

# Информация о выпуске Debian GNU/Linux 4.0 (“etch”) для архитектуры AMD64

Josip Rodin, Bob Hilliard, Adam Di Carlo, Anne Bezemer, Rob Bradford, Frans Pop  
(текущее издание) Andreas Barth (текущее издание), Javier Fernández-Sanguino  
Peña (текущее издание), Steve Langasek (текущее издание)  
<debian-doc@lists.debian.org>

---

# Оглавление

1	Введение	1
1.1	Как сообщить об ошибке в этом документе	1
1.2	Предоставление отчёта об обновлении	2
1.3	Исходный текст этого документа	2
2	Что нового в Debian GNU/Linux 4.0	3
2.1	Что нового в дистрибутиве?	4
2.1.1	Управление пакетами	5
2.1.2	Сервис <code>debian-volatile</code> теперь предоставляется официально	5
2.2	Улучшения в системе	6
2.3	Основные изменения в ядре	7
2.3.1	Изменения в пакетировании ядра	7
2.3.2	Новые утилиты для генерации <code>initrd</code>	8
2.3.3	Динамическое управление <code>/dev</code> и обнаружение оборудования	8
3	Система установки	11
3.1	Что нового в системе установки?	11
3.1.1	Главные изменения	11
3.1.2	Автоматизируемая установка	14
3.2	Частота использования пакетов	14
4	Обновление с предыдущих выпусков	15
4.1	Подготовка к обновлению	15
4.1.1	Создание резервной копии всех данных и настроек	15
4.1.2	Предварительное информирование пользователей	16

---

4.1.3	Подготовка к откату	16
4.1.4	Подготовка безопасного окружения для обновления	17
4.1.5	Прекращение поддержки ядер версий 2.2	17
4.2	Проверка состояния системы	17
4.2.1	Проверка отложенных действий менеджера пакетов	18
4.2.2	Отключение фиксации АРТ	18
4.2.3	Проверка состояния пакетов	18
4.2.4	Неофициальные источники и адаптации (backports)	19
4.3	Снятие пометки с пакетов вручную	20
4.4	Подготовка источников АРТ	20
4.4.1	Как добавить Интернет-источник АРТ	20
4.4.2	Как добавить локальное зеркало АРТ	21
4.4.3	Как добавить источник АРТ на компакт-диске или DVD	22
4.5	Обновление пакетов	22
4.5.1	Запись сессии	23
4.5.2	Обновление списка пакетов	23
4.5.3	Проверка достаточности места для обновления	23
4.5.4	Минимальное обновление системы	25
4.5.5	Обновление ядра	26
4.5.6	Обновление остальной системы	27
4.5.7	Получение подписей пакетов	28
4.5.8	Возможные проблемы во время обновления	28
4.6	Обновление ядра и сопутствующих пакетов	29
4.6.1	Установка ядерного метапакета	30
4.6.2	Обновление ядра версии 2.6	30
4.6.3	Переопределение нумерации устройств	31
4.6.4	Проблемы с синхронизацией при загрузке	32
4.7	Перед перезагрузкой	32
4.7.1	Переход с devfs	32
4.7.2	Перезапуск lilo	32
4.7.3	Обновление mdadm	33

---

4.8	Подготовка к следующему выпуску . . . . .	33
4.9	Исключаемые пакеты . . . . .	34
4.10	Устаревшие пакеты . . . . .	34
4.10.1	Пакеты-пустышки . . . . .	35
5	Что надо знать о etch . . . . .	37
5.1	Возможные проблемы . . . . .	37
5.1.1	Проблемы с устройствами из-за udev . . . . .	37
5.1.2	Некоторые сайты в сети недоступны по TCP . . . . .	37
5.1.3	Медленно обновляются пакетные индексные файлы APT . . . . .	38
5.1.4	Асинхронная инициализация сети может приводить к непредсказуемому поведению . . . . .	38
5.1.5	Проблема при использовании WPA-безопасности в беспроводных сетях . . . . .	38
5.1.6	Проблемы с не-ASCII символами в именах файлов . . . . .	38
5.1.7	Повреждение данных в оборудовании IOMMU на чипсетах Nvidia . . . . .	39
5.1.8	Перестал работать звук . . . . .	39
5.2	Переход с XFree86 на X.Org . . . . .	39
5.3	Во многих приложениях нет поддержки 8-битного отображения цветов . . . . .	40
5.4	Переход с exim на exim4 . . . . .	40
5.5	Обновление apache2 . . . . .	41
5.6	Обновление Zope и Plone . . . . .	41
5.7	Раскрытие шаблонов (подстановка) в GNU tar . . . . .	42
5.8	NIS и Network Manager . . . . .	42
5.9	Запрещены небезопасные конфигурации php . . . . .	42
5.10	Безопасность в продуктах Mozilla . . . . .	43
5.11	Рабочий стол KDE . . . . .	43
5.12	Поддержка и изменения рабочего стола GNOME . . . . .	43
5.13	Редактор по умолчанию . . . . .	44
5.14	Сообщение дня . . . . .	44
5.15	Нет поддержки unicode в emacs21* по умолчанию . . . . .	45

---

6	Дополнительная информация о Debian GNU/Linux	47
6.1	Что ещё можно прочитать . . . . .	47
6.2	Если нужна помощь . . . . .	47
6.2.1	Списки рассылки . . . . .	47
6.2.2	IRC . . . . .	48
6.3	Как сообщить об ошибке . . . . .	48
6.4	Как помочь Debian . . . . .	48
A	Управление системой sarge	51
A.1	Обновление системы sarge . . . . .	51
A.2	Проверка списка источников АРТ . . . . .	51

# Глава 1

## Введение

Основной целью информации о выпуске является информирование пользователей об основных изменениях в этом выпуске дистрибутива Debian GNU/Linux, о том, как безопасно обновиться с предыдущей версии и, наконец, сообщить пользователям об известных потенциальных проблемах, с которыми они могут столкнуться при обновлении или использовании выпуска `etch`.

Заметим, что здесь невозможно перечислить все известные проблемы и поэтому были выбраны только те, которые, как ожидается, могут возникнуть у большинства людей, и которые оказывают наибольшее влияние в целом.

Самая новая версия этого документа всегда доступна по адресу <http://www.debian.org/releases/stable/releasenotes>. Если версия, которую вы читаете, выпущена больше месяца назад<sup>1</sup>, стоит загрузить более новую.

Обратите внимание, что теперь поддерживается и описывается только обновление с предыдущего выпуска Debian (то есть этот документ описывает обновление с `sarge`). Если вам требуется обновить систему с более старого выпуска, обратитесь к ранним редакциям этого документа или сначала выполните обновление до `sarge`.

### 1.1 Как сообщить об ошибке в этом документе

Мы пытались провести все возможные шаги обновления, описанные в этом документе, а также попытались предвидеть все возможные проблемы, с которыми могут столкнуться пользователи.

Тем не менее, если вы думаете, что нашли ошибку в этом описании (есть неправильная или отсутствующая информация), пожалуйста, отправьте сообщение об ошибке в пакете `release-notes` в систему отслеживания ошибок (<http://bugs.debian.org/>).

---

<sup>1</sup>Дата проставлена на первой странице документа в формате PDF и в конце страницы оглавления онлайн версии в формате HTML

## 1.2 Предоставление отчёта об обновлении

Мы рады любой информации о проведённом обновлении с sarge до etch от пользователей. Если вы хотите поделиться своим опытом, отправьте сообщение об ошибке в пакете release-notes в систему отслеживания ошибок (<http://bugs.debian.org/>) описав результаты. Пожалуйста, сжимайте все прикладываемые к сообщению файлы (с помощью gzip).

Пожалуйста включите в отправляемое сообщение следующую информацию:

- Состояние базы данных пакетов до и после обновления: база данных dpkg содержится в файле `/var/lib/dpkg/status`, а aptitude-информация о состоянии пакетов содержится в файле `/var/lib/aptitude/pkgstates`. Перед обновлением выполните резервное копирование согласно описанию ‘Создание резервной копии всех данных и настроек’ на стр. 15, помимо этого, вы можете найти резервные копии данной информации в каталоге `/var/backups`.
- Журналы сеанса работы с помощью script, как это описано в ‘Запись сессии’ на стр. 23.
- Файлы журналов aptitude из каталога `/var/log/aptitude`.

Замечание: обязательно просмотрите и удалите всю личную и/или конфиденциальную информацию из журналов перед тем как включать их в сообщение об ошибке, так как эта информация будет доступна в публичной базе данных.

## 1.3 Исходный текст этого документа

Этот документ создан с помощью пакета debiandoc-sgml. Исходный текст документа доступен в CVS проекта документирования Debian. Для доступа к отдельным файлам и для просмотра изменений вы можете использовать веб интерфейс (<http://cvs.debian.org/ddp/manuals.sgml/release-notes/?root=debian-doc>). Подробнее о доступе к CVS описано на страницах CVS проекта документирования Debian (<http://www.debian.org/doc/cvs>).

## Глава 2

# Что нового в Debian GNU/Linux 4.0

В этот выпуск добавлена официальная поддержка архитектуры AMD64, в которую входят 64-битные процессоры Intel (EM64T) и AMD (AMD64). Для предыдущего выпуска Debian GNU/Linux 3.1 ('sarge') была доступна неофициальная версия переноса. Обновление с неофициальной версии возможно с помощью этой информации о выпуске, но официально Debian оно не поддерживается.

Официальная поддержка архитектуры Motorola 680x0 ('m68k') прекращена, так как перестала удовлетворять критериям, установленным менеджерами выпуска Debian. Основными причинами являются производительность и ограниченная поддержка со стороны разработчиков компонент, необходимых для пересборки (toolchain). Однако, перенос m68k предполагается оставить активным и он будет доступен для установки, даже если не войдёт в состав официального стабильного выпуска.

Список официально поддерживаемых архитектур для Debian GNU/Linux etch:

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



Более подробную информацию о переносе на различные архитектуры и сведения, относящиеся к конкретным переносам, вы можете получить на странице переносов Debian (<http://www.debian.org/ports/amd64/>).

## 2.1 Что нового в дистрибутиве?

Новый выпуск Debian опять содержит намного больше программ, чем предшественник, sarge. Дистрибутив включает в себя более 6500 новых пакетов, всего их стало более 18200. Большая часть программ, входящих в дистрибутив, была обновлена: более 10700 пакетов (что составляет 68% дистрибутива sarge). Также, по различным причинам многие пакеты (более 3500, 23% дистрибутива sarge) были удалены из дистрибутива. Эти пакеты не будут обновлены, в программах управления пакетами они будут помечены как “устаревшие” (obsolete).

В этом выпуске Debian GNU/Linux был осуществлён переход с XFree86 на X.Org версии 7.1, что дало поддержку более широкого спектра аппаратного обеспечения и лучшее автоопределение устройств. Теперь можно использовать Compiz, один из первых compositing window managers для X Window System, что позволяет воспользоваться преимуществами аппаратного OpenGL-ускорения в поддерживающих его устройствах.

Debian GNU/Linux снова распространяется с несколькими типами рабочих столов и окружениями. Помимо прочих, теперь включены окружения рабочего стола GNOME 2.14<sup>1</sup>, KDE 3.5.5a и Xfce 4.4. Обновлены офисные приложения: пакеты OpenOffice.org 2.0.4a и KOffice 1.6, а также GNUcash 2.0.5, GNUMeric 1.6.3 и Abiword 2.4.6.

Обновлены другие приложения рабочего стола: Evolution 2.6.3 и Gaim 2.0. Пакет Mozilla также был обновлён, но его основные программы были переименованы: iceweasel (версии 2.0.0.2) — это веб-браузер Firefox без бренда и icedove (версии 1.5) — это почтовый клиент Thunderbird без бренда.

В этом выпуске, среди прочего, было обновлено следующее программное обеспечение:

- библиотека GNU C, версия 2.3.6
- компилятором по умолчанию является GNU Compiler Collection 4.1
- интерпретаторы языков: Python 2.4, PHP 5.2
- серверное ПО:
  - почтовые серверы: Exim 4.63 (почтовый сервер по умолчанию при установке “с нуля”), Postfix 2.3, Courier 0.53, Cyrus 2.2
  - веб-серверы: Apache 2.2, fnord 1.10
  - серверы баз данных: MySQL 5.0.32, PostgreSQL 8.1
  - сервер OpenSSH, версия 4.3

---

<sup>1</sup>С несколькими модулями из GNOME 2.16.

- серверы имён: Bind 9.3, namedns 1.2
- сервер каталогов: OpenLDAP 2.3

Официальный дистрибутив Debian GNU/Linux теперь занимает от 19 до 23 компакт-дисков (в зависимости от архитектуры) с собранными пакетами, приблизительно столько же дисков занимают пакеты с исходными кодами. Также доступна версия дистрибутива на DVD.

