

Notes de llançament per Debian 10 (buster), 64-bit PC

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

26 de setembre de 2020

Notes de llançament per Debian 10 (buster), 64-bit PC

Aquest document és programari lliure, el podeu redistribuir 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 COMERCIALITACIÓ o IDONEÏTAT PER A UN PROPÒSIT PARTICULAR. Consulteu la llicència general pública de GNU per a més detalls.

Hauríeu 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> i a `/usr/share/common-licenses/GPL-2` en els sistemes Debian.

Índex

1	Introducció	1
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	What's new in Debian 10	3
2.1	Supported architectures	3
2.2	What's new in the distribution?	3
2.2.1	UEFI Secure Boot	4
2.2.2	AppArmor enabled per default	4
2.2.3	Optional hardening of APT	5
2.2.4	Unattended-upgrades for stable point releases	5
2.2.5	Substantially improved man pages for German speaking users	5
2.2.6	Network filtering based on nftables framework by default	5
2.2.7	Cryptsetup defaults to on-disk LUKS2 format	6
2.2.8	Driverless printing with CUPS 2.2.10	6
2.2.9	Basic support for Allwinner A64 based devices	6
2.2.10	News from Debian Med Blend	7
2.2.11	GNOME defaults to Wayland	7
2.2.12	Merged /usr on fresh installs	7
2.2.13	News from Debian Live team	7
3	El sistema d'instal·lació	9
3.1	Què hi ha de nou al sistema d'instal·lació?	9
3.1.1	Instal·lació automatitzada	9
4	Upgrades from Debian 9 (stretch)	11
4.1	Preparació de l'actualització	11
4.1.1	Back up any data or configuration information	11
4.1.2	Inform users in advance	11
4.1.3	Prepare for downtime on services	11
4.1.4	Prepare for recovery	12
4.1.4.1	Debug shell during boot using initrd	12
4.1.4.2	Debug shell during boot using systemd	12
4.1.5	Prepare a safe environment for the upgrade	13
4.1.6	Verify network interface name support	13
4.2	Checking APT configuration status	13
4.2.1	The proposed-updates section	14
4.2.2	Unofficial sources	14
4.2.3	Disabling APT pinning	14
4.2.4	Checking packages status	14
4.3	Preparing APT source-list files	15
4.3.1	Adding APT Internet sources	15
4.3.2	Adding APT sources for a local mirror	16
4.3.3	Adding APT sources from optical media	16
4.4	Upgrading packages	16
4.4.1	Recording the session	17
4.4.2	Updating the package list	17
4.4.3	Make sure you have sufficient space for the upgrade	18
4.4.4	Minimal system upgrade	20
4.4.5	Upgrading the system	20
4.5	Possible issues during upgrade	20
4.5.1	Dist-upgrade fails with "Could not perform immediate configuration"	20
4.5.2	Expected removals	20

4.5.3	Conflicts or Pre-Depends loops	21
4.5.4	File conflicts	21
4.5.5	Configuration changes	21
4.5.6	Change of session to console	21
4.6	Upgrading your kernel and related packages	22
4.6.1	Installing a kernel metapackage	22
4.7	Preparing for the next release	22
4.7.1	Purging removed packages	23
4.8	Paquets obsolets	23
4.8.1	Transitional dummy packages	24
5	Problemes a tenir en compte a buster	25
5.1	Upgrade specific items for buster	25
5.1.1	Hidepid mount option for procsfs unsupported	25
5.1.2	ybind fails to start with -no-dbus	25
5.1.3	NIS server does not answer NIS client requests by default	25
5.1.4	sshd fails to authenticate	25
5.1.5	Daemons fail to start or system appears to hang during boot	26
5.1.6	Migrating from legacy network interface names	26
5.1.7	Module configuration for bonding and dummy interfaces	27
5.1.8	OpenSSL default version and security level raised	27
5.1.9	Some applications don't work in GNOME on Wayland	27
5.1.10	Noteworthy obsolete packages	27
5.1.11	Components desfasats a buster	28
5.1.12	Things to do post upgrade before rebooting	28
5.1.13	SysV init related packages no longer required	28
5.2	Limitaciones en el suport de seguretat	29
5.2.1	Estat de seguretat als navegadors web i els seus motors de renderització	29
5.2.2	Paquets basats en el llenguatge Go	29
5.3	Problems específics de paquets	29
5.3.1	Glibc requires Linux kernel 3.2 or higher	30
5.3.2	Semantics for using environment variables for su changed	30
5.3.3	Existing PostgreSQL databases need to be reindexed	30
5.3.4	mutt i neomutt	30
5.3.5	Accessing GNOME Settings app without mouse	30
5.3.6	gnome-disk-utility fails to change LUKS password causing permanent data loss (buster 10.0 only)	30
5.3.7	evolution-ews has been dropped, and email inboxes using Exchange, Office365 or Outlook server will be removed	31
5.3.8	Calamares installer leaves disk encryption keys readable	31
5.3.9	S3QL URL changes for Amazon S3 buckets	31
5.3.10	Split in configuration for logrotate	31
6	Més informació sobre Debian	33
6.1	Llegir més	33
6.2	Trobar ajuda	33
6.2.1	Llistes de correu	33
6.2.2	Internet Relay Chat	33
6.3	Informes d'error	33
6.4	Col·laborar amb Debian	34
7	Glossari	35
A	Gestió del vostre sistema stretch abans de l'actualització	37
A.1	Actualització del vostre sistema stretch	37
A.2	Comprovació dels fitxers de llista de fonts d'APT	37
A.3	Eliminar fitxers de configuració obsolets	38
A.4	Actualitza la configuració local antiga a UTF-8	38

B	Contribuïdors de les notes de llançament	39
	Índex alfabètic	41

Capítol 1

Introducció

Este document informa als usuaris de la distribució Debian dels principals canvis a la versió 10 (nom en clau buster).

Les notes de llançament ofereixen informació sobre com actualitzar de forma segura des de la versió 9 (nom en clau stretch) a la versió actual, i informen als usuaris de problemes potencials coneguts que es poden trobar durant eixe procés.

Podeu aconseguir la versió més recent d'este document a <https://www.debian.org/releases/buster/releasenotes>. Si teniu dubtes, comproveu la data del document que hi ha a la primera pàgina i assegureu-vos que esteu llegint la darrera versió.

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 9). 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 9.

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 puguen 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/) (<https://bugs.debian.org/>), al paquet `release-notes`. Reviseu primer si hi ha cap [informe d'error existent](https://bugs.debian.org/release-notes) (<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.

