

Installation

4

Contents

Installation Methods 52

Installing, Step by Step 54

After the First Boot 73

To use Debian, you need to install it on a computer; this task is taken care of by the debian-installer program. A proper installation involves many operations. This chapter reviews them in their chronological order.

Installing a computer is always simpler when you are familiar with the way it works. If you are not, make a quick detour to appendix B, “[Short Remedial Course](#)” page 475 before reading this chapter.

The installer for *Buster* is based on `debian-installer`. Its modular design enables it to work in various scenarios and allows it to evolve and adapt to changes. Despite the limitations implied by the need to support a large number of architectures, this installer is very accessible to beginners, since it assists users at each stage of the process. Automatic hardware detection, guided partitioning, and graphical user interfaces have solved most of the problems that newbies used to face in the early years of Debian.

Installation requires 128 MB of RAM (Random Access Memory) and at least 2 GB of hard drive space. All Falcot computers meet these criteria. Note, however, that these figures apply to the installation of a very limited system without a graphical desktop. A minimum of 1 GB of RAM and 10 GB of hard drive space are really recommended for a basic office desktop workstation.

If you already have Debian *Stretch* installed on your computer, this chapter is not for you! Unlike other distributions, Debian allows updating a system from one version to the next without having to reinstall the system. Reinstalling, in addition to being unnecessary, could even be dangerous, since it could remove already installed programs.

The upgrade process will be described in section 6.7, “[Upgrading from One Stable Distribution to the Next](#)” page 134.

4.1. Installation Methods

A Debian system can be installed from several types of media, as long as the BIOS of the machine allows it. You can for instance boot with a CD-ROM, a USB key, or even through a network.

BIOS (which stands for Basic Input/Output System) is a software that is included in the motherboard (the electronic board connecting all peripherals) and executed when the computer is booted, in order to load an operating system (via an adapted bootloader). It stays in the background to provide an interface between the hardware and the software (in our case, the Linux kernel).

4.1.1. Installing from a CD-ROM/DVD-ROM

The most widely used installation method is from a CD-ROM (or DVD-ROM, which behaves exactly the same way): the computer is booted from this media, and the installation program takes over.

Various CD-ROM families have different purposes: *netinst* (network installation) contains the installer and the base Debian system; all other programs are then downloaded. Its “image”, that

is the ISO-9660 filesystem that contains the exact contents of the disk, only takes up about 150 to 280 MB (depending on architecture). On the other hand, the complete set offers all packages and allows for installation on a computer that has no Internet access; it requires around 16 DVD-ROMs (or 4 Blu-ray disks). There is no more official CD-ROMs set as they were really huge, rarely used and now most of the computers use DVD-ROMs as well as CD-ROMs. But the programs are divided among the disks according to their popularity and importance; the first disk will be sufficient for most installations, since it contains the most used software.

There is a last type of image, known as `mini.iso`, which is only available as a by-product of the installer. The image only contains the minimum required to configure the network and everything else is downloaded (including parts of the installer itself, which is why those images tend to break when a new version of the installer is released). Those images can be found on the normal Debian mirrors under the `dists/release/main/installer-arch/current/images/netboot/` directory.

Multi-architecture disks

TIP

Most installation CD- and DVD-ROMs work only with a specific hardware architecture. If you wish to download the complete images, you must take care to choose those which work on the hardware of the computer on which you wish to install them.

Some CD/DVD-ROM images can work on several architectures. We thus have a CD-ROM image combining the *netinst* images of the *i386* and *amd64* architectures.

To acquire Debian CD-ROM images, you may, of course, download them and burn them to disk. You may also purchase them, and, thus, provide the project with a little financial support. Check the website to see the list of DVD-ROM image vendors and download sites.

➡ <https://www.debian.org/CD/index.html>

4.1.2. Booting from a USB Key

Since most computers are able to boot from USB devices, you can also install Debian from a USB key (this is nothing more than a small flash-memory disk).

The installation manual explains how to create a USB key that contains the `debian-installer`. The procedure is very simple because ISO images for *i386* and *amd64* are hybrid images that can boot from a CD-ROM as well as from a USB key.

You must first identify the device name of the USB key (ex: `/dev/sdb`); the simplest means to do this is to check the messages issued by the kernel using the `dmesg` command. Then you must copy the previously downloaded ISO image (for example, `debian-10.0.0-amd64-netinst.iso`) with the command `cat debian-10.0.0-amd64-netinst.iso >/dev/sdb; sync`. This command requires administrator rights, since it accesses the USB key directly and blindly erases its content.

A more detailed explanation is available in the installation manual. Among other things, it describes an alternative method of preparing a USB key that is more complex, but that allows to customize the installer's default options (those set in the kernel command line).

➡ <https://www.debian.org/releases/stable/amd64/ch04s03>

4.1.3. Installing through Network Booting

Many BIOSes allow booting directly from the network by downloading a kernel and a minimal filesystem image. This method (which has several names, such as PXE or TFTP boot) can be a life-saver if the computer does not have a CD-ROM reader, or if the BIOS can't boot from such media.

This installation method works in two steps. First, while booting the computer, the BIOS (or the network card) issues a BOOTP/DHCP request to automatically acquire an IP address. When a BOOTP or DHCP server returns a response, it includes a filename, as well as network settings. After having configured the network, the client computer then issues a TFTP (Trivial File Transfer Protocol) request for a file whose name was previously indicated. Once this file is acquired, it is executed as though it were a bootloader. This then launches the Debian installation program, which is executed as though it were running from the hard drive, a CD-ROM, or a USB key.

All the details of this method are available in the installation guide ("Preparing files for TFTP Net Booting" section).

➡ <https://www.debian.org/releases/stable/amd64/ch05s01#boot-tftp-x86>

➡ <https://www.debian.org/releases/stable/amd64/ch04s05>

4.1.4. Other Installation Methods

When we have to deploy customized installations for a large number of computers, we generally choose an automated rather than a manual installation method. Depending on the situation and the complexity of the installations to be made, we can use FAI (Fully Automatic Installer, described in section 12.3.1, "[Fully Automatic Installer \(FAI\)](#)" page 366), or even a customized installation DVD with preseeding (see section 12.3.2, "[Preseeding Debian-Installer](#)" page 367).

4.2. Installing, Step by Step

4.2.1. Booting and Starting the Installer

Once the BIOS has begun booting from the CD- or DVD-ROM, the Isolinux bootloader menu appears. At this stage, the Linux kernel is not yet loaded; this menu allows you to choose the kernel to boot and enter possible parameters to be transferred to it in the process.