### 2.1.1 Управление пакетами

Рекомендуемой программой управления пакетами из консоли является aptitude. aptitude поддерживает большую часть операций командной строки apt-get и гораздо лучше разрешает зависимости. Если вы всё ещё пользуетесь dselect, то вам нужно перейти на официальный интерфейс управления пакетами aptitude.

Для etch в aptitude реализован более интеллектуальный механизм разрешения конфликтов, который пытается найти лучшее решение при обнаружении конфликта в зависимостях между пакетами.

Защищённый АРТ теперь входит в etch. Это свойство добавляет дополнительную безопасность в системы Debian GNU/Linux через проверку цифровых подписей скачиваемых пакетов и стойкого шифрования. Данный выпуск содержит программу apt-key, которая позволяет добавлять новые ключи в связку ключей apt, по умолчанию содержащую только действующий ключ подписания архива Debian, предоставляемый пакетом debian-archive-keyring.

По умолчанию apt будет выдавать предупреждение, если пакеты скачиваются из не аутентифицированных источников. В будущих выпусках возможно будет включена проверка всех пакетов перед их загрузкой. Администраторам неофициальных источников apt предлагается создать криптографический ключ и подписать им свои файлы Release, а также предоставить безопасный способ распространения своих общих ключей.

Подробная информация дана на странице руководства по apt(8), в главе Подписание пакетов в Debian (<http://www.debian.org/doc/manuals/securing-debian-howto/ch7#s-deb-pack-sign>) руководства по безопасности Debian и в вики (<http://wiki.debian.org/SecureApt>).

В apt было добавлено ещё одно свойство: способность скачивать только изменения в файлах Packages, произошедших с последнего обновления. Подробности по адресу ‘Медленно обновляются пакетные индексные файлы АРТ’ на стр. 38.

### 2.1.2 Сервис debian-volatile теперь предоставляется официально

Неофициальный сервис debian-volatile, появившийся в выпуске sarge, теперь стал официальным сервисом Debian GNU/Linux.

Это означает, что теперь используется адрес [.debian.org](http://www.debian.org)<sup>2</sup>. Если вы уже пользуетесь этим сервисом, исправьте настройку в файле `/etc/apt/sources.list`.

---

<sup>2</sup>Старый адрес [.debian.net](http://www.debian.net) также останется рабочим на некоторое время.

Сервис `debian-volatile` позволяет пользователям легко обновлять те стабильные пакеты, которые содержат быстро устаревающую информацию. В их число, например, входят антивирусные сигнатуры и наборы шаблонов для спам-фильтров. Дополнительную информацию и список сайтов-зеркал можно найти на странице архива (<http://volatile.debian.org/>).

## 2.2 Улучшения в системе

В дистрибутиве есть некоторые изменения, выгода от которых видна только при установке новой системы `etch`, и они не могут применится автоматически при обновлении с `sarge`. Данный раздел кратко описывает большинство таких изменений.

Понижен приоритет основных пакетов разработчика. Некоторые пакеты разработчика, которые имели приоритет стандартный теперь имеют приоритет необязательный, означающий, что они теперь не устанавливаются по умолчанию. Это относится к стандартному компилятору `C/C++`, `gcc`, а также к другому ПО (`dpkg-dev`, `flex`, `make`) и заголовочным файлам для разработки (`libc6-dev`, `linux-kernel-headers`).

Если вы хотите установить все эти пакеты в систему, самым простым способом является установка пакета `build-essential`, что повлечёт за собой установку большинства из них.

`SELinux` имеет стандартный приоритет, но по умолчанию не активирован. Необходимым для поддержки `SELinux` пакетам был присвоен приоритет стандарт. Это означает, что они будут установлены по умолчанию в новых установках. В существующих системах вы можете установить `SELinux` командой:

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

Заметим, что поддержка `SELinux` по умолчанию выключена. Информацию по активации и настройке можно найти на страницах Debian вики (<http://wiki.debian.org/SELinux>).

Новый супердемон `inet` по умолчанию. Супердемоном `inet` по умолчанию в `etch` является `openbsd-inetd` вместо `netkit-inetd`. Он не будет запущен, если нет настроенных сервисов (по умолчанию не настроен ни один). Новая версия по умолчанию будет установлена при обновлении автоматически.

Изменён клон `vi` по умолчанию. Устанавливаемый теперь по умолчанию клон `vi` является компактной версией `vim` (`vim-tiny`) вместо `nvim`.

Изменения в свойствах по умолчанию для `ext2/ext3`. Новые файловые системы `ext2` и `ext3` по умолчанию создаются со свойствами `dir_index` и `resize_inode`. Первое свойство ускоряет операции над каталогами с большим количеством файлов, а второе делает возможным изменение раздела файловой системы на лету (то есть без размонтирования).

Пользователи, обновляющиеся с `sarge`, могут установить флаги `dir_index` вручную с помощью `tune2fs`<sup>3</sup>; флаг `reset_inode` не может быть добавлен в существующую файловую систему. Для проверки, установлен ли этот флаг в файловой системе, можно воспользоваться командой `dumpe2fs -h`.

Кодировкой по умолчанию в `etch` является UTF-8. Кодировкой по умолчанию в новых установках Debian GNU/Linux является UTF-8. Некоторые приложения также будут настроены на использование UTF-8 по умолчанию.

Пользователи, обновляющиеся с `etch`, которые хотят перейти на UTF-8, могут перенастроить своё окружение и локаль. Глобальные настройки в системе можно изменить командой `dpkg-reconfigure locales`; сначала выберите локаль UTF-8 для вашего языка и страну, а затем установите это значение по умолчанию. Заметим, что переход на UTF-8 означает, что вам, вероятно, также придётся переконвертировать существующие файлы из старой кодировки в UTF-8.

В пакете `utf8-migration-tool` содержится инструмент, который может помочь с переходом, однако этот пакет доступен только пока в нестабильном дистрибутиве, так как был не готов для `etch`. Перед использованием программы настоятельно рекомендуется сделать резервную копию данных и конфигурационных файлов.

Заметим, что пока не все приложения корректно работают в окружении UTF-8 (в основном проблемы с отображением данных).

Более подробную информацию смотрите на страницах Debian вики (<http://wiki.debian.org/Sarge2EtchUpgrade>), где описана процедура обновления с `sarge` до `etch`.

## 2.3 Основные изменения в ядре

В Debian GNU/Linux 4.0 включено ядро версии 2.6.18 для всех архитектур; выпуск пока почти совместим с ядрами 2.4, но Debian больше не предоставляет или поддерживает пакеты с ядрами 2.4.

В самом ядре и в пакетировании ядра для Debian произошли большие изменения. Некоторые из них усложнили процедуру обновления, и это может привести к проблемам при перезагрузке системы после обновления до `etch`. В этом разделе кратко описаны наиболее важные изменения; возможные проблемы и инструкции как обойти их даны позже в следующих главах.

### 2.3.1 Изменения в пакетировании ядра

Пакеты с ядрами переименованы. Все пакеты с ядрами Linux были переименованы из `kernel-*` в `linux-*` для большей ясности в именах. Это в будущем облегчит включение не-Linux ядер в Debian.

---

<sup>3</sup>Флаг `filetype` скорее всего уже установлен на большинстве файловых систем, кроме, возможно, систем, установленных ранее `sarge`.

Одно общее ядро для AMD64 В sarge для каждого семейства процессоров этой архитектуры были включены несколько вариантов ядер. Так как ядро теперь само автоматически оптимизируется под процессор(ы) системы, отдельные варианты ядер больше не нужны.

В стандартные ядра включена поддержка SMP Для многопроцессорных систем больше не нужны \*-smp варианты ядра Linux. Для AMD64 в пакеты linux-image без суффикса -smp включена поддержка однопроцессорных и многопроцессорных систем.

Для исключённых пакетов, где это возможно, были добавлены пустые переходные пакеты, которые зависят от новых пакетов.

### 2.3.2 Новые утилиты для генерации initrd

Для загрузки системы пакетам Debian с образом ядра для AMD64 требуется initrd. Из-за изменений в ядре, утилитой используемой для генерации initrds в sarge, initrd-tools больше пользоваться нельзя. Для её замены были разработаны две утилиты: initramfs-tools и uaird. Принципы работы новых утилит очень разные; краткий обзор есть в вики Debian (<http://wiki.debian.org/InitrdReplacementOptions>). Для генерации initrd обе используют файловую систему initramfs, которая представляет собой сжатый архив сpio. Рекомендуемой и используемой по умолчанию утилитой является initramfs-tools.

При обновлении до ядра из etch по умолчанию происходит установка initramfs-tools.

Пакет initrd-tools всё ещё включён в etch, так как он требуется для обновления с sarge. Он будет исключён из следующего выпуска.

### 2.3.3 Динамическое управление /dev и обнаружение оборудования

Ядра в etch больше не поддерживают devfs.

Заменой devfs стал udev, реализация devfs в пользовательском пространстве.

udev монтируется в каталог /dev и создаёт в нём файлы устройств, которые поддерживаются ядром. Он также динамически добавляет и удаляет устройства при загрузке и выгрузке модулей, основываясь на событиях, генерируемых ядром. udev более гибок чем devfs и предоставляет сервисы, которые используются другими пакетами типа hal (абстрактный аппаратный слой).

Вместе с ядром, udev также заботится об обнаружении устройств и загрузке модулей для найденных устройств. Из-за этого он конфликтует с hotplug. В sarge, также можно было использовать discover для загрузки модулей во время процесса включения машины, но в его новой версии из etch этой функции больше нет. discover всё ещё используется X.Org для определения системной графической карты.

Если вы устанавливаете ядро Debian, то udev будет установлен по умолчанию, так как от него зависит пакет initramfs-tools.

Вы можете избежать установки `udev`, скомпилировав собственное не модульное ядро или используя альтернативный генератор `initrd`, например `yaird`. Однако, `initramfs-tools` является рекомендуемым генератором `initrd`.



## Глава 3

# Система установки

Программа установки Debian является официальной системой установки Debian. Она предлагает широкий выбор методов установки. Какие из них будут доступны на вашей системе зависит от архитектуры.

Образы программы установки для etch и руководство по установке можно найти на веб-сайте Debian (<http://www.debian.org/releases/stable/debian-installer/>).

Руководство по установке также включено на первый CD/DVD из официального набора Debian CD/DVD. Его можно найти в каталоге:

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

Также следует ознакомиться с списком известных ошибок (<http://www.debian.org/releases/stable/debian-installer/index#errata>) программы установки.

### 3.1 Что нового в системе установки?

Всё время с последнего официального выпуска sarge велась активная разработка программы установки Debian. В результате улучшилась работа с аппаратным обеспечением и были добавлены новые захватывающие возможности.

В информации о выпуске мы опишем только главные изменения в программе установки. Если вам нужен краткий обзор различий с sarge, обратитесь к анонсам выпуска etch beta и RC-выпусков, доступным из старых новостей (<http://www.debian.org/devel/debian-installer/News/>) о программе установки Debian.

#### 3.1.1 Главные изменения

Нет перезагрузки во время установки Ранее, процесс установки происходил в два этапа: установка базовой системы и делание её загрузочной, после чего происходила пере-



загрузка и запускалась программа `base-config`, с помощью которой выполнялась настройка пользователей, системы управления пакетами и установка дополнительных пакетов (посредством `tasksel`).

В `etch` второй этап был включён в саму программу установки Debian. Это дало ряд преимуществ: улучшилась безопасность, после перезагрузки в конце установки новой системы есть уже правильный часовой пояс, если вы установили среду рабочего стола, то будет сразу запущен графический интерфейс пользователя.

В новых системах кодировкой по умолчанию является UTF-8. Программа установки устанавливает системной кодировку UTF-8, то есть системная кодировка теперь не зависит от выбранного языка (типа ISO-8859-1, EUC-JP или KOI-8).

Более гибкая разметка. Теперь стало возможно настраивать файловые системы на томе LVM с помощью направляющей разметки.

Также, через программу установки можно настраивать зашифрованные файловые системы. Используя разметку вручную, вы можете выбирать между `dm-crypt` и `loop-aes`, использовать ключевую фразу или произвольный ключ, а также настраивать другие различные параметры. При использовании направляющей разметки, программа установки создаст зашифрованный раздел LVM, который может содержать любые файловые системы (кроме `/boot`) в логических томах.

Графический интерфейс пользователя. Если вы предпочитаете графический интерфейс пользователя, запустите программу установки командой `installgui`.

Функциональность графической программы установки практически идентична обычной программе установки — отличается только внешний вид. Есть одно исключение: в графическом интерфейсе не поддерживается настройка зашифрованных разделов с помощью произвольных ключей.