Apreciem, 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 d'stretch a buster és benvinguda. Si voleu compartir alguna informació, envieu un informe d'error amb els vostres resultats al [sistema de seguiment d'errors](https://bugs.debian.org/) (<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 a `/var/lib/dpkg/status` en `/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

El codi font d'este document està en format DocBook XML. La versió HTML es genera utilitzant `docbook-xsl` i `xsltproc`. La versió en PDF es genera utilitzant `dblatex` o bé `xmlroff`. El codi font de les notes de llançament està en el dipòsit Git del *Projecte de documentació de Debian*. Podeu utilitzar la **interfície web** (<https://salsa.debian.org/ddp-team/release-notes/>) per accedir individualment als seus fitxers des de la web i veure els canvis. Per a més informació de com accedir a Git, consulteu les **pàgines d'informació de VCS del projecte de documentació de Debian** (<https://www.debian.org/doc/vcs>).

Capítol 2

What's new in Debian 10

The [Wiki](https://wiki.debian.org/NewInBuster) (<https://wiki.debian.org/NewInBuster>) has more information about this topic.

2.1 Supported architectures

The following are the officially supported architectures for Debian 10:

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

You can read more about port status, and port-specific information for your architecture at the [Debian port web pages](https://www.debian.org/ports/) (<https://www.debian.org/ports/>).

2.2 What's new in the distribution?

This new release of Debian again comes with a lot more software than its predecessor stretch; the distribution includes over 13370 new packages, for a total of over 57703 packages. Most of the software in the distribution has been updated: over 35532 software packages (this is 62% of all packages in stretch). Also, a significant number of packages (over 7278, 13% of the packages in stretch) have for various reasons been removed from the distribution. You will not see any updates for these packages and they will be marked as "obsolete" in package management front-ends; see Secció 4.8.

Debian again ships with several desktop applications and environments. Among others it now includes the desktop environments GNOME 3.30, KDE Plasma 5.14, LXDE 10, LXQt 0.14, MATE 1.20, and Xfce 4.12.

Productivity applications have also been upgraded, including the office suites:

- LibreOffice is upgraded to version 6.1;
- Calligra is upgraded to 3.1.
- GNUMcash is upgraded to 3.4;

With buster, Debian for the first time brings a mandatory access control framework enabled per default. New installations of Debian buster will have AppArmor installed and enabled per default. See below for more information.

Besides, buster is the first Debian release to ship with Rust based programs such as Firefox, rippgrep, fd, exa, etc. and a significant number of Rust based libraries (more than 450). Buster ships with Rustc 1.34.

Updates of other desktop applications include the upgrade to Evolution 3.30.

Among many others, this release also includes the following software updates:

Package	Version in 9 (stretch)	Version in 10 (buster)
Apache	2.4.25	2.4.38
BIND DNS Server	9.10	9.11
Cryptsetup	1.7	2.1
Dovecot MTA	2.2.27	2.3.4
Emacs	24.5 and 25.1	26.1
Exim default e-mail server	4.89	4.92
GNU Compiler Collection as default compiler	6.3	7.4 and 8.3
GIMP	2.8.18	2.10.8
GnuPG	2.1	2.2
Inkscape	0.92.1	0.92.4
the GNU C library	2.24	2.28
lighttpd	1.4.45	1.4.53
Linux kernel image	4.9 series	4.19 series
LLVM/Clang toolchain	3.7	6.0.1 and 7.0.1 (default)
MariaDB	10.1	10.3
Nginx	1.10	1.14
OpenJDK	8	11
OpenSSH	7.4p1	7.9p1
Perl	5.24	5.28
PHP	7.0	7.3
Postfix MTA	3.1.8	3.3.2
PostgreSQL	9.6	11
Python 3	3.5.3	3.7.3
Rustc		1.34
Samba	4.5	4.9
Vim	8.0	8.1

2.2.1 UEFI Secure Boot

Secure Boot is a feature enabled on most PCs that prevents loading unsigned code, protecting against some kinds of bootkit and rootkit.

Debian can now be installed and run on most PCs with Secure Boot enabled.

It is possible to enable Secure Boot on a system that has an existing Debian installation, if it already boots using UEFI. Before doing this, it's necessary to install `shim-signed`, `grub-efi-amd64-signed` or `grub-efi-ia32-signed`, and a Linux kernel package from buster.

Some features of GRUB and Linux are restricted in Secure Boot mode, to prevent modifications to their code.

More information can be found on the Debian wiki at [SecureBoot](https://wiki.debian.org/SecureBoot) (<https://wiki.debian.org/SecureBoot>).

2.2.2 AppArmor enabled per default

Debian buster has AppArmor enabled per default. AppArmor is a mandatory access control framework for restricting programs' capabilities (such as mount, ptrace, and signal permissions, or file read, write, and execute access) by defining per-program profiles.

The `apparmor` package ships with AppArmor profiles for several programs. Some other packages, such as `evince`, include profiles for the programs they ship. More profiles can be found in the

`apparmor-profiles-extra` package.

AppArmor is pulled in due to a `Recommends` by the buster Linux kernel package. On systems that are configured to not install recommended packages by default, the `apparmor` package can be installed manually in order to enable AppArmor.

2.2.3 Optional hardening of APT

All methods provided by APT (e.g. `http`, and `https`) except for `cdrom`, `gpgv`, and `rsh` can make use of `seccomp-BPF` sandboxing as supplied by the Linux kernel to restrict the list of allowed system calls, and trap all others with a `SIGSYS` signal. This sandboxing is currently opt-in and needs to be enabled with:

```
APT::Sandbox::Seccomp is a boolean to turn it on/off
```

Two options can be used to configure this further:

```
APT::Sandbox::Seccomp::Trap is a list of names of more syscalls to trap
APT::Sandbox::Seccomp::Allow is a list of names of more syscalls to allow
```

2.2.4 Unattended-upgrades for stable point releases

Previous versions of `unattended-upgrades` defaulted to installing only upgrades that came from the security suite. In buster it now also automates upgrading to the latest stable point release. For details, see the package's `NEWS.Debian` file.

2.2.5 Substantially improved man pages for German speaking users

The documentation (man-pages) for several projects like `systemd`, `util-linux` and `mutt` has been substantially extended. Please install `manpages-de` to benefit from the improvements. During the lifetime of buster further new/improved translations will be provided within the `backports` archive.

2.2.6 Network filtering based on nftables framework by default

Starting with `iptables` v1.8.2 the binary package includes `iptables-nft` and `iptables-legacy`, two variants of the `iptables` command line interface. The `nftables`-based variant, using the `nf_tables` Linux kernel subsystem, is the default in buster. The legacy variant uses the `x_tables` Linux kernel subsystem. The `update-alternatives` system can be used to select one variant or the other.