For a standard installation, you only need to choose "Install" or "Graphical install" (with the arrow keys), then press the Enter key to initiate the remainder of the installation process. If the

DVD-ROM is a “Multi-arch” disk, and the machine has an Intel or AMD 64-bit processor, those menu options enable the installation of the 64-bit variant (*amd64*) and the installation of the 32-bit variant remains available in a dedicated sub-menu (“32-bit install options”). If you have a 32-bit processor, you don’t get a choice and the menu entries install the 32-bit variant (*i386*).

GOING FURTHER

32 or 64 bits?

The fundamental difference between 32- and 64-bit systems is the size of memory addresses. In theory, a 32-bit system can not work with more than 4 GB of RAM (2^{32} bytes). In practice, it is possible to work around this limitation by using the 686-pae kernel variant, so long as the processor handles the PAE (Physical Address Extension) functionality. Using it does have a notable influence on system performance, however. This is why it is useful to use the 64-bit mode on a server with a large amount of RAM.

For an office computer (where a few percent difference in performance is negligible), you must keep in mind that some proprietary programs are not available in 64-bit versions. It is technically possible to make them work on 64-bit systems, but you have to install the 32-bit versions of all the necessary libraries (see section 5.4.5, “Multi-Arch Support” page 101), and sometimes to use *setarch* or *linux32* (in the *util-linux* package) to trick applications regarding the nature of the system.

IN PRACTICE

Installation alongside an existing Windows system

If the computer is already running Windows, it is not necessary to delete the system in order to install Debian. You can have both systems at once, each installed on a separate disk or partition, and choose which to start when booting the computer. This configuration is often called “dual boot”, and the Debian installation system can set it up. This is done during the hard drive partitioning stage of installation and while setting up the bootloader (see the sidebars “Shrinking a Windows partition” page 65 and “Bootloader and dual boot” page 71).

If you already have a working Windows system, you can even avoid using a CD-ROM; Debian offers a Windows program that will download a light Debian installer and set it up on the hard disk. You then only need to reboot the computer and choose between normal Windows boot or booting the installation program. You can also find it on a dedicated website with a rather explicit title...

- ➡ <http://ftp.debian.org/debian/tools/win32-loader/stable/>
- ➡ <https://people.debian.org/~rmh/goodbye-microsoft/>

BACK TO BASICS

Bootloader

The bootloader is a low-level program that is responsible for booting the Linux kernel just after the BIOS passes off its control. To handle this task, it must be able to locate the Linux kernel to boot on the disk. On the *i386* and *amd64* architectures, the two most used programs to perform this task are LILO, the older of the two, and GRUB, its modern replacement. Isolinux and Syslinux are alternatives frequently used to boot from removable media.

Each menu entry hides a specific boot command line, which can be configured as needed by pressing the TAB key before validating the entry and booting. The “Help” menu entry displays the old command line interface, where the F1 to F10 keys display different help screens detailing the various options available at the prompt. You will rarely need to use this option except in very specific cases.

The “expert” mode (accessible in the “Advanced options” menu) details all possible options in the process of installation, and allows navigation between the various steps without them happening automatically in sequence. Be careful, this very verbose mode can be confusing due to the multitude of configuration choices that it offers.

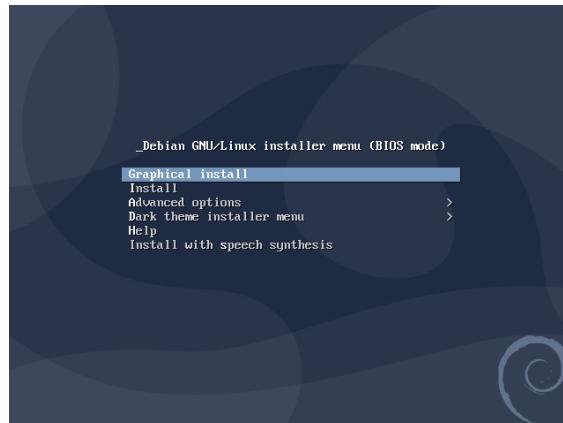


Figure 4.1 *Boot screen*

Once booted, the installation program guides you step by step throughout the process. This section presents each of these steps in detail. Here we follow the process of an installation from an amd64 DVD-ROM (more specifically, the rc1 version of the installer for *Buster*); *netinst* installations, as well as the final release of the installer, may look slightly different. We will also address installation in graphical mode, but the only difference from “classic” (text-mode) installation is in the visual appearance.

4.2.2. Selecting the language

The installation program begins in English, but the first step allows the user to choose the language that will be used in the rest of the process. Choosing French, for example, will provide an installation entirely translated into French (and a system configured in French as a result). This choice is also used to define more relevant default choices in subsequent stages (notably the keyboard layout).

BACK TO BASICS

Navigating with the keyboard

Some steps in the installation process require you to enter information. These screens have several areas that may “have focus” (text entry area, checkboxes, list of choices, OK and Cancel buttons), and the TAB key allows you to move from one to another.

In graphical mode, you can use the mouse as you would normally on an installed graphical desktop.

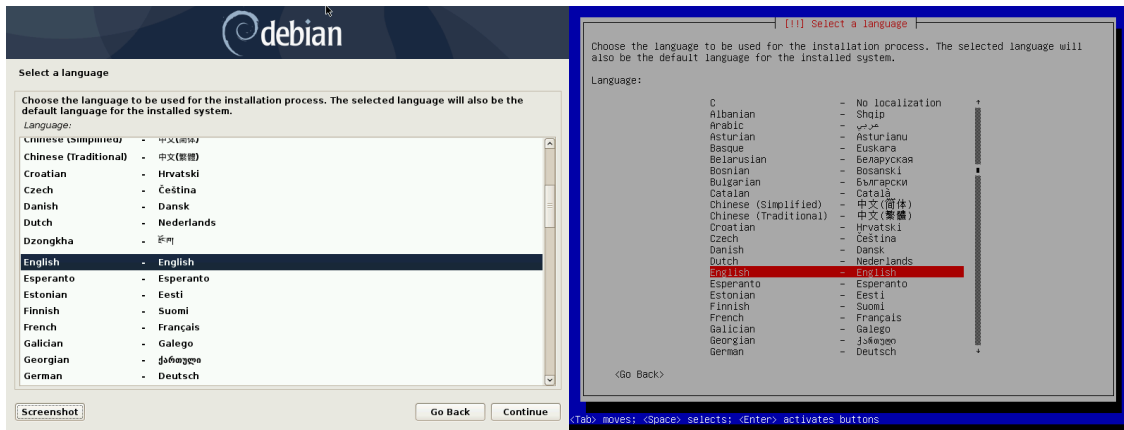


Figure 4.2 Selecting the language

4.2.3. Selecting the country

The second step consists in choosing your country. Combined with the language, this information enables the program to offer the most appropriate keyboard layout. This will also influence the configuration of the time zone. In the United States, a standard QWERTY keyboard is suggested, and a choice of appropriate time zones is offered.

