Notes de llançament per Debian 12 (bookworm), 64-bit MIPS (little endian)

El projecte de documentació de Debian (https://www.debian.org/doc/)

2 de gener de 2023
Aquest document és programari lliure, el podeu redistributeix i/o modificar sota els termes de la Llicència Pública General de GNU versió 2 de la llicència, tal com la publica la Free Software Foundation.

Aquest programa es distribueix amb l'esperança que serà útil, però sense cap garantia, fins i tot sense la garantia implícita de COMERCIALITZACIÓ o IDONEITAT PER A UN PROPÒSIT PARTICULAR. Consulteu la llicència general pública de GNU per a més detalls.

Haurièu d'haver rebut una còpia de la llicència pública general GNU amb este programa; si no, escriviu a la Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Podeu també trobar el text amb la llicència a [https://www.gnu.org/licenses/gpl-2.0.html](https://www.gnu.org/licenses/gpl-2.0.html) i a /usr/share/common-licenses/GPL-2 en els sistemes Debian.
Índex

1 Introducció
1.1 Informar d’errors d’este document ................................. 1
1.2 Col·laborar amb informes d’actualització ............................ 1
1.3 Font d’este document .............................................. 2

2 Què hi ha de nou a Debian 12
2.1 Arquitectures suportades ........................................... 3
2.2 Què hi ha de nou a la distribució? .................................. 3
2.2.1 Desktops and well known packages ................................ 3
2.2.2 Something ..................................................... 4

3 El sistema d’instal·lació
3.1 Què hi ha de nou al sistema d’instal·lació? ......................... 5
3.1.1 Something ..................................................... 5
3.1.2 Instal·lació automatitzada ...................................... 5
3.2 Container and Virtual Machine images ............................. 5

4 Upgrades from Debian 11 (bullseye)
4.1 Preparació de l’actualització ....................................... 7
4.1.1 Back up any data or configuration information .................. 7
4.1.2 Inform users in advance ........................................ 7
4.1.3 Prepare for downtime on services ............................... 7
4.1.4 Prepare for recovery ........................................... 8
4.1.4.1 Debug shell during boot using initrd ......................... 8
4.1.4.2 Debug shell during boot using systemd ....................... 8
4.1.5 Prepare a safe environment for the upgrade .................... 9

4.2 Start from “pure” Debian ............................................. 9
4.2.1 Upgrade to Debian 11 (bullseye) ................................ 9
4.2.2 Debian Backports .............................................. 9
4.2.3 Remove non-Debian packages ................................... 10
4.2.4 Upgrade to latest point release ................................ 10
4.2.5 Prepare the package database .................................. 10
4.2.6 Remove obsolete packages ..................................... 10
4.2.7 Clean up leftover configuration files ............................ 10
4.2.8 The proposed-updates section ................................ 10
4.2.9 Unofficial sources ............................................ 11
4.2.10 Disabling APT pinning ........................................ 11
4.2.11 Check package status ......................................... 11

4.3 Preparing APT source-list files .................................... 12
4.3.1 Adding APT Internet sources .................................... 12
4.3.2 Adding APT sources for a local mirror ......................... 12
4.3.3 Adding APT sources from optical media ....................... 13

4.4 Upgrading packages .................................................. 13
4.4.1 Recording the session .......................................... 14
4.4.2 Updating the package list ...................................... 14
4.4.3 Make sure you have sufficient space for the upgrade ........ 14
4.4.4 Minimal system upgrade ...................................... 16
4.4.5 Upgrading the system .......................................... 16

4.5 Possible issues during upgrade ..................................... 17
4.5.1 Dist-upgrade fails with “Could not perform immediate configuration” .......................................................... 17
4.5.2 Expected removals .............................................. 17
4.5.3 Conflicts or Pre-Depends loops ................................ 17
4.5.4 File conflicts .................................................. 18
4.5.5 Configuration changes ......................................... 18
Capítol 1

Introducció

Este document informa als usuaris de la distribució Debian dels principals canvis a la versió 12 (nom en clau bookworm).

Les notes de llançament ofereixen informació sobre com actualitzar de forma segura des de la versió 11 (nom en clau bullseye) a la versió actual, i informen als usuaris de problemes potencials coneguts que es poden trobar durant eixe procés.

You can get the most recent version of this document from https://www.debian.org/releases/bookworm/releasenotes.

ATENCIÓ

Tingueu en compte que és impossible fer una llista amb tots els problemes coneguts i per tant ha calgut fer una selecció basada en una combinació de l'abast i l'impacte dels problemes.

Tingueu en compte també, que tan sols es suporta i documenta l’actualització des de la versió anterior de Debian (en este cas, l’actualització des de 11). Si necessiteu actualitzar des de versions més antigues, us suggerim que llegiu primer les edicions anteriors de les notes de llançament i actualitzeu primer a 11.

1.1 Informar d’errors d’este document

S’ha intentat fer comprovacions dels diversos passos del procés d’actualització que es descriuen en este document, intentant també anticipar possibles problemes que els nostres usuaris puguin trobar.

No obstant això, si creieu que heu trobat un error en aquesta documentació (informació incorrecta o manca d’informació), envieu un informe d’error al sistema de seguiment d’errors (https://bugs.debian.org/), al paquet release-notes. Reviseu primer si hi ha cap informe d’error existent (https://bugs.debian.org/release-notes) en cas de que ja s’haja informat del problema. No dubteu a afegir qualsevol informació addicional als informes d’error ja existents si creieu que podeu contribuir contingut a aquest document.

Apreciam, i animem, a proporcionar informes amb fitxers de diferències de les fonts del document. Trobareu més informació que descriu com obtenir les fonts d’aquest document en Secció 1.3.

1.2 Col·laborar amb informes d’actualització

Qualsevol informació dels usuaris relacionada amb actualitzacions des de bullseye a bookworm és ben-vinguda. Si voleu compartir alguna informació, envieu un informe d’error amb els vostres resultats al sistema de seguiment d’errors (https://bugs.debian.org/), al paquet upgrade-reports. Us demanem que comprimiu qualsevol adjunt que envieu (fent ús del gzip).

Quan envieu l’informe d’actualització, assegureu-vos d’incloure la informació següent:
• L'estat de la vostra base de dades de paquets abans de l'actualització: La base de dades d'estat de dpkg que trobareu a /var/lib/dpkg/status i la informació d'estat d' apt que trobareu a /var/lib/apt/extended_states. Hauríeu d'haver fet un còpia de seguretat abans d'actualitzar com es descriu a Secció 4.1.1, però també podeu trobar còpies de seguretat de /var/lib/dpkg/status a /var/backups.

• Els registres de sessió creats amb script, tal i com es descriu a Secció 4.4.1.

