

Debian GNU/Linux 5.0 ("lenny"), AMD64 的发行说明

The Debian Documentation Project
[<http://www.debian.org/doc/>]

目录

1. 简介	1
1.1. 报告本文档的 bug	1
1.2. 贡献升级报告	1
1.3. 本文档的源文件	2
2. Debian GNU/Linux 5.0 有什么新东西	3
2.1. 这次发布中有什么新东西?	3
2.1.1. 包管理	4
2.1.2. proposed-updates 区	5
2.2. 系统改进	5
2.3. Debian Live	6
2.4. 内核相关的主要变化	6
2.4.1. 内核打包的变化	6
2.5. Emdebian 1.0 (based on Debian GNU/Linux lenny 5.0)	7
2.6. Netbook support	7
2.7. Java now in Debian	7
3. 安装系统	8
3.1. 安装系统有哪些新特性?	8
3.1.1. 主要更新	8
3.1.2. 自动安装	9
3.1.3. Install Debian GNU/Linux with a Braille display	9
3.1.4. Install Debian GNU/Linux with a hardware speech synthesis	10
4. 从以前的发行版升级	11
4.1. 升级前的准备	11
4.1.1. 备份数据和配置文件	11
4.1.2. 提前告知用户	11
4.1.3. 准备恢复	11
4.1.4. 为升级准备安全环境	12
4.1.5. 为 LILO 准备 initramfs	13
4.2. 检查系统状态	13
4.2.1. 复审包管理器中的未决操作	13
4.2.2. 禁用 APT pinning 操作	14
4.2.3. 检查包状态	14
4.2.4. proposed-updates 区	15
4.2.5. 非官方源和 backports	15
4.3. 手工删除软件包的标记	15
4.4. 为 APT 准备源	16
4.4.1. 添加互联网 APT 源	16
4.4.2. 添加本地镜像 APT 源	17
4.4.3. 从 CD-ROM 或 DVD 添加 APT 源	17
4.5. 升级软件包	18
4.5.1. 记录会话	18
4.5.2. 更新包列表	18
4.5.3. 确保您有足够的空间升级	19
4.5.4. 先升级 apt 和(或) aptitude	20
4.5.5. 使用 aptitudes 记录的 apt 自动安装的软件包列表	21
4.5.6. 最小系统升级	21

4.5.7. 升级系统的其它部分	22
4.5.8. 升级期间可能遇到的问题	22
4.6. 升级内核与相关包	23
4.6.1. 安装内核元数据包	24
4.6.2. 从 2.6 内核升级	24
4.6.3. 从 2.4 内核升级	24
4.6.4. 设备枚举的顺序	25
4.6.5. 引导期间的问题	25
4.7. 在重启之前要做的事情	26
4.7.1. 重新运行 lilo	26
4.8. 系统引导时在显示 Waiting for root file system 后挂起	26
4.8.1. 如何在升级前避免此问题	27
4.8.2. 如何在升级后解决此问题	28
4.9. 为下个发行版做准备	30
4.10. 不赞成地包	31
4.11. 过时的包	31
4.11.1. 哑包	32
5. lenny 中需要注意的问题	33
5.1. 潜在的问题	33
5.1.1. 与 udev 相关的设备的问题	33
5.1.2. 某些应用程序可能无法在 2.4 内核下工作	33
5.1.3. 某些网站无法通过 TCP 进行连接	33
5.1.4. 无法自动关机	33
5.1.5. 异步网络初始化可能引起无法预测的情况	33
5.1.6. 使用 WPA 保护无线网络的问题	34
5.1.7. 文件名中使用非 ASCII 字符可能引起问题	34
5.1.8. 声音停止工作	34
5.2. NFS 挂载现在由 nfs-common 来处理	35
5.3. 罗马尼亚(ro)键盘布局的改变	35
5.4. 升级 apache2	35
5.5. NIS 和网络管理器	35
5.6. Mozilla 产品的安全状态	36
5.7. GNOME 桌面改变与支持	36
5.8. Emacs21* 默认不支持 Unicode	36
5.9. slurpd/replica 已经被删除	36
5.10. 桌面没有使用全部屏幕	36
5.11. DHCP failover issue	37
6. 关于 Debian GNU/Linux 的更多信息	38
6.1. 更多读物	38
6.2. 获得帮助	38
6.2.1. 邮件列表	38
6.2.2. 互联网在线聊天系统	38
6.3. 报告 Bug	38
6.4. 为 Debian 做贡献	39
A. 管理您的 etch 系统	40
A.1. 升级您的 etch 系统	40
A.2. 检查您的 sources list	40
B. Contributors to the Release Notes	41

C. Lenny dedicated to Thiemo Seufer	42
索引	43
术语表	45

第 1 章 简介

本文档告知 Debian GNU/Linux 发行版的用户，关于版本 5.0 (代号 "lenny") 的主要变化。

The release notes provide information on how to upgrade safely from release 4.0 (codenamed etch) to the current release and inform users of known potential issues they could encounter in that process.

You can get the most recent version of this document from <http://www.debian.org/releases/lenny/releasenotes>. If in doubt, check the date on the first page to make sure you are reading a current version.



小心

注意列出所有已知的问题是不可能的，因此选择了那些普遍存在和造成影响的问题。

请注意我们只支持从 Debian 的前一版本 (现在要升级的版本是 4.0) 升级。如果您需要从更早的版本升级，建议您先查看前一版本的发行说明，先升级到 4.0。

1.1. 报告本文档的 bug

We have attempted to test all the different upgrade steps described in this document and to anticipate all the possible issues our users might encounter.

Nevertheless, if you think you have found a bug (incorrect information or information that is missing) in this documentation, please file a bug in the [bug tracking system](http://bugs.debian.org/) [http://bugs.debian.org/] against the `release-notes` package.

1.2. 贡献升级报告

我们欢迎来自用户的关于从 etch 升级到 lenny 的任何信息。如果您愿意分享，请在 [Bug 追踪系统](http://bugs.debian.org/) [http://bugs.debian.org/] 填写一个关于 `release-notes` 软件包的 Bug 以及您所获得的结果。希望您能够 (使用 **gzip**) 压缩任何附件。

当您提交升级报告的时候，请包含以下信息：

- 在升级前后，软件包数据库的状态：**dpkg** 的数据库状态信息在 `/var/lib/dpkg/status`，**aptitude** 的数据库状态信息在 `/var/lib/aptitude/pkgstates`。在升级之前，您应该已经象 [第 4.1.1 节 “备份数据和配置文件”](#) 描述的那样备份，当然您也可以在 `/var/backups` 找到备份信息。
- Session logs created using **script**, as described in [第 4.5.1 节 “记录会话”](#)。
- apt 的日志 `/var/log/apt/term.log`，或 **aptitude** 的日志 `/var/log/aptitude`。



注意

在提交 bug 报告之前，您应该花点时间从日志中检查和删除任何敏感和/或机密信息，因为这些信息都会被发布在公开的数据库中。

1.3. 本文档的源文件

The source of this document is in DocBook XML format. The HTML version is generated using `docbook-xsl` and `xsltproc`. The PDF version is generated using `dblatex` or `xmlroff`. Sources for the Release Notes are available in the SVN repository of the *Debian Documentation Project*. You can use the [web interface](http://svn.debian.org/viewsvn/ddp/manuals/trunk/release-notes/) [http://svn.debian.org/viewsvn/ddp/manuals/trunk/release-notes/] to access its files individually through the web and see their changes. For more information on how to access the SVN please consult the [Debian Documentation Project SVN information pages](http://www.debian.org/doc/cvs/) [http://www.debian.org/doc/cvs/].

第 2 章 Debian GNU/Linux 5.0 有什么新东西

[Wiki](http://wiki.debian.org/NewInLenny) [<http://wiki.debian.org/NewInLenny>] 有更多关于这个主题的信息。

这个版本中加入了对 ARM EABI (armel) 的官方支持。

下面是 Debian GNU/Linux lenny 官方支持的架构：

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

您可以在 [Debian 移植页面](http://www.debian.org/ports/) [<http://www.debian.org/ports/>]，读到更多的关于您的架构的移植状态和移植细节的信息。

2.1. 这次发布中有什么新东西？

Debian 的这次发行再次带来了比它前个发行版本 `etch` 更多的软件；包括 7700 个新软件包，共计 23200 个。在这个发行版中，多数软件包得到了更新：有 13400 个(占 `etch` 软件包总数的 72%)。而且，由于各种原因，有相当数量的软件包(3100 个，占 `etch` 软件包总数的 17%) 从这次发行中被删除了。您将会看不到这些包有任何更新，而且在包管理软件中它们会被标记为“过时的”。

在这次发行中，Debian GNU/Linux 从 X.Org 7.1 更新到 X.Org 7.3。

Debian GNU/Linux 再次与几个桌面程序和环境一起发布。和其它发行版一样，它现在包含 GNOME 2.22 桌面环境。¹，KDE 3.5.10, Xfce 4.4.2, and LXDE 0.3.2.1+svn20080509. Productivity applications have also been upgraded, including the office suites OpenOffice.org 2.4.1 and KOffice 1.6.3 as well as GnuCash 2.2.6, Gnumeric 1.8.3 and Abiword 2.6.4.

¹ 和一些 GNOME 2.20 的模块。

其它的桌面应用程序也被更新了，包含 Evolution 2.22.3 和 Pidgin 2.4.3（即以前的 Gaim）。Mozilla 家族也被更新了：iceweasel(3.0.3 版)是无商标的 Firefox 网页浏览器，icedove(2.0.0.17 版)则为无商标的 Thunderbird 邮件客户端。

和过去一样，这次发行还包含了以下软件的更新：

软件包	在版本 4.0(etch) 中的版本	在版本 5.0(lenny) 中的版本
Apache	2.2.3	2.2.9
BIND DNS 服务器	9.3.4	9.5.0
Cherokee web server	0.5.5	0.7.2
Courier MTA	0.53.3	0.60.0
Dia	0.95.0	0.96.1
Ekiga VoIP 客户端	2.0.3	2.0.12
Exim 默认邮件服务器	4.63	4.69
GNU 编译器套件(默认的编译器)	4.1.1	4.3.2
GIMP	2.2.13	2.4.7
GNU C 函数库	2.3.6	2.7
lighttpd	1.4.13	1.4.19
maradns	1.2.12.04	1.3.07.09
MySQL	5.0.32	5.0.51a
OpenLDAP	2.3.30	2.4.11
OpenSSH	4.3	5.1p1
PHP	5.2.0	5.2.6
Postfix MTA	2.3.8	2.5.5
PostgreSQL	8.1.15	8.3.5
Python	2.4.4	2.5.2
Tomcat	5.5.20	5.5.26

The official Debian GNU/Linux distribution now ships on 4 to 5 binary DVDs or 28 to 32 binary CDs (depending on the architecture) and 4 source DVDs or 28 source CDs. Additionally, there is a *multi-arch* DVD, with a subset of the release for the amd64 and i386 architectures, along with the source code. For the first time, Debian GNU/Linux is also released as Blu-ray images, also for the amd64 and i386 architectures, along with the source code.

Debian now supports Linux Standards Base (LSB) version 3.2. Debian 4.0 did support version 3.1.

2.1.1. 包管理

The preferred program for package management from the command line is **aptitude**, which can perform the same package management functions as **apt-get** and has proven to be better

at dependency resolution. If you are still using **dselect**, you should switch to **aptitude** as the official front-end for package management.

对于 **lenny** 来说, **aptitude** 中已经实现了一个高级的冲突解决机制, 如果它检测到由于软件包依赖关系变化带来的冲突, 它将会尽力找到最好的解决方法。

2.1.2. proposed-updates

All changes to the released stable distribution (and to oldstable) go through an extended testing period before they are accepted into the archives. Each such update of the stable (or oldstable) release is called a point release. Preparation of point releases is done through the proposed-updates mechanism.