Основным преимуществом графического интерфейса пользователя является поддержка большего количества языков по сравнению с обычным пользовательским интерфейсом (`nwt`). Информацию о графическом установщике и самых важных различиях между графическим и обычным установщиком можно найти в приложении к руководству по установке.

Замечание: графический интерфейс доступен не на всех архитектурах.

Режим восстановления. Программу установки также можно использовать для решения проблем с системой, например, если она больше не загружается. Первые шаги выглядят как процесс обычной установки, но программа установки не будет запускать утилиту разметки. Вместо этого вам будет предложено меню восстановления.

Чтобы загрузиться в режиме восстановления, запустите программу установки командой `rescue` или добавьте загрузочный параметр `rescue/enable=true`.

Использование `sudo` вместо учётной записи `root`. Во время установки в режиме эксперта вы можете не настраивать учётную запись `root` (она будет заблокирована), а вместо неё настроить `sudo` так, чтобы первая учётная запись пользователя могла использоваться для администрирования системы.

**Криптографическая проверка скачиваемых пакетов** Пакеты, скачиваемые программой установки, теперь проверяются криптографическими методами с помощью apt, что ещё больше затрудняет компрометацию системы при установке по сети.

**Упрощённая настройка почты** При установке “стандартной системы” программа установки настраивает базовую конфигурацию почтового сервера системы, при которой предполагается только локальная доставка электронной почты. Почтовый сервер будет недоступен для других систем, подключённых к той же сети. Если вам нужно настроить систему для обработки не только локальной почты (для отправки или для приёма писем), выполните перенастройку почтовой системы после установки.

**Выбор рабочего стола** Система установки установит рабочий стол GNOME в качестве рабочего стола по умолчанию, если об этом попросит пользователь.

Однако, пользователи, желающие установить другое окружение рабочего стола, могут легко это сделать добавив параметры загрузки: `tasks="standard, kde-desktop"` для KDE и `tasks="standard, xfce-desktop"` для Xfce. Заметим, что это не сработает при установке с полного образа CD без использования сервера-зеркала из сети в качестве дополнительного источника пакетов. Установка возможна при использовании образа DVD или любого другого метода установки.

Для установки по умолчанию окружения рабочего стола KDE или Xfce доступны отдельные CD-образы.

**Новые языки интерфейса** Благодаря огромному труду переводчиков, установку Debian теперь можно выполнять на 47 языках через текстовый пользовательский интерфейс программы установки. Это на шесть языков больше чем было в sarge. Языки, добавленные в этом выпуске: белорусский, эсперанто, эстонский, курдский, македонский, тагалог, вьетнамский и волоф. В этом выпуске из-за недостаточной поддержки обновлений перевода были убраны персидский и велш.

При использовании графического интерфейса программы установки дополнительно поддерживаются одиннадцать языков. Эти языки можно выбрать только в графическом установщике, так как их наборы символов нельзя отобразить в не графическом окружении. Новые языки: бенгальский, джонка, гуарати, хинди, грузинский, кхмерский, малайялам, непальский, пунжаби, тамильский и тайский.

Если пользователь не хочет использовать какую-то определённую локаль, то теперь он может выбрать C в качестве предпочитаемой локали при выборе языка установки. Более подробная информация о языках доступна в списке языков d-i (<http://d-i.alioth.debian.org/i18n-doc/languages.html>).

**Упрощение выбора локализации и часового пояса** Настройка языка, страны и часового пояса была упрощена для уменьшения количества информации, запрашиваемой от пользователя. Теперь программа установки на основе выбранного языка определяет страну и часовой пояс для системы, или предлагает ограниченный список, если не может этого сделать. Если требуется, пользователь может самостоятельно выбрать желаемую комбинацию.

**Улучшение общей локализации системы** Большинство задач интернационализации и локализации, которые ранее выполнялись программой `localization-config`, теперь включены

в программу установки Debian или в конкретные пакеты. Это означает, что при выборе языка будут автоматически установлены пакеты (словари, документация, шрифты...), необходимые для поддержки этого языка в стандартном окружении и на рабочем столе. Некоторые настройки теперь не выполняются автоматически. Сюда входят выбор размера бумаги и расширенная настройка клавиатуры X Windows для работы сразу с несколькими языками.

Заметим, что пакеты поддержки языка будут установлены автоматически, только если они доступны во время установки.

### 3.1.2 Автоматизируемая установка

Множество изменений, упомянутых в предыдущем разделе, также затронули и автоматическую установку согласно конфигурационному файлу настройки. Это означает, что если у вас уже есть конфигурационные файлы настройки, которые работали с программой установки `sarge`, то не ждите, что они будут работать с новой программой установки без изменений.

В руководстве по установке (<http://www.debian.org/releases/stable/installmanual>) теперь есть отдельное приложение, в котором подробно описывается использование автоматической настройки.

В программу установки `etch` добавлено несколько новых потрясающих возможностей, которые позволяют легко автоматизировать установки до мелочей. Также поддерживается усовершенствованная разметка RAID, LVM и шифрованные LVM. Подробности смотрите в документации.

## 3.2 Частота использования пакетов

Программа установки снова предлагает установить пакет `popularity-contest`. Этот пакет не устанавливался по умолчанию в `sarge`, но устанавливался в более старых выпусках.

`popularity-contest` снабжает проект Debian ценной информацией о том, какие пакеты дистрибутива действительно используются. Эта информация не только влияет на то, в каком порядке пакеты будут расположены на установочных компакт-дисках, но и используется разработчиками Debian для принятия решения о продолжении/прекращении поддержки пакетов, у которых больше нет сопровождающих.

Информация, собранная `popularity-contest`, обрабатывается анонимно. Мы будем вам очень благодарны, если вы установите этот пакет и примете участие в этом официальном опросе; этим вы поможете улучшить Debian.

## Глава 4

# Обновление с предыдущих выпусков

### 4.1 Подготовка к обновлению

Перед обновлением мы рекомендуем вам также прочитать информацию в ‘Что надо знать о etch’ на стр. 37. В этом разделе дано описание потенциальных проблем, которые не имеют прямого отношения к процессу обновления, но всё же важно знать об этом.

#### 4.1.1 Создание резервной копии всех данных и настроек

Перед обновлением настоятельно рекомендуется сделать резервную копию всей системы или, по крайней мере, всех данных и настроек, которые вы не можете позволить себе потерять. Инструменты для обновления и сам процесс достаточно надёжны, но сбой аппаратного обеспечения в ходе обновления может привести к серьёзному повреждению системы.

В первую очередь необходимо сделать резервные копии содержимого каталогов `/etc`, `/var/lib/dpkg`, `/var/lib/aptitude/pkgstates` и результат работы команды `dpkg --get-selections "*" (кавычки необходимы)`.

Обновление само по себе никак не затрагивает каталог `/home`. Однако некоторые приложения (в частности, некоторые приложения Mozilla, окружения рабочего стола GNOME и KDE) при первом запуске новой версии перезаписывают существующие пользовательские настройки новыми настройками по умолчанию. На всякий случай стоит сделать резервную копию скрытых файлов и каталогов (“точечных файлов”) из домашних каталогов пользователей. Это поможет вам в случае необходимости восстановить старые настройки. Не лишним будет предупредить пользователей об этой проблеме.

Установка любого пакета должна производиться с привилегиями суперпользователя. Поэтому вы должны, либо войти в систему под именем суперпользователя (`root`), либо использовать программы `su` или `sudo`, чтобы получить необходимые права.

Для обновления есть несколько предварительных условий; проверьте, что они соблюдаются.

### 4.1.2 Предварительное информирование пользователей

Желательно заранее сообщить всем пользователям о любых планируемых обновлениях, хотя пользователи, работающие в системе через `ssh`, не должны испытать особых проблем в процессе обновления и вполне могут продолжать работу.

Если вы хотите принять дополнительные меры предосторожности, создайте резервную копию каталога с файлами пользователей (`/home`) или перед обновлением размонтируйте раздел, где он расположен.

Вероятно при обновлении до `etch` вам потребуется обновить ядро, поэтому перезагрузка чаще всего нужна. Обычно, это делается после завершения обновления.

### 4.1.3 Подготовка к откату

Так как ядро в `sarge` и `etch` сильно различается, а именно: в драйверах, определении оборудования, именовании и порядке файлов устройств, есть реальная опасность возникновения проблем после перезагрузки в обновлённую систему. Большая часть возможных проблем описана здесь и далее в следующих главах информации о выпуске.

Поэтому имеет смысл проверить, что вы сможете откатиться назад, если система не сможет загрузиться или, для систем, управляемых удалённо, не заработает сеть.

Если обновление производится удалённо через канал `ssh`, то настоятельно рекомендуем вам предусмотреть подключение к серверу через удалённый последовательный терминал. Может случиться так, что после обновления ядра и перезагрузки, некоторые устройства будут переименованы (это описывается в ‘Переопределение нумерации устройств’ на стр. 31) и вы сможете исправить системные настройки через локальную консоль. Также, если система случайно перезагрузится в середине процесса обновления, в этом случае вам потребуется использовать локальную консоль для восстановления.

Очевидно, что сначала нужно попробовать загрузиться со старым ядром. Однако, по различным причинам, описанным в разных частях этого документа, это не обязательно срывает.

Если загрузиться не удалось, вам может потребоваться альтернативный способ загрузки системы для доступа к ней и восстановления. Для этого можно воспользоваться специальным образом восстановления или Linux лайв CD. После загрузки каким-то способом, вы сможете смонтировать имеющуюся корневую файловую систему и выполнить в неё `chroot`, чтобы найти и исправить ошибку.

Также мы хотим порекомендовать вам воспользоваться режимом восстановления программы установки Debian `etch`. Преимущество в использовании программы установки в том, что вы можете выбрать из разных методов установки, тот который лучше подходит в вашем случае. Подробная информация приведена в разделе “Восстановление неработающей системы” главы 8 руководства по установке (<http://www.debian.org/releases/stable/installmanual>) и в FAQ по программе установки Debian (<http://wiki.debian.org/DebianInstaller/FAQ>).

Отладочная оболочка из `initrd` во время загрузки

`initramfs-tools` включает отладочную оболочку<sup>1</sup> в создаваемый `initrd`. Пример: если `initrd` не удастся смонтировать корневую файловую систему, то вы попадёте в эту отладочную оболочку, в которой доступны основные команды, которые помогут вам выяснить в чём дело и, возможно, даже всё исправить.

Что нужно проверить: наличие правильных файлов устройств в каталоге `/dev`; какие модули загружены (`cat /proc/modules`); проверить в выводе `dmesg` нет ли ошибок при загрузке драйверов. Вывод `dmesg` также покажет, какие файлы устройств были назначены каким дискам; убедитесь, что результатом `cat $ROOT` является ожидаемое устройство для корневой файловой системы.

Если вы решили проблему, для выхода из отладочной оболочки наберите `exit`, и процесс загрузки продолжится с места останова. Естественно, вам нужно устранить причину проблемы и регенерировать `initrd` для, того чтобы следующая загрузка стала удачной.

#### 4.1.4 Подготовка безопасного окружения для обновления

Обновление дистрибутива возможно как локально через виртуальную текстовую консоль (или подключённого напрямую последовательного терминала), так и удалённо через `ssh`.

Чтобы ещё более повысить безопасность, мы предлагаем запускать обновление в виртуальной консоли, созданной с помощью программы `screen`, которая позволяет безопасно переподключаться и таким образом обновление не будет прервано, даже если произойдёт обрыв связи.

Внимание! Ни в коем случае не обновляйте систему через `telnet`, `rlogin`, `rsh` или из графического сеанса под управлением `xdm`, `gdm`, `kdm` и т.д., запущенного на обновляемой системе. Это связано с тем, что все эти сервисы могут быть перезапущены при обновлении, что может привести к недоступности системы, обновлённой только наполовину.

#### 4.1.5 Прекращение поддержки ядер версий 2.2

Если у вас работает ядро с версией меньшей 2.4.1, то перед тем как обновлять `glibc`, вам нужно провести обновление до ветки 2.4 (как минимум). Это лучше сделать до начала обновления. Рекомендуется обновить ядро до версии 2.6.8, доступной в `sarge`, вместо обновления до ветки 2.4.

## 4.2 Проверка состояния системы

При описании процесса обновления в этой главе предполагалось, что производится обновление с “чистого” `sarge` без установленных пакетов сторонних производителей. В частности, из-за перехода на `X.Org` (‘Переход с `XFree86` на `X.Org`’ на стр. 39), возникают проблемы со

<sup>1</sup>Эту возможность можно выключить, добавив параметр `panic=0` к параметрам загрузки.

сторонними пакетами, программы которых установлены в `/usr/X11R6/bin/`. Для большей надёжности проведения процесса обновления лучше удалить сторонние пакеты из системы.