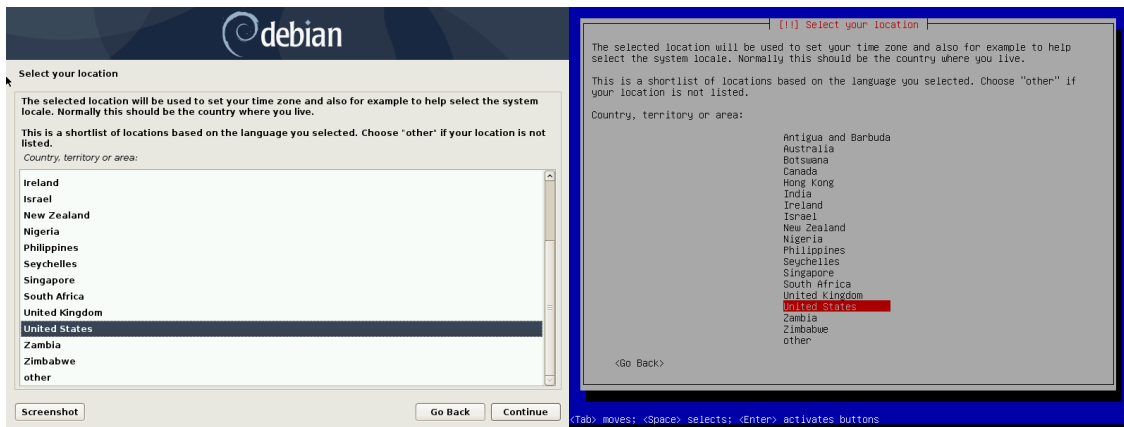


Figure 4.3 Selecting the country

4.2.4. Selecting the keyboard layout

The proposed “American English” keyboard corresponds to the usual QWERTY layout.

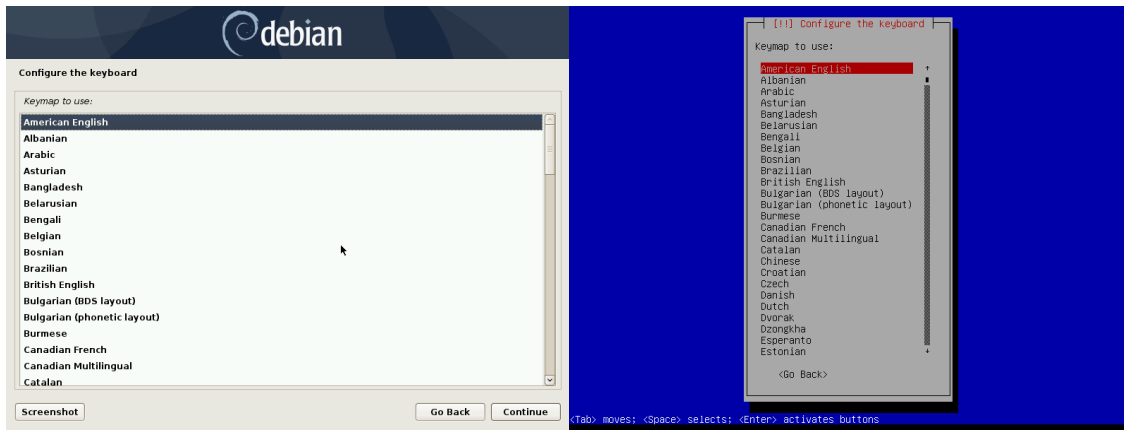


Figure 4.4 *Choice of keyboard*

4.2.5. Detecting Hardware

This step is completely automatic in the vast majority of cases. The installer detects your hardware, and tries to identify the CD-ROM drive used in order to access its content. It loads the modules corresponding to the various hardware components detected, and then “mounts” the CD-ROM in order to read it. The previous steps were completely contained in the boot image included on the CD, a file of limited size and loaded into memory by the BIOS when booting from the CD.

The installer can work with the vast majority of drives, especially standard ATAPI peripherals (sometimes called IDE and EIDE). However, if detection of the CD-ROM reader fails, the installer offers the choice to load a kernel module (for instance, from a USB key) corresponding to the CD-ROM driver.

4.2.6. Loading Components

With the contents of the CD now available, the installer loads all the files necessary to continue with its work. This includes additional drivers for the remaining hardware (especially the network card), as well as all the components of the installation program.

4.2.7. Detecting Network Hardware

This automatic step tries to identify the network card and load the corresponding module. If automatic detection fails, you can manually select the module to load. If no module works, it is possible to load a specific module from a removable device. This last solution is usually only needed if the appropriate driver is not included in the standard Linux kernel, but available elsewhere, such as the manufacturer’s website.

This step must absolutely be successful for *netinst* installations, since the Debian packages must be loaded from the network.

4.2.8. Configuring the Network

In order to automate the process as much as possible, the installer attempts an automatic network configuration by DHCP (for IPv4) and by IPv6 network discovery. If this fails, it offers more choices: try again with a normal DHCP configuration, attempt DHCP configuration by declaring the name of the machine, or set up a static network configuration.

This last option requires an IP address, a subnet mask, an IP address for a potential gateway, a machine name, and a domain name.

TIP
**Configuration without
DHCP**

If the local network is equipped with a DHCP server that you do not wish to use because you prefer to define a static IP address for the machine during installation, you can add the `netcfg/use_dhcp=false` option when booting from the CD-ROM. You just need to go to the desired menu entry by pressing the TAB key and add the desired option before pressing the Enter key.

BEWARE
Do not improvise

Many local area networks are based on an implicit assumption that all machines can be trusted, and inadequate configuration of a single computer will often perturb the whole network. As a result, do not connect your machine to a network without first agreeing with its administrator on the appropriate settings (for example, the IP address, netmask, and broadcast address).

