Debian GNU/Linux - Ghid de instalare

30 septembrie 2019

Debian GNU/Linux - Ghid de instalare

Cuprins

1 Bine aţi venit în Debian **1**
- 1.1 Ce este Debian? . . . 1
- 1.2 Ce este GNU/Linux? . . . 1
- 1.3 Ce este Debian GNU/Linux? . . . 2
- 1.4 Ce este Debian GNU/kFreeBSD? . . . 3
- 1.5 Ce este Debian GNU/Hurd? . . . 3
- 1.6 What is the Debian Installer? . . . 3
- 1.7 Cum obţineţi Debian . . . 3
- 1.8 Cum obţineţi cea mai recentă versiune a acestui document . . . 4
- 1.9 Organizarea acestui document . . . 4
- 1.10 Ajutorul dumneavoastră la documentaţie este bine-venit . . . 4
- 1.11 Despre drepturi de autor şi licenţe de software . . . 4

2 System Requirements **6**
- 2.1 Supported Hardware . . . 6
  - 2.1.1 Supported Architectures . . . 6
  - 2.1.2 CPU Support . . . 7
  - 2.1.3 Laptops . . . 7
  - 2.1.4 Multiple Processors . . . 7
  - 2.1.5 Graphics Hardware Support . . . 7
  - 2.1.6 Network Connectivity Hardware . . . 7
    - 2.1.6.1 Wireless Network Cards . . . 7
  - 2.1.7 Braille Displays . . . 8
  - 2.1.8 Hardware Speech Synthesis . . . 8
  - 2.1.9 Peripherals and Other Hardware . . . 8
- 2.2 Devices Requiring Firmware . . . 8
- 2.3 Purchasing Hardware Specifically for GNU/Linux . . . 8
  - 2.3.1 Avoid Proprietary or Closed Hardware . . . 9
- 2.4 Installation Media . . . 9
  - 2.4.1 CD-ROM/DVD-ROM/BD-ROM . . . 9
  - 2.4.2 USB Memory Stick . . . 9
  - 2.4.3 Network . . . 9
  - 2.4.4 Hard Disk . . . 10
  - 2.4.5 Un*x or GNU system . . . 10
  - 2.4.6 Supported Storage Systems . . . 10
- 2.5 Memory and Disk Space Requirements . . . 10

3 Before Installing Debian GNU/Linux **11**
- 3.1 Overview of the Installation Process . . . 11
- 3.2 Back Up Your Existing Data! . . . 12
- 3.3 Information You Will Need . . . 12
  - 3.3.1 Documentation . . . 12
    - 3.3.1.1 Installation Manual . . . 12
    - 3.3.1.2 Hardware documentation . . . 12
  - 3.3.2 Finding Sources of Hardware Information . . . 12
  - 3.3.3 Hardware Compatibility . . . 13
    - 3.3.3.1 Testing hardware compatibility with a Live-System . . . 14
  - 3.3.4 Network Settings . . . 14
- 3.4 Meeting Minimum Hardware Requirements . . . 14
- 3.5 Pre-Partitioning for Multi-Boot Systems . . . 15
- 3.6 Pre-Installation Hardware and Operating System Setup . . . 16
  - 3.6.1 Invoking the BIOS Set-Up Menu . . . 16
  - 3.6.2 Boot Device Selection . . . 16
  - 3.6.3 Systems with UEFI firmware . . . 16

3.6.4 Disabling the Windows 8 „fast boot” feature . . . 17
3.6.5 Hardware Issues to Watch Out For . . . 17

4 Obtaining System Installation Media **18**
4.1 Official Debian GNU/Linux installation images . . . 18
4.2 Downloading Files from Debian Mirrors . . . 18
4.2.1 Where to Find Installation Images . . . 18
4.3 Preparing Files for USB Memory Stick Booting . . . 18
4.3.1 Preparing a USB stick using a hybrid CD/DVD image . . . 19
4.3.2 Manually copying files to the USB stick . . . 19
4.3.3 Manually copying files to the USB stick — the flexible way . . . 20
4.3.3.1 Partitioning the USB stick . . . 20
4.3.3.2 Adding the installer image . . . 20
4.4 Preparing Files for Hard Disk Booting . . . 21
4.4.1 Hard disk installer booting from Linux using **LILO** or **GRUB** . . . 21
4.4.2 Hard disk installer booting from DOS using **loadlin** . . . 21
4.5 Preparing Files for TFTP Net Booting . . . 21
4.5.1 Setting up a DHCP server . . . 22
4.5.1.1 Enabling PXE Booting in the DHCP configuration . . . 22
4.5.2 Setting up a BOOTP server . . . 23
4.5.3 Enabling the TFTP Server . . . 23
4.5.4 Move TFTP Images Into Place . . . 24
4.6 Automatic Installation . . . 24
4.6.1 Automatic Installation Using the Debian Installer . . . 24

5 Booting the Installation System **25**
5.1 Booting the Installer on 64-bit PC . . . 25
5.1.1 Booting from USB Memory Stick . . . 25
5.1.2 Booting from optical disc (CD/DVD) . . . 25
5.1.3 Booting from Windows . . . 26
5.1.4 Booting from DOS using loadlin . . . 26
5.1.5 Booting from Linux using **LILO** or **GRUB** . . . 26
5.1.6 Booting with TFTP . . . 27
5.1.6.1 NIC or Motherboard that support PXE . . . 27
5.1.6.2 NIC with Network BootROM . . . 27
5.1.6.3 Etherboot . . . 27
5.1.7 The Boot Screen . . . 27
5.1.8 The Graphical Installer . . . 28
5.2 Accessibility . . . 29
5.2.1 Installer front-end . . . 29
5.2.2 USB Braille Displays . . . 29
5.2.3 Serial Braille Displays . . . 29
5.2.4 Software Speech Synthesis . . . 29
5.2.5 Hardware Speech Synthesis . . . 30
5.2.6 Board Devices . . . 30
5.2.7 High-Contrast Theme . . . 30
5.2.8 Zoom . . . 30
5.2.9 Expert install, rescue mode, automated install . . . 30
5.2.10 Accessibility of the installed system . . . 30
5.3 Boot Parameters . . . 30
5.3.1 Boot console . . . 31
5.3.2 Debian Installer Parameters . . . 31
5.3.3 Using boot parameters to answer questions . . . 33
5.3.4 Passing parameters to kernel modules . . . 33
5.3.5 Blacklisting kernel modules . . . 34
5.4 Troubleshooting the Installation Process . . . 34
5.4.1 Reliability of optical media . . . 34
5.4.1.1 Common issues . . . 34
5.4.1.2 How to investigate and maybe solve issues . . . 34

5.4.2 Boot Configuration . . . 35
5.4.3 Software Speech Synthesis . . . 35
5.4.4 Common 64-bit PC Installation Problems . . . 36
5.4.4.1 System Freeze During the PCMCIA Configuration Phase . . . 36
5.4.5 Interpreting the Kernel Startup Messages . . . 36
5.4.6 Reporting Installation Problems . . . 36
5.4.7 Submitting Installation Reports . . . 36

6 Cum se folosește Programul de instalare al Debian 38
6.1 Cum funcționează programul de instalare . . . 38
6.1.1 Using the graphical installer . . . 39
6.2 Componentele - Introducere . . . 39
6.3 Folosirea componentelor individuale . . . 40
6.3.1 Pregătirea Programului de instalare al Debian și a configurației hardware . . . 40
6.3.1.1 Check available memory / low memory mode . . . 41
6.3.1.2 Selectarea opțiunilor de localizare . . . 41
6.3.1.3 Alegerea tastaturii . . . 42
6.3.1.4 Căutarea imaginii ISO a Programului de instalare al Debian . . . 42
6.3.1.5 Configuring the Network . . . 42
6.3.1.5.1 Automatic network configuration . . . 42
6.3.1.5.2 Manual network configuration . . . 43
6.3.1.5.3 IPv4 and IPv6 . . . 43
6.3.2 Setting Up Users And Passwords . . . 43
6.3.2.1 Set the Root Password . . . 43
6.3.2.2 Create an Ordinary User . . . 43
6.3.3 Configuring the Clock and Time Zone . . . 44
6.3.4 Partiționarea și selectarea punctelor de montare . . . 44
6.3.4.1 Supported partitioning options . . . 44
6.3.4.2 Guided Partitioning . . . 45
6.3.4.3 Manual Partitioning . . . 46
6.3.4.4 Configuring Multidisk Devices (Software RAID) . . . 47
6.3.4.5 Configurarea Managerului de volume logice (LVM) . . . 49
6.3.4.6 Configurarea volumelor criptate . . . 50
6.3.5 Installing the Base System . . . 52
6.3.6 Installing Additional Software . . . 52
6.3.6.1 Configuring apt . . . 52
6.3.6.1.1 Installing from more than one CD or DVD image . . . 53
6.3.6.1.2 Using a network mirror . . . 53
6.3.6.1.3 Choosing a network mirror . . . 53
6.3.6.2 Selecting and Installing Software . . . 54
6.3.7 Making Your System Bootable . . . 55
6.3.7.1 Detecting other operating systems . . . 55
6.3.7.2 Install the **Grub** Boot Loader on a Hard Disk . . . 55
6.3.7.3 Install the **LILO** Boot Loader on a Hard Disk . . . 55
6.3.7.4 Continue Without Boot Loader . . . 56
6.3.8 Finishing the Installation . . . 56
6.3.8.1 Setting the System Clock . . . 56
6.3.8.2 Reboot the System . . . 56
6.3.9 Troubleshooting . . . 56
6.3.9.1 Saving the installation logs . . . 56
6.3.9.2 Using the Shell and Viewing the Logs . . . 56
6.3.10 Installation Over the Network . . . 57
6.4 Loading Missing Firmware . . . 58
6.4.1 Preparing a medium . . . 58
6.4.2 Firmware and the Installed System . . . 59

7 Booting Into Your New Debian System **60**
7.1 The Moment of Truth . . . 60
7.2 Mounting encrypted volumes . . . 60
7.2.1 Troubleshooting . . . 61
7.3 Log In . . . 61

8 Next Steps and Where to Go From Here **62**
8.1 Shutting down the system . . . 62
8.2 Orienting Yourself to Debian . . . 62
8.2.1 Debian Packaging System . . . 62
8.2.2 Additional Software Available for Debian . . . 63
8.2.3 Application Version Management . . . 63
8.2.4 Cron Job Management . . . 63
8.3 Further Reading and Information . . . 63
8.4 Setting Up Your System To Use E-Mail . . . 63
8.4.1 Default E-Mail Configuration . . . 64
8.4.2 Sending E-Mails Outside The System . . . 64
8.4.3 Configuring the Exim4 Mail Transport Agent . . . 64
8.5 Compiling a New Kernel . . . 65
8.6 Recovering a Broken System . . . 65

A Installation Howto **66**
A.1 Preliminaries . . . 66
A.2 Booting the installer . . . 66
A.2.1 Optical disc . . . 66
A.2.2 USB memory stick . . . 66
A.2.3 Booting from network . . . 67
A.2.4 Booting from hard disk . . . 67
A.3 Installation . . . 67
A.4 Send us an installation report . . . 68
A.5 And finally… . . . 68

B Automating the installation using preseeding **69**
B.1 Introduction . . . 69
B.1.1 Preseeding methods . . . 69
B.1.2 Limitations . . . 70
B.2 Using preseeding . . . 70
B.2.1 Loading the preconfiguration file . . . 70
B.2.2 Using boot parameters to preseed questions . . . 71
B.2.3 Auto mode . . . 71
B.2.4 Aliases useful with preseeding . . . 72
B.2.5 Using a DHCP server to specify preconfiguration files . . . 73
B.3 Creating a preconfiguration file . . . 73
B.4 Contents of the preconfiguration file (for bullseye) . . . 74
B.4.1 Localization . . . 74
B.4.2 Network configuration . . . 75
B.4.3 Network console . . . 76
B.4.4 Mirror settings . . . 77
B.4.5 Account setup . . . 77
B.4.6 Clock and time zone setup . . . 78
B.4.7 Partitioning . . . 78
B.4.7.1 Partitioning example . . . 78
B.4.7.2 Partitioning using RAID . . . 80
B.4.7.3 Controlling how partitions are mounted . . . 81
B.4.8 Base system installation . . . 81
B.4.9 Apt setup . . . 81
B.4.10 Package selection . . . 82
B.4.11 Boot loader installation . . . 83
B.4.12 Finishing up the installation . . . 83

B.4.13 Preseeding other packages . . . 84
B.5 Advanced options . . . 84
B.5.1 Running custom commands during the installation . . . 84
B.5.2 Using preseeding to change default values . . . 84
B.5.3 Chainloading preconfiguration files . . . 85

C Partitioning for Debian **86**
C.1 Deciding on Debian Partitions and Sizes . . . 86
C.2 The Directory Tree . . . 86
C.3 Recommended Partitioning Scheme . . . 87
C.4 Device Names in Linux . . . 88
C.5 Debian Partitioning Programs . . . 88
C.5.1 Partitioning for 64-bit PC . . . 88

D Random Bits **90**
D.1 Linux Devices . . . 90
D.1.1 Setting Up Your Mouse . . . 90
D.2 Disk Space Needed for Tasks . . . 91
D.3 Installing Debian GNU/Linux from a Unix/Linux System . . . 91
D.3.1 Getting Started . . . 92
D.3.2 Install **debootstrap** . . . 92
D.3.3 Run **debootstrap** . . . 93
D.3.4 Configure The Base System . . . 93
D.3.4.1 Create device files . . . 93
D.3.4.2 Mount Partitions . . . 94
D.3.4.3 Setting Timezone . . . 94
D.3.4.4 Configure Networking . . . 95
D.3.4.5 Configure Apt . . . 96
D.3.4.6 Configure Locales and Keyboard . . . 96
D.3.5 Install a Kernel . . . 96
D.3.6 Set up the Boot Loader . . . 96
D.3.7 Remote access: Installing SSH and setting up access . . . 97
D.3.8 Finishing touches . . . 97
D.4 Installing Debian GNU/Linux over Parallel Line IP (PLIP) . . . 97
D.4.1 Requirements . . . 98
D.4.2 Setting up source . . . 98
D.4.3 Installing target . . . 98
D.5 Installing Debian GNU/Linux using PPP over Ethernet (PPPoE) . . . 99

E Administrivia **100**
E.1 About This Document . . . 100
E.2 Contributing to This Document . . . 100
E.3 Major Contributions . . . 100
E.4 Trademark Acknowledgement . . . 101

F GNU General Public License **102**
F.1 Preamble . . . 102
F.2 GNU GENERAL PUBLIC LICENSE . . . 102
F.3 How to Apply These Terms to Your New Programs . . . 105

Listă de tabele

3 Before Installing Debian GNU/Linux
3.1 Hardware Information Helpful for an Install 13
3.2 Recommended Minimum System Requirements 15

Rezumat

Acest document conţine instrucţiuni de instalare pentru sistemul Debian GNU/Linux 11 (nume de cod „bullseye”), pentru arhitectura 64-bit PC („amd64”). Conţine, de asemenea, indicii pentru a găsi mai multe informaţii, în general, şi informaţii legate de felul în care puteţi să profitaţi la maxim de noul dumneavoastră sistem Debian, în particular.

AVERTISMENT

This translation of the installation guide is not up-to-date and currently there is noone actively working on updating it. Keep this in mind when reading it; it may contain outdated or wrong information. Read or double-check the English variant, if in doubt. If you can help us with updating the translation, please contact debian-boot@lists.debian.org or the debian-l10n-xxx mailinglist for this language. Many thanks

Instalarea Debian GNU/Linux 11 pentru amd64

Suntem încântaţi că v-aţi decis să încercaţi Debian, şi suntem siguri că veţi alfa că distribuiţia Debian GNU/Linux este unică. Debian GNU/Linux aduce laolaltă software liber de înaltă calitate de peste tot din lume, întegrându-l într-un tot coerent. Credem că veţi considera că rezultatul este, într-adevăr, mai bun decât părţile însumate.

Înţelegem că mulţi dintre voi doriţi să instalaţi Debian fără să cititi acest manual, iar programul de instalare Debian este conceput astfel încât să facă acest lucru posibil. Dacă nu aveţi timp să citiţi în întregime Ghidul de instalare chiar acum, vă recomandăm să citiţi Reţetarul instalării, care vă va conduce prin procesul de instalare de bază, şi face referiri la manual, în cazul subiectelor mai avansate sau pentru cazurile în care lucrurile merg rău. Reţetarul de instalare poate fi găsit în Anexa A.

Acestea fiind zise, sperăm că veţi avea timp să citiţi o mare parte din acest manual, şi, procedând astfel, să fiţi conduşi către o instalare în cunoştinţă de cauză şi cu şanse mai mare de a avea o experienţă de instalare cu un succes mai mare.

Capitolul 1

Bine aţi venit în Debian

Acest capitol oferă o vedere de ansamblu asupra Proiectului Debian şi Debian GNU/Linux. Dacă cunoaşteţi deja cu isoria Proiectului Debian şi distribuţia Debian GNU/Linux, puteţi să treceţi direct la capitolul următor.

1.1 Ce este Debian?

Debian is an all-volunteer organization dedicated to developing free software and promoting the ideals of the Free Software community. The Debian Project began in 1993, when Ian Murdock issued an open invitation to software developers to contribute to a complete and coherent software distribution based on the relatively new Linux kernel. That relatively small band of dedicated enthusiasts, originally funded by the Free Software Foundation and influenced by the GNU philosophy, has grown over the years into an organization of around 1000 *Debian Developers*.

Dezvoltatorii Debian sunt implicaţi în multe activităţi, inclusiv administrarea paginilor de web şi siturilor FTP designul de grafică, analizelor legale a licenţelor de software, scrierea de documentaţie şi, desigur, mentenenţa pachetelor de software.

În scopul de a comunica filosofia noastră şi în scopul atragerii altor dezvoltatori care cred în principiile pentru care Debian luptă, Proiectul Debian a publicat o serie de documente ce schiţează valorile noastre şi servesc drept ghid pentru a înţelege ce înseamnă a fi Dezvoltator Debian:

- Contracul Social al Debian este o declaraţie a Debian în care se afirmă angajamentul acestuia faţă de Comunitatea Software-ului Liber. Oricine este de acord să respecte Contractul Social poate deveni un responsabil de pachet (întreţinător). Orice responsabil de pechet poate introduce noi programe în Debian — cu condiţia ca programele în cauză să respecte criteriile noastre de libertate şi pachetele respectă standardele noastre de calitate.
- Ghidul Debian al Software-ului Liber este o declaraţie clară şi concisă a criteriilor Debian pentru software-ul liber. DFSG-ul (eng., acronim pentru Ghidul Debian al Software-ul Liber) este un document foarte influent în Mişcare Software-ului Liber, şi a fost foundaţia Definiţiei pentru Open Source.
- Manualul Regulilor Debian este un set de specificaţii extensive a standardelor de calitate ale Proiectului Debian.

Dezvoltatorii Debian sunt împlicaţi şi în alte proiecte; unele sunt specifice Debian, iar altele implicând o parte sau întreaga comunitate Linux. Unele exemple includ:

- Standardul pentru ierarhia sistemului de fişiere (FHS) este un efort de standardizare a aranjamentului sistemului de fişiere Linux. FHS-ul va permite dezvoltatorilor de software să-şi concentreze efortul pentru a proiecta programe, fără să se îngrijoreze în legătură cu felul în care pachetul va fi instalat în diferitele distribuţii GNU-/Linux.
- Debian Jr. este un proiect intern, care este orientat spre scopul de a asigura că Debian are ceva de oferit celor mai tineri utilizatori ai noştri.

Pentru mai multe informaţii generale despre Debian, a se vedea Debian FAQ.

1.2 Ce este GNU/Linux?

Linux este un sistem de operare: o serie de programe care vă lasă să interacţionaţi cu calculatorul şi să rulaţi alte programe.

Un sistem de operare este format din diverse programe fundamentale care sunt necesare calculatorului dumneavoastră să comunice şi să recepţioneze instrucţiuni de la utilizatori; citirea şi scrierea datelor pe discurile fixe, casete, şi la imprimante; controlul utilizării memoriei; şi rularea de alte programe. Cea mai importantă parte a unui sistem de operare este nucleul. Într-un sistem GNU/Linux, Linux este nucleul. Restul sistemului e format din alte programe, multe dintre acestea fiind scrise de către sau pentru Proiectul GNU. Deoarece doar nucleul Linux nu formează un sistem de operare funcţional, noi preferăm să folosim termenul „GNU/Linux” pentru a ne referi la sistemele la care mulţi oameni le spun neglijent „Linux”.

Linux este modelat pe baza sistemului de operare Unix. De la început a fost proiectat să fie un sistem multi-tasking şi multi-utilizator. Doar aceste lucruri sunt suficiente pentru a face ca Linux să fie diferit faţă de alte sisteme de operare bine cunoscute. Totuşi, Linux este şi mai diferit, mai mult decât v-aţi putea imagina. Faţă de alte sisteme de operare, nimieni nu deţine Linux. Mare parte din dezvoltarea sa este făcută de către volutari neplătiţi.

Development of what later became GNU/Linux began in 1984, when the Free Software Foundation began development of a free Unix-like operating system called GNU.

The GNU Project has developed a comprehensive set of free software tools for use with Unix™ and Unix-like operating systems such as Linux. These tools enable users to perform tasks ranging from the mundane (such as copying or removing files from the system) to the arcane (such as writing and compiling programs or doing sophisticated editing in a variety of document formats).

În timp ce multe grupuri şi mulţi indivizi au contribuit la Linux, cel mai mare contribuitor încă este Fundaţia pentru Software Liber care a creat nu numai majoritatea uneltelor folosite în Linux, ci şi filosofia şi comunitatea care a făcut să Linux să fie posibil.

Nucleul Linux a apărut iniţial în 1991, când un student finlandez care studia ştiinţa calculatoarelor, numit Linus Torvalds a anunţat în grupurile de ştiri Usenet **`comp.os.minix`** o versiune incipientă a unui nucleu ce înlocuia nucleul Minix. A se vedea Pagina istoriei Linux scrisă de Linux International.

Linus Torvalds continues to coordinate the work of several hundred developers with the help of a number of subsystem maintainers. There is an official website for the Linux kernel. Information about the **`linux-kernel`** mailing list can be found on the linux-kernel mailing list FAQ.

Utilizatorii de Linux au o libertate imensă de alegere a programelor lor. De exemplu, utilizatorii de Linux pot alege dintr-o duzină de console pentru linia de comandă şi câteva medii grafice. Această selecţie adesea dezorientează utilizatorii altor sisteme de operare care nu sunt obişnuiţi să se gândească la linia de comandă sau mediul grafic ca la ceva ce se poate schimba.

De asemenea, Linux-ul are şanse mai mici să se blocheze sau să se defecteze, este mai capabil să ruleze mai mult de un program în acelaşi timp, şi este mult mai bine securizat decât multe sisteme de operare. Cu aceste avantaje, Linux este sistemul de operare cu cea mai mare rată de creştere de pe piaţa de servere. Recent, Linux a început să fie popular şi între utilizatorii casnici şi cei din mediul de afaceri.