Эта процедура также предполагает, что ваша система обновлена до последнего выпуска `sarge`. Если это не так или вы в этом не уверены, следуйте инструкциям раздела ‘Обновление системы `sarge`’ на стр. 51.

#### 4.2.1 Проверка отложенных действий менеджера пакетов

Иногда, использование `apt-get` для установки пакетов вместо `aptitude`, может привести к тому, что `aptitude` считает пакет “неиспользуемым” и планирует его удалить. В общем, система должна быть полностью обновлена и считаться “чистой”.

Во-первых, вы должны проверить, не ожидается ли каких-нибудь действий в менеджере пакетов `aptitude`. Если для пакета запланировано удаление или обновление в менеджере пакетов, то это может отрицательно сказаться на процедуре обновления. Заметим, что исправление возможно, пока в вашем `sources.list` указаны `sarge`, а не `stable` или `etch`; смотрите ‘Проверка списка источников АРТ’ на стр. 51.

Для этого запустите `aptitude` в интерактивном режиме и нажмите ‘g’ (“Начали”). Если после этого будет предложено что-то сделать, вам нужно просмотреть, что именно и, или отменить эти действия, или подтвердить их выполнение. Если никаких действий совершить не предлагается, то будет показано сообщение “Ни одного пакета не будет установлено, обновлено или удалено”.

#### 4.2.2 Отключение фиксации АРТ

Если вы настроили АРТ на установку некоторых пакетов из дистрибутива, отличного от стабильного (например, тестируемого), вам, возможно, придётся изменить конфигурацию АРТ (сохранённую в файле `/etc/apt/preferences`), чтобы было возможно обновление пакетов до версии в новом стабильном выпуске. Более подробную информацию о фиксации пакетов в АРТ можно найти в `apt_preferences(5)`.

#### 4.2.3 Проверка состояния пакетов

Независимо от способа обновления, рекомендуется сначала проверить состояние всех пакетов и убедиться, что все пакеты могут быть обновлены. Следующая команда покажет частично установленные (Half-Installed) пакеты, пакеты, настройка которых не удалась (Failed-Config), а также все пакеты в ошибочных состояниях.

```
# dpkg --audit
```

Можно также проверить состояние всех пакетов системы с помощью программ `dselect`, `aptitude` или таких команд, как

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

или

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

Перед обновлением желательно снять все фиксации с пакетов. Если один из необходимых для обновления пакетов зафиксирован, обновление завершится неудачей.

Заметим, что `aptitude` использует отличный от `apt-get` и `dselect` метод регистрации зафиксированных пакетов. Определить, какие пакеты зафиксированы через `aptitude`, можно с помощью

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

Если вы хотите проверить, какие пакеты зафиксированы через `apt-get`, используйте

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

Если вы изменяли и перекомпилировали пакет локально, но при этом не переименовали его и не добавили “эпоху” к версии пакета, то его нужно зафиксировать, чтобы предотвратить обновление.

Состояние фиксации пакета (“hold”) для `aptitude` может быть изменено с помощью:

```
# aptitude hold имя_пакета
```

Замените `hold` на `unhold`, чтобы сбросить состояние “hold”.

Если вам нужно что-то изменить, то лучше сначала убедиться, что ваш `sources.list` всё ещё ссылается на `sarge`, как объясняется в разделе ‘Проверка списка источников АРТ’ на стр. 51.

#### 4.2.4 Неофициальные источники и адаптации (backports)

Если вы устанавливали некоторые пакеты не из репозитория Debian, то знайте, что при обновлении они могут быть удалены по причине конфликтующих зависимостей. Если для установки пакетов вы добавляли записи в файл `/etc/apt/sources.list`, проверьте, содержит ли этот источник пакеты для `etch` и соответственно исправьте строку источника вместе со строками остальных источников пакетов Debian.

У некоторых пользователей могут быть уже установлены более “новые” версии пакетов Debian, собранные для `sarge`. Такие пакеты, скорее всего, станут причиной проблем при обновлении, так как они могут вызвать конфликты файлов<sup>2</sup>. В разделе ‘Возможные проблемы во время обновления’ на стр. 28 дана некоторая информация о том, что делать при конфликте файлов, если они возникают.

---

<sup>2</sup>Обычно, система управления пакетами Debian не позволяет пакету удалить или заменить файл, принадлежащий другому пакету, за исключением случаев, когда явно указано, что один пакет заменяет другой.



### 4.3 Снятие пометки с пакетов вручную

Чтобы не дать aptitude удалить некоторые пакеты, которые были установлены для разрешения зависимостей, вам нужно вручную снять с них пометку авто. К таким пакетам относятся OpenOffice и Vim при установке на рабочую станцию:

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

Также это касается образов ядра 2.6, если вы устанавливали их с помощью ядерного мета-пакета:

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

Примечание: чтобы посмотреть какие пакеты отмечены в aptitude как авто, выполните:

```
# aptitude search '~M <имя пакета>'
```

### 4.4 Подготовка источников АРТ

Перед началом процесса обновления вы должны настроить файл конфигурации apt /etc/apt/sources.list.

Пакет apt при запросе на установку установит самую новую версию пакета из тех, которые можно найти в источниках, соответствующих строкам deb этого файла. При равных версиях приоритет будут иметь строки, находящиеся в файле выше (таким образом, в случае использования нескольких зеркал обычно лучше первым указать каталог на локальном жёстком диске, затем компакт-диск и, наконец, зеркала HTTP/FTP).

На выпуск Debian часто ссылаются по его кодовому имени (например, “sarge” или “etch”) или по его статусу (например, “oldstable” (“старый стабильный”), “stable” (“стабильный”), “testing” (“тестируемый”), “unstable” (“нестабильный”). Ссылки на выпуск по его кодовому имени имеют то преимущество, что вы никогда неожиданно для себя не начнёте пользоваться новым выпуском. Поэтому мы здесь так и делаем. В этом случае, вам, конечно, придётся самим следить за новыми выпусками. Если вы будете ссылаться на дистрибутив по его статусному имени, то после выхода нового выпуска вы просто увидите большое количество обновлённых пакетов.

#### 4.4.1 Как добавить Интернет-источник АРТ

Конфигурация по умолчанию позволяет произвести установку с главных серверов Debian, но вы можете изменить файл /etc/apt/sources.list и использовать другие серверы-зеркала. Лучше использовать ближайший к вам (в смысле скорости передачи данных) сервер-зеркало.

Адреса HTTP- и FTP-зеркал Debian можно найти по адресу <http://www.debian.org/distrib/ftplist> (см. раздел “Список зеркал Debian”). HTTP-зеркала обычно быстрее, чем FTP.

Допустим, к примеру, что ближайшим к вам зеркалом Debian является <http://mirrors.kernel.org/debian>. Просматривая содержимое зеркала веб-браузером или FTP-клиентом, можно заметить, что основные каталоги организованы следующим образом:

```
http://mirrors.kernel.org/debian/dists/etch/main/binary-amd64/...
http://mirrors.kernel.org/debian/dists/etch/contrib/binary-amd64/...
```

Чтобы использовать это зеркало в программе apt, добавьте в файл sources.list:

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

Заметим, что ‘dists’ добавляется автоматически, а параметры после имени выпуска используются для дополнения сетевого пути до реальных каталогов.

После того, как вы добавили новый источник, отключите присутствовавшие ранее в файле sources.list строки deb, поставив в начале каждой строки решётку (#).

#### 4.4.2 Как добавить локальное зеркало АРТ

Вместо HTTP- и FTP-зеркал, можно использовать зеркало на локальном диске (возможно, смонтированном по сети), изменив соответствующим образом файл /etc/apt/sources.list.

Например, если ваше зеркало пакетов находится в каталоге /var/ftp/debian/, то его основные каталоги будут называться:

```
/var/ftp/debian/dists/etch/main/binary-amd64/...
/var/ftp/debian/dists/etch/contrib/binary-amd64/...
```

Чтобы программа apt могла использовать этот источник пакетов, добавьте в файл sources.list:

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

Заметим, что ‘dists’ добавляется автоматически, а параметры после имени выпуска используются для дополнения сетевого пути до реальных каталогов.

После того, как вы добавили новый источник, отключите присутствовавшие ранее в файле sources.list строки deb, поставив в начале каждой строки решётку (#).

### 4.4.3 Как добавить источник АРТ на компакт-диске или DVD

Если вы хотите использовать только компакт-диски, прокомментируйте все присутствующие в файле `/etc/apt/sources.list` строки `deb`, поставив в начале каждой строки знак диеза (`#`).

Убедитесь в том, что в файле `/etc/fstab` есть строка, позволяющая смонтировать компакт-диск в каталог `/cdrom` (программа `apt-cdrom` требует монтирования именно в этом каталоге). Например, если файлом устройства вашего привода чтения компакт-дисков служит `/dev/hdc`, то файл `/etc/fstab` должен содержать строку:

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

Обратите внимание на то, что между словами `defaults,noauto,ro` в четвёртом поле строки не должно быть пробелов.

Чтобы проверить, что строка `fstab` написана верно, вставьте компакт-диск в привод и выполните команды

```
# mount /cdrom # эта команда примонтирует CD к точке монтирования
# ls -alF /cdrom # эта команда должна показать корневой каталог CD-диска
# umount /cdrom # эта команда размонтирует CD
```

Затем запустите команду

```
# apt-cdrom add
```

для каждого компакт-диска Debian Binary, чтобы добавить все диски в базу данных АРТ.

## 4.5 Обновление пакетов

Для обновления Debian GNU/Linux до нового выпуска рекомендуется использовать программу управления пакетами `aptitude`. Этот инструмент принимает более безопасные решения об установке пакетов, чем при запуске непосредственно `apt-get`.

Не забудьте смонтировать все необходимые разделы (в первую очередь, корневой раздел и каталог `/usr`) для чтения и записи. Это можно сделать командой

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

Затем надо проверить и перепроверить, что источники АРТ (в `/etc/apt/sources.list`) ссылаются, либо на `etch`, либо на `stable`. Не должно быть источников, указывающих на `sarge`. Примечание: обратите внимание, источники на CD-ROM часто ссылаются на дистрибутив `unstable`, хоть это и странно, но исправлять их не нужно.

### 4.5.1 Запись сессии

Настоятельно рекомендуем использовать программу `/usr/bin/script` для записи ваших действий при обновлении. В этом случае, при возникновении каких-либо проблем у вас будет журнал произошедших событий, и если понадобится, вы сможете включить эту информацию в отчёт об ошибке. Чтобы начать запись, выполните команду

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

или похожую. Не размещайте файл с записью во временных каталогах, таких как `/tmp` и `/var/tmp` (файлы в этих каталогах могут быть удалены при обновлении или перезагрузке).

Запись также поможет вам прочитать информацию, которая была “промотана” за границу экрана. Просто переключитесь на вторую консоль (VT2, нажав Alt-F2) и, войдя в систему, запустите `less -R ~/root/upgrade-etch.script` для просмотра файла.

После завершения обновления, можно остановить `script`, набрав `exit` в командной строке.

Если параметром к `script` указать `-t`, то вы сможете воспользоваться программой `scriptreplay` для повтора всей сессии:

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

### 4.5.2 Обновление списка пакетов

Во-первых, нужно обновить список доступных пакетов нового выпуска. Это делается командой:

```
# aptitude update
```

Запуск команды в первый раз для новых источников приведёт к выводу нескольких предупреждений о доступности источников. Эти предупреждения безобидны и больше не появятся при последующих запусках команды.

### 4.5.3 Проверка достаточности места для обновления

Перед обновлением всей системы, как описано в ‘Обновление остальной системы’ на стр. 27, вы должны убедиться, что у вас достаточно места на диске. Сначала все нужные для установки пакеты скачиваются из сети и сохраняются в `/var/cache/apt/archives` (и на время скачивания, в подкаталог `partial/`), поэтому вы должны проверить место на разделе, который содержит `/var/`, так как туда будут временно записываться загружаемые пакеты для установки. После загрузки вам, вероятно, понадобится больше места в других разделах файловой системы для установки обновляемых пакетов (которые могут содержать больше данных) и новых пакетов, которые будут установлены из-за обновлений. Если в системе не

будет достаточно места, то в конечном счёте это приведёт к неполному обновлению, что позднее бывает сложно исправить.