• Els registres d'apt els podeu trobar a /var/log/apt/term.log, i els registres d'aptitude els trobareu a /var/log/aptitude.

NOTA
Hauríeu de dedicar un cert temps a revisar i esborrar qualsevol informació sensible i/o confidencial als fitxers de registre abans d'incloure'ls a l'informe d'error, ja que la informació es publicarà a una base de dades pública.

1.3 Font d'este document
Capítol 2

Què hi ha de nou a Debian 12

Al Viqui (https://wiki.debian.org/NewInBookworm) trobareu més informació d’este tema.

2.1 Arquitectures suportades

Les arquitectures suportades oficialment a Debian bookworm són les següents:

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

Podeu llegir més de l’estat dels ports, i informació específica del port de la vostra arquitectura a les pàgines web dels ports de Debian (https://www.debian.org/ports/).

2.2 Què hi ha de nou a la distribució?

| TODO: Make sure you update the numbers in the .ent file using the changes-release.pl script found under ../ |

Este nou llançament de Debian ve altra vegada amb molt més programari que el seu predecessor bullseye, la distribució inclou més de 11294 nous paquets, d’un total de 59551 paquets. S’ha actualitzat la major part del programari de la distribució: més de 42821 paquets de programes (açò suposa el 72% de tots els paquets de bullseye). També, un nombre significatiu de paquets (uns 9519, un 16% dels paquets de bullseye) s’han esborrat de la distribució per motius diversos. No voreu cap actualització per aquests paquets i es marcaran com a «obsolete» als entorns de gestió de paquets.

2.2.1 Desktops and well known packages

Debian again ships with several desktop applications and environments. Among others it now includes the desktop environments GNOME 3.38, KDE Plasma 5.20, LXDE 11, LXQt 0.16, MATE 1.24, and Xfce 4.16.

Les aplicacions de productivitat també han sigut actualitzades, incloent la suite d’oficina:

- LibreOffice is upgraded to version 7.0;
Calligra is upgraded to 3.2.

GNUcash is upgraded to 4.4;

Entre altres, este llançament també inclou les actualitzacions de programari següents:

<table>
<thead>
<tr>
<th>Paquet</th>
<th>Versió a 11 (bullseye)</th>
<th>Versió a 12 (bookworm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache</td>
<td>2.4.38</td>
<td>2.4.48</td>
</tr>
<tr>
<td>BIND servidor de DNS</td>
<td>9.11</td>
<td>9.16</td>
</tr>
<tr>
<td>Cryptsetup</td>
<td>2.1</td>
<td>2.3</td>
</tr>
<tr>
<td>Dovecot MTA</td>
<td>2.3.4</td>
<td>2.3.13</td>
</tr>
<tr>
<td>Emacs</td>
<td>26.1</td>
<td>27.1</td>
</tr>
<tr>
<td>Exim servidor de correu per defecte</td>
<td>4.92</td>
<td>4.94</td>
</tr>
<tr>
<td>GNU Compiler Collection com a compilador per defecte</td>
<td>8.3</td>
<td>10.2</td>
</tr>
<tr>
<td>GIMP</td>
<td>2.10.8</td>
<td>2.10.22</td>
</tr>
<tr>
<td>GnuPG</td>
<td>2.2.12</td>
<td>2.2.27</td>
</tr>
<tr>
<td>Inkscape</td>
<td>0.92.4</td>
<td>1.0.2</td>
</tr>
<tr>
<td>la biblioteca C de GNU</td>
<td>2.28</td>
<td>2.31</td>
</tr>
<tr>
<td>lighttpd</td>
<td>1.4.53</td>
<td>1.4.59</td>
</tr>
<tr>
<td>Imatge del nucli Linux</td>
<td>sèrie 4.19</td>
<td>5.10 series</td>
</tr>
<tr>
<td>LLVM/Clang toolchain</td>
<td>6.0.1 i 7.0.1 (per defecte)</td>
<td>9.0.1 and 11.0.1 (default)</td>
</tr>
<tr>
<td>MariaDB</td>
<td>10.3</td>
<td>10.5</td>
</tr>
<tr>
<td>Nginx</td>
<td>1.14</td>
<td>1.18</td>
</tr>
<tr>
<td>OpenJDK</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>OpenSSH</td>
<td>7.4p1</td>
<td>8.4p1</td>
</tr>
<tr>
<td>Perl</td>
<td>5.28</td>
<td>5.32</td>
</tr>
<tr>
<td>PHP</td>
<td>7.3</td>
<td>7.4</td>
</tr>
<tr>
<td>Postfix MTA</td>
<td>3.4</td>
<td>3.5</td>
</tr>
<tr>
<td>PostgreSQL</td>
<td>11</td>
<td>13</td>
</tr>
<tr>
<td>Python 3</td>
<td>3.7.3</td>
<td>3.9.1</td>
</tr>
<tr>
<td>Rustc</td>
<td>1.41 (1.34 for armel)</td>
<td>1.48</td>
</tr>
<tr>
<td>Samba</td>
<td>4.9</td>
<td>4.13</td>
</tr>
<tr>
<td>Vim</td>
<td>8.1</td>
<td>8.2</td>
</tr>
</tbody>
</table>

2.2.2 Something

Text
Capítol 3

El sistema d’instal·lació

El sistema d’instal·lació oficial a Debian és el Debian Installer. Ens proporciona diversos mètodes d’instal·lació. Els mètodes dels que disposeu per instal·lar el vostre sistema dependran de la vostra arquitectura.

Les imatges de l’instal·lador de bookworm es poden trobar junt a la guia d’instal·lació a la web de Debian (https://www.debian.org/releases/bookworm/debian-installer/).

La guia d’instal·lació també està inclosa al primer suport dels conjunts de DVD de Debian (CD/blu-ray), a:
/doc/install/manual/ca/index.html

Pot ser també voldreu comprovar les errates (https://www.debian.org/releases/bookworm/debian-installer/index#errata) del debian-installer per obtenir la llista de problemes coneguts.

3.1 Què hi ha de nou al sistema d’instal·lació?

S’ha fet molt desenvolupament a l’instal·lador de Debian des de el seu anterior llançament oficial a Debian 11 amb el resultat que s’han aconseguit millores tant al suport de maquinari com noves i excitants característiques.

Si teniu interès en un visió general dels canvis detallats produïts des de bullseye, comproveu els anuncis de llançament de la beta de bookworm i els llançaments RC que trobareu a l’historial de notícies (https://www.debian.org/devel/debian-installer/News/) de l’instal·lador de Debian.

3.1.1 Something

3.1.2 Instal·lació automatitzada

Some changes also imply changes in the support in the installer for automated installation using pre-configuration files. This means that if you have existing preconfiguration files that worked with the bullseye installer, you cannot expect these to work with the new installer without modification.