4.2.9. Administrator Password

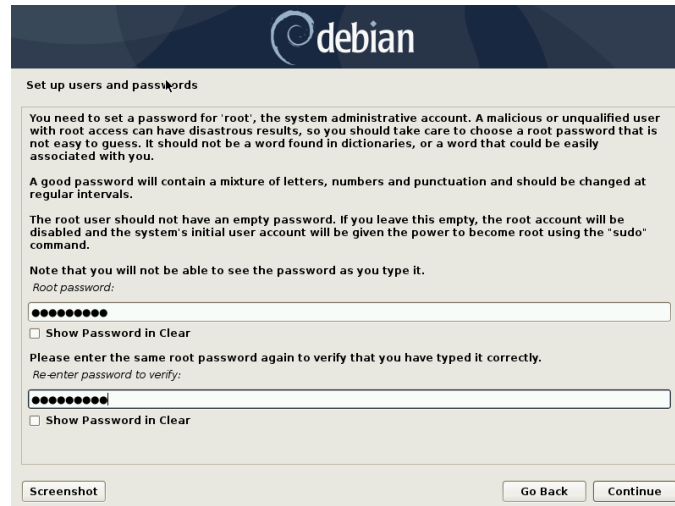
The super-user root account, reserved for the machine’s administrator, is automatically created during installation; this is why a password is requested. The installer also asks for a confirmation of the password to prevent any input error which would later be difficult to amend. Note that you can leave both fields empty if you want the root account to be disabled. In that case, the first regular user — that will be created by the installer in the next step — will have administrative rights through `sudo` (see section 8.9.4, “[Sharing Administrator Rights](#)” page 186).

SECURITY
Administrator password

The root user’s password should be long (12 characters or more) and impossible to guess. Indeed, any computer (and a fortiori any server) connected to the Internet is regularly targeted by automated connection attempts with the most obvious passwords. Sometimes it may even be subject to dictionary attacks, in which many combinations of words and numbers are tested as password. Avoid using the names of children or parents, dates of birth, etc.: many of your co-workers might know them, and you rarely want to give them free access to the computer in question.

These remarks are equally applicable for other user passwords, but the consequences of a compromised account are less drastic for users without administrative rights.

If inspiration is lacking, do not hesitate to use password generators, such as pwgen (in the package of the same name).



Set up users and passwords

You need to set a password for 'root', the system administrative account. A malicious or unqualified user with root access can have disastrous results, so you should take care to choose a root password that is not easy to guess. It should not be a word found in dictionaries, or a word that could be easily associated with you.

A good password will contain a mixture of letters, numbers and punctuation and should be changed at regular intervals.

The root user should not have an empty password. If you leave this empty, the root account will be disabled and the system's initial user account will be given the power to become root using the "sudo" command.

Note that you will not be able to see the password as you type it.

Root password:

••••••••

☐ Show Password in Clear

Please enter the same root password again to verify that you have typed it correctly.

Re-enter password to verify:

••••••••

☐ Show Password in Clear

[Screenshot](#) [Go Back](#) [Continue](#)

Figure 4.5 Administrator Password

4.2.10. Creating the First User

Debian also imposes the creation of a standard user account so that the administrator doesn't get into the bad habit of working as root. The precautionary principle essentially means that each task is performed with the minimum required rights, in order to limit the damage caused by human error. This is why the installer will ask for the complete name of this first user, their username, and their password (twice, to prevent the risk of erroneous input).



Figure 4.6 *Name of the first user*

4.2.11. Configuring the Clock

If the network is available, the system’s internal clock is updated (in a one-shot way) from an NTP server. This way the timestamps on logs will be correct from the first boot. For them to remain consistently precise over time, an NTP daemon needs to be set up after initial installation (see section 8.9.2, “**Time Synchronization**” page 184).

4.2.12. Detecting Disks and Other Devices

This step automatically detects the hard drives on which Debian may be installed. They will be presented in the next step: partitioning.

4.2.13. Starting the Partitioning Tool

CULTURE **Uses of partitioning**

Partitioning, an indispensable step in installation, consists in dividing the available space on the hard drives (each subdivision thereof being called a “partition”) according to the data to be stored on it and the use for which the computer is intended. This step also includes choosing the filesystems to be used. All of these decisions will have an influence on performance, data security, and the administration of the server.

The partitioning step is traditionally difficult for new users. It is necessary to define the various portions of the disks (or “partitions”) on which the Linux filesystems and virtual memory (swap) will be stored. This task is complicated if another operating system that you want to

keep is already on the machine. Indeed, you will then have to make sure that you do not alter its partitions (or that you resize them without causing damage).

Fortunately, the partitioning software has a “guided” mode which recommends partitions for the user to make — in most cases, you can simply validate the software’s suggestions.

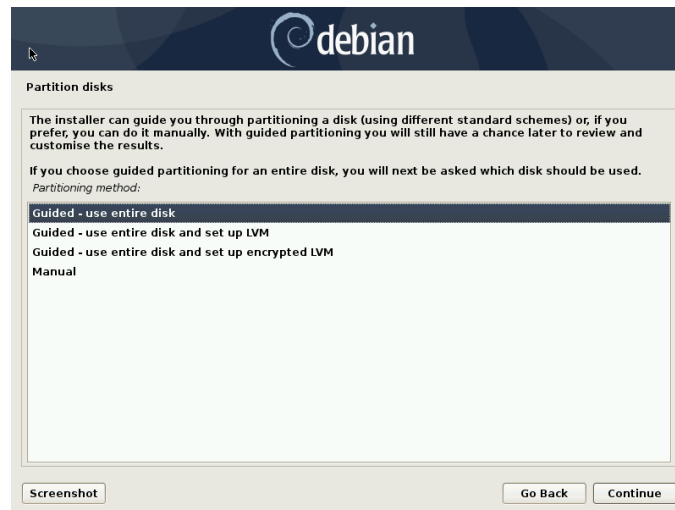


Figure 4.7 *Choice of partitioning mode*

The first screen in the partitioning tool offers the choice of using an entire hard drive to create various partitions. For a (new) computer which will solely use Linux, this option is clearly the simplest, and you can choose the option “Guided - use entire disk”. If the computer has two hard drives for two operating systems, setting one drive for each is also a solution that can facilitate partitioning. In both of these cases, the next screen offers to choose the disk where Linux will be installed by selecting the corresponding entry (for example, “SCSI1 (0,0,0) (sda) - 21.5 GB ATA QEMU HARDDISK”). You then start guided partitioning.

