąć je, zanim dołączysz je do raportu, jako że zostaną one opublikowane w dostępnej dla publicznie bazie danych.

1.3 Źródła dla tego dokumentu

Ten dokument używa pakietu `debiandoc-sgml`. Jego źródła są dostępne w repozytorium CVS dla *Debian Documentation Project*. Możesz użyć przeglądarki `www` (<http://cvs.debian.org/ddp/manuals.sgml/release-notes/?root=debian-doc>), aby mieć dostęp do każdego z plików i obserwować ich zmiany. Więcej informacji o dostępie do CVS jest dostępne na stronach Debian Documentation Project CVS (<http://www.debian.org/doc/cvs>).

Rozdział 2

Co nowego w Debian GNU/Linux 4.0

W tym wydaniu dodano oficjalne wsparcie dla architektury AMD64, obsługującej 64-bitowe procesory, zarówno Intela (EM64T), jak AMD (AMD64). W poprzednim wydaniu Debian GNU/Linux 3.1 ('sarge'), nieoficjalna wersja tego portu również była dostępna.

Oficjalne wsparcie dla architektury Motorola 680x0 ('m68k') nie będzie kontynuowane, gdyż dla tej architektury nie są spełnione kryteria określone przez Zarządzających Projektem. Najważniejszym powodem tej decyzji jest niedostateczna wydajność i ograniczona obsługa najważniejszych komponentów systemu. Tym niemniej port m68k pozostaje aktywny i dostępny do instalacji, chociaż nie jako część oficjalnego wydania stabilnego.

Oficjalnie wspierane architektury dla Debian GNU/Linux etch:

- Intel x86 ('i386')
- Alpha ('alpha')
- SPARC ('sparc')
- PowerPC ('powerpc')
- ARM ('arm')
- MIPS ('mips' (big-endian) i 'mipsel' (little-endian))
- Intel Itanium ('ia64')
- HP PA-RISC ('hppa')
- S/390 ('s390')
- AMD64 ('amd64')

Więcej nt. statusu portów i specyficznych dla nich informacji można znaleźć na stronach o portach Debiana (<http://www.debian.org/ports/alpha/>).

2.1 Co nowego w dystrybucji?

Nowe wydanie Debiana będzie znów zawierać o wiele więcej oprogramowania niż poprzednik sarge; dystrybucja obejmuje ponad 6500 nowych pakietów spośród wszystkich ponad 18200. Większość oprogramowania dystrybucji zostało zaktualizowane: ponad 10700 pakietów (czyli 68% wszystkich pakietów z sarge). Również znaczna liczba pakietów (ponad 3500, 23% pakietów z sarge) została, z różnych przyczyn, usunięta z tej dystrybucji. Nie będą dostępne dla nich aktualizacje i zostaną one oznaczone jako 'przestarzałe' w programach obsługi pakietów.

W tym wydaniu Debian GNU/Linux zamieniono XFree86 na wersję 7.1 X.Org, która obsługuje znacznie więcej sprzętu i lepiej go wykrywa. Pozwoli to na użycie Compiza, który jest jednym z pierwszych kompozytowych systemów zarządzania oknami dla X Windows, pozwalającym na pełne wykorzystanie sprzętowej akceleracji OpenGL dla kart, które ją obsługują.

Debian GNU/Linux znów dostarcza wiele aplikacji i środowiska 'na biurko' (ang. desktop). Oprócz innych, zawiera środowiska desktop GNOME 2.14¹, KDE 3.5.5a i Xfce 4.4. Aplikacje biurowe również zostały zaktualizowane, włączając w to pakiety biurowe OpenOffice.org 2.0.4a i KOffice 1.6, a także GNUcash 2.0.5, GNUMERIC 1.6.3 oraz Abiword 2.4.6.

Aktualizacja obejmuje też inne aplikacje desktop, włączając w to aktualizację Evolution 2.6.3 i Gaim 2.0. Mozilla suite została także zaktualizowana, ale główne programy zmieniły nazwy: iceweasel (wersja 2.0.0.2) jest 'niemarkową' przeglądarką www Firefox, a icedove (wersja 1.5) jest 'niemarkowym' klientem pocztowym Thunderbird.

Oprócz wielu innych, to wydanie zawiera także następujące aktualizacje:

- biblioteka GNU C library, wersja 2.3.6
- GNU Compiler Collection 4.1, jako podstawowy kompilator
- interpretery języków: Python 2.4, PHP 5.2
- oprogramowanie serwerowe:
 - serwery pocztowe: Exim 4.63 (domyślny dla nowych instalacji), Postfix 2.3, Courier 0.53, Cyrus 2.2
 - serwery www: Apache 2.2, fnord 1.10
 - serwery baz danych: MySQL 5.0.32, PostgreSQL 8.1
 - serwer OpenSSH, wersja 4.3
 - serwery nazw (DNS): Bind 9.3, maradns 1.2
 - serwer katalogów (directory server): OpenLDAP 2.3

Oficjalna dystrybucja Debian GNU/Linux będzie wydawana na 19 do 23 płytach CD z binariami (zależnie od architektury) i podobną liczbą płyt ze źródłami. Dostępna jest też wersja DVD dystrybucji.

¹Z niektórymi modułami z GNOME 2.16

2.1.1 Zarządzanie pakietami

`aptitude` jest zalecanym programem do obsługi pakietów z konsoli. `aptitude` wspiera większość poleceń wykonywanych przez `apt-get` i sprawdzono, że jest lepszy w rozwiązywaniu zależności niż `apt-get`. Jeśli ciągle używasz `dselect`, zalecamy przejść na `aptitude` jako oficjalny program zarządzania pakietami.

Dla `etch` został zaimplementowany zaawansowany mechanizm rozwiązywania konfliktów pakietów w programie `aptitude`, który próbuje znaleźć najlepsze rozwiązanie konfliktów wykrytych ze względu na zmiany zależności pomiędzy pakietami.

Secure APT jest teraz dostępny w `etch`. Zwiększa on znacznie poziom bezpieczeństwa systemów Debian GNU/Linux przez zastosowanie silnego szyfrowania i podpisów elektronicznych zastosowanych do sprawdzenia ściągniętych z sieci pakietów. Wydanie to zawiera program `apt-key` do dodawania nowych kluczy do istniejącego zestawu, który normalnie zawiera tylko bieżące klucze archiwów Debiana, udostępniane w pakiecie `debian-archive-keyring`.

W swojej podstawowej konfiguracji `apt` będzie ostrzegał, jeśli pakiety są pobierane z nieautoryzowanych źródeł. Następne wydania mogą wymagać, aby wszystkie pakiety zostały zweryfikowane przed pobraniem. Administratorzy nieoficjalnych źródeł `apt` są zachęceni do generowania kluczy kryptograficznych i podpisywania nimi plików Wydania, jak też wdrażania bezpiecznych sposobów dystrybucji kluczy publicznych.

Więcej informacji znajduje się na stronach podręcznika `apt(8)`, Podpisywanie pakietów w Debianie (<http://www.debian.org/doc/manuals/securing-debian-howto/ch7#s-deb-pack-sign>) rozdział podręcznika *Securing Debian Manual* i Debian Wiki (<http://wiki.debian.org/SecureApt>).

Inną funkcją dodaną do `apt` jest możliwość ściągnięcia tylko zmian w plikach pakietów zaszytych od ostatniej aktualizacji. Więcej o tej funkcji w 'Spowolnione update'y plików indeksowych APT' na stronie 36.

2.1.2 `debian-volatile` jest teraz oficjalnym serwisem

Serwis *debian-volatile*, który był nieoficjalnym serwisem w wydaniu `sarge`, teraz staje się oficjalnym serwisem Debian GNU/Linux.

Oznacza to, że teraz używa on adresu `.debian.org`². Prosimy dokonać zmiany w pliku `/etc/apt/sources.list`, jeśli używałeś już tego serwisu.

debian-volatile pozwala użytkownikom łatwo aktualizować pakiety wersji stabilnej zawierające szybko dezaktualizujące się informacje. Przykładem mogą być sygnatury wirusów skanerów antywirusowych lub wzorce dla filtrów spamu. Więcej informacji i listę mirrorów można znaleźć na tej stronie (<http://volatile.debian.org/>).

²Stary adres `volatile.debian.net` pozostaje przez pewien czas prawidłowym adresem.

2.2 Udoskonalenia systemu

Jest wiele zmian w dystrybucji, których korzyści są widoczne w nowych instalacjach etch, ale nie są przenoszone automatycznie podczas aktualizacji z sarge. Ta sekcja omawia najistotniejsze z nich.

Zmniejszony priorytet podstawowych pakietów deweloperskich Pakiety deweloperskie, które dotąd miały priorytet *standard*, mają teraz priorytet *optional*, co oznacza, że nie zostaną zainstalowane domyślnie. Dotyczy w to m. in. standardowego kompilatora C/C++, `gcc`, jak też innych programów (`dpkg-dev`, `flex`, `make`), a także nagłówków deweloperskich (`libc6-dev`, `linux-kernel-headers`).

Jeśli chcesz mieć te pakiety w systemie, najprościej zainstalować je przez instalację pakietu `build-essential`, który zawiera większość z nich.

SELinux ma priorytet standard, ale nie jest włączony Pakiety niezbędne dla SELinux (ang. Security-enhanced Linux, Linux o rozszerzonym bezpieczeństwie) zostały podniesione do priorytetu *standard*. Oznacza to, że zostaną one zainstalowane domyślnie w nowych instalacjach. Dla systemów istniejących można zainstalować SELinux poprzez:

```
# aptitude install selinux-basics
```

Trzeba zauważyć, że SELinux *nie* jest domyślnie włączany. Informacje o konfiguracji i włączaniu SELinuxa można znaleźć na Debian Wiki (<http://wiki.debian.org/SELinux>).

Nowy superdemon inet Domyślnym superdemonem inet dla etch jest `openbsd-inetd` zamiast `netkit-inetd`. Nie wystartuje on, jeśli nie zostanie skonfigurowana żadna z usług, która z niego korzysta, domyślnie tak właśnie jest. Nowy demon zostanie zainstalowany automatycznie podczas aktualizacji.

Zmieniony domyślny klon vi Instalowanym wariantem edytora `vi` jest domyślnie wersja zmniejszona `vim` (pakiet `vim-tiny`) zamiast `nvi`.

Zmiany w domyślnych właściwościach ext2/ext3 Nowe filesystemy `ext2` i `ext3` będą zakładane z włączonymi opcjami `dir_index` i `resize_inode`. Pierwsza z nich przyśpiesza operacje na katalogach zawierających wiele plików, druga umożliwia zmianę rozmiaru partycji "w locie" (on-line - na zamontowanym filesystemie).

Użytkownicy dokonujący `upgrade'u` z `sarge` mogą rozważyć ręczne dodanie flagi `dir_index` za pomocą `tune2fs`³; flaga `resize_inode` nie może być dodana do istniejącego filesystemu. Można sprawdzić, które flagi są ustawione za pomocą `dumpe2fs -h`.

Domyślną stroną kodową w etch jest UTF-8 Domyślną stroną kodową dla nowych instalacji Debian GNU/Linux jest UTF-8. Wiele aplikacji ma ustawioną tą stronę kodową jako domyślną.

³Flaga `filetype` powinna już być ustawiona na większości filesystemów, za wyjątkiem instalowanych przed `sarge`.

Użytkownicy aktualizujący system do etch, którzy chcą zmienić stronę kodową na UTF-8, muszą zrekonfigurować środowisko i zmienne lokalizujące. Dla całego systemu można to zrobić za pomocą `dpkg-reconfigure locales`; najpierw należy wybrać stronę UTF-8 dla Twojego kraju i języka, a potem przełączyć ją jako domyślną. Należy dodać, że po przełączeniu na UTF-8, prawdopodobnie konieczna będzie konwersja istniejących plików z poprzedniej (dotychczasowej) strony kodowej na UTF-8.

Pakiet `utf8-migration-tool` zawiera narzędzie, które może pomóc w migracji, jednak pakiet ten jest dostępny jedynie w edycji niestabilnej, gdyż nie został wystarczająco przygotowany dla etch. Usilnie zalecamy wykonanie backup'u danych przed wykorzystaniem tego pakietu.

Niektóre aplikacje mogą jeszcze nie działać prawidłowo pod UTF-8, chodzi głównie o błędne wyświetlanie czcionek.

Pod adresem Debian Wiki (<http://wiki.debian.org/Sarge2EtchUpgrade>) podano więcej informacji nt. różnic pomiędzy sarge a etch.

2.3 Główne zmiany dotyczące jądra

Debian GNU/Linux 4.0 posiada jądro wersji 2.6.18 dla wszystkich architektur; to wydanie jest ciągle zasadniczo⁴ kompatybilne z jądrem 2.4, ale Debian już nie wspiera pakietów jądra 2.4.

Dokonano zasadniczych zmian, zarówno w samym jądrze, jak i pakietach współpracujących z jądrem Debiana. Niektóre z tych zmian komplikują procedurę upgrade'u i potencjalnie mogą prowadzić do problemów w czasie reboot'u systemu po aktualizacji do etch. Ta sekcja daje przegląd najistotniejszych zmian; potencjalne problemy i informacje o ich rozwiązywaniu są podane w kolejnych rozdziałach.

Jeśli używasz jądra 2.4, powinieneś przeczytać dokładnie 'Upgrade do jądra 2.6' na stronie 37.

2.3.1 Zmiany w pakietach jądra

Zmiana nazw pakietów jądra Wszystkie pakiety jądra zostały przemianowane z `kernel-*` na `linux-*` w celu uporządkowania przestrzeni nazw. Ułatwi to w przyszłości dodawanie do Debiana jąder nielinuksowych.

Tam, gdzie to jest możliwe, dla usuwanych pakietów zostały zastosowane puste pakiety przejściowe, które zależą od nowych pakietów.

⁴Niektóre pakiety mogą nie współpracować prawidłowo z jądrem 2.4; zobacz 'Niektóre aplikacje nie będą pracować z jądrem 2.4' na stronie 35.

