
Release-Notes for Debian 13 (trixie)

发行版本 13.0

Debian Documentation Team

2023-09-03

1 简介	3
1.1 报告文档错误	3
1.2 贡献升级报告	4
1.3 本文档的源文件	4
2 Debian 13 的新变化	5
2.1 支持的架构	5
2.2 软件仓库分区	6
2.3 这次发布中有什么新变化?	6
2.3.1 桌面和知名软件包	6
2.3.2 手册页翻译更新	7
2.3.3 来自 Debian Med Blend 团队的新闻	8
2.3.4 来自 Debian Astro Blend 团队的新闻	8
2.3.5 ARM64 的安全启动	8
3 安装系统	9
3.1 安装系统有哪些新特性?	9
3.2 云服务安装	9
3.3 容器和虚拟机映像	10
4 从 Debian 12 (bookworm) 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	12
4.1.3 准备服务停机	12
4.1.4 准备故障恢复	12
4.1.5 为升级准备安全环境	13
4.2 Start from "pure" Debian	14
4.2.1 Upgrade to Debian 12 (bookworm)	14
4.2.2 升级至最新的小版本更新	14
4.2.3 Debian Backports	14
4.2.4 准备软件包数据库	15
4.2.5 移除过时的软件包	15
4.2.6 移除非 Debian 软件包	15
4.2.7 清理配置文件残留	15
4.2.8 non-free 和 non-free-firmware 区	15
4.2.9 proposed-updates 区	15

4.2.10	非官方源	16
4.2.11	禁用 APT pinning	16
4.2.12	检查 gpgv 是否已安装	16
4.2.13	检查包状态	16
4.3	准备 APT source-list 文件	17
4.3.1	添加互联网 APT 源	17
4.3.2	添加本地镜像 APT 源	18
4.3.3	从光学介质中添加 APT 源	18
4.4	升级软件包	18
4.4.1	记录会话	19
4.4.2	更新软件包列表	19
4.4.3	确保您有足够的空间升级	19
4.4.4	停止监控系统	21
4.4.5	最小系统升级	21
4.4.6	升级系统	21
4.5	升级期间可能遇到的问题	22
4.5.1	Full-upgrade 失败, 显示”无法立即配置”	22
4.5.2	预期的删除	22
4.5.3	冲突或预依赖循环	22
4.5.4	文件冲突	23
4.5.5	配置文件变化	23
4.5.6	将会话切换到控制台	23
4.6	升级内核与相关包	23
4.6.1	安装内核元软件包	24
4.7	为下个发布版本做准备	24
4.7.1	清理已删除的软件包	24
4.8	过时的软件包	25
4.8.1	过渡哑包	25
5	trixie 中需要注意的问题	27
5.1	升级到 trixie 时可能出现的问题	27
5.1.1	非自由固件在仓库中被划分到了单独的分区	27
5.1.2	设置系统时钟的软件包的变更	28
5.1.3	Puppet 配置管理系统升级到了版本 7	28
5.1.4	youtube-dl 被 yt-dlp 取代	28
5.1.5	不同版本的 Fcix 将不能共存	28
5.1.6	MariaDB 软件包名不再包含版本号	29
5.1.7	系统日志的变更	29
5.1.8	影响日志分析工具 (如 logcheck) 的 rsyslog 变更	30
5.1.9	rsyslog 创建的日志文件更少	30
5.1.10	slapd 的升级可能需要人工干预	30
5.1.11	GRUB 不再默认运行 os-prober	31
5.1.12	GNOME 对屏幕阅读器的支持有所下降	31
5.1.13	Baseline for 32-bit PC is now i686	31
5.1.14	polkit 配置的变更	31
5.1.15	”合并的 /usr” 现在是必需的	32
5.1.16	从 buster 升级 (不受支持) 会因为 libcrypt1 而失败	32
5.1.17	openssh-server no longer reads ~/.pam_environment	33
5.1.18	OpenSSH no longer supports DSA keys	33
5.2	升级后在重启前需要做的事	33
5.2.1	升级过程之外的注意事项	33
5.2.2	安全支持上的局限性	33
5.2.3	Python 解释器标记为由外部管理	34
5.2.4	VLC 中视频编/解码的硬件加速支持受限	34

5.2.5	systemd-resolved 被分割成单独的软件包	35
5.2.6	systemd-boot 被分割成单独的软件包	35
5.2.7	systemd-journal-remote 不再使用 GnuTLS	35
5.2.8	bookworm 中的 adduser 的重大变更	35
5.2.9	Xen 网络接口的可预测命名	35
5.2.10	dash 对 “^” 的处理方式的变更	35
5.2.11	netcat-openbsd 支持抽象套接字	35
5.3	过时与废弃内容	36
5.3.1	值得注意的过时软件包	36
5.3.2	trixie 的废弃组件	36
5.4	已知的严重缺陷	36
6	关于 Debian 的更多信息	39
6.1	扩展阅读	39
6.2	获得帮助	39
6.2.1	邮件列表	39
6.2.2	IRC	40
6.3	报告 Bug	40
6.4	为 Debian 做贡献	40
7	在升级前管理您的 bookworm 系统	41
7.1	升级您的 bookworm 系统	41
7.2	检查您的 APT source-list 文件	41
7.3	Performing the upgrade to latest bookworm release	42
7.4	删除过时的配置文件	42
8	发行说明的贡献者	43

The Debian Documentation Project <<https://www.debian.org/doc>>.

本文档是自由软件；您可以在自由软件基金会发布的 GNU 通用公共许可证的条款下重新发布或修改它；您应当使用该许可证的第二版本。

本程序发布的目的是希望它对您有用，但没有任何担保，甚至不保证它有经济价值和适合特定用途。请查阅 GNU 通用公共许可证以获得更多细节。

您应当在收到本程序的同时也收到了一份 GNU 通用公共许可证的副本；如果没有收到，请给自由软件基金会写信。地址是：51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

协议文本可以在 <https://www.gnu.org/licenses/gpl-2.0.html> 和 Debian 系统中的 `/usr/share/common-licenses/GPL-2` 找到。

This document informs users of the Debian distribution about major changes in version 13 (codenamed trixie).

The release notes provide information on how to upgrade safely from release 12 (codenamed bookworm) 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 <https://www.debian.org/releases/trixie/releasenotes>.

小心： 注意列出所有已知的问题是不可能的，因此我们根据影响范围和严重程度选择了一部分问题加以说明。

请注意，我们仅支持从 Debian 的前一版本升级（对于本次发行，即为从 bookworm 升级）并对其提供说明文档。如果您需要从更早的版本升级，我们建议您先查看前一版本的发行说明，并先升级到 bookworm。

1.1 报告文档错误

我们已尽可能地测试了本文档中描述的所有不同的升级步骤，并预测了用户可能遇到的所有问题。

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](#) against the **release-notes** package. You might first want to review the [existing bug reports](#) in case the issue you've found has already been reported. Feel free to add additional information to existing bug reports if you can contribute content for this document.

We appreciate, and encourage, reports providing patches to the document's sources. You will find more information describing how to obtain the sources of this document in *Sources for this document*.

1.2 贡献升级报告

We welcome any information from users related to upgrades from bookworm to trixie. If you are willing to share information please file a bug in the [bug tracking system](#) against the **upgrade-reports** package with your results. We request that you compress any attachments that are included (using `gzip`).

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** 的状态数据库在 `/var/lib/dpkg/status`，**apt** 的软件包状态信息在 `/var/lib/apt/extended_states`。您应当在升级之前先根据[备份数据](#)和[配置文件](#)里的说明进行备份，但您也可以在 `/var/backups` 中找到 `/var/lib/dpkg/status` 文件的备份。
- Session logs created using `script`, as described in [记录会话](#).
- Your apt logs, available at `/var/log/apt/term.log`, or your aptitude logs, available at `/var/log/aptitude`.

备注：在提交错误报告之前，您应该花点时间检查和删除日志中的任何敏感和/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 reStructuredText format, using the sphinx converter. The HTML version is generated using `sphinx-build -b html`. The PDF version is generated using `sphinx-build -b latex`. Sources for the Release Notes are available in the Git repository of the *Debian Documentation Project*. You can use the [web interface](#) to access its files individually through the web and see their changes. For more information on how to access Git please consult the [Debian Documentation Project VCS information pages](#).

[维基页面](#) 提供了更多关于这个主题的信息。

2.1 支持的架构

下面是 Debian 13 官方支持的架构：

- 32-bit PC (`i386`) and 64-bit PC (`amd64`)
- 64-bit ARM (`arm64`)
- ARM EABI (`armel`)
- ARMv7 (EABI hard-float ABI, `armhf`)
- little-endian MIPS (`mipsel`)
- 64-bit little-endian MIPS (`mips64el`)
- 64-bit little-endian PowerPC (`ppc64el`)
- IBM System z (`s390x`)

Baseline bump for 32-bit PC to i686

The 32-bit PC support (known as the Debian architecture `i386`) now requires the "long NOP" instruction. Please refer to [Baseline for 32-bit PC is now i686](#) for more information.

您可以在 [Debian 移植页面](#) 阅读更多的关于您的架构的移植状态和移植细节的信息。

2.2 软件仓库分区

正如社群契约以及 Debian 政策中所提到的，长期以来，软件仓库中包含以下分区：

- `main`：Debian 发行版；
- `contrib`：可以和 Debian 发行版协同工作的补充软件包，但需要发行版之外的软件才能编译或者运行；
- `non-free`：可以和 Debian 发行版协同工作的补充软件包，但与 DFSG 不兼容，或者存在其他影响分发的的问题。

Following the [2022 General Resolution about non-free firmware](#), the 5th point of the Social Contract was extended with the following sentence:

Debian 官方介质可以包含在其他情况下不属于 Debian 系统的一部分的固件，以使得 Debian 可以在需要此类固件的硬件上工作。

虽然尚未在社群契约和 Debian 政策中明确提及，软件仓库新增了一个分区，使得非自由固件能和其他非自由软件包分开处理：

- `non-free-firmware`

作为 Debian 13 发布准备工作的一部分，大多数非自由固件软件包已从 `non-free` 分区移动到了 `non-free-firmware` 分区。这一清晰的划分使得使用 `main` 和 `non-free-firmware` 的软件包构建官方安装映像成为可能，而不需要 `contrib` 和 `non-free` 的软件包。正因为此，构建的安装映像可以安装仅包含 `main` 和 `non-free-firmware` 的系统，而不包含 `contrib` 和 `non-free`。