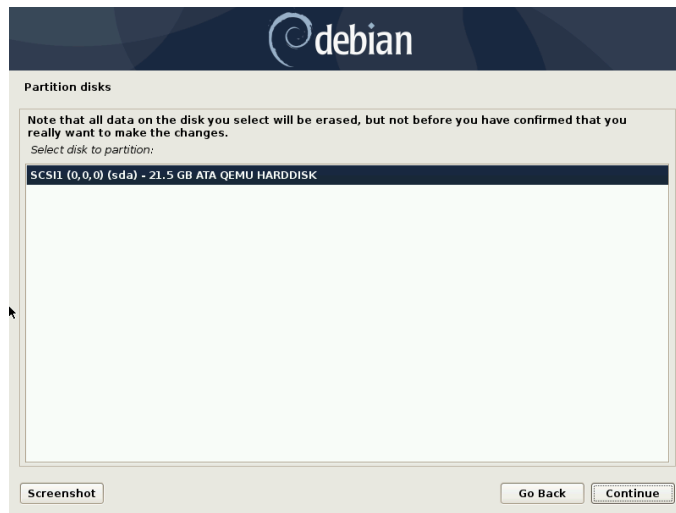


Figure 4.8 *Disk to use for guided partitioning*

Guided partitioning can also set up LVM logical volumes instead of partitions (see below). Since the remainder of the operation is the same, we will not go over the option “Guided - use entire disk and set up LVM” (encrypted or not).

In other cases, when Linux must work alongside other already existing partitions, you need to choose manual partitioning.

Guided partitioning

The guided partitioning tool offers three partitioning methods, which correspond to different usages.

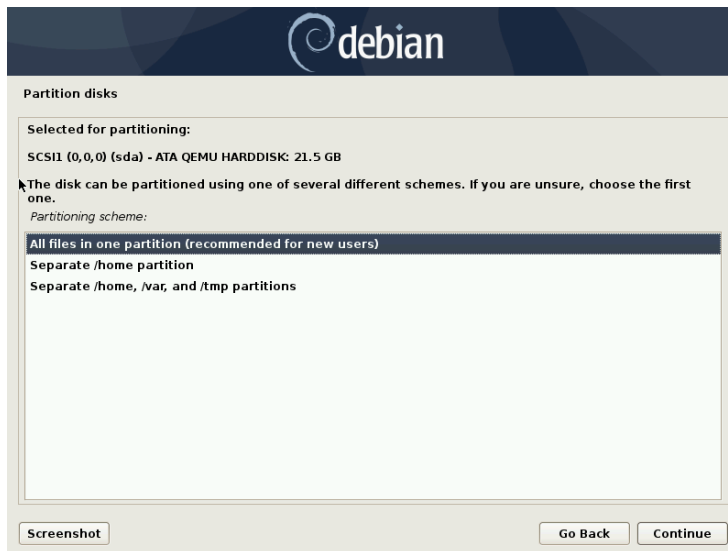


Figure 4.9 Guided partitioning

The first method is called “All files in one partition”. The entire Linux system tree is stored in a single filesystem, corresponding to the root `/` directory. This simple and robust partitioning fits perfectly for personal or single-user systems. In fact, two partitions will be created: the first will house the complete system, the second the virtual memory (swap).

The second method, “Separate `/home` partition”, is similar, but splits the file hierarchy in two: one partition contains the Linux system (`/`), and the second contains “home directories” (meaning user data, in files and subdirectories available under `/home/`).

The last partitioning method, called “Separate `/home`, `/var`, and `/tmp` partitions”, is appropriate for servers and multi-user systems. It divides the file tree into many partitions: in addition to the root (`/`) and user accounts (`/home/`) partitions, it also has partitions for server software data (`/var/`), and temporary files (`/tmp/`). These divisions have several advantages. Users can not lock up the server by consuming all available hard drive space (they can only fill up `/tmp/` and `/home/`). The daemon data (especially logs) can no longer clog up the rest of the system.

BACK TO BASICS

Choosing a filesystem

A filesystem defines the way in which data is organized on the hard drive. Each existing filesystem has its merits and limitations. Some are more robust, others more effective: if you know your needs well, choosing the most appropriate filesystem is possible. Various comparisons have already been made; it seems that *ReiserFS* is particularly efficient for reading many small files; *XFS*, in turn, works faster with large files. *Ext4*, the default filesystem for Debian, is a good compromise, based on the three previous versions of filesystems historically used in Linux (*ext*, *ext2* and *ext3*). *Ext4* overcomes certain limitations of *ext3* and is particularly appropriate for very large capacity hard drives. Another option would be to experiment with the very promising *btrfs*, which includes numerous features that require, to this day, the use of LVM and/or RAID.

A journaled filesystem (such as *ext3*, *ext4*, *btrfs*, *reiserfs*, or *xfs*) takes special measures to make it possible to return to a prior consistent state after an abrupt interruption without completely analyzing the entire disk (as was the case with the *ext2* system). This functionality is carried out by filling in a journal that describes the operations to conduct prior to actually executing them. If an operation is interrupted, it will be possible to “replay” it from the journal. Conversely, if an interruption occurs during an update of the journal, the last requested change is simply ignored; the data being written could be lost, but since the data on the disk has not changed, they have remained coherent. This is nothing more nor less than a transactional mechanism applied to the filesystem.

After choosing the type of partition, the software calculates a suggestion, and describes it on the screen; the user can then modify it if needed. You can, in particular, choose another filesystem if the standard choice (*ext4*) isn’t appropriate. In most cases, however, the proposed partitioning is reasonable and it can be accepted by selecting the “Finish partitioning and write changes to disk” entry.

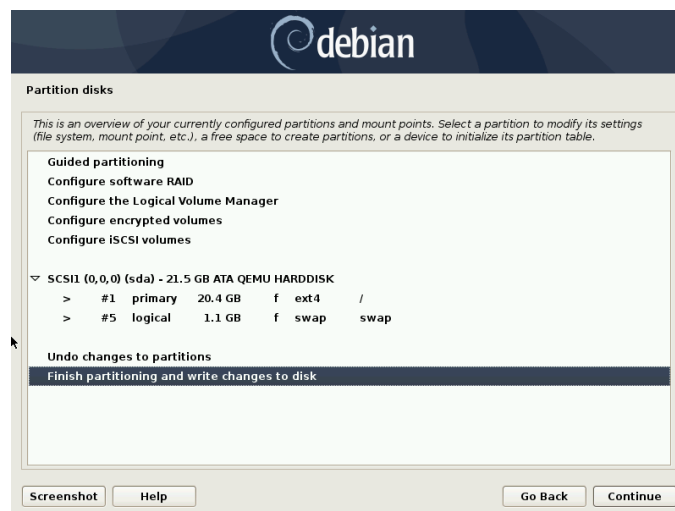


Figure 4.10 Validating partitioning

Manual Partitioning

Manual partitioning allows greater flexibility, allowing the user to choose the purpose and size of each partition. Furthermore, this mode is unavoidable if you wish to use software RAID.