2.3.2 Nowe narzędzia do generowania initrds

Pakiety jądra Debiana dla Alpha wymagają `initrd` do zbootowania systemu. Ze względu na zmiany w jądrze, narzędzie używane w `sarge` do generowania `initrd`, `initrd-tools`, nie jest już używane i stało się przestarzałe. Zostało zastąpione dwoma nowymi pakietami: `initramfs-tools` i `yaird`. Koncepcje wykorzystane do ich napisania są zupełnie inne; ich przegląd jest dostępny na Debian Wiki (<http://wiki.debian.org/InitrdReplacementOptions>). Obydwa generują `initrd` przy użyciu systemu plików `initramfs`, który jest skompresowanym archiwum `cpio`. Domyślnym i zalecanym narzędziem jest `initramfs-tools`.

Aktualizacja do jądra `etch` spowoduje, że zostanie zainstalowany pakiet `initramfs-tools`. Jeśli dokonujesz `upgrade'u` z jądra 2.4 do 2.6, musisz użyć pakietu `initramfs-tools`. Użycie `yaird` spowoduje, że instalacje `linux-image-2.6` nie będą działały, jeśli poprzednio miałeś jądro 2.2 lub 2.4.

Pakiet `initrd-tools` został zachowany w `etch` ze względu na przejście z `sarge`. W następnej edycji zostanie on usunięty.

2.3.3 Dynamiczne zarządzanie plikami `/dev` i wykrywanie sprzętu

Jądro `etch` nie ma już obsługi dla `devfs`.

`devfs` jest zastąpione przez `udev`, który obsługuje urządzenia użytkowników (ang. `userspace`).

`udev` jest montowany w katalogu `/dev` i propaguje jego urządzenia do obsługi przez jądro. Może też dynamicznie dodawać i usuwać urządzenia, analogicznie jak ładowalne moduły jądra, opierając się na zdarzeniach generowanych przez jądro. `udev` jest o wiele wszechstronniejszy, niż `devfs` i oferuje usługi, które mogą być użyte przez inne pakiety, np. `hal` (hardware abstraction layer - warstwa abstrakcji sprzętowej).

Współpracując z jądrem `udev` zajmuje się również wykrywaniem sprzętu i ładowaniem modułów dla wykrytych urządzeń. Z tego względu jest w konflikcie z pakietem `hotplug`. Stosowany w `sarge` pakiet `discover` mógłby być również użyty do ładowania modułów w czasie procesu bootowania, ale w nowej wersji zastosowanej w `etch` nie wykonuje on tej funkcji. Pakiet `discover` jest ciągle używany przez X.Org do wykrywania, jaka karta graficzna jest użyta w systemie.

Jeżeli instalujesz obraz jądra Debiana, `udev` zostanie zainstalowane domyślnie, ponieważ `initramfs-tools` zależy od niego.

Możesz zapobiec instalowaniu `udev` przez skompilowanie własnego niemodularnego jądra lub przez użycie alternatywnego generatora `initrd`, np. `yaird`. Jednakże `initramfs-tools` jest rekomendowanym generatorem `initrd`.

Rozdział 3

System instalacyjny

Instalator Debiana jest oficjalnym systemem instalacyjnym dla Debiana. Oferuje on różne metody instalacji. To, które z nich są dostępne dla Twojego systemu, zależy od jego architektury.

Obrazy płyt instalatora etch można pobrać wraz z Podręcznikiem Instalacji ze strony Debiana (<http://www.debian.org/releases/stable/debian-installer/>).

Installation Guide jest również dodany do pierwszej płyty oficjalnego zestawu CD/DVD na:

```
/doc/install/manual/language/index.html
```

Można także sprawdzić listę znanych problemów, czyli erratę (<http://www.debian.org/releases/stable/debian-installer/index#errata>) dla instalatora Debiana (debian-installer).

Instalator może być użyty do zainstalowania tylko na tych systemach alpha, które wspierają konsolę SRM. Przed rozpoczęciem instalacji upewnij się, że Twój system jest przełączony na SRM. Jeśli maszyna wspiera jedynie konsolę AlphaBIOS/ARC, zalecanym sposobem instalacji etch jest zainstalowanie (minimalnej) edycji woody, potem upgrade do sarge i wreszcie do etch. Więcej informacji nt. różnych konsoli na stronie Debian alpha port (<http://www.debian.org/ports/alpha>).

3.1 Co nowego w instalatorze?

Włożono wiele pracy w Instalatora Debiana od czasu pierwszej oficjalnej edycji sarge, co owocowało zarówno poprawieniem obsługi sprzętu, jak też nowymi interesującymi rozwiązaniami.

W niniejszych Uwagach tylko wymieniamy najważniejsze zmiany dokonane w instalatorze. Jeśli jesteś zainteresowany szczegółami wprowadzonymi od czasu sarge, prosimy sprawdzić informacje ogłaszane dla etch, wydań beta i RC, dostępne na stronie historii wiadomości (<http://www.debian.org/devel/debian-installer/News/>) Instalatora Debiana.

3.1.1 Główne zmiany

Brak restartu komputera podczas instalacji Dotychczas proces instalacji podzielony był na dwie fazy: konfiguracja systemu bazowego wraz z konfiguracją bootowania, po czym następowało przeładowanie i wykonanie programu `base-config`, który zajmował się m. in. konfiguracją użytkowników, konfiguracją zarządcy pakietów i instalacją pozostałych pakietów (za pomocą `tasksel`'a).

W `etch` drugi etap został zintegrowany z Instalatorem Debiana. Daje to wiele korzyści w postaci zwiększenia bezpieczeństwa, a także ze względu na fakt, że po restarcie, na końcu instalacji system powinien mieć już prawidłowo ustawioną strefę czasową i, jeśli instalowałeś środowisko graficzne, powinien od razu zacząć pracować z wykorzystaniem interfejsu graficznego.

Domyślne kodowanie w UTF-8 Instalator ustanawia jako domyślne kodowanie UTF-8 zamiast dotychczasowego, zależnego od języka (czyli np. ISO-8859-1, EUC-JP czy KOI-8).

Elastyczniejsze partycjonowanie Obecnie możliwe jest skonfigurowanie systemów plików (filesystem) opartego na LVM (ang. Logical Volume Management - Logiczne Zarządzanie Wolumenami) używając przewodnika partycjonowania.

Instalator pozwala też na zdefiniowanie zaszyfrowanych filesystemów. Ręczne partycjonowanie daje wybór pomiędzy `dm-crypt` a `loop-aes`, przy użyciu hasła lub klucza losowego oraz pozwala na użycie wielu innych opcji. Przewodnik partycjonowania (ang. guided partitioning) tworzy zaszyfrowaną partycję LVM, która zawiera dowolne inne filesystemy (poza `/boot`) jako wolumeny logiczne.

Tryb awaryjny Możesz użyć instalatora do rozwiązania problemów z systemem, np. z nieoczekiwanym restartem. Pierwsze kroki będą wtedy identyczne jak przy normalnej instalacji, ale instalator nie rozpocznie partycjonowania. Oferujemy wtedy specjalne menu opcji awaryjnych.

Aktywacja trybu awaryjnego nastąpi przy uruchomieniu instalatora jako `rescue` lub przez dodanie parametru restartu `rescue/enable=true`.

Użycie `sudo` zamiast konta `root` W czasie instalacji zaawansowanych (`expert`) można nie ustawiać konta `root` (zostanie ono zablokowane), a zamiast tego wybrać `sudo`, za pomocą którego pierwszy użytkownik może administrować systemem.

Kryptograficzna weryfikacja ściąganych pakietów Pakiety ściągane przez instalator są obecnie sprawdzane kryptograficznie przy użyciu `apt`, co zmniejsza prawdopodobieństwo wystąpienia błędów w czasie instalacji przez sieć.

Uproszczona konfiguracja poczty Jeśli zostanie zainstalowany "system standardowy", instalator wykona jedynie podstawową konfigurację serwera pocztowego, tylko dla lokalnego dostarczania poczty. Serwer pocztowy nie będzie dostępny dla innych systemów dołączonych do tej samej sieci. Jeśli chcesz skonfigurować system, tak aby obsługiwał również nielokalną pocztę (wychodzącą bądź przychodzącą), będziesz musiał zrekonfigurować serwer pocztowy po instalacji.

Wybór środowiska graficznego (desktop) System instalacyjny zainstaluje środowisko GNOME jako domyślne środowisko graficzne, jeśli użytkownik zleci instalację środowiska graficznego.

Jednak użytkownicy, którzy chcą zainstalować środowiska alternatywne, mogą to łatwo uczynić dodając parametr bootowania: `tasks="standard, kde-desktop"` dla KDE lub `tasks="standard, xfce-desktop"` dla Xfce. Trzeba jednak zwrócić uwagę, że to nie zadziała, jeśli instalacja jest przeprowadzana z pełnej płyty CD bez użycia serwera lustrzanego jako dodatkowego źródła pakietów; za to zadziała, jeśli użyjemy płyty DVD lub jakiegokolwiek innej metody instalacji.

Są też przygotowane osobne płyty CD, które umożliwiają instalację KDE lub Xfce jako środowisko domyślne.

Nowe języki Dzięki znacznemu wysiłkowi tłumaczy Debian może być teraz instalowany w 47 językach przy użyciu instalatora tekstowego. Jest to 6 języków więcej, niż w sarge. Języki dodane w tym wydaniu to: białoruski, esperanto, estoński, kurdyjski, macedoński, tagalog, wietnamski i wolof. Ze względu na braki w uzupełnieniu tłumaczeń, zostały usunięte perski i walijski.

Użytkownicy, którzy nie chcą używać żadnej lokalizacji mogą teraz wybrać C jako preferowaną lokalizację w czasie wyboru języka instalatora. Więcej informacji o obsłudze języków jest dostępnych na stronie d-i languages list (<http://d-i.alioth.debian.org/i18n-doc/languages.html>).

Uproszczona lokalizacja i wybór stref czasowych Konfiguracja języków, krajów i stref czasowych została uproszczona, tak aby ograniczyć ilość informacji podawanych przez użytkownika. Instalator domyśla się kraju i strefy czasowej na podstawie wybranego języka lub, jeśli nie zdoła, prosi o podanie niezbędnego minimum. Użytkownicy mogą też wprowadzać inne kombinacje, jeśli jest to potrzebne.

Lokalizacja rozszerzona dla całego systemu Większość zadań internacjonalizacji i lokalizacji, które były przedtem wykonywane przez narzędzie `localization-config`, zostało włączonych do głównego trzonu instalatora Debiana lub do samych pakietów. Oznacza to, że wybór języka automatycznie określa instalację pakietów niezbędnych dla danego języka (słowniki, dokumentacja, czcionki...), zarówno w środowisku znakowym, jak i graficznym. Konfiguracja, która nie będzie już wykonywana automatycznie, obejmuje definiowanie rozmiaru papieru i niektóre zaawansowane ustawienia klawiatury dla X Windows w niektórych językach.

Należy zwrócić uwagę, że pakiety przygotowane językowo (np. spolonizowane [uwaga tłum.]), będą instalowane automatycznie tylko w przypadku, gdy są one dostępne w czasie instalacji.

3.1.2 Zautomatyzowana instalacja

Wiele zmian opisanych w poprzednich punktach pociąga za sobą zmiany instalacji poprzez jej zautomatyzowanie za pomocą plików prekonfiguracyjnych. Oznacza to, że pliki prekonfiguracyjne pochodzące z sarge nie będą współpracować z nowym instalatorem bez modyfikacji.

Dobłą wiadomością jest, że Installation Guide (<http://www.debian.org/releases/stable/installmanual>) ma teraz osobny dodatek z obszerną dokumentacją dotyczącą prekonfiguracji.

Instalator etch wprowadza kilka ciekawych rozwiązań ułatwiających automatyzację instalacji. Dodane jest wsparcie dla zaawansowanego partycjonowania RAID, LVM oraz szyfrowanego LVM. Szczegóły w dokumentacji.

3.2 Badanie popularności

System instalacyjny wprowadza z powrotem pakiet `popularity-contest` (badanie popularności). Nie był on instalowany domyślnie w `sarge`, chociaż występował we wcześniejszych wydaniach.

`popularity-contest` pozwala projektowi Debiana na otrzymanie wielu cennych informacji o aktualnie używanych pakietach dystrybucji. Informacje te są używane głównie do podejmowania decyzji o kolejności umieszczania pakietów na płytach instalacyjnych, jak też pomocniczo dla deweloperów, czy i w jaki sposób rozwijać dany pakiet.

Informacje z pakietu `popularity-contest` są zbierane i przetwarzane anonimowo. Chcemy wyrazić naszą wdzięczność, jeśli zgadzasz się brać udział w tym badaniu, które może ulepszać Debiana.

Rozdział 4

Upgrade z poprzednich wydań

4.1 Przygotowanie aktualizacji

Zalecamy przed aktualizacją przeczytać także informację 'Kwestie, o których warto wiedzieć mając etch' na stronie [35](#). Ten rozdział opisuje potencjalne problemy niebezpośrednio związane z procesem upgrade'u. Wiedza o nich jest jednak ważna przed rozpoczęciem tego procesu.

4.1.1 Zrób kopię wszelkich danych i informacji konfiguracyjnych

Przed aktualizacją systemu zdecydowanie zalecamy zrobienie pełnej kopii zapasowej (backup) albo przynajmniej najważniejszych danych i zbiorów konfiguracyjnych, które nie powinny zostać utracone. Narzędzia aktualizujące są zasadniczo niezawodne, ale wszelkie awarie sprzętowe w czasie aktualizacji mogą spowodować poważne uszkodzenie systemu.

The main things you'll want to back up are the contents of `/etc`, `/var/lib/dpkg`, `/var/lib/aptitude/pkgstates` and the output of `dpkg --get-selections "*" (the quotes are important)`.

Proces aktualizacji nie wprowadza modyfikacji w katalogu `/home`. Jednak niektóre aplikacje (np. programy Mozilla suite, środowiska GNOME czy KDE) nadpisują ustawienia użytkowników nowymi wartościami w momencie pierwszego użycia przez użytkownika. Profilaktycznie można zrobić kopie plików i katalogów ukrytych ("dotfiles") z katalogów domowych użytkowników. Pozwoli to na ewentualne odzyskanie starych ustawień. Można też poinformować o tym użytkowników.

Operacje instalacji pakietów muszą być wykonywane z uprawnieniami superużytkownika, czyli że musisz zalogować się jako root lub też użyć `su` lub `sudo`, aby otrzymać odpowiednie uprawnienia.