La guia d’instal·lació (https://www.debian.org/releases/bookworm/installmanual) té un apèndix separat amb una extensa documentació de com utilitzar la configuració prèvia.

3.2 Container and Virtual Machine images

Multi-architecture Debian bookworm container images are available on Docker Hub (https://hub.docker.com/_/debian). In addition to the standard images, a “slim” variant is available that reduces disk usage.

Virtual machine images for the Hashicorp Vagrant VM manager are published to Vagrant Cloud (https://app.vagrantup.com/debian).
Capítol 4

Upgrades from Debian 11 (bullseye)

4.1 Preparació de l’actualització

We suggest that before upgrading you also read the information in Capítol 5. That chapter covers potential issues which are not directly related to the upgrade process but could still be important to know about before you begin.

4.1.1 Back up any data or configuration information

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.

The upgrade has a few preconditions; you should check them before actually executing the upgrade.

4.1.2 Inform users in advance

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 bookworm, so a reboot will be necessary. Typically, this will be done after the upgrade is finished.

4.1.3 Prepare for downtime on services

There might be services that are offered by the system which are associated with packages that will be included in the upgrade. If this is the case, please note that, during the upgrade, these services will be stopped while their associated packages are being replaced and configured. During this time, these services will not be available.
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 unavailable\(^1\) for a significant period of time.

If the system being upgraded provides critical services for your users or the network\(^2\), you can reduce the downtime if you do a minimal system upgrade, as described in Secció 4.4.4, 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 Secció 4.4.5. 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 Prepare for recovery

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.

For this reason it makes sense to ensure that you will be able to recover if your system should fail to reboot or, for remotely managed systems, fail to bring up networking.

If you are upgrading remotely via an \textit{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 \textit{rescue mode} of the bookworm 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 \textit{Installation Guide} (https://www.debian.org/releases/bookworm/installmanual) and the Debian Installer FAQ (https://wiki.debian.org/DebianInstaller/FAQ).

If that fails, you will need an alternative way to boot your system so you can access and repair it. One option is to use a special rescue or \textit{live install} (https://www.debian.org/CD/live/) image. After booting from that, you should be able to mount your root file system and \textit{chroot} into it to investigate and fix the problem.

#### 4.1.4.1 Debug shell during boot using initrd

The \textit{initramfs-tools} package includes a debug shell\(^3\) 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 \texttt{/dev}; what modules are loaded (\texttt{cat /proc/modules}); output of \texttt{dmesg} for errors loading drivers. The output of \texttt{dmesg} will also show what device files have been assigned to which disks; you should check that against the output of \texttt{echo $ROOT} to make sure that the root file system is on the expected device.

If you do manage to fix the problem, typing \texttt{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.

#### 4.1.4.2 Debug shell during boot using systemd

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 \texttt{systemd.unit=rescue.target} to the kernel parameters.

\(^1\)If the \texttt{debconf} priority is set to a very high level you might prevent configuration prompts, but services that rely on default answers that are not applicable to your system will fail to start.

\(^2\)For example: DNS or DHCP services, especially when there is no redundancy or failover. In the DHCP case end-users might be disconnected from the network if the lease time is lower than the time it takes for the upgrade process to complete.

\(^3\)This feature can be disabled by adding the parameter \texttt{panic=0} to your boot parameters.
CAPÍTOL 4. UPGRADES FROM DEBIAN 11 … 4.2.  START FROM “PURE” DEBIAN

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:

```bash
# 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 (https://freedesktop.org/wiki/Software/systemd/Debugging/) article.

4.1.5  Prepare a safe environment for the upgrade

```markdown
**IMPORTANT**

If you are using some VPN services (such as tinc) consider that they might not be available throughout the upgrade process. Please see Secció 4.1.3.
```

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.

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 bullseye, 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)` (https://manpages.debian.org//bookworm/apt/sources.list.5.html). If your system is using multiple source-list files then you will need to ensure they stay consistent.

4.2.1  Upgrade to Debian 11 (bullseye)

Only upgrades from Debian 11 (bullseye) are supported. Display your Debian version with:

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

Please follow the instructions in the Release Notes for Debian 11 (https://www.debian.org/releases/bullseye/rela#enotes) to upgrade to Debian 11 first if needed.

4.2.2  Debian Backports

Debian Backports (https://backports.debian.org/) 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 bullseye-backports have version numbers lower than the version in bookworm, so they should upgrade normally to bookworm in the same way as “pure” bullseye 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.
4.2.3 Remove non-Debian packages

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

\[ \text{\$ aptitude search '?narrow(?installed, ?not(?origin(Debian)))'} \]
\[ \text{\$ apt-forktracer | sort} \]

4.2.4 Upgrade to latest point release

This procedure assumes your system has been updated to the latest point release of bullseye. If you have not done this or are unsure, follow the instructions in Secció A.1.

4.2.5 Prepare the package database

You should make sure the package database is ready before proceeding with the upgrade. If you are a user of another package manager like aptitude or synaptic, review any pending actions. A package scheduled for installation or removal might interfere with the upgrade procedure. Note that correcting this is only possible if your APT source-list files still point to bullseye and not to stable or bookworm; see Secció A.2.

4.2.6 Remove obsolete packages

It is a good idea to remove obsolete packages from your system before upgrading. They may introduce complications during the upgrade process, and can present security risks as they are no longer maintained.

4.2.7 Clean up leftover configuration files

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:

\[ \# \text{ find /etc -name '*.dpkg-*' -o -name '*.ufc-*' -o -name '*.merge-error'} \]

4.2.8 The proposed-updates section

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.
4.2.9 Unofficial sources

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 bookworm 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 bullseye system. Such packages are most likely to cause problems during an upgrade as they may result in file conflicts. Section 4.5 has some information on how to deal with file conflicts if they should occur.

4.2.10 Disabling APT pinning

If you have configured APT to install certain packages from a distribution other than stable (e.g. from testing), you may have to change your APT pinning configuration (stored in /etc/apt/preferences and /etc/apt/preferences.d/) to allow the upgrade of packages to the versions in the new stable release. Further information on APT pinning can be found in apt_preferences(5) (https://manpages.debian.org/bookworm/apt/apt_preferences.5.en.html).

4.2.11 Check package status

Regardless of the method used for upgrading, it is recommended that you check the status of all packages first, and verify that all packages are in an upgradable state. The following command will show any packages which have a status of Half-Installed or Failed-Config, and those with any error status.