This applies to all related tools and utilities:

- `iptables`
- `iptables-save`
- `iptables-restore`
- `ip6tables`
- `ip6tables-save`
- `ip6tables-restore`
- `arptables`
- `arptables-save`
- `arptables-restore`
- `ebtables`
- `ebtables-save`
- `ebtables-restore`

All these have also gained `-nft` and `-legacy` variants. The `-nft` option is for users who can't or don't want to migrate to the native `nftables` command line interface. However, users are strongly encouraged to switch to the `nftables` interface rather than using `iptables`.

`nftables` provides a full replacement for `iptables`, with much better performance, a refreshed syntax, better support for IPv4/IPv6 dual-stack firewalls, full atomic operations for dynamic ruleset updates, a Netlink API for third party applications, faster packet classification through enhanced generic set and map infrastructures, and **many other improvements** (<https://wiki.nftables.org>).

This change is in line with what other major Linux distributions are doing, such as RedHat, which now uses `nftables` as its **default firewalling tool** (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8-beta/html-single/8.0_beta_release_notes/index#networking_2).

Also, please note that all `iptables` binaries are now installed in `/usr/sbin` instead of `/sbin`. A compatibility symlink is in place, but will be dropped after the buster release cycle. Hardcoded paths to the binaries in scripts will need to be corrected and are worth avoiding.

Extensive documentation is available in the package's README and NEWS files and on the **Debian Wiki** (<https://wiki.debian.org/nftables>).

2.2.7 Cryptsetup defaults to on-disk LUKS2 format

The `cryptsetup` version shipped with Debian buster uses the new on-disk LUKS2 format. New LUKS volumes will use this format by default.

Unlike the previous LUKS1 format, LUKS2 provides redundancy of metadata, detection of metadata corruption, and configurable PBKDF algorithms. Authenticated encryption is supported as well, but still marked as experimental.

Existing LUKS1 volumes will not be updated automatically. They can be converted, but not all LUKS2 features will be available due to header size incompatibilities. See the **cryptsetup** (<https://manpages.debian.org/buster/cryptsetup>) manpage for more information.

Please note that the GNU GRUB bootloader doesn't support the LUKS2 format yet. See the corresponding **documentation** (<https://cryptsetup-team.pages.debian.net/cryptsetup/encrypted-boot.html>) for further information on how to install Debian 10 with encrypted boot.

2.2.8 Driverless printing with CUPS 2.2.10

Debian 10 provides CUPS 2.2.10 and `cups-filters` 1.21.6. Together these give a user everything that is needed to take advantage of **driverless printing** (<https://wiki.debian.org/DriverlessPrinting>). The principal requirement is that a network print queue or printer offers an AirPrint service. A modern IPP printer is highly likely to be AirPrint-capable; a Debian CUPS print queue is always AirPrint-enabled.

In essence, the DNS-SD (Bonjour) broadcasts from a CUPS server advertising a queue, or those from IPP printers, are capable of being displayed in the print dialogs of applications without any action being required on the part of a user. An additional benefit is that the use of non-free vendor printing drivers and plugins can be dispensed with.

A default installation of the `cups` package also installs the package `cups-browsed`; print queues and IPP printers will now be automatically set up and managed by this utility. This is the **recommended way** (<https://wiki.debian.org/QuickPrintQueuesCUPS>) for a user to experience seamless and trouble-free driverless printing.

2.2.9 Basic support for Allwinner A64 based devices

Thanks to the efforts of the **linux-sunxi community** (<https://linux-sunxi.org>) Debian buster will have basic support for many devices based on the Allwinner A64 SoC. This includes FriendlyARM NanoPi A64; Olimex A64-OLinuXino and TERES-A64; PINE64 PINE A64/A64+/A64-LTS, SOPINE, and Pinebook; SINOVOIP Banana Pi BPI-M64; and Xunlong Orange Pi Win(Plus).

The essential features of these devices (e.g. serial console, ethernet, USB ports and basic video output) should work with the kernel from buster. More advanced features (e.g. audio or accelerated video) are included or scheduled to be included in later kernels, which will be made available as usual through the **backports archive** (<https://backports.debian.org>). See also **the status page** (https://linux-sunxi.org/Linux_mainlining_effort) for the Linux mainlining effort.

2.2.10 News from Debian Med Blend

The Debian Med team has added several new packages and updates for software targeting life sciences and medicine. The effort to add Continuous Integration support for the packages in this field was (and will be) continued.

To install packages maintained by the Debian Med team, install the metapackages named `med-*`, which are at version 3.3 for Debian buster. Feel free to visit the [Debian Med tasks pages](http://blends.debian.org/med/tasks) (<http://blends.debian.org/med/tasks>) to see the full range of biological and medical software available in Debian.

2.2.11 GNOME defaults to Wayland

Following upstream, GNOME in buster defaults to using the Wayland display server instead of Xorg. Wayland has a simpler and more modern design, which has advantages for security.

The Xorg display server is still installed by default and the default display manager still allows you to choose it as the display server for the next session, which may be needed if you want to use some applications (see Secció [5.1.9](#)).

People requiring accessibility features of the display server, e.g. global keyboard shortcuts, are recommended to use Xorg instead of Wayland.

2.2.12 Merged `/usr` on fresh installs

On fresh installs, the content of `/bin`, `/sbin` and `/lib` will be installed into their `/usr` counterpart by default. `/bin`, `/sbin` and `/lib` will be soft-links pointing at their directory counterpart under `/usr/`. In graphical form:

```
/bin → /usr/bin
/sbin → /usr/sbin
/lib → /usr/lib
```

When upgrading to buster, systems are left as they are, although the `usrmerge` package exists to do the conversion if desired. The [freedesktop.org](https://www.freedesktop.org) (<https://www.freedesktop.org>) project hosts a [Wiki](https://www.freedesktop.org/wiki/Software/systemd/TheCaseForTheUsrMerge/) (<https://www.freedesktop.org/wiki/Software/systemd/TheCaseForTheUsrMerge/>) with most of the rationale.

This change shouldn't impact normal users that only run packages provided by Debian, but it may be something that people that use or build third party software want to be aware of.

2.2.13 News from Debian Live team

The Debian Live team is proud to introduce LXQt live ISOs as a new flavor. LXQt is a lightweight Qt desktop environment. It will not get in your way. It will not hang or slow down your system. It is focused on being a classic desktop with a modern look and feel.