1.3 Ce este Debian GNU/Linux?

Combinaţia dintre filosofia şi metodologia Debian, uneltele GNU, nucleul Linux şi alte programe libere importante formează o distribuiţie unică numită Debian GNU/Linux.Această distribuţie este formată dintr-un număr mare de *pachete* de software. Fiecare pachet din distribuţie conţine executabile, script-uri, documentaţie şi informaţii de configurare şi au un *responabil* care este în special delegat să ţină pachetul actualizat, să urmărească rapoartele cu probleme şi să comunice cu autorul sau autorii software-ului împachetat. numărul nostru mare de utilizatori, combinat cu sistemul nostru de raportare a erorilor asigură ca problemele să fie găsite şi reparate repede.

Atenţia proiectului Debian la detalii îi permite să producă o distribuţie de înaltă calitate, stabilă şi scalabilă. Instalările pot fi configurate uşor sa servească în multe roluri, de la firewall-uri simple sau sisteme pentru staţii de lucru ştiinţific până la servere profesionale.

Debian este în special popular printre utilizatorii avansaţi datorită calităţii sale tehnice excelente şi puternicii sale dedicări spre a satisface nevoile şi aşteptările comunităţii Linux. Debian a introdus, de asemenea, multe facilităţi în Linux care, în prezent, sunt foarte comune.

De exemplu, Debian a fost prima distribuţie care a introdus un sistem de mnagement al pachetelor pentru instalarea şi ştergerea facilă a programelor. Tot Debian a fost prima distribuţie Linux care a făcut posibilă actualizarea la o nouă versiune lansată, fără a fi nevoie să fie reinstalată.

Debian continuă să fie un lider în dezvoltarea Linux. Procesul său de dezvoltare este un exemplu care arată cât de bine poate funcţiona modelul de dezvoltare Open Source — chiar şi pentru sarcini complexe precum construirea şi mentenanţa unui sistem de operare complet.

Caracteristica ce distinge Debian de celelalte distribuţii Linux este sistemul său de management al pachetelor. Aceste unelte oferă administratorului unui sistem Debian controlul complet asupra pachetelor instalate pe acel sistem, inclusiv abilitatea de a instala un singur pachet, sau de a înnoi complet întregul sistem de operare. Pachetele individuale

pot fi şi protejate de operaţia de înnoire. Puteţi chiar comunica sistemului de pachete informaţii în legătură cu programe pe care le-aţi compilat chiar dvs. şi dependenţle pe care le satisface.

Pentru a va proteja sistemul de „cai troieni" şi alte programe maliţioase, serverele Debian verifică şi faptul că pachetele publicate provin de la proprii lor responsabili. Responsabilii de pachete au o grijă foarte mare să-şi configureze pachetele într-o manieră securizată. Atunci când apar probleme de securitate în pachetele livrate, versiuni reparate apar, de obicei, foarte repede. Cu opţiunile de înnoire simple ale Debian, pachetele corectate pot fi descărcate şi instalate automat prin Internet.

Metoda primordială, şi cea mai bună, de a primi suport pentru sistemul dumeavoastră Debian GNU/Linux şi de a comunica cu dezvoltatorii Debian este prin intermediul listelor de discuţii care sunt întreţinute de Proiectul Debian (existând mai mult de 317 la momentul scrierii acestui document). Cel mai uşor mod de a vă înscrie pe una sau mai multe dintre aceste liste, este de a vizita pagina de înscrieri la listele de discuţii Debian şi de a completa formularul pe care-l veţi găsi acolo.

1.4 Ce este Debian GNU/kFreeBSD?

Debian GNU/kFreeBSD is a Debian GNU system with the kFreeBSD kernel.

This port of Debian is currently only being developed for the i386 and amd64 architectures, although ports to other architectures is possible.

Please note that Debian GNU/kFreeBSD is not a Linux system, and thus some information on Linux system may not apply to it.

For more information, see the Debian GNU/kFreeBSD ports page and the debian-bsd@lists.debian.org mailing list.

1.5 Ce este Debian GNU/Hurd?

Debian GNU/Hurd is a Debian GNU system with the GNU Hurd — a set of servers running on top of the GNU Mach microkernel.

The Hurd is still unfinished, and is unsuitable for day-to-day use, but work is continuing. The Hurd is currently only being developed for the i386 architecture, although ports to other architectures will be made once the system becomes more stable.

Please note that Debian GNU/Hurd is not a Linux system, and thus some information on Linux system may not apply to it.

Pentru mai multe informaţii, a se vedea pagina portului Debian GNU/Hurd şi lista de discuţii debian-hurd@lists.debian.org.

1.6 What is the Debian Installer?

Debian Installer, also known as „d-i", is the software system to install a basic working Debian system. A wide range of hardware such as embedded devices, laptops, desktops and server machines is supported and a large set of free software for many purposes is offered.

The installation is conducted by answering a basic set of questions. Also available are an expert mode that allows to control every aspect of the installation and an advanced feature to perform automated installations. The installed system can be used as is or further customized. The installation can be performed from a multitude of sources: USB, CD/DVD/Blu-Ray or the network. The installer supports localized installations in more than 80 languages.

The installer has its origin in the boot-floppies project, and it was first mentioned by Joey Hess in 2000. Since then the installation system has been continuously developed by volunteers improving and adding more features.

More information can be found on the Debian Installer page, on the Wiki and on the debian-boot mailing list.

1.7 Cum obţineţi Debian

For information on how to download Debian GNU/Linux from the Internet or from whom official Debian installation media can be purchased, see the distribution web page. The list of Debian mirrors contains a full set of official Debian mirrors, so you can easily find the nearest one.

Debian poate fi actuallizat, după instalare, foarte uşor. Procedura de instalare vă va ajuta să configuraţi sistemul astfel încât să puteţi să faceţi actualizările, la terminarea instalării, dacă este nevoie.

1.8 Cum obţineţi cea mai recentă versiune a acestui document

Acst document este revizuit constant. Verificaţi paginile Debian 11 pentru a verifica dacă există informaţii de ultimă oră în legătură cu versiunea 11 a sistemului Debian GNU/Linux. Versiuni actualizate ale acstui manual de instalare sunt, de asemenea, disponibile de pe paginile oficiale ale Manualilui de instalare.

1.9 Organizarea acestui document

Acest document este menit să servească pe post de manual pentru utilizatorii de Debian incepători. Încearcă să facă cât mai puţine presupuneri în legătură cu nivelul dumneavoastră de expertiză. Totuşi, presupunem că aveţi o idee generală despre felul în care funcţionează componentele din calculatorul dvs.

Utilizatorii experimenetaţi pot găsi şi ei informaţii interesante de referinţă, inclusiv dimensiunile minime de instalare, detalii legate de componentele suportate de sistemul de instalare Debian ş.a.m.d. Încurajăm utlizatorii exprimentaţi să răsfoiască în voie acest document.

În general, acest manual este dispus într-o manieră liniară, purtându-vă prin procesul de instalare, de la început până la sfârşit. Iată aici paşii necesari pentru a instala Debian GNU/Linux şi secţiunile acestui document care sunt corelate cu fiecare dintre paşi:

1. Determinaţi dacă hardware-ul dvs. respectă condiţiile necesare pentru a folosi sistemul de instalare, în Cap. 2.
2. Faceţi o copie de siguranţă a sistemului, planificaţi şi configuraţi, în caz că este nevoie, componentele, înainte de a instala Debian în Cap. 3. Dacă pregătiţi un sistem cu mai multe sisteme de operare, probabil că va trebui să creaţi spaţiu partiţionabil pe disc pentru a fi folosit de Debian.
3. În Cap. 4, veţi obţine fişierele necesare instalării, conform cu metoda dvs. de instalare.
4. The next Cap. 5 describes booting into the installation system. This chapter also discusses troubleshooting procedures in case you have problems with this step.
5. Perform the actual installation according to Cap. 6. This involves choosing your language, configuring peripheral driver modules, configuring your network connection, so that remaining installation files can be obtained directly from a Debian server (if you are not installing from a set of CD/DVD installation images), partitioning your hard drives and installation of a base system, then selection and installation of tasks. (Some background about setting up the partitions for your Debian system is explained in Anexa C.)
6. Pornirea în sistemul de bază proaspăt instalat, de la Cap. 7.

Odată instalat sistemul dumneavoastră, puteţi citi Cap. 8. Acel capitol explică unde trebuie să căutaţi mai multe informaţii legate de Unix şi de Debian, şi cum să înlocuiţi nucleul.

În sfârşit, informaţii despre acest document şi cum să contribuiţi la el pot fi găsite în Anexa E.

1.10 Ajutorul dumneavoastră la documentaţie este bine-venit

Orice ajutor, sugestii şi, în special, corecţii, sunt foarte apreciate. Versiunea în lucru a acestui document poate fi găsită la `https://d-i.debian.org/manual/`. Acolo veţi găsi o listă a diverselor arhitecturi şi limbi pentru care acest document este disponibil.

Sursele sunt şi ele disponibile public; căutaţi în Anexa E mai multe informaţii legate de modul în care puteţi contribui. Sunt bine-venite sugestiile, comentariile, corecţiile şi rapoartele de probleme (folosiţi pachetul `installation-guide` pentru probleme, dar verificaţi dacă problema este deja raportată deja).

1.11 Despre drepturi de autor şi licenţe de software

Suntem siguri că aţi citit câteva dintre licenţele care vin cu majoritatea programeleor comerciale — de obicei spun că puteţi folosi doar o copie a programului pe un singur calculator. Licenţa acestui sistem nu este deloc aşa. Vă încurajăm să puneţi o copie pe fiecare dintre calculatoarele din şcoala dvs., sau de la locul dvs. de muncă. Împrumutaţi mediul dvs. de instalare prietenilor dvs. şi ajutaţi-i să instaleze sistemul pe calculatoarele lor! Puteţi face chiar mii de copii şi să le *vindeţi* — cu câteva restricţii. Libertatea dvs, de a instala şi folosi sistemul vine direct din faptul că Debian este bazat pe *software liber*.

Calling software *free* doesn't mean that the software isn't copyrighted, and it doesn't mean that installation media containing that software must be distributed at no charge. Free software, in part, means that the licenses of individual

programs do not require you to pay for the privilege of distributing or using those programs. Free software also means that not only may anyone extend, adapt, and modify the software, but that they may distribute the results of their work as well.

Notă

The Debian project, as a pragmatic concession to its users, does make some packages available that do not meet our criteria for being free. These packages are not part of the official distribution, however, and are only available from the **contrib** or **non-free** areas of Debian mirrors or on third-party CD/DVD-ROMs; see the Debian FAQ, under „The Debian FTP archives", for more information about the layout and contents of the archives.

Multe din programele din sistem sunt licenţiate sub *Licenţa Publică Generală GNU*, căreia i se spune des „GPL". GPL-ul cere să faceţi disponibil *codul sursă* al progamelor disponibile, ori de câte ori distribuiţi o copie a programului; această dispoziţie a licenţei asigură faptul că orice utilizator va putea modifica programul. Datorită acestei dispoziţii, codul sursă[1] al tuturor programelor de acest fel din sistemul Debian este disponibil.

Există alte câteva forme de texte despre drepturile de autor şi texte de licenţe folosite de programele din Debian. Puteţi găsi drepturile de autor şi licenţele pentru fiecare pachet instalat în sistemul dvs. dacă vă uitaţi în fişierul `/usr/share/doc/`*`package-name`*`/copyright` , odată ce aţi instalat un pachete în sistemul dvs.

Pentru mai multe informaţii legate de licenţe şi felul în care Debian determină dacă software-ul e îndeajuns de liber pentru a fi inclus în distribuţia principală, a se vedea Ghidul Debian pentru software liber.

Cea mai importantă notă juridică este că acest software vine *fără nici un fel de garanţie*. Programatorii care au creat acest software au făcut acest lucru spre beneficiul comunităţii. Nu se oferă nici o garanţie legată de potrivirea software-ului cu oricare scop dat. Totuşi, din moment ce software-ul e liber, puteţi să-l modificaţi atfel încât să fie potrivit pentru propriile dvs. nevoi — şi să vă bucuraţi de schimbările făcute de alţii care au extins software-ul în acest mod.

[1] Pentru informaţii legate de modul în care puteţi găsi, despacheta şi construi pachete binare din pachetele sursă Debian, a se vedea Întrebările frecvente despre Debian, în „Bazele sistemului de management al pachetelor Debian".

Capitolul 2

System Requirements

This section contains information about what hardware you need to get started with Debian. You will also find links to further information about hardware supported by GNU and Linux.

2.1 Supported Hardware

Debian does not impose hardware requirements beyond the requirements of the Linux or kFreeBSD kernel and the GNU tool-sets. Therefore, any architecture or platform to which the Linux or kFreeBSD kernel, libc, **gcc**, etc. have been ported, and for which a Debian port exists, can run Debian. Please refer to the Ports pages at `https://www.debian.org/ports/amd64/` for more details on 64-bit PC architecture systems which have been tested with Debian GNU/Linux.

Rather than attempting to describe all the different hardware configurations which are supported for 64-bit PC , this section contains general information and pointers to where additional information can be found.

2.1.1 Supported Architectures

Debian GNU/Linux 11 supports ten major architectures and several variations of each architecture known as „flavors”.

Architecture	Debian Designation	Subarchitecture	Flavor
AMD64 & Intel 64	amd64		
Intel x86-based	i386	default x86 machines	default
		Xen PV domains only	xen
ARM	armel	Marvell Kirkwood and Orion	marvell
ARM with hardware FPU	armhf	multiplatform	armmp
64bit ARM	arm64		
32bit MIPS (big-endian)	mips	MIPS Malta	4kc-malta
		Cavium Octeon	octeon
64bit MIPS (little-endian)	mips64el	MIPS Malta	5kc-malta
		Cavium Octeon	octeon
		Loongson 3	loongson-3
32bit MIPS (little-endian)	mipsel	MIPS Malta	4kc-malta
		Cavium Octeon	octeon
		Loongson 3	loongson-3
Power Systems	ppc64el	IBM POWER8 or newer machines	
64bit IBM S/390	s390x	IPL from VM-reader and DASD	generic

This document covers installation for the *64-bit PC* architecture using the *Linux* kernel. If you are looking for information on any of the other Debian-supported architectures take a look at the Debian-Ports pages.

2.1.2 CPU Support

Both AMD64 and Intel 64 processors are supported.

2.1.3 Laptops

From a technical point of view, laptops are normal PCs, so all information regarding PC systems applies to laptops as well. Installations on laptops nowadays usually work out of the box, including things like automatically suspending the system on closing the lid and laptop specfic hardware buttons like those for disabling the wifi interfaces („airplane mode"). Nonetheless sometimes the hardware vendors use specialized or proprietary hardware for some laptop-specific functions which might not be supported. To see if your particular laptop works well with GNU/Linux, see for example the Linux Laptop pages.

2.1.4 Multiple Processors

Multiprocessor support — also called „symmetric multiprocessing" or SMP — is available for this architecture. The standard Debian 11 kernel image has been compiled with *SMP-alternatives* support. This means that the kernel will detect the number of processors (or processor cores) and will automatically deactivate SMP on uniprocessor systems.

Having multiple processors in a computer was originally only an issue for high-end server systems but has become common in recent years nearly everywhere with the introduction of so called „multi-core" processors. These contain two or more processor units, called „cores", in one physical chip.

2.1.5 Graphics Hardware Support

Debian's support for graphical interfaces is determined by the underlying support found in X.Org's X11 system, and the kernel. Basic framebuffer graphics is provided by the kernel, whilst desktop environments use X11. Whether advanced graphics card features such as 3D-hardware acceleration or hardware-accelerated video are available, depends on the actual graphics hardware used in the system and in some cases on the installation of additional „firmware" images (see Secțiune 2.2).

On modern PCs, having a graphical display usually works out of the box. In very few cases there have been reports about hardware on which installation of additional graphics card firmware was required even for basic graphics support, but these have been rare exceptions. For quite a lot of hardware, 3D acceleration also works well out of the box, but there is still some hardware that needs binary blobs to work well.

Details on supported graphics hardware and pointing devices can be found at `https://wiki.freedesktop.org/xorg/`. Debian 11 ships with X.Org version 7.7.

2.1.6 Network Connectivity Hardware

Almost any network interface card (NIC) supported by the Linux kernel should also be supported by the installation system; drivers should normally be loaded automatically. This includes most PCI/PCI-Express cards as well as PCMCIA/Express Cards on laptops.

ISDN is supported, but not during the installation.

2.1.6.1 Wireless Network Cards

Wireless networking is in general supported as well and a growing number of wireless adapters are supported by the official Linux kernel, although many of them do require firmware to be loaded.

If firmware is needed, the installer will prompt you to load firmware. See Secțiune 6.4 for detailed information on how to load firmware during the installation.

Wireless NICs that are not supported by the official Linux kernel can generally be made to work under Debian GNU/Linux, but are not supported during the installation.

If there is a problem with wireless and there is no other NIC you can use during the installation, it is still possible to install Debian GNU/Linux using a full CD-ROM or DVD image. Select the option to not configure a network and install using only the packages available from the CD/DVD. You can then install the driver and firmware you need after the installation is completed (after the reboot) and configure your network manually.

In some cases the driver you need may not be available as a Debian package. You will then have to look if there is source code available in the internet and compile the driver yourself. How to do this is outside the scope of this manual. If no Linux driver is available, your last resort is to use the `ndiswrapper` package, which allows you to use a Windows driver.

2.1.7 Braille Displays

Support for braille displays is determined by the underlying support found in `brltty`. Most displays work under `brltty`, connected via either a serial port, USB or bluetooth. Details on supported braille devices can be found on the `brltty` website. Debian GNU/Linux 11 ships with `brltty` version 5.6.

2.1.8 Hardware Speech Synthesis

Support for hardware speech synthesis devices is determined by the underlying support found in `speakup`. `speakup` only supports integrated boards and external devices connected to a serial port (no USB, serial-to-USB or PCI adapters are supported). Details on supported hardware speech synthesis devices can be found on the `speakup` website. Debian GNU/Linux 11 ships with `speakup` version 3.1.6.

2.1.9 Peripherals and Other Hardware

Linux supports a large variety of hardware devices such as mice, printers, scanners, PCMCIA/CardBus/ExpressCard and USB devices. However, most of these devices are not required while installing the system.

USB hardware generally works fine. On some very old PC systems some USB keyboards may require additional configuration (see Secțiune 3.6.5). On modern PCs, USB keyboards and mice work without requiring any specific configuration.

2.2 Devices Requiring Firmware

Besides the availability of a device driver, some hardware also requires so-called *firmware* or *microcode* to be loaded into the device before it can become operational. This is most common for network interface cards (especially wireless NICs), but for example some USB devices and even some hard disk controllers also require firmware. With many graphics cards, basic functionality is available without additional firmware, but the use of advanced features requires an appropriate firmware file to be installed in the system.

On many older devices which require firmware to work, the firmware file was permanently placed in an EEPROM/Flash chip on the device itself by the manufacturer. Nowadays most new devices do not have the firmware embedded this way anymore, so the firmware file must be uploaded into the device by the host operating system every time the system boots.

In most cases firmware is non-free according to the criteria used by the Debian GNU/Linux project and thus cannot be included in the main distribution or in the installation system. If the device driver itself is included in the distribution and if Debian GNU/Linux legally can distribute the firmware, it will often be available as a separate package from the non-free section of the archive.

However, this does not mean that such hardware cannot be used during an installation. Starting with Debian GNU/Linux 5.0, `debian-installer` supports loading firmware files or packages containing firmware from a removable medium, such as a USB stick. See Secțiune 6.4 for detailed information on how to load firmware files or packages during the installation.

If the `debian-installer` prompts for a firmware file and you do not have this firmware file available or do not want to install a non-free firmware file on your system, you can try to proceed without loading the firmware. There are several cases where a driver prompts for additional firmware because it may be needed under certain circumstances, but the device does work without it on most systems (this e.g. happens with certain network cards using the tg3 driver).

2.3 Purchasing Hardware Specifically for GNU/Linux

There are several vendors, who ship systems with Debian or other distributions of GNU/Linux pre-installed. You might pay more for the privilege, but it does buy a level of peace of mind, since you can be sure that the hardware is well-supported by GNU/Linux.

If you do have to buy a machine with Windows bundled, carefully read the software license that comes with Windows; you may be able to reject the license and obtain a rebate from your vendor. Searching the Internet for „windows refund” may get you some useful information to help with that.

Whether or not you are purchasing a system with Linux bundled, or even a used system, it is still important to check that your hardware is supported by the Linux kernel. Check if your hardware is listed in the references found above. Let your salesperson (if any) know that you're shopping for a Linux system. Support Linux-friendly hardware vendors.

2.3.1 Avoid Proprietary or Closed Hardware

Some hardware manufacturers simply won't tell us how to write drivers for their hardware. Others won't allow us access to the documentation without a non-disclosure agreement that would prevent us from releasing the driver's source code, which is one of the central elements of free software. Since we haven't been granted access to usable documentation on these devices, they simply won't work under Linux.

In many cases there are standards (or at least some de-facto standards) describing how an operating system and its device drivers communicate with a certain class of devices. All devices which comply to such a (de-facto-)standard can be used with a single generic device driver and no device-specific drivers are required. With some kinds of hardware (e.g. USB „Human Interface Devices", i.e. keyboards, mice, etc., and USB mass storage devices like USB flash disks and memory card readers) this works very well and practically every device sold in the market is standards-compliant.

In other fields, among them e.g. printers, this is unfortunately not the case. While there are many printers which can be addressed via a small set of (de-facto-)standard control languages and therefore can be made to work without problems in any operating system, there are quite a few models which only understand proprietary control commands for which no usable documentation is available and therefore either cannot be used at all on free operating systems or can only be used with a vendor-supplied closed-source driver.

Even if there is a vendor-provided closed-source driver for such hardware when purchasing the device, the practical lifespan of the device is limited by driver availability. Nowadays product cycles have become short and it is not uncommon that a short time after a consumer device has ceased production, no driver updates get made available any more by the manufacturer. If the old closed-source driver does not work anymore after a system update, an otherwise perfectly working device becomes unusable due to lacking driver support and there is nothing that can be done in this case. You should therefore avoid buying closed hardware in the first place, regardless of the operating system you want to use it with.

You can help improve this situation by encouraging manufacturers of closed hardware to release the documentation and other resources necessary for us to provide free drivers for their hardware.

2.4 Installation Media

This section will help you determine which different media types you can use to install Debian. There is a whole chapter devoted to media, Cap. 4, which lists the advantages and disadvantages of each media type. You may want to refer back to this page once you reach that section.

2.4.1 CD-ROM/DVD-ROM/BD-ROM

Installation from optical disc is supported for most architectures.

On PCs SATA, IDE/ATAPI, USB and SCSI optical drives are supported, as are FireWire devices that are supported by the ohci1394 and sbp2 drivers.

2.4.2 USB Memory Stick

USB flash disks a.k.a. USB memory sticks have become a commonly used and cheap storage device. Most modern computer systems also allow booting the `debian-installer` from such a stick. Many modern computer systems, in particular netbooks and thin laptops, do not have an optical drive anymore at all and booting from USB media is the standard way of installing a new operating system on them.

2.4.3 Network

The network can be used during the installation to retrieve files needed for the installation. Whether the network is used or not depends on the installation method you choose and your answers to certain questions that will be asked during the installation. The installation system supports most types of network connections (including PPPoE, but not ISDN or PPP), via either HTTP or FTP. After the installation is completed, you can also configure your system to use ISDN and PPP.

You can also *boot* the installation system over the network without needing any local media like CDs/DVDs or USB sticks. If you already have a netboot-infrastructure available (i.e. you are already running DHCP and TFTP services in your network), this allows an easy and fast deployment of a large number of machines. Setting up the necessary infrastructure requires a certain level of technical experience, so this is not recommended for novice users.