```
# dpkg --audit
```

You could also inspect the state of all packages on your system using aptitude or with commands such as

```
# dpkg -l | pager

o

# dpkg --get-selections "*" > ~/pkts-actuals.txt
```

It is desirable to remove any holds before upgrading. If any package that is essential for the upgrade is on hold, the upgrade will fail.

Note that aptitude uses a different method for registering packages that are on hold than apt and dselect. You can identify packages on hold for aptitude with

```
# aptitude search "~ahold"
```

If you want to check which packages you had on hold for apt, you should use

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

If you changed and recompiled a package locally, and didn’t rename it or put an epoch in the version, you must put it on hold to prevent it from being upgraded.

The “hold” package state for apt can be changed using:

```
# echo nom_de_paquet hold | dpkg --set-selections
```

Replace hold with install to unset the “hold” state.

If there is anything you need to fix, it is best to make sure your APT source-list files still refer to bullseye as explained in Secció A.2.

---

4Debian’s package management system normally does not allow a package to remove or replace a file owned by another package unless it has been defined to replace that package.
4.3 Preparing APT source-list files

Before starting the upgrade you must reconfigure APT source-list files (/etc/apt/sources.list and files under /etc/apt/sources.list.d/) to add sources for bookworm and typically to remove sources for bullseye.

APT will consider all packages that can be found via any configured archive, and install the package with the highest version number, giving priority to the first entry in the files. Thus, if you have multiple mirror locations, list first the ones on local hard disks, then CD-ROMs, and then remote mirrors.

A release can often be referred to both by its codename (e.g. bullseye, bookworm) and by its status name (i.e. oldstable, stable, testing, unstable). Referring to a release by its codename has the advantage that you will never be surprised by a new release and for this reason is the approach taken here. It does of course mean that you will have to watch out for release announcements yourself. If you use the status name instead, you will just see loads of updates for packages available as soon as a release has happened.

Debian provides two announcement mailing lists to help you stay up to date on relevant information related to Debian releases:

- By subscribing to the Debian announcement mailing list (https://lists.debian.org/debian-announce/), you will receive a notification every time Debian makes a new release. Such as when bookworm changes from e.g. testing to stable.

- By subscribing to the Debian security announcement mailing list (https://lists.debian.org/debian-security-announce/), you will receive a notification every time Debian publishes a security announcement.

4.3.1 Adding APT Internet sources

On new installations the default is for APT to be set up to use the Debian APT CDN service, which should ensure that packages are automatically downloaded from a server near you in network terms. As this is a relatively new service, older installations may have configuration that still points to one of the main Debian Internet servers or one of the mirrors. If you haven’t done so yet, it is recommended to switch over to the use of the CDN service in your APT configuration.

To make use of the CDN service, add a line like this to your APT source configuration (assuming you are using main and contrib):

```
deb http://deb.debian.org/debian bookworm main contrib
```

After adding your new sources, disable the previously existing “deb” lines by placing a hash sign (#) in front of them.

However, if you get better results using a specific mirror that is close to you in network terms, this option is still available.

Debian mirror addresses can be found at https://www.debian.org/distrib/ftplist (look at the “list of Debian mirrors” section).

For example, suppose your closest Debian mirror is http://mirrors.kernel.org. If you inspect that mirror with a web browser, you will notice that the main directories are organized like this:

```
http://mirrors.kernel.org/debian/dists/bookworm/main/binary-mips64el/...
http://mirrors.kernel.org/debian/dists/bookworm/contrib/binary-mips64el/...
```

To configure APT to use a given mirror, add a line like this (again, assuming you are using main and contrib):

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

Note that the “dists” is added implicitly, and the arguments after the release name are used to expand the path into multiple directories.

Again, after adding your new sources, disable the previously existing archive entries.

4.3.2 Adding APT sources for a local mirror

Instead of using remote package mirrors, you may wish to modify the APT source-list files to use a mirror on a local disk (possibly mounted over NFS).
For example, your package mirror may be under `/var/local/debian/`, and have main directories like this:

```
/var/local/debian/dists/bookworm/main/binary-mips64el/...
/var/local/debian/dists/bookworm/contrib/binary-mips64el/...
```

To use this with `apt`, add this line to your `sources.list` file:

```
debug file:/var/local/debian bookworm main contrib
```

Note that the “dists” is added implicitly, and the arguments after the release name are used to expand the path into multiple directories.

After adding your new sources, disable the previously existing archive entries in the APT source-list files by placing a hash sign (`#`) in front of them.

### 4.3.3 Adding APT sources from optical media

If you want to use only DVDs (or CDs or Blu-ray Discs), comment out the existing entries in all the APT source-list files by placing a hash sign (`#`) in front of them.

Make sure there is a line in `/etc/fstab` that enables mounting your CD-ROM drive at the `/media/cdrom` mount point. For example, if `/dev/sr0` is your CD-ROM drive, `/etc/fstab` should contain a line like:

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

Note that there must be no spaces between the words `noauto,ro` in the fourth field.

To verify it works, insert a CD and try running:

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

Next, run:

```
# apt-cdrom add
```

for each Debian Binary CD-ROM you have, to add the data about each CD to APT’s database.

### 4.4 Upgrading packages

The recommended way to upgrade from previous Debian releases is to use the package management tool `apt`.

---

**NOTA**

`apt` is meant for interactive use, and should not be used in scripts. In scripts one should use `apt-get`, which has a stable output better suitable for parsing.

Don’t forget to mount all needed partitions (notably the root and `/usr` partitions) read-write, with a command like:

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

Next you should double-check that the APT source entries (in `/etc/apt/sources.list` and files under `/etc/apt/sources.list.d/`) refer either to “bookworm” or to “stable”. There should not be any sources entries pointing to bullseye.
4.4.1 Recording the session

It is strongly recommended that you use the `/usr/bin/script` program to record a transcript of the upgrade session. Then if a problem occurs, you will have a log of what happened, and if needed, can provide exact information in a bug report. To start the recording, type:

```
# script -t 2>/upgrade-bookwormstep.time -a ~/upgrade-bookwormstep.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-bookworm.script` to view the file.

After you have completed the upgrade, you can stop `script` by typing `exit` at the prompt.

`apt` will also log the changed package states in `/var/log/apt/history.log` and the terminal output in `/var/log/apt/term.log`. `dpkg` will, in addition, log all package state changes in `/var/log/dpkg.log`. If you use `aptitude`, it will also log state changes in `/var/log/aptitude`.

If you have used the `-t` switch for `script` you can use the `scriptreplay` program to replay the whole session:

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

4.4.2 Updating the package list

First the list of available packages for the new release needs to be fetched. This is done by executing:

```
# apt update
```

4.4.3 Make sure you have sufficient space for the upgrade

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 Secció 4.4.5. 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 can show you detailed information about the disk space needed for the installation. Before executing the upgrade, you can see this estimate by running:

```bash
# 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.
```

**NOTA**

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 Secció 4.4.4 before running this command to estimate the disk space.