Aktualizacja ma kilka warunków wstępnych, powinieneś sprawdzić je przed wykonywaniem upgrade'u.

4.1.2 Poinformuj użytkowników zawczasu

Warto też wcześniej poinformować wszystkich użytkowników o rozpoczęciu planowanego upgrade'u, chociaż użytkownicy dostający się do systemu poprzez `ssh` powinni zobaczyć w czasie upgrade'u tylko krótką uwagę na ten temat i móc kontynuować pracę.

Jeśli chcesz przedsięwziąć dodatkowe środki ostrożności, zrób backup lub odmontuj partycje użytkowników (`/home`) przed upgrade'm.

Prawdopodobnie będziesz też musiał dokonać aktualizacji jądra podczas upgrade'u do `etch`, tak więc konieczny będzie restart maszyny. Zwykle może być to zrobione po zakończeniu aktualizacji.

4.1.3 Przygotowanie do odzyskiwania

Ze względu na mnogość zmian w jądrze pomiędzy `sarge` a `etch` dotyczących sterowników, wykrywania sprzętu, nazewnictwa i kolejności plików urządzeń, istnieje realne niebezpieczeństwo, że będziesz miał problemy z `reboot`'em systemu po dokonaniu aktualizacji. Wiele z potencjalnych problemów jest opisanych w tym i następnym rozdziałach niniejszych Uwag.

Z tego powodu zalecamy podjąć działania mające na celu umożliwienie przywrócenie stanu sprzed `reboot`'u zakończonego niepowodzeniem, a dla systemów zdalnych sprzed utraty połączenia sieciowego.

Jeśli wykonujesz upgrade zdalnie przez `ssh`, gorąco zalecamy zapewnić sobie dostęp do serwera również przez zdalny terminal szeregowy. Istnieje prawdopodobieństwo, że po aktualizacji jądra i reboocie, niektóre urządzenia zmieniają swoje nazwy (jak opisano w 'Przeorganizowanie nazw urządzeń' na stronie 29) i będziesz musiał poprawić konfigurację systemu poprzez lokalną konsolę. Również w przypadku, kiedy system zostanie zrestartowany w środku aktualizacji, musisz mieć szansę powrotu (`recovery`) poprzez lokalną konsolę.

Najbardziej oczywistą rzeczą, jako pierwsze podejście, jest próba `reboot`'u ze starym jądrem. Jednak, z wielu powodów opisanych w tym dokumencie, nie ma gwarancji sukcesu tej próby.

Jeśli to zawiedzie, musisz mieć jakąś alternatywę startu systemu, dostępu do niego i możliwości naprawy. Pierwszą możliwością jest użycie płyty naprawczej (`rescue`) lub płyty Linux Live. Po starcie z takiej płyty powinno być możliwe zamontowanie filesystemu `root` i przejście na niego za pomocą `chroot` w celu zdiagnozowania i naprawienia problemu.

Drugim sposobem, który chcemy zarekomendować, jest użycie *rescue mode* (trybu ratunkowego) instalatora `etch`. Zaletą użycia instalatora jest możliwość wyboru pomiędzy wieloma metodami instalacji w sposób najoptymalniejszy w Twojej sytuacji. Więcej informacji znajduje się w sekcji "Recovering a Broken System" rozdziału 8. Installation Guide (<http://www.debian.org/releases/stable/installmanual>) i w Debian Installer FAQ (<http://wiki.debian.org/DebianInstaller/FAQ>).

Powłoka debugująca w czasie bootowania za pomocą initrd

Pakiet `initramfs-tools` zawiera powłokę debugującą¹ w `initrd`. Jeśli np. `initrd` nie może zamontować filesystemu root, zostanie uruchomiona powłoka, z której możesz wywołać proste polecenia, pomagające wykryć problem i, być może, naprawić go.

Podstawowe rzeczy do sprawdzenia to: obecność prawidłowych plików w katalogu `/dev`; załadowane moduły (`cat /proc/modules`); wyjście programu `dmesg` z błędami ładowania sterowników. `dmesg` pokazuje też, które pliki urządzeń są dowiązane do których dysków; powinieneś porównać to z wyjściem polecenia `echo $ROOT`, upewniając się, że filesystem root jest na odpowiednim urządzeniu.

Jeśli naprawiłeś problem, możesz napisać `exit` opuszczając powłokę debugowania i kontynuując proces bootowania od miejsca, w którym został przerwany. Oczywiście, dla rozwiązania problemu może być też konieczne odbudowanie `initrd`, tak że dopiero następny boot przebiegnie bez błędu.

4.1.4 Przygotuj bezpieczne środowisko dla aktualizacji

Aktualizacja dystrybucji powinna być wykonywana albo lokalnie przez wirtualną konsolę tekstową (ewentualnie bezpośrednio przyłączony terminal szeregowy), albo zdalnie poprzez `ssh`.

W celu zwiększenia marginesu bezpieczeństwa podczas upgrade'u zdalnego, sugerujemy wykonywać ten proces w wirtualnej konsoli udostępnianej przez program `screen`, który posiada możliwość bezpiecznego przełączania i zapewnia nieprzerywalny proces aktualizacji, nawet gdyby zawiodły procesy zdalnego połączenia.

Ważna uwaga: *Nie* powinieneś dokonywać upgrade'u za pomocą programów: `telnet`, `rlogin`, `rsh`, ani sesji X zarządzanych przez `xdm`, `gdm` czy `kdm` na maszynie, którą aktualizujesz. Jest tak dlatego, że każdy z tych serwisów może zostać zatrzymany w czasie upgrade'u, co może spowodować, że system stanie się *niedostępny* w połowie aktualizacji.

4.1.5 Brak wsparcia dla jądra 2.2

Jeśli używasz jądra o wersji wcześniejszej niż 2.4.1, powinieneś je zaktualizować do (co najmniej) serii 2.4 przed upgrade'm `glibc`. Powinno to być zrobione przed rozpoczęciem aktualizacji systemu. Zalecamy bezpośrednią aktualizację do jądra wersji 2.6.8 dostępnego w `sarge`, a nie do jądra 2.4.

4.2 Sprawdzenie statusu systemu

The upgrade process described in this chapter has been designed for upgrades from "pure" `sarge` systems without third-party packages. In particular, there are known problems with

¹Może być ona wyłączona przez dodanie parametru `panic=0` do parametrów bootowania.

third-party packages which install programs under `/usr/X11R6/bin/` causing problems with upgrades due to the X.Org transition ('Przejsie z XFree86 na X.Org' na stronie 39). For greatest reliability of the upgrade process, you may wish to remove third-party packages from your system before you begin upgrading.

Zakładamy również, że aktualizujemy system z najnowszego sarge. Jeśli tak nie jest, lub nie jesteś pewien, postępuj według instrukcji w 'Aktualizacja systemu sarge' na stronie 49.

4.2.1 Przejrzyj akcje przewidywane przez menadżera pakietów

W niektórych przypadkach, jeśli używano `apt-get` do instalowania pakietów zamiast `aptitude`, może się zdarzyć, że `aptitude` rozpozna pakiet jako "nieużywany" i zaznaczy go do usunięcia. Ogólnie biorąc, powinieneś upewnić się, że system jest w pełni aktualny i "czysty", zanim rozpoczniesz aktualizację.

Z tego powodu powinieneś przejrzeć, czy `aptitude` nie wykona jakichś niespodziewanych akcji. Jeśli pakiet jest zaznaczony do usunięcia lub `update'u` w zarządcy pakietów, może mieć to negatywny wpływ na przebieg `upgrade'u`. Naprawienie tego jest możliwe tylko, jeśli plik `sources.list` wskazuje na *sarge*, a nie na *stable* lub *etch*; zobacz 'Sprawdzenie listy źródeł' na stronie 49.

Aby to uczynić, musisz uruchomić `aptitude` i nacisnąć klawisz 'g' ("Go"). Jeśli pokażą się jakieś akcje, powinieneś przejrzeć je i albo poprawić, albo przeprowadzić zgodnie z wyświetlanymi sugestiami. Jeśli nie ma żadnych sugerowanych akcji, zostanie wyświetlony komunikat "Brak pakietów do instalacji, usunięcia lub aktualizacji".

4.2.2 Wyłączenie pinningu APT

Jeśli APT został skonfigurowany do instalowania pakietów z dystrybucji innej niż stabilna (np. `testing`), powinieneś zmienić ustawienia pinningu APT (przechowywanego w `/etc/apt/preferences`), aby umożliwić aktualizację pakietów z nowej wersji stabilnej. Dalsze informacje nt. pinningu APT można znaleźć w `apt_preferences(5)`.

4.2.3 Sprawdzenie statusu pakietów

Niezależnie od metody użytej przy aktualizacji, zalecamy sprawdzić najpierw status wszystkich pakietów, weryfikując, że wszystkie są w stanie umożliwiającym `upgrade`. Następujące polecenie pokaże pakiety, które są połowicznie zainstalowane (`Half-Installed`) lub błędnie skonfigurowane (`Failed-Config`) i te, które mają dowolny status wskazujący na błąd.

```
# dpkg --audit
```

Można także sprawdzić stan wszystkich pakietów w systemie używając `dselect`, `aptitude`, lub poleceń takich jak

```
# dpkg -l | pager
```

lub

```
# dpkg --get-selections "*" > ~/curr-pkgs.txt
```

Przed upgrde’*m* warto usunąć pakiety o statusie ‘hold’ (utrzymane, aktualizacja niedozwolona). Jeśli jakiś ważny z punktu widzenia upgrde’*u* pakiet ma status ‘hold’, cała aktualizacja nie powiedzie się.

Zwracamy uwagę, że `aptitude` używa innej metody rejestracji stanu ‘hold’ pakietów niż `apt-get` i `dselect`. Możesz zidentyfikować stan ‘hold’ pakietów dla `aptitude` poprzez

```
# aptitude search "~ahold" | grep "^h"
```

Jeśli chcesz sprawdzić pakiety w stanie ‘hold’ dla `apt-get`, powinieneś użyć

```
# dpkg --get-selections | grep hold
```

Jeśli lokalnie zmieniłeś i zrekompiłowałeś jakiś pakiet bez zmiany nazwy albo przez wstawienie wersji ‘epoch’, musisz nadać mu status ‘hold’, aby zapobiec jego aktualizacji.

Status “hold” dla pakietu `aptitude` może być zmieniony za pomocą:

```
# aptitude hold package_name
```

Zamień `hold` na `unhold` w celu zlikwidowania statusu “hold”.

Jeśli znajdziesz tam coś, co wymaga naprawienia, najlepiej upewnij się, że plik `sources.list` odwołuje się do `sarge`, jak to opisano w ‘Sprawdzenie listy źródeł’ na stronie [49](#).

4.2.4 Źródła nieoficjalne i backporty

Jeśli na Twoim systemie znajdują się pakiety nie-Debianowe, powinieneś wiedzieć, że mogą one zostać usunięte w czasie aktualizacji ze względu na konflikty zależności. Jeśli pakiety te zostały zainstalowane poprzez dodanie archiwum w pliku `/etc/apt/sources.list`, powinieneś sprawdzić, czy to archiwum oferuje także pakiety skompilowane dla `etch` i zmienić linię w źródle odpowiednio na ten sam czas w stosunku do linii źródeł dla pakietów Debiana.

Niektórzy użytkownicy mają wydane nieoficjalnie (ang. `backported`) “nowsze” wersje pakietów, które są zainstalowane na ich wersji `sarge` systemu. Pakiety te są najczęściej źródłem problemów w czasie upgrde’*u* ze względu na konflikty zależności². Sekcja ‘Możliwe problemy podczas upgrde’*u*’ na stronie [26](#) zawiera informacje o tym, jak sobie poradzić w takich wypadkach.

²System zarządzania pakietami Debiana normalnie nie pozwala pakietom usuwać lub zmieniać plików, których właścicielami są inne pakiety, chyba że zostaną one zdefiniowane jako dopuszczone do zmiany przez ten pakiet.

4.3 Ręczne usunięcie znaczników pakietów

Aby zopobiec usunięciu niektórych pakietów przez `aptitude`, które zostały zaznaczone do usunięcia ze względu na zależności, może okazać się niezbędne ręczne usunięcie ich znacznika `auto`. Dotyczy to OpenOffice i Vim dla instalacji graficznych:

```
# aptitude unmarkauto openoffice.org vim
```

oraz obrazów jądra 2.6, jeśli instalowałeś je używając metapakietu jądra:

```
# aptitude unmarkauto $(dpkg-query -W 'kernel-image-2.6.*' | cut -f1)
```

Uwaga: Możesz sprawdzić, które pakiety są oznaczone jako `auto` w `aptitude` przez wykonanie:

```
# aptitude search 'i~M <package name>'
```

4.4 Przygotowanie źródeł dla APT

Zanim rozpoczniesz upgrade, musisz zmodyfikować pliki konfiguracyjne `apt` zawierające listy pakietów `/etc/apt/sources.list`.

`apt` przeszukuje wszystkie pakiety, które mogą zostać znalezione w liniach „deb” i instaluje pakiet z najwyższym numerem wersji, dając pierwszeństwo pierwszej takiej linii (czyli w przypadku istnienia kilku mirrorów, zwykle używa się najpierw lokalnego dysku, potem CD-ROM’u, a potem serwerów HTTP/FTP).

Odwołanie do wydania może odbywać się zarówno poprzez jego nazwę kodową (np. `sarge`, `etch`), jak też nazwę statusu (czyli `oldstable`, `stable`, `testing`, `unstable`). Odwołanie przez nazwę kodową ma tę zaletę, że użytkownik nie jest zaskoczony nowym wydaniem i z tego powodu przyjmujemy tu takie podejście. Oznacza to oczywiście, że samodzielnie musisz śledzić ogłaszane nowe wydania. Jeśli używasz nazwy statusu, zaobserwujesz po prostu ładowanie i aktualizowanie nowych pakietów, tak szybko, jak zostaną one wydane.

4.4.1 Dodanie do APT źródeł internetowych

Domyślna konfiguracja jest ustawiona na instalację z głównych serwerów internetowych Debiana, ale możesz zmienić `/etc/apt/sources.list`, aby używać innych mirrorów, najlepiej sieciowo najbliższych Twojej lokalizacji.