Diskless installation, using network booting from a local area network and NFS-mounting of all local filesystems, is another option.

2.4.4 Hard Disk

Booting the installation system directly from a hard disk is another option for many architectures. This will require some other operating system to load the installer onto the hard disk. This method is only recommended for special cases when no other installation method is available.

2.4.5 Un*x or GNU system

If you are running another Unix-like system, you could use it to install Debian GNU/Linux without using the `debian-installer` described in the rest of this manual. This kind of install may be useful for users with otherwise unsupported hardware or on hosts which can't afford downtime. If you are interested in this technique, skip to the Secțiune D.3. This installation method is only recommended for advanced users when no other installation method is available.

2.4.6 Supported Storage Systems

The Debian installer contains a kernel which is built to maximize the number of systems it runs on.

Generally, the Debian installation system includes support for IDE (also known as PATA) drives, SATA and SCSI controllers and drives, USB, and FireWire. The supported file systems include FAT, Win-32 FAT extensions (VFAT) and NTFS.

2.5 Memory and Disk Space Requirements

You must have at least 550MB of memory and 850MB of hard disk space to perform a normal installation. Note that these are fairly minimal numbers. For more realistic figures, see Secțiune 3.4.

Installation on systems with less memory[1] or disk space available may be possible but is only advised for experienced users.

[1] Installation images that support the graphical installer require more memory than images that support only the textual installer and should not be used on systems with less than 550MB of memory. If there is a choice between booting the text-based and the graphical installer, the former should be selected on such systems.

Capitolul 3

Before Installing Debian GNU/Linux

This chapter deals with the preparation for installing Debian before you even boot the installer. This includes backing up your data, gathering information about your hardware, and locating any necessary information.

3.1 Overview of the Installation Process

First, just a note about re-installations. With Debian, a circumstance that will require a complete re-installation of your system is very rare; perhaps mechanical failure of the hard disk would be the most common case.

Many common operating systems may require a complete installation to be performed when critical failures take place or for upgrades to new OS versions. Even if a completely new installation isn't required, often the programs you use must be re-installed to operate properly in the new OS.

Under Debian GNU/Linux, it is much more likely that your OS can be repaired rather than replaced if things go wrong. Upgrades never require a wholesale installation; you can always upgrade in-place. And the programs are almost always compatible with successive OS releases. If a new program version requires newer supporting software, the Debian packaging system ensures that all the necessary software is automatically identified and installed. The point is, much effort has been put into avoiding the need for re-installation, so think of it as your very last option. The installer is *not* designed to re-install over an existing system.

Here's a road map for the steps you will take during the installation process.

1. Back up any existing data or documents on the hard disk where you plan to install.
2. Gather information about your computer and any needed documentation, before starting the installation.
3. Locate and/or download the installer software and any specialized driver or firmware files your machine requires.
4. Set up boot media such as CDs/DVDs/USB sticks or provide a network boot infrastructure from which the installer can be booted.
5. Boot the installation system.
6. Select the installation language.
7. Activate the ethernet network connection, if available.
8. If necessary, resize existing partitions on your target harddisk to make space for the installation.
9. Create and mount the partitions on which Debian will be installed.
10. Watch the automatic download/install/setup of the *base system*.
11. Select and install *additional software*.
12. Install a *boot loader* which can start up Debian GNU/Linux and/or your existing system.
13. Load the newly installed system for the first time.

For 64-bit PC you have the option of using a graphical version of the installation system. For more information about this graphical installer, see Secțiune 5.1.8.

If you have problems during the installation, it helps to know which packages are involved in which steps. Introducing the leading software actors in this installation drama:

The installer software, `debian-installer`, is the primary concern of this manual. It detects hardware and loads appropriate drivers, uses `dhcp-client` to set up the network connection, runs `debootstrap` to install the base system packages, and runs `tasksel` to allow you to install certain additional software. Many more actors play smaller parts in this process, but `debian-installer` has completed its task when you load the new system for the first time.

To tune the system to your needs, `tasksel` allows you to choose to install various predefined bundles of software like a Web server or a Desktop environment.

One important option during the installation is whether or not to install a graphical desktop environment, consisting of the X Window System and one of the available graphical desktop environments. If you choose not to select the „Desktop environment" task, you will only have a relatively basic, command line driven system. Installing the Desktop environment task is optional because in relation to a text-mode-only system it requires a comparatively large amount of disk space and because many Debian GNU/Linux systems are servers which don't really have any need for a graphical user interface to do their job.

Just be aware that the X Window System is completely separate from `debian-installer`, and in fact is much more complicated. Troubleshooting of the X Window System is not within the scope of this manual.

3.2 Back Up Your Existing Data!

Before you start, make sure to back up every file that is now on your system. If this is the first time a non-native operating system is going to be installed on your computer, it is quite likely you will need to re-partition your disk to make room for Debian GNU/Linux. Anytime you partition your disk, you run a risk of losing everything on the disk, no matter what program you use to do it. The programs used in the installation of Debian GNU/Linux are quite reliable and most have seen years of use; but they are also quite powerful and a false move can cost you. Even after backing up, be careful and think about your answers and actions. Two minutes of thinking can save hours of unnecessary work.

If you are creating a multi-boot system, make sure that you have the distribution media of any other present operating systems on hand. Even though this is normally not necessary, there might be situations in which you could be required to reinstall your operating system's boot loader to make the system boot or in a worst case even have to reinstall the complete operating system and restore your previously made backup.

3.3 Information You Will Need

3.3.1 Documentation

3.3.1.1 Installation Manual

The document you are now reading, which is a development version of the Installation Guide for the next release of Debian; available in various formats and translations.

3.3.1.2 Hardware documentation

Often contains useful information on configuring or using your hardware.

- The Debian Wiki hardware page

3.3.2 Finding Sources of Hardware Information

In many cases, the installer will be able to automatically detect your hardware. But to be prepared, we do recommend familiarizing yourself with your hardware before the install.

Hardware information can be gathered from:

- The manuals that come with each piece of hardware.
- The BIOS setup screens of your computer. You can view these screens when you start your computer by pressing a combination of keys. Check your manual for the combination. Often, it is the **Delete** or the **F2** key,

but some manufacturers use other keys or key combinations. Usually upon starting the computer there will be a message stating which key to press to enter the setup screen.

- The cases and boxes for each piece of hardware.
- The System window in the Windows Control Panel.
- System commands or tools in another operating system, including file manager displays. This source is especially useful for information about RAM and hard drive memory.
- Your system administrator or Internet Service Provider. These sources can tell you the settings you need to set up your networking and e-mail.

Tabela 3.1 Hardware Information Helpful for an Install

Hardware	Information You Might Need
Hard Drives	How many you have.
	Their order on the system.
	Whether IDE (also known as PATA), SATA or SCSI.
	Available free space.
	Partitions.
	Partitions where other operating systems are installed.
Network interfaces	Type/model of available network interfaces.
Printer	Model and manufacturer.
Video Card	Type/model and manufacturer.

3.3.3 Hardware Compatibility

Many products work without trouble on Linux. Moreover, hardware support in Linux is improving daily. However, Linux still does not run as many different types of hardware as some operating systems.

Drivers in Linux in most cases are not written for a certain „product" or „brand" from a specific manufacturer, but for a certain hardware/chipset. Many seemingly different products/brands are based on the same hardware design; it is not uncommon that chip manufacturers provide so-called „reference designs" for products based on their chips which are then used by several different device manufacturers and sold under lots of different product or brand names.

This has advantages and disadvantages. An advantage is that a driver for one chipset works with lots of different products from different manufacturers, as long as their product is based on the same chipset. The disadvantage is that it is not always easy to see which actual chipset is used in a certain product/brand. Unfortunately sometimes device manufacturers change the hardware base of their product without changing the product name or at least the product version number, so that when having two items of the same brand/product name bought at different times, they can sometimes be based on two different chipsets and therefore use two different drivers or there might be no driver at all for one of them.

For USB and PCI/PCI-Express/ExpressCard devices, a good way to find out on which chipset they are based is to look at their device IDs. All USB/PCI/PCI-Express/ExpressCard devices have so called „vendor" and „product" IDs, and the combination of these two is usually the same for any product based on the same chipset.

On Linux systems, these IDs can be read with the **lsusb** command for USB devices and with the **lspci -nn** command for PCI/PCI-Express/ExpressCard devices. The vendor and product IDs are usually given in the form of two hexadecimal numbers, seperated by a colon, such as „1d6b:0001".

An example for the output of **lsusb**: „Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub", whereby 1d6b is the vendor ID and 0002 is the product ID.

An example for the output of **lspci -nn** for an Ethernet card: „03:00.0 Ethernet controller [0200]: Realtek Semiconductor Co., Ltd. RTL8111/8168B PCI Express Gigabit Ethernet controller [10ec:8168] (rev 06)". The IDs are given inside the rightmost square brackets, i.e. here 10ec is the vendor- and 8168 is the product ID.

As another example, a graphics card could give the following output: „04:00.0 VGA compatible controller [0300]: Advanced Micro Devices [AMD] nee ATI RV710 [Radeon HD 4350] [1002:954f]".

On Windows systems, the IDs for a device can be found in the Windows device manager on the tab „details", where the vendor ID is prefixed with VEN_ and the product ID is prefixed with DEV_. On Windows 7 systems, you have to select the property „Hardware IDs" in the device manager's details tab to actually see the IDs, as they are not displayed by default.

Searching on the internet with the vendor/product ID, „Linux" and „driver" as the search terms often results in information regarding the driver support status for a certain chipset. If a search for the vendor/product ID does not yield usable results, a search for the chip code names, which are also often provided by lsusb and lspci („RTL8111"/„RTL8168B" in the network card example and „RV710" in the graphics card example), can help.

3.3.3.1 Testing hardware compatibility with a Live-System

Debian GNU/Linux is also available as a so-called „live system" for certain architectures. A live system is a preconfigured ready-to-use system in a compressed format that can be booted and used from a read-only medium like a CD or DVD. Using it by default does not create any permanent changes on your computer. You can change user settings and install additional programs from within the live system, but all this only happens in the computer's RAM, i.e. if you turn off the computer and boot the live system again, everything is reset to its defaults. If you want to see whether your hardware is supported by Debian GNU/Linux, the easiest way is to run a Debian live system on it and try it out.

There are a few limitations in using a live system. The first is that as all changes you do within the live system must be held in your computer's RAM, this only works on systems with enough RAM to do that, so installing additional large software packages may fail due to memory constraints. Another limitation with regards to hardware compatibility testing is that the official Debian GNU/Linux live system contains only free components, i.e. there are no non-free firmware files included in it. Such non-free packages can of course be installed manually within the system, but there is no automatic detection of required firmware files like in the `debian-installer`, so installation of non-free components must be done manually if needed.

Information about the available variants of the Debian live images can be found at the Debian Live Images website.

3.3.4 Network Settings

If your computer is connected to a fixed network (i.e. an Ethernet or equivalent connection — not a dialup/PPP connection) which is administered by somebody else, you should ask your network's system administrator for this information:

- Your host name (you may be able to decide this on your own).
- Your domain name.
- Your computer's IP address.
- The netmask to use with your network.
- The IP address of the default gateway system you should route to, if your network *has* a gateway.
- The system on your network that you should use as a DNS (Domain Name Service) server.

If the network you are connected to uses DHCP (Dynamic Host Configuration Protocol) for configuring network settings, you don't need this information because the DHCP server will provide it directly to your computer during the installation process.

If you have internet access via DSL or cable modem (i.e. over a cable tv network) and have a router (often provided preconfigured by your phone or catv provider) which handles your network connectivity, DHCP is usually available by default.

As a rule of thumb: if you run a Windows system in your home network and did not have to manually perform any network settings there to achieve Internet access, network connectivity in Debian GNU/Linux will also be configured automatically.

If you use a WLAN/WiFi network, you should find out:

- The ESSID („network name") of your wireless network.
- The WEP or WPA/WPA2 security key to access the network (if applicable).

3.4 Meeting Minimum Hardware Requirements

Once you have gathered information about your computer's hardware, check that your hardware will let you do the type of installation that you want to do.

Depending on your needs, you might manage with less than some of the recommended hardware listed in the table below. However, most users risk being frustrated if they ignore these suggestions.

A Pentium 4, 1GHz system is the minimum recommended for a desktop system.

Tabela 3.2 Recommended Minimum System Requirements

Install Type	RAM (minimum)	RAM (recommended)	Hard Drive
No desktop	256 megabytes	512 megabytes	2 gigabytes
With Desktop	1 gigabytes	2 gigabytes	10 gigabytes

The minimum values assumes that swap will be enabled. The „No desktop" value assumes that the non-graphical installer is used.

The actual minimum memory requirements are a lot less than the numbers listed in this table. With swap enabled, it is possible to install Debian with as little as 170MB. The same goes for the disk space requirements, especially if you pick and choose which applications to install; see Secțiune D.2 for additional information on disk space requirements.

It is possible to run a graphical desktop environment on older or low-end systems, but in that case it is recommended to install a window manager that is less resource-hungry than those of the GNOME or KDE Plasma desktop environments; alternatives include `xfce4`, `icewm` and `wmaker`, but there are others to choose from.

It is practically impossible to give general memory or disk space requirements for server installations as those very much depend on what the server is to be used for.

Remember that these sizes don't include all the other materials which are usually to be found, such as user files, mail, and data. It is always best to be generous when considering the space for your own files and data.

Disk space required for the smooth operation of the Debian GNU/Linux system itself is taken into account in these recommended system requirements. Notably, the `/var` partition contains a lot of state information specific to Debian in addition to its regular contents, like logfiles. The **dpkg** files (with information on all installed packages) can easily consume 40MB. Also, **apt** puts downloaded packages here before they are installed. You should usually allocate at least 200MB for `/var`, and a lot more if you install a graphical desktop environment.

3.5 Pre-Partitioning for Multi-Boot Systems

Partitioning your disk simply refers to the act of breaking up your disk into sections. Each section is then independent of the others. It's roughly equivalent to putting up walls inside a house; if you add furniture to one room it doesn't affect any other room.

If you already have an operating system on your system (Windows 9x, Windows NT/2000/XP/2003/Vista/7, OS/2, MacOS, Solaris, FreeBSD, ...) which uses the whole disk and you want to stick Debian on the same disk, you will need to repartition it. Debian requires its own hard disk partitions. It cannot be installed on Windows or Mac OS X partitions. It may be able to share some partitions with other Unix systems, but that's not covered here. At the very least you will need a dedicated partition for the Debian root filesystem.

You can find information about your current partition setup by using a partitioning tool for your current operating system, such as the integrated Disk Manager in Windows or fdisk in DOS. Partitioning tools always provide a way to show existing partitions without making changes.

In general, changing a partition with a file system already on it will destroy any information there. Thus you should always make backups before doing any repartitioning. Using the analogy of the house, you would probably want to move all the furniture out of the way before moving a wall or you risk destroying it.

Several modern operating systems offer the ability to move and resize certain existing partitions without destroying their contents. This allows making space for additional partitions without losing existing data. Even though this works quite well in most cases, making changes to the partitioning of a disk is an inherently dangerous action and should only be done after having made a full backup of all data. For FAT/FAT32 and NTFS partitions as used by DOS and Windows systems, the ability to move and resize them losslessly is provided both by `debian-installer` as well as by the integrated Disk Manager of Windows 7.

To losslessly resize an existing FAT or NTFS partition from within `debian-installer`, go to the partitioning step, select the option for manual partitioning, select the partition to resize, and simply specify its new size.

Creating and deleting partitions can be done from within `debian-installer` as well as from an existing operating system. As a rule of thumb, partitions should be created by the system for which they are to be used, i.e. partitions to be used by Debian GNU/Linux should be created from within `debian-installer` and partitions to be used from another operating system should be created from there. `debian-installer` is capable of creating non-Linux partitions, and partitions created this way usually work without problems when used in other operating systems, but there are a few rare corner cases in which this could cause problems, so if you want to be sure, use the native partitioning tools to create partitions for use by other operating systems.

If you are going to install more than one operating system on the same machine, you should install all other system(s) before proceeding with the Debian installation. Windows and other OS installations may destroy your

ability to start Debian, or encourage you to reformat non-native partitions.

You can recover from these actions or avoid them, but installing the native system first saves you trouble.

3.6 Pre-Installation Hardware and Operating System Setup

This section will walk you through pre-installation hardware setup, if any, that you will need to do prior to installing Debian. Generally, this involves checking and possibly changing BIOS/system firmware settings for your system. The „BIOS" or „system firmware" is the core software used by the hardware; it is most critically invoked during the bootstrap process (after power-up).

3.6.1 Invoking the BIOS Set-Up Menu

The BIOS provides the basic functions needed to boot your machine and to allow your operating system to access your hardware. Your system provides a BIOS setup menu, which is used to configure the BIOS. To enter the BIOS setup menu you have to press a key or key combination after turning on the computer. Often it is the **Delete** or the **F2** key, but some manufacturers use other keys. Usually upon starting the computer there will be a message stating which key to press to enter the setup screen.

3.6.2 Boot Device Selection

Within the BIOS setup menu, you can select which devices shall be checked in which sequence for a bootable operating system. Possible choices usually include the internal harddisks, the CD/DVD-ROM drive and USB mass storage devices such as USB sticks or external USB harddisks. On modern systems there is also often a possibility to enable network booting via PXE.

Depending on the installation media (CD/DVD ROM, USB stick, network boot) you have chosen you should enable the appropriate boot devices if they are not already enabled.

Most BIOS versions allow you to call up a boot menu on system startup in which you select from which device the computer should start for the current session. If this option is available, the BIOS usually displays a short message like „press **F12** for boot menu" on system startup. The actual key used to select this menu varies from system to system; commonly used keys are **F12**, **F11** and **F8**. Choosing a device from this menu does not change the default boot order of the BIOS, i.e. you can start once from a USB stick while having configured the internal harddisk as the normal primary boot device.

If your BIOS does not provide you with a boot menu to do ad-hoc choices of the current boot device, you will have to change your BIOS setup to make the device from which the `debian-installer` shall be booted the primary boot device.

Unfortunately some computers contain buggy BIOS versions. Booting `debian-installer` from a USB stick might not work even if there is an appropriate option in the BIOS setup menu and the stick is selected as the primary boot device. On some of these systems using a USB stick as boot medium is impossible; others can be tricked into booting from the stick by changing the device type in the BIOS setup from the default „USB harddisk" or „USB stick" to „USB ZIP" or „USB CDROM". In particular if you use an isohybrid installation image on a USB stick (see Secțiune 4.3.1), changing the device type to „USB CDROM" helps on some BIOSes which will not boot from a USB stick in USB harddisk mode. You may need to configure your BIOS to enable „USB legacy support".

If you cannot manipulate the BIOS to boot directly from a USB stick you still have the option of using an ISO copied to the stick. Boot `debian-installer` using Secțiune 4.4 and, after scanning the hard drives for an installer ISO image, select the USB device and choose an installation image.

3.6.3 Systems with UEFI firmware

UEFI („Unified Extensible Firmware Interface") is a new kind of system firmware that is used on many modern systems and is - among other uses - intended to replace the classic PC BIOS.

Currently most PC systems that use UEFI also have a so-called „Compatibility Support Module" (CSM) in the firmware, which provides exactly the same interfaces to an operating system as a classic PC BIOS, so that software written for the classic PC BIOS can be used unchanged. Nonetheless UEFI is intended to one day completely replace the old PC BIOS without being fully backwards-compatible and there are already a lot of systems with UEFI but without CSM.

On systems with UEFI there are a few things to take into consideration when installing an operating system. The way the firmware loads an operating system is fundamentally different between the classic BIOS (or UEFI in CSM mode) and native UEFI. One major difference is the way the harddisk partitions are recorded on the harddisk. While

the classic BIOS and UEFI in CSM mode use a DOS partition table, native UEFI uses a different partitioning scheme called „GUID Partition Table" (GPT). On a single disk, for all practical purposes only one of the two can be used and in case of a multi-boot setup with different operating systems on one disk, all of them must therefore use the same type of partition table. Booting from a disk with GPT is only possible in native UEFI mode, but using GPT becomes more and more common as hard disk sizes grow, because the classic DOS partition table cannot address disks larger than about 2 Terabytes while GPT allows for far larger disks. The other major difference between BIOS (or UEFI in CSM mode) and native UEFI is the location where boot code is stored and in which format it has to be. This means that different bootloaders are needed for each system.

The latter becomes important when booting `debian-installer` on a UEFI system with CSM because `debian-installer` checks whether it was started on a BIOS- or on a native UEFI system and installs the corresponding bootloader. Normally this simply works but there can be a problem in multi-boot environments. On some UEFI systems with CSM the default boot mode for removable devices can be different from what is actually used when booting from hard disk, so when booting the installer from a USB stick in a different mode from what is used when booting another already installed operating system from the hard disk, the wrong bootloader might be installed and the system might be unbootable after finishing the installation. When choosing the boot device from a firmware boot menu, some systems offer two seperate choices for each device, so that the user can select whether booting shall happen in CSM or in native UEFI mode.

Another UEFI-related topic is the so-called „secure boot" mechanism. Secure boot means a function of UEFI implementations that allows the firmware to only load and execute code that is cryptographically signed with certain keys and thereby blocking any (potentially malicious) boot code that is unsigned or signed with unknown keys. In practice the only key accepted by default on most UEFI systems with secure boot is a key from Microsoft used for signing the Windows bootloader. As the boot code used by `debian-installer` is not signed by Microsoft, booting the installer requires prior deactivation of secure boot in case it is enabled. Secure boot is often enabled by default on systems that come preinstalled with a 64-bit version of Windows 8 and there is unfortunately no standard way to disable it in the UEFI setup. On some systems, the option to disable secure boot is only made visible when a BIOS password has been set by the user, so if you have a system with secure boot enabled, but cannot find an option to disable it, try setting a BIOS password, powercycle the machine and look again for an appropriate option.

3.6.4 Disabling the Windows 8 „fast boot" feature

Windows 8 offers a feature called „fast boot" to cut down system startup time. Technically, when this feature is enabled, Windows 8 does not do a real shutdown and a real cold boot afterwards when ordered to shut down, but instead does something resembling a partial suspend to disk to reduce the „boot" time. As long as Windows 8 is the only operating system on the machine, this is unproblematic, but it can result in problems and data loss when you have a dual boot setup in which another operating system accesses the same filesystems as Windows 8 does. In that case the real state of the filesystem can be different from what Windows 8 believes it to be after the „boot" and this could cause filesystem corruption upon further write accesses to the filesystem. Therefore in a dual boot setup, to avoid filesystem corruption the „fast boot" feature has to be disabled within Windows.

It may also be necessary to disable „fast boot" to even allow access to UEFI setup to choose to boot another operating system or `debian-installer`. On some UEFI systems, the firmware will reduce „boot" time by not initialising the keyboard controller or USB hardware; in these cases, it is necessary to boot into Windows and disable this feature to allow for a change of boot order.

3.6.5 Hardware Issues to Watch Out For

USB BIOS support and keyboards If you have no PS/2-style keyboard, but only a USB model, on some very old PCs you may need to enable legacy keyboard emulation in your BIOS setup to be able to use your keyboard in the bootloader menu, but this is not an issue for modern systems. If your keyboard does not work in the bootloader menu, consult your mainboard manual and look in the BIOS for „Legacy keyboard emulation" or „USB keyboard support" options.