И `aptitude`, и `apt` отображают подробную информацию о количестве требуемого места для установки. Перед выполнением обновления вы можете увидеть размер необходимого места, запустив:

```
# aptitude -y -s -f --with-recommends dist-upgrade
[ ... ]
XXX пакетов обновлено, XXX установлено новых, XXX пакетов отмечено для удаления
и XXX пакетов не обновлено.
Необходимо получить хх.хМБ/уууМБ архивов. После распаковки будет использовано
АААМБ.
Будет выполнена загрузка/установка/удаление пакетов.
```

3

Если вам не хватает места для обновления — освободите место заранее. Вы можете:

- Удалить пакеты, которые были ранее загружены для установки (из каталога `/var/cache/apt/archive`). При очистке кэша пакетов с помощью команды `apt-get clean` или `aptitude clean` будут удалены все ранее скачанные файлы пакетов.
- Удалить старые пакеты, которые больше не используются. Если у вас установлена программа `popularity-contest`, то с помощью `popcon-largest-unused` можно получить список пакетов, которые вы не используете в системе и которые занимают наибольшее количество места. Также вы можете использовать `deborphan` или `debfoaster`, чтобы найти устаревшие пакеты (смотрите ‘Устаревшие пакеты’ на стр. 34). Ещё вы можете запустить `aptitude` в интерактивном режиме и найти устаревшие пакеты в меню “Устаревшие и пакеты, созданные локально”.
- Удалить пакеты, которые занимают слишком много места и в которых нет срочной необходимости (вы всегда можете переустановить их после обновления). Список пакетов, которые занимают большую часть дискового пространства, можно получить с помощью программы `dpigs` (из пакета `debian-goodies`) или `wajig` (запустив `wajig size`).
- Временно перенести на другую машину или удалить системные журналы из `/var/log/`.

Заметим, что для безопасного удаления пакетов рекомендуется в `sources.list` исправить источники обратно на `sarge`, как объясняется в разделе ‘Проверка списка источников АРТ’ на стр. 51.

---

<sup>3</sup>Запуск этой команды в начале процесса обновления может приводить к ошибке по причинам, описанным далее. В этом случае запуск этой команды оценки дискового пространства вам нужно провести после выполнения минимального обновления системы, описанного в ‘Минимальное обновление системы’ на стр. 25 и обновления ядра (‘Обновление ядра’ на стр. 26).

#### 4.5.4 Минимальное обновление системы

Так как определённые необходимые пакеты конфликтуют при обновлении `sarge` до `etch`, непосредственный запуск `aptitude dist-upgrade` часто приводит к удалению большого числа пакетов, которые вы хотели бы сохранить. Поэтому мы рекомендуем производить обновление в две стадии — сначала минимальное обновление, чтобы разобраться с конфликтами, а затем полный `dist-upgrade`.

Сначала запустите:

```
# aptitude upgrade
```

Это приведёт к обновлению только тех пакетов, которые можно обновить без необходимости удаления или установки других пакетов.

Затем минимальное обновление с помощью:

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

По этой команде автоматически обновятся `libc6` и `locales` и вытянутся библиотеки поддержки SE linux (`libslinux1`). После этого будут перезапущены некоторые работающие сервисы, включая `xdm`, `gdm` и `kdm`, а следовательно, будут прерваны локальные сеансы X11.

Следующий шаг зависит от набора установленных пакетов. В данном документе даны общие советы о том, какие методы должны быть использованы, но если есть сомнения, то перед тем как продолжить, рекомендуется проверить каждым методом предлагаемые к удалению пакеты.

Некоторые общие пакеты, которые будут удалены: `base-config`, `hotplug`, `xlibs`, `netkit-inetd`, `python2.3`, `xfree86-common` и `xserver-common`. Полный список устаревших пакетов в `etch` смотрите ‘Устаревшие пакеты’ на стр. 34.

#### Обновление настольной системы

Этот способ обновления проверен на системах `sarge` с установленной задачей рабочий стол. Вероятно, это лучший метод для систем с установленной задачей рабочий стол или установленными пакетами `gnome` или `kde`.

Вероятно, это неправильный метод, если у вас не установлены пакеты `libfam0c102` и `xlibmesa-glu`:

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

Если у вас установлена полная настольная система, выполните:

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

### Обновление системы с несколькими установленными пакетами для X

Для систем с установленными только несколькими пакетами для X, не полной с задачей рабочий стол, требуется другой метод. Этот метод, в основном, применяется к системам с установленным пакетом `xfree86-common`, включая некоторые серверные системы, для которых с помощью `tasksel` были установлены инструменты управления сервером с графическим интерфейсом. Это, вероятно, корректный метод для систем с запущенными X, но без установленной полностью задачей рабочий стол.

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

Во-первых, проверьте, что установлены пакеты `libfam0c102` и `xlibmesa-glu`.

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

Если у вас не установлен `libfam0c102` не включайте `libfam0` в командную строку ниже. Если у вас не установлен `xlibmesa-glu` не включайте его в командную строку ниже.<sup>4</sup>

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

Заметим, что при установке `libfam0` также устанавливается монитор слежения за состоянием файлов (`fam`), а также RPC `portmapper` (`portmap`), если их ещё нет в системе. Оба пакета предоставляют новый сетевой сервис в системе, хотя их можно настроить на работу через (внутреннее) сетевое устройство обратной петли.

### Обновление системы без установленных пакетов для X

Если в системе не установлены X, то дополнительная команда `aptitude install` не нужна, и вы можете переходить к следующему шагу.

#### 4.5.5 Обновление ядра

Версия `udev` из `etch` не поддерживает ядра версий менее 2.6.15 (в `sarge` включены ядра версии 2.6.8), и наоборот, версия `udev` из `sarge` не заработает с последними версиями ядер. Как следствие, ядро из предыдущего ядерного пакета, вероятно, не загрузится после обновления. Также, установка `udev` из `etch` приводит к принудительному удалению пакета `hotplug`, который используется ядрами Linux ветки 2.4.

<sup>4</sup> Данная команда определит, нужно ли устанавливать `libfam0` и `xlibmesa-glu` и автоматически выберет что нужно:

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

В результате, предыдущий пакет с ядром, вероятно, больше не сможет загрузить машину после обновления. Также, образуется промежуток времени, в котором `udev` уже обновлён, а ядро ещё нет. Если система будет перезагружена в этот момент, в середине обновления, то она может больше не загрузиться, так как устройства будут не найдены и драйверы не загрузятся. (Смотрите в ‘Подготовка безопасного окружения для обновления’ на стр. 17 рекомендации по подготовке к этому, если вы производите обновление удалённо.)

Если в системе установлена задача рабочий стол или другие пакеты, которые могут привести к неприемлемо большому числу удаляемых пакетов, рекомендуется обновлять ядро отдельно от всего остального.

Для обновления ядра выполните:

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

Информация из ‘Установка ядерного метапакета’ на стр. 30 поможет вам определить нужный вариант пакета с ядром для установки.

В случае рабочей станции, к сожалению, невозможно гарантировать, что пакет с ядром установит сразу после обновления `udev`, поэтому этот промежуток времени неопределённой длины когда система не имеет установленного ядра с полной поддержкой автоопределения оборудования. Обратитесь к ‘Обновление ядра и сопутствующих пакетов’ на стр. 29 за информацией по настройке системы, которая не зависит от автоопределения оборудования при загрузке.

#### 4.5.6 Обновление остальной системы

Теперь можно приступить собственно к обновлению. Выполните команду:

```
# aptitude dist-upgrade
```

Эта команда произведёт полное обновление системы, т.е. установку последних доступных версий всех пакетов и разрешение всех изменившихся между выпусками зависимостей. При необходимости будут установлены новые пакеты (обычно это новые версии библиотек или переименованные пакеты) и удалены все вызывающие конфликты устаревшие пакеты.

При обновлении с набора компакт-дисков система несколько раз попросит вас вставить в привод соответствующий диск. Возможно, вам придётся вставлять один диск несколько раз. Это вызвано тем, что взаимосвязанные пакеты могут находиться на разных дисках.

Новые версии уже установленных пакетов, которые нельзя обновить не меняя состояние других пакетов, не будут установлены (программа выведет сообщение, что пакет зафиксирован, “held back”). Разрешить ситуацию можно, либо отметив такой пакет как подлежащий установке с помощью `aptitude`, либо выполнив команду `aptitude -f install пакет`.



### 4.5.7 Получение подписей пакетов

После обновления с новой версией apt теперь вы можете обновить информацию о пакетах, в которой теперь есть новый механизм проверки подписей пакетов:

```
# aptitude update
```

При обновлении уже выполняется получение и активация ключей с подписями для архивов пакетов Debian. Если у вас добавлены другие (неофициальные) источники пакетов, то apt будет выдавать предупреждение о невозможности проверки и требовать подтверждения, что пакеты загружены из правильных источников и не были подделаны. Подробнее об этом смотрите в ‘Управление пакетами’ на стр. 5.

При использовании новой версии apt, вы заметите, что скачиваются файлы различий (pdiff), а не полный список пакетов. Подробнее об этой возможности читайте в ‘Медленно обновляются пакетные индексные файлы АРТ’ на стр. 38.

### 4.5.8 Возможные проблемы во время обновления

Если работа с aptitude, apt-get или dpkg завершается с ошибкой

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

то это значит, что размера кэша по умолчанию недостаточно. Проблему можно решить, либо удалив ненужные записи в файле /etc/apt/sources.list, либо увеличив размер кэша. Увеличить размер кэша можно с помощью параметра АРТ::Cache-Limit в файле /etc/apt/apt.conf. Следующая команда увеличит кэш до размера, достаточного для обновления:

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

Здесь предполагается, что эта переменная раньше не была указана в файле.

Иногда необходимо активизировать параметр АРТ::Force-LoopBreak в АРТ, для того чтобы можно было временно удалять пакеты первой необходимости при возникновении циклической зависимости типа конфликт/требование предварительной установки. Программа aptitude сообщит об этом и прекратит обновление. Чтобы решить эту проблему, следует указать в командной строке aptitude параметр -o АРТ::Force-LoopBreak=1.

Возможна ситуация, когда структура зависимостей в системе может быть настолько нарушена, что потребуются ручное вмешательство. Обычно это делается с помощью aptitude или

```
# dpkg --remove имя_пакета
```

чтобы удалить “проблемные” пакеты или

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

В чрезвычайных случаях вам может потребоваться принудительная переустановка командой типа

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

При обновлении с “чистого” sarge конфликтов файлов быть не должно, но они вполне возможны, если у вас установлены неофициальные адаптации (backports). При конфликте файлов появляются ошибки:

```
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>
```

Конфликт файлов можно попытаться разрешить принудительным удалением пакета, указанного в последней строке сообщения об ошибке:

```
# dpkg -r --force-depends имя_пакета
```

После исправления ошибок вы можете продолжить обновление, повторив ранее описанные команды aptitude.

Во время обновления система будет задавать вопросы о настройке или перенастройке некоторых пакетов. Если вас спросят, нужно ли заменить какой-то файл в каталоге /etc/init.d или /etc/terminfo, или файл /etc/manpath.config версией, предоставленной сопровождающим пакета, обычно следует ответить ‘да’, поскольку иначе будет нарушена целостность системы. Вы всегда сможете вернуть старые настройки, потому что файлы конфигурации будут сохранены с расширением .dpkg-old.

Если вы не уверены в том, что нужно делать, запишите имя пакета или файла и разберитесь с проблемой позднее. Информацию, выводившуюся на экран во время обновления, вы сможете найти в файле транскрипта.

## 4.6 Обновление ядра и сопутствующих пакетов

В этом разделе описано как обновить ядро и выявить потенциальные проблемы, связанные с этим. Вы можете установить один из пакетов linux-image-\*, предоставленных Debian, или скомпилировать ядро из исходников самостоятельно.

Заметим, что большая часть информации раздела основана на предположении, что вы будете использовать одно из модульных ядер Debian вместе с `initramfs-tools` и `udev`. Если вы будете использовать своё собственное ядро, которому не требуется `initrd`, или вы используете другой генератор `initrd`, то некоторая информация может быть для вас не существенна.

Заметим также, что если `udev` не установлен в системе, то для обнаружения устройств всё ещё возможно использовать `hotplug`.

#### 4.6.1 Установка ядерного метапакета

При `dist-upgrade` с `sarge` до `etch` настоятельно рекомендуется установить новый метапакет `linux-image-2.6-*`. Этот пакет может быть установлен автоматически в процессе `dist-upgrade`. Вы можете проверить это командой:

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

Если в результате вы ничего не увидели, то вам нужно установить новый пакет `linux-image` вручную. Чтобы увидеть список доступных метапакетов `linux-image-2.6`, выполните:

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

Если вы не знаете какой пакет выбрать, запустите `uname -r` и поищите пакет с похожим именем. Например, если вы увидели `'2.4.27-3-686'`, то рекомендуется установить `linux-image-2.6-686`. Для выбора подходящего из возможных, вы можете воспользоваться `apt-cache`, чтобы посмотреть подробное описание каждого пакета. Например:

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

Ядро нужно устанавливать с помощью `aptitude install`. Чтобы воспользоваться преимуществами новой версии ядра, следует перезагрузить систему при первой возможности.