Adresy mirrorów HTTP lub FTP można znaleźć na <http://www.debian.org/distrib/ftplist> (w sekcji “Full list of mirrors” - pełna lista mirrorów). Serwery HTTP są generalnie szybsze niż FTP.

Załóżmy, że Twoim najbliższym mirrorem jest `http://mirrors.kernel.org/debian/`. Podczas sprawdzania tego mirroru za pomocą przeglądarki lub programu FTP, znajdziesz na nim katalogi zorganizowane w taki sposób:

```
http://mirrors.kernel.org/debian/dists/etch/main/binary-alpha/...
http://mirrors.kernel.org/debian/dists/etch/contrib/binary-alpha/...
```

Aby użyć mirroru programem `apt`, trzeba dodać linię do pliku `sources.list`:

```
deb http://mirrors.kernel.org/debian etch main contrib
```

Zauważ, że `'dists'` jest dodawane domyślnie, a argument po nazwie wydania jest używany do rozwinięcia ścieżki na wiele katalogów.

Po dodaniu nowych źródeł, zdeaktywuj poprzednio istniejące linie `"deb"` w `sources.list`, poprzez umieszczenie przed nimi znaku hash (`#`).

4.4.2 Dodanie źródeł APT dla lokalnego mirrora

Zamiast używać serwerów HTTP czy FTP, można zmodyfikować `/etc/apt/sources.list`, aby mieć mirror na dysku lokalnym (np. zamontowany poprzez NFS).

Na przykład mirror może być w katalogu `/var/ftp/debian/` i mieć główne katalogi:

```
/var/ftp/debian/dists/etch/main/binary-alpha/...
/var/ftp/debian/dists/etch/contrib/binary-alpha/...
```

Żeby użyć ich programem `apt`, dodaj linię do pliku `sources.list`:

```
deb file:/var/ftp/debian etch main contrib
```

Zauważ, że `'dists'` jest dodawane domyślnie, a argument po nazwie wydania jest używany do rozwinięcia ścieżki na wiele katalogów.

Po dodaniu nowych źródeł, zdeaktywuj poprzednio istniejące linie `"deb"` w `sources.list`, poprzez umieszczenie przed nimi znaku hash (`#`).

4.4.3 Dodanie źródeł APT z CD-ROM'u lub DVD

Jeśli używasz *tylko* płyt, zakomentuj istniejące linie `"deb"` w `/etc/apt/sources.list` przez umieszczenie przed nimi znaku hash (`#`).

Upewnij się, że w pliku `/etc/fstab` jest linia, która umożliwia zamontowanie CD-ROM'u w punkcie montowania `/cdrom` (wymagany jest dokładnie `/cdrom` dla `apt-cdrom`). Jeśli np. plik `/dev/hdc` reprezentuje napęd CD-ROM, `/etc/fstab` powinien zawierać linię:

```
/dev/hdc /cdrom auto defaults,noauto,ro 0 0
```


Zauważ, że pomiędzy `defaults`, `noauto`, `ro` w czwartym polu *nie ma spacji*.

Żeby sprawdzić, czy to zadziała, włóż płytkę i uruchom

```
# mount /cdrom      # montuje CD w punkcie montowania
# ls -alF /cdrom    # wyświetla zawartość katalogu głównego CD
# umount /cdrom     # odmontowuje CD-ROM
```

A następnie uruchom:

```
# apt-cdrom add
```

dla każdego CD-ROM’u z binariami Debiana, aby dodać dane nt. każdej z płyt do bazy danych APT.

4.5 Aktualizacja pakietów

Rekomendowanym sposobem aktualizacji z poprzednich edycji jest użycie narzędzia do zarządzania pakietami `aptitude`. Program ten podejmuje lepsze decyzje dot. instalacji pakietów, niż sam `apt-get`.

Nie zapomnij zamontować wszystkich niezbędnych partycji (zwłaszcza `roota` i `/usr`) jako zapisywalnych, poleceniem:

```
# mount -o remount,rw /mountpoint
```

Następnie upewnij się dwa razy, że źródła APT (w pliku `/etc/apt/sources.list`) odwołują się do `etch` lub do `stable`. Nie powinno być linii odwołujących się do `sarge`. Uwaga: linie dotyczące CD-ROM’u często odwołują się do `unstable`; pomimo, że wydaje się to dziwne, *nie* powinieneś tego zmieniać.

4.5.1 Nagranie sesji

Jest bardzo zalecane użycie programu `/usr/bin/script` do zapisania przebiegu sesji `upgrade’u`. Jeśli wystąpiłby jakiś problem, będziesz miał log zawierający wszystko, co się działo i, jeśli to konieczne, może udostępnić dokładne informacje do raportu o błędzie. Zapis rozpoczynamy przez:

```
# script -t 2>>~/upgrade-etch.time -a ~/upgrade-etch.script
```

albo podobnie. Nie należy kierować pliku wynikowego na katalogi tymczasowe, takie jak `/tmp` czy `/var/tmp` (pliki z tych katalogów mogą zostać skasowane podczas aktualizacji lub restartu).

Skrypt może również posłużyć do przejrzania informacji, które zostały przewinięte poza ekran. W tym celu przełącz się na VT2 (za pomocą `Alt-F2`) i po zalogowaniu wykonaj `less -R ~root/upgrade-etch.script`, aby zobaczyć plik.

Po zakończeniu aktualizacji możesz zatrzymać `script` przez napisanie `exit` po znaku zachęty.

Jeśli użyłeś parametru `-t` w `script`, możesz zastosować `scriptreplay` do odtworzenia całej sesji:

```
# scriptreplay ~/upgrade-etch.time ~/upgrade-etch.script
```

4.5.2 Aktualizacja listy pakietów

Najpierw konieczne jest ściągnięcie listy dostępnych pakietów nowego wydania. Robimy to przez wykonanie:

```
# aptitude update
```

Jeśli robisz to pierwszy raz, nowopobierane źródła będą powodować wyświetlanie ostrzeżeń dotyczących ich dostępności. Ostrzeżenia te są nieszkodliwe i nie będą się pojawiać, jeśli powtórzysz polecenie.

4.5.3 Upewnij się, że jest dość miejsca na dysku

Musisz upewnić się przed aktualizacją, że masz wystarczającą przestrzeń dyskową, zanim rozpoczniesz pełny upgrade systemu opisany w 'Upgrade reszty systemu' na stronie 25. Każdy ściągnięty z sieci pakiet jest najpierw przechowywany w katalogu `/var/cache/apt/archives` (i podkatalogu `partial/` w czasie ściągnięcia), więc musisz mieć wystarczająco dużo miejsca na partycji systemowej `/var/`, gdzie są przechowywane tymczasowo pakiety, które zostaną zainstalowane w systemie. Po ich ściągnięciu prawdopodobnie będzie potrzebne dodatkowe miejsce na innych partycjach, zarówno na zaktualizowanie istniejących pakietów (które mogą zawierać większe binaria lub dane), jak też na nowe pakiety ściągnięte do upgrade'u. Jeśli Twój system nie ma wystarczającej przestrzeni, aktualizacja może zakończyć niecałkowicie i być trudna do odratowania.

Zarówno `aptitude` jak `apt` pokazują szczegółowe informacje o przestrzeni dyskowej niezbędnej do instalacji. Przed rozpoczęciem aktualizacji, możesz sprawdzić to przez wykonanie:

```
# aptitude -y -s -f --with-recommends dist-upgrade
[ ... ]
XXX zaktualizowanych, XXX nowoinstalowanych, XXX do usunięcia i XXX nie aktua
Potrzebne ściągnięcie xx.xMB/yyyMB archiwów. Po rozpakowaniu zostanie użyte A
Zostaną ściągnięte/zainstalowane/usunięte pakiety.
```

3

Jeśli nie ma wystarczającej przestrzeni na upgrade, musisz przedtem zwolnić miejsce. Możesz:

- Usunąć pakiety, które zostały uprzednio ściągnięte do instalacji (w `/var/cache/apt/archive`). Czyszczenie cache pakietów przez wykonanie `apt-get clean` lub `aptitude clean` usunie wszystkie wcześniej pobierane pakiety.
- Usunąć stare pakiety, które już nie są używane. Jeśli masz zainstalowany `popularity-contest`, możesz sprawdzić listę nieużywanych pakietów, które zajmują większość miejsca za pomocą `popcon-largest-unused`. Możesz też użyć `deborphan` lub `debfoaster`, żeby znaleźć przestarzałe pakiety (zobacz ‘Przestarzałe pakiety’ na stronie 32). Jako alternatywę możesz uruchomić `aptitude` w “trybie wizualnym” i znaleźć przestarzałe pakiety w sekcji “Pakiety przestarzałe i utworzone lokalnie”.
- Usunąć pakiety zabierające dużo miejsca, które obecnie nie są potrzebne (możesz je zawsze doinstalować po aktualizacji). Możesz sprawdzić listę pakietów zajmujących najwięcej miejsca za pomocą `dpigs` (z pakietu `debian-goodies`) lub `wajig` (polecenie `wajig size`).
- Tymczasowo przenieść na inny system lub całkiem usunąć logi systemowe z katalogu `/var/log/`.

Aby bezpiecznie usunąć niepotrzebne pakiety, dobrze jest sprawdzić, czy plik `sources.list` odwołuje się z powrotem do `sarge`, jak to opisano w ‘Sprawdzenie listy źródeł’ na stronie 49.

4.5.4 Wstępny upgrade systemu

Ponieważ niektóre ważne pakiety są w konflikcie pomiędzy edycjami `sarge` i `etch`, wykonanie polecenia `aptitude dist-upgrade` może spowodować usunięcie znacznej liczby pakietów. Z tego powodu zalecamy przeprowadzenie upgrade’u w dwóch fazach, faza wstępna zapobiegnie tym konfliktom, a po niej nastąpi pełen upgrade `dist-upgrade`.

Najpierw uruchom:

```
# aptitude upgrade
```

³Running this command at the beginning of the upgrade process may give an error, for the reasons described in the next sections. In that case you will need to wait until you’ve done the minimal system upgrade as in ‘Wstępny upgrade systemu’ na bieżącej stronie and upgraded your kernel as in ‘Upgrade jądra’ na stronie 24 before running this command to estimate the disk space.

Spowoduje to zaktualizowanie tych pakietów, które mogą zostać zaktualizowane bez potrzeby usuwania lub instalowania innych pakietów.

Przeprowadź upgrade wstępny poprzez:

```
# aptitude install initrd-tools
```

Ten krok spowoduje aktualizację pakietów `libc6` i `locales`, a także zainstaluje biblioteki obsługi SELinuxa (`libselinux1`). W tym momencie niektóre serwisy zostaną zrestartowane, np. `xdm`, `gdm` i `kdm`. W konsekwencji zostaną też odłączone lokalne sesje X11.

Następny krok zależy w znacznym stopniu od tego, jakie pakiety są zainstalowane w systemie. Niniejsze uwagi zawierają ogólne porady dotyczące metod postępowania, w razie wątpliwości zalecamy sprawdzenie, które pakiety zostaną usunięte przed wykonaniem poszczególnych czynności.

Spodziewaj się usunięcia pakietów takich jak: `base-config`, `hotplug`, `xlibs`, `netkit-inetd`, `python2.3`, `xfree86-common` i `xserver-common`. Pełną listę przestarzałych pakietów `etch` można znaleźć tu 'Przestarzałe pakiety' na stronie [32](#).

Upgrade środowiska graficznego (desktop system)

Ten sposób upgrade'u został przetestowany na systemach z zainstalowanym zadaniem `desktop`. Prawdopodobnie jest to najlepsza metoda dla systemów z zainstalowanymi zadaniami (pakietami) `desktop`, `gnome` lub `kde`.

Prawdopodobnie *nie* jest to dobra metoda, jeśli masz zainstalowane pakiety `libfam0c102` i `xlibmesa-glu`:

```
# dpkg -l libfam0c102 | grep ^ii
# dpkg -l xlibmesa-glu | grep ^ii
```

Jeśli masz zainstalowany pełen system środowiska graficznego, wykonaj:

```
# aptitude install libfam0 xlibmesa-glu
```

Aktualizowanie systemu z częściowo zainstalowanym systemem X

Systemy z zainstalowanymi pakietami X, ale bez wykonanego zadania `desktop`, wymagają innej metody. Metoda ta dotyczy w ogólności systemów z zainstalowanym pakietem `xfree86-common`, łącznie z niektórymi serwerami, które mają zainstalowane zadania serwera `tasksel`, jak zadania wymagające graficznych narzędzi zarządzania. Jest poprawną metodą do użycia w systemach, które używają X-ów, ale nie mają zainstalowanego pełnego zadania `desktop`.

```
# dpkg -l xfree86-common | grep ^ii
```

Najpierw sprawdź, czy są zainstalowane pakiety `libfam0c102` i `xlibmesa-glu`.

```
# dpkg -l libfam0c102 | grep ^ii
# dpkg -l xlibmesa-glu | grep ^ii
```

Jeśli `libfam0c102` nie jest zainstalowany, nie włączaj `libfam0` do podanego polecenia. Jeśli nie jest zainstalowany `xlibmesa-glu`, nie włączaj go do tego polecenia.⁴

```
# aptitude install x11-common libfam0 xlibmesa-glu
```

Zauważ, że instalacja `libfam0` spowoduje zainstalowanie File Alteration Monitor (`fam`), jak też RPC portmapera (`portmap`), jeśli nie było ich w systemie. Oba pakiety powodują włączenie nowej usługi sieciowej, chociaż mogą być skonfigurowane tak, aby używać wewnętrznego (`loopback`) urządzenia sieciowego.

Aktualizacja systemu bez X-ów

On a system with no X, no additional `aptitude install` command should be required, and you can move on to the next step.

4.5.5 Upgrade jądra

Wersja `udev` z `etch` nie wspiera wcześniejszych wersji jądra niż 2.6.15 (co obejmuje jądro 2.6.8 z `sarge`) i odwrotnie, wersja `udev` z `sarge` nie będzie pracować poprawnie z najnowszym jądrem. W dodatku zainstalowanie pakietu `udev` z `etch` wymusza usunięcie pakietu `hotplug`, używanego przez jądro 2.4.