软件可以通过两种途径进入 proposed-updates 区。首先, 加入 security.debian.org 的软件包安全更新会被自动加入 proposed-updates 区。其次, %debian; 开发者可以将新的软件包直接上传至 proposed-updates 区。该区最新软件包列表可以在 <http://ftp-master.debian.org/proposed-updates.html> 处看到。

如果您希望在软件包正式加入发行点之前帮忙测试包的更新, 您可以通过将 proposed-updates 加入 `sources.list` 中来进行:

```
deb      http://mirrors.kernel.org/debian/debian lenny-proposed-
updates main contrib
deb-src  http://mirrors.kernel.org/debian/debian lenny-proposed-
updates main contrib
```

下一次您运行 **aptitude update** 命令时, 系统将识别 proposed-updates 区中的软件包, 并在升级包时查找这一区域。

严格意义上来说, 这并不是 Debian 的新特性, 而是一个以前未被太多关注过的特性。

2.2. 系统改进

这次发行版中的大量改变能使全新安装的 **lenny** 受益, 但是可能不会自动启用在从 **etch** 升级上来的系统中。这一节给出大部分相关改变的概述。

SELinux 优先级为标准, 但是默认不启用

The packages needed for SELinux (Security-Enhanced Linux) support have been promoted to priority *standard*. This means that they will be installed by default during new installations. For existing systems you can install SELinux using:

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

注意对 SELinux 支持不是默认启用的。有关设置和启用 SELinux 的信息可以在 [Debian Wiki](http://wiki.debian.org/SELinux) [<http://wiki.debian.org/SELinux>] 找到。

新的默认 syslog 守护程序

The package `rsyslog` takes over as default system and kernel logging daemon for Debian 5.0, replacing **syslogd** and **klogd**. With stock logging rules, it can be used as a drop-in replacement; if you have custom rules, you should migrate them to the new configuration file, `/etc/rsyslog.conf`.

从 `etch` 升级的用户需要手工安装 `rsyslog`, 删除 `sysklogd`。在升级到 `lenny` 的过程中, 并不会自动替换系统日志服务程序。

更好地支持 UTF-8

许多软件将被设置成默认使用 UTF-8 或比以前更好地支持 UTF-8。请参阅 <http://wiki.debian.org/UTF8BrokenApps>, 查看处理 UTF-8 仍然有问题的应用程序信息。

[Debian Wiki](http://wiki.debian.org/Etch2LennyUpgrade) [<http://wiki.debian.org/Etch2LennyUpgrade>] 有些关于 `etch` 和 `lenny` 之间变更的额外信息。

2.3. Debian Live

With `lenny` Debian presents official Live systems for the amd64 and i386 architectures.

A Debian Live system is a Debian system that can be booted directly from removable media (CD-ROMs, DVDs, USB keys) or from another computer over the network without the need of installation. The images are produced by a tool named `live-helper`, which can easily be used to create custom live images. More information about the Debian Live project can be found at <http://debian-live.alioth.debian.org/>.

2.4. 内核相关的主要变化

Debian GNU/Linux 5.0 对所有架构均使用内核版本 2.6.26。

There have been major changes both in the kernel itself and in the packaging of the kernel for Debian. Some of these changes complicate the upgrade procedure and can potentially result in problems while rebooting the system after the upgrade to `lenny`. This section gives an overview of the most important changes; information on how to work around potential issues is included in later chapters.

2.4.1. 内核打包的变化

新的 OpenVZ 内核标识

Debian GNU/Linux 5.0 provides pre-built kernel images for OpenVZ, a second virtualization solution to go alongside the Linux-VServer support included in `etch`. Advantages of OpenVZ include support for live migration, at the expense of a slightly higher overhead.

X86 架构内核软件包名称已统一

In previous releases there was a special `-k7` kernel flavor for 32-bit AMD Athlon/Duron/Sempron processors. This variant has been dropped; the single flavor `-686` now handles all AMD/Intel/VIA 686 class processors.

在可能的情况下，依赖于新软件包的伪过渡包已经被提供，以取代过时的软件包。

2.5. Emdebian 1.0 (based on Debian GNU/Linux lenny 5.0)

Lenny 现在包含 Emdebian 的制作工具。它允许 Debian 源文件包被交叉构建，并且裁剪，以适合嵌入式的 ARM 系统。

Emdebian 1.0 发布版包含了已经构建好的 ARM 包，足够为特殊的机器和机器变体创建定制的 root 文件系统。内核和内核模块需要单独提供。对 armel 和 i386 的支持正在开发中。请参见 [Emdebian 网页](http://www.emdebian.org/) [http://www.emdebian.org/] 获得更详细的信息。

2.6. Netbook support

Netbooks, such as the EEE PC by Asus, are now supported by Debian. For the EEE PC, have a look at the `eeepc-acpi-scripts`. Also, Debian features a new Lightweight X11 Desktop Environment, `lxde`, which is beneficial for netbooks or other computers with relatively low performance.

2.7. Java now in Debian

The OpenJDK Java Runtime Environment `openjdk-6-jre` and Development Kit `openjdk-6-jdk`, needed for executing Java GUI and Webstart programs or building such programs, are now in Debian. The packages are built using the IcedTea build support and patches from the IcedTea project.

第 3 章 安装系统

Debian 安装程序是 Debian 官方的安装系统。它提供了 Debian 操作系统的各种安装方式。您适合用何种安装方式，取决于您的计算机硬件架构。

您可以借助 [Debian 网站](http://www.debian.org/releases/stable/debian-installer/) [http://www.debian.org/releases/stable/debian-installer/] 上的安装指南找到 lenny 的安装程序映像。

安装手册也可以在 Debian 官方 CD/DVD 的第一张盘上找到，路径是：

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

您也可以检查[勘误表](http://www.debian.org/releases/stable/debian-installer/index#errata) [http://www.debian.org/releases/stable/debian-installer/index#errata]，以查看 Debian 安装程序的已知问题列表。

3.1. 安装系统有哪些新特性？

There has been a lot of development on the Debian Installer since its first official release with Debian GNU/Linux 3.1 (sarge) resulting in both improved hardware support and some exciting new features.

在发布说明中，我们只列举安装程序的主要变更。如果您想对从 etch 开始各项变更的详细情况有个大致的了解，请阅读 lenny 之 beta 和 RC 版本的发布声明，它们位于 Debian 安装程序的[新闻](http://www.debian.org/devel/debian-installer/News/) [http://www.debian.org/devel/debian-installer/News/] 网页上。

3.1.1. 主要更新

支持在安装期间加载固件

当固件在 Debian 安装媒体之外提供时，现在可以从可移动媒体加载二进制文件。

支持从 Microsoft Windows 安装

The installation media are now provided with an application that allows preparing the system to install Debian from Microsoft Windows environments.

支持 SATA RAID

尽早升级有安全更新的包

When used with functional network access, the installer will upgrade all packages that have been updated since the initial release of lenny. This upgrade happens during the installation step, before the installed system is booted.

从现已安装的系统更难被在发布 lenny 到安装期间，已知且被修正的安全问题所影响。

支持 *volatile*

The installer can now optionally set up the installed system to use updated packages from `volatile.debian.org`. This archive hosts packages providing data that needs to be regularly updated over time, such as timezones definitions, anti-virus signature files, etc.

为 Intel x86 和 AMD64 新增引导菜单

为 i386/amd64 新增了交互式引导菜单，让用户选择特别的选项或引导方法更直观。

新移植架构

现在支持 armel 架构。还提供了 i386 Xen 客户安装镜像。

支持硬件语音合成设备

现在安装程序支持几种硬件语音合成设备，因此提高了有视力障碍用户的易访问性。

支持挂接选项 `relatime`

The installer can now set up partitions with the `relatime` mount option, so that access time on files and directories is updated only if the previous access time was earlier than the current modify or change time.

在安装期间执行 NTP 时钟同步

在安装期间将计算机时钟与 NTP 同步，所以安装好的系统立即有准确的时钟。

新增语言支持

Thanks to the huge efforts of translators, Debian can now be installed in 63 languages (50 using the text-based installation user interface and 13 supported only with the graphical user interface). This is five languages more than in etch. Languages added in this release include Amharic, Marathi, Irish, Northern Sami, and Serbian. Due to lack of translation updates, one language has been dropped in this release: Estonian. Another language that was disabled in etch has been reactivated: Welsh.

The languages that can only be selected using the graphical installer as their character sets cannot be presented in a non-graphical environment are: Amharic, Bengali, Dzongkha, Gujarati, Hindi, Georgian, Khmer, Malayalam, Marathi, Nepali, Punjabi, Tamil, and Thai.

简化的国家选择

现在国家列表按照大陆分组，当用户不想使用与选择的语言关联的国家时，选择国家更容易。

3.1.2. 自动安装

前一节中所提到的一些变更，也意味着安装程序使用预配置文件进行自动安装的调整。这意味着，如果您有能在 `etch` 安装程序中正常使用的预配置文件，您不能指望它能在新安装程序中正常使用，而无需修改。

[安装指南](http://www.debian.org/releases/stable/installmanual) [<http://www.debian.org/releases/stable/installmanual>]现在有一个单独附录，用来描述使用预配置文件的方法。

3.1.3. Install Debian GNU/Linux with a Braille display

You can install Debian GNU/Linux 5.0 (lenny) with a Braille display. The Braille display must be connected to a USB or a serial port. If your Braille display is connected to a USB port and the American Braille table is supposed to be used, you can press **Enter** in the boot menu. The screen reader BrlTTY is standardized to search for Braille displays at the USB port and uses the American Braille table. If the Braille display is connected to a serial port or if you want to use a different

Braille table, you have to press the **Tab** key in the boot menu first. Then, you can configure the screen reader BrlTTY with the kernel parameter `brltty`.

For the parameter, the following syntax is valid:

```
brltty=driver,device,table
```

All operands are optional.

driver

The driver for the employed Braille display. Here, you must enter either a code consisting of two letters or the word `auto`. If this argument is not entered, the automated recognition is activated by default.

device

The device can be entered as relative to `/dev/` as well as an absolute specification.

table

Defines the Braille table for the desired language. By default, the US table is employed.

3.1.3.1. Examples

```
brltty=ht,ttyS0,de
```

The Handy Tech Driver is used. The Braille display is connected to Com1. The German Braille table is used.

```
brltty=,,de
```

Here, only the German Braille table is specified. Therefore, BrlTTY will try to find a Braille display at a USB port.

3.1.4. Install Debian GNU/Linux with a hardware speech synthesis



注意

Support for hardware speech synthesis is only available in the textual version of the installer. For size reasons, however, it is enabled along with support for the graphical installer, which needs more space anyway. You thus need to select the `Graphical install` entry in the boot menu.

Hardware speech synthesis can not be automatically detected. You thus need to append the **`speakup.synth=driver`** boot parameter to tell Speakup which driver it should use. **`driver`** should be replaced by the driver code for your device, see <http://www.linux-speakup.org/spkguide.txt> for a list. The textual version of the installer will then be automatically selected, and support for the speech synthesis will be automatically installed on the target system.