关于从 `bookworm` 升级的注意事项，请阅读[non-free](#) 和 [non-free-firmware](#) 区。

2.3 这次发布中有什么新变化？

Debian 的这次发行再次带来了比上一版本 `bookworm` 更多的软件；本次发行包括 11294 个新软件包，软件包的总数达到了 59551 个。这个发行版的多数软件包得到了更新：更新了 42821 个软件包（占 `bookworm` 软件包总数的 72%）。而且，由于各种原因，有相当数量的软件包（9519 个，占 `bookworm` 软件包总数的 16%）从这次发行中被删除了。您将不会看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”；参见过时的软件包。

2.3.1 桌面和知名软件包

如往常一样，Debian 也提供了多个桌面程序和环境。提供的桌面环境包括 GNOME 43, KDE Plasma 5.27, LXDE 11, LXQt 1.2.0, MATE 1.26, 以及 Xfce 4.18。

生产力应用也得到了升级，包括办公套件：

- LibreOffice 已升级到 7.4 版；
- GNUMcash 已升级到 4.13 版；

这次发行包含了许多软件的更新，其中包括：

软件包	在 12 (bookworm) 中的版本	在 13 (trixie) 中的版本
Apache	2.4.54	2.4.57
Bash	5.1	5.2.15
BIND DNS Server	9.16	9.18
Cryptsetup	2.3	2.6
Emacs	27.1	28.2
Exim default e-mail server	4.94	4.96
GNU Compiler Collection as default compiler	10.2	12.2
GIMP	2.10.22	2.10.34
GnuPG	2.2.27	2.2.40
Inkscape	1.0.2	1.2.2
the GNU C library	2.31	2.36
Linux 内核映像	5.10 系列	6.1 系列
LLVM/Clang 工具链	9.0.1 和 11.0.1 (默认) 和 13.0.1	13.0.1 和 14.0 (默认) 和 15.0.6
MariaDB	10.5	10.11
Nginx	1.18	1.22
OpenJDK	11	17
OpenLDAP	2.4.57	2.5.13
OpenSSH	8.4p1	9.2p1
OpenSSL	1.1.1n	3.0.8
Perl	5.32	5.36
PHP	7.4	8.2
Postfix MTA	3.5	3.7
PostgreSQL	13	15
Python 3	3.9.2	3.11.2
Rustc	1.48	1.63
Samba	4.13	4.17
Systemd	247	252
Vim	8.2	9.0

2.3.2 手册页翻译更新

感谢翻译人员的努力，许多 man-page 格式的文档支持的语言比以往任何时候都要多。例如，许多手册页现在支持捷克语、丹麦语、希腊语、芬兰语、印度尼西亚语、马其顿语、挪威语（书面挪威语）、俄语、塞尔维亚语、瑞典语、乌克兰语和越南语，并且所有 **systemd** 手册页现在都支持德语。

To ensure the `man` command shows the documentation in your language (where possible), install the right `manpages-lang` package and make sure your locale is correctly configured by using

```
# dpkg-reconfigure locales
```

。

2.3.3 来自 Debian Med Blend 团队的新闻

正如每次发布一样，医学和生命科学领域新增了很多软件包。新软件包 **shiny-server** 尤其值得一提，因为它能使用 R 使 Web 科学应用程序变得更简单。我们还继续努力为 Debian Med 团队维护的软件包提供持续集成支持。

Debian Med 团队总是对用户的反馈很感兴趣，尤其是对尚未打包的自由软件的打包请求，以及从新软件包或 testing 中的更高版本的软件包向后移植的请求。

如需安装由 Debian Med 团队维护的软件包，您可以安装名为 med-* 的元软件包；在 Debian bookworm 中，其版本号为 3.8.x。您还可以访问 [Debian Med 任务页面](#) 以查看 Debian 中全部可用的生物和医疗软件。

2.3.4 来自 Debian Astro Blend 团队的新闻

Debian bookworm 带来了 Debian Astro Pure Blend 的 4.0 版本，它对于专业天文学家、天文爱好者和所有对天文学感兴趣的人来说，仍然是一个很棒的一站式解决方案。Debian Astro 中几乎所有的软件包都更新到了新版本，不过也有几个新增的软件包。

对于射电天文学家，新增了开源相关器 **openvbi**。新软件包 **astap** 和 **planetary-system-stacker** 对于图像堆叠和天体测量的解析很有用。我们还打包了大量支持 INDI 协议的新驱动程序和库，并将它们随 Debian 一起提供。

新增了新的 Astropy 附属包 **python3-extinction**、**python3-sncosmo**、**python3-specreduce** 和 **python3-synphot**，以及围绕 **python3-yt** 和 **python3-sunpy** 构建的软件包。Python 对 ASDF 文件格式的支持得到了很大扩展，而 Java 生态系统扩展了处理 ECSV 和 TFCAT 文件格式的库，主要是和 **topcat** 配合使用。

请浏览 [Astro Blend 页面](#) 获取完整的列表以及更多信息。

2.3.5 ARM64 的安全启动

trixie 重新支持了 ARM64 上的安全启动。使用支持 UEFI 的 ARM64 硬件的用户可以在启用安全启动模式的情况下启动系统，并完全享受此安全特性带来的好处。要启用安全启动，请确保安装了 **grub-efi-arm64-signed** 和 **shim-signed** 软件包、在您设备的固件界面中启用安全启动，然后重新启动。

[维基页面](#) 提供了更多关于如何使用和调试安全启动的信息。

Debian 安装程序是 Debian 官方的安装系统。它提供了多种安装方式。实际可用的安装方式取决于您的处理器架构。

Images of the installer for trixie can be found together with the Installation Guide on the Debian website (<https://www.debian.org/releases/trixie/debian-installer/>).

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

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

You may also want to check the errata for debian-installer at <https://www.debian.org/releases/trixie/debian-installer#errata> for a list of known issues.

3.1 安装系统有哪些新特性？

自从 Debian 安装程序上一次随 Debian 12 发布以来，我们进行了大量的开发工作，以提供更好的硬件支持及各种激动人心的新功能与改进。

If you are interested in an overview of the changes since bookworm, please check the release announcements for the trixie beta and RC releases available from the Debian Installer's [news history](#).

3.2 云服务安装

The [cloud team](#) publishes Debian trixie for several popular cloud computing services including:

- Amazon Web Services
- Microsoft Azure
- OpenStack
- 普通 VM

云映像将使用 `cloud-init` 提供自动化的钩子，并使用特别优化过的内核软件包和 `grub` 配置达到快速建立实例的效果。在需要的场合，映像将支持不同的硬件架构；同时云团队致力于支持云服务提供商提供的所有功能。

The cloud team will provide updated images until the end of the LTS period for trixie. New images are typically released for each point release and after security fixes for critical packages. The cloud team's full support policy can be found [here](#).

More details are available at <https://cloud.debian.org/> and on the [wiki](#).

3.3 容器和虚拟机映像

可以在 [Docker Hub](#) 找到多种架构的 Debian trixie 容器映像。除了标准映像之外，另有提供剪裁版（“slim”版本）以便缩减磁盘使用量。

用于 Hashicorp Vagrant VM manager 的虚拟机映像在 [Vagrant Cloud](#) 处发布。

从 Debian 12 (bookworm) 升级

4.1 升级前的准备

我们建议您在升级前阅读 *trixie* 中需要注意的问题。它描述的潜在问题与升级过程没有直接联系，但您仍可能需要了解这些信息。

4.1.1 备份数据和配置文件

Before upgrading your system, it is strongly recommended that you make a full backup, or at least back up any data or configuration information you can't afford to lose. The upgrade tools and process are quite reliable, but a hardware failure in the middle of an upgrade could result in a severely damaged system.

The main things you'll want to back up are the contents of `/etc`, `/var/lib/dpkg`, `/var/lib/apt/extended_states` and the output of:

```
$ dpkg --get-selections '*' # (the quotes are important)
```

If you use `aptitude` to manage packages on your system, you will also want to back up `/var/lib/aptitude/pkgstates`.

The upgrade process itself does not modify anything in the `/home` directory. However, some applications (e.g. parts of the Mozilla suite, and the GNOME and KDE desktop environments) are known to overwrite existing user settings with new defaults when a new version of the application is first started by a user. As a precaution, you may want to make a backup of the hidden files and directories ("dotfiles") in users' home directories. This backup may help to restore or recreate the old settings. You may also want to inform users about this.

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。

4.1.2 提前告知用户

It's wise to inform all users in advance of any upgrades you're planning, although users accessing your system via an `ssh` connection should notice little during the upgrade, and should be able to continue working.

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

You will have to do a kernel upgrade when upgrading to trixie, so a reboot will be necessary. Typically, this will be done after the upgrade is finished.

4.1.3 准备服务停机

系统提供的服务可能与升级包含的软件包相关联。如果存在这种情况，请注意，在升级期间，当相关软件包被更换和配置时，这些服务将被停止。在此期间，这些服务将无法使用。

The precise downtime for these services will vary depending on the number of packages being upgraded in the system, and it also includes the time the system administrator spends answering any configuration questions from package upgrades. Notice that if the upgrade process is left unattended and the system requests input during the upgrade there is a high possibility of services being unavailable¹ for a significant period of time.

If the system being upgraded provides critical services for your users or the network², you can reduce the downtime if you do a minimal system upgrade, as described in *Minimal system upgrade*, followed by a kernel upgrade and reboot, and then upgrade the packages associated with your critical services. Upgrade these packages prior to doing the full upgrade described in *Upgrading the system*. This way you can ensure that these critical services are running and available through the full upgrade process, and their downtime is reduced.

4.1.4 准备故障恢复

Although Debian tries to ensure that your system stays bootable at all times, there is always a chance that you may experience problems rebooting your system after the upgrade. Known potential issues are documented in this and the next chapters of these Release Notes.

因此，当您的系统重启失败（或对于远程管理的系统来说，无法连接网络）时，请确保您能够将其恢复。

If you are upgrading remotely via an `ssh` link it is recommended that you take the necessary precautions to be able to access the server through a remote serial terminal. There is a chance that, after upgrading the kernel and rebooting, you will have to fix the system configuration through a local console. Also, if the system is rebooted accidentally in the middle of an upgrade there is a chance you will need to recover using a local console.

For emergency recovery we generally recommend using the *rescue mode* of the trixie Debian Installer. The advantage of using the installer is that you can choose between its many methods to find one that best suits your situation. For more information, please consult the section "Recovering a Broken System" in chapter 8 of the Installation Guide (at <https://www.debian.org/releases/trixie/installmanual>) and the [Debian Installer FAQ](#).

如果上述步骤失败了，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或者 `live install` 镜像。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。

¹ 如果 `debconf` 优先级被设置为非常高的级别，则可能不会弹出配置提示，但如果默认应答不适用于您的系统，依赖于此的服务将无法启动。

² 例如：DNS 或 DHCP 服务，特别是当没有冗余或故障转移时。以 DHCP 为例，如果租用时间低于升级过程完成所需的时间，终端用户可能会断开网络连接。

initrd 引导时使用调试环境

The **initramfs-tools** package includes a debug shell³ in the initrds it generates. If for example the initrd is unable to mount your root file system, you will be dropped into this debug shell which has basic commands available to help trace the problem and possibly fix it.