Jako konsekwencja, poprzednia wersja jądra nie zbootuje się prawidłowo po aktualizacji. Analogicznie, istnieje pewien moment w czasie `upgrade'u`, w którym pakiet `udev` zostanie zaktualizowany, ale nie zostanie zainstalowana jeszcze najnowsza wersja jądra. Jeśli w tym czasie system zostanie zresetowany, w środku `upgrade'u`, może on przestać się bootować ze względu na brak możliwości prawidłowej detekcji i ładowania sterowników (zobacz 'Przygotuj bezpieczne środowisko dla aktualizacji' na stronie 15 zalecenia dotyczące przygotowania się na tę okoliczność w czasie zdalnej aktualizacji).

Jeżeli system nie ma zainstalowanego zadania `desktop` lub innych pakietów, które spowodowałyby znaczną liczbę usunięć, zalecamy `upgrade` jądra w tym momencie.

Upgrade jądra wykonujemy poprzez:

⁴To polecenie sprawdza, czy jest konieczne zainstalowanie `libfam0` i `xlibmesa-glu` i pokazuje status:

```
# aptitude install x11-common \ $(dpkg-query -showformat '${Package} ${Status}\n' -W
libfam0c102 xlibmesa-glu \ | grep 'ok installed$' | sed -e's/ .*//; s/c102//')
```

```
# aptitude install linux-image-2.6-flavor
```

Zobacz ‘Instalacja metapakieta jądra’ na stronie [27](#), aby określić, który obraz jądra powinienies zainstalować.

W przypadku środowiska graficznego niestety nie jest możliwe określenie, który pakiet jądra został zainstalowany bezpośrednio po instalacji nowego pakietu `udev`, tak więc mamy do czynienia z nieokreślonym przedziałem czasu, w którym nie ma zainstalowanego jądra z pełną obsługą hotplug. Zobacz ‘Aktualizacja jądra i jego pakietów’ na stronie [27](#) informacje nt. konfiguracji niezależnej od hotplug w celu zbootowania.

4.5.6 Upgrade reszty systemu

Teraz jesteśmy gotowi na przeprowadzenie głównej części aktualizacji. Wykonujemy:

```
# aptitude dist-upgrade
```

Zostanie wykonana pełna aktualizacja systemu, to jest instalacja najnowszych dostępnych wersji wszystkich pakietów, rozwiązanie wszystkich możliwych konfliktów zależności pomiędzy pakietami w różnych wydaniach. Jeśli konieczne, instalacja nowych pakietów (zwykle nowych wersji bibliotek lub pakietów przenazwanych) oraz usunięcie pakietów przestarzałych pozostających w konfliktach.

Jeśli aktualizujesz ze zbioru płyt CD, będziesz proszony o wkładanie różnych płyt w wielu momentach aktualizacji. Możliwe jest wkładanie tych samych płyt wiele razy z powodu zależnych od siebie pakietów umieszczonych na różnych płytach.

Nowe wersje bieżąco instalowanych pakietów, które nie mogą zostać zaktualizowane bez zmiany statusu innego pakietu, zostaną pozostawione w ich obecnych wersjach (wyświetlanych jako “held back” - utrzymane). Rozwiązaniem jest wtedy albo użycie `aptitude` do wybrania tych pakietów do instalacji, albo próba zastosowania `aptitude -f install package`.

4.5.7 Pobranie podpisów pakietów

Po aktualizacji, mając mową wersję `apt`, można dokonać update’u informacji o pakietach, włączając w to uruchomienie nowego mechanizmu kontroli podpisów:

```
# aptitude update
```

Po aktualizacji są już dostępne i włączone klucze do podpisywania archiwów pakietów Debiana. Jeśli dodasz inne (nieoficjalne) źródło pakietów, `apt` będzie wyświetlał ostrzeżenia dotyczące niezgodności w pochodzeniu pakietów. Więcej informacji ‘Zarządzanie pakietami’ na stronie [5](#).

Zostaniesz ostrzeżony, że od momentu, w którym zaczniesz używać nowej wersji `apt`, będą ściągane pliki różnicowe (`pdiff`) zamiast pełnej listy indeksu pakietów. Więcej informacji o tej funkcjonalności jest na ‘Spowolnione update’y plików indeksowych APT’ na stronie [36](#).

4.5.8 Możliwe problemy podczas upgrade'u

Jeśli operacja używająca `aptitude`, `apt-get` lub `dpkg` zakończy się błędem

```
E: Dynamic MMap ran out of room
```

domyślna przestrzeń cache jest niewystarczająca. Możesz naprawić to albo przez usunięcie lub zakomentowanie linii, które nie są potrzebne w `/etc/apt/sources.list`, albo poprzez zwiększenie przestrzeni cache. Może ona zostać powiększona przez ustawienie `APT::Cache-Limit` w pliku `/etc/apt/apt.conf`. Następujące polecenie ustawia jej wartość na wystarczającą do aktualizacji:

```
# echo 'APT::Cache-Limit "12500000";' >> /etc/apt/apt.conf
```

Przyjęto, że zmienna ta nie była jeszcze ustawiana w pliku.

Czasem konieczne jest włączenie opcji `APT::Force-LoopBreak`, żeby umożliwić tymczasowe usunięcie jakiegoś ważnego pakietu ze względu na konflikty zależności. `aptitude` będzie ostrzegać o tym i przerywać aktualizację. Możesz kontynuować pracę przez dodanie opcji `-o APT::Force-LoopBreak=1` przy wywołaniu `aptitude`.

Jest możliwe, że struktura zależności w systemie jest uszkodzona, tak że wymaga ręcznej interwencji. Zwykle oznacza to użycie `aptitude` lub

```
# dpkg --remove package_name
```

aby wyeliminować jakieś przeszkadzające pakiety lub

```
# aptitude -f install
# dpkg --configure --pending
```

W wyjątkowych przypadkach konieczne może być wymuszenie reinstalacji poleceniem typu

```
# dpkg --install /path/to/package_name.deb
```

Konflikty plików nie powinny wystąpić, jeśli aktualizujesz "czysty" system sarge, ale mogą zdarzyć się, jeśli masz zainstalowane nieoficjalne backporty. Konflikt pliku objawia się błędem:

```
Unpacking <package-foo> (from <package-foo-file>) ...
dpkg: error processing <package-foo> (--install):
 trying to overwrite '<some-file-name>',
 which is also in package <package-bar>
dpkg-deb: subprocess paste killed by signal (Broken pipe)
Errors were encountered while processing:
 <package-foo>
```

Możesz próbować rozwiązać konflikt przez wymuszenie usunięcia pakietu wymienionego w *ostatniej* linii komunikatu o błędzie:

```
# dpkg -r --force-depends package_name
```

Po naprawie powinieneś móc powrócić do aktualizacji przez powtórzenie opisanych wcześniej poleceń `aptitude`.

W czasie aktualizacji będziesz proszony o podanie odpowiedzi na pytania dotyczące konfiguracji lub rekonfiguracji niektórych pakietów. Jeśli będziesz pytany, czy jakiś plik z katalogów `/etc/init.d` lub `/etc/terminfo` lub plik `/etc/manpath.config` ma być zastąpiony, należy dać odpowiedź 'tak' ('yes'), aby zapewnić spójność systemu. Zawsze możesz przywrócić starą wersję, gdyż są one zachowywane z rozszerzeniem `.dpkg-old`.

Jeśli nie jesteś pewien, co zrobić, zapisz sobie nazwę pakietu lub pliku i pozostaw sprawy do przemyślenia. Możesz też sprawdzić skrypt wykonania i stamtąd uzyskać informacje, które pojawiały się na ekranie w czasie aktualizacji.

4.6 Aktualizacja jądra i jego pakietów

Ta sekcja wyjaśnia, w jaki sposób dokonać upgrade'u jądra i omawia potencjalne zagrożenia odnoszące się do tego. Możesz albo zainstalować jeden z pakietów `linux-image-*` udostępnianych przez Debiana, lub też skompilować jądro ze źródeł.

Informacje podane w tej sekcji opierają się na założeniu, że używasz jednego z dostępnych modularnych jąder Debiana łącznie z pakietami `initramfs-tools` i `udev`. Jeśli wybrałeś możliwość kompilacji jądra, które nie używa `initrd`, lub też używasz innego generatora `initrd`, niektóre z podanych są dla Ciebie nieprzydatne.

Jeśli pakiet `udev` *nie* jest instalowany w Twoim systemie, możliwe jest użycie pakietu `hotplug` do wykrywania sprzętu.

Jeśli używasz jądra 2.4, powinieneś przeczytać dokładnie także 'Upgrade do jądra 2.6' na stronie [37](#).

4.6.1 Instalacja metapakietu jądra

W czasie upgrade'u z `sarge` do `etch`, gorąco zalecamy zainstalować nowy metapakiet `linux-image-2.6-*`. Może on zostać zainstalowany automatycznie w czasie procesu upgrade'u. Można to sprawdzić przez uruchomienie:

```
# dpkg -l "linux-image*" | grep ^ii
```

Jeśli nie ma wyniku, powinieneś zainstalować nowy pakiet `linux-image` ręcznie. Aby sprawdzić listę dostępnych metapakietów `linux-image-2.6` wykonaj:


```
# apt-cache search linux-image-2.6- | grep -v transition
```

Jeśli nie jesteś pewien, który pakiet wybrać, wykonaj `uname -r` i wybierz pakiet o podobnej nazwie. Np. jeśli widzisz u siebie '2.4.27-3-686', najlepiej zainstaluj `linux-image-2.6-686`. Możesz też użyć `apt-cache`, aby przeczytać opisy każdego z pakietów i w ten sposób wybrać najbardziej odpowiedni. Przykładowo:

```
# apt-cache show linux-image-2.6-686
```

Powinieneś użyć `aptitude install` do zainstalowania pakietu. Od momentu zainstalowania nowego jądra i zrestartowania maszyny przy najbliższej okazji, możesz korzystać z zalet nowej wersji jądra.

Jeszcze więcej zalet ma skompilowanie na Debian GNU/Linuxie jądra dostosowanego do Twoich zasobów. Zainstaluj narzędzie `kernel-package` i przeczytaj dokumentację w `/usr/share/doc/kernel-package`.

4.6.2 Aktualizacja wewnątrz serii 2.6

Jeśli obecnie używasz jądra serii 2.6 z `sarge` ta aktualizacja będzie miała miejsce automatycznie po wykonaniu pełnego upgrade'u pakietów systemu (jak opisano w 'Aktualizacja pakietów' na stronie 20).

Jeśli jest to możliwe, zyskasz sporo robiąc upgrade pakietu jądra oddzielnie od głównego `dist-upgrade`, co zredukuje czas, w którym system jest niebootowalny. Zobacz 'Upgrade jądra' na stronie 24, gdzie opisano ten proces. Prosimy zwrócić uwagę, że powinno to być robione po wykonaniu fazy wstępnej upgrade'u, opisanej w 'Wstępny upgrade systemu' na stronie 22.

Możesz także wykonać ten krok, jeśli używasz swojego własnego jądra i chcesz używać jądra z wersji `etch`. Jeśli Twoje jądro nie jest wspierane przez `udev`, zalecamy jego aktualizację po wstępnej fazie upgrade'u. Jeśli Twoja wersja jest wspierana przez `udev`, możesz bezpiecznie poczekać, aż zakończy się pełna aktualizacja systemu.

4.6.3 Upgrade z jądra 2.4

Jeśli masz zainstalowane jądro 2.4 i system wykrywa sprzęt używając pakietu `hotplug`, powinieneś najpierw dokonać upgrade'u jądra do wersji 2.6 z `sarge`, zanim przystąpisz do aktualizacji. Upewnij się, że system bootuje się z jądrem 2.6 i cały sprzęt jest wykrywany poprawnie, zanim przystąpisz do aktualizacji. Pakiet `hotplug` jest usuwany z systemu (na rzecz pakietu `udev`), kiedy wykonujesz pełną aktualizację systemu. Jeśli nie wykomasz przedtem upgrade'u jądra, system może nie zbootować się prawidłowo. Jeśli wykonałeś już upgrade do jądra serii 2.6 w `sarge`, możesz aktualizować jądro, jak to jest opisane w 'Aktualizacja wewnątrz serii 2.6' na bieżącej stronie.

Jeśli system nie bazuje na pakiecie `hotplug`⁵, możesz opóźnić upgrade jądra i wykonać go po pełnej aktualizacji systemu, jak to zostało opisane w ‘Upgrade reszty systemu’ na stronie 25. Jeśli system został już zaktualizowany, możesz wykonać, co następuje (zmiana nazwy pakietu jądra na bardziej odpowiednią, poprzez zastąpienie `<obraz>`):

```
# aptitude install linux-image-2.6-<obraz>
```

4.6.4 Przeorganizowanie nazw urządzeń

`etch` posiada bardziej wydajny mechanizm wykrywania sprzętu, niż poprzednie wydania. Jednak może to skutkować zmianami w przyporządkowaniu wyszukiwanych urządzeń do odpowiadających im nazw. Na przykład jeśli masz dwie karty sieciowe, które korzystają z różnych sterowników, odwołania do urządzeń `eth0` i `eth1` mogą zostać zamienione. Zauważ, że nowy mechanizm powoduje również, że jeśli np. zmieniasz karty sieciowe w pracującym `etch`, system może przyporządkować nowej karcie nową nazwę interfejsu.

Dla urządzeń sieciowych można zapobiec takim zamianom za pomocą reguł pakietu `udev`, a zwłaszcza definicjom reguł zawartym w `/etc/udev/rules.d/z25_persistent-net.rules`⁶. Jako alternatywę możesz użyć programu `ifrename` do dowiązania urządzeń fizycznych do nazw używanych podczas bootowania. Zobacz `ifrename(8)` i `iftab(5)`. Obie możliwości (`udev` i `ifrename`) nie powinny być używane jednocześnie.

Dla urządzeń przechowywania danych (ang. storage devices) możesz ominąć zmianę kolejności za pomocą pakietu `initramfs-tools` skonfigurowanego w taki sposób, aby łądował sterowniki urządzeń w kolejności, w jakiej są obecnie załadowane. Aby to zrobić, musisz zidentyfikować kolejność, w jakiej sterowniki są łądowane wywołując `lsmod`. `lsmod` pokazuje listę modułów w kolejności odwrotnej do kolejności łądowania, czyli pierwszy moduł na liście został załadowany na końcu. Działa to jednak tylko dla urządzeń, które jądro wykrywa w stałej kolejności (np. karty PCI).