Если вы не боитесь трудностей, то можете скомпилировать ядро самостоятельно из исходного кода. Установите пакет `kernel-package` и прочитайте документацию в каталоге `/usr/share/doc/kernel-package`.

#### 4.6.2 Обновление ядра версии 2.6

Если вы используете ядро ветки 2.6 из `sarge`, то его обновление будет выполнено автоматически только после полного обновления пакетов системы (как описано в ‘Обновление пакетов’ на стр. 22).

Если возможно, то предпочтительно обновлять ядро отдельно от основного `dist-upgrade`, чтобы уменьшить шансы получить временно незагружаемую систему. Описание этого процесса смотрите в ‘Обновление ядра’ на стр. 26. Заметим, что это нужно выполнять только

после процесса минимального обновления, описанного в ‘Минимальное обновление системы’ на стр. 25.

Также это работает, если вы используете ядро собственной сборки и хотели бы перейти на ядро из `etch`. Если ваша версия ядра не поддерживается пакетом `udev`, то рекомендуется выполнять обновление после минимального обновления. Если ваша версия поддерживается `udev`, то вы можете ожидать благополучного окончания полного обновления системы.

### 4.6.3 Переопределение нумерации устройств

В `etch` встроено более мощный механизм обнаружения аппаратных средств, чем в предыдущих выпусках. Однако это может приводить к изменению порядка устройств обнаруженных в системе, что повлияет на назначенные имена устройств. Например, если у вас есть две сетевые карты, которые работают через разные драйверы, то устройства `eth0` и `eth1` могут поменяться местами. Обратите внимание на то, что новый механизм работает так, что если вы, например, поставите другую карту ethernet в систему с `etch`, то новой карте будет присвоено новое имя интерфейса.

Для сетевых устройств вы можете избежать этого переназначения с помощью правил `udev`, определив устройства в файле `/etc/udev/rules.d/z25_persistent-net.rules`<sup>5</sup>. Также для решения можно воспользоваться утилитой `ifrename`, привязав физические устройства к определённым именам во время загрузки. Подробности смотрите в `ifrename(8)` и `iftab(5)`. Оба варианта (`ifrename` и `udev`) нельзя использовать одновременно.

Для устройств хранения данных можно избежать переназначения с помощью утилиты `initramfs-tools` и настройки загрузки драйверов модулей в одном и том же порядке, в котором они загружены в данный момент. Для этого определите порядок модулей устройств хранения в системе с помощью команды `lsmod`. `lsmod` выводит список модулей в обратном порядке, в котором они загружались, то есть первые модули в списке были загружены последними. Заметим, что это работает только для устройств, которые ядро нумерует стабильным порядком (как устройства PCI).

Однако на этот порядок также влияет удаление и перезагрузка модулей после первоначальной загрузки. Также некоторые драйвера могут быть включены в ядро статически, и их названия не появятся в результатах `lsmod`. Вы можете определить имена таких драйверов и их порядок загрузки, проанализировав файл `/var/log/kern.log` или результат команды `dmesg`.

Добавьте имена этих модулей в файл `/etc/initramfs-tools/modules` в том порядке, в котором они должны быть загружены во время запуска машины. Некоторые имена модулей в `etch` могут называться не так как в `sarge`. Например, `sym53c8xx_2` теперь `sym53c8xx`.

Затем вам нужно будет регенерировать образ(ы) `initramfs` запустив `update-initramfs -u -k all`.

---

<sup>5</sup>Для назначения постоянных имён сетевым интерфейсам правила генерируются автоматически сценарием `/etc/udev/rules.d/z45_persistent-net-generator.rules`. Удалите эту символическую ссылку для выключения постоянного именования сетевых устройств пакетом `udev`.

После загрузки ядра из `etch` и `udev`, вы можете перенастроить доступ к дискам системы через псевдонимы, которые не зависят от порядка загрузки драйверов. Эти псевдонимы находятся в подкаталогах `/dev/disk/`.

#### 4.6.4 Проблемы с синхронизацией при загрузке

Если для запуска системы используется `initrd`, созданный `initramfs-tools`, то в некоторых случаях создание файлов устройств с помощью `udev` происходит слишком поздно, после отработки сценариев загрузки, которые с ними работают.

Обычно, признаком такого поведения является отказ загрузки системы из-за невозможности смонтировать корневую файловую систему и переход в отладочную оболочку. Но если проверить нужные устройства, то окажется, что в `/dev` всё есть. Это наблюдалось, когда корневая файловая система была расположена на диске USB или RAID, особенно при использовании `lilo`.

Эту проблему можно решить добавив параметр загрузки `rootdelay=9`. Значение задержки (в секундах) может потребоваться изменить.

### 4.7 Перед перезагрузкой

Когда команда `aptitude dist-upgrade` закончила работу, формально обновление завершено, но перед перезагрузкой следует проделать ещё несколько действий.

#### 4.7.1 Переход с `devfs`

Ядра Debian больше не содержат поддержку `devfs`, поэтому пользователям `devfs` нужно вручную преобразовать их системы перед обновлением до ядра из `etch`.

Если вы видите строку `'devfs'` в файле `/proc/mounts`, то почти наверняка используется `devfs`. Имена в стиле `devfs` во всех конфигурационных файлах нужно заменить на соответствующие имена в стиле `udev`. Имена в стиле `devfs` обычно есть в файлах `/etc/fstab`, `/etc/lilo.conf`, `/boot/grub/menu.lst` и `/etc/inittab`.

Подробности о возможных проблемах есть в отчёте об ошибке `#341152` (<http://bugs.debian.org/341152>).

#### 4.7.2 Перезапуск `lilo`

Если в качестве системного загрузчика используется `lilo` (загрузчик по умолчанию некоторых установок `sarge`), то настоятельно рекомендуется перезапустить `lilo` после обновления:

```
# /sbin/lilo
```

Заметим, что это нужно сделать даже если вы не обновляли ядро системы, так как вторая стадия lilo могла быть изменена при обновлении пакета.

Также, пересмотрите содержимое файла `/etc/kernel-img.conf` и убедитесь, что в нём есть строка `do_bootloader = Yes`. Это включит перезапуск системного загрузчика каждый раз после обновления ядра.

Если возникли какие-то проблемы при запуске lilo, проверьте символические ссылки на `vmlinuz` и `initrd` в `/`, а также содержимое файла `/etc/lilo.conf` на соответствие с ними.

Если вы забудете перезапустить lilo перед перезагрузкой или система случайно перезагрузится до того как вы смогли перезапустить lilo вручную, система может больше не загрузиться. При запуске системы вместо приглашения lilo вы увидите только LI<sup>6</sup>. Для восстановления работоспособности обратитесь к ‘Подготовка к откату’ на стр. 16.

### 4.7.3 Обновление mdadm

Для mdadm теперь нужен конфигурационный файл для составления MD массивов (RAID) с первоначального `gamdisk` и во время процесса инициализации системы. Прочитайте и выполните инструкции из `/usr/share/doc/mdadm/README.upgrading-2.5.3.gz` после обновления пакета и до перезагрузки. Последняя версия этого файла доступна по адресу <http://svn.debian.org/wsvn/pkg-mdadm/mdadm/trunk/debian/README.upgrading-2.5.3?op=file>; обратитесь к ней, если возникнут проблемы.

## 4.8 Подготовка к следующему выпуску

После обновления можно сделать несколько вещей для подготовки к следующему выпуску.

- Если используется `grub`, отредактируйте файл `/etc/kernel-img.conf` и измените нахождение программы `update-grub` с `/sbin/update-grub` на `/usr/sbin/update-grub`.
- Если новый метапакет образа ядра был стянут по зависимости от старого, то он будет помечен как установленный автоматически, что должно быть исправлено:

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

- Удалите метапакеты `sarge` для ядра, запустив:

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

- Перенесите все настройки из `/etc/network/options` в `/etc/sysctl.conf`. Подробности читайте в `/usr/share/doc/netbase/README.Debian`.
- Удалите устаревшие и неиспользуемые пакеты как это описано в ‘Устаревшие пакеты’ на стр. 34. Проверьте, какие конфигурационные файлы они использовали и подумайте над удалением пакетов вместе с их конфигурационными файлами.

---

<sup>6</sup>Подробнее о кодах ошибок lilo смотрите в The Linux Bootdisk HOWTO (<http://tldp.org/HOWTO/Bootdisk-HOWTO/a1483.html>).

## 4.9 Исключаемые пакеты

Начиная с выпуска Lenny, большое число серверных пакетов будет исключено, так как переход на новые версии теперь можно выполнить без проблем при переходе на Lenny.

Это относится к следующим пакетам:

- apache (1.x) на apache2
- bind8 на bind9
- php4 на php5
- postgresql-7.4 на postgresql-8.1
- exim 3 на exim4

## 4.10 Устаревшие пакеты

Вместе с появлением девяти тысяч новых пакетов, более двух тысяч пакетов из состава sarge с появлением etch устаревают. Устаревшие пакеты не включены в etch и не будут больше обновляться. Ничто не мешает вам продолжить пользоваться этими пакетами, но приблизительно через год <sup>7</sup> после выхода etch команда безопасности прекратит выпускать для них исправления. Мы рекомендуем перейти на использование альтернативных пакетов, если есть такая возможность.

Пакет может быть удалён из дистрибутива по многим причинам: разработка программы прекращена, пакет не интересен никому из разработчиков Debian, программа вытеснена другой, более функциональной программой или версией программы, пакет содержит ошибки, из-за которых пока не может быть включён в etch. В последнем случае пакет может быть доступен в нестабильном (“unstable”) дистрибутиве.

Определить, какие пакеты в обновлённой системе устарели очень легко, так как программы управления помечают их как “устаревшие”. Если вы используете aptitude, то полный список этих пакетов находится в разделе “Устаревшие и локально созданные пакеты”. В программе dselect есть аналогичный раздел, но список пакетов в нём может несколько отличаться. Также, если в sarge вы использовали aptitude для установки пакетов вручную, то aptitude помнит об этих пакетах, и сможет пометить как устаревшие те пакеты, которые были установлены только для удовлетворения зависимостей и которые больше не нужны, если пакет, который от них зависел, был удалён. В отличие от deborphan, aptitude не помечает устаревшими пакеты, которые вы установили вручную, в отличие от тех, которые были установлены автоматически для удовлетворения зависимостей.

Существуют и другие инструменты для определения неиспользуемых пакетов. В их число входят deborphan, debfoster и cruft. Советуем использовать deborphan, хотя по умолчанию

---

<sup>7</sup> Или при появлении нового выпуска Debian. Обычно поддержку получают одновременно не более двух стабильных выпусков.

он ищет только ненужные библиотеки (пакеты в секциях “libs” и “oldlibs”, от которых не зависит ни один другой пакет). Не удаляйте подряд все пакеты, найденные такими инструментами (в особенности, если при запуске вы указали опции, отличные от опций по умолчанию), чтобы не удалить что-нибудь нужное. Настоятельно рекомендуем перед удалением внимательно просмотреть список предложенных пакетов (прочтеть описание, посмотреть входящие файлы).

В системе отслеживания ошибок Debian (<http://bugs.debian.org/>) зачастую содержится дополнительная информация о причинах удаления пакета. Просмотрите архив сообщений об ошибках интересующего вас пакета, а также архивированные ошибки для псевдо-пакета `ftp.debian.org` (<http://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=ftp.debian.org#38:archive=yes>).

#### 4.10.1 Пакеты-пустышки

Многие пакеты из `sarge` в `etch` были поделены на несколько разных пакетов, чаще всего чтобы облегчить поддержку системы. Для простоты обновления в состав `etch` нередко входит “пакет-пустышка” (пустой пакет с именем старого пакета из `sarge`). Такой пакет при установке устанавливает новые пакеты, от которых зависит. Сразу же после установки “пустышки” становятся неиспользуемыми и их можно спокойно удалять.

В описании большинства (но не всех) пакетов-пустышек ясно указано их предназначение. К сожалению, никаких правил для описания пустышек нет, поэтому для их обнаружения вам может пригодиться `deborphan` с опцией `—guess`. Некоторые пустышки удалять после обновления не нужно, так как они впоследствии используются для определения установленной версии программы.





## Глава 5

# Что надо знать о etch

### 5.1 Возможные проблемы

Иногда перемены к лучшему приводят к побочным эффектам, которых нельзя избежать без появления ошибок где-нибудь ещё. Здесь мы опишем проблемы, которые уже известны. Прочитайте также список известных ошибок, соответствующую документацию на пакеты, отчёты об ошибках и другую информацию, указанную в ‘Что ещё можно прочитать’ на стр. 47.