Basic things to check are: presence of correct device files in `/dev`; what modules are loaded (`cat /proc/modules`); output of `dmesg` for errors loading drivers. The output of `dmesg` will also show what device files have been assigned to which disks; you should check that against the output of `echo $ROOT` to make sure that the root file system is on the expected device.

If you do manage to fix the problem, typing `exit` will quit the debug shell and continue the boot process at the point it failed. Of course you will also need to fix the underlying problem and regenerate the initrd so the next boot won't fail again.

systemd 引导时使用调试 shell

If the boot fails under systemd, it is possible to obtain a debug root shell by changing the kernel command line. If the basic boot succeeds, but some services fail to start, it may be useful to add `systemd.unit=rescue.target` to the kernel parameters.

Otherwise, the kernel parameter `systemd.unit=emergency.target` will provide you with a root shell at the earliest possible point. However, this is done before mounting the root file system with read-write permissions. You will have to do that manually with:

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

Another approach is to enable the systemd "early debug shell" via the `debug-shell.service`. On the next boot this service opens a root login shell on `tty9` very early in the boot process. It can be enabled with the kernel boot parameter `systemd.debug-shell=1`, or made persistent with `systemctl enable debug-shell` (in which case it should be disabled again when debugging is completed).

More information on debugging a broken boot under systemd can be found in the [Freedesktop.org Diagnosing Boot Problems](#) article.

4.1.5 为升级准备安全环境

重要: If you are using some VPN services (such as **tinc**) consider that they might not be available throughout the upgrade process. Please see *Prepare for downtime on services*.

In order to gain extra safety margin when upgrading remotely, we suggest that you run upgrade processes in the virtual console provided by the `screen` program, which enables safe reconnection and ensures the upgrade process is not interrupted even if the remote connection process temporarily fails.

使用由 **micro-evtd** 提供的 `watchdog` 守护程序的用户应该在升级前停止守护进程并禁用 `watchdog timer`，以避免在升级过程中发生不必要的重启：

```
# service micro-evtd stop
# /usr/sbin/microap1 -a system_set_watchdog off
```

³ 此功能可以通过在您的启动参数中添加参数 `panic=0` 来禁用。

4.2 Start from "pure" Debian

The upgrade process described in this chapter has been designed for "pure" Debian stable systems. APT controls what is installed on your system. If your APT configuration mentions additional sources besides bookworm, or if you have installed packages from other releases or from third parties, then to ensure a reliable upgrade process you may wish to begin by removing these complicating factors.

The main configuration file that APT uses to decide what sources it should download packages from is `/etc/apt/sources.list`, but it can also use files in the `/etc/apt/sources.list.d/` directory - for details see `sources.list(5)`. If your system is using multiple source-list files then you will need to ensure they stay consistent.

4.2.1 Upgrade to Debian 12 (bookworm)

Only upgrades from Debian 12 (bookworm) are supported. Display your Debian version with:

```
$ cat /etc/debian_version
```

Please follow the instructions in the Release Notes for Debian 12 at <https://www.debian.org/releases/bookworm/releasenotes> to upgrade to Debian 12 first if needed.

4.2.2 升级至最新的小版本更新

This procedure assumes your system has been updated to the latest point release of bookworm. If you have not done this or are unsure, follow the instructions in [升级您的 bookworm 系统](#).

4.2.3 Debian Backports

Debian Backports allows users of Debian stable to run more up-to-date versions of packages (with some tradeoffs in testing and security support). The Debian Backports Team maintains a subset of packages from the next Debian release, adjusted and recompiled for usage on the current Debian stable release.

Packages from bookworm-backports have version numbers lower than the version in trixie, so they should upgrade normally to trixie in the same way as "pure" bookworm packages during the distribution upgrade. While there are no known potential issues, the upgrade paths from backports are less tested, and correspondingly incur more risk.

小心: While regular Debian Backports are supported, there is no clean upgrade path from sloppy backports (which use APT source-list entries referencing bookworm-backports-sloppy).

As with *Unofficial sources*, users are advised to remove "bookworm-backports" entries from their APT source-list files before the upgrade. After it is completed, they may consider adding "trixie-backports" (see <https://backports.debian.org/Instructions/>).

For more information, consult the [Backports Wiki page](#).

4.2.4 准备软件包数据库

在继续升级之前，还应确保软件包数据库已准备就绪。如果您是 `aptitude` 或 `synaptic` 等其他软件包管理器的用户，请查看任何待处理的操作。在包管理器中被计划安装或删除的软件包可能会干扰升级过程。请注意，仅当您的 `APT source-list` 文件仍指向“bookworm”而不是“stable”或“trixie”时，才能更正此项；请参阅检查您的 `APT source-list` 文件。

4.2.5 移除过时的软件包

在升级前从您的系统中移除过时的软件包是一个好想法。未移除即升级可能会使升级过程复杂化，且它们因不再得到维护而可能带来安全风险。

4.2.6 移除非 Debian 软件包

Below there are two methods for finding installed packages that did not come from Debian, using either `apt` or `apt-forktracer`. Please note that neither of them are 100% accurate (e.g. the `apt` example will list packages that were once provided by Debian but no longer are, such as old kernel packages).

```
$ apt list '?narrow(?installed, ?not(?origin(Debian)))'  
$ apt-forktracer | sort
```

4.2.7 清理配置文件残留

A previous upgrade may have left unused copies of configuration files; *old versions* of configuration files, versions supplied by the package maintainers, etc. Removing leftover files from previous upgrades can avoid confusion. Find such leftover files with:

```
# find /etc -name '*.dpkg-*' -o -name '*.ucf-*' -o -name '*.merge-error'
```

4.2.8 non-free 和 non-free-firmware 区

If you have non-free firmware installed it is recommended to add `non-free-firmware` to your `APT sources-list`. For details see [软件仓库分区](#) and [非自由固件在仓库中被划分到了单独的分区](#).

4.2.9 proposed-updates 区

If you have listed the `proposed-updates` section in your `APT source-list` files, you should remove it before attempting to upgrade your system. This is a precaution to reduce the likelihood of conflicts.

4.2.10 非官方源

If you have any non-Debian packages on your system, you should be aware that these may be removed during the upgrade because of conflicting dependencies. If these packages were installed by adding an extra package archive in your APT source-list files, you should check if that archive also offers packages compiled for trixie and change the source item accordingly at the same time as your source items for Debian packages.

Some users may have *unofficial* backported "newer" versions of packages that *are* in Debian installed on their bookworm system. Such packages are most likely to cause problems during an upgrade as they may result in file conflicts⁴. *Possible issues during upgrade* has some information on how to deal with file conflicts if they should occur.

4.2.11 禁用 APT pinning

如果您已经设置 APT 从一个非 stable (如 testing) 版安装特定的软件包, 您可能必须改变 APT pinning 设置 (保存在 `/etc/apt/preferences` 和 `/etc/apt/preferences.d/` 中), 以允许将软件包升级至新的 stable 中的版本。如需更多有关 APT pinning 的信息, 请参阅 `apt_preferences(5)`。

4.2.12 检查 gpgv 是否已安装

APT 需要 `gpgv` 版本 2 或更高, 以验证给 trixie 发布版本签名的密钥。鉴于 `gpgv1` 在技术上满足了依赖关系, 但仅能在一些特殊情况下使用, 您可能想要使用以下命令确保已经安装了正确的版本:

```
# apt install gpgv
```

4.2.13 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 `aptitude` 来检查系统中的所有软件包的状态, 也可以使用如下命令

```
$ dpkg -l | pager
```

或是

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

或者您也可以使用 `apt`。

```
# apt list --installed > ~/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。

```
$ apt-mark showhold
```

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

`apt` 中的“hold”状态的包可以使用以下命令修改:

⁴ 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。

```
# apt-mark 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，最好确保您的 APT `source-list` 文件仍然指向 `bookworm`，就像检查您的 APT `source-list` 文件中所解释的那样。

4.3 准备 APT source-list 文件

在开始升级之前，您必须重新配置 APT 的 `source-list` 文件（`/etc/apt/sources.list` 以及 `/etc/apt/sources.list.d/` 下的文件）以添加用于 `trixie` 的源以及在通常情况下移除用于 `bookworm` 的源。

APT 将考虑所有已配置的源中包含的所有软件包，并安装具有最高版本号的软件包，优先考虑文件中的第一个条目。因此，如果您有多个镜像位置，请将本地硬盘放在第一位，然后是 CD-ROM，最后是远程镜像。

一个发行版通常既能通过它的代号（如：“`bookworm`”，“`trixie`”）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，您绝对不会因为新版本的发布而感到惊讶，因此本文使用这种方法。当然，这也意味着您不得不自己关注新版的发行公告。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。

Debian 提供了两个公告邮件列表，以帮助您及时了解与 Debian 发布相关的信息：

- 通过 订阅 Debian 公告邮件列表，每次 Debian 发布新版本时，您都会收到通知。例如当“`trixie`”从“`testing`”变为“`stable`”时。
- 通过 订阅 Debian 安全公告邮件列表，每次 Debian 发布安全公告时，您都会收到通知。

4.3.1 添加互联网 APT 源

在新版安装中，APT 默认使用 Debian APT CDN 服务，该服务确保软件包自动从网络上离您较近的一个服务器下载。由于这是一项相对较新的服务，旧版安装的配置可能仍然指向 Debian Internet 主服务器之一或其中一个镜像。如果您尚未这样做，建议在 APT 配置中切换为使用 CDN 服务。

要使用 CDN 服务，请在 APT 源配置中添加这样一行（假设您在使用 `main` 和 `contrib`）：

```
deb https://deb.debian.org/debian trixie main contrib
```

添加新源后，通过在以前存在的“`deb`”行前放置一个井号（`#`）来禁用它们。

不过，如果您通过使用在网络上更接近您的特定的镜像得到了更好的结果，您仍然可以继续这么做。

Debian 镜像地址可以在 <https://www.debian.org/distrib/ftplist> 找到（参见“list of Debian mirrors”一节）。

例如，假设离您最近的 Debian 镜像是 <http://mirrors.kernel.org>。如果使用网络浏览器检查该镜像，您会注意到主目录的组织方式如下：

```
http://mirrors.kernel.org/debian/dists/trixie/main/...
http://mirrors.kernel.org/debian/dists/trixie/contrib/...
```