第 4 章 从以前的发行版升级

4.1. 升级前的准备

我们建议您在升级前阅读[第 5 章 *lenny* 中需要注意的问题](#)中的信息。它描述了与升级过程有间接联系的潜在问题，但在升级前了解这些信息对您来说还是很重要的。

4.1.1. 备份数据和配置文件

在升级系统前，强烈建议您进行一次完整的备份，或者至少备份您不愿意丢失的数据和配置信息。升级工具和流程是非常可靠的，但升级过程中的硬件错误会对您的系统造成严重损害。

您需要备份的内容包括 `/etc`, `/var/lib/dpkg`, `/var/lib/aptitude/pkgstates` 下的文件以及 `dpkg --get-selections "*"` 命令的输出(命令行中的引号必须要有)。

升级过程本身不会修改 `/home` 下的任何内容。但某些程序(比如部分 Mozilla 套件，以及 GNOME 和 KDE 桌面环境)会在新版本的程序第一次启动时用新的默认值来覆盖现有的用户设置。做为一种预防措施，您也许想备份用户目录下的隐藏文件和目录("dotfiles")。该备份有助于您恢复或者重建原有设置。另外，您可能还需要把该事项通知给用户。

Any package installation operation must be run with superuser privileges, so either log in as `root` or use **su** or **sudo** to gain the necessary access rights.

升级需要几个前提条件，您应当在实际升级前检查它们。

4.1.1.1. Make sure you are on a suitable kernel

lenny's version of `glibc` will not work with kernels older than `2.6.8` on any architecture and some architectures have higher requirements. We strongly recommend that you upgrade to and test an `etch 2.6.18` or `2.6.24` kernel or a custom kernel of at least version `2.6.18` before beginning the upgrade process.

4.1.2. 提前告知用户

提前通知所有用户您正在计划的任何升级将是明智的决定，但是通过 **ssh** 连接访问您系统的用户可能在升级过程中不会注意到什么东西，而且应该可以继续正常工作。

If you wish to take extra precautions, back up or unmount the `/home` partition before upgrading.

当升级到 *lenny* 时您可能需要进行内核升级，所以通常需要重启。一般说来，这将在升级完成后进行。

4.1.3. 准备恢复

由于 *etch* 和 *lenny* 的内核在驱动、硬件探测与命名以及设备文件的命名和排序等方面有着诸多变化，在更新后您可能在重启系统时遇到很大的问题。本章和发行注记的下一章记录了很多已知的潜在问题。

由于这个原因，当您的系统重启失败(或对于远程管理系统来说，无法从网络唤醒)时，请确保您能够将其恢复。

如果您通过 **ssh** 远程连接进行升级，则强烈建议您采取必要的预防措施以便通过远程串行终端来访问该服务器。因为在升级内核并重启后，某些设备可能会被改名(如在第 4.6.4 节“设备枚举的顺序”中所描述的情况)，而您将不得不通过本地控制台修复系统配置。还有，如果系统在升级过程中意外重启您可能需要使用本地控制台进行修复。

很明显首先应尝试的就是以您原来的内核重启。然而，由于本文档其它地方所描述的各种各样的问题，并不保证可以成功重启。

如果重启失败了，您需要采取替代方案来启动您的系统以便访问和修复它。其中一种选择是使用特制的急救盘或者 Linux live CD。以这些介质引导后，您应该可以挂载您的根文件系统并 **chroot** 到它里面来检查并修复问题。

我们推荐的另一种选择是使用 **lenny Debian** 安装程序的**拯救模式**。使用安装程序的优点是，您可以从众多安装方式中选择最适合您情况的一种。更多信息请查询**安装手册** [<http://www.debian.org/releases/stable/installmanual>]第 8 章的“修复损坏的系统”一节以及**Debian 安装程序常见问题** [<http://wiki.debian.org/DebianInstaller/FAQ>]。

4.1.3.1. **initrd** 引导时使用调试环境

initramfs-tools 在其生成的 **initrd** 中包含一个调试 shell¹。如果 **initrd** 无法挂载您的根文件系统，您将会进入此调试 shell，里面可以使用基本的命令以帮助跟踪问题甚至有可能解决之。

需要检查的几个基本项目：**/dev** 下是否存在正确的设备文件；加载了什么模块(**cat /proc/modules**)；**dmesg** 的输出中有关驱动加载的错误信息。**dmesg** 的输出还会显示哪个设备文件被指定给哪个磁盘；您还应该检查 **echo \$ROOT** 的输出，以确保根文件系统在预想的设备上。

如果您确实修复了问题，输入 **exit** 会退出调试 shell，并从先前失败的地方继续启动进程。当然您还需要修复更深层次的问题，并重新生成 **initrd**，以使下次引导不再失败。

4.1.4. 为升级准备安全环境

发行版升级应该从本机的文本模式虚拟控制台(或者直连串行终端)，或者通过 **ssh** 远程连接进行。

为了在远程升级时获得额外的可靠性保障，我们建议您在由 **screen** 程序生成的虚拟控制台中执行升级过程。它能进行可靠的重新连接并确保，即使远程连接失败，升级过程也不会被打断。



重要

您不应该在您要升级的机器上使用 **telnet**、**rlogin**、**rsh** 之类的东西，或者从 **xdm**、**gdm**、**kdm** 之类管理的 X 会话中升级。因为那些服务中的任意一个在升级中都有可能被中止，进而导致出现一个不可访问的，只升级了一半的系统。

¹ 此功能可以通过在您的启动参数中添加参数 **panic=0** 来禁用。

4.1.5. 为 LILO 准备 initramfs

使用 LILO 启动管理器的用户应该注意，`initramfs-tools` 目前的默认设置生成的 `initramfs` 对于 LILO 来说太大而无法加载。这些用户要么把启动管理器换成 `grub`，要么就编辑 `/etc/initramfs-tools/initramfs.conf` 文件，更改这行

```
MODULES=most
```

为

```
MODULES=dep
```

Note, however, that doing this will cause `initramfs-tools` to install onto the `initramfs` only those modules that are required for the particular hardware that it is being run on. If you want to generate boot media that will work on more hardware than just the machine you're generating it on, you should leave the line as

```
MODULES=most
```

并确保不使用 LILO。

4.2. 检查系统状态

此章所述的升级过程是为了从“纯粹的” `etch` 升级而安排的，即系统中没有第三方的软件。为使升级过程最可靠，您可能会希望在升级前，从系统中移除第三方的软件包。

此过程也假定您的系统已经更新至 `etch` 的最新发行点。如果您还未做或是不确定，请按照第 A.1 节“升级您的 `etch` 系统”中的说明做。

4.2.1. 复审包管理器中的未决操作

在某些情况下，使用 `apt-get` 代替 `aptitude` 安装软件包时，可能会让 `aptitude` 认为包“未被使用”，并安排它被移除。一般来说，在正式升级前您应该确保系统是最新且“干净的”。

Because of this you should review if there are any pending actions in the package manager `aptitude`. If a package is scheduled for removal or update in the package manager, it might negatively impact the upgrade procedure. Note that correcting this is only possible if your `sources.list` still points to *etch* and not to *stable* or *lenny*; see 第 A.2 节“检查您的 `sources list`”。

To perform this review, launch `aptitude` in “visual mode” and press **g** (“Go”). If it shows any actions, you should review them and either fix them or implement the suggested actions. If no actions are suggested you will be presented with a message saying “No packages are scheduled to be installed, removed, or upgraded”.

4.2.2. 禁用 APT pinning 操作

如果已经设置 APT 从一个非 `stable` (如 `testing`) 版安装特定软件包，您可能必须改变 APT pinning 设置 (保存在 `/etc/apt/preferences` 中) 以允许升级至新的 `stable` 版中的包。更多有关 APT pinning 的信息可以在 `apt_preferences(5)` 中找到。

4.2.3. 检查包状态

不管用什么方法升级，我们都建议您先检查所有软件包的状态，并验证所有包都处于可升级状态。以下命令会显示任何半安装或是配置失败的包，还有那些有任何错误状态的包。

```
# dpkg --audit
```

您也可以使用 **dselect**、**aptitude** 来审查系统中的所有包的状态，也可以用如下的命令来审查

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

或是

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

在升级前移除所有的 `hold` 状态是很有必要的。如果有任何对升级而言有重大影响的包处于 `hold` 状态，升级会失败。

注意 **aptitude** 使用一种与 **apt-get** 及 **dselect** 不同的方法来注册 `hold` 状态的包。您可以用以下命令来确认 **aptitude** 中处于 `hold` 状态的包。

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

如果您想检查 **apt-get** 中哪些包设置了 `hold` 状态，可以用

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

如果您在本地更改并重新编译了一个包，且并未重命名，或是在版本号中放入新的纪元号，您就必须将它设为 `hold` 状态，以防止它被升级。

aptitude 中的 “`hold`” 状态的包能用以下命令更改：

```
# aptitude hold package_name
```

用 `unhold` 代替 `hold` 即可清除 “`hold`” 状态。

如果有任何需要修复的东西，您最好确保 `sources.list` 仍然指向 `etch`，就像第 A.2 节 “[检查您的 sources list](#)” 中所说的那样。

4.2.4. proposed-updates 区

如果您已经把 `proposed-updates` 区加入您的 `/etc/apt/sources.list` 文件，那么您应当在升级前移除它，这样可以降低冲突发生的概率。

4.2.5. 非官方源和 backports

如果有任何非 Debian 的软件包安装在您的系统中，您应该意识到它们可能会在升级期间因为依赖性冲突而被移除。如果这些包是通过在 `/etc/apt/source.list` 中添加一个额外的软件源而安装的，您应该检查那个软件源是否也提供 `lenny` 下编译好的包，并同时像您的 Debian 软件源那样更改相应的源。

某些用户可能有非官方的 `backported` “较新的” 包版本存在于他们安装的 Debian `etch` 系统中。升级期间那样的包是最可能引起问题的，因为它们可能会导致文件冲突²。第 4.5.8 节 “升级期间可能遇到的问题” 已经有一些关于如何处理将会出现的文件冲突的信息了。

4.2.5.1. 使用 backports.org 包

`backports.org` 是一个由 Debian GNU/Linux 开发者提供的半官方的仓库，它基于“testing”仓库重新编译，来为 `stable` 版提供新的软件包。

The `backports.org` repository mainly contains packages from “testing”, with reduced version numbers so that the upgrade path from `etch` backports to `lenny` still works. However, there are a few backports which are made from unstable: security updates, plus the following exceptions: Firefox, the Linux kernel, OpenOffice.org, and X.Org.

如果您不使用这些特例，您可以安全地升级到 `lenny`。如果您使用了这些特例，临时对 `lenny` 的所有包设置 `Pin-Priority` (参看 `apt_preferences(5)`) 到 1001，这样您就可以放心地使用 `dist-upgrade` 了。参考 [backports FAQ](http://backports.org/dokuwiki/doku.php?id=faq) [http://backports.org/dokuwiki/doku.php?id=faq]。

4.3. 手工删除软件包的标记

要防止 **aptitude** 移除某些因依赖关系而安装的包，您需要手工删除这些包上的 `auto` 标记。对于桌面安装来说，包括 OpenOffice 和 Vim:

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

如果您已经用内核元数据包安装了 2.6 内核镜像，那还要包括它：

```
# aptitude unmarkauto $(dpkg-query --W -f='${Package} ${Version} ${Architecture}\n' -f='${Package} ${Version} ${Architecture}\n' | grep -i kernel-image-2.6.* | cut --f1)
```

² Debian 的包管理系统正常情况下不允许一个软件包移除或是替换另一个软件包所拥有的文件，除非已经被设定为替换那个包。



注意

您可以复查那些在 `aptitude` 中被标记为 *auto* 的包，执行：

```
# aptitude search -'i~M'
```

4.4. 为 APT 准备源

在开始升级前，您必须在软件包源列表 `/etc/apt/sources.list` 中设定 `apt` 的配置文件。

`apt` will consider all packages that can be found via any “deb” line, and install the package with the highest version number, giving priority to the first line in the file (thus where you have multiple mirror locations, you'd typically first name a local hard disk, then CD-ROMs, and then HTTP/FTP mirrors).