#### 5.1.1 Проблемы с устройствами из-за udev

Хотя udev и был хорошо протестирован, у вас могут возникнуть вопросы, связанные работой некоторых устройств. Чаще всего ошибки вызваны изменением владельца или разрешений файла-устройства. Некоторые устройства по умолчанию не могут быть созданы (например, /dev/video и /dev/radio).

Все эти проблемы можно решить настройкой udev. Более подробная информация - в udev(8) и /etc/udev.

#### 5.1.2 Некоторые сайты в сети недоступны по TCP

Начиная с версии 2.6.17, Linux использует изменение окна TCP, описанное в RFC 1323, в агрессивной манере. Некоторые серверы не удовлетворяют RFC и объявляют неверные размеры своих окон. Подробнее смотрите ошибки #381262 (<http://bugs.debian.org/381262>), #395066 (<http://bugs.debian.org/395066>), #401435 (<http://bugs.debian.org/401435>).

Чтобы обойти эту проблему, обычно применяют два способа: или уменьшают максимально возможные значения окна TCP до самого маленького значения (предпочтительный способ), или выключают изменение окна TCP совсем ( не рекомендуется). Примеры команд можно найти на странице известных недоработок debian-installer (<http://www.debian.org/devel/debian-installer/errata>).

### 5.1.3 Медленно обновляются пакетные индексные файлы АРТ

По умолчанию в версии apt из etch используется новый способ обновления пакетных индексных файлов АРТ (при запуске `aptitude update`), при котором скачиваются файлы изменений (вместо полного индексного файла пакетов), называемые `pdiff`. Данная новая возможность должна использовать меньшую ширину канала и работать быстрее на большинстве систем. К сожалению, есть и обратный эффект: обновление замедляется на системах с быстрым подключением к сети (или есть близко расположенное сервер-зеркало), для которых обновление выполняется редко, так как для этих систем слияние файлов изменений занимает больше времени, чем загрузка полного индекса пакетов. Эту возможность можно отключить, добавив `Acquire::Pdiffs "false"`; в конфигурационный файл `/etc/apt/apt.conf`.

Это изменение, в основном, предназначено для пользователей нестабильной и тестируемой веток Debian GNU/Linux в следствие постоянного изменения этих архивов. Пользователи etch заметят эту возможность только при обновлении состояния пакетов из архива безопасности.

### 5.1.4 Асинхронная инициализация сети может приводить к непредсказуемому поведению

В системах, использующих `udev` для загрузки драйверов для сетевых интерфейсов, из-за асинхронной природы `udev` может случиться так, что сетевой драйвер не будет загружен до запуска `/etc/init.d/networking` при загрузке системы. Хотя включение `allow-hotplug` в `/etc/network/interfaces` (в дополнении к `auto`) обеспечит, что сетевой интерфейс будет включён сразу как станет доступным, но не гарантируется, что это завершится перед началом старта сетевых сервисов, некоторые из которых могут повести себя неправильно при отсутствии сетевого интерфейса.

### 5.1.5 Проблема при использовании WPA-безопасности в беспроводных сетях

В `sarge`, пакет `wpa_supplicant` устанавливался как системный сервис и настраивался через `/etc/default/wpa_supplicant` и `/etc/wpa_supplicant.conf` (настраивался пользователем).

В `etch`, сценарий `/etc/init.d/wpa_supplicant` был удалён, а пакет Debian теперь интегрирован с `/etc/network/interfaces`, как и другие пакеты типа `wireless-tools`. Это означает, что `wpa_supplicant` больше не предоставляет системный сервис явно.

Информацию по настройке `wpa_supplicant` можно найти в `/usr/share/doc/wpa_supplicant/README.modes.gz`, там же можно найти несколько примеров для `/etc/network/interfaces`. Обновляемую информацию об использовании пакета `wpa_supplicant` в Debian можно найти в Debian вики (<http://wiki.debian.org/WPA>).

### 5.1.6 Проблемы с не-ASCII символами в именах файлов

При использовании смонтированных файловых систем `vfat`, `ntfs` или `iso9660`, в которых содержатся файлы в именах которых есть не-ASCII символы, возникает ошибка, если мон-

тирование производилось без параметра `utf8`. Это может проявляться сообщениями вида: `'Invalid or incomplete multibyte or wide character'`. Возможным решением является использование `defaults,utf8` в качестве параметров монтирования файловых систем `vfat`, `ntfs` и `iso9660`, если они содержат имена файлов с не-ASCII символами.

Заметим, что ядро Linux поддерживает только регистрозависимую обработку имён файлов в `vfat` при использовании параметра `utf8`.

### 5.1.7 Повреждение данных в оборудовании IOMMU на чипсетах Nvidia

Проблема была обнаружена в системах AMD64 с чипсетами Nvidia и имеющими более 3Гб оперативной памяти. Она заключается в спонтанных повреждениях данных при использовании оборудования IOMMU. Эта проблема всё ещё исследуется разработчиками ядра Linux и производителями оборудования, и официального исправления пока не выпущено. До тех пор, пока решение не найдено, для защиты целостности своих данных, пользователям таких систем предлагается вручную выключить использование оборудования IOMMU перед загрузкой, добавив `iommu=soft` в загрузочные параметры ядра.

Подробности об этой проблеме доступны в Debian ошибке #404148 (<http://bugs.debian.org/404148>) и ошибке Linux Kernel #7768 ([http://bugzilla.kernel.org/show\\_bug.cgi?id=7768](http://bugzilla.kernel.org/show_bug.cgi?id=7768)).

### 5.1.8 Перестал работать звук

В редких случаях после обновления может перестать работать звук. Если это случилось, проверьте `alsa`: запустите `alsacnf` с правами суперпользователя, добавьте своего пользователя в группу `audio`, с помощью `alsamixer` убедитесь, что выставлен приемлемый уровень громкости и звук не выключен, убедитесь, что сервисы `arts` и `esound` остановлены, проверьте, что модули OSS не загружены, проверьте, что колонки включены, проверьте, что команда `cat /dev/urandom > /dev/dsp` работает с правами суперпользователя.

## 5.2 Переход с XFree86 на X.Org

Переход на X.Org приносит некоторые структурные изменения. В случае, если установлены все пакеты из Debian и также включённые в etch, то обновление должно пройти без проблем. Опыт, однако, показывает, что нужно сделать несколько изменений, так как известно, что без них могут возникнуть проблемы во время обновления.

Наиболее важным изменением является то, что каталог `/usr/X11R6/bin` больше не существует и остаётся только в виде символической ссылки на `/usr/bin`. Это значит, что этот каталог будет пуст в момент установки новых пакетов. Новые пакеты конфликтуют с большинством пакетов, которые используют `/usr/X11R6/bin`, но в некоторых случаях может понадобиться ручное вмешательство. Не запускайте обновление в X-сессии.

В случае, если обновление прервалось во время установки X.Org, убедитесь, не осталось ли файлов в `/usr/X11R6/bin`. Вы можете использовать `dpkg -S`, чтобы выяснить, какому пакету

Debian принадлежит оставшийся файл (если таковые имеются), и удалить эти пакеты с помощью `dpkg --remove`. Запомните какие пакеты удаляете, для того чтобы позже можно было установить пакеты, заменяющие их. Перед тем как продолжить обновление, нужно удалить все файлы в каталоге `/usr/X11R6/bin`.

Подробности и другие проблемы можно найти на <http://wiki.debian.org/Xorg69To7>.

Если у вас возникли проблемы с X.Org после перезапуска, стоит попробовать также перезапустить сервер шрифтов выполнив команду `/etc/init.d/xfs restart`. Это случается из-за того, что в файле `/etc/X11/fs/xfs.options` содержится строка `no-restart-on-upgrade`, а пути к шрифтам изменились.

### 5.3 Во многих приложениях нет поддержки 8-битного отображения цветов

После обновления X.Org и библиотек, X-терминалы, поддерживающие только 8-ный цвет, работать не будут. Это происходит из-за векторной графической библиотеки Cairo 2D (`libcairo2`), которая не поддерживает 8-ный псевдо-цвет. Эта библиотека используется рабочими столами GNOME и Xfce, а также многими приложениями для рабочих столов, собранных с инструментарием Gtk2+, например `abiword`.

Известно, что такое происходит с некоторыми машинами Sun и X-терминалами Tektronix, NCD, IBM и SGI, а также некоторыми другими удалёнными оконными системами X. Если возможно, вам нужно настроить в этих терминалах использование 16-ного цвета.

Подробности доступны в сообщениях об ошибках Freedesktop bug #4945 ([https://bugs.freedesktop.org/show\\_bug.cgi?id=4945](https://bugs.freedesktop.org/show_bug.cgi?id=4945)).

### 5.4 Переход с `exim` на `exim4`

Одним из устаревших пакетов к выпуску etch стал агент пересылки почты (MTA) `exim`, который был заменён полностью новым пакетом `exim4`.

Программа `exim` (версии 3.xx) не поддерживалась разработчиком несколько лет, и поэтому в Debian также была прекращена поддержка этой версии. Если вы всё ещё используете `exim 3.xx`, обновите установленный `exim` до `exim4` вручную. Так как `exim4` уже был в `sarge`, вы можете выполнить обновление в системе `sarge` перед тем как обновляться до etch, или после обновления до etch, как хотите. Помните, что старый пакет `exim` обновлён не будет и для него не будет обеспечиваться поддержка безопасности после того, как поддержка `sarge` будет прекращена.

Заметим, что в зависимости от вашей настройки `debconf`, вам могут быть не заданы вопросы во время установки `exim4`. Если вопросы не задавались, по умолчанию система будет настроена на 'локальную доставку'. Для перенастройки используйте команду `dpkg-reconfigure exim4-config`.

Пакеты `exim4` в Debian очень хорошо документированы. Домашняя страница пакетов в Debian Wiki <http://wiki.debian.org/PkgExim4>, файл README можно найти на странице <http://pkg-exim4.alioth.debian.org/README/README.Debian.html>, а также внутри пакетов.

В файле README есть глава Packaging, в которой описаны различия пакетов, которые мы предлагаем, также есть глава об обновлении с Exim 3, которая поможет вам выполнить переход.

## 5.5 Обновление apache2

Apache был обновлён до новой версии 2.2. Хотя это не должно ничего сломать в большинстве случаев, есть несколько возможных проблем, с которыми лучше ознакомиться.

На странице <http://httpd.apache.org/docs/2.2/upgrading.html> описаны изменения, сделанные разработчиком. Прочитайте эту страницу и особенно обратите внимание на:

- нужно перекомпилировать все модули
- были пересортированы и переименованы модули авторизации
- были переименованы некоторые параметры конфигурации

Из изменений сделанных Debian следует отметить, что строка SSL больше не определена, так как ssl теперь поддерживается в пакете по умолчанию.

Если вы используете экспериментальный ИТК MPM (из пакета `apache2-mpm-itk`), то по умолчанию модуль `cgi` не будет включён по умолчанию. Чтобы его правильно включить, вам нужно вручную выключить `mod_cgid` и включить `mod_cgi`:

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

## 5.6 Обновление Zope и Plone

Был обновлён Zope и все связанные с ним пакеты. Также многие продукты были удалены (из-за устаревания или из-за несовместимости с новыми версиями Zope, CMF или Plone).

К сожалению, нет лёгкого и гарантирующего способа обновления сложного сервера zope или plone. Хотя и Plone включает инструмент перехода, опыт показал, что автоматическая миграция может легко завершиться неудачно.

По этой причине, пользователям рекомендуется устанавливать свои системы так, чтобы они могли продолжать работать со старой установкой `sarge Zope/Plone` одновременно с новыми версиями `etch` на время тестирования результата миграции.

Самый лёгкий и безопасный способ сделать это — скопировать свою систему sarge на другой жёсткий диск или раздел, а затем обновить только одну из двух копий. После этого вы можете использовать chroot для запуска версии из sarge параллельно с версией из etch.

Нельзя установить старую и новую версию Zope/Plone вместе на системе etch, в частности, из-за того, что старые пакеты зависят от python2.3, который не может быть установлен вместе с python2.4.

## 5.7 Раскрытие шаблонов (подстановка) в GNU tar

В предыдущих версиях GNU tar предполагалось использование подстановки в стиле оболочек командной строки при извлечении файлов из архива. Например, по команде:

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

извлекутся все файлы, чьи имена заканчиваются на '.c'. Такое поведение не было описано и не совместимо с традиционными реализациями tar. Поэтому, начиная с версии 1.15.91, GNU tar больше не использует подстановку по умолчанию. Например, вызов предыдущей команды будет рассмотрен как запрос на извлечение из архива файла с именем '\*.c'.

Подробнее смотрите в `/usr/share/doc/tar/NEWS.gz`.