要使用给定的镜像配置 APT，请添加类似如下的一行（同样，假设您正在使用 `main` 和 `contrib`）：

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

注意“`dists`”会隐式添加，而版本名称后的参数则用于将路径扩展到多个目录。

同样，添加新源后，禁用以前存在的软件源条目。

4.3.2 添加本地镜像 APT 源

您可能希望修改 APT source-list 文件以使用本地磁盘上的镜像（可能挂载在 NFS 上），而不是使用远程软件包镜像。

例如，您的软件包镜像可能位于 /var/local/debian/ 下，并且具有如下的目录结构：

```
/var/local/debian/dists/trixie/main/...  
/var/local/debian/dists/trixie/contrib/...
```

要让 apt 使用它，需要把这一行添加至 sources.list 文件：

```
deb file:/var/local/debian trixie main contrib
```

注意“dists”会隐式添加，而版本名称后的参数则用于将路径扩展到多个目录。

添加新源后，通过在 APT source-list 文件中以前存在的软件源条目前放置一个井号（#）来禁用它们。

4.3.3 从光学介质中添加 APT 源

如果您只想使用 DVD（或 CD 或蓝光光盘），请在所有 APT source-list 文件中注释掉现有条目，方法是在它们前面放置一个井号（#）。

确保在 /etc/fstab 中有一行允许您挂载 CD-ROM 于 /media/cdrom 挂载点。例如，假设 /dev/sr0 就是您的 CD-ROM，/etc/fstab 中应该带有一行：

```
/dev/sr0 /media/cdrom auto noauto,ro 0 0
```

注意在第四列 noauto,ro 之间必须没有空格。

要验证设置是否有效，插入一片 CD，尝试运行

```
# mount /media/cdrom # this will mount the CD to the mount point  
# ls -alF /media/cdrom # this should show the CD's root directory  
# umount /media/cdrom # this will unmount the 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。

4.4 升级软件包

从以前的 Debian 版本升级的推荐方法是使用包管理工具 apt。

备注： apt 被设计用于交互式使用，故不应在脚本中使用。在脚本中，应该使用 apt-get，它具有更适合进行文本处理的稳定输出。

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
```


接下来，您应该仔细检查确认 APT 源条目（在 `/etc/apt/sources.list` 以及 `/etc/apt/sources.list.d/` 下的文件）要么指向“trixie”，要么指向“stable”。不应该有任何源条目指向 `bookworm`。

备注： CD-ROM 的源有时会指向“unstable”。尽管这令人困惑，但您 不应该改变它。

4.4.1 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，您就有一份日志，而且需要的话，它也可以在缺陷报告中提供准确的信息。要开始记录，输入：

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

or similar. If you have to rerun the typescript (e.g. if you have to reboot the system) use different *step* values to indicate which step of the upgrade you are logging. Do not put the typescript file in a temporary directory such as `/tmp` or `/var/tmp` (files in those directories may be deleted during the upgrade or during any restart).

The typescript will also allow you to review information that has scrolled off-screen. If you are at the system’s console, just switch to VT2 (using `Alt+F2`) and, after logging in, use

```
# less -R ~root/upgrade-trixie.script
```

to view the file.

在完成升级后，您可以在提示符下输入 `exit` 停止 `script`。

`apt` 也会把软件包的状态变更记录在 `/var/log/apt/history.log`，并把终端输出记录在 `/var/log/apt/term.log`。`dpkg` 会额外地把软件包的状态变更记录在 `/var/log/dpkg.log`。如果您使用 `aptitude`，它也会把软件包的状态变更记录在 `/var/log/aptitude`。

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-trixie-step.time ~/upgrade-trixie-step.script
```

4.4.2 更新软件包列表

首先，需要获取新发行版的可用包列表。执行：

```
# apt update
```

备注： `apt-secure` 的用户可能会在使用 `aptitude` 或 `apt-get` 时遇到问题。对于 `apt-get`，您可以使用 `apt-get update --allow-releaseinfo-change`。

4.4.3 确保您有足够的空间升级

You have to make sure before upgrading your system that you will have sufficient hard disk space when you start the full system upgrade described in *Upgrading the system*. First, any package needed for installation that is fetched from the network is stored in `/var/cache/apt/archives` (and the `partial/` subdirectory, during download), so you must make sure you have enough space on the file system partition that holds `/var/` to temporarily download the packages that will be installed in your system. After the download, you will probably need more space in other file system partitions in order to both install upgraded packages (which might contain bigger binaries or more data) and new packages that will be pulled in for the upgrade. If your system does not have sufficient space you might end up with an incomplete upgrade that is difficult to recover from.

apt 可以显示有关安装所需磁盘空间的详细信息。在执行升级之前，您可以通过运行以下命令来查看此估计：

```
# apt -o APT::Get::Trivial-Only=true full-upgrade
[ ... ]
XXX upgraded, XXX newly installed, XXX to remove and XXX not upgraded.
Need to get xx.xMB of archives.
After this operation, AAAMB of additional disk space will be used.
```

备注： 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 *Minimal system upgrade* before running this command to estimate the disk space.

如果您没有足够的空间进行升级，apt 将通过类似这样的消息警告您：

```
E: You don't have enough free space in /var/cache/apt/archives/.
```

在这种情况下，请确保事先释放空间。您可以：

- 删除以前安装时下载的软件包（位于 `/var/cache/apt/archives`）。通过运行 `apt clean` 来清理软件包缓存，这将会删除所有以前下载过的软件包文件。
- 删除被遗忘的软件包。如果您曾用 `aptitude` 或 `apt` 手动在 `bookworm` 中安装过包，它将会保存手动安装的记录，并且对于由依赖关系拉入的包，在主包删除时能自动识别为不再需要的并标记为冗余。不会将您手动安装的包标记为删除。要删除自动安装的、不再使用的软件包，请运行：

```
# apt autoremove
```

您可以用其它的工具来找出多余的包，例如 `deborphan`、`debfooster` 或 `cruft`。不要盲目地移除那些工具找到的包，尤其是当您使用了激进的非默认的选项时，很容易产生假阳性结果。强烈推荐您在移除它们之前，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）。

- Remove packages that take up too much space and are not currently needed (you can always reinstall them after the upgrade). 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 find the packages that just take up the most disk space with `dpigs` (available in the **debian-goodies** package) or with `wajig` (running `wajig size`). They can also be found with **aptitude**. Start `aptitude` in full-terminal mode, select Views > New Flat Package List, press `l` and enter `~i`, then press `S` and enter `~installsize`. This will give you a handy list to work with.
- 如果不需要的话可以从系统中移除翻译和本地化文件。您可以安装并配置 **localepurge** 软件包，这样只有一小部分指定的 locale 被保留在系统中。这将减少 `/usr/share/locale` 所使用的磁盘空间。
- 将 `/var/log/` 下的系统日志临时放到其它系统，或是永久删除。
- 使用临时的 `/var/cache/apt/archives`：您可以使用其他文件系统中的临时缓存目录（USB 存储设备，临时硬盘，已在使用的文件系统等等）。

备注： 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 驱动器挂在 `/media/usbkey` 下：

1. 删除以前安装时下载的包：

```
# apt clean
```

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

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

3. 将临时缓存目录挂载至当前使用的目录：

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

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

```
# umount /var/cache/apt/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。

- Do a minimal upgrade of the system (see *Minimal system upgrade*) or partial upgrades of the system followed by a full upgrade. This will make it possible to upgrade the system partially, and allow you to clean the package cache before the full upgrade.

请注意，为了安全地删除软件包，建议将 APT source-list 文件切换回 bookworm，如检查您的 *APT source-list* 文件中所述。

4.4.4 停止监控系统

因为 apt 可能需要暂时停止您的计算机上运行的服务，所以可能有必要在升级期间停止有可能重启其他被停止的服务的监控服务。Debian 中的 **monit** 就是这种服务的一个例子。

4.4.5 最小系统升级

In some cases, doing the full upgrade (as described below) directly might remove large numbers of packages that you will want to keep. We therefore recommend a two-part upgrade process: first a minimal upgrade to overcome these conflicts, then a full upgrade as described in *Upgrading the system*.

要开始，请运行：

```
# apt upgrade --without-new-pkgs
```

这会升级那些不需要删除或安装其它任何包的软件。

当系统空间紧张，并且由于空间限制而无法运行完整升级时，最小的系统升级也将非常有用。

If the **apt-listchanges** package is installed, it will (in its default configuration) show important information about upgraded packages in a pager after downloading the packages. Press `q` after reading to exit the pager and continue the upgrade.

4.4.6 升级系统

一旦您完成了上述步骤，您就可以继续进行升级的主要部分。请执行：

```
# apt full-upgrade
```

这将对系统进行一次全面的升级，安装所有包的最新可用版本，解决不同发行版本的软件包之间的所有潜在的依赖关系变化。如有必要，它会安装一些新包（通常是新版本的库，或是被重命名的软件包），并删除任何存在冲突的过时包。

当从一组 CD/DVD/BD 升级时，您会在升级期间的某些时刻被要求插入指定的盘片。您可能不得不多次插入相同的盘片，这是因为互相依赖的包已经被散布到整个盘片组中去了。

无法在不改变其他软件包的安装状态的情况下升级的已安装软件包将停留在当前版本（显示为“未被升级”）。这个问题可以通过使用 `aptitude` 选择这些包来安装或是用 `apt install` 软件包来解决。

4.5 升级期间可能遇到的问题

以下部分描述升级到 `trixie` 期间已知可能会出现的问题。

4.5.1 Full-upgrade 失败，显示“无法立即配置”

在某些情况下，下载软件包后，`apt full-upgrade` 步骤可能会失败：

```
E: Could not perform immediate configuration on 'package'. Please see man 5 apt.conf
↳under APT::Immediate-Configure for details.
```

如果发生这种情况，运行 `apt full-upgrade -o APT::Immediate-Configure=0` 应该可以使升级继续。

此问题的另一个可能解决方法是临时将 `bookworm` 和 `trixie` 源都添加到您的 `APT source-list` 文件中，并运行 `apt update`。

4.5.2 预期的删除

The upgrade process to `trixie` might ask for the removal of packages on the system. The precise list of packages will vary depending on the set of packages that you have installed. These release notes give general advice on these removals, but if in doubt, it is recommended that you examine the package removals proposed by each method before proceeding. For more information about packages obsoleted in `trixie`, see *Obsolete packages*.