提示

You might need to add an GPG checking exception for DVDs and CD-ROMs. Add the following line to `/etc/apt/apt.conf`, if it's not already in `/etc/apt/apt.conf.d/00trustedcdrom`:

```
APT::Authentication::TrustCDROM -"true";
```

This does not work with DVD/CD-ROM image files, however.

一个发行版通常既能通过它的代号(如: `etch`, `lenny`)引用，也可以用它的状态名引用(如: `oldstable`, `stable`, `testing`, `unstable`)。引用发行版的代号的好处在于，您绝对不会被新的发行版问题困扰且被带至此处。当然，这也意味着您不得不自己关注新版的发行。如果转而使用状态名，一旦有新版发行，您将只会看到一堆可用的软件包的更新。

4.4.1. 添加互联网 APT 源

默认配置用于从主 Debian 网络服务器上安装，但您可能希望修改 `/etc/apt/sources.list` 以使用其它的镜像，离您最近的网络镜像是一种更好的选择。

Debian HTTP 或 FTP 镜像地址能在 <http://www.debian.org/distrib/ftplist> 找到(参看“Debian 镜像列表”一节)。HTTP 镜像通常比 FTP 镜像快。

例如，假设离您最近的 Debian 镜像是 `http://mirrors.kernel.org/debian`。当用网络浏览器或是 FTP 程序检查那个镜像时，您会注意到主目录被组织成了如下情况：

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

要让 `apt` 使用这个镜像，您将此行添加至 `sources.list` 文件：

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

注意明显加上去的 ‘dists’，这个发行版名字后的参数用于将路径扩展成多个目录。

添加新源后，在 `sources.list` 中通过在以前存在的 “deb” 行前加上一个井号(#)来禁用它们。

4.4.2. 添加本地镜像 APT 源

您可能希望通过修改 `/etc/apt/sources.list` 来使用一个本地磁盘上的镜像(也许是通过 NFS 挂载的)，来代替 HTTP 或 FTP 软件源镜像。

例如，假设您的镜像位于 `/var/ftp/devian/` 下，并且有如下的主目录：

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

要让 apt 使用它，需要把这一行添加至 `sources.list` 文件：

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

注意明显加上去的 ‘dists’，这个发行版名字后的参数用于将路径扩展成多个目录。

添加新源后，在 `sources.list` 中通过在以前存在的 “deb” 行前加上一个井号(#)来禁用它们。

4.4.3. 从 CD-ROM 或 DVD 添加 APT 源

如果您只想用 CD 安装，在 `/etc/apt/sources.list` 中的 “deb” 行前放上井号(#)注释掉它们。

确保在 `/etc/fstab` 中有一行允许您挂载 CD-ROM 于 `/cdrom` 挂载点(**apt-cdrom** 必须指定挂载点为 `/cdrom`)。例如，假设 `/dev/hdc` 就是您的 CD-ROM，`/etc/fstab` 中应该带有一行：

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

注意在第四段 `defaults,noauto,ro` 之间必须无空格。

要验证设置是否有效，插入一片 CD，尝试运行

```
# mount -/cdrom      # 挂载 CD 至挂载点
# ls --alF -/cdrom   # 显示 CD 的根目录
# umount -/cdrom     # 卸载 CD
```

下一步，运行：

```
# apt-cdrom add
```

每片您所拥有的 Debian 二进制 CD-ROM 都要这么做，以便将每片 CD 的数据添加至 APT 的数据库。

4.5. 升级软件包

推荐的从以前 Debian GNU/Linux 发行版升级的方法是使用包管理器 **aptitude**。此程序会对安装包采取比直接运行 **apt-get** 更保守的选择。

不要忘记挂载所有必需的分区（尤其是根分区和 `/usr` 分区）为可读写状态，用以下命令：

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

下一步您应该详细检查 APT 源记录 (`/etc/apt/source.list`) 要么指向 “lenny” 又或指向 “stable”。不应该有任何指向 **etch** 的源记录。



注意

CD-ROM 的源通常会指向 “unstable”。尽管这令人困惑，但您不应该改变它。

4.5.1. 记录会话

强烈推荐您使用 `/usr/bin/script` 程序来记录升级会话中的交互信息。这样如果有问题了，您就有一份问题报告。而且需要的话，您也可以在错误报告中提供额外信息。要开始记录，输入：

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

或是类似命令。不要将输出文件放在临时目录下，如 `/tmp` 或 `/var/tmp`（这些目录下的文件可能会在升级或重启时被删除）

输出文件也可让您复查屏幕上滚动过去的信息。只要切换至虚拟终端 2（使用 **Alt+F2**），在登入后，用 `less -R ~/root/upgrade-lenny.script` 查看文件。

在完成升级后，您可以在提示符下输入 `exit` 停止 **script**。

如果已经对 **script** 使用了 `-t` 选项，您就可以用 **scriptplay** 程序来回放整个过程：

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

4.5.2. 更新包列表

首先，需要获取新发行版的可用包列表。执行：

```
# aptitude update
```

首次执行此命令，会更新新源并打印出一些有关源可用性的警告信息。这些警告没关系，而且在您下一次执行时就没有了。

4.5.3. 确保您有足够的空间升级

在升级系统前必须确保在进行如第 4.5.7 节“升级系统的其它部分”中所述的完全系统升级时，您有足够的磁盘空间。首先，安装过程所需的任何从网上下载的包都被保存在 `/var/cache/apt/archives`（下载期间还会放在 `partial/` 子目录下），因此您必须确保在文件系统分区 `/var/` 下有足够的空间存放临时下载包，这些包将会安装在系统中。下载之后，您可能在其它文件系统分区中需要更多的空间来同时安装升级包（可能会包含更大的二进制程序或更多数据）和升级中带入的新包。如果系统中没有足够的空间，您可能以一个未完成的升级结束，而这可能难以恢复。

aptitude 和 **apt** 都会显示给您安装所需磁盘空间的详细信息。在执行升级操作前，您可以运行以下命令来看到这个估计值：

```
# aptitude --y --s --f ---with-recommends dist-upgrade
[ -... -]
XXX 个已升级，XXX 个刚装上，XXX 个待删除以及 XXX 个未升级。
需要获取 xx.xMB/yyyMB。解开后，将占用 AAAMB。
会下载/安装/删除包。
```



注意

Running this command at the beginning of the upgrade process may give an error, for the reasons described in the next sections. In that case you will need to wait until you've done the minimal system upgrade as in 第 4.5.6 节“最小系统升级” and upgraded your kernel before running this command to estimate the disk space.

如果没有足够空间来升级，确保您预先释放了空间。可以这样：

- 删除以前安装时下载的包（位于 `/var/cache/apt/archives`）。用 **apt-get clean** 或 **aptitude clean** 来清理包缓存，这会删除所有以前下载过的包文件。
- Remove forgotten packages. If you have `popularity-contest` installed, you can use **popcon-largest-unused** to list the packages you do not use that occupy the most space. You can also use **deborphan** or **debfoaster** to find obsolete packages (see 第 4.11 节“过时的包”). Alternatively you can start **aptitude** in “visual mode” and find obsolete packages under “Obsolete and Locally Created Packages”.
- Remove packages that take up too much space and are not currently needed (you can always reinstall them after the upgrade). You can list the packages that take up the most disk space with **dpigs** (available in the `debian-goodies` package) or with **wajig** (running `wajig size`).

You can list packages that take up most of the disk space with **aptitude**. Start **aptitude** into *visual mode*, select Views → New Flat Package List (this menu entry is available only after etch

version), press **I** and enter `~i`, press **S** and enter `~installsize`, then it will give you nice list to work with. Doing this after upgrading `aptitude` should give you access to this new feature.

- Remove translations and localization files from the system if they are not needed. You can install the `localepurge` package and configure it so that only a few selected locales are kept in the system. This will reduce the disk space consumed at `/usr/share/locale`.
- 将 `/var/log/` 下的系统日志临时放到其它系统，或是永久删除。
- 使用临时目录 `/var/cache/apt/archives`：您可以使用来自另一个系统的临时缓存目录(USB 存储设备，临时硬盘，使用中的文件系统……)



注意

不要使用 NFS，这是因为网络连接可能在升级期间断开。

例如，假设您将 USB 盘挂在 `/media/usbkey` 下：

1. 删除以前安装时下载的包：

```
# apt-get clean
```

2. 将 `/var/cache/apt/archives` 目录复制到 USB 磁盘：

```
# cp --ax -/var/cache/apt/archives -/media/usbkey/
```

3. 将临时缓存目录挂载至当前目录：

```
# mount ---bind -/media/usbkey/archives -/var/cache/apt/archives
```

4. 升级后，还原 `/var/cache/apt/archives` 目录：

```
# umount -/media/usbkey/archives
```

5. 删除留下的 `/media/usbkey/archives`。

您可以在挂载于系统中的任何文件系统上创建临时缓存目录。

注意为了安全地移除软件包，建议您换回 `etch` 的 `sources.list`，就像第 A.2 节“检查您的 `sources list`”中描述的那样。

4.5.4. 先升级 `apt` 和(或) `aptitude`

Several bug reports have shown that the versions of the `aptitude` and `apt` packages in `etch` are often unable to handle the upgrade to `lenny`. In `lenny`, `apt` is better at dealing with complex chains of packages requiring immediate configuration and `aptitude` is smarter at searching for solutions to satisfy the dependencies. These two features are heavily involved during the dist-upgrade to `lenny`, so it is necessary to upgrade these two packages before upgrading anything else. For `apt`, run:

```
# apt-get install apt
```

对于 `aptitude` (如果您安装过的话) 要执行:

```
# aptitude install aptitude
```

This step will automatically upgrade `libc6` and `locales` and will pull in SELinux support libraries (`libselinux1`). At this point, some running services will be restarted, including **x**dm, **g**dm and **k**dm. As a consequence, local X11 sessions might be disconnected.

4.5.5. 使用 `aptitudes` 记录的 `apt` 自动安装的软件包列表

`aptitude` 维护了一个自动安装的包列表(就像另一个包的依赖一样)。在 `lenny` 中, `apt` 现在也有这个功能了。

首次运行 `lenny` 版本的 `aptitude` 时, 它会先自动读入安装包的列表, 然后再转换成 `lenny apt` 可用的版本。如果您安装了 `aptitude`, 您至少应该先发出一条 **aptitude** 指令以便转换。一种方法是搜索一个不存在的包:

```
# aptitude search -"?false"
```

4.5.6. 最小系统升级

由于在 `etch` 和 `lenny` 间有某些必需包存在冲突, 因此直接运行 `aptitude dist-upgrade` 通常会删除大量您想要保留的包。因此我们推荐一种两步升级过程, 首先做一次最小升级来解决这些冲突, 然后做一次完整的 `dist-upgrade` 升级。

首先, 运行:

```
# aptitude upgrade
```

这会升级那些不需要删除或安装其它任何包的软件。

下一步会相当依赖您安装过的包。发行注记会给出所用方法的通用建议, 但如有疑问, 推荐您在继续操作前检查每种方法建议的需要删除的包。

某些通用包要被删除, 包括 `base-config`, `hotplug`, `xlibs`, `netkit-inetd`, `python2.3`, `xfree86-common` 和 `xserver-common`。要了解更多在 `lenny` 中废弃包的信息, 参看第 4.11 节 “过时的包”。

4.5.6.1. 升级桌面系统

This step concerns systems with the `desktop` task installed, or with the GNOME or KDE packages installed.

Check whether you have the `libfam0c102` package installed:

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

If you do have it installed, run:

```
# aptitude install libfam0
```

Note that installing `libfam0` will also install the File Alteration Monitor (`fam`) as well as the RPC portmapper (`portmap`) if not already available in your system. Both packages will enable a new network service in the system although they can both be configured to be bound to the (internal) loopback network device.