IN PRACTICE

Shrinking a Windows partition

To install Debian alongside an existing operating system (Windows or other), you must have some available hard drive space that is not being used by the other system in order to be able to create the partitions dedicated to Debian. In most cases, this means shrinking a Windows partition and reusing the freed space.

The Debian installer allows this operation when using the manual mode for partitioning. You only need to choose the Windows partition and enter its new size (this works the same with both unencrypted FAT and NTFS partitions).

If Windows is using BitLocker-encrypted partitions, the steps to resize them requires to use the BitLocker Management together with the Windows Disk Management tool.

The first screen displays the available disks, their partitions, and any possible free space that has not yet been partitioned. You can select each displayed element; pressing the Enter key then gives a list of possible actions.

You can erase all partitions on a disk by selecting it.

When selecting free space on a disk, you can manually create a new partition. You can also do this with guided partitioning, which is an interesting solution for a disk that already contains another operating system, but which you may wish to partition for Linux in a standard manner. See section 4.2.13.1, “[Guided partitioning](#)” page 63 for more details on guided partitioning.

BACK TO BASICS

Mount point

The mount point is the directory tree that will house the contents of the filesystem on the selected partition. Thus, a partition mounted at `/home/` is traditionally intended to contain user data.

When this directory is named “`/`”, it is known as the *root* of the file tree, and therefore the root of the partition that will actually host the Debian system.

BACK TO BASICS

Virtual memory, swap

Virtual memory allows the Linux kernel, when lacking sufficient memory (RAM), to free a bit of memory by storing the parts of the RAM that have been inactive for some time on the swap partition of the hard disk.

To simulate the additional memory, Windows uses a swap file that is directly contained in a filesystem. Conversely, Linux uses a partition dedicated to this purpose, hence the term “swap partition”.

When choosing a partition, you can indicate the manner in which you are going to use it:

- format it and include it in the file tree by choosing a mount point;
- use it as a swap partition;
- make it into a “physical volume for encryption” (to protect the confidentiality of data on certain partitions, see below);
- make it a “physical volume for LVM” (this concept is discussed in greater detail later in this chapter);
- use it as a RAID device (see later in this chapter);
- you can also choose not to use it, and therefore leave it unchanged.

Configuring Multidisk Devices (Software RAID)

Some types of RAID allow the duplication of information stored on hard drives to prevent data loss in the event of a hardware problem affecting one of them. Level 1 RAID keeps a simple, identical copy (mirror) of a hard drive on another drive, while level 5 RAID splits redundant data over several disks, thus allowing the complete reconstruction of a failing drive.

We will only describe level 1 RAID, which is the simplest to implement. The first step involves creating two partitions of identical size located on two different hard drives, and to label them “physical volume for RAID”.

You must then choose “Configure software RAID” in the partitioning tool to combine these two partitions into a new virtual disk and select “Create MD device” in the configuration screen. You then need to answer a series of questions about this new device. The first question asks about the RAID level to use, which in our case will be “RAID1”. The second question asks about the number of active devices — two in our case, which is the number of partitions that need to be included in this MD device. The third question is about the number of spare devices — 0; we have not planned any additional disk to take over for a possible defective disk. The last question requires you to choose the partitions for the RAID device — these would be the two that we have set aside for this purpose (make sure you only select the partitions that explicitly mention “raid”).

Back to the main menu, a new virtual “RAID” disk appears. This disk is presented with a single partition which can not be deleted, but whose use we can choose (just like for any other partition).

For further details on RAID functions, please refer to section 12.1.1, “**Software RAID**” page 328.

Configuring the Logical Volume Manager (LVM)

LVM allows you to create “virtual” partitions that span over several disks. The benefits are twofold: the size of the partitions are no longer limited by individual disks but by their cumulative volume, and you can resize existing partitions at any time, possibly after adding an additional disk when needed.

LVM uses a particular terminology: a virtual partition is a “logical volume”, which is part of a “volume group”, or an association of several “physical volumes”. Each of these terms in fact corresponds to a “real” partition (or a software RAID device).

This technique works in a very simple way: each volume, whether physical or logical, is split into blocks of the same size, which are made to correspond by LVM. The addition of a new disk will cause the creation of a new physical volume, and these new blocks can be associated to any volume group. All of the partitions in the volume group that is thus expanded will have additional space into which they can extend.

The partitioning tool configures LVM in several steps. First you must create on the existing disks the partitions that will be “physical volumes for LVM”. To activate LVM, you need to choose “Configure the Logical Volume Manager (LVM)”, then on the same configuration screen “Create

a volume group”, to which you will associate the existing physical volumes. Finally, you can create logical volumes within this volume group. Note that the automatic partitioning system can perform all these steps automatically.

In the partitioning menu, each physical volume will appear as a disk with a single partition which can not be deleted, but that you can use as desired.

The usage of LVM is described in further detail in section 12.1.2, “LVM” page 339.

Setting Up Encrypted Partitions

To guarantee the confidentiality of your data, for instance in the event of the loss or theft of your computer or a hard drive, it is possible to encrypt the data on some partitions. This feature can be added underneath any filesystem, since, as for LVM, Linux (and more particularly the dm-crypt driver) uses the Device Mapper to create a virtual partition (whose content is protected) based on an underlying partition that will store the data in an encrypted form (thanks to LUKS, Linux Unified Key Setup, a standard format that enables the storage of encrypted data as well as meta-information that indicates the encryption algorithms used).

SECURITY

Encrypted swap partition

When an encrypted partition is used, the encryption key is stored in memory (RAM). Since retrieving this key allows the decryption of the data, it is of utmost importance to avoid leaving a copy of this key that would be accessible to the possible thief of the computer or hard drive, or to a maintenance technician. This is, however, something that can easily occur with a laptop, since when hibernating the contents of RAM is stored on the swap partition. If this partition isn’t encrypted, the thief may access the key and use it to decrypt the data from the encrypted partitions. This is why, when you use encrypted partitions, it is imperative to also encrypt the swap partition.

The Debian installer will warn the user if they try to make an encrypted partition while the swap partition isn’t encrypted.