4.5.3 冲突或预依赖循环

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，这是由于冲突或是预依赖循环。`apt` 会警告您这一点并中止升级。您可以通过在 `apt` 命令行中指定 `-o APT::Force-LoopBreak=1` 选项来解决这个问题。

有时一个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 `apt` 或

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

来消除某些引起问题的包，或是

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

在极端情况下，您可能不得不用类似下面的命令强制重新安装某个包

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

4.5.4 文件冲突

如果您从“纯净的”bookworm 系统升级就不会出现文件冲突，但如果您装有非官方的向后移植的软件包就可能出现冲突。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>
```

您可以尝试强制删除错误信息中的 * 最后 * 一行提到的软件包来解决文件冲突：

```
# 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 apt 命令来继续升级。

4.5.5 配置文件变化

升级期间，您将会被询问有关配置或是重新配置一些软件包的问题。如果您被问到是否用软件包维护者的版本替换 /etc/init.d 或 /etc/manpath.config 下的文件时，通常有必要回答“yes”来确保系统一致性。您总是可以恢复老版本的配置文件，因为它们会被保存为带有 .dpkg-old 后缀名的文件。

如果您不确定该做什么，那就记下软件包或文件的名称，以后再妥善处理这些问题。您可以通过在 typescript 文件中搜索来查看升级期间显示在屏幕上的信息。

4.5.6 将会话切换到控制台

如果您使用系统的本地控制台升级，则可能会发现在升级过程中的某些时候，控制台切换到了不同的视图，并且您无法看到升级过程。例如，在含有图形界面的系统中，当显示管理器重新启动时，就可能发生这种情况。

To recover the console where the upgrade was running you will have to use `Ctrl+Alt+F1` (if in the graphical startup screen) or `Alt+F1` (if in the local text-mode console) to switch back to the virtual terminal 1. Replace `F1` with the function key with the same number as the virtual terminal the upgrade was running in. You can also use `Alt+Left Arrow` or `Alt+Right Arrow` to switch between the different text-mode terminals.

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不适用。

4.6.1 安装内核元软件包

当您从 bookworm 完整升级至 trixie 时，如果没有安装，强烈推荐您安装 `linux-image-*` 元包。这些元包将在升级过程中自动引入新版本的内核。您可以运行以下命令验证是否安装了上述元包：

```
$ dpkg -l 'linux-image*' | grep ^ii | grep -i meta
```

如果您没有看到任何输出，那么您需要手动安装一个新的 `linux-image` 软件包，或者安装 `linux-image` 元包。要查看可用的 `linux-image` 元包列表，请运行：

```
$ apt-cache search linux-image- | grep -i meta | grep -v transition
```

如果您不确定要选哪个包，那就运行 `uname -r` 并查找带有类似名称的包。例如，如果您看到“4.9.0-8-amd64”，那推荐您安装 **linux-image-amd64**。您也可以使用 `apt` 来查看每个包的详细描述，以帮助您选择最好用的那个。例如：

```
$ apt show linux-image-amd64
```

然后您应该使用 `apt install` 来安装它。安装此新内核后，您应该在下一个可能的时刻重新启动，以获得新内核版本提供的特性。但是，在升级之后的第一次重启之前，请先查看[升级后在重启前需要做的事](#)。

对于热爱冒险的用户，有一个简单的方法可以在 Debian 上编译自己的定制内核。安装由 **linux-source** 包提供的内核源码。您可以使用源文件的 `makefile` 中的 `deb-pkg` 目标来构建二进制包。更多信息可以在 [Debian Linux 内核手册](#) 中找到，**debian-kernel-handbook** 包也提供同样内容。

If possible, it is to your advantage to upgrade the kernel package separately from the main `full-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 *Minimal system upgrade*.

4.7 为下个发布版本做准备

升级完成后，您可以为下一个发布版本做些准备工作。

- Remove newly redundant or obsolete packages as described in [Make sure you have sufficient space for the upgrade and Obsolete packages](#). You should review which configuration files they use and consider purging the packages to remove their configuration files. See also [Purging removed packages](#).

4.7.1 清理已删除的软件包

通常建议清理已删除的软件包。如果这些软件包已经在之前的版本升级（例如升级到 bookworm）过程中被删除，或者它们是由第三方供应商提供的，则尤其如此。特别地，已知旧的 `init.d` 脚本会导致问题。

小心： 清理软件包通常也会清除其日志文件，因此您可能希望先备份它们。

以下命令显示所有已删除、但可能在系统上留下配置文件的软件包的列表（如果有的话）：

```
$ apt list '~c'
```

可以使用 `apt purge` 删除这些软件包。假设您想一次性清除它们，可以使用以下命令：

```
# apt purge '~c'
```

4.8 过时的软件包

引进新软件包的同时，trixie 也清除了一些曾位于 bookworm 的旧软件包。它不提供对这些过时包的升级。虽然没有什么可以阻止您继续使用过时的软件包，但 Debian 项目通常会在 trixie 发布一年后停止对它的安全更新⁵，并且在此期间通常不会提供其他支持。如果有的话，建议用可用的替代品替换它们。

软件包被从发行版中移除有很多原因，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缺陷，使得它们被认为不适用于 trixie。最后一种情况下，这些包有可能仍然位于“unstable”版之下。

以下命令可以列出并清除“过时的和在本地创建的软件包”：

```
$ apt list '~o'
# apt purge '~o'
```

Debian 缺陷跟踪系统 通常会提供有关这个包为什么会被移除的额外信息。您应该既查看此包自身的归档缺陷报告，同时也要查看 ftp.debian.org 伪软件包的归档缺陷报告。

要获得 trixie 的过时包列表，请参阅值得注意的 [过时软件包](#)。

4.8.1 过渡哑包

来自 bookworm 的一些软件包可能已在 trixie 中被过渡哑包（transitional dummy package）替换，这些软件包是用于简化升级的空占位符。例如，如果以前单个包的应用程序已被拆分为多个，则可以提供与旧包具有相同名称的过渡包，并设置合适的依赖以使新的包被安装。发生这种情况后，可以安全地移除冗余哑包。

绝大多数过渡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；特别是，一些“哑”包被设计为保持安装，以便引入完整的软件套件或跟踪某些程序的当前最新版本。您也可以使用 `deborphan` 命令的 `--guess-*` 选项（例如 `--guess-dummy`）来检测系统上的过渡哑包。

⁵ 或者，直到这段时间内有另一个版本发布为止。在任意时刻，一般仅对两个 stable 版本提供支持。

trixie 中需要注意的问题

Sometimes, changes introduced in a new release have side-effects we cannot reasonably avoid, or they expose bugs somewhere else. This section documents issues we are aware of. Please also read the errata, the relevant packages' documentation, bug reports, and other information mentioned in [扩展阅读](#).

5.1 升级到 trixie 时可能出现的问题

本节介绍从 bookworm 升级到 trixie 的相关问题。

5.1.1 非自由固件在仓库中被划分到了单独的分区

如[软件仓库分区](#)所述，非自由固件软件包已在仓库中被划分到了一个专门的分区，名为 `non-free-firmware`。为了确保已安装的非自由固件软件包能正常收到更新，您需要修改 APT 的配置。假设您仅仅需要将 `non-free` 分区加入 APT `source-list`，以便于安装固件，则修改过的 APT `source-list` 条目将类似于：

```
deb https://deb.debian.org/debian bookworm main non-free-firmware
```

如果 apt 指引您来阅读本章节，而您不想再收到关于此变更的提醒，您可以创建一个名为 `/etc/apt/apt.conf.d/no-bookworm-firmware.conf` 的 `apt.conf(5)` 文件，内容如下：

```
APT::Get::Update::SourceListWarnings::NonFreeFirmware "false";
```

5.1.2 设置系统时钟的软件包的变更

ntp 软件包曾经是与 NTP（Network Time Protocol，网络时间协议）服务器进行时间同步的默认方式。它已被 **ntpsec** 取代。

绝大多数用户不需要进行任何特殊的操作，即可从 **ntp** 过渡到 **ntpsec**。

在 **bookworm** 中，也有其他一些软件包提供类似的服务。Debian 现在默认使用 **systemd-timesyncd**，对于只需要 **ntp** 客户端来设置时钟的用户而言，它已经够用了。**bookworm** 也收录了 **chrony** 和 **openntpd**，它们支持更高级的特性，例如搭建您自己的 NTP 服务器。

5.1.3 Puppet 配置管理系统升级到了版本 7

Puppet 从版本 5 升级到了版本 7，完全跳过了 Puppet 6 系列。这给 Puppet 生态系统带来了多项重大变化。

经典的基于 Ruby 的 Puppet Master 5.5.x 应用已经被上游废弃，因此在 Debian 中已不再可用。它已被 Puppet Server 7.x 取代，该应用由软件包 **puppetserver** 提供。该软件包将作为过渡软件包 **puppet-master** 的依赖被自动安装。

某些情况下，Puppet Server 可以无缝替代 Puppet Master，不过您应当检查 `/etc/puppet/puppetserver` 下的配置文件，以确保新的默认值适合于您的部署。特别需要注意的是，`auth.conf` 的旧格式已被弃用，请阅读 `auth.conf` 文档了解详细信息。

推荐的方式是先升级服务器，再升级客户端。Puppet 7 Server 与旧的客户端向后兼容；Puppet 5 Server 可以处理被升级的代理，但是不能注册新的 Puppet 7 代理。所以，如果您先部署新的 Puppet 7 代理，再升级服务器，您将无法将它们加入 **fleet**。

puppet 软件包已被 **puppet-agent** 软件包取代，并已成为过渡软件包，以确保平滑升级。

最后，**puppetdb** 软件包虽然已在 **bullseye** 中被移除，但是重新进入了 **bookworm**。

5.1.4 youtube-dl 被 yt-dlp 取代

可以下载多个网站的视频（包括但不限于 YouTube 视频）的流行工具，**youtube-dl**，不再被包含于 Debian。它已经被一个空的过渡软件包取代，该软件包会安装 **yt-dlp** 软件包。**yt-dlp** 是 **youtube-dl** 的分支，当前开发比较活跃。

There are no compatibility wrappers provided, so you'll need to modify your scripts and personal behavior to call `yt-dlp` instead of `youtube-dl`. The functionality should be mostly the same, although some options and behavioral details have changed. Be sure to check `yt-dlp.1` for details, and in particular `yt-dlp.1.html#Differences_in_default_behavior`.