4.5.6.2. 升级没有 X 支持的系统

在不带 X 的系统上，不需要额外的 `aptitude install` 命令，您可以转入下一步。

4.5.7. 升级系统的其它部分

现在，您要准备继续升级重要部分。执行：

```
# aptitude dist-upgrade
```

这将对系统进行一次全面的升级，比如：安装所有包的最新可用版本，解决不同发行版的包之间所有潜在的依赖性变动。如有必要，它会安装一些新包（通常是新库，或是重命名的包），并删除任何有冲突的过时包。

When upgrading from a set of CD-ROMs (or DVDs), you will be asked to insert specific CDs at several points during the upgrade. You might have to insert the same CD multiple times; this is due to inter-related packages that have been spread out over the CDs.

如果不改变被指定在当前状态（显示为 “held back”）的另一个包，当前安装包的新版本就无法升级。这个问题可以通过使用 **aptitude** 选择这些包来安装或是用 `aptitude -f install package` 来解决。

4.5.8. 升级期间可能遇到的问题

如果使用 **aptitude**, **apt-get** 或 **dpkg** 操作时出现错误

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

the default cache space is insufficient. You can solve this by either removing or commenting lines you don't need in `/etc/apt/sources.list` or increasing the cache size. The cache size can be increased by setting `APT::Cache-Limit` in `/etc/apt/apt.conf`. The following command will set it to a value that should be sufficient for the upgrade:

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

这里假设您还未在那个文件中对该变量进行过设置。

有时有必要在 APT 中开启 `APT::Force-LoopBreak` 选项使之能够临时移除一个重要的包，这是为了满足冲突或是预依赖循环。**aptitude** 会警告您这一点并退出升级。您可以通过在 **aptitude** 命令行上指定 `-o APT::Force-LoopBreak=1` 选项来解决这个问题。

有可能系统的依赖关系太乱了以至于需要手工干预。通常这意味着使用 **aptitude** 或是

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

来消除某些引起问题的包，或是

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

在极端情况下，您可能不得不像以下命令一样强制重新安装某个包

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

如果您从“纯粹的” **etch** 系统升级就不会出现文件冲突，但如果您装有非官方的 **backport** 包就可能出现冲突。文件冲突会导致类似以下这样的错误：

```
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>
```

您可以尝试用错误信息中的最后一行提示，强制删除包来解决文件冲突：

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

在修复这些问题后，您应该可以通过重复前述的 **aptitude** 命令来继续升级。

升级期间，您会被问到有关配置或是重新配置几个包的问题。如果您被问到是否任何存在于 `/etc/init.d` 或 `/etc/terminfo` 目录，或是 `/etc/manpath.config` 目录下的文件应该被包维护者的版本所替换时，通常有必要回答‘yes’来确保系统一致性。由于它们会被以 `.dpkg-old` 为后缀的文件名保存，所以您总是可以恢复成老版本的配置文件。

如果您不确定该做什么，那就记下包或文件的名称，并稍后排出它们的顺序。您可以通过在 **script** 命令记录的输出文件中搜索来查看升级期间显示在屏幕上的信息。

4.6. 升级内核与相关包

这一节解释了如何升级您的内核，以及明确有关此次升级的潜在问题。您可以安装由 Debian 提供的 `linux-image-*` 包，或者从源码编译一个自定义的内核。

注意本节的很多信息基于假设，即您会使用一个模块化的 **Debian** 内核，以及 `initramfs-tools` 和 `udev`。如果您选择使用一个不需要 `initrd` 的自定义内核，或是您使用了一种不同的 `initrd` 生成器，对您来说某些信息可能会不一样。

4.6.1. 安装内核元数据包

当您从 `etch` 全面升级至 `lenny` 时，强烈推荐您安装新的 `linux-image-2.6-*` 元数据包。此包可能由 `dist-upgrade` 自动安装了。您可以验证此步，执行：

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

如果您没有看到任何输出，那么您需要手工安装一个新的 `linux-image` 包。要查看可用的 `linux-image-2.6` 的元数据包列表，执行：

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

If you are unsure about which package to select, run `uname -r` and look for a package with a similar name. For example, if you see `'2.6.18-6-686'`, it is recommended that you install `linux-image-2.6-686`. (Note that the `k7` flavor no longer exists; if you are currently using the `k7` kernel flavor, you should install the `686` flavor instead.) You may also use **apt-cache** to see a long description of each package in order to help choose the best one available. For example:

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

然后您应该使用 `aptitude install` 来安装。一旦安装了这个新内核，您应该在下次可行时重启机器以获取新内核所提供的特性。

对于爱冒险的人，这里有个简单方法来编译您的自定义 `Debian GNU/Linux` 内核。安装 `kernel-package` 工具，然后阅读 `/usr/share/doc/kernel-package` 下的文档。

4.6.2. 从 2.6 内核升级

如果您目前运行来自 `etch` 的 2.6 系列的内核，在您做了一次完整的系统包升级后，会自动发生这个升级(就像[第 4.5 节 “升级软件包”](#)所描述的那样)。

If possible, it is to your advantage to upgrade the kernel package separately from the main `dist-upgrade` to reduce the chances of a temporarily non-bootable system. Note that this should only be done after the minimal upgrade process described in [第 4.5.6 节 “最小系统升级”](#).

如果您正使用您自定义的内核并想使用 `lenny` 中可用的内核，您也可以进行此步操作。如果 `udev` 不支持您的内核，建议您在最小升级后再升级内核。如果 `udev` 支持您的内核，您就可以安稳地等着，直到整个系统升级完成。

4.6.3. 从 2.4 内核升级

如果您装有 2.4 内核，且您的系统依赖于 `hotplug` 探测硬件，那在您尝试升级前应该先从 `etch` 升级至 2.6 系列的内核。确保 2.6 系列内核可引导您的系统，且您所有的硬件都能在进行升级前被正常检测到。

如果您的系统不依赖于 `hotplug`³，按第 4.5.7 节“升级系统的其它部分”中所说，您可以在系统全面升级后再升级内核。一旦系统已经升级，您就可以做以下事情(通过替换 `flavor` 将内核包的名字改成最适合您系统的那个)：

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

4.6.4. 设备枚举的顺序

`lenny` 带来一个比上一发行版更健全的硬件探测机制。但是，这会引起系统中所探测到的设备顺序变化，即影响设备名的分配顺序。例如，如果您有两张驱动不同的网卡，`eth0` 和 `eth1` 设备名可能被互换了。请注意，新机制意味着假设您在运行着的 `lenny` 系统中互换以太网适配器，新的适配器也会获得一个新的接口名。

对于网络设备，您可以通过使用 `udev` 机制避免此重排，更明确点，通过在 `/etc/udev/rules.d/70-persistent-net.rules` 中指定⁴。另一种方法，您可以在引导时用 **ifrename** 工具将物理设备绑定至指定的名字。参看 `ifrename(8)` 和 `iftab(5)` 获取更多信息。两个可选方法 (`udev` 和 **ifrename**) 不应该同时使用。

对于存储设备，通过使用 `initramfs-tools` 并配置它按当前的顺序载入存储设备，这样您就可以避免这个顺序重排。要这么做的话，通过查看 **lsmod** 命令的输出来明确您系统中载入的存储模块的顺序。**lsmod** 以它们载入的反序显示，即列表中的第一个模块是最后一个被载入的。注意这只对内核以固定顺序枚举出的设备有效(像 PCI 设备)。

但是，在首次引导后删除并重新载入模块会影响这个顺序。还有，您的内核可能有些静态链接的驱动，而且这些模块名称不会出现在 **lsmod** 的输出中。您可能会解读这些驱动名并从 `/var/log/kern.log` 或是 **dmesg** 的输出中载入引导顺序。

Add these module names to `/etc/initramfs-tools/modules` in the order they should be loaded at boot time. Some module names may have changed between `etch` and `lenny`. For example, `sym53c8xx_2` has become `sym53c8xx`.

然后您需要运行 `update-initramfs -u -k all` 重新生成 `initramfs` 镜像。

一旦您正运行一个 `lenny` 内核和 `udev`，您可以重新配置系统以通过一个不依赖于驱动载入顺序的别名来访问磁盘。这些别名存在于 `/dev/disk/` 目录结构下。

4.6.5. 引导期间的问题

如果用 `initramfs-tools` 创建的 `initrd` 引导系统，在某些情况下由 `udev` 创建的设备文件可能对于启动脚本的执行来讲太慢了点。

这通常表现为引导失败，由于根文件系统无法挂载，您会进入调试环境。但当您回头检查时，所需的所有设备都存在于 `/dev` 下。只要根文件系统存在于 USB 磁盘或是 RAID 上，尤其是使用了 `LILO` 时，就会出现这种问题。

³ 您可以通过合理地配置 `/etc/modules` 来静态载入系统所需的内核模块。

⁴ 脚本 `/etc/udev/rules.d/75-persistent-net-generator.rules` 自动生成的规则让网卡拥有固定的名字。删除此符号链接可以关闭 `udev` 对 NIC 设备的固定命名。

解决此问题的方法就是用引导参数 `rootdelay=9`。这个超时值(秒)可能需要调整一下。

4.7. 在重启之前要做的事情

当 `aptitude dist-upgrade` 完成时,“正式的”升级就算结束了。但在下一次重启之前,还有一些其它的事情需要关注一下。

4.7.1. 重新运行 lilo

如果您正在用 `lilo` 作为您的启动管理器(这是某些 `etch` 安装的默认引导器),强烈建议您在升级后重新运行 **lilo**:

```
# -/sbin/lilo
```

注意,即使您不升级系统内核也需要这一步,这是由于包的升级 **lilo** 的 `second stage` 会改变。

还有,检查 `/etc/kernel-img.conf` 中的内容并确保其中有 `do_bootloader = Yes` 这一行。这样,在内核升级后,启动管理器总会重新运行一下。

如果您在运行 **lilo** 时遇到任何问题,请检查 / 下至 `vmlinuz` 和 `initrd` 的符号链接,以及 `/etc/lilo.conf` 中内容不一致的地方。

如果您在重启前或是在您未手工重启系统就偶然重启前,忘记重新运行 **lilo** 的话,您的系统可能会无法引导。当系统引导时您将只会看到 `LI` 而不是 `lilo` 提示符⁵。有关如何从此处恢复的信息请参看第 4.1.3 节“准备恢复”。

4.8. 系统引导时在显示 **Waiting for root file system** 后挂起

从 `/dev/hda` 变成 `/dev/sda` 的补救方法

Some users have reported that an upgrade could cause the kernel not to find the system root partition after a system reboot.

在那种情况下,系统引导会挂起并出现以下信息:

```
Waiting for root file system -...
```

and after a few seconds a bare busybox prompt will appear.

当内核升级使用新一代的 IDE 驱动程序时,就会出现这个问题。老的 IDE 磁盘的命名规则是 `hda`, `hdb`, `hdc`, `hdd`。新的规则将分别命名为 `sda`, `sdb`, `sdc`, `sdd`。当升级时未生成

⁵ 想知道更多关于 **lilo** 启动错误代码的信息,请参看 [The Linux Bootdisk HOWTO](http://tldp.org/HOWTO/Bootdisk-HOWTO/a1483.html) [<http://tldp.org/HOWTO/Bootdisk-HOWTO/a1483.html>]。

一个新的 `/boot/grub/menu.lst` 文件来产生新的命名规则时，就会出现这个问题。引导时，Grub 会传递给内核一个无法找到的系统根分区。

如果您在升级后已经遇到此问题，转至[第 4.8.2 节“如何在升级后解决此问题”](#)。要在升级前避免此问题，就提前看一下。

4.8.1. 如何在升级前避免此问题

One can avoid this problem entirely by using an identifier for the root filesystem that does not change from one boot to the next. There are two possible methods for doing this - labeling the filesystem, or using the filesystem's universally unique identifier (UUID). These methods are supported in Debian since the 'etch' release.