The LXQt desktop environment offered in the Debian Live LXQt project is pure, unmodified, so you will get the standard desktop experience that the LXQt developers created for their popular operating system. Users are presented with the standard LXQt layout comprised of a single panel (taskbar) located on the bottom edge of the screen, which includes various useful applets, such as the Main Menu, task manager, app launcher, system tray area, and integrated calendar.

The buster live images come with something new that a bunch of other distributions have also adopted, which is the Calamares installer. Calamares is an independent installer project (they call it "The universal installer framework") which offers a Qt based interface for installing a system. It doesn't replace `debian-installer` on the live images; rather, it serves a different audience.

Calamares is really easy to use, with friendly guided partitioning and really simple full-disk encryption setup. It doesn't cover all the advanced features of `debian-installer` (although it very recently got RAID support) and it doesn't have an unattended install mode either. However, for 95%+ of desktop and laptop users, Calamares is a much easier way to get a system installed, which makes it very appropriate for live systems. For anyone who needs anything more complicated, or who's doing a mass-install, `debian-installer` is still available in both text and GUI forms.

Debian Live Buster re-introduces the standard live image. This is a basic Debian image that contains a base Debian system without any graphical user interface. Because it installs from a squashfs image

rather than installing the system files using **dpkg**, installation times are a lot faster than installing from a minimal Debian installation image.

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 buster es podent trobar junt a la guia d'instal·lació a la [web de Debian](https://www.debian.org/releases/buster/debian-installer/) (<https://www.debian.org/releases/buster/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/buster/debian-installer/index#errata) (<https://www.debian.org/releases/buster/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 9 amb el resultat que s'han aconseguit millores tant al suport de maquinari com noves i excitants característiques.

Sobretot hi ha el suport inicial per l'arrancada segura amb UEFI (vegeu Section [2.2.1](#)), que s'ha afegit a les imatges d'instal·lació.

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

3.1.1 Instal·lació automatitzada

Alguns dels canvis que s'han anomenat en la secció anterior impliquen canvis en el suport de l'instal·lador per instal·lacions automàtiques amb fitxers de configuració prèvia. Açò vol dir que si ja teníeu fitxers de configuració prèvia de la instal·lació d'stretch, no espereu que funcionen en el nou instal·lador sense cap modificació.

La [guia d'instal·lació](https://www.debian.org/releases/buster/installmanual) (<https://www.debian.org/releases/buster/installmanual>) té un apèndix separat amb una extensa documentació de com utilitzar la configuració prèvia.

Capítol 4

Upgrades from Debian 9 (stretch)

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 buster, 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 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 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 `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 buster 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](https://www.debian.org/releases/buster/installmanual) (<https://www.debian.org/releases/buster/installmanual>) and the [Debian Installer FAQ](https://wiki.debian.org/DebianInstaller/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 image or a Linux live CD. After booting from that, you should be able to mount your root file system and `chroot` into it to investigate and fix the problem.

4.1.4.1 Debug shell during boot using `initrd`

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.

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 `systemd.unit=rescue.target` to the kernel parameters.

¹If the `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.

²For 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.

³This feature can be disabled by adding the parameter `panic=0` to your boot 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 /
```

More information on debugging a broken boot under `systemd` can be found in the [Diagnosing Boot Problems](https://freedesktop.org/wiki/Software/systemd/Debugging/) (<https://freedesktop.org/wiki/Software/systemd/Debugging/>) article.

4.1.5 Prepare a safe environment for the upgrade

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.

4.1.6 Verify network interface name support

Systems upgraded from older releases that still use network interfaces with names like `eth0` or `wlan0` are at risk of losing networking once they switch to buster; see Secció [5.1.6](#) for migration instructions.

4.2 Checking APT configuration status

The upgrade process described in this chapter has been designed for “pure” Debian stable systems. If your APT configuration mentions additional sources besides `stretch`, 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/buster//buster/apt/sources.list.5.html) (<https://manpages.debian.org/buster//buster/apt/sources.list.5.html>). If your system is using multiple source-list files then you will need to ensure they stay consistent.

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).

```
$ aptitude search '~i(!~ODebian)'
$ apt-forktracer | sort
```

Direct upgrades from Debian releases older than 9 (`stretch`) are not supported. Please follow the instructions in the [Release Notes for Debian 9](https://www.debian.org/releases/stretch/releasenotes) (<https://www.debian.org/releases/stretch/releasenotes>) to upgrade to Debian 9 first.

This procedure also assumes your system has been updated to the latest point release of `stretch`. If you have not done this or are unsure, follow the instructions in Secció [A.1](#).

You should also 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 `stretch` and not to `stable` or `buster`; see Secció [A.2](#).

It is a good idea to **remove obsolete packages** from your system before upgrading.

4.2.1 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.

4.2.2 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 buster 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 stretch system. Such packages are most likely to cause problems during an upgrade as they may result in file conflicts⁴. Secció 4.5 has some information on how to deal with file conflicts if they should occur.

4.2.3 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)`.

4.2.4 Checking packages 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:

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.

```
# 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 stretch as explained in Secció [A.2](#).

4.3 Preparing APT source-list files

Before starting the upgrade you must reconfigure APT’s source-list files (`/etc/apt/sources.list` and files under `/etc/apt/sources.list.d/`).

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. `stretch`, `buster`) 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/) (<https://lists.debian.org/debian-announce/>) you will receive a notification every time Debian makes a new release. Such as when `buster` changes from e.g. `stable` to `oldstable`.
- By [subscribing to the Debian security announcement mailing list](https://lists.debian.org/debian-security-announce/) (<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 buster 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/buster/main/binary-amd64/...
http://mirrors.kernel.org/debian/dists/buster/contrib/binary-amd64/...
```

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 buster 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/buster/main/binary-amd64/...
/var/local/debian/dists/buster/contrib/binary-amd64/...
```

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

```
deb file:/var/local/debian buster 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 -lF /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 /puntu_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 “buster” or to “stable”. There should not be any sources entries pointing to stretch.

NOTA



Source lines for a CD-ROM might sometimes refer to “unstable”; although this may be confusing, you should *not* change it.

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-busterstep.time -a ~/upgrade-busterstep.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-buster.s` 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-busterstep.time ~/upgrade-busterstep.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
```

NOTA



Users of `apt-secure` may find issues when using **aptitude** or **apt-get**. For `apt-get`, you can use **apt-get update --allow-releaseinfo-change**.

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:

```
# 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 stretch 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:

```
# apt autoremove
```


You can also use **deborphan**, **debfoister**, 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 **l** 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, ...).

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 /media/usbkey/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 stretch 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-get upgrade
```

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 buster.