5.1.5 不同版本的 Fcitx 将不能共存

软件包 **fcitx** 和 **fcitx5** 提供了流行的 Fcitx 输入法框架的第 4 版和第 5 版。遵照上游的建议，它们将不能在一个操作系统中共存。之前同时安装了 **fcitx** 和 **fcitx5** 的用户需要决定想要保留 Fcitx 的哪个版本。

强烈建议用户在升级之前清除不想要的 Fcitx 版本的所有相关的软件包（Fcitx 4 是 `fcitx-*`，Fcitx 5 是 `fcitx5-*`）。升级完成之后，请考虑重新执行 `im-config` 以选择系统中想使用的输入法框架。

您可以阅读 发布于邮件列表中的通知 以了解更多背景信息（通知以简体中文写成）。

5.1.6 MariaDB 软件包名不再包含版本号

和 bullseye 在 MariaDB 软件包名中包含版本号（例如 `mariadb-server-10.5` 和 `mariadb-client-10.5`）不同，在 bookworm 中，对应的 MariaDB 10.11 软件包名完全不包含版本号（例如 `mariadb-server` 和 `mariadb-client`）。在软件包版本元数据中仍然可以看到 MariaDB 的版本。

There is at least one known upgrade scenario (Bug #1035949) where the transition to versionless package names fails: running

```
# apt-get install default-mysql-server
```

可能会失败，因为在升级过程中，在 MariaDB 服务器 SysV init 服务使用 `mariadb-admin` 执行关闭操作前，删除了 `mariadb-client-10.5` 和它包含的文件 `/usr/bin/mariadb-admin`。解决办法是执行

```
# apt upgrade
```

然后再执行

```
# apt full-upgrade
```

。

要了解关于 MariaDB 软件包名变更的更多信息，请阅读 [/usr/share/doc/mariadb-server/NEWS.Debian.gz](#)。

5.1.7 系统日志的变更

大多数系统已经不再需要 `rsyslog` 软件包，因此您可能可以移除它。

许多程序通过日志消息告诉用户它们在做什么。这些消息可以用 `systemd` 的“journal”进行管理，也可以用“syslog 守护进程”进行管理，比如 `rsyslog`。

在 bookworm 中，会默认安装 `rsyslog`，并且将 `systemd journal` 配置为转发日志消息到 `rsyslog`，后者会将消息写到多个文本文件中，例如 `/var/log/syslog`。

从 trixie 开始，不再默认安装 `rsyslog`。如果您不想继续使用 `rsyslog`，在升级完成后您可以先运行

```
# apt-mark auto rsyslog
```

将它标记为自动安装，再运行

```
# apt autoremove
```

来删除它（如果可能的话）。如果您是从旧的 Debian 版本升级来的，并且没有使用默认设置，那么 `journal` 可能不会把消息保存到持久化存储。要启用此功能，请阅读 [journal.conf\(5\)](#) 中的步骤。

如果您决定不再使用 `rsyslog`，您可以使用 `journalctl` 阅读以二进制格式保存在 `/var/log/journal` 的日志消息。例如，

```
# journalctl -e
```

显示 `journal` 中最近的一些日志消息，而

```
# journalctl -ef
```

会在新的消息被写入的同时显示它们（和运行

```
# tail -f /var/log/syslog
```

类似）。

5.1.8 影响日志分析工具（如 logcheck）的 rsyslog 变更

rsyslog 现在默认使用“高精度时间戳”，这可能影响其他分析系统日志的工具。您可以阅读 `rsyslog.conf(5)` 了解关于如何修改此设置的更多信息。

由于时间戳上的变更，您可能需要更新本地创建的 **logcheck** 规则。`logcheck` 读取系统日志中的消息，并将它们和自定义的正则表达式（称为“规则”）构成的数据库进行对比。您需要更新用于匹配消息的创建时间的规则，以兼容新的 `rsyslog` 格式。**logcheck-database** 软件包提供的默认规则已经更新了，但其他规则，包括本地创建的规则，需要被更新，才能识别出新的格式。`/usr/share/doc/logcheck-database/NEWS.Debian.gz` 中有一个可以更新本地 `logcheck` 规则脚本文本。

5.1.9 rsyslog 创建的日志文件更少

rsyslog 修改了它将创建哪些日志文件，所以您可以删除 `/var/log` 中的一些文件了。

If you are continuing to use **rsyslog** (see *Changes to system logging*), some log files in `/var/log` will no longer be created by default. The messages that were written to these files are also in `/var/log/syslog` but are no longer created by default. Everything that used to be written to these files will still be available in `/var/log/syslog`.

不再创建的文件包括：

- `/var/log/mail.{info,warn,err}`

这些文件包含来自本地的邮件传送代理（MTA）的消息，按照优先级分成不同的文件。

因为 `/var/log/mail.log` 已经包含了所有和邮件相关的消息，所以可以安全地删除这些文件（以及对应的已被归档的旧文件）。如果您使用这些文件来监测异常状况，可以考虑使用诸如 `logcheck` 的工具来替代。

- `/var/log/lpr.log`

这些文件包含和打印有关的日志消息。`Debian` 默认的打印系统是 **cups**，它不使用该文件，所以除非您使用另一个打印系统，否则就可以删除这些文件（以及对应的已被归档的旧文件）。

- `/var/log/{messages,debug,daemon.log}`

可以删除这些文件（以及对应的已被归档的旧文件）。原先写入这些文件的内容还会出现在 `/var/log/syslog`。

5.1.10 slapd 的升级可能需要人工干预

OpenLDAP 2.5 is a major new release and includes several incompatible changes as described in [the upstream release announcement](#). Depending on the configuration, the `slapd` service might remain stopped after the upgrade, until necessary configuration updates are completed.

以下是一些已知的不兼容的变更：

- `slapd-bdb(5)` 和 `slapd-hdb(5)` 数据库后端已经被移除。如果您在 `bookworm` 中使用这些后端中的一个，强烈建议您在升级到 `trixie` 之前迁移到 `slapd-mdb(5)` 后端。
- `slapd-shell(5)` 数据库后端已经被移除。
- `slapo-ppolicy(5)` 叠加层将它的模式编译进了模块。如果存在旧的外部模式，它会与新的内置模式冲突。
- `slapd-pw-argon2.5` 这个由社区贡献的密码模块已改名为 `argon2`。

完成升级和恢复 `slapd` 服务所需的步骤可以在 `/usr/share/doc/slapd/README.Debian.gz`。You should also consult [the upstream upgrade notes](#) 中找到。您也可以参考上游提供的升级注意事项。

5.1.11 GRUB 不再默认运行 os-prober

很长一段时间以来，**grub** 都在使用 **os-prober** 软件包检测计算机上安装的其他操作系统，并将它们加入引导菜单中。不幸的是，某些情况下这可能会引起问题（例如虚拟机的客户机正在运行的时候），所以上游最新版本默认禁用了此行为。

如果您使用 **GRUB** 引导您的系统，并且希望在引导菜单中继续显示其他操作系统，您可以更改此行为。您可以编辑 `/etc/default/grub` 文件，确保设置了 `GRUB_DISABLE_OS_PROBER=false`，然后重新运行 `update-grub`。或者，您也可以执行

```
# dpkg-reconfigure <GRUB_PACKAGE>
```

来更改此行为。这条命令可以以用户友好的方式修改多种 **GRUB** 设置。

5.1.12 GNOME 对屏幕阅读器的支持有所下降

许多 **GNOME** 应用从 **GTK3** 切换到了 **GTK4**。令人难过的是，这使得这些应用与屏幕阅读器（例如 **orca**）之间的兼容性下降了很多。

如果您依赖屏幕阅读器，您应当考虑切换到另一个桌面，例如 **Mate**，它有更好的辅助功能支持。您可以安装 **mate-desktop-environment** 软件包进行切换。关于如何在 **Mate** 中使用 **Orca** 的信息可以在 [这里](#) 找到。

5.1.13 Baseline for 32-bit PC is now i686

Debian's support for 32-bit PC (known as the Debian architecture i386) now no longer covers any i586 processor. The new minimum requirement is i686. This means that the i386 architecture now requires the "long NOP" (NOPL) instruction, while bullseye still supported some i586 processors without that instruction (e.g. the "AMD Geode").

如果您的机器不满足该要求，建议您在 **bullseye** 剩余的支持周期内停留在该版本。

5.1.14 polkit 配置的变更

为了与上游和其他发行版保持一致，**polkit**（以前叫做 `PolicyKit`）服务修改了本地策略规则的语法和存储位置。**polkit** 允许无特权的程序访问有特权的系统服务。您现在应当使用 **JavaScript** 编写用来自定义安全策略的本地规则，然后将它们放在 `/etc/polkit-1/rules.d/*.rules`。以新格式编写的示例规则可以在 `/usr/share/doc/polkitd/examples/` 找到。更多信息可以在 [polkit.8.html#AUTHORIZATION_RULES](#) 中找到。

以前的情况是，可以用 **pkla** 格式编写规则，然后将它们放在 `/etc/polkit-1/localauthority` 或 `/var/lib/polkit-1/localauthority` 的子目录中。但是，**.pkla** 文件应当被视为是已废弃的，并且仅在安装了 **polkitd-pkla** 软件包的情况下才会继续工作。通常情况下，在您升级到 **bookworm** 时，会自动安装该软件包，但是未来的 **Debian** 发布版本可能不会包含该软件包，因此所有覆盖默认策略的本地策略未来都需要迁移到 **JavaScript** 格式。

5.1.15 ”合并的 /usr” 现在是必需的

Debian 已经批准了一个名为”合并的 /usr”的文件系统布局，它不再包含遗留的 /bin、/sbin、/lib 目录，也不包含其他可选的变种，例如 /lib64。在新的布局中，遗留的目录被指向对应位置 (/usr/bin、/usr/sbin、/usr/lib 和 /usr/lib64) 的符号链接替代。这意味着，比如说，/bin/bash 和 /usr/bin/bash 都会启动 bash。

对于安装时为 buster 或 bullseye 的系统而言，不会有任何变化，因为在这两个版本中，默认已经使用新的文件系统布局了。但是，旧的布局已经不再受支持，并且使用旧布局的系统在升级到 bookworm 时会转换到新布局。

转换到新布局的过程对大多数用户没有影响。所有文件都会被自动移动到新的位置，包括本地安装的文件和来自并非由 Debian 提供的软件包的文件，并且硬编码的路径，例如 /bin/sh，仍然会继续工作。但是，也有一些潜在的问题：

```
# dpkg --search
```

对于被移动到新位置的文件会给出错误的结果：

```
# dpkg --search /usr/bin/bash
```

不能识别出 bash 来自一个软件包。（但是

```
# dpkg --search /bin/bash
```

仍然会和预期一样正常工作。）

- 并非由 Debian 提供的本地软件可能不支持新布局，并且，比如说，可能会依赖于 /usr/bin/name 和 /bin/name 是两个不同的文件。在合并的系统中（包括从 buster 开始新安装的系统），这是不受支持的，所以在升级之前，任何此类软件都必须被修复或删除。
- 依赖于一个不可直接写入的”基本层”的系统（例如 WSL1 映像，或者使用多层 overlayfs 文件系统的容器系统）不能被安全地转换，所以要么需要被替换（例如从商店中安装一个新的 WSL1 映像），要么需要升级每个层（例如独立更新 overlayfs 的 Debian 基本层），而不是使用 dist-upgrade。

要了解更多信息，请阅读 [The Case for the /usr merge](#) 一文，以及 [Debian 技术委员会决议](#)。

5.1.16 从 buster 升级（不受支持）会因为 libcrypt1 而失败

Debian officially supports upgrades only from one stable release to the next, e.g. from bullseye to bookworm. Upgrades from buster to bookworm are not supported, and will fail due to [Bug #993755](#) with the following error:

```
Setting up libc6:ARCH (2.36-9) ...
/usr/bin/perl: error while loading shared libraries: libcrypt.so.1: cannot open_
↳shared object file: No such file or directory
dpkg: error processing package libc6:ARCH (--configure):
installed libc6:ARCH package post-installation script subprocess returned error exit_
↳status 127
```

但是，可能可以通过强制安装新的 **libcrypt1** 软件包来手动从该状况中恢复：

```
# cd $(mktemp -d)
# apt download libcrypt1
# dpkg-deb -x libcrypt1_*.deb .
# cp -ra lib/* /lib/
# apt --fix-broken install
```

5.1.17 openssh-server no longer reads ~/.pam_environment

The Secure Shell (SSH) daemon provided in the **openssh-server** package, which allows logins from remote systems, no longer reads the user's `~/.pam_environment` file by default; this feature has a [history of security problems](#) and has been deprecated in current versions of the Pluggable Authentication Modules (PAM) library. If you used this feature, you should switch from setting variables in `~/.pam_environment` to setting them in your shell initialization files (e.g. `~/.bash_profile` or `~/.bashrc`) or some other similar mechanism instead.