The two approaches have advantages and disadvantages. The labeling approach is more readable, but there may be problems if another filesystem on your machine has the same label. The UUID approach is uglier, but having two clashing UUIDs is highly unlikely.

以下范例中，我们假设根文件系统位于 `/dev/hda6`，同时再假设您的系统装有运行正常的 `udev` 和 `ext2` 或是 `ext3` 文件系统。

To implement the labeling approach:

1. 通过执行命令 **e2label /dev/hda6 rootfilesys** 来标记文件系统（名称必须小于 16 个字符）。
2. 编辑 `/boot/grub/menu.lst` 并更改以下这行：

```
# kopt=root=/dev/hda6 ro
```

为

```
# kopt=root=LABEL=rootfilesys ro
```



注意

不要删除行首的 #，那个要的。

3. 通过运行 **update-grub** 更新 `menu.lst` 中的 `kernel` 行。
4. 编辑 `/etc/fstab` 并改变挂载 / 分区的那行，如：

```
/dev/hda6      -/      ext3  defaults,errors=remount-ro 0 1
```

为

```
LABEL=rootfilesys      -/      ext3  defaults,errors=remount-ro 0 1
```

在这里改变的是第一栏，您不必修改此行的其它栏目。

To implement the UUID approach:

1. 通过 `ls -l /dev/disk/by-uuid | grep hda6` 找出您文件系统的通用唯一标识符。

您应该得到类似于以下内容的一行：

```
lrwxrwxrwx 1 root root 24 2008-09-25 08:16 d0dfcc8a-417a-41e3-
ad2e-9736317f2d8a --> ../../hda6
```

UUID 是指向 `/dev/hda6` 的符号链接的名字，如 `d0dfcc8a-417a-41e3-ad2e-9736317f2d8a`。



注意

您文件系统的 **UUID** 可能是不同的字符串。

2. 编辑 `/boot/grub/menu.lst` 并更改以下这行：

```
# kopt=root=/dev/hda6 ro
```

为

```
# kopt=root=UUID=d0dfcc8a-417a-41e3-ad2e-9736317f2d8 ro
```



注意

不要删除行首的 `#`，那个要的。

3. 通过运行 **update-grub** 更新 `menu.lst` 中的 `kernel` 行。

4. 编辑 `/etc/fstab` 并改变挂载 / 分区的那行，如：

```
/dev/hda6 -/ ext3 defaults,errors=remount-ro 0 1
```

为

```
UUID=d0dfcc8a-417a-41e3-ad2e-9736317f2d8 -/ ext3
defaults,errors=remount-ro 0 1
```

在这里改变的是第一栏，您不必修改此行的其它栏目。

4.8.2. 如何在升级后解决此问题

4.8.2.1. 方案一

当 **Grub** 显示菜单界面让您选择您想启动的菜单项时，可以做这步操作。如果没出现那样的菜单，尝试在内核引导显示它之前，按下 **Esc** 键。如果您还是无法进入此菜单，尝试[第 4.8.2.2 节“方案二”](#)或是[第 4.8.2.3 节“方案三”](#)。

1. 在 **Grub** 菜单中，高亮您想启动的菜单项。按下 **e** 键来编辑有关此条目的选项。您将会看到一些东西，类似于：

```
root (hd0,0)
```

```
kernel -/vmlinuz-2.6.26-1-686 root=/dev/hda6 ro  
initrd -/initrd.img-2.6.26-1-686
```

2. 高亮此行

```
kernel -/vmlinuz-2.6.26-1-686 root=/dev/hda6 ro
```

按下 **e** 键并用 `sdX` 代替 `hdX` (`X` 可以是字符 `a`, `b`, `c` 或 `d`, 这依赖于您的系统)。在我的例子中这一行变为:

```
kernel -/vmlinuz-2.6.26-1-686 root=/dev/sda6 ro
```

然后按下 **Enter** 来保存修改。如果还有其它带 `hdX` 字样的行, 则一并修改。不要改变类似于 `root (hd0,0)` 这样的条目。一旦做好所有的修改, 按下 **b** 键。然后您的系统现在应该能正常启动。

3. 现在您的系统已经启动了, 您需要永久修复这个问题。转至第 4.8.1 节 “如何在升级前避免此问题” 并使用两个方法中的一个。

4.8.2.2. 方案二

Boot from Debian GNU/Linux installation media (CD/DVD) and when prompted, pick `rescue` to launch rescue mode. Select your language, location, and keyboard mapping; then let it configure the network (no matter whether it succeeds or not). After a while, you should be asked to select the partition you want to use as root file system. The proposed choices will look something like:

```
/dev/ide/host0/bus0/target0/lun0/part1  
/dev/ide/host0/bus0/target0/lun0/part2  
/dev/ide/host0/bus0/target0/lun0/part5  
/dev/ide/host0/bus0/target0/lun0/part6
```

如果您知道哪个分区是您的根文件系统, 就选择正确的那个。如果您不知道, 只好试一下第一个。如果它报告说是无效的根文件系统分区, 就试下一个, 依次类推。一个接一个地尝试不会损伤您的分区, 而且如果您仅在磁盘上装有一个系统, 您应该很容易就找出正确的根文件系统分区。如果您在磁盘上装有多系统, 最好能准确地知道哪个是正确的分区。

一旦您已选中一个分区, 有几个选项可以选择。选择在所选分区中启动 `shell` 环境的选项。如果启动失败, 那就试另一个分区。

Now you should have shell access as user `root` on your root file system mounted on `/target`. You need access to the contents of the `/boot`, `/sbin` and `/usr` directories on your hard disk, which should now be available under `/target/boot`, `/target/sbin` and `/target/usr`. If these directories need to be mounted from other partitions, do so (see `/etc/fstab` if you have no idea of which partition to mount).

转至第 4.8.1 节 “如何在升级前避免此问题” 并使用两个方法中的一个来永久修复这个问题。然后输入 `exit` 退出急救 `shell` 环境, 并选择 `reboot` 正常重启系统(别忘了拿走可引导介质)。

4.8.2.3. 方案三

1. 从您最喜欢的 LiveCD 发行版启动，比如 Debian Live, Knoppix, Ubuntu Live。
2. 挂载 /boot 目录所在的分区。如果您不知道是哪个，利用 **dmesg** 的输出来找出是否您的磁盘是 hda, hdb, hdc, hdd 或是 sda, sdb, sdc, sdd。一旦您知道是哪个磁盘，例如是 sdb，执行命令 **fdisk -l /dev/sdb** 来查看分区表并找出正确的分区。
3. 假设，您已经在 /mnt 下挂载了正确的分区，而且此分区含有 /boot 目录及其内容，那就编辑 /mnt/boot/grub/menu.lst 文件。

找到类似于以下的这一段：

```
## ## End Default Options ##

title          Debian GNU/Linux, kernel 2.6.26-1-686
root           (hd0,0)
kernel         -/vmlinuz-2.6.26-1-686 root=/dev/hda6 ro
initrd         -/initrd.img-2.6.26-1-686

title          Debian GNU/Linux, kernel 2.6.26-1-686 (single-
user mode)
root           (hd0,0)
kernel         -/vmlinuz-2.6.26-1-686 root=/dev/hda6 ro single
initrd         -/initrd.img-2.6.26-1-686

### END DEBIAN AUTOMAGIC KERNELS LIST

and replace every hda, hdb, hdc, hdd with sda, sdb, sdc, sdd, as appropriate. Don't modify
the line similar to:

root           (hd0,0)
```

4. 重启系统，弹出 LiveCD，您的系统应该能正确的启动了。
5. 当系统引导时，使用第 4.8.1 节 “如何在升级前避免此问题” 中所建议的两种处理方法中的一个来永久修复这个问题。

4.9. 为下个发行版做准备

升级完成后，您可以为下个发行版做点事。

- 如果新的内核镜像元数据包被当作旧镜像的依赖包而安装，那新镜像将被标记为自动安装，这应该被更正一下：

```
# aptitude unmarkauto $(dpkg-query --W -f='${Package} ${Version} ${Architecture}\n' -|
cut --f1)
```

- 像第 4.11 节 “过时的包” 中描述的那样移除过时的和未使用的包。您应该核查这些包所用的配置文件，并考虑完全删除这些软件包以移除它们的配置文件。

4.10. 不赞成地包

随着下个版本 `&debian; 6.0` (开发代号 `squeeze`) 的发行，大量的服务包都会被标记为过时，因此当升级至 6.0 时，更新至这些软件包的新版本可以解决您的困扰。

包含以下这些包：

- `apache (1.x)`, successor is `apache2`
- `bind8`, successor is `bind9`
- `exim3`, successor is `exim4`
- `php4`, successor is `php5`
- `postgresql-7.4`, successor is `postgresql-8.1`
- 对于 `uw-imapd`，尚没有定义继承者。

4.11. 过时的包

在带来几千个新软件包的同时，`lenny` 也清除了曾经存在于 `etch` 中超过两千个的旧软件包。`lenny` 不提供这些过时包的升级。当然，在您需要的地方没什么可以阻止您继续使用这些过时的包，Debian 计划通常会在 `lenny` 发行版后的一年间不断的提供安全更新⁶，且通常不会同时提供其它支持。推荐用可用的替代品代替它们。

为什么包会从发行版中移除？这有很多原因，如：它们不再被上游作者维护了；没有 Debian 开发者对维护这个包感兴趣；这些包提供的功能被不同的软件（或新版本）替代了；或者由于它们自身的问题，因而被认为不适用于 `lenny`。最后一种情况下，这些包有可能仍然位于 “unstable” 版之下。

Detecting which packages in an updated system are “obsolete” is easy since the package management front-ends will mark them as such. If you are using **aptitude**, you will see a listing of these packages in the “Obsolete and Locally Created Packages” entry. **dselect** provides a similar section but the listing it presents might differ.

Also, if you have used **aptitude** to manually install packages in `etch` it will have kept track of those packages you manually installed and will be able to mark as obsolete those packages pulled in by dependencies alone which are no longer needed if a package has been removed. Also, **aptitude**, unlike **deborphan** will not mark as obsolete packages that you manually installed, as opposed to those that were automatically installed through dependencies.