Zwróć uwagę, że usunięcie i powtórne załadowanie modułów po starcie systemu, spowoduje zmiany w ich kolejności. Oprócz tego, niektóre sterowniki są łączone (link) przez jądro statycznie i nie zostaną wyświetlone przez `lsmod`. Możesz odcyfrować ich nazwy i kolejność łądowania albo z pliku `/var/log/kern.log`, albo programem `dmesg`.

Dodaj nazwy tych modułów do pliku `/etc/initramfs-tools/modules` w kolejności, w jakiej powinny się łądować w czasie bootu. Niektóre nazwy mogły ulec zmianie pomiędzy `sarge` a `etch`. Np. `sym53c8xx_2` stało się `sym53c8xx`.

Trzeba też zregenerować obraz(y) `initramfs` przez wykonanie `update-initramfs -u -k all`.

⁵Może tak być, jeśli moduły jądra są łądowane statycznie poprzez konfigurację zapisaną w pliku `/etc/modules`

⁶Reguły są generowane automatycznie przez skrypt `/etc/udev/rules.d/z45_persistent-net-generator.rules` w celu zachowania niezmienności nazw interfejsów sieciowych. Usunięcie tego linku spowoduje wyłączenie niezmienności nazw dla NIC-ów pakietu `udev`.

Jeśli używasz już jądra `etch` i `udev`, możesz przekonfigurować system, aby uzyskać dostęp do dysków za pomocą aliasów, co uniezależnia go od kolejności ładowania sterowników. Aliasy znajdują się w plikach `/dev/disk/`.

4.6.5 Problemy z bootowaniem

Jeśli do bootowania jest używane `initrd` utworzone przy pomocy pakietu `initramfs-tools`, w niektórych przypadkach tworzenie plików urządzeń przez pakiet `udev` może następować zbyt późno, aby były przydatne dla skryptu bootowania.

The usual symptoms are that the boot will fail because the root file system cannot be mounted and you are dropped into a debug shell, but that when you check afterwards, all devices that are needed are present in `/dev`. This has been observed in cases where the root file system is on a USB disk or on RAID, especially if `lilo` is used.

Rozwiązaniem tego problemu jest użycie parametru bootowania `rootdelay=9`. Wartość opóźnienia (w sekundach) może być zmieniona.

4.7 Zadania przed restartem komputera

Kiedy zakończy się działanie `aptitude dist-upgrade`, aktualizacja “formalnie” jest gotowa, ale warto pamiętać o kilku rzeczach przed zrebootowaniem maszyny.

4.7.1 Konwersja z devfs

Jądro Debiana nie wspiera już `devfs`, tak więc użytkownicy `devfs` muszą ręcznie skonwertować swój system, zanim zbootują jądro `etch`.

Jeśli w pliku `/proc/mounts` widać napis `'devfs'`, oznacza to zwykle, że używasz `devfs`. Wszystkie pliki konfiguracyjne, które odwołują się do nazw używanych przez `devfs`, powinny zostać zmienione i używać nazw odpowiednich dla pakietu `udev`. Przede wszystkim są to pliki `/etc/fstab`, `/etc/lilo.conf`, `/boot/grub/menu.lst` i `/etc/inittab`.

Więcej informacji na temat potencjalnych problemów znajduje się w raporcie o błędzie #341152 (<http://bugs.debian.org/341152>).

4.7.2 Aktualizacja mdadm

`mdadm` wymaga obecnie pliku konfiguracyjnego do złożenia macieży MD (RAID) z inicjalnego ramdysku podczas sekwencji inicjalizacji systemu. Prosimy przeczytać i wykonać instrukcje zawarte w pliku `/usr/share/doc/mdadm/README.upgrading-2.5.3.gz` po aktualizacji pakietu, **a przed rebootem**. Najnowsza wersja tego pliku jest dostępna na <http://svn.debian.org/wsvn/pkg-mdadm/mdadm/trunk/debian/README.upgrading-2.5.3?op=file>; prosimy przeczytać w razie problemów.

4.8 Przygotowanie do nowej edycji

Po upgrade można zrobić jeszcze kilka rzeczy, aby całkowicie przygotować się do używania nowego wydania.

- Jeżeli używasz grub, wyedytuj plik `/etc/kernel-img.conf` i popraw położenie programu `update-grub` z `/sbin/update-grub` na `/usr/sbin/update-grub`.
- Jeśli nowy obraz jądra został poprzez zależności wpisany na miejsce starego, zostanie on zaznaczony jako automatycznie zainstalowany, co trzeba poprawić:

```
# aptitude unmarkauto $(dpkg-query -W 'linux-image-2.6-*' | cut -f1)
```

- Usuń metapakiety jądra sarge poprzez:

```
# aptitude purge kernel-image-2.6-<flavor>
```

- Move any configuration options from `/etc/network/options` to `/etc/sysctl.conf`. Please see `/usr/share/doc/netbase/README.Debian` for details.
- Usuń przestarzałe i nieużywane pakiety, jak to opisano w 'Przestarzałe pakiety' na następnej stronie. Powinieneś przejrzeć, których plików konfiguracyjnych one używały i rozważyć wyczyszczenie (purge) tych pakietów w celu usunięcia zbędnych plików.

4.9 Deprecated packages

With the release of Lenny a bigger number of server packages will be deprecated, thus updating to newer versions of those now will save you from trouble when updating to Lenny.

This includes the following packages:

- `apache (1.x)`, successor is `apache2`
- `bind8`, successor is `bind9`
- `php4`, successor is `php5`
- `postgresql-7.4`, successor is `postgresql-8.1`
- `exim 3`, successor is `exim4`

4.10 Przeszarzałe pakiety

Wprowadzając tysiące nowych pakietów etch usuwa i pomija ponad dwa tysiące starych pakietów, które były w sarge. Dla tych pakietów aktualizacja nie jest przewidziana. Aby nie pozbawiać bezpieczeństwa tych pakietów, projekt Debian jak zwykle zaprzestanie wspierać je w zakresie bezpieczeństwa po roku od wydania etch⁷ i nie będzie prowadził żadnego innego wsparcia w tym czasie. Jeśli to możliwe, należy zastąpić je dostępnymi alternatywami.

Jest wiele powodów, dla których pakiety mogły zostać usunięte z dystrybucji: zakończono wydawanie ich wersji pierwotnych, nie ma już Deweloperów Debiana zainteresowanych pielęgnacją danego pakietu, funkcjonalność została przejęta przez inne oprogramowanie (lub nową wersję) lub też zadecydowano, że nie jest ona już odpowiednia dla etch ze względu na błędy. W tym ostatnim przypadku, pakiet może być dostępny w dystrybucji “unstable”.

Sprawdzenie, które pakiety w aktualizowanym systemie są “przeszarzałe” jest proste, gdyż programy obsługi pakietów pokazują to odpowiednim znacznikiem. Jeśli używasz `aptitude`, zobaczysz listę tych pakietów w sekcji “Pakiety przeszarzałe i utworzone lokalnie”. `dselect` oferuje zbliżoną sekcję, choć przedstawiona lista może się różnić. Jeśli więc użyłeś `aptitude` do ręcznej instalacji pakietów w sarge, będzie ona trzymać do nich ścieżkę i będzie mogła oznaczyć jako przeszarzałe te pakiety, z których zależności wynika, że nie są już potrzebne lub powinny zostać usunięte. Wynika z tego, że `aptitude`, inaczej niż `deborphan`, nie zaznaczy jako przeszarzałe pakietów, które zainstalowałeś ręcznie, w przeciwieństwie do instalowanych automatycznie ze względu na zależności.

Istnieją też dodatkowe narzędzia, którymi możesz znaleźć przeszarzałe pakiety, takie jak `deborphan`, `debfooster` czy `cruft`. `deborphan` jest usilnie zalecany, chociaż (w trybie standardowym) raportuje tylko przeszarzałe biblioteki: pakiety w sekcjach “libs” lub “oldlibs”, które nie są używane przez inne pakiety. Nie należy wyrzucać bez zastanowienia pakietów pokazywanych przez te narzędzia, zwłaszcza jeśli używasz agresywnych, niestandardowych opcji, które mogą generować tzw. fałszywe pozytywy. Bardzo prosimy ręcznie sprawdzić pakiety sugerowane do usunięcia (ich zawartość, rozmiar i opis), zanim je wyrzucisz.

Strona System Śledzenia Błędów Debiana (<http://bugs.debian.org/>) często zawiera dodatkowe informacje nt. przyczyn usunięcia danego pakietu. Powinieneś przejrzeć zarówno archiwa błędów samego pakietu jak i raporty dla `ftp.debian.org` pseudopakietów (<http://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=ftp.debian.org&archive=yes>).

4.10.1 Ślepe pakiety

Niektóre pakiety z sarge zostały połączone w inne w etch, często po to, aby ułatwić opiekę nad systemem. Aby ułatwić sposób aktualizacji w takich przypadkach, etch często wprowadza “ślepe” pakiety: puste pakiety, które mają taką samą nazwę, jak te stare w sarge z zależnościami, które wprowadzają nowe pakiety do instalacji. Te “ślepe” pakiety stają się przeszarzałe po upgrade i mogą być bezpiecznie usunięte.

⁷Lub aż do wydania nowej wersji stabilnej w tym czasie. Zwykle są wspierane tylko dwa wydania stabilne jednocześnie.

Większość (ale nie wszystkie) opisów ślepych pakietów pokazuje ich przeznaczenie. Opisy ślepych pakietów nie są ujednolicone, ale za pomocą `deborphan` z opcją `--guess` możesz wyszukać je w swoim systemie. Dodajmy, że ślepe pakiety nie są przeznaczone do usunięcia po upgrade systemu, ale odwrotnie, trzymają one ślad do obecnie dostępnych wersji programów.

Rozdział 5

Kwestie, o których warto wiedzieć mając etch

5.1 Potencjalne problemy

Czasem poprawki mają efekty uboczne, którym nie udało się zapobiec lub też opisujemy błędy w innych miejscach. Tutaj dokumentujemy te problemy, o których warto wspomnieć. Prosimy przeczytać erratę, odnośną dokumentację pakietów, raporty o błędach i inne informacje wspomniane w 'Co czytać dalej' na stronie [45](#).

5.1.1 Problemy z urządzeniami odnoszącymi się do udev

Pomimo, że udev był szeroko testowany, możesz napotkać pewne problemy z niektórymi urządzeniami wymagające jeszcze naprawy. Zwykle dotyczą one zmiany uprawnień lub właściciela urządzenia. W niektórych przypadkach urządzenie może nie zostać domyślnie utworzone (np. `/dev/video i /dev/radio`).

Pakiet udev dostarcza mechanizmów konfiguracyjnych pozwalających na rozwiązanie tych problemów. Zobacz szczegóły w `udev(8)` i `/etc/udev`.

5.1.2 Niektóre aplikacje nie będą pracować z jądrem 2.4

Niektóre aplikacje w etch mogą nie współpracować z jądrem 2.4, gdyż na przykład wymagają funkcji `epoll()`, która nie jest dostępna w jądrze 2.4. Takie aplikacje mogą albo całkiem nie działać, albo działać nieprawidłowo, dopóki system nie ma jądra 2.6.

Przykładem jest serwer HTTP proxy `squid`.

5.1.3 Właściwe miejsca sieciowe nie mogą zostać osiągnięte przez TCP

Od wersji 2.6.17 Linux agresywnie używa skalowania okna TCP, wyspecyfikowanego w RFC 1323. Niektóre serwery zachowują się nieprawidłowo i pokazują błędną wielkość okna. Prosimy przeczytać opisy błędów #381262 (<http://bugs.debian.org/381262>), #395066 (<http://bugs.debian.org/395066>), #401435 (<http://bugs.debian.org/401435>).

Zwykle są dwa sposoby rozwiązania tych problemów: albo zmiana maksymalnej wielkości okna TCP na mniejszą wartość (zalecane), albo całkowite wyłączenie skalowania okna (przeszarżale). Przykładowe polecenia są na stronie debian-installer errata (<http://www.debian.org/devel/debian-installer/errata>).

5.1.4 Spowolnione update'y plików indeksowych APT

Domyślnie wersja programu `apt` z `etch` używa nowego sposobu update'u plików indeksowych APT (kiedy jest uruchamiane polecenie `aptitude update`), który pobiera pliki różnicowe (zamiast pełnych plików indeksowych pakietów), zwanego `pdiff`. Funkcja ta powinna używać mniejszego pasma przesyłowego i być szybsza dla większości systemów. Niestety, daje ona efekt odwrotny w wypadku aktualizacji dla systemów z szybkimi łączami sieciowymi (lub bardzo blisko serwerów), które nie są często aktualizowane, gdyż łączenie plików różnicowych może wtedy przebiegać wolniej niż pobieranie pełnych indeksów. Można wyłączyć tę funkcjonalność przez dodanie `Acquire::Pdiffs "false";` w pliku `/etc/apt/apt.conf`.

Zmiana dotyczy bardziej użytkowników dystrybucji *unstable* i *testing* Debian GNU/Linux, ze względu na naturalną zmienność tych archiwów. Użytkownicy `etch` powinni zwrócić na nią uwagę głównie przy update'ach statusów pakietów archiwum bezpieczeństwa.

5.1.5 Inicjalizacja sieci asynchronicznej może mieć nieprzewidziany przebieg

Na systemach, które używają pakietu `udev` do załadowania interfejsów sieciowych, możliwe jest, że ze względu na asynchroniczną naturę `udev`, sterownik sieciowy nie zostanie załadowany przed uruchomieniem `/etc/init.d/networking` w czasie bootowania systemu. Pomimo, że dodanie `allow-hotplug` do `/etc/network/interfaces` (jako dodatku do `auto`) zagwarantuje, że interfejs sieciowy zostanie włączony, kiedy tylko stanie się dostępny, ale nie ma gwarancji, że stanie się to zanim sekwencja bootująca rozpocznie uruchamianie serwisów sieciowych, tak więc niektóre z nich mogą zachowywać się nieprawidłowo ze względu na brak intefejsu.