Capitolul 4

Obtaining System Installation Media

4.1 Official Debian GNU/Linux installation images

By far the easiest way to install Debian GNU/Linux is from a set of official Debian installation images. You can buy a set of CDs/DVDs from a vendor (see the CD vendors page). You may also download the installation images from a Debian mirror and make your own set, if you have a fast network connection and a CD/DVD burner (see the Debian CD/DVD page and Debian CD FAQ for detailed instructions). If you have such optical installation media, and they are bootable on your machine, which is the case on all modern PCs, you can skip right to Cap. 5. Much effort has been expended to ensure the most-used files are on the first CD and DVD image, so that a basic desktop installation can be done with only the first DVD or - to a limited extent - even with only the first CD image.

As CDs have a rather limited capacity by today's standards, not all graphical desktop environments are installable with only the first CD; for some desktop environments a CD installation requires either network connectivity during the installation to download the remaining files or additional CDs.

Also, keep in mind: if the installation media you are using don't contain some packages you need, you can always install those packages afterwards from your running new Debian system (after the installation has finished). If you need to know on which installation image to find a specific package, visit https://cdimage-search.debian.org/.

If your machine doesn't support booting from optical media (only relevant on very old PC systems), but you do have a set of CD/DVD, you can use an alternative strategy such as hard disk, usb stick, net boot, or manually loading the kernel from the disc to initially boot the system installer. The files you need for booting by another means are also on the disc; the Debian network archive and folder organization on the disc are identical. So when archive file paths are given below for particular files you need for booting, look for those files in the same directories and subdirectories on your installation media.

Once the installer is booted, it will be able to obtain all the other files it needs from the disc.

If you don't have an installation media set, then you will need to download the installer system files and place them on the hard disk or usb stick or a connected computer so they can be used to boot the installer.

4.2 Downloading Files from Debian Mirrors

To find the nearest (and thus probably the fastest) mirror, see the list of Debian mirrors.

4.2.1 Where to Find Installation Images

The installation images are located on each Debian mirror in the directory debian/dists/bullseye/main/installer-amd64/-current/images/ — the MANIFEST lists each image and its purpose.

4.3 Preparing Files for USB Memory Stick Booting

To prepare the USB stick, you will need a system where GNU/Linux is already running and where USB is supported. With current GNU/Linux systems the USB stick should be automatically recognized when you insert it. If it is not you should check that the usb-storage kernel module is loaded. When the USB stick is inserted, it will be mapped to a device named `/dev/sdX`, where the „X" is a letter in the range a-z. You should be able to see to which device the USB stick was mapped by running the command **dmesg** after inserting it. To write to your stick, you may have to turn off its write protection switch.

AVERTISMENT

The procedures described in this section will destroy anything already on the device! Make very sure that you use the correct device name for your USB stick. If you use the wrong device the result could be that all information on for example a hard disk could be lost.

4.3.1 Preparing a USB stick using a hybrid CD/DVD image

Debian installation images can now be written directly to a USB stick, which is a very easy way to make a bootable USB stick. Simply choose an image (such as the netinst, CD, DVD-1, or netboot) that will fit on your USB stick. See Secțiune 4.1 to get an installation image.

Alternatively, for very small USB sticks, only a few megabytes in size, you can download the `mini.iso` image from the `netboot` directory (at the location mentioned in Secțiune 4.2.1).

The installation image you choose should be written directly to the USB stick, overwriting its current contents. For example, when using an existing GNU/Linux system, the image file can be written to a USB stick as follows, after having made sure that the stick is unmounted:

```
# cp debian.iso /dev/sdX
# sync
```

Information about how to do this on other operating systems can be found in the Debian CD FAQ.

IMPORTANT

The image must be written to the whole-disk device and not a partition, e.g. /dev/sdb and not /dev/sdb1. Do not use tools like **unetbootin** which alter the image.

IMPORTANT

Simply writing the installation image to USB like this should work fine for most users. The other options below are more complex, mainly for people with specialised needs.

The hybrid image on the stick does not occupy all the storage space, so it may be worth considering using the free space to hold firmware files or packages or any other files of your choice. This could be useful if you have only one stick or just want to keep everything you need on one device.

Create a second, FAT partition on the stick, mount the partition and copy or unpack the firmware onto it. For example:

```
# mount /dev/sdX2 /mnt
# cd /mnt
# tar zxvf /path/to/firmware.tar.gz
# cd /
# umount /mnt
```

You might have written the `mini.iso` to the USB stick. In this case the second partition doesn't have to be created as, very nicely, it will already be present. Unplugging and replugging the USB stick should make the two partitions visible.

4.3.2 Manually copying files to the USB stick

An alternative way to set up your USB stick is to manually copy the installer files, and also an installation image to it. Note that the USB stick should be at least 1 GB in size (smaller setups are possible if you follow Secțiune 4.3.3).

There is an all-in-one file `hd-media/boot.img.gz` which contains all the installer files (including the kernel) as well as `syslinux` and its configuration file .

Note that, although convenient, this method does have one major disadvantage: the logical size of the device will be limited to 1 GB, even if the capacity of the USB stick is larger. You will need to repartition the USB stick and create new file systems to get its full capacity back if you ever want to use it for some different purpose.

To use this image simply extract it directly to your USB stick:

```
# zcat boot.img.gz > /dev/sdX
```

After that, mount the USB memory stick (**`mount /dev/sdX /mnt`**), which will now have a FAT filesystem on it, and copy a Debian ISO image (netinst or full CD/DVD) to it. Unmount the stick (**`umount /mnt`**) and you are done.

4.3.3 Manually copying files to the USB stick — the flexible way

If you like more flexibility or just want to know what's going on, you should use the following method to put the files on your stick. One advantage of using this method is that — if the capacity of your USB stick is large enough — you have the option of copying any ISO image, even a DVD image, to it.

4.3.3.1 Partitioning the USB stick

We will show how to set up the memory stick to use the first partition, instead of the entire device.

Notă

Since most USB sticks come pre-configured with a single FAT16 partition, you probably won't have to repartition or reformat the stick. If you have to do that anyway, use **cfdisk** or any other partitioning tool to create a FAT16 partition[a], install an MBR using:

```
# install-mbr /dev/sdX
```

The **install-mbr** command is contained in the `mbr` Debian package. Then create the filesystem using:

```
# mkdosfs /dev/sdX1
```

Take care that you use the correct device name for your USB stick. The **mkdosfs** command is contained in the `dosfstools` Debian package.

[a] Don't forget to set the „bootable" bootable flag.

In order to start the kernel after booting from the USB stick, we will put a boot loader on the stick. Although any boot loader (e.g. `lilo`) should work, it's convenient to use `syslinux`, since it uses a FAT16 partition and can be reconfigured by just editing a text file. Any operating system which supports the FAT file system can be used to make changes to the configuration of the boot loader.

To put `syslinux` on the FAT16 partition on your USB stick, install the `syslinux` and `mtools` packages on your system, and do:

```
# syslinux /dev/sdX1
```

Again, take care that you use the correct device name. The partition must not be mounted when starting **syslinux**. This procedure writes a boot sector to the partition and creates the file `ldlinux.sys` which contains the boot loader code.

4.3.3.2 Adding the installer image

Mount the partition (**`mount /dev/sdX1 /mnt`**) and copy the following installer image files to the stick:

- `vmlinuz` or `linux` (kernel binary)
- `initrd.gz` (initial ramdisk image)

You can choose between either the text-based or the graphical version of the installer. The latter can be found in the `gtk` subdirectory. If you want to rename the files, please note that `syslinux` can only process DOS (8.3) file names.

Next you should create a `syslinux.cfg` configuration file, which at a bare minimum should contain the following line (change the name of the kernel binary to „`linux`" if you used a `netboot` image):

```
default vmlinuz initrd=initrd.gz
```

For the graphical installer you should add **`vga=788`** to the line. Other parameters can be appended as desired.

To enable the boot prompt to permit further parameter appending, add a **`prompt 1`** line.

If you used an `hd-media` image, you should now copy the ISO file of a Debian ISO image[1] onto the stick. When you are done, unmount the USB memory stick (**`umount /mnt`**).

4.4 Preparing Files for Hard Disk Booting

The installer may be booted using boot files placed on an existing hard drive partition, either launched from another operating system or by invoking a boot loader directly from the BIOS.

A full, „pure network" installation can be achieved using this technique. This avoids all hassles of removable media, like finding and burning CD/DVD images.

4.4.1 Hard disk installer booting from Linux using LILO or GRUB

This section explains how to add to or even replace an existing linux installation using either **LILO** or **GRUB**.

At boot time, both bootloaders support loading in memory not only the kernel, but also a disk image. This RAM disk can be used as the root file-system by the kernel.

Copy the following files from the Debian archives to a convenient location on your hard drive (note that LILO can not boot from files on an NTFS file system), for instance to `/boot/newinstall/`.

- `vmlinuz` (kernel binary)
- `initrd.gz` (ramdisk image)

Finally, to configure the bootloader proceed to Secțiune 5.1.5.

4.4.2 Hard disk installer booting from DOS using loadlin

This section explains how to prepare your hard drive for booting the installer from DOS using **loadlin**.

Copy the following directories from a Debian installation image to `c:\`.

- `/install.amd` (kernel binary and ramdisk image)
- `/tools` (loadlin tool)

4.5 Preparing Files for TFTP Net Booting

If your machine is connected to a local area network, you may be able to boot it over the network from another machine, using TFTP. If you intend to boot the installation system from another machine, the boot files will need to be placed in specific locations on that machine, and the machine configured to support booting of your specific machine.

You need to set up a TFTP server, and for many machines a DHCP server, or BOOTP server.

BOOTP is an IP protocol that informs a computer of its IP address and where on the network to obtain a boot image. The DHCP (Dynamic Host Configuration Protocol) is a more flexible, backwards-compatible extension of BOOTP. Some systems can only be configured via DHCP.

The Trivial File Transfer Protocol (TFTP) is used to serve the boot image to the client. Theoretically, any server, on any platform, which implements these protocols, may be used. In the examples in this section, we shall provide commands for SunOS 4.x, SunOS 5.x (a.k.a. Solaris), and GNU/Linux.

[1] You can use either a netinst or a full CD/DVD image (see Secțiune 4.1). Be sure to select one that fits. Note that the „netboot `mini.iso`" image is not usable for this purpose.

Notă

For a Debian GNU/Linux server we recommend `tftpd-hpa`. It's written by the same author as the `syslinux` bootloader and is therefore least likely to cause issues. A good alternative is `atftpd`.

4.5.1 Setting up a DHCP server

One free software DHCP server is ISC **dhcpd**. For Debian GNU/Linux, the `isc-dhcp-server` package is recommended. Here is a sample configuration file for it (see `/etc/dhcp/dhcpd.conf`):

```
option domain-name "example.com";
option domain-name-servers ns1.example.com;
option subnet-mask 255.255.255.0;
default-lease-time 600;
max-lease-time 7200;
server-name "servername";

subnet 192.168.1.0 netmask 255.255.255.0 {
  range 192.168.1.200 192.168.1.253;
  option routers 192.168.1.1;
}

host clientname {
  filename "/tftpboot.img";
  server-name "servername";
  next-server servername;
  hardware ethernet 01:23:45:67:89:AB;
  fixed-address 192.168.1.90;
}
```

In this example, there is one server *`servername`* which performs all of the work of DHCP server, TFTP server, and network gateway. You will almost certainly need to change the domain-name options, as well as the server name and client hardware address. The *`filename`* option should be the name of the file which will be retrieved via TFTP.

After you have edited the **dhcpd** configuration file, restart it with **`/etc/init.d/isc-dhcp-server restart`**.

4.5.1.1 Enabling PXE Booting in the DHCP configuration

Here is another example for a `dhcp.conf` using the Pre-boot Execution Environment (PXE) method of TFTP.

```
option domain-name "example.com";

default-lease-time 600;
max-lease-time 7200;

allow booting;
allow bootp;

# The next paragraph needs to be modified to fit your case
subnet 192.168.1.0 netmask 255.255.255.0 {
  range 192.168.1.200 192.168.1.253;
  option broadcast-address 192.168.1.255;
# the gateway address which can be different
# (access to the internet for instance)
  option routers 192.168.1.1;
# indicate the dns you want to use
  option domain-name-servers 192.168.1.3;
}
```

```
group {
  next-server 192.168.1.3;
  host tftpclient {
# tftp client hardware address
  hardware ethernet  00:10:DC:27:6C:15;
  filename "pxelinux.0";
 }
}
```

Note that for PXE booting, the client filename `pxelinux.0` is a boot loader, not a kernel image (see Secțiune 4.5.4 below). If your machine uses UEFI to boot, you will have to specify a boot loader appropriate for UEFI machines, for example

```
group {
  next-server 192.168.1.3;
  host tftpclient {
# tftp client hardware address
  hardware ethernet  00:10:DC:27:6C:15;
  filename "debian-installer/amd64/bootnetx64.efi";
 }
}
```

4.5.2 Setting up a BOOTP server

There are two BOOTP servers available for GNU/Linux. The first is CMU **bootpd**. The other is actually a DHCP server: ISC **dhcpd**. In Debian GNU/Linux these are contained in the `bootp` and `isc-dhcp-server` packages respectively.

To use CMU **bootpd**, you must first uncomment (or add) the relevant line in `/etc/inetd.conf`. On Debian GNU/Linux, you can run **`update-inetd --enable bootps`**, then **`/etc/init.d/inetd reload`** to do so. Just in case your BOOTP server does not run Debian, the line in question should look like:

```
bootps  dgram  udp  wait  root  /usr/sbin/bootpd  bootpd -i -t 120
```

Now, you must create an `/etc/bootptab` file. This has the same sort of familiar and cryptic format as the good old BSD `printcap`, `termcap`, and `disktab` files. See the `bootptab` manual page for more information. For CMU **bootpd**, you will need to know the hardware (MAC) address of the client. Here is an example `/etc/bootptab`:

```
client:\
  hd=/tftpboot:\
  bf=tftpboot.img:\
  ip=192.168.1.90:\
  sm=255.255.255.0:\
  sa=192.168.1.1:\
  ha=0123456789AB:
```

You will need to change at least the „ha" option, which specifies the hardware address of the client. The „bf" option specifies the file a client should retrieve via TFTP; see Secțiune 4.5.4 for more details.

By contrast, setting up BOOTP with ISC **dhcpd** is really easy, because it treats BOOTP clients as a moderately special case of DHCP clients. Some architectures require a complex configuration for booting clients via BOOTP. If yours is one of those, read the section Secțiune 4.5.1. Otherwise you will probably be able to get away with simply adding the **`allow bootp`** directive to the configuration block for the subnet containing the client in `/etc/dhcp/dhcpd.conf`, and restart **dhcpd** with **`/etc/init.d/isc-dhcp-server restart`**.

4.5.3 Enabling the TFTP Server

To get the TFTP server ready to go, you should first make sure that **tftpd** is enabled.

In the case of `tftpd-hpa` there are two ways the service can be run. It can be started on demand by the system's `inetd` daemon, or it can be set up to run as an independent daemon. Which of these methods is used is selected when the package is installed and can be changed by reconfiguring the package.

Notă

Historically, TFTP servers used `/tftpboot` as directory to serve images from. However, Debian GNU/Linux packages may use other directories to comply with the Filesystem Hierarchy Standard. For example, `tftpd-hpa` by default uses `/srv/tftp`. You may have to adjust the configuration examples in this section accordingly.

All **in.tftpd** alternatives available in Debian should log TFTP requests to the system logs by default. Some of them support a **-v** argument to increase verbosity. It is recommended to check these log messages in case of boot problems as they are a good starting point for diagnosing the cause of errors.

4.5.4 Move TFTP Images Into Place

Next, place the TFTP boot image you need, as found in Secțiune 4.2.1, in the **tftpd** boot image directory. You may have to make a link from that file to the file which **tftpd** will use for booting a particular client. Unfortunately, the file name is determined by the TFTP client, and there are no strong standards.

For PXE booting, everything you should need is set up in the `netboot/netboot.tar.gz` tarball. Simply extract this tarball into the **tftpd** boot image directory. Make sure your dhcp server is configured to pass `pxelinux.0` to **tftpd** as the filename to boot. For UEFI machines, you will need to pass an appropriate EFI boot image name (such as `/debian-installer/amd64/bootnetx64.efi`).

4.6 Automatic Installation

For installing on multiple computers it's possible to do fully automatic installations. Debian packages intended for this include `fai-quickstart` (which can use an install server) and the Debian Installer itself. Have a look at the FAI home page for detailed information.

4.6.1 Automatic Installation Using the Debian Installer

The Debian Installer supports automating installs via preconfiguration files. A preconfiguration file can be loaded from the network or from removable media, and used to fill in answers to questions asked during the installation process.

Full documentation on preseeding including a working example that you can edit is in Anexa B.

Capitolul 5

Booting the Installation System

5.1 Booting the Installer on 64-bit PC

AVERTISMENT

If you have any other operating systems on your system that you wish to keep (dual boot setup), you should make sure that they have been properly shut down *before* you boot the installer. Installing an operating system while another operating system is in hibernation (has been suspended to disk) could result in loss of, or damage to the state of the suspended operating system which could cause problems when it is rebooted.

NOTĂ

For information on how to boot the graphical installer, see Secțiune 5.1.8.

5.1.1 Booting from USB Memory Stick

If your computer will boot from USB, this will probably be the easiest route for installation. Assuming you have prepared everything from Secțiune 3.6.2 and Secțiune 4.3, just plug your USB stick into some free USB connector and reboot the computer. The system should boot up, and unless you have used the flexible way to build the stick and not enabled it, you should be presented with a graphical boot menu (on hardware that supports it). Here you can select various installer options, or just hit **Enter**.

5.1.2 Booting from optical disc (CD/DVD)

If you have a set of optical discs, and your machine supports booting directly off those, great! Simply configure your system for booting off an optical disc as described in Secțiune 3.6.2, insert the disc, reboot, and proceed to the next chapter.

Note that certain optical drives may require special drivers, and thus be inaccessible in the early installation stages. If it turns out the standard way of booting off an optical disc doesn't work for your hardware, revisit this chapter and read about alternate kernels and installation methods which may work for you.

Even if you cannot boot from optical disc, you can probably install the Debian system components and any packages you want from such disc. Simply boot using a different medium and when it's time to install the operating system, base system, and any additional packages, point the installation system at the optical drive.

If you have problems booting, see Secțiune 5.4.

5.1.3 Booting from Windows

To start the installer from Windows, you can either

- obtain installation media as described in Secțiune 4.1 or Secțiune 4.3 or
- download a standalone Windows executable, which is available as tools/win32-loader/stable/win32-loader.exe on the Debian mirrors.

If you use optical installation media, a pre-installation program should be launched automatically when you insert the disc. In case Windows does not start it automatically, or if you are using a USB memory stick, you can run it manually by accessing the device and executing **setup.exe**.

After the program has been started, a few preliminary questions will be asked and the system will be prepared to reboot into the Debian GNU/Linux installer.

5.1.4 Booting from DOS using loadlin

Boot into DOS (not Windows). To do this, you can for instance boot from a recovery or diagnostic disk.

If you can access the installation CD, change the current drive to the CD-ROM drive, e.g.

```
d:
```

else make sure you have first prepared your hard disk as explained in Secțiune 4.4.2, and change the current drive to it if needed.

Enter the subdirectory for the flavor you chose, e.g.,

```
cd \install.amd
```

If you prefer using the graphical installer, enter the `gtk` sub-directory.

```
cd gtk
```

Next, execute **install.bat**. The kernel will load and launch the installer system.

5.1.5 Booting from Linux using LILO or GRUB

To boot the installer from hard disk, you must first download and place the needed files as described in Secțiune 4.4.

If you intend to use the hard drive only for booting and then download everything over the network, you should download the `netboot/debian-installer/amd64/initrd.gz` file and its corresponding kernel `netboot/debian-installer/amd64/linux`. This will allow you to repartition the hard disk from which you boot the installer, although you should do so with care.

Alternatively, if you intend to keep an existing partition on the hard drive unchanged during the install, you can download the `hd-media/initrd.gz` file and its kernel, as well as copy an installation image to the hard drive (make sure the file is named ending in `.iso`). The installer can then boot from the hard drive and install from the installation image, without needing the network.

For **LILO**, you will need to configure two essential things in `/etc/lilo.conf`:

- to load the `initrd.gz` installer at boot time;
- have the `vmlinuz` kernel use a RAM disk as its root partition.

Here is a `/etc/lilo.conf` example:

```
image=/boot/newinstall/vmlinuz
       label=newinstall
       initrd=/boot/newinstall/initrd.gz
```

For more details, refer to the initrd(4) and lilo.conf(5) man pages. Now run **`lilo`** and reboot.

The procedure for **GRUB1** is quite similar. Locate your `menu.lst` in the `/boot/grub/` directory (or sometimes `/boot/boot/grub/`) and add an entry for the installer, for example (assuming `/boot` is on the first partition of the first disk in the system):

```
title  New Install
root   (hd0,0)
kernel /boot/newinstall/vmlinuz
initrd /boot/newinstall/initrd.gz
```

The procedure for **GRUB2** is very similar. The file is named `grub.cfg` instead of `menu.lst`. An entry for the installer would be for instance for example:

```
menuentry 'New Install' {
insmod part_msdos
insmod ext2
set root='(hd0,msdos1)'
linux /boot/newinstall/vmlinuz
initrd /boot/newinstall/initrd.gz
}
```

From here on, there should be no difference between **GRUB** or **LILO**.

5.1.6 Booting with TFTP

Booting from the network requires that you have a network connection and a TFTP network boot server (and probably also a DHCP, RARP, or BOOTP server for automatic network configuration).

The server-side setup to support network booting is described in Secțiune 4.5.

There are various ways to do a TFTP boot on i386.

5.1.6.1 NIC or Motherboard that support PXE

It could be that your Network Interface Card or Motherboard provides PXE boot functionality. This is a Intel™ re-implementation of TFTP boot. If so, you may be able to configure your BIOS to boot from the network.

5.1.6.2 NIC with Network BootROM

It could be that your Network Interface Card provides TFTP boot functionality.

Let us (debian-boot@lists.debian.org) know how did you manage it. Please refer to this document.

5.1.6.3 Etherboot

The etherboot project provides bootdiskettes and even bootroms that do a TFTPboot.

5.1.7 The Boot Screen

When the installer boots, you should be presented with a friendly graphical screen showing the Debian logo and a menu:

```
Debian GNU/Linux installer boot menu

Graphical install
Install
Advanced options            >
Help
Install with speech synthesis
```

> Notă
>
> 
>
> This graphical screen will look very slightly different depending on how your computer has booted (BIOS or UEFI), but the same options will be shown.

Depending on the installation method you are using, the „Graphical install” option may not be available. Bi-arch images additionally have a 64 bit variant for each install option, right below it, thus almost doubling the number of options.

For a normal installation, select either the „Graphical install” or the „Install” entry — using either the arrow keys on your keyboard or by typing the first (highlighted) letter — and press **Enter** to boot the installer. The „Graphical install” entry is already selected by default.

The „Advanced options" entry gives access to a second menu that allows to boot the installer in expert mode, in rescue mode and for automated installs.

If you wish or need to add any boot parameters for either the installer or the kernel, press **Tab** (BIOS boot), or **e** then **down arrow** three times then **end** (UEFI boot). This will bring the boot command for the selected menu entry and allow you to edit it to suit your needs. Note that the keyboard layout at this point is still QWERTY. The help screens (see below) list some common possible options. Press **Enter** (BIOS boot) or **F10** (UEFI boot) to boot the installer with your options; pressing **Esc** will return you to the boot menu and undo any changes you made.