## 5.8 NIS и Network Manager

Версия urbind, включённая в nis для etch, содержит поддержку Network Manager. Эта поддержка заставляет urbind выключать функциональность клиента NIS, если Network Manager сообщает, что компьютер отключён от сети. Когда Network Manager не используется, он сообщает, что компьютер отключён, поэтому пользователи NIS с системным клиентом NIS должны проверить, что поддержка Network Manager выключена на этих системах.

Для этого можно удалить пакет network-manager, или отредактировать `/etc/default/nis`, добавив `-no-dbus` в `YPBINDARGS`.

Использование `-no-dbus` добавляется по умолчанию при новой установке Debian, но не при обновлении с предыдущих выпусков.

## 5.9 Запрещены небезопасные конфигурации php

Многие годы было известно, что включение настройки `register_globals` в PHP ненадёжно и опасно, и в пакете она была отключена по умолчанию. Теперь эту настройку вообще не рекомендуется использовать на системах Debian, как слишком опасную. То же самое касается недостатков для `safe_mode` и `open_basedir`, которым также некоторое время не уделялось должного внимания.

Начиная с этого выпуска, команда обеспечения безопасности в Debian не предоставляет поддержку безопасности тем настройкам PHP, которые заранее считаются небезопасными. Самое главное: проблемы, возникшие из-за включённого `register_globals`, больше не рассматриваются.

Если вы используете старые приложения, которым требуется `register_globals`, включайте его только для соответствующего пути, например, через конфигурационный файл Apache. Подробности описаны в файле `README.Debian.security` из каталога с документацией PHP (`/usr/share/doc/php4`, `/usr/share/doc/php5`).

## 5.10 Безопасность в продуктах Mozilla

Программы Mozilla `firefox` и `thunderbird` (версии без бранда в Debian называются `iceweasel` и `icedove`, соответственно), являются важными инструментами для многих пользователей. К сожалению, политика безопасности разработчиков Mozilla принуждает пользователей обновляться до новой версии, которую выпустил разработчик, что противоречит политике Debian не делать больших функциональных изменений при обновлениях безопасности. Мы не можем ничего предсказать, но во время жизни `etch` команда безопасности Debian может подойти к такому моменту, когда поддержка продуктов Mozilla станет невозможной, и ей придётся объявить о прекращении обеспечения безопасности продуктов Mozilla. Вы должны принять это во внимание при развёртывании Mozilla и рассмотреть альтернативы, имеющиеся в Debian, если отсутствие поддержки в области безопасности создаст проблему для вас.

## 5.11 Рабочий стол KDE

В версии KDE, доступной в `etch`, была изменена обработка носителей: вместо `device:/` используется `media/`. В некоторых пользовательских конфигурационных файлах могут храниться ссылки `device:/`, которые нужно преобразовать. Особенно это видно по файлу `~/.kde/share/apps/konqsidebar/tng/virtual_folders/services`, который содержит такую ссылку и может быть безопасно удалён, так как не создаётся при настройке у нового пользователя.

Многое изменилось в окружении рабочего стола KDE в версии из `etch` по сравнению с версией из `sarge`. Более подробную информацию вы можете найти в информации о выпуске KDE 3.5 (<http://www.kde.org/announcements/announce-3.5.php>).

## 5.12 Поддержка и изменения рабочего стола GNOME

Если в `sarge` вы использовали рабочий стол GNOME, то вы не получите некоторых новых изменений, предоставленных конфигурацией по умолчанию в Debian `etch`. В некоторых крайне редких случаях рабочий стол GNOME может неправильно обработать ваши старые настройки и может неправильно функционировать.



Если вам не очень жалко своих настроек рабочего стола GNOME, то может быть лучше переименовать домашний пользовательский каталог `.gconf` (например в `.gconf.old`) для того, чтобы он пересоздался с настройками по умолчанию для etch при запуске новой сессии.

В выпуске etch Debian больше нет пакетов устаревшей версии 1 выпуска GNOME, хотя некоторые пакеты остались, чтобы обеспечить поддержку другим пакетам Debian, которые ещё не были адаптированы к версии GNOME 2. Пакеты для GTK1.2 полностью поддерживаются.

Многое изменилось в окружении рабочего стола GNOME в версии из etch по сравнению с версией из sarge. Более подробную информацию вы можете найти в информации о выпуске GNOME 2.14 (<http://www.gnome.org/start/2.14/notes/en/>).

### 5.13 Редактор по умолчанию

Если в качестве редактора по умолчанию использовался `vim`, то при обновлении он может быть изменён на `nano`.

Если администраторы хотят изменить редактор по умолчанию для всех пользователей, то для этого нужно использовать команду обновления альтернатив системы:

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

Пользователи могут изменить редактор по умолчанию определив переменную окружения `EDITOR`, вставив следующие строки в свои файлы профилей:

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

### 5.14 Сообщение дня

Файл `/etc/motd` теперь является символической ссылкой на `/var/run/motd`, который создаётся с помощью `/etc/init.d/bootmisc.sh` из шаблона, `/etc/motd.tail`, при каждой перезагрузке. Это означает, что изменения, сделанные в `/etc/motd` будут потеряны. Изменения в файле `/etc/motd.tail` не применяются автоматически к `/etc/motd`, это происходит только при перезагрузке.

Также, переменная `EDITMOTD` в файле `/etc/default/rcS` больше не используется. Если вы хотите выключить обновление `motd` или готовить сообщение дня самостоятельно, то просто измените символическую ссылку `/etc/motd` на другой файл, например, на `/etc/motd.static`, и редактируйте его как хотите.

## 5.15 Нет поддержки unicode в emacs21\* по умолчанию

Emacs21 и emacs21-нох не настроены по умолчанию на поддержку использования Unicode. Подробнее об этом и как это обойти, смотрите ошибку #419490 (<http://bugs.debian.org/419490>).



## Глава 6

# Дополнительная информация о Debian GNU/Linux

### 6.1 Что ещё можно прочитать

Помимо этой информации о выпуске и руководства по установке, существует документация по Debian GNU/Linux, разрабатываемая проектом документации Debian (DDP), целью которого является создание высококачественной документации для пользователей и разработчиков Debian. В состав этой документации входят руководство по Debian, руководство нового сопровождающего Debian, FAQ по Debian и многое другое. Полную информацию о доступных документах можно посмотреть на веб-сайте DDP (<http://www.debian.org/doc/ddp>).

Документация по конкретным пакетам устанавливается в каталог `/usr/share/doc/пакет`. Там может находиться информация об авторских правах, специфичная для Debian информация и документация от авторов исходной программы.

### 6.2 Если нужна помощь

Пользователи Debian могут воспользоваться помощью, советами и поддержкой из разных источников, но к ним следует прибегать, только не удалось найти ответа на вопрос в документации. Этот раздел содержит краткое описание дополнительных источников помощи для новых пользователей.

#### 6.2.1 Списки рассылки

Основной интерес для пользователей Debian представляют англоязычный список рассылки `debian-user` и списки `debian-user-язык` или `debian-язык` для других языков. Для русского языка это список `debian-russian`. Информацию о списках рассылки и о том, как на них подписаться смотрите по адресу <http://lists.debian.org/>. Пожалуйста, перед тем, как отправить

вопрос в список рассылки, поищите ответ на него в архивах. Просим также придерживаться общепринятых норм почтового этикета.

### 6.2.2 IRC

Для поддержки пользователей Debian есть IRC-канал, расположенный в сети OFTC IRC. Чтобы войти на канал, соединитесь с сервером `irc.debian.org` с помощью своего любимого IRC-клиента и присоединитесь к каналу `#debian`. Для русскоязычных пользователей существует канал `#debian-russian` в той же сети.

Просим вас следовать правилам поведения на канале и уважать других пользователей. Правила поведения на канале описаны в вики Debian (<http://wiki.debian.org/DebianIRC>).

Более подробную информацию об OFTC можно получить на её веб-сайте (<http://www.oftc.net/>).

## 6.3 Как сообщить об ошибке

Мы приложили немало усилий, чтобы сделать Debian GNU/Linux операционной системой высокого качества, однако это не означает, что поставляемые нами пакеты не содержат абсолютно никаких ошибок. Мы предоставляем нашим пользователям полную информацию обо всех обнаруженных ошибках в нашей системе отслеживания ошибок (Bug Tracking System, BTS), расположенной по адресу `bugs.debian.org` (<http://bugs.debian.org/>). Такой подход согласуется с концепцией открытой разработки Debian.

Если вы обнаружите ошибку в дистрибутиве или каком-то из его пакетов, пожалуйста, сообщите о ней, чтобы в следующем выпуске она была исправлена. Чтобы сообщить об ошибке, требуется работающий адрес электронной почты. Это необходимо для того, чтобы мы могли отслеживать ошибки, а разработчики связываться с отправителями отчётов об ошибках, если им понадобится дополнительная информация.

Отправить сообщение об ошибке можно с помощью программы `reportbug` или вручную по электронной почте. Более подробную информацию о системе отслеживания ошибок и о том, как её использовать, можно найти в каталоге `/usr/share/doc/debian` после установки пакета `doc-debian` или на сайте системы отслеживания ошибок (<http://bugs.debian.org/>).

## 6.4 Как помочь Debian

Чтобы помочь Debian, не нужно быть большим специалистом. Помогая пользователям в решении их проблем в списках рассылки (<http://lists.debian.org/>) вы уже помогаете обществу. Выявление (или, что ещё лучше, решение) проблем, связанных с разработкой дистрибутива, участие в работе списков для разработчиков (<http://lists.debian.org/>), также очень важно. Чтобы помочь поддерживать высокое качество Debian, отправляйте сообщения об ошибках (<http://bugs.debian.org/>) и помогайте разработчикам отслеживать и исправлять их. Если вы хорошо владеете языком, то можете помочь в написании документа-

ции (<http://www.debian.org/doc/ddp>) или переводе (<http://www.debian.org/international/>) существующей документации на родной язык.

Если вы можете уделить Debian немного времени, можно взяться за сопровождение какой-нибудь свободной программы в Debian. В частности, полезной была бы поддержка пакетов, которые кто-то особенно хотел бы видеть в Debian. Подробную информацию можно найти на странице требующих доработки и планируемых пакетов (<http://www.debian.org/devel/wnpp/>). Если вам интересна какая-то специфическая группа пользователей, возможно, вы захотите участвовать в одном из дочерних проектов Debian. Среди них перенос Debian на другие архитектуры, проекты Debian для детей (<http://www.debian.org/devel/debian-jr/>) и Debian в медицине (<http://www.debian.org/devel/debian-med/>).

В любом случае, если вы участвуете в работе сообщества Свободного ПО в качестве пользователя, программиста, писателя или переводчика, вы уже помогаете Свободному программному обеспечению. Такое участие приносит удовольствие и радость, даёт возможность познакомиться с новыми людьми.



## Приложение А

# Управление системой sarge

В этом приложении содержится информация о том, как установить или обновить пакеты sarge перед обновлением до etch. В некоторых случаях это может понадобиться.

### А.1 Обновление системы sarge

В основном это обновление ничем не отличается от всех предыдущих обновлений sarge, которые вы делали. Единственное отличие состоит в том, что сначала надо убедиться, что ваши списки пакетов всё ещё содержат пакеты sarge, как описано в разделе ‘Проверка списка источников АРТ’ на стр. 51.

Если вы обновляете систему с сервера-зеркала Debian, то автоматически будет выполнено обновление до последнего выпуска sarge.

### А.2 Проверка списка источников АРТ

Если одна или несколько записей в вашем `/etc/apt/sources.list` ссылаются на стабильный дистрибутив (`'stable'`), то вы уже “используете” etch. Если вы уже запустили `apt-get update`, то ещё не поздно всё отменить, просто выполнив инструкцию, приведённую ниже.

Если вы успели установить пакеты из etch, то особого смысла в установке пакетов из sarge уже нет. В этом случае вам следует решить — довести до конца обновление или нет. Вернуться к использованию старых версий пакетов возможно, но эта процедура выходит за рамки данного документа.

Откройте файл `/etc/apt/sources.list` с помощью любого текстового редактора (потребуется привилегии суперпользователя). Во всех строках, начинающихся с `deb http:` или с `deb ftp:` замените `“stable”` на `“sarge”`.

Если вы нашли строки, начинающиеся с `deb file:`, то вам придётся самим проверить, какие пакеты хранятся в указанном каталоге — sarge или etch.



Внимание! Не исправляйте строки, начинающиеся с `deb cdrom:`! Если вы исправите такую строку, то вам придётся снова запускать `apt-cdrom`. Не беспокойтесь, если источник “`cdrom`” ссылается на нестабильный (“`unstable`”) дистрибутив. Как это ни странно, так и должно быть.

Если вы внесли какие-нибудь изменения, сохраните файл и выполните команду

```
# apt-get update
```

для обновления списка пакетов.