Existing SSH connections will not be affected, but new connections may behave differently after the upgrade. If you are upgrading remotely, it is normally a good idea to ensure that you have some other way to log into the system before starting the upgrade; see [准备故障恢复](#).

5.1.18 OpenSSH no longer supports DSA keys

Digital Signature Algorithm (DSA) keys, as specified in the Secure Shell (SSH) protocol, are inherently weak: they are limited to 160-bit private keys and the SHA-1 digest. The SSH implementation provided by the **openssh-client** and **openssh-server** packages has disabled support for DSA keys by default since OpenSSH 7.0p1 in 2015, released with Debian 9 ("stretch"), although it could still be enabled using the `HostKeyAlgorithms` and `PubkeyAcceptedAlgorithms` configuration options for host and user keys respectively.

The only remaining uses of DSA at this point should be connecting to some very old devices. For all other purposes, the other key types supported by OpenSSH (RSA, ECDSA, and Ed25519) are superior.

As of OpenSSH 9.8p1 in trixie, DSA keys are no longer supported even with the above configuration options. If you have a device that you can only connect to using DSA, then you can use the `ssh1` command provided by the **openssh-client-ssh1** package to do so.

In the unlikely event that you are still using DSA keys to connect to a Debian server (if you are unsure, you can check by adding the `-v` option to the `ssh` command line you use to connect to that server and looking for the "Server accepts key:" line), then you must generate replacement keys before upgrading. For example, to generate a new Ed25519 key and enable logins to a server using it, run this on the client, replacing `username@server` with the appropriate user and host names:

```
$ ssh-keygen -t ed25519
$ ssh-copy-id username@server
```

5.2 升级后在重启前需要做的事

当 `apt full-upgrade` 完成时，"形式上的"升级就完成了。对于向 trixie 的升级而言，重启前没有什么特别的操作需要完成。

5.2.1 升级过程之外的注意事项

5.2.2 安全支持上的局限性

有一些软件包，Debian 不能保证针对安全漏洞提供最小的向后移植。这些将在以下小节中介绍。

备注： **debian-security-support** 软件包可帮助跟踪已安装软件包的安全支持状态。

网页浏览器及其渲染引擎的安全支持状态

Debian 13 includes several browser engines which are affected by a steady stream of security vulnerabilities. The high rate of vulnerabilities and partial lack of upstream support in the form of long term branches make it very difficult to support these browsers and engines with backported security fixes. Additionally, library interdependencies make it extremely difficult to update to newer upstream releases. Applications using the **webkit2gtk** source package (e.g. **epiphany**) are covered by security support, but applications using qtwebkit (source package **qtwebkit-opensource-src**) are not.

对于通用网页浏览器，我们推荐 Firefox 和 Chromium。这些软件将使用最新的 ESR 版本持续在 stable 中予以更新。这同样适用于 Thunderbird。

一旦一个发布版本成为 oldstable，官方支持的浏览器的支持周期可能短于发布版本的标准支持周期。例如，Chromium 在 oldstable 中只会获得 6 个月的安全支持，而不是通常的 12 个月。

基于 Go 和 Rust 的软件包

Debian 当前的基础架构在重新构建系统化使用静态链接的软件包时存在一些问题。随着 Go 和 Rust 生态系统的成长，这些软件包将只能得到有限的安全支持，直到基础架构得到改进使得这些软件包更加易于维护。

大多数情况下，如果 Go 或 Rust 开发库需要更新，这些更新只能通过定期的小版本更新提供。

5.2.3 Python 解释器标记为由外部管理

为了遵循 PEP-668，Debian 提供的 python3 解释器软件包 (**python3.11** 和 **pypy3**) 已被标记为由外部管理。Debian 提供的 **python3-pip** 也遵循此提案，并且会拒绝在 Debian 的 python 解释器中手动安装软件包，除非使用了 `--break-system-packages` 选项。

如果您想安装的 Python 应用程序（或者您想安装的版本）尚未被 Debian 打包，我们建议您使用 `pipx`（位于 Debian 软件包 **pipx**）进行安装。pipx 会创建一个与其他应用程序和系统 Python 模块相隔离的环境，并将该应用程序及其依赖关系安装至其中。`

If you need to install a Python library module (or version) that isn't packaged in Debian, we recommend installing it into a virtualenv, where possible. You can create virtualenvs with the `venv` Python stdlib module (in the **python3-venv** Debian package) or the `virtualenv` Python 3rd-party tool (in the **virtualenv** Debian package). For example, instead of running `pip install --user foo`, run: `mkdir -p ~/.venvs && python3 -m venv ~/.venvs/foo && ~/.venvs/foo/bin/python -m pip install foo` to install it in a dedicated virtualenv.

阅读 `/usr/share/doc/python3.11/README.venv` 以了解详细信息。

5.2.4 VLC 中视频编/解码的硬件加速支持受限

VLC 视频播放器通过 VA-API 和 VDPAU 支持视频编码和解码的硬件加速。但是，VLC 对 VA-API 的支持和 FFmpeg 的版本密切相关。因为 FFmpeg 升级到了 5.x 分支，VLC 的 VA-API 支持已被禁用。使用支持原生 VA-API 的 GPU（例如 Intel 和 AMD 的 GPU）的用户可能在播放和编码视频时发现 CPU 使用率较高。

使用提供原生 VDPAU 支持的 GPU（例如使用非自由驱动的 NVIDIA GPU）的用户不受此问题影响。

Support for VA-API and VDPAU can be checked with `vainfo` and `vdpauintfo` (each provided in a Debian package of the same name).

5.2.5 systemd-resolved 被分割成单独的软件包

新的 **systemd-resolved** 软件包在升级时不会被默认安装。如果您在使用 `systemd-resolved` 系统服务，请在升级完成后手动安装这一新软件包，并注意，在安装完该软件包之前，DNS 解析可能无法工作，因为系统中没有该服务。安装该软件包后，会把对 `/etc/resolv.conf` 的控制权自动交给 `systemd-resolved`。关于 `systemd-resolved` 的更多信息，请参考官方 [文档](#)。注意，`systemd-resolved` 以前不是，现在也不是 Debian 默认的 DNS 解析程序。如果您未配置您的机器使用 `systemd-resolved` 作为 DNS 解析程序，则不必进行任何操作。

5.2.6 systemd-boot 被分割成单独的软件包

新的 **systemd-boot** 软件包在升级时不会被默认安装。如果您在使用 `systemd-boot`，请手动安装这一新软件包，并注意，在安装完该软件包之前，系统将使用旧版本的 `systemd-boot` 作为引导加载器。安装该软件包后，`systemd-boot` 会被自动配置为该机器的引导加载器。Debian 默认的引导加载器仍然是 GRUB。如果您未配置您的机器使用 `systemd-boot` 作为引导加载器，则不必进行任何操作。

5.2.7 systemd-journal-remote 不再使用 GnuTLS

可选的 `systemd-journal-gatewayd` 和 `systemd-journal-remote` 服务现在在构建时不再包含 GnuTLS 支持，这意味着两个程序都不再提供 `--trust` 选项，如果指定了该选项，会产生错误。

5.2.8 bookworm 中的 adduser 的重大变更

adduser 有多项变更。最明显的变化是 `--disabled-password` 和 `--disabled-login` 现在是等价的。请阅读 `/usr/share/doc/adduser/NEWS.Debian.gz` 了解更多细节。

5.2.9 Xen 网络接口的可预测命名

systemd 中网络接口的可预测命名逻辑已得到扩充，现在它可以根据 Xen netfront 设备信息生成稳定的名称。这意味着接口现在具有 `enX#` 形式的稳定名称，而不再使用以前由内核分配的名称系统。请在升级之后、重新启动之前调整您的系统。您可以在 [NetworkInterfaceNames](#) 维基页面上找到更多信息。

5.2.10 dash 对 “^” 的处理方式的变更

在 Debian 中默认提供系统 shell `/bin/sh` 的 `dash` 现在将 `^` 视为字面意义上的字符，这也是一直以来符合 POSIX 标准的行为。这意味着在 `bookworm` 中 `[^0-9]` 的意思不再是“除了 0 到 9”，而是“0 到 9 和 ^”。

5.2.11 netcat-openbsd 支持抽象套接字

The `netcat` utility for reading and writing data across network connections supports abstract sockets (see [unix.7.html#Abstract_sockets](#), and uses them by default in some circumstances.