Choosing the „Help" entry will result in the first help screen being displayed which gives an overview of all available help screens. To return to the boot menu after the help screens have been displayed, type „menu" at the boot prompt and press **Enter**. All help screens have a boot prompt at which the boot command can be typed:

```
Press F1 for the help index, or ENTER to boot:
```

At this boot prompt you can either just press **Enter** to boot the installer with default options or enter a specific boot command and, optionally, boot parameters. A number of boot parameters which might be useful can be found on the various help screens. If you do add any parameters to the boot command line, be sure to first type the boot method (the default is **`install`**) and a space before the first parameter (e.g., **`install fb=false`**).

Notă

The keyboard is assumed to have a default American English layout at this point. This means that if your keyboard has a different (language-specific) layout, the characters that appear on the screen may be different from what you'd expect when you type parameters. Wikipedia has a schema of the US keyboard layout which can be used as a reference to find the correct keys to use.

Notă

If you are using a system that has the BIOS configured to use serial console, you may not be able to see the initial graphical splash screen upon booting the installer; you may even not see the boot menu. The same can happen if you are installing the system via a remote management device that provides a text interface to the VGA console. Examples of these devices include the text console of Compaq's „integrated Lights Out" (iLO) and HP's „Integrated Remote Assistant" (IRA).

To bypass the graphical boot screen you can either blindly press **Esc** to get a text boot prompt, or (equally blindly) press „H" followed by **Enter** to select the „Help" option described above. After that your keystrokes should be echoed at the prompt. To prevent the installer from using the framebuffer for the rest of the installation, you will also want to add **`vga=normal fb=false`** to the boot prompt, as described in the help text.

5.1.8 The Graphical Installer

The graphical version of the installer is only available for a limited number of architectures, including 64-bit PC . The functionality of the graphical installer is essentially the same as that of the text-based installer as it basically uses the same programs, but with a different frontend.

Although the functionality is identical, the graphical installer still has a few significant advantages. The main advantage is that it supports more languages, namely those that use a character set that cannot be displayed with the text-based „newt" frontend. It also has a few usability advantages such as the option to use a mouse, and in some cases several questions can be displayed on a single screen.

The graphical installer is available with all CD/DVD images and with the hd-media installation method. To boot the graphical installer simply select the relevant option from the boot menu. Expert and rescue mode for the graphical installer can be selected from the „Advanced options" menu. The previously used boot methods **`installgui`**, **`expertgui`** and **`rescuegui`** can still be used from the boot prompt which is shown after selecting the „Help" option in the boot menu.

There is also a graphical installer image that can be netbooted. And there is a special „mini" ISO image[1], which is mainly useful for testing.

Just as with the text-based installer it is possible to add boot parameters when starting the graphical installer.

Notă

The graphical installer requires significantly more memory to run than the text-based installer: 800MB. If insufficient memory is available, it will automatically fall back to the text-based „newt" frontend.

If the amount of memory in your system is below 550MB, the graphical installer may fail to boot at all while booting the text-based installer would still work. Using the text-based installer is recommended for systems with little available memory.

5.2 Accessibility

Some users may need specific support because of e.g. some visual impairment. USB braille displays are detected automatically (not serial displays connected via a serial-to-USB converter), but most other accessibility features have to be enabled manually. On machines that support it, the boot menu emits beeps when it is ready to receive keystrokes. It beeps once on BIOS systems, and beeps twice on UEFI systems. Some boot parameters can then be appended to enable accessibility features (see also Secțiune 5.1.7). Note that on most architectures the boot loader interprets your keyboard as a QWERTY keyboard.

5.2.1 Installer front-end

The Debian installer supports several front-ends for asking questions, with varying convenience for accessibility: notably, **`text`** uses plain text while **`newt`** uses text-based dialog boxes. The choice can be made at the boot prompt, see the documentation for **`DEBIAN_FRONTEND`** in Secțiune 5.3.2.

5.2.2 USB Braille Displays

USB braille displays should be automatically detected. A textual version of the installer will then be automatically selected, and support for the braille display will be automatically installed on the target system. You can thus just press **Enter** at the boot menu. Once `brltty` is started, you can choose a braille table by entering the preference menu. Documentation on key bindings for braille devices is available on the `brltty` website.

5.2.3 Serial Braille Displays

Serial braille displays cannot safely be automatically detected (since that may damage some of them). You thus need to append the **`brltty=`*`driver`*`,`*`port`** boot parameter to tell `brltty` which driver and port it should use. *`driver`* should be replaced by the two-letter driver code for your terminal (see the BRLTTY manual). *`port`* should be replaced by the name of the serial port the display is connected to, **`ttyS0`** is the default, **`ttyUSB0`** can be typically used when using a serial-to-USB converter. A third parameter can be provided, to choose the name of the braille table to be used (see the BRLTTY manual); the English table is the default. Note that the table can be changed later by entering the preference menu. A fourth parameter can be provided to pass parameters to the braille driver, such as **`protocol=foo`** which is needed for some rare models. Documentation on key bindings for braille devices is available on the `brltty` website.

5.2.4 Software Speech Synthesis

Support for software speech synthesis is available on all installer images which have the graphical installer, i.e. all netinst, CD and DVD images, and the netboot gtk variant. It can be activated by selecting it in the boot menu by typing **`s`** **Enter**. The textual version of the installer will then be automatically selected, and support for software speech synthesis will be automatically installed on the target system.

If several sound cards are detected, you will be prompted to press **Enter** when you hear speech from the desired sound card.

[1] The mini ISO image can be downloaded from a Debian mirror as described in Secțiune 4.2. Look for `netboot/gtk/mini.iso`.

The first question (language) is spoken in english, and the remainder of installation is spoken in the selected language (if available in `espeak`).

The default speech rate is quite slow. To make it faster, press CapsLock-6. To make it slower, press CapsLock-5. The default volume should be medium. To make it louder, press CapsLock-2. To make it quieter, press CapsLock-1. To get more details on the browsing shortcuts, see the Speakup guide. To just accept the default answer for a question, simply press **Enter** at the prompt. To provide an empty answer for a question, type **!** at the prompt. To get back to the previous question, type **<** at the prompt.

5.2.5 Hardware Speech Synthesis

Support for hardware speech synthesis devices is available on all installer images which have the graphical installer, i.e. all netinst, CD and DVD images, and the netboot gtk variant. You thus need to select a „Graphical install" entry in the boot menu.

Hardware speech synthesis devices cannot be automatically detected. You thus need to append the **`speakup.synth=`*`driver`*** boot parameter to tell `speakup` which driver it should use. *`driver`* should be replaced by the driver code for your device (see driver code list). The textual version of the installer will then be automatically selected, and support for the speech synthesis device will be automatically installed on the target system.

5.2.6 Board Devices

Some accessibility devices are actual boards that are plugged inside the machine and that read text directly from the video memory. To get them to work framebuffer support must be disabled by using the **`vga=normal fb=false`** boot parameter. This will however reduce the number of available languages.

If desired a textual version of the bootloader can be activated before adding the boot parameter by typing **`h`** **Enter**.

5.2.7 High-Contrast Theme

For users with low vision, the installer can use a high-contrast color theme that makes it more readable. To enable it, you can use the „Accessible high contrast" entry from the boot screen with the **`d`** shortcut, or append the **`theme=dark`** boot parameter.

5.2.8 Zoom

For users with low vision, the graphical installer has a very basic zoom support: the Control-+ and Control– shortcuts increase and decrease the font size.

5.2.9 Expert install, rescue mode, automated install

Expert, Rescue, and Automated installation choices are also available with accessibility support. To access them, one has to first enter the „Advanced options" submenu from the boot menu by typing **`a`**. When using a BIOS system (the boot menu will have beeped only once), this has to be followed by **Enter** ; for UEFI systems (the boot menu will have beeped twice) that must not be done. Then, to enable speech synthesis, **`s`** can optionally be pressed (followed again by **Enter** on BIOS systems but not on UEFI systems). From there, various shortcuts can be used: **`x`** for expert installation, **`r`** for rescue mode, or **`a`** for automated installation. Again these need to be followed by **Enter** when using a BIOS system.

The automated install choice allows to install Debian completely automatically by using preseeding, whose source can be entered after accessibility features get started. Preseeding itself is documented in Anexa B.

5.2.10 Accessibility of the installed system

Documentation on accessibility of the installed system is available on the Debian Accessibility wiki page.

5.3 Boot Parameters

Boot parameters are Linux kernel parameters which are generally used to make sure that peripherals are dealt with properly. For the most part, the kernel can auto-detect information about your peripherals. However, in some cases you'll have to help the kernel a bit.

If this is the first time you're booting the system, try the default boot parameters (i.e., don't try setting parameters) and see if it works correctly. It probably will. If not, you can reboot later and look for any special parameters that inform the system about your hardware.

Information on many boot parameters can be found in the Linux BootPrompt HOWTO, including tips for obscure hardware. This section contains only a sketch of the most salient parameters. Some common gotchas are included below in Secțiune 5.4.

5.3.1 Boot console

If you are booting with a serial console, generally the kernel will autodetect this. If you have a videocard (framebuffer) and a keyboard also attached to the computer which you wish to boot via serial console, you may have to pass the **console=*device*** argument to the kernel, where *device* is your serial device, which is usually something like ttyS0.

You may need to specify parameters for the serial port, such as speed and parity, for instance **console=ttyS0,9600n8**; other typical speeds may be 57600 or 115200. Be sure to specify this option after „---", so that it is copied into the bootloader configuration for the installed system (if supported by the installer for the bootloader).

In order to ensure the terminal type used by the installer matches your terminal emulator, the parameter **TERM=*type*** can be added. Note that the installer only supports the following terminal types: linux, bterm, ansi, vt102 and dumb. The default for serial console in debian-installer is **vt102**. If you are using an IPMI console, or a virtualization tool which does not provide conversion into such terminals types itself, e.g. QEMU/KVM, you can start it inside a **screen** session. That will indeed perform translation into the screen terminal type, which is very close to vt102.

5.3.2 Debian Installer Parameters

The installation system recognizes a few additional boot parameters[2] which may be useful.

A number of parameters have a „short form" that helps avoid the limitations of the kernel command line options and makes entering the parameters easier. If a parameter has a short form, it will be listed in brackets behind the (normal) long form. Examples in this manual will normally use the short form too.

debconf/priority (priority) This parameter sets the lowest priority of messages to be displayed.

The default installation uses **priority=high**. This means that both high and critical priority messages are shown, but medium and low priority messages are skipped. If problems are encountered, the installer adjusts the priority as needed.

If you add **priority=medium** as boot parameter, you will be shown the installation menu and gain more control over the installation. When **priority=low** is used, all messages are shown (this is equivalent to the *expert* boot method). With **priority=critical**, the installation system will display only critical messages and try to do the right thing without fuss.

DEBIAN_FRONTEND This boot parameter controls the type of user interface used for the installer. The current possible parameter settings are:

- **DEBIAN_FRONTEND=noninteractive**
- **DEBIAN_FRONTEND=text**
- **DEBIAN_FRONTEND=newt**
- **DEBIAN_FRONTEND=gtk**

The default frontend is **DEBIAN_FRONTEND=newt**. **DEBIAN_FRONTEND=text** may be preferable for serial console installs. Some specialized types of install media may only offer a limited selection of frontends, but the **newt** and **text** frontends are available on most default install media. On architectures that support it, the graphical installer uses the **gtk** frontend.

BOOT_DEBUG Setting this boot parameter to 2 will cause the installer's boot process to be verbosely logged. Setting it to 3 makes debug shells available at strategic points in the boot process. (Exit the shells to continue the boot process.)

BOOT_DEBUG=0 This is the default.

[2] With current kernels (2.6.9 or newer) you can use 32 command line options and 32 environment options. If these numbers are exceeded, the kernel will panic. Also there is a limit of 255 characters for the whole kernel command line, everything above this limit may be silently truncated.

BOOT_DEBUG=1 More verbose than usual.

BOOT_DEBUG=2 Lots of debugging information.

BOOT_DEBUG=3 Shells are run at various points in the boot process to allow detailed debugging. Exit the shell to continue the boot.

log_host, log_port Causes the installer to send log messages to a remote syslog on the specified host and port as well as to a local file. If not specified, the port defaults to the standard syslog port 514.

lowmem Can be used to force the installer to a lowmem level higher than the one the installer sets by default based on available memory. Possible values are 1 and 2. See also Secțiune 6.3.1.1.

noshell Prevents the installer from offering interactive shells on tty2 and tty3. Useful for unattended installations where physical security is limited.

debian-installer/framebuffer (fb) Some architectures use the kernel framebuffer to offer installation in a number of languages. If framebuffer causes a problem on your system you can disable the feature using the parameter **vga=normal fb=false**. Problem symptoms are error messages about bterm or bogl, a blank screen, or a freeze within a few minutes after starting the install.

debian-installer/theme (theme) A theme determines how the user interface of the installer looks (colors, icons, etc.). What themes are available differs per frontend. Currently both the newt and gtk frontends only have a „dark” theme that was designed for visually impaired users. Set the theme by booting with **theme=*dark***.

netcfg/disable_autoconfig By default, the `debian-installer` automatically probes for network configuration via IPv6 autoconfiguration and DHCP. If the probe succeeds, you won't have a chance to review and change the obtained settings. You can get to the manual network setup only in case the automatic configuration fails.

If you have an IPv6 router or a DHCP server on your local network, but want to avoid them because e.g. they give wrong answers, you can use the parameter **netcfg/disable_autoconfig=true** to prevent any automatic configuration of the network (neither v4 nor v6) and to enter the information manually.

hw-detect/start_pcmcia Set to **false** to prevent starting PCMCIA services, if that causes problems. Some laptops are well known for this misbehavior.

disk-detect/dmraid/enable (dmraid) Set to **true** to enable support for Serial ATA RAID (also called ATA RAID, BIOS RAID or fake RAID) disks in the installer. Note that this support is currently experimental. Additional information can be found on the Debian Installer Wiki.

preseed/url (url) Specify the url to a preconfiguration file to download and use for automating the install. See Secțiune 4.6.

preseed/file (file) Specify the path to a preconfiguration file to load for automating the install. See Secțiune 4.6.

preseed/interactive Set to **true** to display questions even if they have been preseeded. Can be useful for testing or debugging a preconfiguration file. Note that this will have no effect on parameters that are passed as boot parameters, but for those a special syntax can be used. See Secțiune B.5.2 for details.

auto-install/enable (auto) Delay questions that are normally asked before preseeding is possible until after the network is configured. See Secțiune B.2.3 for details about using this to automate installs.

finish-install/keep-consoles During installations from serial or management console, the regular virtual consoles (VT1 to VT6) are normally disabled in `/etc/inittab`. Set to **true** to prevent this.

cdrom-detect/eject By default, before rebooting, `debian-installer` automatically ejects the optical media used during the installation. This can be unnecessary if the system does not automatically boot off such media. In some cases it may even be undesirable, for example if the optical drive cannot reinsert the media itself and the user is not there to do it manually. Many slot loading, slim-line, and caddy style drives cannot reload media automatically.

Set to **false** to disable automatic ejection, and be aware that you may need to ensure that the system does not automatically boot from the optical drive after the initial installation.

base-installer/install-recommends (recommends) By setting this option to **`false`**, the package management system will be configured to not automatically install „Recommends", both during the installation and for the installed system. See also Secțiune 6.3.5.

Note that this option allows to have a leaner system, but can also result in features being missing that you might normally expect to be available. You may have to manually install some of the recommended packages to obtain the full functionality you want. This option should therefore only be used by very experienced users.

debian-installer/allow_unauthenticated By default the installer requires that repositories be authenticated using a known gpg key. Set to **`true`** to disable that authentication. **Warning: insecure, not recommended.**

rescue/enable Set to **`true`** to enter rescue mode rather than performing a normal installation. See Secțiune 8.6.

5.3.3 Using boot parameters to answer questions

With some exceptions, a value can be set at the boot prompt for any question asked during the installation, though this is only really useful in specific cases. General instructions how to do this can be found in Secțiune B.2.2. Some specific examples are listed below.

debian-installer/language (language), debian-installer/country (country), debian-installer/locale (locale) There are two ways to specify the language, country and locale to use for the installation and the installed system.

The first and easiest is to pass only the parameter `locale`. Language and country will then be derived from its value. You can for example use **`locale=de_CH`** to select German as language and Switzerland as country (`de_CH.UTF-8` will be set as default locale for the installed system). Limitation is that not all possible combinations of language, country and locale can be achieved this way.

The second, more flexible option is to specify `language` and `country` separately. In this case `locale` can optionally be added to specify a specific default locale for the installed system. Example: **`language=en country=DE locale=en_GB.UTF-8`**.

anna/choose_modules (modules) Can be used to automatically load installer components that are not loaded by default. Examples of optional components that may be useful are `openssh-client-udeb` (so you can use **scp** during the installation) and `ppp-udeb` (see Secțiune D.5).

netcfg/disable_autoconfig Set to **`true`** if you want to disable IPv6 autoconfiguration and DHCP and instead force static network configuration.

mirror/protocol (protocol) By default the installer will use the http protocol to download files from Debian mirrors and changing that to ftp is not possible during installations at normal priority. By setting this parameter to **`ftp`**, you can force the installer to use that protocol instead. Note that you cannot select an ftp mirror from a list, you have to enter the hostname manually.

tasksel:tasksel/first (tasks) Can be used to select tasks that are not available from the interactive task list, such as the `kde-desktop` task. See Secțiune 6.3.6.2 for additional information.

5.3.4 Passing parameters to kernel modules

If drivers are compiled into the kernel, you can pass parameters to them as described in the kernel documentation. However, if drivers are compiled as modules and because kernel modules are loaded a bit differently during an installation than when booting an installed system, it is not possible to pass parameters to modules as you would normally do. Instead, you need to use a special syntax recognized by the installer which will then make sure that the parameters are saved in the proper configuration files and will thus be used when the modules are actually loaded. The parameters will also be propagated automatically to the configuration for the installed system.

Note that it is now quite rare that parameters need to be passed to modules. In most cases the kernel will be able to probe the hardware present in a system and set good defaults that way. However, in some situations it may still be needed to set parameters manually.

The syntax to use to set parameters for modules is:

```
module_name.parameter_name=value
```

If you need to pass multiple parameters to the same or different modules, just repeat this. For example, to set an old 3Com network interface card to use the BNC (coax) connector and IRQ 10, you would pass:

```
3c509.xcvr=3 3c509.irq=10
```

5.3.5 Blacklisting kernel modules

Sometimes it may be necessary to blacklist a module to prevent it from being loaded automatically by the kernel and udev. One reason could be that a particular module causes problems with your hardware. The kernel also sometimes lists two different drivers for the same device. This can cause the device to not work correctly if the drivers conflict or if the wrong driver is loaded first.

You can blacklist a module using the following syntax: ***module_name*.blacklist=yes**. This will cause the module to be blacklisted in `/etc/modprobe.d/blacklist.local` both during the installation and for the installed system.

Note that a module may still be loaded by the installation system itself. You can prevent that from happening by running the installation in expert mode and unselecting the module from the list of modules displayed during the hardware detection phases.

5.4 Troubleshooting the Installation Process

5.4.1 Reliability of optical media

Sometimes, especially with older drives, the installer may fail to boot from an optical disc. The installer may also — even after booting successfully from such disc — fail to recognize the disc or return errors while reading from it during the installation.

There are many different possible causes for these problems. We can only list some common issues and provide general suggestions on how to deal with them. The rest is up to you.

There are two very simple things that you should try first.

- If the disc does not boot, check that it was inserted correctly and that it is not dirty.
- If the installer fails to recognize the disc, try just running the option Detect and mount installation media a second time. Some DMA related issues with very old CD-ROM drives are known to be resolved in this way.

If this does not work, then try the suggestions in the subsections below. Most, but not all, suggestions discussed there are valid for CD-ROM and DVD.

If you cannot get the installation working from optical disc, try one of the other installation methods that are available.

5.4.1.1 Common issues

- Some older CD-ROM drives do not support reading from discs that were burned at high speeds using a modern CD writer.
- Some very old CD-ROM drives do not work correctly if „direct memory access” (DMA) is enabled for them.

5.4.1.2 How to investigate and maybe solve issues

If the optical disc fails to boot, try the suggestions listed below.

- Check that your BIOS actually supports booting from optical disc (only an issue for very old systems) and that booting from such media is enabled in the BIOS.
- If you downloaded an iso image, check that the md5sum of that image matches the one listed for the image in the `MD5SUMS` file that should be present in the same location as where you downloaded the image from.

```
$ md5sum debian-testing-i386-netinst.iso
a20391b12f7ff22ef705cee4059c6b92  debian-testing-i386-netinst.iso
```

Next, check that the md5sum of the burned disc matches as well. The following command should work. It uses the size of the image to read the correct number of bytes from the disc.

```
$ dd if=/dev/cdrom | \
> head -c `stat --format=%s debian-testing-i386-netinst.iso` | \
> md5sum
a20391b12f7ff22ef705cee4059c6b92  -
262668+0 records in
262668+0 records out
134486016 bytes (134 MB) copied, 97.474 seconds, 1.4 MB/s
```

If, after the installer has been booted successfully, the disc is not detected, sometimes simply trying again may solve the problem. If you have more than one optical drive, try changing the disc to the other drive. If that does not work or if the disc is recognized but there are errors when reading from it, try the suggestions listed below. Some basic knowledge of Linux is required for this. To execute any of the commands, you should first switch to the second virtual console (VT2) and activate the shell there.

- Switch to VT4 or view the contents of `/var/log/syslog` (use **nano** as editor) to check for any specific error messages. After that, also check the output of **dmesg**.
- Check in the output of **dmesg** if your optical drive was recognized. You should see something like (the lines do not necessarily have to be consecutive):

```
ata1.00: ATAPI: MATSHITADVD-RAM UJ-822S, 1.61, max UDMA/33
ata1.00: configured for UDMA/33
scsi 0:0:0:0: CD-ROM            MATSHITA DVD-RAM UJ-822S  1.61 PQ: 0 ANSI: 5
sr0: scsi3-mmc drive: 24x/24x writer dvd-ram cd/rw xa/form2 cdda tray
cdrom: Uniform CD-ROM driver Revision: 3.20
```

  If you don't see something like that, chances are the controller your drive is connected to was not recognized or may be not supported at all. If you know what driver is needed for the controller, you can try loading it manually using **modprobe**.
- Check that there is a device node for your optical drive under `/dev/`. In the example above, this would be `/dev/sr0`. There should also be a `/dev/cdrom`.
- Use the **mount** command to check if the optical disc is already mounted; if not, try mounting it manually:

```
$ mount /dev/hdc /cdrom
```

  Check if there are any error messages after that command.
- Check if DMA is currently enabled:

```
$ cd /proc/ide/hdc
$ grep using_dma settings
using_dma       1       0       1       rw
```

  A „1" in the first column after `using_dma` means it is enabled. If it is, try disabling it:

```
$ echo -n "using_dma:0" >settings
```

  Make sure that you are in the directory for the device that corresponds to your optical drive.
- If there are any problems during the installation, try checking the integrity of the installation media using the option near the bottom of the installer's main menu. This option can also be used as a general test if the disc can be read reliably.