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 stretch and buster sources to your APT source-list files and run `apt update`.

4.5.2 Expected removals

The upgrade process to buster 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 buster, 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” stretch 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 stretch to buster, 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:

```
# 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:

```
# 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 5.1.12 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/) (<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 stretch) 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):

```
# 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:

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

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

```
# aptitude search '~c'
# aptitude purge '~c'
```

4.8 Paquets obsolets

Introducing lots of new packages, buster also retires and omits quite a few old packages that were in stretch. 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 buster's release⁵, 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 buster 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 Buster, please refer to Secció **5.1.10**.

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

4.8.1 Transitional dummy packages

Some packages from stretch may have been replaced in buster 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.

Capítol 5

Problemes a tenir en compte a buster

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 Secció 6.1.

5.1 Upgrade specific items for buster

This section covers items related to the upgrade from stretch to buster.

5.1.1 Hidepid mount option for procs unsupported

Using the `hidepid` mount option for `/proc` is known to cause problems with current versions of `systemd`, and is considered by `systemd` upstream to be an unsupported configuration. Users who have modified `/etc/fstab` to enable this option are advised to disable it before the upgrade, to ensure login sessions work on buster. (A possible route to re-enabling it is outlined on the wiki's [Hardening](https://wiki.debian.org/Hardening#Mounting_.2Fproc_with_hidepid) (https://wiki.debian.org/Hardening#Mounting_.2Fproc_with_hidepid) page.)

5.1.2 ypbind fails to start with -no-dbus

The default options of `ypbind` have changed. However, if you have modified this file the old default will not be updated and you must make sure that the `YPBINDARGS=` option in `/etc/default/nis` does not include `-no-dbus`. With `-no-dbus` present, **`ypbind`** will fail to start, and you may not be able to log in. For more info see [bug #906436](https://bugs.debian.org/906436) (<https://bugs.debian.org/906436>).

5.1.3 NIS server does not answer NIS client requests by default

The default behavior of `rpcbind` has changed to no longer answer remote calls from NIS clients. On NIS servers you will need to add the (Debian-specific) `-r` flag to the command line options of **`rpcbind`**, otherwise users will not be able to log into your NIS client machines. For more info see [bug #935492](https://bugs.debian.org/935492) (<https://bugs.debian.org/935492>).

5.1.4 sshd fails to authenticate

The semantics of `PubkeyAcceptedKeyTypes` and the similar `HostbasedAcceptedKeyTypes` options for `sshd` have changed. These now specify signature algorithms that are accepted for their respective authentication mechanism, where previously they specified accepted key types. This distinction matters when using the RSA/SHA2 signature algorithms `rsa-sha2-256`, `rsa-sha2-512` and their certificate counterparts. Configurations that override these options but omit these algorithm names may cause unexpected authentication failures.

No action is required for configurations that accept the default for these options.

5.1.5 Daemons fail to start or system appears to hang during boot

Due to `systemd` needing entropy during boot and the kernel treating such calls as blocking when available entropy is low, the system may hang for minutes to hours until the randomness subsystem is sufficiently initialized (`random: crng init done`). For amd64 systems supporting the `RDRAND` instruction this issue is avoided by the Debian kernel using this instruction by default (`CONFIG_RANDOM_TRUST_CPU`).

Non-amd64 systems and some types of virtual machines need to provide a different source of entropy to continue fast booting. `haveged` has been chosen for this within the Debian Installer project and may be a valid option if hardware entropy is not available on the system. On virtual machines consider forwarding entropy from the host to the VMs via `virtio_rng`.

If you read this after upgrading a remote system to buster, ping the system on the network continuously as this adds entropy to the randomness pool and the system will eventually be reachable by `ssh` again.

See [the wiki](https://wiki.debian.org/BoottimeEntropyStarvation) (<https://wiki.debian.org/BoottimeEntropyStarvation>) and [DLange's overview of the issue](https://daniel-lange.com/archives/152-hello-buster.html) (<https://daniel-lange.com/archives/152-hello-buster.html>) for other options.

5.1.6 Migrating from legacy network interface names

If your system was upgraded from an earlier release, and still uses the old-style network interface names that were deprecated with stretch (such as `eth0` or `wlan0`), you should be aware that the mechanism of defining their names via `/etc/udev/rules.d/70-persistent-net.rules` is officially not supported by `udev` in buster (while it may still work in some cases). To avoid the danger of your machine losing networking after the upgrade to buster, it is recommended that you migrate in advance to the new naming scheme (usually meaning names like `enp0s1` or `wlp2s5`, which incorporate PCI bus- and slot-numbers). Take care to update any interface names hard-coded in configuration for firewalls, `ifupdown`, and so on.

The alternative is to switch to a supported mechanism for enforcing the old naming scheme, such as a `systemd .link` file (see [systemd.link\(5\)](https://manpages.debian.org/buster/systemd.link) (<https://manpages.debian.org/buster/systemd.link>)). The `net.ifnames=0` kernel commandline option might also work for systems with only one network interface (of a given type).

To find the new-style names that will be used, first find the current names of the relevant interfaces:

```
$ echo /sys/class/net/[ew]*
```

For each of these names, check whether it is used in configuration files, and what name `udev` would prefer to use for it:

```
$ sudo rgrep -w eth0 /etc
$ udevadm test-builtin net_id /sys/class/net/eth0 2>/dev/null
```

This should give enough information to devise a migration plan. (If the `udevadm` output includes an “onboard” or “slot” name, that takes priority; MAC-based names are normally treated as a fallback, but may be needed for USB network hardware.)

Once you are ready to carry out the switch, disable `70-persistent-net.rules` either by renaming it or by commenting out individual lines. On virtual machines you will need to remove the files `/etc/systemd/network/99-default.link` and (if using `virtio` network devices) `/etc/systemd/network/50-virtio-kernel-names.link`. Then rebuild the `initrd`:

```
$ sudo update-initramfs -u
```

and reboot. Your system should now have new-style network interface names. Adjust any remaining configuration files, and test your system.

See [the wiki](https://wiki.debian.org/NetworkInterfaceNames) (<https://wiki.debian.org/NetworkInterfaceNames>), [upstream documentation](https://www.freedesktop.org/software/systemd/man/systemd.net-naming-scheme.html) (<https://www.freedesktop.org/software/systemd/man/systemd.net-naming-scheme.html>), and the `udev` `README.Debian` for further information.

5.1.7 Module configuration for bonding and dummy interfaces

Systems using channel bonding and/or dummy interfaces, for instance to configure a machine as a router, may encounter problems upgrading to buster. New versions of `systemd` install a file `/lib/modprobe.d/systemd.conf` (intended to simplify configuration via `systemd-networkd`) which contains the lines