To create an encrypted partition, you must first assign an available partition for this purpose. To do so, select a partition and indicate that it is to be used as a “physical volume for encryption”. After partitioning the disk containing the physical volume to be made, choose “Configure encrypted volumes”. The software will then propose to initialize the physical volume with random data (making the localization of the real data more difficult), and will ask you to enter an “encryption passphrase”, which you will have to enter every time you boot your computer in order to access the content of the encrypted partition. Once this step has been completed, and you have returned to the partitioning tool menu, a new partition will be available in an “encrypted volume”, which you can then configure just like any other partition. In most cases, this partition is used as a physical volume for LVM so as to protect several partitions (LVM logical volumes) with the same encryption key, including the swap partition (see sidebar “**Encrypted swap partition**” page 68).

4.2.14. Installing the Base System

This step, which doesn't require any user interaction, installs the Debian “base system” packages. This includes the `dpkg` and `apt` tools, which manage Debian packages, as well as the utilities necessary to boot the system and start using it.

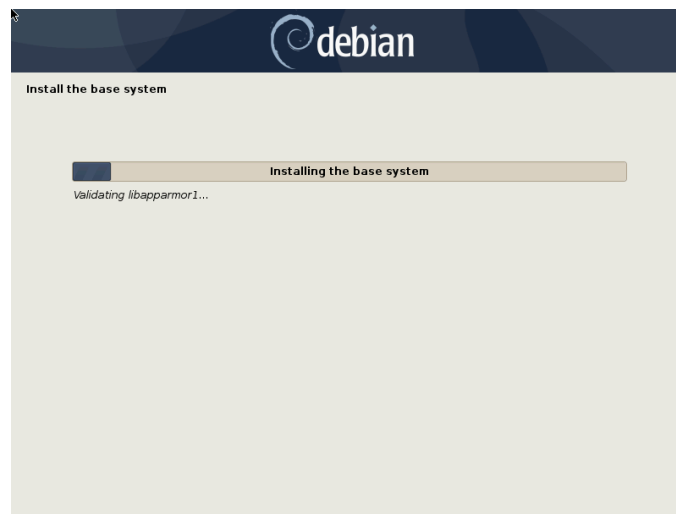


Figure 4.11 *Installation of the base system*

4.2.15. Configuring the Package Manager (apt)

In order to be able to install additional software, APT needs to be configured and told where to find Debian packages. This step is as automated as possible. It starts with a question asking if it must use a network source for packages, or if it should only look for packages on the CD-ROM.

NOTE
Debian CD-ROM in the drive

If the installer detects a Debian installation disk in the CD/DVD reader, it is not necessary to configure APT to go looking for packages on the network: APT is automatically configured to read packages from a removable media drive. If the disk is part of a set, the software will offer to “explore” other disks in order to reference all of the packages stored on them.

If getting packages from the network is requested, the next two questions allow to choose a server from which to download these packages, by choosing first a country, then a mirror available in that country (a mirror is a public server hosting copies of all the files of the Debian master archive).

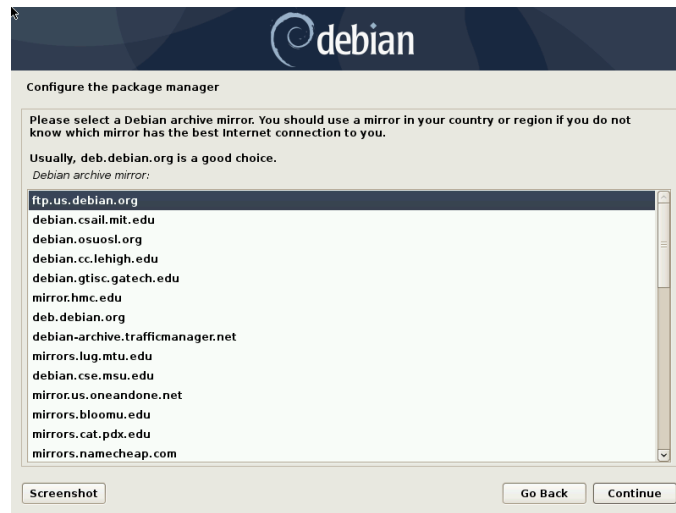


Figure 4.12 *Selecting a Debian mirror*

Finally, the program proposes to use an HTTP proxy. If there is no proxy, Internet access will be direct. If you type `http://proxy.falcot.com:3128`, APT will use the Falcot *proxy/cache*, a “Squid” program. You can find these settings by checking the configurations of a web browser on another machine connected to the same network.

The files `Packages.xz` and `Sources.xz` are then automatically downloaded to update the list of packages recognized by APT.

BACK TO BASICS

HTTP proxy

An HTTP proxy is a server that forwards an HTTP request for network users. It sometimes helps to speed up downloads by keeping a copy of files that have been transferred through it (we then speak of *proxy/cache*). In some cases, it is the only means of accessing an external web server; in such cases it is essential to answer the corresponding question during installation for the program to be able to download the Debian packages through it.

Squid is the name of the server software used by Falcot Corp to offer this service.

4.2.16. Debian Package Popularity Contest

The Debian system contains a package called *popularity-contest*, whose purpose is to compile package usage statistics. Each week, this program collects information on the packages installed and those used recently, and anonymously sends this information to the Debian project servers. The project can then use this information to determine the relative importance of each package, which influences the priority that will be granted to them. In particular, the most “popular” packages will be included in the installation CD-ROM, which will facilitate their access for users who do not wish to download them or to purchase a complete set.

This package is only activated on demand, out of respect for the confidentiality of users’ usage.

4.2.17. Selecting Packages for Installation

The following step allows you to choose the purpose of the machine in very broad terms; the ten suggested tasks correspond to lists of packages to be installed. The list of the packages that will actually be installed will be fine-tuned and completed later on, but this provides a good starting point in a simple manner.

Some packages are also automatically installed according to the hardware detected (thanks to the program `discover-pkginstall` from the *discover* package).



Figure 4.13 Task choices

4.2.18. Installing the GRUB Bootloader

The bootloader is the first program started by the BIOS. This program loads the Linux kernel into memory and then executes it. It often offers a menu that allows the user to choose the kernel to load and/or the operating system to boot.

BEWARE Bootloader and dual boot

This phase in the Debian installation process detects the operating systems that are already installed on the computer, and automatically adds corresponding entries in the boot menu, but not all installation programs do this.

In particular, if you install (or reinstall) Windows thereafter, the bootloader will be erased. Debian will still be on the hard drive, but will no longer be accessible from the boot menu (except for Windows 10, where it will still be accessible through the Windows recovery console). You would then have to boot the Debian installation system in **rescue** mode to set up a less exclusive bootloader. This operation is described in detail in the installation manual.