5.1.6 Trudności z używaniem bezpiecznych sieci bezprzewodowych WPA

W `sarge` pakiet `wpasupplicant` był usługą systemową konfigurowaną przez pliki `/etc/default/wpasupplicant` i `/etc/wpasupplicant.conf`.

W etch plik `/etc/init.d/wpa_supplicant` został usunięty i zintegrowany z `/etc/network/interfaces`, podobnie jak inne pakiety, takie jak `wireless-tools`. Oznacza to, że `wpa_supplicant` nie jest już usługą systemową w sposób bezpośredni.

Więcej o konfiguracji `wpa_supplicant` prosimy przeczytać w pliku `/usr/share/doc/wpa_supplicant/README.modes.gz`, gdzie są przykładowe pliki `/etc/network/interfaces`. Aktualne informacje o pakiecie `wpa_supplicant` można znaleźć na Debian Wiki (<http://wiki.debian.org/WPA>).

5.1.7 Problemy z plikami o nazwach zawierających znaki nie-ASCII

Podczas montowania filesystemów typu `vfat`, `ntfs` lub `iso9660`, które zawierają w nazwach plików znaki spoza zakresu ASCII, można spodziewać się błędów, jeśli nie montujemy ich z opcją `utf8`. Wskazówką może być komunikat o błędzie `'Invalid or incomplete multi-byte or wide character'`. Rozwiązaniem tego problemu może być użycie opcji montowania `defaults,utf8` dla filesystemów `vfat`, `ntfs` i `iso9660`, które zawierają takie znaki.

Jądro Linuksa nie zawsze rozróżnia wielkość liter w nazwach plików dla `vfat`, jeśli została użyta opcja `utf8`.

5.1.8 Sound stops working

In rare cases the sound might stop working after the upgrade. If this happens, go through the `alsa` checklist: run `alsactool` as root user, add your user to the `audio` group, use `alsamixer` and make sure levels are up and it is unmuted, make sure `arts` or `esound` stopped, make sure `OSS` modules unloaded, make sure speakers are on, check whether the command `cat /dev/urandom > /dev/dsp` works for root.

5.2 Upgrade do jądra 2.6

Jądro serii 2.6 zawiera poważne zmiany w stosunku do serii 2.4. Moduły mają zmienione nazwy, wiele sterowników zostało częściowo, a czasem prawie całkowicie przerobionych. Upgrade do jądra 2.6 z wcześniejszej wersji nie jest więc procesem prostym do przeprowadzenia. Ta sekcja ma na celu zwrócić uwagę na niektóre, związane z tym, problemy.

Jeśli kompilowałeś swoje własne jądro ze źródeł, upewnij się, że masz zainstalowany pakiet `module-init-tools` przed restartem z jądrem 2.6. Pakiet ten zastępuje `modutils` dla jądra serii 2.6. Jeśli instalowałeś jeden z pakietów `linux-image` Debiana, rzeczony pakiet zostanie zainstalowany automatycznie ze względu na zależności.

Jeśli używasz `LVM`, powinieneś zainstalować także `lvm2` zanim zrebootujesz z jądrem 2.6, ponieważ nie wspiera ono bezpośrednio `LVM1`. Aby mieć dostęp do woluminów `LVM1`, jest używana warstwa kompatybilności z `lvm2` (moduł `dm-mod`). Możesz pozostawić zainstalowany pakiet `lvm10`; skrypt `init`, wyszukujący pracującą wersję jądra, wybierze prawidłową wersję pakietu.

Jeśli masz wpisy w pliku `/etc/modules` (lista modułów ładowanych podczas bootu systemu), zwróć uwagę, że nazwy niektórych modułów uległy zmianie. Jeśli tak jest, musisz poprawić ten plik o nowe nazwy modułów.

Jeśli już zainstalowałeś jądro 2.6, ale przed restartem, upewnij się, że potrafisz odzyskać system po ewentualnej awarii. Po pierwsze, sprawdź, że konfiguracja bootloadera posiada wpisy odnoszące się do obydwu jąder, zarówno nowego, jak i starego, pracującego jądra 2.4. Powinieneś też mieć dyskietkę "ratunkową" lub CD-ROM, z którego skorzystasz w przypadku błędu konfiguracji, aby zbootować się ze starego jądra.

5.2.1 Konfiguracja klawiatury

Najdalej idące zmiany w jądrze 2.6 dotyczą podstaw warstwy obsługi wejścia. Zmiany te sprawiają, że wszystkie klawiatury wyglądają jak "normalne" klawiatury PC. Oznacza to, że jeśli używasz innego typu klawiatury (np. USB-MAC lub klawiatury Sun), jest bardzo możliwe, że po reboocie z nowym jądrem 2.6, Twoja klawiatura przestanie działać.

Jeśli masz połączenie SSH z innego systemu, możesz rozwiązać ten problem przez uruchomienie `dpkg-reconfigure console-data`, wybierając opcję "Select keymap from full list" i wybierając klawiaturę "pc".

Jeśli wspomniany problem dotyczy klawiatury konsoli, prawdopodobnie powinieneś dodatkowo zrekonfigurować klawiaturę w X Window System. Możesz zrobić to albo przez wykonanie `dpkg-reconfigure xserver-xorg`, albo bezpośrednio przez edycję pliku `/etc/X11/xorg.conf`. Nie zapomnij przeczytać odnośnej dokumentacji w 'Zadania przed restartem komputera' na stronie [30](#).

Zwróć uwagę, że jeśli używasz klawiatury USB, może ona być skonfigurowana albo jako "normalna" klawiatura PC lub jako klawiatura USB-MAC. Tego pierwszego przypadku problem ten nie dotyczy.

5.2.2 Konfiguracja myszy

Również przez zmiany w warstwie obsługi wejścia, może być konieczna rekonfiguracja X Window System i pakietu `gpm`, jeśli Twoja mysz przestała pracować po upgrade do jądra 2.6. Najczęstszą tego przyczyną jest zmiana nazwy urządzenia, które pobiera dane z myszy. Być może trzeba będzie załadować inne moduły.

5.2.3 Konfiguracja dźwięku

Dla jądra serii 2.6 sterowniki dźwięku ALSA są rekomendowane zamiast starszych OSS. Sterowniki ALSA są dostarczane domyślnie jako moduły. Żeby umożliwić działanie dźwięku, konieczne jest załadowanie odpowiednich dla Twojego sprzętu modułów ALSA. Zasadniczo odbywa się to w sposób automatyczny, jeśli oprócz pakietu `alsa-base` masz zainstalowany albo pakiet `hotplug` albo `discover`. Pakiet `alsa-base` "oczernia" moduły OSS, żeby

zablokować ich załadowanie przez `hotplug` lub `discover`. Jeśli w pliku `/etc/modules` masz moduły OSS, powinieneś je stamtąd usunąć.

5.3 Przejście z XFree86 na X.Org

Przejście na X.Org wprowadza pewne zmiany strukturalne. W przypadku, jeśli wszystkie zainstalowane pakiety pochodzą z Debiana i są włączone do etch, aktualizacja nie powinna nastęrczać problemów. Mamy jednak doświadczenia, że należy zwrócić uwagę na kilka zmian, które mogą potencjalnie spowodować pewne problemy w czasie upgrade'u.

Najważniejszą zmianą jest usunięcie katalogu `/usr/X11R6/bin` i pozostawienie w jego miejsce linku do `/usr/bin`. Oznacza to, że powinien on być pusty w czasie instalowania nowych pakietów. Nowe pakiety są w konflikcie z większością pakietów używanych w `/usr/X11R6/bin`, ale w niektórych przypadkach może być konieczna ręczna interwencja. Prosimy pamiętać o tym, żeby nie uruchamiać aktualizacji z sesji X'ów.

W przypadku niepowodzenia aktualizacji w czasie instalacji X.Org, powinieneś sprawdzić, czy pozostały jakieś pliki w katalogu `/usr/X11R6/bin`. Możesz użyć `dpkg -S`, żeby stwierdzić, który pakiet Debiana zainstalował dany plik (jeśli to był pakiet Debiana) i usunąć go za pomocą `dpkg --remove`. Prosimy zanotować, które pakiety usuwasz, aby zainstalować potem odpowiednie pakiety zastępujące. Zanim wznowisz aktualizację, wszystkie pliki w katalogu `/usr/X11R6/bin` powinny zostać usunięte.

Prosimy przeczytać <http://wiki.debian.org/Xorg69To7>, są tam opisane szczegóły i inne problemy.

If you experience problems with X.Org after restarting, it might be also worth to restart the font server by running `/etc/init.d/xfst restart`. This happens due to `/etc/X11/fs/xfst.options` containing a line with `no-restart-on-upgrade`, but the font paths have changed.

5.4 Brak wsparcia dla 8-bitowych wyświetlaczy w wielu aplikacjach

After the upgrade to the X.Org and the latest libraries, X terminals which can only represent colors 8 bits depth will not work. This is because the Cairo 2D vector graphics library (`libcairo2`) doesn't have 8-bit pseudocolor support. This library is used by the GNOME and Xfce desktops as well as by many desktop applications compiled with the Gtk2+ toolkit, such as `abiword`.

Systemy, których dotyczy ten problem to niektóre maszyny Sun oraz terminale Tektronix, NCD, IBM i SGI, jak też niektóre inne systemy zdalnych X-ów. Powinieneś skonfigurować te terminale tak, aby używały kolorów 16-bitowych, jeśli jest to możliwe.

Więcej informacji na ten temat jest dostępne na stronie błędów Freedesktop bug #4945 (https://bugs.freedesktop.org/show_bug.cgi?id=4945).

5.5 Upgrade z exim do exim4

Jednym z pakietów, który stał się pakietem przestarzałym w etch jest pakiet agenta transferu poczty (ang. Mail Transfer Agent - MTA) `exim`, który został zastąpiony całkowicie nowym pakietem `exim4`.

`exim` (wersja 3.xx) nie był pielęgnowany przez autorów już od lat, Debian również zaprzestał wspierania tej wersji. Jeśli ciągle używasz `exim`'a 3.xx, prosimy zaktualizować pakiet `exim` do `exim4` ręcznie. Od momentu, gdy `exim4` jest częścią `sarge`, możesz wybrać jego aktualizację przed lub po upgrade do etch. Pamiętaj tylko, że stary `exim` nie będzie już aktualizowany i nie będzie dla niego poprawek bezpieczeństwa po wygaśnięciu `sarge`.

W zależności od konfiguracji pakietu `debconf`, w czasie instalacji pakietu `exim4` możesz nie zostać zapytany o jego parametry konfiguracyjne. Jeśli nie zostaniesz o nic zapytany, system zostanie skonfigurowany domyślnie jako 'dostarczanie lokalne'. Konfiguracja jest możliwa za pomocą polecenia `dpkg-reconfigure exim4-config`.

Pakiety wchodzące w skład `exim4` w Debianie są dobrze udokumentowane. Strona domowa pakietu to <http://wiki.debian.org/PkgExim4> na Debian Wiki i plik `README`, który można znaleźć na <http://pkg-exim4.alioth.debian.org/README/README.Debian.html> i w samym pakiecie.

Plik `README` zawiera rozdział o sposobie pakowania, który opisuje różne dostępne możliwości, które oferujemy, oraz rozdział o aktualizacji z `Exim 3`, który pomoże Ci w przejściu na aktualną wersję.

5.6 Aktualizacja apache2

Apache został zaktualizowany do nowej wersji 2.2. Chociaż nie powinno to mieć większego wpływu dla większości użytkowników, zwracamy uwagę na kilka potencjalnych problemów.

<http://httpd.apache.org/docs/2.2/upgrading.html> zawiera zmiany opisane przez autorów. Prosimy przeczytać i pamiętać szczególnie że:

- wszystkie moduły powinny zostać zrekompilowane
- moduły autoryzacyjne zostały zrewidowane i przenazwane
- niektóre opcje konfiguracyjne mają inne nazwy

Zmianą specyficzną dla Debiana jest usunięcie definicji napisu SSL, gdyż `ssl` jest teraz włączony domyślnie.

Jeśli używasz eksperymentalnego ITK MPM (z pakietu `apache2-mpm-itk`), moduł `cgi` nie zostanie domyślnie prawidłowo uruchomiony. Aby włączyć go poprawnie, musisz ręcznie wyłączyć `mod_cgid` i włączyć `mod_cgi`:

```
# cd /etc/apache2/mods-enabled
# rm cgid.conf cgid.load
# ln -s ../mods-available/cgi.load .
# /etc/init.d/apache2 force-reload
```

5.7 Upgrade Zope i Plone

Zope i jego pochodne zostały zaktualizowane. Wiele z nich zostało usuniętych z dystrybucji (czy to z powodu nieaktualności, czy z to z powodu braku kompatybilności z nowymi wersjami Zope, CMF lub Plone).

Niestety nie możemy zagwarantować, że przejście na nowe wersje `zope` czy `plone` będzie proste. Nawet pomimo tego, że Plone zawiera narzędzie migracyjne, doświadczenie pokazuje, że automatyczna migracja nie zawsze musi się powieść.

Z tego powodu rekomendujemy w taki sposób skonfigurować system, żeby pracować nadal na wersjach Zope/Plone z `sarge`, a równoległe przeprowadzić test migracji na wersje z `etch`.

Najłatwiej i najbezpieczniej osiągnąć to poprzez wykonanie kopii `sarge` na inny dysk lub partycję i wykonać upgrade tylko jednej kopii. Można wtedy użyć `chroot` do uruchamiania wersji z `sarge` równoległe z wersją z `etch`.

Nie jest możliwe posiadanie starej i nowej wersji Zope/Plone zainstalowanych jednocześnie na `etch`, m. in. ze względu na to, że stare pakiety zależą od pakietu `python2.3`, który nie może być zainstalowany jednocześnie z pakietem `python2.4`.