```
options bonding max_bonds=0
options dummy numdummies=0
```

Admins who were depending on different values will need to ensure they are set in the correct way to take precedence. A file in `/etc/modprobe.d` will override one with the same name under `/lib/modprobe.d`, but the names are processed in alphabetical order, so `/lib/modprobe.d/systemd.conf` follows and overrides (for instance) `/etc/modprobe.d/dummy.conf`. Make sure that any local configuration file has a name that sorts after “`systemd.conf`”, such as “`/etc/modprobe.d/zz-local.conf`”.

5.1.8 OpenSSL default version and security level raised

Following various security recommendations, the default minimum TLS version has been changed from TLSv1 to TLSv1.2.

The default security level for TLS connections has also been increased from level 1 to level 2. This moves from the 80 bit security level to the 112 bit security level and will require 2048 bit or larger RSA and DHE keys, 224 bit or larger ECC keys, and SHA-2.

The system wide settings can be changed in `/etc/ssl/openssl.cnf`. Applications might also have an application specific way to override the defaults.

In the default `/etc/ssl/openssl.cnf` there is a `MinProtocol` and `CipherString` line. The `CipherString` can also set the security level. Information about the security levels can be found in the [SSL_CTX_set_security_level\(3ssl\)](https://manpages.debian.org/buster/SSL_CTX_set_security_level(3ssl)) ([https://manpages.debian.org/buster/SSL_CTX_set_security_level\(3ssl\)](https://manpages.debian.org/buster/SSL_CTX_set_security_level(3ssl))) manpage. The list of valid strings for the minimum protocol version can be found in [SSL_CONF_cmd\(3ssl\)](https://manpages.debian.org/buster/SSL_CONF_cmd(3ssl)) ([https://manpages.debian.org/buster/SSL_CONF_cmd\(3ssl\)](https://manpages.debian.org/buster/SSL_CONF_cmd(3ssl))). Other information can be found in [ciphers\(1ssl\)](https://manpages.debian.org/buster/ciphers(1ssl)) ([https://manpages.debian.org/buster/ciphers\(1ssl\)](https://manpages.debian.org/buster/ciphers(1ssl))) and [config\(5ssl\)](https://manpages.debian.org/buster/config(5ssl)) ([https://manpages.debian.org/buster/config\(5ssl\)](https://manpages.debian.org/buster/config(5ssl))).

Changing the system wide defaults in `/etc/ssl/openssl.cnf` back to their previous values can be done by setting:

```
MinProtocol = None
CipherString = DEFAULT
```

It's recommended that you contact the remote site if the defaults cause problems.

5.1.9 Some applications don't work in GNOME on Wayland

GNOME in buster has changed its default display server from Xorg to Wayland (see Section 2.2.11). Some applications, including the popular package manager `synaptic`, the default Simplified Chinese input method, `fcitx`, and most screen recording applications, have not been updated to work properly under Wayland. In order to use these packages, one needs to log in with a GNOME on Xorg session.

5.1.10 Noteworthy obsolete packages

The following is a list of known and noteworthy obsolete packages (see Secció 4.8 for a description).

The list of obsolete packages includes:

- The package `mcelog` is no longer supported with kernel versions above 4.12. `rasdaemon` can be used as its replacement.
- The package `revelation`, which is used to store passwords, is not included in buster. `keepass2` can import previously exported password XML files from `revelation`. Please make sure you export your data from `revelation` before upgrading, to avoid losing access to your passwords.

- The package `phpmyadmin` is not included in buster.
- `ipsec-tools` and `raccoon` have been removed from buster as their source has been lagging behind in adapting to new threats.

Users are encouraged to migrate to `libreswan`, which has broader protocol compatibility and is being actively maintained upstream.

`libreswan` should be fully compatible in terms of communication protocols since it implements a superset of `raccoon`'s supported protocols.

- The simple MTA `ssmtp` has been dropped for buster. This is due to it currently not validating TLS certs; see [bug #662960](https://bugs.debian.org/662960) (<https://bugs.debian.org/662960>).
- The `ecryptfs-utils` package is not part of buster due to an unfixed serious bug ([#765854](https://bugs.debian.org/765854)) (<https://bugs.debian.org/765854>). At the time of writing this paragraph, there was no clear advice for users of `eCryptfs`, except not to upgrade.

5.1.11 Components desfasats a buster

With the next release of Debian 11 (codenamed bullseye) some features will be deprecated. Users will need to migrate to other alternatives to prevent trouble when updating to Debian 11.

This includes the following features:

- Python 2 will stop being supported by its upstream on **January 1, 2020** (<https://www.python.org/dev/peps/pep-0373/>). Debian hopes to drop `python-2.7` for Debian 11. If users have functionality that relies on **python**, they should prepare to migrate to **python3**.
- Icinga 1.x is EOL upstream since 2018-12-31; while the `icinga` package is still present, users should use the buster lifetime to migrate to Icinga 2 (`icinga2` package) and Icinga Web 2 (`icingaweb2` package). The `icinga2-classicui` package is still present to use the Icinga 1.x CGI web interface with Icinga 2, but the support for it will be removed in Icinga 2.11. Icinga Web 2 should be used instead.
- The Mailman mailing list manager suite version 3 is newly available in this release. Mailman has been split up into various components; the core is available in the package `mailman3` and the full suite can be obtained via the `mailman3-full` metapackage.

The legacy Mailman version 2.1 remains available in this release in the package `mailman`, so you can migrate any existing installations at your own pace. The Mailman 2.1 package will be kept in working order for the foreseeable future, but will not see any major changes or improvements. It will be removed from the first Debian release after Mailman upstream has stopped support for this branch.

Everyone is encouraged to upgrade to Mailman 3, the modern release under active development.

- The packages `spf-milter-python` and `dkim-milter-python` are no longer actively developed upstream, but their more feature-rich replacements, `pyspf-milter` and `dkimpy-milter`, are available in buster. Users should migrate to the new packages before the old ones are removed in bullseye.

5.1.12 Things to do post upgrade before rebooting

When `apt full-upgrade` has finished, the “formal” upgrade is complete. For the upgrade to buster, there are no special actions needed before performing a reboot.

5.1.13 SysV init related packages no longer required

NOTA

This section does not apply if you have decided to stick with sysvinit-core.

After the switch to systemd as default init system in Jessie and further refinements in Stretch, various SysV related packages are no longer required and can now be purged safely via