5.4.2 Boot Configuration

If you have problems and the kernel hangs during the boot process, doesn't recognize peripherals you actually have, or drives are not recognized properly, the first thing to check is the boot parameters, as discussed in Secțiune 5.3.

In some cases, malfunctions can be caused by missing device firmware (see Secțiune 2.2 and Secțiune 6.4).

5.4.3 Software Speech Synthesis

If software speech synthesis does not work, there is most probably an issue with your sound board, usually because either the driver for it is not included in the installer, or because it has unusual mixer level names which are set to muted by default. You should thus submit a bug report which includes the output of the following commands, run on the same machine from a Linux system which is known to have sound working (e.g., a live CD).

- **`dmesg`**
- **`lspci`**
- **`lsmod`**
- **`amixer`**

5.4.4 Common 64-bit PC Installation Problems

There are some common installation problems that can be solved or avoided by passing certain boot parameters to the installer.

If your screen begins to show a weird picture while the kernel boots, eg. pure white, pure black or colored pixel garbage, your system may contain a problematic video card which does not switch to the framebuffer mode properly. Then you can use the boot parameter **fb=false** to disable the framebuffer console. Only a reduced set of languages will be available during the installation due to limited console features. See Secțiune 5.3 for details.

5.4.4.1 System Freeze During the PCMCIA Configuration Phase

Some very old laptop models produced by Dell are known to crash when PCMCIA device detection tries to access some hardware addresses. Other laptops may display similar problems. If you experience such a problem and you don't need PCMCIA support during the installation, you can disable PCMCIA using the **hw-detect/start_pcmcia=false** boot parameter. You can then configure PCMCIA after the installation is completed and exclude the resource range causing the problems.

Alternatively, you can boot the installer in expert mode. You will then be asked to enter the resource range options your hardware needs. For example, if you have one of the Dell laptops mentioned above, you should enter **exclude port 0x800-0x8ff** here. There is also a list of some common resource range options in the System resource settings section of the PCMCIA HOWTO. Note that you have to omit the commas, if any, when you enter this value in the installer.

5.4.5 Interpreting the Kernel Startup Messages

During the boot sequence, you may see many messages in the form `can't find` *something*, or *something* `not present`, `can't initialize` *something*, or even `this driver release depends on` *something*. Most of these messages are harmless. You see them because the kernel for the installation system is built to run on computers with many different peripheral devices. Obviously, no one computer will have every possible peripheral device, so the operating system may emit a few complaints while it looks for peripherals you don't own. You may also see the system pause for a while. This happens when it is waiting for a device to respond, and that device is not present on your system. If you find the time it takes to boot the system unacceptably long, you can create a custom kernel later (see Secțiune 8.5).

5.4.6 Reporting Installation Problems

If you get through the initial boot phase but cannot complete the install, the menu option Save debug logs may be helpful. It lets you store system error logs and configuration information from the installer on a storage medium, or download them using a web browser. This information may provide clues as to what went wrong and how to fix it. If you are submitting a bug report, you may want to attach this information to the bug report.

Other pertinent installation messages may be found in `/var/log/` during the installation, and `/var/log/installer/` after the computer has been booted into the installed system.

5.4.7 Submitting Installation Reports

If you still have problems, please submit an installation report. We also encourage installation reports to be sent even if the installation is successful, so that we can get as much information as possible on the largest number of hardware configurations.

Note that your installation report will be published in the Debian Bug Tracking System (BTS) and forwarded to a public mailing list. Make sure that you use an e-mail address that you do not mind being made public.

If you have a working Debian system, the easiest way to send an installation report is to install the `installation-report` and `reportbug` packages (**apt install installation-report reportbug**), configure `reportbug` as explained in Secțiune 8.4.2, and run the command **reportbug installation-reports**.

Alternatively you can use this template when filling out installation reports, and file the report as a bug report against the `installation-reports` pseudo package, by sending it to submit@bugs.debian.org.

```
Package: installation-reports

Boot method: <How did you boot the installer? CD/DVD? USB stick? Network?>
Image version: <Full URL to image you downloaded is best>
Date: <Date and time of the install>
```

```
Machine: <Description of machine (eg, IBM Thinkpad R32)>
Processor:
Memory:
Partitions: <df -Tl will do; the raw partition table is preferred>

Output of lspci -knn (or lspci -nn):

Base System Installation Checklist:
[O] = OK, [E] = Error (please elaborate below), [ ] = didn't try it

Initial boot:           [ ]
Detect network card:    [ ]
Configure network:      [ ]
Detect media:           [ ]
Load installer modules: [ ]
Detect hard drives:     [ ]
Partition hard drives:  [ ]
Install base system:    [ ]
Clock/timezone setup:   [ ]
User/password setup:    [ ]
Install tasks:          [ ]
Install boot loader:    [ ]
Overall install:        [ ]

Comments/Problems:

<Description of the install, in prose, and any thoughts, comments
      and ideas you had during the initial install.>
```

In the bug report, describe what the problem is, including the last visible kernel messages in the event of a kernel hang. Describe the steps that you did which brought the system into the problem state.

Capitolul 6

Cum se folosește Programul de instalare al Debian

6.1 Cum funcționează programul de instalare

Programul de instalare al Debian este format dintr-o colecție de componente cu scop specific în procesul de instalare , la fiecare dintre pașii acestuia. Fiecare componentă își efectuează propria sarcină, punând necesarul întrebări utilizatorului pentru a-și putea îndeplini sarcina. Aceste întrebări au priorități și pragul priorității întrebărilor care sunt puse utilizatorului este stabilit la pornirea programului de instalare.

Când se efectuează o instalare implicită, numai întrebările esențiale (prioritate mare) vor fi afișate utilizatorului. Aceasta duce la un proces de instalare puternic automatizat cu puține intervenții din partea utilizatorului. Componentele sunt rulate automat secvențial; alegerea componentelor care sunt rulate depinde în principal de metoda de instalare folosită și de componentele hardware. Programul de instalare va folosi valorile implicite pentru întrebările neafișate.

Dacă apare vreo problemă, utilizatorul va vedea un mesaj de eroare și este posibil ca meniul programului de instalare să fie afișat pentru a selecta o altă acțiune. Dacă nu apar probleme, utilizatorul nu va vedea deloc meniul, ci va răspunde la întrebările fiecărei componente, la rând. Notifcările legate de erorile grave au prioritatea „critică” astfel încât utilizatorul va fi mereu notificat.

Some of the defaults that the installer uses can be influenced by passing boot arguments when `debian-installer` is started. If, for example, you wish to force static network configuration (IPv6 autoconfiguration and DHCP are used by default if available), you could add the boot parameter **`netcfg/disable_autoconfig=true`**. See Secțiune 5.3.2 for available options.

Utilizatorii avansați se vor simți, probabil, mai confortabil într-o interfață bazată pe meniu, unde fiecare pas este controlat de utilizator în loc ca programul de instalare să efectueze automat și secvențial pașii necesari la instalare. Pentru a folosi mediul de instalare într-o manieră bazată pe meniu, adăugați la argumentele de pornire **`priority=medium`**.

Dacă pentru vreuna dintre componentele calculatorului trebuie pasate opțiuni modulelor nucleului la instalare, va trebui să porniți programul de instalare în modul „expert”. Acest lucru se poate face fie folosind comanda **expert** pentru a porni programul de instalare, fie prin adăugarea parametrului de pornire **`priority=low`**. Modul expert vă oferă controlul total asupra `debian-installer`.

In the character-based environment the use of a mouse is not supported. Here are the keys you can use to navigate within the various dialogs. The **Tab** or **right** arrow keys move „forward”, and the Shift-Tab or **left** arrow keys move „backward” between displayed buttons and selections. The **up** and **down** arrow select different items within a scrollable list, and also scroll the list itself. In addition, in long lists, you can type a letter to cause the list to scroll directly to the section with items starting with the letter you typed and use **Pg-Up** and **Pg-Down** to scroll the list in sections. The **space bar** selects an item such as a checkbox. Use **Enter** to activate choices.

Some dialogs may offer additional help information. If help is available this will be indicated on the bottom line of the screen by displaying that help information can be accessed by pressing the **F1** key.

Mesajele de eroare și jurnalele sunt redirecționate spre a patra consolă. Puteți activa această consolă apăsând Alt stânga-F4 (țineți apăsată tasta **Alt stânga** în timp ce apăsați tasta funcțională **F4**); vă puteți întoarce la procesul de instalare cu Alt stânga-F1.

Aceste mesaje pot fi găsite și în `/var/log/syslog`. După instalare, acest jurnal este copiat în `/var/log/installer/syslog` pe noul sistem. Alte mesaje de la instalare pot fi găsite în `/var/log/` în timpul instalării și în `/var/log/installer/` după ce calculatorul a pornit în sistemul instalat.

6.1.1 Using the graphical installer

The graphical installer basically works the same as the text-based installer and thus the rest of this manual can be used to guide you through the installation process.

If you prefer using the keyboard over the mouse, there are two things you need to know. To expand a collapsed list (used for example for the selection of countries within continents), you can use the **+** and **-** keys. For questions where more than one item can be selected (e.g. task selection), you first need to tab to the Continue button after making your selections; hitting enter will toggle a selection, not activate Continue.

If a dialog offers additional help information, a Help button will be displayed. The help information can be accessed either by activating the button or by pressing the **F1** key.

To switch to another console, you will also need to use the **Ctrl** key, just as with the X Window System. For example, to switch to VT2 (the first debug shell) you would use: Ctrl-Left Alt-F2. The graphical installer itself runs on VT5, so you can use Left Alt-F5 to switch back.

6.2 Componentele - Introducere

Iată o listă a componentelor programului de instalare cu descrieri sumare ale scopurilor lor. Dacă aveți nevoie să aflați mai multe detalii legate de utilizarea unei anumite componente le puteți găsi în Secțiune 6.3.

meniul principal Afișează utilizatorului lista componentelor în timpul operării programului de instalare și pornește o componentă când este selectată. Întrebările meniului principal au prioritatea medie, deci, în cazul în care aveți prioritatea înaltă sau critică (înaltă este implicită), nu veți vedea meniul. Pe de altă parte, dacă apare o eroare care necesită intervenția dumneavoastră, este posibil ca prioritatea să fie coborâtă temporar pentru a vă permite să rezolvați problema în cauză, deci, în acest caz, este posibil să apară meniul.

You can get to the main menu by selecting the Go Back button repeatedly to back all the way out of the currently running component.

alegătorul de locală - localechooser Permite utilizatorului să selecteze opțiuni de localizare pentru sistemul de instalare și pentru cel instalat: limba, țara și localele. Programul de instalare va afișa mesajele în limba selectată, doar dacă nu se întâmplă ca traducerea pentru acea limbă să nu fie completă, caz în care mesajele ar putea fi afișate în engleză.

alegătorul de tastaură - console-setup Shows a list of keyboard (layouts), from which the user chooses the one which matches his own model.

detectorul de hardware - hw-detect Detectează automat cele mai multe dintre componentele sistemului, inclusiv plăcile de rețea, discurile și PCMCIA.

detectorul de cdrom - cdrom-detect Looks for and mounts a Debian installation media.

configuratorul de rețea - netcfg Configurează conexiunile la rețea ale calculatorului pentru a putea comunica prin internet.

examinatorul de imagini iso - iso-scan Caută imagini ISO (fișiere `.iso`) pe discurile fixe.

alegătorul de arhive - choose-mirror Prezintă o listă situri cu arhivele Debian. Utilizatorul poate alege sursa pachetelor pe care le instalează.

verificatorul de CD-uri - cdrom-checker Checks integrity of installation media. This way, the user may assure him/herself that the installation image was not corrupted.

memorie scăzută - lowmem Lowmem încearcă să detecteze sistemele cu memorie puțină și apoi face diverse trucuri pentru a înlătura părți nenecesare ale lui `debian-installer` din memorie (sacrificând câteva din facilități).

anna Anna's Not Nearly APT. Installs packages which have been retrieved from the chosen mirror or installation media.

configuratorul pentru utilizator - user-setup Pregătește parola „de root”, și adaugă un utilizator obișnuit.

configuratorul ceasului - clock-setup Actualizează ceasul sistemului și determină dacă ceasul este configurat conform cu UTC sau nu.

configuratorul fusului orar - tzsetup Selectează fusul orar, bazat pe locația selectată anterior.

managerul de partiții - partman Permite utilizatorului să partiționeze discurile atașate la sistem, creează sisteme de fișiere pe partițiile selectate și le atașează la punctele de montare. Sunt incluse și facilități interesante precum modul complet automatizat sau suportul pentru LVM. Acesta este unealta de partiționare preferată în Debian.

partiționatorul - partitioner Permite utilizatorului să partiționeze discurile atașate la sistem. Este ales un program de partiționare potrivit arhitecturii calculatorului dumneavoastră.

configuratorul de partiții - partconf Afișează o listă de partiții și creează sisteme de fișiere pe partițiile selectate, conform cu instrucțiunile utilizatorului.

partman-lvm Ajută utilizatorul la configurarea *LVM* (managerul de volume logice).

partman-md Permite utilizatorului să configureze *RAID-ul* (matrice redundantă de discuri ieftine) software. Acest RAID software este, de obicei, superior față de controlere RAID IDE ieftine (pseudo-hardware) găsite pe plăcile de bază mai noi.

instalatorul bazei - base-installer Instalează un set de pachete de bază care ar permite calculatorului să opereze în Debian GNU/Linux după ce va fi repornit.

pregătitorul apt - apt-setup Configurează apt, în mare parte automat, bazat pe mediul de pe care rulează mediul de instalare.

selectorul de pachete - pkgsel Folosește `tasksel` pentru a selecta și instala programe adiționale.

testorul de sisteme de operare - os-prober Detectează sistemele de operare instalate deja pe calculator și pasează această informație către instalatorul de încărcător de sistem, care vă poate oferi posibilitatea să adăugați sistemele de operare descoperite la meniul de pornire a încărcătorului de sistem. În acest fel utilizatorul poate alege cu ușurință, la pornire, sistemul de operare care să fie pornit.

instalatorul de încărcător de sistem - bootloader-installer The various bootloader installers each install a boot loader program on the hard disk, which is necessary for the computer to start up using Linux without using a USB stick or CD-ROM. Many boot loaders allow the user to choose an alternate operating system each time the computer boots.

consola - shell Permite utilizatorului să execute o interfață-consolă din meniu, sau să o execute în cea de-a doua consolă.

salvatorul de jurnale - save-logs Provides a way for the user to record information on a USB stick, network, hard disk, or other media when trouble is encountered, in order to accurately report installer software problems to Debian developers later.

6.3 Folosirea componentelor individuale

În această secțiune vom descrie în detaliu fiecare componentă a programului de instalare. Componentele au fost grupate în etape care ar trebui să fie ușor recunoscute de utilizatori. Sunt prezentate în ordinea în care apar în timpul instalării. A se reține că nu toate modulele vor fi folosite în orice instalare; care module care vor fi folosite de fapt, depinde de metoda de instalare folosită și de componentele hardware.

6.3.1 Pregătirea Programului de instalare al Debian și a configurației hardware

Let's assume the Debian Installer has booted and you are facing its first screen. At this time, the capabilities of `debian-installer` are still quite limited. It doesn't know much about your hardware, preferred language, or even the task it should perform. Don't worry. Because `debian-installer` is quite clever, it can automatically probe your hardware, locate the rest of its components and upgrade itself to a capable installation system. However, you still need to help `debian-installer` with some information it can't determine automatically (like selecting your preferred language, keyboard layout or desired network mirror).

Veți observa că `debian-installer` încearcă *detecția hardware* de câteva ori în timpul acestei etape. Prima încercare are ca țintă specifică componentele care sunt necesare încărcării componentelor programului de instalare (ex.: CD-ROM-ul sau placa de rețea). Cum este posibil ca toate drivere-le să fie disponibile în timpul acestei prime încercări, procesul de detecție trebuie repetat mai târziu.

During hardware detection `debian-installer` checks if any of the drivers for the hardware devices in your system require firmware to be loaded. If any firmware is requested but unavailable, a dialog will be displayed that allows the missing firmware to be loaded from a removable medium. See Secțiune 6.4 for further details.

6.3.1.1 Check available memory / low memory mode

Unul dintre primele lucruri pe care le face `debian-installer` este verificarea memoriei disponibile. Dacă memoria disponibilă este limitată, această componentă va face câteva schimbări în procesul de instalare care se speră că vă va permite să instalați Debian GNU/Linux pe sistemul dumneavoastră.

The first measure taken to reduce memory consumption by the installer is to disable translations, which means that the installation can only be done in English. Of course, you can still localize the installed system after the installation has completed.

Dacă aceasta nu este suficient, programul de instalare va reduce și mai mult consumul de memorie prin încărcarea doar a acelor componente ce sunt esențiale pentru a realiza o instalare de bază. Acest lucru reduce funcționalitatea sistemului de instalare. Vi se va oferi posibilitatea să încărcați manual componente adiționale, dar trebuie să realizați că fiecare componentă selectată va folosi memorie suplimentară lucru care ar putea să facă instalarea să eșueze.

Dacă programul de instalare rulează în modul „memorie puțină”, se recomandă crearea unei partiții de swap relativ mare (64–128Mo). Partiția de swap va fi folosită ca memorie virtuală și va crește cantitatea de memorie disponibilă în sistem. Programul de instalare va activa partiția de swap cât mai curând posibil în timpul procesului de instalare. A se reține că utilizarea masivă a swap-ului va reduce performanța sistemului și poate genera o activitate ridicată a discului.

În ciuda acestor măsuri, este totuși posibil ca sistemul să se blocheze, să apară erori neașteptate sau ca procesele să fie „omorâte” de către nucleu deoarece sistemul rămâne fără memorie (lucru care va rezulta în mesaje „Out of memory” la VT4 - consola virtuală 4 - și în syslog).

De exemplu, au existat rapoarte că încercarea de a crea un sistem mare de fișiere ext3 în modul „memorie puțină” eșuează dacă nu există spațiu de swap suficient. Dacă nici un spațiu de swap mai mare nu ajută, încercați să creați sistemul de fișiere ca ext2 (componentă esențială a programului de instalare). O partiție ext2 poate fi schimbată în ext3 după instalare.

Programul de instalare poate fi forțat să folosească un nivel mai ridicat pentru limita de memorie puțină decât cea bazată pe memoria disponibilă prin intermediul parametrului de pornire „lowmem” așa cum este descris în Secțiune 5.3.2.

6.3.1.2 Selectarea opțiunilor de localizare

In most cases the first questions you will be asked concern the selection of localization options to be used both for the installation and for the installed system. The localization options consist of language, location and locales.

Limba aleasă va fi folosită pentru restul procesului de instalare, cu condiția să existe traduceri pentru diversele ferestre de dialog afișate. Dacă nu există o traducere validă pentru limba selectată, programul de instalare va folosi implicit engleza.

The selected geographic location (in most cases a country) will be used later in the installation process to select the correct time zone and a Debian mirror appropriate for that country. Language and country together will help determine the default locale for your system and select the correct keyboard layout.

Veți fi întrebat mai întâi să selectați limba preferată. Numele limbilor sunt afișate atât în engleză (în partea stângă) și în limba respectivă (în partea dreaptă); numele din partea dreaptă sunt afișate chiar în tipul de scriere potrivit pentru limbă. Lista este ordonată după numele englezești. În capul listei există o opțiune suplimentară care vă permite să selectați locala „C” în loc de o limbă. Dacă alegeți locala „C”, procesul de instalare se va face în engleză, iar sistemul instalat nu va avea suport de localizare deoarece pachetul `locales` nu va fi instalat.

Next you will be asked to select your geographic location. If you selected a language that is recognized as an official language for more than one country[1], you will be shown a list of only those countries. To select a country that is not in that list, choose other (the last option). You will then be presented with a list of continents; selecting a continent will lead to a list of relevant countries on that continent.

If the language has only one country associated with it, a list of countries will be displayed for the continent or region the country belongs to, with that country selected as the default. Use the Go Back option to select countries on a different continent.

> Notă
>
> 
>
> It is important to select the country where you live or where you are located as it determines the time zone that will be configured for the installed system.

[1] In technical terms: where multiple locales exist for that language with differing country codes.

If you selected a combination of language and country for which no locale is defined and there exist multiple locales for the language, then the installer will allow you to choose which of those locales you prefer as the default locale for the installed system[2]. In all other cases a default locale will be selected based on the selected language and country.

Any default locale selected as described in the previous paragraph will use *UTF-8* as character encoding.

If you are installing at low priority, you will have the option of selecting additional locales, including so-called „legacy" locales[3], to be generated for the installed system; if you do, you will be asked which of the selected locales should be the default for the installed system.

6.3.1.3 Alegerea tastaturii

Tastaturile sunt adesea proiectate după caracterele utilizate într-o limbă. Selectați un aranjament care este conform cu tastatura folosită, sau selectați ceva apropiat, dacă aranjamentul de tastatură dorit nu este reprezentat. Odată instalat sistemul, veți putea să selectați un aranjament de tastatură dintr-o gamă mai largă de opțiuni (rulați **dpkg-reconfigure keyboard-configuration** ca root după ce ați terminat instalarea).

Move the highlight to the keyboard selection you desire and press **Enter**. Use the arrow keys to move the highlight — they are in the same place in all national language keyboard layouts, so they are independent of the keyboard configuration.

6.3.1.4 Căutarea imaginii ISO a Programului de instalare al Debian

Când se instalează folosind metoda *hd-media*, va exista un moment când va trebui să găsiți și să montați imaginea iso a programului de instalare al Debian, pentru a obține restul de fișiere de instalare. Componenta **iso-scan** face chiar acest lucru.

At first, **iso-scan** automatically mounts all block devices (e.g. partitions and logical volumes) which have some known filesystem on them and sequentially searches for filenames ending with `.iso` (or `.ISO` for that matter). Beware that the first attempt scans only files in the root directory and in the first level of subdirectories (i.e. it finds `/`*`whatever`*`.iso`, `/data/`*`whatever`*`.iso`, but not `/data/tmp/`*`whatever`*`.iso`). After an iso image has been found, **iso-scan** checks its content to determine if the image is a valid Debian iso image or not. In the former case we are done, in the latter **iso-scan** seeks for another image.

În cazul în care încercarea anterioară de a găsi o imagine iso eșuează, **iso-scan** vă va întreba dacă să se efectueze o căutare mai amănunțită. Această trecere nu doar caută în cele mai de sus directoare, ci chiar traversează întregul sistem de fișiere.

Dacă **iso-scan** nu vă descoperă imaginea iso a programului de instalare, reporniți în sistemul de operare original și verificați dacă imaginea are numele corect (dacă se termină în `.iso`), dacă imaginea se află pe un sistem de fișiere recunoscut de `debian-installer` și dacă nu este coruptă (verificați suma de control). Utilizatorii experimentați de Unix ar putea face aceste lucruri fără a reporni, de la a doua consolă.

Note that the partition (or disk) hosting the ISO image can't be reused during the installation process as it will be in use by the installer. To work-around this, and provided that you have enough system memory, the installer can copy the ISO image into RAM before mounting it. This is controlled by the low priority `iso-scan/copy_iso_to_ram` debconf question (it is only asked if the memory requirement is met).

6.3.1.5 Configuring the Network

La intrarea în acest pas, dacă sistemul detectează că aveți mai mult de un dispozitiv de rețea, vi se va cere să alegeți care dispozitiv va fi interfața de rețea *primară*, în consecință aceea care doriți să fie folosită la instalare. Celelalte interfețe nu vor fi configurate la acest moment. Veți putea să configurați alte interfețe după ce se va termina instalarea; a se vedea pagina de manual interfaces(5).