您可以用其它的工具来找出过时的包，例如 **deborphan**，**debfoister** 或 **cruft**。推荐使用 **deborphan**，尽管它（默认情况下）只报告过时的库，即“`libs`”或是“`oldlibs`”部分中的未

⁶ 只要在那段时间内没有另外的发行版释出。在任意给定时间内，一般仅提供两个 `stable` 发行版的安全更新。

被其它包依赖的软件包。不要盲目的移除那些工具要用到的包，尤其是当您正在使用激进的非默认的选项，它们容易引起严重错误。极度推荐在您移除它们之前，手工核查那些建议移除的包(例如：它们的内容、大小以及描述信息)。

[Debian 错误跟踪系统](http://bugs.debian.org/) [http://bugs.debian.org/]通常会提供额外信息，这些都是有关这个包为什么要被移除的信息。您应该既查看此包自身的归档错误报告，同时也要看一下 [ftp.debian.org pseudo-package](http://ftp.debian.org/pseudo-package) [http://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=ftp.debian.org&archive=yes] 中的归档错误报告。

4.11.1. 哑包

来自 `etch` 的某些包在 `lenny` 中已经被分为好几个包，通常这是为了改善系统的可维护性。在此情况下，为了能平滑升级，`lenny` 通常提供“哑”包(即空包)，它与 `etch` 中的旧包名称相同，但有依赖性，促使它们安装新包。这些“哑”包在升级成功后就会被认为是过时的包，且能够被安全地移除。

绝大多数(并非所有)哑包的描述信息会显示它们的用途。但哑包的描述信息并不统一，因此您可能会发现带上 `--guess` 选项的 **deborphan** 命令对于找出您系统中的哑包是很有用的。注意，某些哑包在升级后是不打算被系统移除的，相反它们被用于长期跟踪某程序的当前可用版本这样的目的。

第 5 章 lenny 中需要注意的问题

5.1. 潜在的问题

有时候，我们无法避免某些变化带来的副作用，或者我们在别的地方暴露了 Bug。我们在这里记录我们已知的问题。请您同时阅读勘误，相关软件包的文档，Bug 报告以及在第 6.1 节“更多读物”中提到的其它信息。

5.1.1. 与 udev 相关的设备的问题

虽然 udev 经过了大量的测试，您仍可能在使用某些设备时遇到问题，这些需要修复。最常见的情况是需要修改设备的访问许可和所有权。有些情况下设备可能不会被默认创建(比如 `/dev/video` and `/dev/radio`)。

udev 提供配置机制以处理这些问题。请见 `udev(8)` 和 `/etc/udev` 以获得更多信息。

5.1.2. 某些应用程序可能无法在 2.4 内核下工作

某些 lenny 中的程序可能无法在 2.4 内核下工作，例如有些需要 `epoll()` 支持，而 2.4 内核中没有。这些程序可能根本无法工作或者工作不正常，除非系统重启并使用 2.6 内核。

一个例子是 HTTP 代理 `squid`。

5.1.3. 某些网站无法通过 TCP 进行连接

从 2.6.17 开始，Linux 大量使用 RFC 1323 中描述的 TCP 窗口缩放。某些服务器会有不正常的行为，并向他们自己宣布错误的窗口大小。更多细节，请参考 bug 报告 [#381262](http://bugs.debian.org/381262) [<http://bugs.debian.org/381262>], [#395066](http://bugs.debian.org/395066) [<http://bugs.debian.org/395066>], [#401435](http://bugs.debian.org/401435) [<http://bugs.debian.org/401435>]。

通常有两个替代方法：或者将最大允许 TCP 窗口大小改回到一个较小的值(推荐)或者彻底关闭 TCP 窗口缩放(不推荐)。请参见 [debian-installer errata page](http://www.debian.org/devel/debian-installer/errata) [<http://www.debian.org/devel/debian-installer/errata>] 中的示例命令。

5.1.4. 无法自动关机

在一些旧系统中，`shutdown -h` 可能无法关闭机器(只是将其停止)。这是因为这里需要使用 APM。将 `acpi=off` `apm=power_off` 添加到内核命令行，比如添加在 `grub` 或者 `lilo` 的配置文件中，应该可以修正这个问题。请参见 bug [#390547](http://bugs.debian.org/390547) [<http://bugs.debian.org/390547>] 以获取更多信息。

5.1.5. 异步网络初始化可能引起无法预测的情况

在使用 udev 为网络接口加载驱动的系统上，因为 udev 的异步天性，使得系统启动时网络驱动可能无法在 `/etc/init.d/networking` 执行前加载完成。虽然在 `/etc/network/`

interfaces 中加入 allow-hotplug (与 auto 一起使用) 能保证在网络接口可用才将其启用, 这仍然无法保证在启动进程开始启动网络服务前完成, 而这些网络服务可能因没有网络接口而无法正常运行。

5.1.6. 使用 WPA 保护无线网络的问题

在 etch 中, wpasupplicant 软件包设置为系统服务, 通过 /etc/default/wpa_supplicant 和用户提供的 /etc/wpa_supplicant.conf 进行配置。

在 lenny 中, /etc/init.d/wpa_supplicant 被抛弃, 现在此 Debian 软件包已经和 /etc/network/interfaces 集成, 类似其它软件包如 wireless-tools。这意味着 wpa_supplicant 不再直接提供系统服务。

更多有关配置 wpa_supplicant 的信息请参考 /usr/share/doc/wpa_supplicant/README.modes.gz, 里面有 /etc/network/interfaces 文件的例子。Debian 中有关使用 wpa_supplicant 软件包的更新信息可以在 [Debian Wiki](http://wiki.debian.org/WPA) [http://wiki.debian.org/WPA] 中找到。

5.1.7. 文件名中使用非 ASCII 字符可能引起问题

挂载 vfat, ntfs 或者 iso9660 文件系统时, 如果其中有文件的名称包含非 ASCII 字符时, 会在尝试使用这些文件名时失败, 除非使用 utf8 选项来挂载。迹象是可能会出现这样的失败: “无效或不完整的多字节字符或宽字符”。可行的解决方法是使用 defaults,utf8 作为 vfat, ntfs 和 iso9660 文件系统的挂载参数, 如果它们包含有非 ASCII 字符名称的文件。

请注意当使用 utf8 选项时 Linux 内核不支持处理 vfat 文件系统时忽略文件名的大小写。

5.1.8. 声音停止工作

在极端情况下, 升级后可能无声。如果发生了这种情况, 请执行 ALSA 检查列表:

- 以 root 身份执行 **alsacnf**,
- 将你的用户增加到组 audio,
- 确认各个选项已开启并没有静音(使用命令 **alsamixer**),
- 确认 **arts** 和 **esound** 没有启动,
- 确认没有 OSS 模块被加载,
- 确认扬声器已打开, 以及
- 检查如下的命令

```
cat -/dev/urandom > -/dev/dsp
```

能否以root用户运行。

5.2. NFS 挂载现在由 nfs-common 来处理

Since `util-linux` 2.13 NFS mounts are no longer handled by `util-linux` itself, but by `nfs-common`. Since not all systems mount NFS shares and to avoid a standard portmapper installation `util-linux` only suggests `nfs-common`. If you need to mount NFS shares, make sure `nfs-common` is installed on your system. The preinstallation script of the `mount` package checks whether NFS mounts exist and aborts if `/usr/sbin/mount.nfs` from `nfs-common` is not present or if `nfs-common` is out-of-date. Either upgrade `nfs-common` or unmount any NFS mounts prior to upgrading `mount`.

5.3. 罗马尼亚(ro)键盘布局的改变

由于在 `lenny` 中升级到 `xkb-data` 版本 1.3 默认的罗马尼亚(ro)布局变体现在可以产生正确的 `șț` 字符(注意下面的逗号)而不是 `ŝţ` (注意下面的下加符)。还有一些变体已经改名。旧的变体名字仍然可以工作, 不过建议用户更新他们的 `/etc/X11/xorg.conf`。因这个改变更多信息以及可能导致的副作用在 [wiki](http://wiki.debian.org/L10N/Romanian/Lenny/Notes) (仅罗马尼亚语) [<http://wiki.debian.org/L10N/Romanian/Lenny/Notes>]

5.4. 升级 apache2

Apache2 默认配置在一些方面有所改变, 这需要手工改变您的配置。最重要的改变有:

`NameVirtualHost *` 被改为 `NameVirtualHost *:80`。如果您加入了更多基于名称的虚拟主机, 您需要依次将 `<VirtualHost *>` 修改为 `<VirtualHost *:80>`。

`Apache` `User` 和 `Group` 以及 `PidFile` 路径现在在 `/etc/apache2/envvars` 中进行配置。如果您从这些设置的默认值进行了修改, 您需要修改那个文件。这也意味着您无法通过 `apache2 -k start` 启动 `apache2`, 您需要使用 `/etc/init.d/apache2` 或者 `apache2ctl`。

`Mod_suexec` 所需要的 `suexec` 帮助程序现在通过单独的软件包 `apache2-suexec` 来发行, 此软件包不会默认安装。

更多模块相关的配置从 `/etc/apache2/apache2.conf` 移到了 `/etc/apache2/mods-available/*.conf`。

要获取更多详细信息, 请参见 `/usr/share/doc/apache2.2-common/NEWS.Debian.gz` 和 `/usr/share/doc/apache2.2-common/README.Debian.gz`。

5.5. NIS 和网络管理器

在 `lenny` 中, `nis` 内的 `ypbind` 支持网络管理器。当网络管理器报告电脑从网络断开时, 它会导致 `ypbind` 禁用 NIS 客户端功能。因为网络管理器通常会在不使用网络时发出此报告, NIS 客户系统的 NIS 用户应确认禁用这些系统上的网络管理器支持。

要做到这一点可以通过卸载 `network-manager` 软件包, 或者编辑 `/etc/default/nis` 并添加 `-no-dbus` 到 `YPBINDARGS`。

新安装的 Debian 会默认使用 `-no-dbus`，而之前的发行版本不是默认的。

5.6. Mozilla 产品的安全状态

The Mozilla programs `firefox`, `thunderbird`, and `sunbird` (rebranded in Debian to `iceweasel`, `icedove`, and `iceowl`, respectively), are important tools for many users. Unfortunately the upstream security policy is to urge users to update to new upstream versions, which conflicts with Debian's policy of not shipping large functional changes in security updates. We cannot predict it today, but during the lifetime of lenny the Debian Security Team may come to a point where supporting Mozilla products is no longer feasible and announce the end of security support for Mozilla products. You should take this into account when deploying Mozilla and consider alternatives available in Debian if the absence of security support would pose a problem for you.

`iceape`, the unbranded version of the `seamonkey` internet suite has been removed from lenny (with the exception of a few internal library packages).

5.7. GNOME 桌面改变与支持

lenny 版本中的GNOME 桌面环境相比 `etch` 中发行的版本有大量的变化，您可以在 [GNOME 2.22 发行说明](http://library.gnome.org/misc/release-notes/2.22/) [http://library.gnome.org/misc/release-notes/2.22/] 找到更多信息。

5.8. Emacs21* 默认不支持 Unicode

Emacs21 和 `emacs21-nox` 默认不配置为支持 Unicode。更多信息和替代方案请参见 [Bug #419490](http://bugs.debian.org/419490) [http://bugs.debian.org/419490]。可以考虑切换到 `emacs22`, `emacs22-gtk` 或 `emacs22-nox`。

5.9. slurpd/replica 已经被删除

OpenLDAP 从版本 2.4.7 开始，不再支持通过 `slurpd` 服务执行 LDAP 复制。原先的配置需要修改为使用 LDAP 同步复制引擎(`syncrepl`)。更详细的文档位于 <http://www.openldap.org/doc/admin24/replication.html>。

5.10. 桌面没有使用全部屏幕

Intel Mobile GM965 的驱动会错误地把输出类型检测为 VGA，并且设置一个较低的屏幕分辨率以适应输出。这个 bug 的典型特征为桌面管理器仅使用屏幕的一部分。添加如下的代码到配置文件 `/etc/X11/xorg.conf` 能纠正驱动的行为。

```
Section "Monitor"
    Identifier "VGA"
    Option "Ignore" "true"
EndSection
```

更多信息请参考 bug [#496169](http://bugs.debian.org/496169) [http://bugs.debian.org/496169]。

5.11. DHCP failover issue

When running a failover pair of DHCP servers, the peer names need to be consistent, otherwise DHCP will crash.

Please see <http://bugs.debian.org/513506> and <https://lists.isc.org/pipermail/dhcp-users/2007-September/004538.html> for more information.