If you do not have enough space for the upgrade, apt will warn you with a message like this:

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

In this situation, make sure you free up space beforehand. You can:

- Remove packages that have been previously downloaded for installation (at /var/cache/apt/archives). Cleaning up the package cache by running apt clean will remove all previously downloaded package files.

- Remove forgotten packages. If you have used aptitude or apt to manually install packages in bullseye it will have kept track of those packages you manually installed, and will be able to mark as redundant those packages pulled in by dependencies alone which are no longer needed due to a package being removed. They will not mark for removal packages that you manually installed. To remove automatically installed packages that are no longer used, run:

  ```bash
  # apt autoremove
  ```

  You can also use deborphan, debfoster, or cruft to find redundant packages. Do not blindly remove the packages these tools present, especially if you are using aggressive non-default options that are prone to false positives. It is highly recommended that you manually review the packages suggested for removal (i.e. their contents, sizes, and descriptions) before you remove them.

- 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 I and enter -i, then press S and enter ~installsize. This will give you a handy list to work with.

- 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.

- Temporarily move to another system, or permanently remove, system logs residing under /var/log/.

- Use a temporary /var/cache/apt/archives: You can use a temporary cache directory from another filesystem (USB storage device, temporary hard disk, filesystem already in use, ...).
4.4. UPGRADING PACKAGES

NOTA

Do not use an NFS mount as the network connection could be interrupted during the upgrade.

For example, if you have a USB drive mounted on /media/usbkey:

1. remove the packages that have been previously downloaded for installation:
   
   ```
   # apt clean
   ```

2. copy the directory /var/cache/apt/archives to the USB drive:
   
   ```
   # cp -ax /var/cache/apt/archives /media/usbkey/
   ```

3. mount the temporary cache directory on the current one:
   
   ```
   # mount --bind /media/usbkey/archives /var/cache/apt/archives
   ```

4. after the upgrade, restore the original /var/cache/apt/archives directory:
   
   ```
   # umount /var/cache/apt/archives
   ```

5. remove the remaining /media/usbkey/archives.

You can create the temporary cache directory on whatever filesystem is mounted on your system.

- Do a minimal upgrade of the system (see Secció 4.4.4) 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.

Note that in order to safely remove packages, it is advisable to switch your APT source-list files back to bullseye as described in Secció A.2.

4.4.4 Minimal system upgrade

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 Secció 4.4.5.

To do this, first run:

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

This has the effect of upgrading those packages which can be upgraded without requiring any other packages to be removed or installed.

The minimal system upgrade can also be useful when the system is tight on space and a full upgrade cannot be run due to space constraints.

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.

4.4.5 Upgrading the system

Once you have taken the previous steps, you are now ready to continue with the main part of the upgrade. Execute:

```
# apt full-upgrade
```
This will perform a complete upgrade of the system, installing the newest available versions of all packages, and resolving all possible dependency changes between packages in different releases. If necessary, it will install some new packages (usually new library versions, or renamed packages), and remove any conflicting obsoleted packages.

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

New versions of currently installed packages that cannot be upgraded without changing the install status of another package will be left at their current version (displayed as “held back”). This can be resolved by either using `aptitude` to choose these packages for installation or by trying `apt install` package.

### 4.5 Possible issues during upgrade

The following sections describe known issues that might appear during an upgrade to bookworm.

#### 4.5.1 Dist-upgrade fails with “Could not perform immediate configuration”

In some cases the `apt full-upgrade` step can fail after downloading packages with:

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

If that happens, running `apt full-upgrade -o APT::Immediate-Configure=0` instead should allow the upgrade to proceed.

Another possible workaround for this problem is to temporarily add both bullseye and bookworm sources to your APT source-list files and run `apt update`.

#### 4.5.2 Expected removals

The upgrade process to bookworm 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 bookworm, see Secció 4.8.

#### 4.5.3 Conflicts or Pre-Depends loops

Sometimes it’s necessary to enable the `APT::Force-LoopBreak` option in APT to be able to temporarily remove an essential package due to a Conflicts/Pre-Depends loop. `apt` will alert you of this and abort the upgrade. You can work around this by specifying the option `-o APT::Force-LoopBreak=1` on the `apt` command line.

It is possible that a system’s dependency structure can be so corrupt as to require manual intervention. Usually this means using `apt` or

```
# dpkg --remove nom_del_paquet
to eliminate some of the offending packages, or

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

In extreme cases you might have to force re-installation with a command like

# dpkg --install /camí/a/nom_del_paquet.deb
```
4.5.4 File conflicts

File conflicts should not occur if you upgrade from a “pure” bullseye system, but can occur if you have unofficial backports installed. A file conflict will result in an error like:

```
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>
```

You can try to solve a file conflict by forcibly removing the package mentioned on the last line of the error message:

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

After fixing things up, you should be able to resume the upgrade by repeating the previously described apt commands.

4.5.5 Configuration changes

During the upgrade, you will be asked questions regarding the configuration or re-configuration of several packages. When you are asked if any file in the /etc/init.d directory, or the /etc/manpath.config file should be replaced by the package maintainer’s version, it’s usually necessary to answer “yes” to ensure system consistency. You can always revert to the old versions, since they will be saved with a .dpkg-old extension.

If you’re not sure what to do, write down the name of the package or file and sort things out at a later time. You can search in the typescript file to review the information that was on the screen during the upgrade.

4.5.6 Change of session to console

If you are running the upgrade using the system’s local console you might find that at some points during the upgrade the console is shifted over to a different view and you lose visibility of the upgrade process. For example, this may happen in systems with a graphical interface when the display manager is restarted.

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 Upgrading your kernel and related packages

This section explains how to upgrade your kernel and identifies potential issues related to this upgrade. You can either install one of the linux-image-* packages provided by Debian, or compile a customized kernel from source.

Note that a lot of information in this section is based on the assumption that you will be using one of the modular Debian kernels, together with initramfs-tools and udev. If you choose to use a custom kernel that does not require an initrd or if you use a different initrd generator, some of the information may not be relevant for you.