5.8 Rozszerzanie symboli (wildcard expansion - globbing) w GNU tar

Poprzednie wersje GNU `tar` stosowały globbing analogiczny, jak w powłokach systemowych do rozpakowywania (ekstrakcji) plików z archiwum. Na przykład:

```
tar xf foo.tar '*.c'
```

wyekstrahowywało wszystkie pliki, których nazwy kończą się na `'.c'`. Takie zachowanie było nie udokumentowane i niekompatybilne z tradycyjnymi implementacjami programu `tar`. Z tego powodu poczynając od wersji 1.15.91, GNU `tar` nie używa domyślnie globbingu. Powyższy przykład zostanie zinterpretowany jako żądanie wyekstrahowania pliku o nazwie `'*.c'`.

Więcej informacji w dokumentacji `/usr/share/doc/tar/NEWS.gz`.

5.9 NIS i Network Manager

Wersja `ybind` zawarta w pakiecie `nis` dla `etch` zawiera wsparcie dla Network Managera. Powoduje to, że `ybind` wyłącza funkcjonowanie klienta NIS, jeśli Network Manager stwierdzi, że komputer jest odłączony od sieci. Ponieważ Network Manager zwykle stwierdza to, kiedy komputer nie jest używany, użytkownicy komputerów z klientami NIS powinni wyłączyć Network Managera w swoich systemach.

Można to zrobić przez deinstalację pakietu `network-manager` lub edycję pliku `/etc/default/nis` dodając `-no-dbus` do `YPBINDARGS`.

W nowych instalacjach Debiana jest używane domyślnie `-no-dbus`, ale nie działa się tak w poprzednich wydaniach.

5.10 Przeszarżałe niezabezpieczone konfiguracje php

Przez wiele lat włączenie opcji `register_globals` w PHP było znane jako nie zabezpieczone i niebezpieczne, choć opcja ta była do tej pory domyślnie wyłączona. Obecnie ten parametr jest całkowicie przeszarżały w Debianie, jako zbyt niebezpieczny. To samo dotyczy parametrów `safe_mode` i `open_basedir`, które od pewnego czasu również nie były wspierane.

Poczynając od tego wydania, zespół bezpieczeństwa Debiana nie zapewnia wsparcia dla tych konfiguracji PHP, które są znane jako niezabezpieczone. Najważniejszy jest całkowity brak wsparcia dla błędów spowodowanych włączoną opcją `register_globals`.

Jeśli używasz działających aplikacji, które wymagają włączonego `register_globals`, włącz je w sposób dozwolony, czyli poprzez plik konfiguracyjny Apache'a. Więcej informacji jest dostępnych w pliku `README.Debian.security` w katalogu z dokumentacją PHP (`/usr/share/doc/php4`, `/usr/share/doc/php5`).

5.11 Status bezpieczeństwa produktów Mozilla

Programy Mozilli `firefox` i `thunderbird` (przemianowane w Debianie odpowiednio na `iceweasel` i `icedove`), są dla wielu użytkowników ważnymi narzędziami. Niestety, polityka bezpieczeństwa autorów, która nakłania użytkowników do aktualizowania programów do autorskich wersji, jest w konflikcie z polityką Debiana, która zakłada niełączenie zmian funkcjonalnych z poprawkami bezpieczeństwa. Nie możemy tego dziś przewidzieć, ale być może w czasie życia `etch` zespół bezpieczeństwa Debiana dojdzie do punktu, w którym wsparcie produktów Mozilli nie będzie możliwe i zostanie ogłoszone jego zakończenie. Powinieneś wziąć to pod uwagę, jeśli używasz produktów Mozilli i rozważyć alternatywne rozwiązania dostępne w Debianie, jeśli brak wsparcia bezpieczeństwa stanowi dla Ciebie problem.

5.12 Środowisko KDE

Obsługa mediów w KDE została zmieniona w wersji dostępnej w etch z użycia `device:/` na `media:/`. Niektóre konfiguracje użytkowników mogą zachowywać linki `device:/` w plikach, co powinno zostać poprawione. Szczególnie plik `~/.kde/share/apps/konqsidebar.tng/virtual_folders/services` zawierający takie odnośniki może być bezpiecznie usunięty i nie będzie już tworzony dla nowych użytkowników.

Dokonano wielu zmian w KDE od wersji dostarczanej z sarge do wersji w etch, możesz znaleźć więcej informacji na KDE 3.5 Release Notes (<http://www.kde.org/announcements/announce-3.5.php>).

5.13 Zmiany w środowisku GNOME

Jeśli używałeś GNOME w sarge, nie odniesiesz wszystkich korzyści z niektórych zmian wprowadzonych do standardowej konfiguracji Debiana dla etch. W niektórych skrajnych przypadkach GNOME może nie obsługiwać prawidłowo dotychczasowej konfiguracji i nie zachowywać się poprawnie.

Jeśli nie zależy Ci zbyt na dotychczasowej konfiguracji GNOME, możesz zmienić nazwę katalogu `.gconf` w katalogu domowym (np. na `.gconf.old`) i zostanie on utworzony automatycznie z domyślną konfiguracją dla etch przy starcie nowej sesji.

W wydaniu etch Debian nie zachowuje pakietów z przestarzałej wersji 1 GNOME. Pomimo tego pozostawiono niektóre pakiety niezbędne do działania starszych pakietów nie zaktualizowanych jeszcze do GNOME2. Pakiety dla GTK1.2 są w pełni wspierane.

Dokonano wielu zmian w GNOME pomiędzy wersjami dostarczonymi w sarge i etch, możesz znaleźć więcej informacji na ten temat w GNOME 2.14 Release Notes (<http://www.gnome.org/start/2.14/notes/en/>).

5.14 Domyślny edytor

Jeśli używasz `vim` jako domyślnego edytora, może on zmienić się na `nano` w czasie aktualizacji.

Administratorzy, którzy chcą zmienić domyślny edytor wszystkim użytkownikom, będą musieli zaktualizować alternatywy systemowe poleceniem:

```
# update-alternatives --config editor
```

Użytkownicy chcący zmienić domyślny edytor mogą zdefiniować zmienną środowiskową `EDITOR` przez poprawienie pliku swojego profilu:


```
EDITOR=vi
export EDITOR
alias editor=$EDITOR
```

5.15 Message of the day

`/etc/motd` is now a symlink to `/var/run/motd` which is rebuilt by `/etc/init.d/bootmisc.sh` from a template, `/etc/motd.tail`, at each reboot. It means that changes made to `/etc/motd` will be lost. Changes made into `/etc/motd.tail` are not automatically applied to `/etc/motd` other than at reboot.

Also, the `EDITMOTD` variable at `/etc/default/rcS` no longer has any effect. If you wish to disable updating of the motd, or want to maintain your own content for the message of the day you just have to point the `/etc/motd` symlink to a different file such as `/etc/motd.static` and make your changes there.

5.16 Not default support for unicode in emacs21*

Emacs21 and emacs21-nox are not configured to use Unicode by default. For more information and a workaround please see Bug #419490 (<http://bugs.debian.org/419490>).

Rozdział 6

Więcej informacji o Debian GNU/Linux

6.1 Co czytać dalej

Oprócz niniejszych Uwag do wydania i podręcznika instalacji, dalsza dokumentacja nt. Debian GNU/Linux jest dostępna w Debian Documentation Project (DDP), którego celem jest tworzenie dokumentacji o wysokiej jakości dla użytkowników i deweloperów Debiana. Dokumentacja ta zawiera Debian Reference, Debian New Maintainers Guide, Debian FAQ i wiele innych. Szczegóły nt. dostępnych zasobów są dostępne na stronach DDP (<http://www.debian.org/doc/ddp>).

Dokumentacja każdego z pakietów jest instalowana w katalogu `/usr/share/doc/pakiet`. Może ona zawierać informacje o prawach autorskich, szczegóły specyficzne dla Debiana i dokumentację źródłową (upstream) pakietu.

6.2 Pomoc

Jest wiele źródeł pomocy, porad i wsparcia dla użytkowników Debiana, ale należy po nią sięgać po stwierdzeniu, że dołączona dokumentacja nie zawiera rozwiązania problemu. Ta sekcja jest krótkim wprowadzeniem do źródeł najbardziej przydatnych dla nowych użytkowników Debiana.

6.2.1 Listy dyskusyjne

Najbardziej interesujące dla użytkowników są listy `debian-user` (po angielsku) i `debian-user-język` (dla innych języków; `debian-user-polish` dla polskiego). Więcej informacji na temat list i sposobu subskrypcji opisano na <http://lists.debian.org/>. Prosimy sprawdzać archiwa, zanim zadasz pytanie i stosować się do zwyczajów obowiązujących na listach dyskusyjnych.

6.2.2 Internet Relay Chat (IRC)

Debian ma kanał IRC przeznaczony do wsparcia i pomocy dla użytkowników zlokalizowany na OFTC IRC network. Żeby wejść na kanał, wskaż swojemu programowi IRC serwer `irc.debian.org` i przyłącz się do kanału `#debian`.

Prosimy stosować się do reguł kanału, mając na względzie innych użytkowników. Wytyczne są dostępne na Debian Wiki (<http://wiki.debian.org/DebianIRC>).

Więcej informacji nt. OFTC na tej stronie (<http://www.oftc.net/>).

6.3 Zgłaszanie błędów

Staramy się utrzymać jak najwyższą jakość systemu Debian GNU/Linux, tym niemniej nie oznacza to, że pakiety są całkowicie wolne od błędów. Będąc w zgodzie z filozofią "otwartego oprogramowania" Debiana i służąc naszym użytkownikom, udostępniamy wszystkie informacje o błędach na naszym własnym Systemie Śledzenia Błędów (Bug Tracking System - BTS). BTS można przeglądać na stronie `bugs.debian.org` (<http://bugs.debian.org/>).

Jeśli znajdziesz błąd w dystrybucji lub którymś z pakietów, będącym jej częścią, prosimy przesłać raport, który umożliwi naprawienie błędu w następnej wersji. Wysyłając raport trzeba podać prawidłowy adres email. Prosimy o to ze względu na śledzenie błędów, pozwala to na kontakt deweloperów z użytkownikami i możliwość zadawania dodatkowych pytań.

You can submit a bug report using the program `reportbug` or manually using email. You can read more about the Bug Tracking System and how to use it by reading the reference cards (available at `/usr/share/doc/debian` if you have `doc-debian` installed) or online at the Bug Tracking System (<http://bugs.debian.org/>).

6.4 Rozwijanie Debiana

Nie musisz być ekspertem, aby pomóc w rozwoju Debiana. Poprzez pomoc innym użytkownikom na listach (<http://lists.debian.org/>), również przyczyniasz się do rozwoju społeczności. Identyfikowanie (i rozwiązywanie) problemów związanych z programowaniem przez uczestnictwo na listach (<http://lists.debian.org/>) deweloperów jest również niezwykle pomocne. Do zapewnienia wysokiej jakości dystrybucji Debiana wskazane jest zgłaszanie błędów (<http://bugs.debian.org/>) i pomoc deweloperom w ich śledzeniu i usuwaniu. Jeśli masz swobodę w posługiwaniu się słowem, możesz rozwijać system bardziej aktywnie przez pomoc w pisaniu dokumentacji (<http://www.debian.org/doc/ddp>) lub tłumaczeniu (<http://www.debian.org/international/>) istniejących dokumentów na swój język.

Jeśli masz więcej czasu, możesz zajmować się jakąś częścią Wolnego Oprogramowania w Debianie. Potrzebna jest zwłaszcza pomoc w adoptowaniu i opiece nad programami, które użytkownicy chcą widzieć w Debianie. Szczegóły dostępne są na stronie Baza pakietów rozwojowych i wymagających pracy (<http://www.debian.org/devel/wnpp/>). Jeśli masz

jakieś specyficzne zainteresowania, być może zechcesz wziąć udział w rozwoju niektórych podprojektów Debiana, m. in. adaptacji dla różnych architektur, Debian Jr. (<http://www.debian.org/devel/debian-jr/>) lub Debian Med (<http://www.debian.org/devel/debian-med/>).

W każdym przypadku, pracując w społeczności wolnego oprogramowania, czy to jako użytkownik, programista, pisarz czy tłumacz, przyczyniasz się do wspólnego dzieła wolnego oprogramowania. Przynosi to satysfakcję i przyjemność, powala też spotkać nowych ludzi i daje wiele pozytywnych emocji.

Dodatek A

Zarządzanie systemem sarge

Ten dodatek zawiera informacje o niezbędnych do zainstalowania lub aktualizacji pakietach w systemie sarge przed przejściem na etch. Jest to niezbędne tylko w niektórych, specyficznych sytuacjach.

A.1 Aktualizacja systemu sarge

Zasadniczo nie ma różnicy w każdym przypadku upgrade'u z sarge. Chodzi o upewnienie się, że pakiety, których używasz znajdują się na liście pakietów sarge, jak to opisano w 'Sprawdzenie listy źródeł' na tej samej stronie.

Jeśli dokonujesz aktualizacji przy użyciu mirrora Debiana, zostanie ona automatycznie wykonana do ostatniej, najnowszej wersji sarge.

A.2 Sprawdzenie listy źródeł

Jeśli jakaś linia w pliku `/etc/apt/sources.list` odnosi się do 'stable', oznacza to że faktycznie "używasz" już etch. Jeśli wykonałeś już `apt-get update`, możesz powrócić jeszcze bez problemów przy pomocy poniższej procedury.

Jeśli masz już zainstalowane pakiety z etch, zapewne nie ma już sensu instalowanie pakietów z sarge. Musisz zdecydować, czy chcesz kontynuować, czy nie. Jest możliwy downgrade pakietów, ale nie jest tu opisany.

Otwórz plik `/etc/apt/sources.list` w edytorze (jako root) i sprawdź linie zaczynające się od `deb http:` lub `deb ftp:`, które odnoszą się do "stable". Jeśli są jakieś, zmień słowo `stable` na `sarge`.

Jeśli są jakieś linie zaczynające się od `deb file:`, musisz sprawdzić samodzielnie, czy wskazują one na pakiety z archiwów sarge, czy z etch.

UWAGA! Prosimy nie zmieniać linii zaczynających się od `deb cdrom:`. Zmiana spowoduje, że staną się one nieprawidłowe i może być konieczne ponowne uruchomienie programu `apt-cdrom`. Nie przejmuj się, jeśli linie 'cdrom' odwołują się do "unstable". Pomimo swej dziwności jest to normalne.

Po dokonaniu zmian zapisz plik i wykonaj

```
# apt-get update
```

aby odświeżyć listę pakietów.