默认情况下，`netcat` 由 **netcat-traditional** 提供。但是，如果 `netcat` 由 **netcat-openbsd** 软件包提供，而您使用 `AF_UNIX` 套接字，则这一默认设置将生效。这种情况下，`nc` 命令的 `-U` 选项将会把以 `@` 开头的参数解释为请求一个抽象套接字，而不是当前目录下以 `@` 开头的文件。这可能会产生安全问题，因为文件权限不能用于控制对抽象套接字的访问。您可以通过在文件名之前加 `./` 或指定绝对路径的方式继续使用以 `@` 开头的文件名。

5.3 过时与废弃内容

5.3.1 值得注意的过时软件包

以下是已知的和值得注意的过时软件包的列表（有关过时软件包的描述，请参阅[过时的软件包](#)）。

过时的软件包包括：

- **libnss-ldap** 软件包已从 trixie 移除。它的功能已由 **libnss-ldapd** 和 **libnss-sss** 代替。
- **libpam-ldap** 软件包已从 trixie 移除。它已被 **libpam-ldapd** 取代。
- **fdflush** 软件包已从 trixie 移除。请使用 **util-linux** 中的 `blockdev --flushbufs` 命令代替它。
- **libgdal-perl** 软件包已从 trixie 移除，因为上游已经不再支持 GDAL 的 Perl 绑定。如果您需要 GDAL 的 Perl 支持，您可以迁移到由 **Geo::GDAL::FFI** 软件包提供的 FFI 接口，在 CPAN 中可以找到。您需要按照 [BookwormGdalPerl](#) 维基页面 中标明的步骤编译您自己的二进制文件。

5.3.2 trixie 的废弃组件

随着下一个版本 Debian 14（代号为 forky）的发布，某些功能将被弃用。用户需要迁移到其他替代方案，以防止在更新到 Debian 14 时出现问题。

这包括以下特性：

- NSS 服务 `gw_name` 的开发已于 2015 年停止。与之对应的软件包 **libnss-gw-name** 可能在未来的 Debian 发布版本中被移除。上游开发者建议使用 **libnss-myhostname** 代替它。
- **dmraid** 上游自 2010 年年底之后就没有任何活动，在 Debian 中的软件包也处于苟延残喘的状态。bookworm 将会是包含该软件包的最后一个发布版本，所以如果您正在使用 **dmraid**，请做好相应的计划。
- **request-tracker4** 在此发布版本中已被 **request-tracker5** 取代，并将在未来的发布版本中被移除。我们建议您在此发布版本的生命周期内计划从 **request-tracker4** 迁移到 **request-tracker5**。
- The **isc-dhcp** suite has been deprecated by the ISC. The [Debian Wiki](#) has a list of alternative implementations, see the [DHCP Client](#) and [DHCP Server](#) pages for the latest. If you are using NetworkManager or systemd-networkd, you can safely remove the **isc-dhcp-client** package as they both ship their own implementation. If you are using the **ifupdown** package, you can experiment with `udhcpd` as a replacement. The ISC recommends the **Kea** package as a replacement for DHCP servers.

The security team will support the **isc-dhcp** package during the bookworm lifetime, but the package will likely be unsupported in the next stable release, see [bug #1035972 \(isc-dhcp EOL'ed\)](#) for more details.

5.4 已知的严重缺陷

Although Debian releases when it's ready, that unfortunately doesn't mean there are no known bugs. As part of the release process all the bugs of severity serious or higher are actively tracked by the Release Team, so an [overview of those bugs](#) that were tagged to be ignored in the last part of releasing trixie can be found in the [Debian Bug Tracking System](#). The following bugs were affecting trixie at the time of the release and worth mentioning in this document:

缺陷编号	软件包 (源码包或二进制包)	描述
1032240	akonadi-backend-mysql	akonadi server fails to start since it cannot connect to mysql database
1032177	faketime	faketime doesn't fake time (on i386)
918984	src:fuse3	provide upgrade path fuse -> fuse3 for bookworm
1016903	g++-12	tree-vectorize: Wrong code at O2 level (-fno-tree-vectorize is working)
1020284	git-daemon-run	fails to purge: deluser -f: Unknown option: f
919296	git-daemon-run	fails with 'warning: git-daemon: unable to open supervise/ok: file does not exist'
1034752	src:gluegen2	embeds non-free headers

关于 Debian 的更多信息

6.1 扩展阅读

Beyond these release notes and the installation guide (at <https://www.debian.org/releases/trixie/installmanual>) further documentation on Debian is available from the Debian Documentation Project (DDP), whose goal is to create high-quality documentation for Debian users and developers, such as the Debian Reference, Debian New Maintainers Guide, the Debian FAQ, and many more. For full details of the existing resources see the [Debian Documentation website](#) and the [Debian Wiki](#).

各个软件包的文档被安装到 `/usr/share/doc/软件包`。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，但您只应该在研究并查阅文档了解问题大致情况后考虑寻求帮助。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

The mailing lists of most interest to Debian users are the `debian-user` list (English) and other `debian-user-language` lists (for other languages). For information on these lists and details of how to subscribe see <https://lists.debian.org/>. Please check the archives for answers to your question prior to posting and also adhere to standard list etiquette.

6.2.2 IRC

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](#) 中找到该准则。

欲了解更多关于 OFTC 的信息，请访问他们的 [网站](#)。

6.3 报告 Bug

我们努力让 Debian 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错误。为了和 Debian 一贯的“开放的开发”原则相呼应，也为了能为我们的用户提供更好的服务，我们在错误追踪系统（BTS）提供全部已经报告的错误的所有信息。您可以通过 <https://bugs.debian.org/> 来访问 BTS。

如果您在本发行版或者软件包中发现了错误，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错误。有这个限制是因为这样我们才能够追踪错误，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` 来提交一个错误报告，或者亲自动手发送电子邮件。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 [错误追踪系统](#)，以获得更多关于错误追踪系统的信息和用法。

6.4 为 Debian 做贡献

You do not need to be an expert to contribute to Debian. By assisting users with problems on the various user support [lists](#) you are contributing to the community. Identifying (and also solving) problems related to the development of the distribution by participating on the development [lists](#) is also extremely helpful. To maintain Debian's high-quality distribution, [submit bugs](#) and help developers track them down and fix them. The tool `how-can-i-help` helps you to find suitable reported bugs to work on. If you have a way with words then you may want to contribute more actively by helping to write [documentation](#) or [translating](#) existing documentation into your own language.

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）](#) 中获得相关的信息。如果您对特定的用户群体感兴趣，那么您可能会很乐意参加 Debian 的某些子项目，包括向特定处理器架构的移植，以及为特定用户群体准备的 [Debian Pure Blends](#)，等等。

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。贡献本身就是一件非常有益而有趣的事情，它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。

在升级前管理您的 bookworm 系统

本附录包含在升级到 trixie 之前，如何确保您能够从 bookworm 安装或升级软件包的相关信息。

7.1 升级您的 bookworm 系统

Basically this is no different from any other upgrade of bookworm you've been doing. The only difference is that you first need to make sure your package list still contains references to bookworm as explained in *Checking your APT source-list files*.

如果您从 Debian 镜像升级，将自动升级到最新的 bookworm 的小版本更新。

7.2 检查您的 APT source-list 文件

如果您的 APT source-list 文件（参见 `sources.list(5)`）的任意一行包含了“stable”这样的字符串，它在新版本发布后实际上已经指向了 trixie。如果您还没准备好升级系统的话，这可能不是您想要的结果。但如果您已经运行过了 `apt update` 命令的话，您仍然可以采取下述步骤退回原状态而不产生额外问题。

如果您已经从 trixie 安装了软件包，就没有太大的必要再从 bookworm 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。降级软件包是可能的，但是不属于本文的讨论范围。

As root, open the relevant APT source-list file (such as `/etc/apt/sources.list`) with your favorite editor, and check all lines beginning with

- `deb http:`
- `deb https:`
- `deb tor+http:`
- `deb tor+https:`
- `URIs: http:`
- `URIs: https:`

- URIs: tor+http:
- URIs: tor+https:

for a reference to "stable". If you find any, change "stable" to "bookworm".

如果有些行以 `deb file:` 或 `URIs: file:` 开始, 您应该自己检查该位置是否包含有 `bookworm` 或者 `trixie` 的档案仓库。

重要: 不要修改任何以 `deb cdrom:` 或 `URIs: 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` 命令以启用从光盘更新的功能。即使在 `cdrom:` 的行发现 "unstable" 这样的字符串也不要感到奇怪。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 update
```

以刷新软件包列表。

7.3 Performing the upgrade to latest bookworm release

To upgrade all packages to the state of the latest point release for bookworm, do

```
# apt full-upgrade
```

7.4 删除过时的配置文件

在将系统升级到 `trixie` 之前, 建议从系统中删除旧的配置文件 (例如 `/etc` 下的 `*.dpkg-{new,old}` 文件)。

发行说明的贡献者

有许多人对发行说明提供了帮助，包括但不限于

- ADAM D. BARRAT (2013 年的多项修复),
- ADAM DI CARLO (先前的版本),
- ANDREAS BARTH ABA (先前的版本: 2005 - 2007),
- ANDREI POPESCU (大量贡献),
- ANNE BEZEMER (先前的版本),
- BOB HILLIARD (之前的发行),
- CHARLES PLESSY (GM965 问题描述),
- CHRISTIAN PERRIER BUBULLE (Lenny 安装程序),
- CHRISTOPH BERG (PostgreSQL 相关的问题),
- DANIEL BAUMANN (Debian Live),
- DAVID PRÉVOT TAFFIT (Wheezy 版本),
- EDDY PETRIȘOR (大量贡献),
- EMMANUEL KASPER (回迁软件包),
- ESKO ARAJÄRVI (重新实现 X11 升级),
- FRANS POP FJP (先前版本 Etch),
- GIOVANNI RAPAGNANI (无数贡献),
- GORDON FARQUHARSON (ARM 移植的问题),
- HIDEKI YAMANE HENRICH (自 2006 年起),
- HOLGER WANSING HOLGERW (自 2009 年起),
- JAVIER FERNÁNDEZ-SANGUINO PEÑA JFS (Etch 版本, Squeeze 版本),
- JENS SEIDEL (德语翻译者, 无数贡献),

- JONAS MEURER (syslog 问题),
- JONATHAN NIEDER (Squeeze 版本, Wheezy 版本),
- JOOST VAN BAAL-ILIĆ JOOSTVB (Wheezy 版本, Jessie 版本),
- JOSIP RODIN (先前的版本),
- JULIEN CRISTAU JCRISTAU (Squeeze 版本, Wheezy 版本),
- JUSTIN B RYE (英语修改),
- LAMONT JONES (NFS 问题的描述),
- LUK CLAES (编辑动员管理员),
- MARTIN MICHLMAYR (ARM 移植的问题),
- MICHAEL BIEBL (syslog 问题),
- MORITZ MÜHLENHOFF (大量贡献),
- NIELS THYKIER NTHYKIER (Jessie 版本),
- NOAH MEYERHANS (无数贡献),
- NORITADA KOBAYASHI (日语翻译 (协调), 无数贡献),
- OSAMU AOKI (大量贡献),
- PAUL GEVERS ELBRUS (Buster 版本),
- PETER GREEN (内核版本注记),
- ROB BRADFORD (Etch 版本),
- SAMUEL THIBAUT (d-i 布莱叶支持描述),
- SIMON BIENLEIN (d-i 布莱叶支持描述),
- SIMON PAILLARD SPAILLAR-GUEST (无数贡献),
- STEFAN FRITSCH (Apache 问题描述),
- STEVE LANGASEK (Etch 版本),
- STEVE MCINTYRE (Debian CD),
- TOBIAS SCHERER ("proposed-update" 描述),
- VICTORY VICTORY-GUEST (标记语言修复, 自 2006 年起),
- VINCENT MCINTYRE ("proposed-update" 描述),
- W. MARTIN BORGERT (编辑 Lenny 发行, 转换到 DocBook XML).

本文档被翻译为多种语言。感谢这些翻译者们!