4.6.1 Installing a kernel metapackage

When you full-upgrade from bullseye to bookworm, it is strongly recommended that you install a linux-image-* metapackage, if you have not done so before. These metapackages will automatically pull in a newer version of the kernel during upgrades. You can verify whether you have one installed by running:

```
# dpkg -l "linux-image*" | grep ^ii | grep -i meta
```
If you do not see any output, then you will either need to install a new linux-image package by hand or install a linux-image metapackage. To see a list of available linux-image metapackages, run:

```bash
# apt-cache search linux-image- | grep -i meta | 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 "4.9.0-8-amd64", it is recommended that you install `linux-image-amd64`. You may also use `apt` to see a long description of each package in order to help choose the best one available. For example:

```bash
# apt show linux-image-amd64
```

You should then use `apt install` to install it. Once this new kernel is installed you should reboot at the next available opportunity to get the benefits provided by the new kernel version. However, please have a look at Secció 5.1.2 before performing the first reboot after the upgrade.

For the more adventurous there is an easy way to compile your own custom kernel on Debian. Install the kernel sources, provided in the `linux-source` package. You can make use of the `deb-pkg` target available in the sources’ makefile for building a binary package. More information can be found in the Debian Linux Kernel Handbook (https://kernel-team.pages.debian.net/kernel-handbook/), which can also be found as the `debian-kernel-handbook` package.

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 Secció 4.4.4.

### 4.7 Preparing for the next release

After the upgrade there are several things you can do to prepare for the next release.

- Remove newly redundant or obsolete packages as described in Secció 4.4.3 and Secció 4.8. You should review which configuration files they use and consider purging the packages to remove their configuration files. See also Secció 4.7.1.

#### 4.7.1 Purging removed packages

It is generally advisable to purge removed packages. This is especially true if these have been removed in an earlier release upgrade (e.g. from the upgrade to bullseye) or they were provided by third-party vendors. In particular, old init.d scripts have been known to cause issues.

**ATENCIÓ**

Purging a package will generally also purge its log files, so you might want to back them up first.

The following command displays a list of all removed packages that may have configuration files left on the system (if any):

```bash
# dpkg -l | awk '/^rc/ { print $2 }'
```

The packages can be removed by using `apt purge`. Assuming you want to purge all of them in one go, you can use the following command:

```bash
# apt purge $(dpkg -l | awk '/^rc/ { print $2 }')
```

If you use `aptitude`, you can also use the following alternative to the commands above:

```bash
# aptitude search '~c'
# aptitude purge '~c'
```
4.8 Paquets obsolets

Introducing lots of new packages, bookworm also retires and omits quite a few old packages that were in bullseye. It provides no upgrade path for these obsolete packages. While nothing prevents you from continuing to use an obsolete package where desired, the Debian project will usually discontinue security support for it a year after bookworm’s release\(^5\), and will not normally provide other support in the meantime. Replacing them with available alternatives, if any, is recommended.

There are many reasons why packages might have been removed from the distribution: they are no longer maintained upstream; there is no longer a Debian Developer interested in maintaining the packages; the functionality they provide has been superseded by different software (or a new version); or they are no longer considered suitable for bookworm due to bugs in them. In the latter case, packages might still be present in the “unstable” distribution.

Some package management front-ends provide easy ways of finding installed packages that are no longer available from any known repository. The `aptitude` textual user interface lists them in the category “Obsolete and Locally Created Packages”, and they can be listed and purged from the commandline with:

```
# aptitude search ‘~o’
# aptitude purge ‘~o’
```

The Debian Bug Tracking System (https://bugs.debian.org/) often provides additional information on why the package was removed. You should review both the archived bug reports for the package itself and the archived bug reports for the ftp.debian.org pseudo-package (https://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=ftp.debian.org&archive=yes).

For a list of obsolete packages for Bookworm, please refer to Secció 5.3.1.

4.8.1 Transitional dummy packages

Some packages from bullseye may have been replaced in bookworm by transitional dummy packages, which are empty placeholders designed to simplify upgrades. If for instance an application that was formerly a single package has been split into several, a transitional package may be provided with the same name as the old package and with appropriate dependencies to cause the new ones to be installed. After this has happened the redundant dummy package can be safely removed.

The package descriptions for transitional dummy packages usually indicate their purpose. However, they are not uniform; in particular, some “dummy” packages are designed to be kept installed, in order to pull in a full software suite, or track the current latest version of some program. You might also find `deborphan` with the `--guess-*` options (e.g. `--guess-dummy`) useful to detect transitional dummy packages on your system.

---

\(^5\)Or for as long as there is not another release in that time frame. Typically only two stable releases are supported at any given time.
Capítol 5

Problemes a tenir en compte a bookworm

De vegades, els canvis introduïts en un nou llançament poden tenir efectes col·laterals que no podem evitar raonablement, o exponen errors en un altre lloc. Aquesta secció documenta els problemes dels tenim coneixement. Llegiu també l’errata, la documentació dels paquets, els informes d’error i altra informació que s’esmenta a Secció 6.1.

5.1 Actualització d’elements específics per bookworm

Aquesta secció inclou elements relacionats amb l’actualització de bullseye a bookworm.

5.1.1 Something

With text

5.1.2 Coses a fer després de reinicialitzar

Quan apt full-upgrade ha acabat, el procés “formal” d’actualització s’ha completat. Per l’actualització a bookworm, no hi ha accions especials a fer abans de reinicialitzar el sistema.

Quan apt full-upgrade ha acabat, el procés “formal” d’actualització ha sigut completat, però hi han altres coses que es deurien de prendre cura abans de la propera reinicialització.

add list of items here

5.2 Items not limited to the upgrade process

5.2.1 Limitacions en el suport de seguretat

Hi ha alguns paquets pels quals Debian no pot prometre proveir els requisits mínims de seguretat via backports. Aquests paquets s’esmenten a les subseccions següents.

NOTA

El paquet debian-security-support ajuda a fer seguiment de l’estat del suport de seguretat dels paquets instal·lats.
5.2.1.1 Estat de seguretat als navegadors web i els seus motors de renderització

Debian 12 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. Therefore, browsers built upon e.g. the webkit and khtml engines are included in bookworm, but not covered by security support. These browsers should not be used against untrusted websites. The webkit2gtk and wpewebkit engines are covered by security support.

Per l’ús generalitzat del navegador web recomanem Firefox o Chromium. Aquests s’actualitzaran re-construint els llançaments ESR actuals per al llançament estable. La mateixa estratègia serà aplicada a Thunderbird.

5.2.1.2 OpenJDK 17

Debian bookworm comes with an early access version of OpenJDK 17 (the next expected OpenJDK LTS version after OpenJDK 11), to avoid the rather tedious bootstrap process. The plan is for OpenJDK 17 to receive an update in bookworm to the final upstream release announced for October 2021, followed by security updates on a best effort basis, but users should not expect to see updates for every quarterly upstream security update.

5.2.1.3 Go-based packages

The Debian infrastructure currently has problems with rebuilding packages of types that systematically use static linking. Before buster this wasn’t a problem in practice, but with the growth of the Go ecosystem it means that Go-based packages will be covered by limited security support until the infrastructure is improved to deal with them maintainably.

If updates are warranted for Go development libraries, they can only come via regular point releases, which may be slow in arriving.

5.2.2 Something

Text.

5.3 Obsolescence and deprecation

5.3.1 Paquets notables obsolets

La següent llista de paquets notablement coneguts han sigut descatalogats (vegeu Secció 4.8 per a una descripció).

<table>
<thead>
<tr>
<th>La llista de paquets obsolets inclou:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• The foo package has been removed from bookworm. The successor of foo is bar.</td>
</tr>
</tbody>
</table>

---

1These engines are shipped in a number of different source packages and the concern applies to all packages shipping them. The concern also extends to web rendering engines not explicitly mentioned here, with the exception of webkit2gtk and the new wpewebkit.
5.3.2 Components desfasats a bookworm

Al proper llançament de Debian 13 (anomenada trixie) algunes de les funcionalitats quedaran desfasades. Els usuaris hauran de migrar a altres alternatives per tal d’evitar problemes a l’hora d’actualitzar a Debian 13.

Això inclou les funcionalitats següents:

- Some explanation about removal of foo.

5.3.3 No-longer-supported hardware

For a number of `arch`-based devices that were supported in bullseye, it is no longer viable for Debian to build the required Linux kernel, due to hardware limitations. The unsupported devices are:

- foo

Users of these platforms who wish to upgrade to bookworm nevertheless should keep the bullseye APT sources enabled. Before upgrading they should add an APT preferences file containing:

Package: linux-image-marvell
Pin: release n=bullseye
Pin-Priority: 900

The security support for this configuration will only last until bullseye’s End Of Life.

5.4 Known severe bugs

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 (https://bugs.debian.org/cgi-bin/pkqreport.cgi?users=release.debian.org@packages.debian.org;tag=bookworm-can-defer) that were tagged to be ignored in the last part of releasing bookworm can be found in the Debian Bug Tracking System (https://bugs.debian.org/). The following bugs were affecting bookworm at the time of the release and worth mentioning in this document:

<table>
<thead>
<tr>
<th>Bug number</th>
<th>Package (source or binary)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>922981</td>
<td>ca-certificates-java</td>
<td>ca-certificates-java: /etc/ca-certificates-update.d/jks-keystore doesn’t update /etc/ssl/certs/java/cacerts</td>
</tr>
</tbody>
</table>
Capítol 6

Més informació sobre Debian

6.1 Llegir més

Beyond these release notes and the installation guide (https://www.debian.org/releases/bookworm/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 (https://www.debian.org/doc/) and the Debian Wiki (https://wiki.debian.org/).

Trobareu informació de cada paquet instal·lada a /usr/share/doc/paquet. Esta pot incloure informació dels drets d’autor, detalls específics de Debian i la documentació del desenvolupament principal.

6.2 Trobar ajuda

Hi ha moltes fonts d’ajuda, consells i suport pels usuaris de Debian, però tan sols hauria de considerar-se si no l’heu aconseguit a cap lloc després de la recerca dins la documentació sobre problema. Esta secció proporciona una introducció curta a les que poden ser d’ajuda pels usuaris de Debian.

6.2.1 Llistes de correu

Les llistes de correu més interessants pels usuaris de Debian són la llista debian-user (en anglès) i altres llistes com debian-user-catalan (en català). Per obtenir més informació d’estes llistes i detalls de com subscriure’s, vegeu https://lists.debian.org/. Abans d’enviar la vostra pregunta, comproveu els arxius amb les respostes i respecteu les normes de la llista.

6.2.2 Internet Relay Chat

Debian té un canal IRC dedicat al suport i ajuda als usuaris de Debian que es troba a la xarxa d’IRC OFTC. Per accedir al canal, dirigiu el vostre client IRC a irc.debian.org i uniu-vos a #debian.


Per obtenir més informació d’OFTC visiteu la pàgina web del lloc web (http://www.oftc.net/).

6.3 Informes d’error

Ens esforçem per fer de Debian un sistema operatiu de gran qualitat, però això no vol dir que els paquets que subministrem estiguin totalmente lliures d’errors. D’acord amb la filosofia de “desenvolupament obert” de Debian, i com un servei als nostres usuaris, facilitem tota la informació dels errors que s’han rebut en el nostre sistema de seguiment d’errors (BTS). Podeu navegar pel BTS a https://bugs.debian.org/.

Si trobeu alguna errada a la distribució o a algun paquet de programari que forme part d’ella, informeu-ne de manera que es puga resoldre de la millor manera en algun llançament posterior. L’enviament
d’informes d’errors requereix d’una adreça vàlida de correu electrònic. El demanem perquè així podrem fer seguiment d’errors i els desenvolupadors podran posar-se en contacte amb els que l’han enviat per si és necessària alguna informació addicional.

Podeu enviar un informe d’error utilitzant el programa reportbug o bé utilitzant manualment el correu electrònic. Podeu llegir més informació del sistema de seguiment d’errors i de com utilitzar-lo llegint la documentació de referència (que trobareu a /usr/share/doc/debian si teniu el paquet doc-debian instal·lat) o en línia al sistema de seguiment d’errors (https://bugs.debian.org/).

6.4 Col·laborar amb Debian


Si podeu dedicar més temps, podrieu gestionar una de les peces de programari lliure de la col·lecció a Debian. És de molta ajuda que la gent adopte o mantinga elements que la gent ha demanat per que s’incloga a Debian. La base de dades de paquets en perspectiva i amb feina pendent (https://www.debian.org/devel/wnpp/) detalla tota esta informació. Si teniu interès en grups especials aleshores trobareu més diversió col·laborant amb alguns subprojectes (https://www.debian.org/devel/#projects) de Debian que inclouen ports a arquitectures concretes, barregues pures de Debian (https://wiki.debian.org/DebianPureBlends) per a grups d’usuaris específics, entre molts altres.

De qualsevol manera, si esteu treballant a la comunitat de programari lliure d’esta manera, com a usuari, programador, escrivint o traduint, ja esteu ajudant a l’esforç pel programari lliure. Col·laborar és gratificant i divertit, i també us permet coneixer nova gent que us donarà un càlid sentiment de comunitat.
Capítol 7

Glossari

ACPI
Advanced Configuration and Power Interface (Configuració avançada i interfície d'energia)

ALSA
Advanced Linux Sound Architecture (Arquitectura avançada de so de Linux)

BD
Disc Blu-ray

CD
Compact Disc (Disc compacte)

CD-ROM
Disc compacte de només lectura

DHCP
Protocol dinàmic de configuració de màquina

DLBD
Disc Blu-ray de dues capes

DNS
Domain Name System (Sistema de noms de domini)

DVD
Digital Versatile Disc

GIMP
GNU Image Manipulation Program (Programa de manipulació d’imatges de GNU)

GNU
GNU’s Not Unix (GNU no és Unix)

GPG
GNU Privacy Guard

LDAP

LSB
Linux Standard Base

LVM
Logical Volume Manager (Gestor de volums lògics)

MTA
Mail Transport Agent (Agent de transport de correu)
### CAPÍTOL 7. GLOSSARI

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NBD</td>
<td>Network Block Device (Dispositiu de blocs en xarxa)</td>
</tr>
<tr>
<td>NFS</td>
<td>Network File System (Sistema de fitxers en xarxa)</td>
</tr>
<tr>
<td>NIC</td>
<td>Network Interface Card (Targeta d’interfície de xarxa)</td>
</tr>
<tr>
<td>NIS</td>
<td>Network Information Service (Servei d’informació de xarxa)</td>
</tr>
<tr>
<td>PHP</td>
<td>PHP: Processador d’hipertext</td>
</tr>
<tr>
<td>RAID</td>
<td>Redundant Array of Independent Disks (Matriu redundant de discs independents)</td>
</tr>
<tr>
<td>SATA</td>
<td>Serial Advanced Technology Attachment (Tecnologia avançada d’accessoris en sèrie)</td>
</tr>
<tr>
<td>SSL</td>
<td>Secure Sockets Layer (Capa de connexió segura)</td>
</tr>
<tr>
<td>TLS</td>
<td>Transport Layer Security (Capa de seguretat de transport)</td>
</tr>
<tr>
<td>UEFI</td>
<td>Unified Extensible Firmware Interface (Interfície unificada i extensible de microprogramari)</td>
</tr>
<tr>
<td>USB</td>
<td>Universal Serial Bus (Bus sèrie universal)</td>
</tr>
<tr>
<td>UUID</td>
<td>Universally Unique Identifier (Identificador únic universal)</td>
</tr>
<tr>
<td>WPA</td>
<td>Wi-Fi Protected Access (Accés protegit a Wi-Fi)</td>
</tr>
</tbody>
</table>
Apèndix A

Gestió del vostre sistema bullseye abans de l’actualització

Este apèndix conté informació de com instal·lar o actualitzar els paquets de bullseye abans d’actualitzar a bookworm. Açò tan sols serà necessari en algunes situacions concretes.

A.1 Actualització del vostre sistema bullseye

Bàsicament no hi ha diferències entre qualsevol altra actualització de bullseye que ja hàgiu fet. L’única diferència és que necessitareu primer assegurar-vos que la vostra llista de paquets encara conté referències a bullseye tal com s’explica a Secció A.2.

Si actualitzeu el vostre sistema utilitzant una rèplica de Debian, s’actualitzarà de forma automàtica a l’últim revisió del llançament de bullseye.

A.2 Comprovació dels fitxers de llista de fonts d’APT

If any of the lines in your APT source-list files (see `sources.list(5)` ([https://manpages.debian.org/bookworm/apt/sources.list.5.html](https://manpages.debian.org/bookworm/apt/sources.list.5.html))) contain references to “stable”, this is effectively pointing to bookworm already. This might not be what you want if you are not yet ready for the upgrade.

If you have already run `apt update`, you can still get back without problems by following the procedure below.

If you have also already installed packages from bookworm, there probably is not much point in installing packages from bullseye anymore. In that case you will have to decide for yourself whether you want to continue or not. It is possible to downgrade packages, but that is not covered here.

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:`, `or URIs: tor+https: for a reference to “stable”. If you find any, change `stable` to `bullseye`.

If you have any lines starting with `deb file:`, `file:`, `or URIs: file:`, you will have to check for yourself if the location they refer to contains a bullseye or bookworm archive.

**IMPORTANT**

Do not change any lines that begin with `deb cdrom:` or `URIs: cdrom:`. Doing so would invalidate the line and you would have to run `apt-cdrom` again. Do not be alarmed if a `cdrom:` source line refers to “unstable”. Although confusing, this is normal.

Si heu fet algun canvi, guardeu el fitxer i executeu