6.3.1.5.1 Automatic network configuration By default, `debian-installer` tries to configure your computer's network automatically as far as possible. If the automatic configuration fails, that may be caused by many factors ranging from an unplugged network cable to missing infrastructure for automatic configuration. For further explanation in case of errors, check the error messages on the fourth console. In any case, you will be asked if you want to retry, or if you want to perform a manual setup. Sometimes the network services used for autoconfiguration can be slow in their responses, so if you are sure everything is in place, simply start the autoconfiguration attempt again. If autoconfiguration fails repeatedly, you can instead choose the manual network setup.

[2] At medium and low priority you can always select your preferred locale from those available for the selected language (if there's more than one).

[3] Legacy locales are locales which do not use UTF-8, but one of the older standards for character encoding such as ISO 8859-1 (used by West European languages) or EUC-JP (used by Japanese).

6.3.1.5.2 Manual network configuration The manual network setup in turn asks you a number of questions about your network, notably `IP address`, `Netmask`, `Gateway`, `Name server addresses`, and a `Hostname`. Moreover, if you have a wireless network interface, you will be asked to provide your `Wireless ESSID` („wireless network name") and a `WEP key` or `WPA/WPA2 passphrase`. Fill in the answers from Secțiune 3.3.

Notă

Some technical details you might, or might not, find handy: the program assumes the network IP address is the bitwise-AND of your system's IP address and your netmask. The default broadcast address is calculated as the bitwise OR of your system's IP address with the bitwise negation of the netmask. It will also guess your gateway. If you can't find any of these answers, use the offered defaults — if necessary, you can change them by editing `/etc/network/interfaces` once the system has been installed.

6.3.1.5.3 IPv4 and IPv6 From Debian GNU/Linux 7.0 („Wheezy") onwards, `debian-installer` supports IPv6 as well as the „classic" IPv4. All combinations of IPv4 and IPv6 (IPv4-only, IPv6-only and dual-stack configurations) are supported.

Autoconfiguration for IPv4 is done via DHCP (Dynamic Host Configuration Protocol). Autoconfiguration for IPv6 supports stateless autoconfiguration using NDP (Neighbor Discovery Protocol, including recursive DNS server (RDNSS) assignment), stateful autoconfiguration via DHCPv6 and mixed stateless/stateful autoconfiguration (address configuration via NDP, additional parameters via DHCPv6).

6.3.2 Setting Up Users And Passwords

Just before configuring the clock, the installer will allow you to set up the „root" account and/or an account for the first user. Other user accounts can be created after the installation has been completed.

6.3.2.1 Set the Root Password

The *root* account is also called the *super-user*; it is a login that bypasses all security protection on your system. The root account should only be used to perform system administration, and only used for as short a time as possible.

Any password you create should contain at least 6 characters, and should contain both upper- and lower-case characters, as well as punctuation characters. Take extra care when setting your root password, since it is such a powerful account. Avoid dictionary words or use of any personal information which could be guessed.

If anyone ever tells you they need your root password, be extremely wary. You should normally never give your root password out, unless you are administering a machine with more than one system administrator.

In case you do not specify a password for the „root" user here, this account will be disabled but the **sudo** package will be installed later to enable administrative tasks to be carried out on the new system. By default, the first user created on the system will be allowed to use the **sudo** command to become root.

6.3.2.2 Create an Ordinary User

The system will ask you whether you wish to create an ordinary user account at this point. This account should be your main personal log-in. You should *not* use the root account for daily use or as your personal login.

Why not? Well, one reason to avoid using root's privileges is that it is very easy to do irreparable damage as root. Another reason is that you might be tricked into running a *Trojan-horse* program — that is a program that takes advantage of your super-user powers to compromise the security of your system behind your back. Any good book on Unix system administration will cover this topic in more detail — consider reading one if it is new to you.

You will first be prompted for the user's full name. Then you'll be asked for a name for the user account; generally your first name or something similar will suffice and indeed will be the default. Finally, you will be prompted for a password for this account.

If at any point after installation you would like to create another account, use the **adduser** command.

6.3.3 Configuring the Clock and Time Zone

Programul de instalare va încerca întâi să se conecteze la un server de timp din internet (folosind protocolul *NTP*) pentru a configura corect timpul sistemului. Dacă aceasta nu reușește, programul de instalare va presupune că timpul și data obținute de la ceasul sistemului la pornire sunt corecte. Ajustarea manuală a timpului sistemului în timpul instalării nu este posibilă.

Depending on the location selected earlier in the installation process, you may be shown a list of time zones relevant for that location. If your location has only one time zone and you are doing a default installation, you will not be asked anything and the system will assume that time zone.

In expert mode or when installing at medium priority, you will have the additional option to select „Coordinated Universal Time" (UTC) as time zone.

Dacă, dintr-un motiv sau altul doriți să folosiți în sistemul instalat un fus orar care *nu* este specific locului selectat, există două opțiuni.

1. The simplest option is to just select a different time zone after the installation has been completed and you've booted into the new system. The command to do this is:

   ```
   # dpkg-reconfigure tzdata
   ```

2. Alternatively, the time zone can be set at the very start of the installation by passing the parameter **`time/zone=`*`value`*** when you boot the installation system. The value should of course be a valid time zone, for example **`Europe/London`** or **`UTC`**.

For automated installations the time zone can be set to any desired value using preseeding.

6.3.4 Partiționarea și selectarea punctelor de montare

At this time, after hardware detection has been executed a final time, `debian-installer` should be at its full strength, customized for the user's needs and ready to do some real work. As the title of this section indicates, the main task of the next few components lies in partitioning your disks, creating filesystems, assigning mountpoints and optionally configuring closely related options like RAID, LVM or encrypted devices.

If you are uncomfortable with partitioning, or just want to know more details, see Anexa C.

La început vi se va oferi posibilitatea de a partiționa automat fie un întreg dispozitiv, fie spațiul liber de pe un disc. Aceasta se mai numește și partiționare „ghidată". Dacă nu doriți să partiționați automat, alegeți din meniu metoda Manuală .

6.3.4.1 Supported partitioning options

The partitioner used in `debian-installer` is fairly versatile. It allows to create many different partitioning schemes, using various partition tables, file systems and advanced block devices.

Exactly which options are available depends mainly on the architecture, but also on other factors. For example, on systems with limited internal memory some options may not be available. Defaults may vary as well. The type of partition table used by default can for example be different for large capacity hard disks than for smaller hard disks. Some options can only be changed when installing at medium or low debconf priority; at higher priorities sensible defaults will be used.

The installer supports various forms of advanced partitioning and use of storage devices, which in many cases can be used in combination.

- *Logical Volume Management (LVM)*
- *Software RAID*

  Supported are RAID levels 0, 1, 4, 5, 6 and 10.
- *Encryption*
- *Serial ATA RAID* (using `dmraid`)

  Also called „fake RAID" or „BIOS RAID". Support for Serial ATA RAID is currently only available if enabled when the installer is booted. Further information is available on our Wiki.
- *Multipath* (experimental)

  See our Wiki for information. Support for multipath is currently only available if enabled when the installer is booted.

The following file systems are supported.

- *ext2*, *ext3*, *ext4*

  The default file system selected in most cases is ext4; for `/boot` partitions ext2 will be selected by default when guided partitioning is used.

- *jfs* (not available on all architectures)
- *xfs* (not available on all architectures)
- *reiserfs* (optional; not available on all architectures)

  Support for the Reiser file system is no longer available by default. When the installer is running at medium or low debconf priority it can be enabled by selecting the `partman-reiserfs` component. Only version 3 of the file system is supported.

- *qnx4*

  Existing partitions will be recognized and it is possible to assign mount points for them. It is not possible to create new qnx4 partitions.

- *FAT16*, *FAT32*
- *NTFS* (read-only)

  Existing NTFS partitions can be resized and it is possible to assign mount points for them. It is not possible to create new NTFS partitions.

6.3.4.2 Guided Partitioning

If you choose guided partitioning, you may have three options: to create partitions directly on the hard disk (classic method), or to create them using Logical Volume Management (LVM), or to create them using encrypted LVM[4].

Notă

Opțiunea de a folosi LVM (criptat) este posibil să nu fie disponibilă pe toate arhitecturile.

Dacă se folosește „LVM” sau „LVM” criptat, programul de instalare va crea majoritatea partițiilor în interiorul unei partiții mari; avantajul acestei metode este că partițiile din această partiție mare pot fi redimensionate relativ ușor, mai târziu. În cazul LVM-ului criptat, această partiție mare nu va putea fi citită fără a cunoaște o parolă-frază specială, astfel oferind un nivel suplimentar de securitate pentru datele (personale ale) dumneavoastră.

Când se folosește LVM criptat, programul de instalare va și șterge automat discul prin scrierea de date aleatoare pe el. Acest lucru îmbunătățește și mai mult securizarea (pentru că distincția dintre părțile goale și cele utilizate ale discului este imposibilă și pentru că eventualele urme ale instalărilor precedente sunt șterse), dar poate dura ceva timp, în funcție de dimensiunea discului.

Notă

Dacă alegeți partiționarea ghidată folosind „LVM” sau „LVM” criptat, unele schimbări în tabela de partiții vor trebui scrise pe discul selectat în timpul pregătirii „LVM”. Aceste schimbări efectiv șterg datele care sunt în prezent pe discul fix selectat și nu le veți putea anula mai târziu. Totuși, programul de instalare vă va cere să confirmați aceste schimbări înainte de a fi scrise pe disc.

If you choose guided partitioning (either classic or using (encrypted) LVM) for a whole disk, you will first be asked to select the disk you want to use. Check that all your disks are listed and, if you have several disks, make sure you select the correct one. The order they are listed in may differ from what you are used to. The size of the disks may help to identify them.

[4] The installer will encrypt the LVM volume group using a 256 bit AES key and makes use of the kernel's „dm-crypt” support.

Any data on the disk you select will eventually be lost, but you will always be asked to confirm any changes before they are written to the disk. If you have selected the classic method of partitioning, you will be able to undo any changes right until the end; when using (encrypted) LVM this is not possible.

Apoi veți putea sa alegeți dintre schemele afișate afișate în tabela de mai jos. Toate schemele au argumente pro și contra, unele dintre ele fiind discutate în Anexa C. Dacă sunteți nesigur(ă), alegeți-o pe prima. Nu uitați că partiționarea ghidată are nevoie de un minim de spațiu liber pe disc cu care să lucreze. Dacă nu îi oferiți cel puțin 1 GO de spațiu (dependent de schema aleasă), partiționarea ghidată va eșua.

Schema de partiționare	**Spațiu minim**	**Partiții create**
Toate fișierele pe o partiție	600MO	/, swap
Partiție /home separată	500MO	/, /home, swap
Separate /home, /var and /tmp partitions	1GO	/, /home, /var, /tmp, swap

If you choose guided partitioning using (encrypted) LVM, the installer will also create a separate /boot partition. The other partitions, including the swap partition, will be created inside the LVM partition.

If you have booted in EFI mode then within the guided partitioning setup there will be an additional partition, formatted as a FAT32 bootable filesystem, for the EFI boot loader. This partition is known as an EFI System Partition (ESP). There is also an additional menu item in the formatting menu to manually set up a partition as an ESP.

După selectarea unei scheme, următorul ecran vă va arăta noua tabelă de partiții, inclusiv informații legate de felul în care, și dacă, vor fi formatate și dacă vor fi montate.

Lista partițiilor ar putea să arate așa:

```
SCSI1 (0,0,0) (sda) - 6.4 GO WDC AC36400L
      #1 primară   16.4 MO  B f ext2       /boot
      #2 primară  551.0 MO      swap       swap
      #3 primară    5.8 GO      ntfs
         pri/log    8.2 MO      SPAȚIU LIBER

SCSI2 (1,0,0) (sdb) - 80.0 GO ST380021A
      #1 primară   15.9 MO      ext3
      #2 primară  996.0 MO      fat16
      #3 primară    3.9 GO      xfs        /home
      #5 logical    6.0 GO    f ext4       /
      #6 logică     1.0 GO    f ext3       /var
      #7 logică   498.8 MO      ext3
```

Acest exemplu ilustrează două discuri fixe pe împărțite în câteva partiții; primul disc are ceva spațiu liber. Fiecare linie cu o partiție constă în numărul partiției, tipul său, dimensiunea, fanioane opționale, sistemul de fișiere și punctul de montare (dacă e cazul). Notă: această configurare nu poate creată folosind partiționarea ghidată, dar ilustrează variațiile posibile care pot fi obținute folosind partiționarea manuală.

Cu aceasta se termină partiționarea ghidată. Dacă sunteți mulțumit(ă) cu tabela de partiții generată, puteți alege Finalizează partiționarea și scrie modificările pe disc din meniu, pentru a pune în practică noua tabelă de partiții (după cum e descris la sfârșitul acestei secțiuni). Dacă nu sunteți mulțumit(ă), puteți alege Anulează schimbările făcute asupra partițiilor și să rulați partiționarea ghidată din nou sau să modificați schimbările propuse după cum se va descrie mai jos la partiționarea manuală.

6.3.4.3 Manual Partitioning

Un ecran similar cu cel dinainte va fi afișat dacă alegeți partiționarea manuală, cu excepția că va fi afișată tabela de partiții actuală și punctele de montare nu vor fi afișate. Felul în care să configurați manual tabela de partiții și felul în care să fie folosite de către noul dumneavoastră sistem Debian vor fi acoperite în restul acestei secțiuni.

If you select a pristine disk which has neither partitions nor free space on it, you will be asked if a new partition table should be created (this is needed so you can create new partitions). After this, a new line entitled „FREE SPACE" should appear in the table under the selected disk.

If you select some free space, you will have the opportunity to create a new partition. You will have to answer a quick series of questions about its size, type (primary or logical), and location (beginning or end of the free space). After this, you will be presented with a detailed overview of your new partition. The main setting is Use as:, which determines if the partition will have a file system on it, or be used for swap, software RAID, LVM, an encrypted file system, or not be used at all. Other settings include mountpoint, mount options, and bootable flag; which settings are

shown depends on how the partition is to be used. If you don't like the preselected defaults, feel free to change them to your liking. E.g. by selecting the option Use as:, you can choose a different filesystem for this partition, including options to use the partition for swap, software RAID, LVM, or not use it at all. When you are satisfied with your new partition, select Done setting up the partition and you will return to **partman**'s main screen.

If you decide you want to change something about your partition, simply select the partition, which will bring you to the partition configuration menu. This is the same screen as is used when creating a new partition, so you can change the same settings. One thing that may not be very obvious at a first glance is that you can resize the partition by selecting the item displaying the size of the partition. Filesystems known to work are at least fat16, fat32, ext2, ext3 and swap. This menu also allows you to delete a partition.

Asigurați-vă că ați creat cel puțin două partiții: una pentru sistemul de fișiere *rădăcină* (care trebuie montat ca /) și una pentru *spațiul de swap*. Dacă uitați să montați sistemul de fișiere rădăcină, **partman** nu vă va lăsa să continuați până nu corectați această problemă.

If you boot in EFI mode but forget to select and format an EFI System Partition, **partman** will detect this and will not let you continue until you allocate one.

Capabilitățile lui **partman** pot fi extinse cu module ale programului de instalare, dar acestea sunt dependente de arhitectura sistemului. Deci, dacă nu observați toate facilitățile promise, verificați că ați încărcat toate modulele necesare (ex.: `partman-ext3`, `partman-xfs`, sau `partman-lvm`).

După ce sunteți mulțumit(ă) de partiționare, selectați Finalizează partiționarea și scrie modificările pe disc din meniul de partiționare. Vi se va afișa un sumar al schimbărilor de făcut asupra discurilor și vi se va cere să confirmați că sistemele de fișiere ar trebui create așa cum s-a cerut.

6.3.4.4 Configuring Multidisk Devices (Software RAID)

If you have more than one harddrive[5] in your computer, you can use **partman-md** to set up your drives for increased performance and/or better reliability of your data. The result is called *Multidisk Device* (or after its most famous variant *software RAID*).

MD este, de fapt, un ansamblu de partiții aflate pe discuri diferite și care, combinate, formează un dispozitiv *logic*. Acest dispozitiv poate fi folosit ca o partiție obișnuită (drept urmare, în **partman** îl puteți formata, îi puteți asocia un punct de montare, etc.).

What benefits this brings depends on the type of MD device you are creating. Currently supported are:

RAID0 Is mainly aimed at performance. RAID0 splits all incoming data into *stripes* and distributes them equally over each disk in the array. This can increase the speed of read/write operations, but when one of the disks fails, you will lose *everything* (part of the information is still on the healthy disk(s), the other part *was* on the failed disk).

The typical use for RAID0 is a partition for video editing.

RAID1 Is suitable for setups where reliability is the first concern. It consists of several (usually two) equally-sized partitions where every partition contains exactly the same data. This essentially means three things. First, if one of your disks fails, you still have the data mirrored on the remaining disks. Second, you can use only a fraction of the available capacity (more precisely, it is the size of the smallest partition in the RAID). Third, file-reads are load-balanced among the disks, which can improve performance on a server, such as a file server, that tends to be loaded with more disk reads than writes.

Optionally you can have a spare disk in the array which will take the place of the failed disk in the case of failure.

RAID5 Is a good compromise between speed, reliability and data redundancy. RAID5 splits all incoming data into stripes and distributes them equally on all but one disk (similar to RAID0). Unlike RAID0, RAID5 also computes *parity* information, which gets written on the remaining disk. The parity disk is not static (that would be called RAID4), but is changing periodically, so the parity information is distributed equally on all disks. When one of the disks fails, the missing part of information can be computed from remaining data and its parity. RAID5 must consist of at least three active partitions. Optionally you can have a spare disk in the array which will take the place of the failed disk in the case of failure.

As you can see, RAID5 has a similar degree of reliability to RAID1 while achieving less redundancy. On the other hand, it might be a bit slower on write operations than RAID0 due to computation of parity information.

RAID6 Is similar to RAID5 except that it uses two parity devices instead of one.

A RAID6 array can survive up to two disk failures.

[5] To be honest, you can construct an MD device even from partitions residing on single physical drive, but that won't give any benefits.

RAID10 RAID10 combines striping (as in RAID0) and mirroring (as in RAID1). It creates `n` copies of incoming data and distributes them across the partitions so that none of the copies of the same data are on the same device. The default value of `n` is 2, but it can be set to something else in expert mode. The number of partitions used must be at least `n`. RAID10 has different layouts for distributing the copies. The default is near copies. Near copies have all of the copies at about the same offset on all of the disks. Far copies have the copies at different offsets on the disks. Offset copies copy the stripe, not the individual copies.

RAID10 can be used to achieve reliability and redundancy without the drawback of having to calculate parity.

To sum it up:

Tipul	**Minim de dispozitive**	**Dispozitive de rezervă**	**Supraviețuiește la un defect de disc?**	**Spațiu disponibil**
RAID0	2	nu	nu	Dimensiunea celei mai mici partiții înmulțită cu numărul de dispozitive din RAID
RAID1	2	opțional	da	Dimensiunea celei mai mici partiții din RAID
RAID5	3	opțional	da	Dimensiunea celei mai mici partiții înmulțită cu (numărul de dispozitive din RAID minus unu)
RAID6	4	opțional	da	Size of the smallest partition multiplied by (number of devices in RAID minus two)
RAID10	2	opțional	da	Total of all partitions divided by the number of chunk copies (defaults to two)

If you want to know more about Software RAID, have a look at Software RAID HOWTO.

Pentru a crea un dispozitiv MD, trebuie să aveți partițiile pe care le doriți parte din RAID, să fie marcate să fie folosite în RAID. (Aceasta se face în **partman** în meniul Configurația partiției unde ar trebui să selectați Folosită ca: → volum fizic pentru RAID.)

Notă

Make sure that the system can be booted with the partitioning scheme you are planning. In general it will be necessary to create a separate file system for `/boot` when using RAID for the root (`/`) file system. Most boot loaders (including lilo and grub) do support mirrored (not striped!) RAID1, so using for example RAID5 for `/` and RAID1 for `/boot` can be an option.

În continuare, va trebui să alegeți Configurează RAID-ul software din meniul principal al lui **partman**. (Meniul va apărea după ce marcați cel puțin o partiție să fie folosită ca volum fizic pentru RAID.) În primul ecran al **partman-md** doar selectați Creează dispozitiv MD. Vi se va afișa o listă de tipuri de dispozitive MD suportate din care trebuie să alegeți unul (ex. RAID1). Ceea ce va urma depinde de tipul de dispozitiv MD selectat.

- RAID0 este simplu — vi se va prezenta lista de partiții RAID disponibile și singurul lucru pe care va trebui să-l faceți este să selectați partițiile care vor forma dispozitivul MD.

- RAID1 is a bit more tricky. First, you will be asked to enter the number of active devices and the number of spare devices which will form the MD. Next, you need to select from the list of available RAID partitions those that will be active and then those that will be spare. The count of selected partitions must be equal to the number provided earlier. Don't worry. If you make a mistake and select a different number of partitions, `debian-installer` won't let you continue until you correct the issue.
- RAID5 has a setup procedure similar to RAID1 with the exception that you need to use at least *three* active partitions.
- Procedura de pregătire a RAID6 este similară cu cea pentru RAID1, cu excepția faptului că va trebui să folosiți cel puțin *patru* partiții active.
- Și RAID10 are o procedură de pregătire similară cu cea pentru RAID1, exceptând modul expert. În modul expert `debian-installer` vă va cere să precizați aranjamentul. Aranjamentul are două părți. Prima parte este tipul de aranjament. Este `n` (copii apropiate - eng. near), `f` (copii îndepărtate - eng. far), sau `o` (copii cu deplasament - eng. offset). A doua parte este numărul de copii de făcut pentru date. Trebuie să fie cel puțin atâtea dispozitive active astfel încât copiile să fie distribuite pe discuri diferite.

It is perfectly possible to have several types of MD at once. For example, if you have three 200 GB hard drives dedicated to MD, each containing two 100 GB partitions, you can combine the first partitions on all three disks into the RAID0 (fast 300 GB video editing partition) and use the other three partitions (2 active and 1 spare) for RAID1 (quite reliable 100 GB partition for `/home`).

După ce configurați dispozitivele MD după bunul dumneavoastră plac, puteți selecta Finalizează din **partman-md** pentru a vă întoarce la **partman** pentru a crea sisteme de fișiere pe noile dispozitive MD și să le asociați atributele obișnuite precum punctele de montare.

6.3.4.5 Configurarea Managerului de volume logice (LVM)

Dacă lucrați cu calculatoarele ca administrator sau ca utilizator „avansat", ați văzut, cu siguranță situația în care pe o partiție (de obicei, cea mai importantă) spațiul liber era foarte limitat, în timp ce o altă partiție era foarte puțin utilizată și a trebuit să rezolvați această problemă mutând fișiere dintr-o parte într-alta, făcând legături simbolice, etc.

Pentru a evita situația descrisă, puteți folosi Managerul de volume logice (LVM). Pe scurt, cu LVM puteți combina partițiile (*volume fizice* în jargonul LVM) pentru a forma un disc virtual (așa-numitul *grup de volume*) care poate fi apoi divizat în partiții virtuale (*volume logice*). Ideea este că volumele logice (și, desigur, grupele de volume care sunt la baza acestora) se pot întinde peste mai multe discuri fizice.