➡ <https://www.debian.org/releases/stable/amd64/ch08s06>

By default, the menu proposed by GRUB contains all the installed Linux kernels, as well as any other operating systems that were detected. This is why you should accept the offer to install it in the Master Boot Record. Since keeping older kernel versions preserves the ability to boot the same system if the most recently installed kernel is defective or poorly adapted to the hardware, it often makes sense to keep a few older kernel versions installed.

GRUB is the default bootloader installed by Debian thanks to its technical superiority: it works with most filesystems and therefore doesn't require an update after each installation of a new kernel, since it reads its configuration during boot and finds the exact position of the new kernel. Version 1 of GRUB (now known as "Grub Legacy") couldn't handle all combinations of LVM and software RAID; version 2, installed by default, is more complete. There may still be situations where it is more recommendable to install LILO (another bootloader); the installer will suggest it automatically.

It is worth noting that GRUB is not a single bootloader, it is more like a collection of bootloaders suited for different cases. The numerous binary packages built out of the GRUB source package reflect that: *grub-efi-amd64* is for 64-bit PC booting in UEFI mode, *grub-efi-ia32* is for 32-bit PC booting in UEFI mode, *grub-pc* is for PC booting in BIOS mode, *grub-uboot* for ARM computers, etc.

For more information on configuring GRUB, please refer to section 8.8.3, "GRUB 2 Configuration" page 182.

CULTURE

Secure Boot and the shim bootloader

Secure Boot is a technology ensuring that you run only software validated by your operating system vendor. To accomplish its work each element of the boot sequences validates the next software component that it will execute. At the deepest level, the UEFI firmware embeds cryptographic keys provided by Microsoft to check the bootloader's signature, ensuring that it is safe to execute. Since getting a binary signed by Microsoft is a lengthy process, Debian decided to not sign GRUB directly. Instead it uses an intermediary bootloader called shim, which almost never needs to change, and whose only role is to check Debian's provided signature on GRUB and execute GRUB. To run Debian on a machine having Secure Boot enabled, you need to install the *shim-signed* package.

Down the stack, GRUB will do a similar check with the kernel, and then the kernel might also check signatures on modules that get loaded. The kernel might also forbid some operations that could alter the integrity of the system.

Debian 10 is the first release supporting Secure Boot. Before, you had to disable that feature in the system setup screen offered by the BIOS or the UEFI.

4.2.19. Finishing the Installation and Rebooting

The installation is now complete, the program invites you to remove the CD-ROM from the reader and to restart the computer.

4.3. After the First Boot

If you activated the task “Debian desktop environment” without any explicit desktop choice (or with the “GNOME” choice), the computer will display the gdm3 login manager.

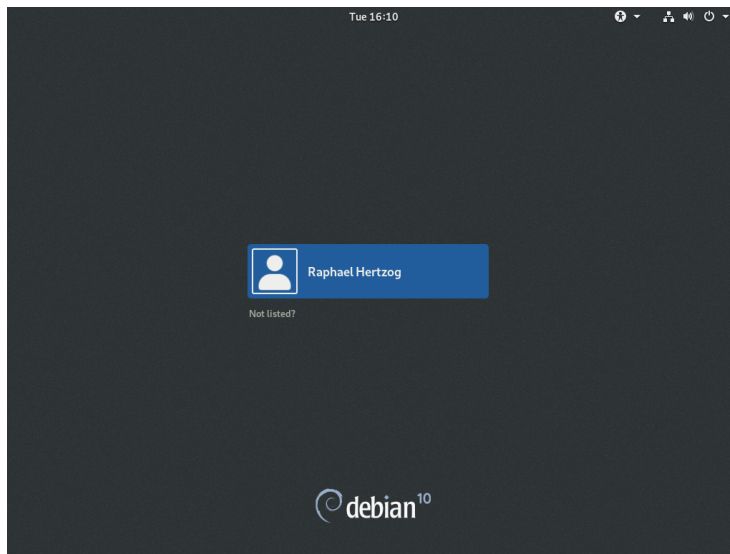


Figure 4.14 *First boot*

The user that has already been created can then log in and begin working immediately.

4.3.1. Installing Additional Software

The installed packages correspond to the profiles selected during installation, but not necessarily to the use that will actually be made of the machine. As such, you might want to use a package management tool to refine the selection of installed packages. The two most frequently used tools (which are installed if the “Debian desktop environment” profile was chosen) are `apt` (accessible from the command line) and `synaptic` (“Synaptic Package Manager” in the menus).

To facilitate the installation of coherent groups of programs, Debian creates “tasks” that are dedicated to specific uses (mail server, file server, etc.). You already had the opportunity to select them during installation, and you can access them again thanks to package management tools such as `aptitude` (the tasks are listed in a distinct section) and `synaptic` (through the menu `Edit → Mark Packages by Task...`).

`Aptitude` is an interface to `APT` in full-screen text mode. It allows the user to browse the list of available packages according to various categories (installed or not-installed packages, by task, by section, etc.), and to view all of the information available on each of them (dependencies, conflicts, description, etc.). Each package can be marked “install” (to be installed, + key) or “re-

move” (to be removed, - key). All of these operations will be conducted simultaneously once you’ve confirmed them by pressing the g key (“g” for “go!”). If you have forgotten some programs, no worries; you will be able to run `aptitude` again once the initial installation has been completed.

TIP

Debian thinks of speakers of non-English languages

Several tasks are dedicated to the localization of the system in other languages beyond English. They include translated documentation, dictionaries, and various other packages useful for speakers of different languages. The appropriate task is automatically selected if a non-English language was chosen during installation.

Of course, it is possible not to select any task to be installed. In this case, you can manually install the desired software with the `apt` or `aptitude` command (which are both accessible from the command line).

VOCABULARY

Package dependencies, conflicts

In the Debian packaging lingo, a “dependency” is another package necessary for the proper functioning of the package in question. Conversely, a “conflict” is a package that can not be installed side-by-side with another.

These concepts are discussed in greater detail in chapter 5, “[Packaging System: Tools and Fundamental Principles](#)” page 78.

4.3.2. Upgrading the System

A first `apt upgrade` (a command used to automatically update installed programs) is generally required, especially for possible security updates issued since the release of the latest Debian stable version. These updates may involve some additional questions through `debconf`, the standard Debian configuration tool. For further information on these updates conducted by `apt`, please refer to section 6.2.3, “[System Upgrade](#)” page 120.



Keywords

Binary package
Source package
dpkg
deb
dependencies
conflict