```
# apt update
```
per refrescar la llista de paquets.

A.3 Eliminar fitxers de configuració obsolets

Abans d’actualitzar el vostre sistema a bookworm és recomanable treure els fitxers de configuració antics (com ara *.dpkg-{new,old} a sota el directori /etc) del sistema.
Apèndix B

Contribuïdors de les notes de llançament

Molta gent ha ajudat en les notes de llançament, estan incloses, però no limitades a
D. Barratt, Adam, Di Carlo, Adam, Barth, Andreas, Popescu, Andrei, Bezemer, Anne, Hilliard, Bob,
Plessy, Charles, Perrier, Christian, Berg, Christoph, Baumann, Daniel, Prévot, David, Petrișor, Eddy,
Kasper, Emmanuel, Arajärvi, Esko, Pop, Frans, Rapagnani, Giovanni, Farquharson, Gordon, Yamane,
Hideki, Wansing, Holger, Fernández-Sanguino Peña, Javier, Seidel, Jens, Meurer, Jonas, Nieder, Jo-
nathan, van Baal-Ilić, Joost, Rodin, Josip, Cristau, Julien, Rye, Justin B, Jones, LaMont, Claes, Luk,
Michlmayr, Martin, Biebl, Michael, Mühlenhoff, Moritz, Thykier, Niels, Meyerhans, Noah, Kobayashi,
Noritada, Aoki, Osamu, Gevers, Paul, Green, Peter, Bradford, Rob, Thibault, Samuel, Bienlein, Simon,
Paillard, Simon, Fritsch, Stefan, Langasek, Steve, McIntyre, Steve, Scherer, Tobias, victory, McIntyre,
Vincent, i Borgert, W. Martin.
Este document s'ha traduït a molts idiomes. Moltes gràcies als traductors!
Índex alfabètic

A
Apache, 4

B
BIND, 4

C
Calligra, 4
Cryptsetup, 4

D
DocBook XML, 2
Dovecot, 4

E
Exim, 4

G
GCC, 4
GIMP, 4
GNOME, 3
GNUcash, 4
GnuPG, 4

I
Inkscape, 4

K
KDE, 3

L
LibreOffice, 3
LXDE, 3
LXQt, 3

M
MariaDB, 4
MATE, 3

N
Nginx, 4

O
OpenJDK, 4
OpenSSH, 4

P
packages
apt, 2, 13
apt-listchanges, 16
aptitude, 10, 15, 19
bar, 22
ca-certificates-java, 23
dblatex, 2
debian-goodies, 15
debian-kernel-handbook, 19
debian-security-support, 21
doc-debian, 26
docbook-xsl, 2
foo, 22
initramfs-tools, 8, 18
linux-image-*, 18
linux-image-amd64, 19
linux-source, 19
localepurge, 15
popularity-contest, 15
release-notes, 1
synaptic, 10
tinc, 9
udev, 18
upgrade-reports, 1
xmlroff, 2
xsltproc, 2
Perl, 4
PHP, 4
Postfix, 4
PostgreSQL, 4

X
Xfce, 3