```
apt purge initscripts sysv-rc insserv startpar
```

5.2 Limitaciones en el suport de seguretat

There are some packages where Debian cannot promise to provide minimal backports for security issues. These are covered in the following subsections.

NOTA

The package `debian-security-support` helps to track the security support status of installed packages.

5.2.1 Estat de seguretat als navegadors web i els seus motors de renderització

Debian 10 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 engines¹ are included in buster, but not covered by security support. These browsers should not be used against untrusted websites. The `webkit2gtk` source package is covered by security support.

For general web browser use we recommend Firefox or Chromium. They will be kept up-to-date by rebuilding the current ESR releases for stable. The same strategy will be applied for Thunderbird.

5.2.2 Paquets basats en el llenguatge Go

The Debian infrastructure currently doesn't properly enable rebuilding packages that statically link parts of other packages on a large scale. Until buster that hasn't been a problem in practice, but with the growth of the Go ecosystem it means that Go based packages won't be covered by regular security support until the infrastructure is improved to deal with them maintainably.

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

5.3 Problems específics de paquets

In most cases, packages should upgrade smoothly between stretch and buster. There are a small number of cases where some intervention may be required, either before or during the upgrade; these are detailed below on a per-package basis.

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`.

5.3.1 Glibc requires Linux kernel 3.2 or higher

Starting with `glibc` 2.26, Linux kernel 3.2 or later is required. To avoid completely breaking the system, the preinst for `libc6` performs a check. If this fails, it will abort the package installation, which will leave the upgrade unfinished. If the system is running a kernel older than 3.2, please update it before starting the distribution upgrade.

5.3.2 Semantics for using environment variables for `su` changed

`su` has changed semantics in buster and no longer preserves the user environment variables `DISPLAY` and `XAUTHORITY`. If you need to run graphical applications with `su`, you will have to explicitly set them to allow access to your display. See [bug #905409](https://bugs.debian.org/905409) (<https://bugs.debian.org/905409>) for an extensive discussion.

5.3.3 Existing PostgreSQL databases need to be reindexed

When upgrading from stretch to buster, the `glibc` locale data is upgraded. Specifically, this changes how PostgreSQL sorts data in text indexes. To avoid corruption, such indexes need to be `REINDEXED` immediately after upgrading the `locales` or `locales-all` packages, before putting the database back into production.

Ordre suggerida:

```
sudo -u postgres reindexdb --all
```

Alternatively, upgrade the databases to PostgreSQL 11 using `pg_upgradecluster`. (This uses `pg_dump` by default which will rebuild all indexes. Using `-m upgrade` or `pg_upgrade` is *not* safe because it preserves the now-wrong index ordering.)

Refer to the [PostgreSQL Wiki](https://wiki.postgresql.org/wiki/Locale_data_changes) (https://wiki.postgresql.org/wiki/Locale_data_changes) for more information.

5.3.4 `mutt` i `neomutt`

In stretch, the package `mutt` had patches applied from the sources at <https://neomutt.org> (<https://neomutt.org>). Starting from buster, the package providing `/usr/bin/mutt` will instead be purely based on the original sources from <http://www.mutt.org> (<http://www.mutt.org>), and a separate `neomutt` package is available providing `/usr/bin/neomutt`.

This means that some of the features that were previously provided by `mutt` are no longer available. If this breaks your configuration you can install `neomutt` instead.

5.3.5 Accessing GNOME Settings app without mouse

Without a pointing device, there is no direct way to change settings in the GNOME Settings app provided by `gnome-control-center`. As a work-around, you can navigate from the sidebar to the main content by pressing the **Right Arrow** twice. To get back to the sidebar, you can start a search with `Ctrl+F`, type something, then hit `Esc` to cancel the search. Now you can use the **Up Arrow** and **Down Arrow** to navigate the sidebar. It is not possible to select search results with the keyboard.

5.3.6 `gnome-disk-utility` fails to change LUKS password causing permanent data loss (buster 10.0 only)

Users of the initial buster release images should not change the LUKS password of encrypted disks with the GNOME graphical interface for disk management. The `gnome-disk-utility` package in buster had a very nasty [bug \(#928893\)](https://bugs.debian.org/928893) (<https://bugs.debian.org/928893>) when used to change the LUKS password: it deleted the old password but failed to correctly set the new one, making all data on the disk inaccessible. This has been fixed in the first point release.

5.3.7 evolution-ews has been dropped, and email inboxes using Exchange, Office365 or Outlook server will be removed

Users using `evolution` as their email client and connecting to a server running Exchange, Office365 or Outlook using the `evolution-ews` plugin should not upgrade to buster without backing up data and finding an alternative solution beforehand, as `evolution-ews` has been dropped due to [bug #926712](https://bugs.debian.org/926712) (<https://bugs.debian.org/926712>) and their email inboxes, calendar, contact lists and tasks will be removed and will no longer be accessible with Evolution.

The `evolution-ews` package has been reintroduced via buster-backports. Users upgrading from stretch to buster can enable buster-backports after the upgrade and then they will be able to reinstall `evolution-ews`.

5.3.8 Calamares installer leaves disk encryption keys readable

When installing Debian from live media using the Calamares installer (Section 2.2.13) and selecting the full disk encryption feature, the disk's unlock key is stored in the `initramfs` which is world readable. This allows users with local filesystem access to read the private key and gain access to the filesystem again in the future.

This can be worked around by adding `UMASK=0077` to `/etc/initramfs-tools/conf.d/initramfs-permi` and running `update-initramfs -u`. This will recreate the `initramfs` without world-readable permissions.

A fix for the installer is being planned (see [bug #931373](https://bugs.debian.org/931373) (<https://bugs.debian.org/931373>)) and will be uploaded to `debian-security`. In the meantime users of full disk encryption should apply the above workaround.

5.3.9 S3QL URL changes for Amazon S3 buckets

When using `s3ql` with Amazon S3 buckets, the configuration needs updating for a change in the URL. The new format is:

```
s3://<region>/<bucket>/<prefix>
```

5.3.10 Split in configuration for logrotate

The shipped configurations for `/var/log/btmp` and `/var/log/wtmp` have been split from the main configuration file (`/etc/logrotate.conf`) into separate standalone files (`/etc/logrotate.d/btmp` and `/etc/logrotate.d/wtmp`).

If you have modified `/etc/logrotate.conf` in this regard, make sure to re-adjust the two new files to your needs and drop any references to `(b|w)tmp` from the main file, since duplicate definitions can cause errors.

Capítol 6

Més informació sobre Debian

6.1 Llegir més

Més enllà d'estes notes de llançament i de la guia d'instal·lació, hi ha més documentació de Debian a la vostra disposició al Projecte de Documentació de Debian (DDP), que té com a objectiu crear documentació de gran qualitat pels usuaris i desenvolupadors de Debian. Esta documentació inclou la guia de referència de Debian, la guia del nou mantenidor, les PMF de Debian i molt més. Per obtenir informació detallada dels recursos al vostre abast, visiteu el [lloc web del projecte de documentació de Debian](https://www.debian.org/doc/) (<https://www.debian.org/doc/>) i el lloc [Viqui de Debian](https://wiki.debian.org/) (<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 de estes llistes i detalls de com subscriure's, vegeu <https://lists.debian.org/>. Abans d'enviar la vostra pregunta, proveueu 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`.

Seguiu les normes del canal, respectant completament els altres usuaris. Les normes les podeu trobar al [Viqui de Debian](https://wiki.debian.org/DebianIRC) (<https://wiki.debian.org/DebianIRC>).

Per obtenir més informació d'OFTC visiteu la pàgina web del [lloc web](http://www.oftc.net/) (<http://www.oftc.net/>).

6.3 Informes d'error

Ens esforcem per fer de Debian un sistema operatiu de gran qualitat, però això no vol dir que els paquets que subministrem estiguen totalment 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 pugui 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

No necessiteu ser experts per col·laborar amb Debian. Ajudant els usuaris en problemes a les diferents **l·listes** (<https://lists.debian.org/>) de suport pels usuaris, ja esteu contribuint a la comunitat. Identificant (i també resolent) els problemes associats al desenvolupament de la distribució mitjançant la participació a les **l·listes** (<https://lists.debian.org/>) de desenvolupament també és de molta ajuda. Per mantenir l'alta qualitat de la distribució Debian, **envieu els informes d'errors** (<https://bugs.debian.org/>) i ajudeu els desenvolupadors a fer el seu seguiment per corregir-los. Si teniu traça amb les paraules aleshores potser voldreu col·laborant de forma més activa ajudant a escriure **documentació** (<https://www.debian.org/doc/vcs>) o **traduint** (<https://www.debian.org/international/>) la que ja existeix en el vostre idioma.

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, **barreges 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 conèixer 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 de configuració de màquina dinàmica

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

Lightweight Directory Access Protocol (Protocol lleuger d'accés a directori)

LSB

Linux Standard Base

LVM

Logical Volume Manager (Gestor de volums lògics)

MTA

Mail Transport Agent (Agent de transport de correu)

NBD

Network Block Device (Dispositiu de block en xarxa)

NFS

Network File System (Sistema de fitxers en xarxa)

NIC

Network Interface Card (Targeta d'interfície de xarxa)

NIS

Network Information Service (Servei d'informació de xarxa)

PHP

PHP: Processador d'hipertext

RAID

Redundant Array of Independent Disks (Matriu redundant de discs independents)

SATA

Serial Advanced Technology Attachment (Tecnologia avançada d'accessoris en serie)

SSL

Secure Sockets Layer (Capa de connexió segura)

TLS

Transport Layer Security (Capa de seguretat de transport)

UEFI

Unified Extensible Firmware Interface (Interfície unificada i extensible de microprogramari)

USB

Universal Serial Bus (Bus sèrie universal)

UUID

Universally Unique Identifier (Identificador únic universal)

WPA

Wi-Fi Protected Access (Accés protegit a Wi-Fi)

Apèndix A

Gestió del vostre sistema stretch abans de l'actualització

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

A.1 Actualització del vostre sistema stretch

Bàsicament no hi ha diferències entre qualsevol altra actualització d'stretch que ja hàgiu fet. L'única diferència és que necessitareu primer assegurar-vos que la vostra llista de paquets encara conté referències a stretch 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 d'stretch.

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/buster//buster/apt/sources.list.5.html>)) contain references to “stable”, this is effectively pointing to buster 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 buster, there probably is not much point in installing packages from stretch 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 `stretch`.

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

Before upgrading your system to buster, it is recommended to remove old configuration files (such as `*.dpkg-{new,old}` files under `/etc`) from the system.

A.4 Actualitza la configuració local antiga a UTF-8

Using a legacy non-UTF-8 locale has been unsupported by desktops and other mainstream software projects for a long time. Such locales should be upgraded by running **`dpkg-reconfigure locales`** and selecting a UTF-8 default. You should also ensure that users are not overriding the default to use a legacy locale in their environment.

Apèndix B

Contribuidors 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, Jonathan, 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, 3

Cryptsetup, 4

D

DocBook XML, 2

Dovecot, 4

E

Evolution, 4

Exim, 4

G

GCC, 4

GIMP, 4

GNOME, 3

GNUCash, 3

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

apparmor, 4, 5

apparmor-profiles-extra, 5

apt, 2, 16

apt-listchanges, 20

aptitude, 13, 19, 23

cryptsetup, 6

cups, 6

cups-browsed, 6

cups-filters, 6

dblatex, 2

debian-goodies, 19

debian-kernel-handbook, 22

debian-security-support, 29

dkim-milter-python, 28

dkimpy-milter, 28

doc-debian, 34

docbook-xsl, 2

ecryptfs-utils, 28

evince, 4

evolution, 31

evolution-ews, 31

fcitx, 27

gnome-control-center, 30

gnome-disk-utility, 30

grub-efi-amd64-signed, 4

grub-efi-ia32-signed, 4

haveged, 26

icinga, 28

icinga2, 28

icinga2-classicui, 28

icingaweb2, 28

ifupdown, 26

initramfs-tools, 12, 22

ipsec-tools, 28

iptables, 5

keepass2, 27

libc6, 30

libreswan, 28

linux-image-*, 22

linux-image-amd64, 22

linux-source, 22

localepurge, 19

locales, 30

locales-all, 30

mailman, 28

mailman3, 28

mailman3-full, 28

manpages-de, 5

mcelog, 27

mutt, 5, 30

neomutt, 30

phpmyadmin, 28

popularity-contest, 19

pyspf-milter, 28

python-2.7, 28

raccoon, 28

rasdaemon, 27

release-notes, 1

revelation, 27

rpcbind, 25

s3ql, 31

shim-signed, 4

spf-milter-python, 28

sshd, 25

ssmtp, 28

synaptic, 13, 27
systemd, 5, 26, 27
tinc, 13
udev, 22, 26
unattended-upgrades, 5
upgrade-reports, 1
usrmerge, 7
util-linux, 5
xmlroff, 2
xsltproc, 2
ypbind, 25

Perl, 4
PHP, 4
Postfix, 4
PostgreSQL, 4

X
Xfce, 3