第 6 章 关于 Debian GNU/Linux 的更多信息

6.1. 更多读物

除了本发行说明和安装指南外，Debian 文档项目 (DDP) 有更多关于 Debian GNU/Linux 的文档，该项目的目标是为 Debian 用户和开发者创建高质量的文档。它包括 Debian 参考手册，Debian 新维护人员指南和 Debian 常见问题等更多的文档。请查阅 [DDP 网站](http://www.debian.org/doc/ddp/) [http://www.debian.org/doc/ddp/]，以获得现有资源的全部详细资料。

各个软件包的文档被安装到 `/usr/share/doc/package`。它包括版权信息，特定于 Debian 的信息，以及上游的文档。

6.2. 获得帮助

Debian 的用户可以从很多渠道获得帮助、建议和支持，但您只应该在查阅所有的文档后，却依然找不到问题的答案时，才考虑寻求帮助。本节简单介绍了对 Debian 新用户可能会有帮助的渠道。

6.2.1. 邮件列表

邮件列表 `debian-user` (英语) 和其它 `debian-user-language` (其它语言) 中有 Debian 用户最感兴趣的内容。想了解这些邮件列表的详细信息，以及如何订阅，请参阅 <http://lists.debian.org/>。发问前请先搜索答案，并注意遵守列表有关的礼仪和准则。

6.2.2. 互联网在线聊天系统

Debian 在 OFTC IRC 网络上有专门对 Debian 用户提供帮助的 IRC 频道。使用您喜欢的 IRC 客户端连接 `irc.debian.org`，加入 `#debian` 频道即可。

请遵守频道的准则，尊敬其他用户。可以在 [Debian 维基](http://wiki.debian.org/DebianIRC) [http://wiki.debian.org/DebianIRC] 中找到该准则。

请访问 [网站](http://www.oftc.net/) [http://www.oftc.net/]，以获得更多关于 OFTC 的信息。

6.3. 报告 Bug

我们努力让 Debian GNU/Linux 成为一款高质量的操作系统，但这并不意味着我们提供的软件包完全没有任何 Bug。为了和 Debian 一贯的“开放的开发”原则，以及为我们的用户提供更好的服务相协调，我们在 Bug 追踪系统(BTS)提供全部已经报告的 Bug 的所有信息。您可以通过 <http://bugs.debian.org/> 来访问 BTS。

如果您在本发行版或者软件包中发现了 Bug，请报告它，从而可以在将来的发行中被修复。您需要一个有效的电子邮件地址才能报告 Bug。有这个限制是因为这样我们才能够追踪 Bug，以及在开发人员需要更多信息时，能够与提交人联系。

您可以使用程序 **reportbug** 来提交一个 Bug 报告，或者亲自动手发送邮件。可以通过阅读参考文档(如果您安装了 doc-debian 包的话，在 /usr/share/doc/debian 可以找到)或者在线的 **Bug 追踪系统** [<http://bugs.debian.org/>]，以获得更多关于 Bug 追踪系统的信息和用法。

6.4. 为 Debian 做贡献

并不是只有高手才能对 Debian 做出贡献。当您在各种支持用户的**邮件列表** [<http://lists.debian.org/>]中帮助其他用户解决问题时，您就是在对整个社区做贡献。参与开发的**邮件列表** [<http://lists.debian.org/>]以帮助开发者发现(以及解决)和系统发行相关的问题，对我们也是极大的帮助。为了维持 Debian 系统一贯的高品质，请 **提交 Bug** [<http://bugs.debian.org/>]，并协助开发人员跟踪和修复它们。如果您有文字方面的天赋，您也可以通过创作**文档** [<http://www.debian.org/doc/ddp>]，或者**翻译** [<http://www.debian.org/international/>]现有文档到您自己的语言，从而做出更直接的贡献。

如果您能投入更多的时间的话，您可以负责维护 Debian 发行版中的部分自由软件。如果您能够新增或维护其他用户希望 Debian 所能包含的软件包的话就更好了，可以在**急需人手和被期待的软件包库 (WNPP)** [<http://www.debian.org/devel/wnpp/>]中获得相关的信息。如果您在某些特别的领域有兴趣，那么您可能会很乐意参加向某个特定硬件平台移植的子项目，以及 **Debian Jr.** [<http://www.debian.org/devel/debian-jr/>] 和 **Debian Med** [<http://www.debian.org/devel/debian-med/>] 等项目。

无论如何，只要您以任何方式参加了自由软件社区的活动，不管您是用户，程序员，作者，还是译者，您就已经为自由软件社区做出了贡献。贡献是非常有益和有趣的，不仅让您能够结交更多的新伙伴，还能让您内心充满温暖的感觉。

附录 A. 管理您的 etch 系统

本附录包含在升级到 lenny 之前，弄清楚您怎样能够从 etch 安装或升级软件包的相关信息。这应该只在特殊的情况下需要。

A.1. 升级您的 etch 系统

基本上这和您已经做过的从 etch 升级没有什么区别。唯一的区别就是，您首先需要确定您的软件包列表仍旧引用像在 第 A.2 节 “检查您的 sources list” 说明的 etch 中的软件包。

如果您从 Debian 镜像升级，将自动升级到最新的 etch。

A.2. 检查您的 sources list

如果在 `/etc/apt/sources.list` 的任何行都引用 `'stable'`，您实际上已经在“使用”lenny。如果您已经运行了 `apt-get update`，您仍然可以按照下面的过程恢复而不会出现问题。

如果您已经从 lenny 安装了软件包，就没有太大的必要从 etch 安装软件包了。在这种情况下，您必须决定是否继续或者终止。可以降级软件包，但是这里不会提及。

用您喜欢的编辑器(以 root 权限)打开文件 `/etc/apt/sources.list`，检查所有以 `deb http:` 或 `deb ftp:` 开头，引用“stable”的行。如果找到了，将 `stable` 改为 `etch`。

如果有些行以 `deb file:` 开始，您应该自己检查该位置是否包含有 etch 或者 lenny 的档案。



重要

不要修改任何以 `deb cdrom:` 开始的行。如果这样做了的话，这些行就会失效，您将需要重新运行 **apt-cdrom**。即使在 `'cdrom'` 的行发现“unstable”，也不要感到奇怪。虽然令人困惑，但这是正常的。

如果做了任何修改，保存文件后执行

```
# apt-get update
```

刷新软件包列表。

附录 B. Contributors to the Release Notes

Many people helped with the release notes, including, but not limited to

Adam Di Carlo, Andreas Barth, Andrei Popescu, Anne Bezemer, Bob Hilliard, Charles Plessy, Christian Perrier, Daniel Baumann, Eddy Petrișor, Emmanuel Kasper, Esko Arajärvi Frans Pop, Giovanni Rapagnani, Gordon Farquharson, Javier Fernández-Sanguino Peña, Jens Seidel, Jonas Meurer, Josip Rodin, Justin B Rye, LaMont Jones, Luk Claes, Martin Michlmayr, Michael Biebl, Moritz Mühlenhoff, Noah Meyerhans, Noritada Kobayashi, Osamu Aoki, Peter Green, Rob Bradford, Samuel Thibault, Simon Bienlein, Simon Paillard, Stefan Fritsch, Steve Langasek, Tobias Scherer, Vincent McIntyre, and W. Martin Borgert.

This document has been translated into many languages. Many thanks to the translators!

附录 C. Lenny dedicated to Thiemo Seufer

The Debian Project has lost an active member of its community. Thiemo Seufer died on December 26th, 2008 in a tragic car accident.

Thiemo was involved in Debian in many ways. He maintained several packages and was the main supporter of the Debian ports to the MIPS architecture. He was also a member of our kernel team, as well as a member of the Debian Installer team. His contributions reached far beyond the Debian project: he also worked on the MIPS port of the Linux kernel, the MIPS emulation of qemu, and far too many smaller projects to be named here.

Thiemo's work, dedication, broad technical knowledge and ability to share this with others will be missed. His contributions will not be forgotten. The high standards of Thiemo's work make it hard to pick up.

To honour his contributions to Debian, the project dedicates the release of Debian GNU/Linux 5.0 "Lenny" to Thiemo.

索引

符号

视力障碍用户, 9

A

Abiword, 3

Apache, 4

B

BIND, 4

Blu-ray, 4

C

CD, 4

Cherokee, 4

Courier, 4

D

Debian Live, 6

Dia, 4

DocBook XML, 2

DVD, 4

E

Ekiga, 4

Emdebian, 7

Evolution, 4

Exim, 4

F

Firefox, 4

G

Gaim, 4

GCC, 4

GIMP, 4

GNOME, 3

GNUCash, 3

GNUmeric, 3

I

IcedTea, 7

J

Java, 7

K

KDE, 3

KOffice, 3

L

LILO, 13, 25

Linux Standards Base, 4

Live system, 6

LXDE, 3

M

Microsoft Windows, 8

Mozilla, 4, 36

MySQL, 4

N

Network Manager, 35

NIS, 35

O

OpenJDK, 7

OpenOffice.org, 3

OpenSSH, 4, 6

P

packages

 apache, 31

 apache2, 31

 apache2-suexec, 35

 apt, 1, 16, 16, 16, 17, 19, 20, 20, 20, 21, 21

 aptitude, 5, 19, 20, 20, 20, 21, 21, 21, 21

 base-config, 21

 bind8, 31

 bind9, 31

 dblatex, 2

 debian-goodies, 19

 doc-debian, 39

 docbook-xsl, 2

 eeepc-acpi-scripts, 7

 emacs22, 36

 emacs22-gtk, 36

 emacs22-nox, 36

 exim3, 31

 exim4, 31

 fam, 22

 firefox, 36

 glibc, 11

 grub, 13, 33

hotplug, 21, 24, 25
iceape, 36
icedove, 4, 36
iceowl, 36
iceweasel, 4, 36
initramfs-tools, 12, 13, 13, 23, 25, 25
kernel-package, 24
libc6, 21
libfam0, 22
libfam0c102, 21
libselinux1, 21
lilo, 26, 33
linux-image-*, 23
linux-image-2.6-686, 24
live-helper, 6
localepurge, 20
locales, 21
lxde, 7
mount, 35, 35
netkit-inetd, 21
network-manager, 35
nfs-common, 35, 35, 35, 35, 35, 35
nis, 35
php4, 31
php5, 31
popularity-contest, 19
portmap, 22
postgresql-7.4, 31
postgresql-8.1, 31
python2.3, 21
release-notes, 1, 1
rsyslog, 6, 6
seamoney, 36
squid, 33
sunbird, 36
sysklogd, 6
thunderbird, 36
udev, 23, 24, 24, 25, 25, 25, 25, 33, 33, 33, 33
util-linux, 35, 35, 35
uw-imapd, 31
wireless-tools, 34
wpasupplicant, 34, 34, 34
xfree86-common, 21
xkb-data, 35
xlibs, 21
xmlroff, 2
xserver-common, 21
xsltproc, 2

PHP, 4
Pidgin, 4
Postfix, 4
PostgreSQL, 4

S

SELinux, 5, 21

T

Thunderbird, 4
Tomcat, 4

U

Unicode, 36

V

virtualization, 6
VServer, 6

W

WPA, 34

X

Xfce, 3

术语表

ACPI	高级配置和电源接口
ALSA	高级 Linux 声音架构
APM	高级电源管理
CD	光盘
CD-ROM	Compact Disc Read Only Memory
DHCP	Dynamic Host Configuration Protocol
DNS	域名系统
DVD	数字通用光盘
GIMP	GNU 图像处理程序
GNU	GNU's Not Unix 的递归缩写
GPG	GNU Privacy Guard
IDE	电子集成驱动器
LDAP	轻量级目录访问协议
LILO	Linux 加载器
LSB	Linux 标准基础
LVM	逻辑卷管理器
MTA	邮件传输代理
NFS	网络文件系统
NIC	网卡
NIS	网络信息服务
OSS	开放声音系统
RAID	独立冗余磁盘阵列
RPC	Remote Procedure Call
SATA	串行高级技术附件
USB	通用串行总线

UUID	通用惟一标识符
VGA	视频图像阵列
WPA	Wi-Fi 安全网络访问