Acum, în momentul în care vă dați seama că aveți nevoie de mai mult spațiu pentru vechea partiție `/home` de 160GO, puteți, pur și simplu, să adăugați un disc nou de 300GO la calculator, să îl adăugați la grupul de volume existent și să redimensionați volumul logic care conține sistemul de fișiere `/home` și, ta-da — utilizatorii vor avea din nou spațiu pe partiția reînnoită de 460GO. Acest exemplu este, desigur, un pic simplificat. Dacă încă nu l-ați citit încă, ar fi bine să citiți Rețetarul LVM.

Pregătirea LVM în `debian-installer` este destul de simplă și complet suportată în **partman**. Mai întâi, va trebui să marcați partiția(iile) care să fie folosite ca volume fizice pentru LVM. Acest lucru se face în meniul Configurația partiției:, unde va trebui să selectați Folosită ca: → Volumul fizic pentru LVM

> AVERTISMENT
>
> 
>
> Be aware: the new LVM setup will destroy all data on all partitions marked with an LVM type code. So, if you already have an LVM on some of your disks, and want to install Debian additionally to that machine, the old (already existing) LVM will be wiped out! The same counts for partitions, which are (for any reason) misleadingly marked with an LVM type code, but contain something different (like an encrypted volume). You need to remove such disks from the system, before performing a new LVM setup!

Odată cu întoarcerea la ecranul principal al lui **partman** veți vedea o nouă opțiune Configurarea Managerului de volume logice (LVM). După selectarea acesteia, mai întâi vi se va cere să confirmați schimbările în așteptare ce se vor face asupra tabelei de partiții (dacă există) și apoi se va afișa meniul de configurare al LVM-ului. Meniul este sensibil la context și afișează doar acțiunile valide. Acțiunile posibile sunt:

- Display configuration details: shows LVM device structure, names and sizes of logical volumes and more

- Create volume group
- Create logical volume
- Delete volume group
- Delete logical volume
- Extend volume group
- Reduce volume group
- Finish: return to the main **partman** screen

Folosiți opțiunile din acel meniu pentru a crea mai întâi un grup de volume și și apoi creați volumele logice în interiorul acestuia.

După ce vă întoarceți la ecranul principal al lui **partman**, volumele logice create vor fi afișate la fel ca partițiile obișnuite (și ar trebui tratate ca atare).

6.3.4.6 Configurarea volumelor criptate

`debian-installer` vă oferă posibilitatea să configurați partiții criptate. Orice fișier scris pe o asemenea partiție este imediat salvat pe dispozitiv în formă criptată. Accesul la datele criptate este permis doar după introducerea *parolei-frază* folosite când a fost creată partiția criptată. Această facilitate este utilă la protejarea datelor confidențiale în eventualitatea furtului laptopului sau a discului fix. Hoțul ar putea căpăta acces fizic la discul fix, dar fără a cunoaște parola-frază corectă, datele de pe disc vor părea a fi caractere aleatoare.

The two most important partitions to encrypt are: the home partition, where your private data resides, and the swap partition, where sensitive data might be stored temporarily during operation. Of course, nothing prevents you from encrypting any other partitions that might be of interest. For example `/var` where database servers, mail servers or print servers store their data, or `/tmp` which is used by various programs to store potentially interesting temporary files. Some people may even want to encrypt their whole system. Generally the only exception here is the `/boot` partition which must remain unencrypted, because historically there was no way to load the kernel from an encrypted partition. (GRUB is now able to do that, but `debian-installer` currently lacks native support for encrypted `/boot`. The setup is therefore covered in a separate document.)

> Notă
>
> 
>
> A se reține că performanța partițiilor criptate va fi mai scăzută decât cea a partițiilor necriptate deoarece datele trebuie să fie criptate sau decriptate pentru fiecare operație de scriere sau citire. Impactul asupra performanței depinde de viteza procesorului, tipul de cifrare ales și lungimea cheii.

To use encryption, you have to create a new partition by selecting some free space in the main partitioning menu. Another option is to choose an existing partition (e.g. a regular partition, an LVM logical volume or a RAID volume). In the Partition settings menu, you need to select physical volume for encryption at the Use as: option. The menu will then change to include several cryptographic options for the partition.

The encryption method supported by `debian-installer` is *dm-crypt* (included in newer Linux kernels, able to host LVM physical volumes).

Let's have a look at the options available when you select encryption via **`Device-mapper (dm-crypt)`**. As always: when in doubt, use the defaults, because they have been carefully chosen with security in mind.

Criptare: `aes` This option lets you select the encryption algorithm (*cipher*) which will be used to encrypt the data on the partition. `debian-installer` currently supports the following block ciphers: *aes*, *blowfish*, *serpent*, and *twofish*. It is beyond the scope of this document to discuss the qualities of these different algorithms, however, it might help your decision to know that in 2000, *AES* was chosen by the American National Institute of Standards and Technology as the standard encryption algorithm for protecting sensitive information in the 21st century.

Dimensiunea cheii: `256` Aici puteți preciza lungimea cheii de criptare. Cu o cheie mai lungă, în general, se îmbunătățește puterea criptării. Pe de altă parte, mărirea cheii duce, de obicei, la o degradare a performanței. Disponibilitatea diverselor dimensiuni de chei depinde de algoritmul de criptare.

IV algorithm: `xts-plain64` *Vectorul de Inițializare* sau algoritmul *VI* este folosit în criptografie pentru a asigura întotdeauna unicitatea *textului cifrat*când se dă un aceleași date *text în clar* și o aceiași cheie. Ideea este de a preveni încercarea unui atacator de a deduce informații din șabloane repetate în datele criptate.

Dintre alternativele oferite, cea implicită, **`xts-plain64`** este în prezent cea mai puțin vulnerabilă la atacurile cunoscute. Folosiți celelalte alternative doar dacă trebuie să asigurați compatibilitatea cu un alt sistem instalat anterior care nu poate folosi algoritmii mai noi.

Cheia de criptare: `Parolă-frază` Aici puteți alege tipul de cheie de criptare pentru această partiție.

Parolă-frază Cheia de criptare va fi calculată [6] pe baza unei parole-frază pe care o veți putea introduce mai târziu în decursul procesului.

Cheie aleatoare O nouă cheie de criptare va fi generată din date aleatoare de fiecare dată când încercați să inițializați partiția criptată. Cu alte cuvinte, la fiecare oprire conținutul partiției se va pierde deoarece cheia este ștearsă din memorie. (Desigur, ați putea să încercați să ghiciți cheia cu un atac de tip „forță brută", dar acest lucru nu este realizabil în decursul unei întregi vieți, decât dacă există o slăbiciune necunoscută în algoritmul de cifrare.)

Random keys are useful for swap partitions because you do not need to bother yourself with remembering the passphrase or wiping sensitive information from the swap partition before shutting down your computer. However, it also means that you will *not* be able to use the „suspend-to-disk" functionality offered by newer Linux kernels as it will be impossible (during a subsequent boot) to recover the suspended data written to the swap partition.

Erase data: `yes` Determines whether the content of this partition should be overwritten with random data before setting up the encryption. This is recommended because it might otherwise be possible for an attacker to discern which parts of the partition are in use and which are not. In addition, this will make it harder to recover any leftover data from previous installations[7].

After you have selected the desired parameters for your encrypted partitions, return back to the main partitioning menu. There should now be a new menu item called Configure encrypted volumes. After you select it, you will be asked to confirm the deletion of data on partitions marked to be erased and possibly other actions such as writing a new partition table. For large partitions this might take some time.

Next you will be asked to enter a passphrase for partitions configured to use one. Good passphrases should be longer than 8 characters, should be a mixture of letters, numbers and other characters and should not contain common dictionary words or information easily associable with you (such as birthdates, hobbies, pet names, names of family members or relatives, etc.).

AVERTISMENT

Before you input any passphrases, you should have made sure that your keyboard is configured correctly and generates the expected characters. If you are unsure, you can switch to the second virtual console and type some text at the prompt. This ensures that you won't be surprised later, e.g. by trying to input a passphrase using a qwerty keyboard layout when you used an azerty layout during the installation. This situation can have several causes. Maybe you switched to another keyboard layout during the installation, or the selected keyboard layout might not have been set up yet when entering the passphrase for the root file system.

If you selected to use methods other than a passphrase to create encryption keys, they will be generated now. Because the kernel may not have gathered a sufficient amount of entropy at this early stage of the installation, the process may take a long time. You can help speed up the process by generating entropy: e.g. by pressing random keys, or by switching to the shell on the second virtual console and generating some network and disk traffic (downloading some files, feeding big files into `/dev/null`, etc.). This will be repeated for each partition to be encrypted.

After returning to the main partitioning menu, you will see all encrypted volumes as additional partitions which can be configured in the same way as ordinary partitions. The following example shows a volume encrypted via dm-crypt.

```
Encrypted volume (sda2_crypt) - 115.1 GB Linux device-mapper
     #1 115.1 GB  F ext3
```

[6] Folosirea unei parole-frază pe post de cheie în prezent înseamnă că partiția va fi configurată folosind LUKS.

[7] It is believed that the guys from three-letter agencies can restore the data even after several rewrites of the magnetooptical media, though.

Now is the time to assign mount points to the volumes and optionally change the file system types if the defaults do not suit you.

Pay attention to the identifiers in parentheses (`sda2_crypt` in this case) and the mount points you assigned to each encrypted volume. You will need this information later when booting the new system. The differences between the ordinary boot process and the boot process with encryption involved will be covered later in Secțiune 7.2.

Once you are satisfied with the partitioning scheme, continue with the installation.

6.3.5 Installing the Base System

Although this stage is the least problematic, it consumes a significant fraction of the install because it downloads, verifies and unpacks the whole base system. If you have a slow computer or network connection, this could take some time.

During installation of the base system, package unpacking and setup messages are redirected to **`tty4`**. You can access this terminal by pressing Left Alt-F4; get back to the main installer process with Left Alt-F1.

The unpack/setup messages generated during this phase are also saved in `/var/log/syslog`. You can check them there if the installation is performed over a serial console.

As part of the installation, a Linux kernel will be installed. At the default priority, the installer will choose one for you that best matches your hardware. In lower priority modes, you will be able to choose from a list of available kernels.

When packages are installed using the package management system, it will by default also install packages that are recommended by those packages. Recommended packages are not strictly required for the core functionality of the selected software, but they do enhance that software and should, in the view of the package maintainers, normally be installed together with that software.

Notă

For technical reasons packages installed during the installation of the base system are installed without their „Recommends”. The rule described above only takes effect after this point in the installation process.

6.3.6 Installing Additional Software

At this point you have a usable but limited system. Most users will want to install additional software on the system to tune it to their needs, and the installer allows you do so. This step can take even longer than installing the base system if you have a slow computer or network connection.

6.3.6.1 Configuring apt

One of the tools used to install packages on a Debian GNU/Linux system is the program **apt**, from the `apt` package[8]. Other front-ends for package management, like **aptitude** and **synaptic**, are also in use. These front-ends are recommended for new users, since they integrate some additional features (package searching and status checks) in a nice user interface.

apt must be configured so that it knows from where to retrieve packages. The results of this configuration are written to the file `/etc/apt/sources.list`. You can examine and edit this file to your liking after the installation is complete.

If you are installing at default priority, the installer will largely take care of the configuration automatically, based on the installation method you are using and possibly using choices made earlier in the installation. In most cases the installer will automatically add a security mirror and, if you are installing the stable distribution, a mirror for the „stable-updates” service.

If you are installing at a lower priority (e.g. in expert mode), you will be able to make more decisions yourself. You can choose whether or not to use the security and/or stable-updates services, and you can choose to add packages from the „contrib” and „non-free” sections of the archive.

[8] Note that the program which actually installs the packages is called **dpkg**. However, this program is more of a low-level tool. **apt** is a higher-level tool, which will invoke **dpkg** as appropriate. It knows how to retrieve packages from your installation media, the network, or wherever. It is also able to automatically install other packages which are required to make the package you're trying to install work correctly.

6.3.6.1.1 Installing from more than one CD or DVD image If you are installing from a CD or DVD image that is part of a larger set, the installer will ask if you want to scan additional installation media. If you have such additional media available, you probably want to do this so the installer can use the packages included on them.

If you do not have any additional media, that is no problem: using them is not required. If you also do not use a network mirror (as explained in the next section), it can mean that not all packages belonging to the tasks you select in the next step of the installation can be installed.

Notă

Packages are included on CD and DVD images in the order of their popularity. This means that for most uses only the first image of a set is needed and that only very few people actually use any of the packages included on the last images of a set.

It also means that buying or downloading and burning a full CD set is just a waste of money as you'll never use most of them. In most cases you are better off getting only the first 3 to 8 CDs and installing any additional packages you may need from the Internet by using a mirror. The same goes for DVD sets: the first DVD, or maybe the first two DVDs will cover most needs.

If you do scan multiple installation media, the installer will prompt you to exchange them when it needs packages from one that isn't currently in the drive. Note that only discs that belong to the same set should be scanned. The order in which they are scanned does not really matter, but scanning them in ascending order will reduce the chance of mistakes.

6.3.6.1.2 Using a network mirror One question that will be asked during most installs is whether or not to use a network mirror as a source for packages. In most cases the default answer should be fine, but there are some exceptions.

If you are *not* installing from a full CD/DVD image, you really should use a network mirror as otherwise you will end up with only a very minimal system. However, if you have a limited Internet connection it is best *not* to select the `desktop` task in the next step of the installation.

If you are installing from a single full CD image, using a network mirror is not required, but is still strongly recommended because a single CD image contains only a fairly limited number of packages. If you have a limited Internet connection it may still be best to *not* select a network mirror here, but to finish the installation using only what's available on the CD image and selectively install additional packages after the installation (i.e. after you have rebooted into the new system).

If you are installing from DVD, any packages needed during the installation should be present on the first DVD image. Use of a network mirror is optional.

One advantage of adding a network mirror is that updates, that have occurred since the CD/DVD images were created and have been included in a point release, will become available for installation, thus extending the life of your CD/DVD set without compromising the security or stability of the installed system.

In summary: selecting a network mirror is generally a good idea, except if you do not have a good Internet connection. If the current version of a package is available from installation media, the installer will always use that. The amount of data that will be downloaded if you do select a mirror thus depends on

1. the tasks you select in the next step of the installation,
2. which packages are needed for those tasks,
3. which of those packages are present on the installation media you have scanned, and
4. whether any updated versions of packages included on the installation media are available from a mirror (either a regular package mirror, or a mirror for security or stable-updates).

Note that the last point means that, even if you choose not to use a network mirror, some packages may still be downloaded from the Internet if there is a security or stable-updates update available for them and those services have been configured.

6.3.6.1.3 Choosing a network mirror Unless you chose not to use a network mirror, you will be presented with a list of network mirrors based upon your country selection earlier in the installation process. Choosing the offered default is usually fine.

The offered default is deb.debian.org, which is not a mirror itself but will redirect to a mirror that should be up-to-date and fast. These mirrors support TLS (https protocol) and IPv6. This service is maintained by the Debian System Administration (DSA) team.

A mirror can also be specified by hand by choosing „enter information manually" . You can then specify a mirror host name and an optional port number. This actually has to be a URL base, i.e. when specifying an IPv6 address, one has to add square brackets around it, for instance „[2001:db8::1]".

If your computer is on an IPv6-only network (which is probably not the case for the vast majority of users), using the default mirror for your country might not work. All the mirrors in the list are reachable via IPv4, but only some of them can be used via IPv6. As connectivity of individual mirrors can change over time, this information is not available in the installer. If there is no IPv6 connectivity for the default mirror for your country, you can either try some of the other mirrors offered to you or choose the „enter information manually" option. You can then specify „ftp.ipv6.debian.org" as the mirror name, which is an alias for a mirror available via IPv6, although it will probably not be the fastest possible one.

6.3.6.2 Selecting and Installing Software

During the installation process, you are given the opportunity to select additional software to install. Rather than picking individual software packages from the 85212 available packages, this stage of the installation process focuses on selecting and installing predefined collections of software to quickly set up your computer to perform various tasks.

These tasks loosely represent a number of different jobs or things you want to do with your computer, such as „Desktop environment", „Web server", or „Print server"[9]. Secțiune D.2 lists the space requirements for the available tasks.

Some tasks may be pre-selected based on the characteristics of the computer you are installing. If you disagree with these selections you can deselect them. You can even opt to install no tasks at all at this point.

INDICAȚIE

In the standard user interface of the installer, you can use the space bar to toggle selection of a task.

NOTĂ

The „Desktop environment" task will install a graphical desktop environment.

By default, `debian-installer` installs the Gnome desktop environment. It is possible to interactively select a different desktop environment during the installation. It is also possible to install multiple desktops, but some combinations of desktop may not be co-installable.

Note that this will only work if the packages needed for the desired desktop environment are actually available. If you are installing using a single full CD image, they will possibly need to be downloaded from a network mirror as they might not be available on the CD image due to its limited amount of space. Installing any of the available desktop environments this way should work fine if you are using a DVD image or any other installation method.

The various server tasks will install software roughly as follows. Web server: `apache2`; Print server: `cups`; SSH server: `openssh`.

The „Standard system" task will install any package that has a priority „standard". This includes a lot of common utilities that are normally available on any Linux or Unix system. You should leave this task selected unless you know what you are doing and want a really minimal system.

If during language selection a default locale other than the „C" locale was selected, **tasksel** will check if any localization tasks are defined for that locale and will automatically try to install relevant localization packages. This

[9] You should know that to present this list, the installer is merely invoking the **tasksel** program. It can be run at any time after installation to install more packages (or remove them), or you can use a more fine-grained tool such as **aptitude**. If you are looking for a specific single package, after installation is complete, simply run **aptitude install *package***, where *package* is the name of the package you are looking for.

includes for example packages containing word lists or special fonts for your language. If a desktop environment was selected, it will also install appropriate localization packages for that (if available).

Once you've selected your tasks, select Continue. At this point, **apt** will install the packages that are part of the selected tasks. If a particular program needs more information from the user, it will prompt you during this process.

You should be aware that especially the Desktop task is very large. Especially when installing from a normal CD-ROM in combination with a mirror for packages not on the CD-ROM, the installer may want to retrieve a lot of packages over the network. If you have a relatively slow Internet connection, this can take a long time. There is no option to cancel the installation of packages once it has started.

Even when packages are included on the CD-ROM, the installer may still retrieve them from the mirror if the version available on the mirror is more recent than the one included on the CD-ROM. If you are installing the stable distribution, this can happen after a point release (an update of the original stable release); if you are installing the testing distribution this will happen if you are using an older image.

6.3.7 Making Your System Bootable

If you are installing a diskless workstation, obviously, booting off the local disk isn't a meaningful option, and this step will be skipped.

6.3.7.1 Detecting other operating systems

Before a boot loader is installed, the installer will attempt to probe for other operating systems which are installed on the machine. If it finds a supported operating system, you will be informed of this during the boot loader installation step, and the computer will be configured to boot this other operating system in addition to Debian.

Note that multiple operating systems booting on a single machine is still something of a black art. The automatic support for detecting and setting up boot loaders to boot other operating systems varies by architecture and even by subarchitecture. If it does not work you should consult your boot manager's documentation for more information.

6.3.7.2 Install the Grub Boot Loader on a Hard Disk

The main amd64 boot loader is called „grub". Grub is a flexible and robust boot loader and a good default choice for new users and old hands alike.

By default, grub will be installed into the Master Boot Record (MBR), where it will take over complete control of the boot process. If you prefer, you can install it elsewhere. See the grub manual for complete information.

If you do not want to install grub, use the Go Back button to get to the main menu, and from there select whatever bootloader you would like to use.

6.3.7.3 Install the LILO Boot Loader on a Hard Disk

The second amd64 boot loader is called „LILO". It is an old complex program which offers lots of functionality, including DOS, Windows, and OS/2 boot management. Please carefully read the instructions in the directory `/usr/share/doc/lilo/` if you have special needs; also see the LILO mini-HOWTO.

> Notă
>
> 
>
> Currently the LILO installation will only create menu entries for other operating systems if these can be *chainloaded*. This means you may have to manually add a menu entry for operating systems like GNU/Linux and GNU/Hurd after the installation.

`debian-installer` offers you three choices on where to install the **LILO** boot loader:

Master Boot Record (MBR) This way the **LILO** will take complete control of the boot process.

new Debian partition Choose this if you want to use another boot manager. **LILO** will install itself at the beginning of the new Debian partition and it will serve as a secondary boot loader.

Other choice Useful for advanced users who want to install **LILO** somewhere else. In this case you will be asked for desired location. You can use traditional device names such as `/dev/sda`.

If you can no longer boot into Windows 9x (or DOS) after this step, you'll need to use a Windows 9x (MS-DOS) boot disk and use the **fdisk /mbr** command to reinstall the MS-DOS master boot record — however, this means that you'll need to use some other way to get back into Debian!

6.3.7.4 Continue Without Boot Loader

This option can be used to complete the installation even when no boot loader is to be installed, either because the arch/subarch doesn't provide one, or because none is desired (e.g. you will use existing boot loader).

If you plan to manually configure your bootloader, you should check the name of the installed kernel in `/target/boot`. You should also check that directory for the presence of an *initrd*; if one is present, you will probably have to instruct your bootloader to use it. Other information you will need are the disk and partition you selected for your `/` filesystem and, if you chose to install `/boot` on a separate partition, also your `/boot` filesystem.

6.3.8 Finishing the Installation

This is the last step in the Debian installation process during which the installer will do any last minute tasks. It mostly consists of tidying up after the `debian-installer`.

6.3.8.1 Setting the System Clock

The installer may ask you if the computer's clock is set to UTC. Normally this question is avoided if possible and the installer tries to work out whether the clock is set to UTC based on things like what other operating systems are installed.

In expert mode you will always be able to choose whether or not the clock is set to UTC. Systems that (also) run Dos or Windows are normally set to local time. If you want to dual-boot, select local time instead of UTC.

At this point `debian-installer` will also attempt to save the current time to the system's hardware clock. This will be done either in UTC or local time, depending on the selection that was just made.

6.3.8.2 Reboot the System

You will be prompted to remove the boot media (CD, USB stick, etc) that you used to boot the installer. After that the system will be rebooted into your new Debian system.

6.3.9 Troubleshooting

The components listed in this section are usually not involved in the installation process, but are waiting in the background to help the user in case something goes wrong.

6.3.9.1 Saving the installation logs

If the installation is successful, the logfiles created during the installation process will be automatically saved to `/var/log/installer/` on your new Debian system.

Choosing Save debug logs from the main menu allows you to save the log files to a USB stick, network, hard disk, or other media. This can be useful if you encounter fatal problems during the installation and wish to study the logs on another system or attach them to an installation report.

6.3.9.2 Using the Shell and Viewing the Logs

There are several methods you can use to get a shell while running an installation. On most systems, and if you are not installing over serial console, the easiest method is to switch to the second *virtual console* by pressing Left Alt-F2[10] (on a Mac keyboard, Option-F2). Use Left Alt-F1 to switch back to the installer itself.

For the graphical installer see also Secțiune 6.1.1.

If you cannot switch consoles, there is also an Execute a Shell item on the main menu that can be used to start a shell. You can get to the main menu from most dialogs by using the Go Back button one or more times. Type **exit** to close the shell and return to the installer.

At this point you are booted from the RAM disk, and there is a limited set of Unix utilities available for your use. You can see what programs are available with the command **ls /bin /sbin /usr/bin /usr/sbin** and by typing **help**. The shell is a Bourne shell clone called **ash** and has some nice features like autocompletion and history.

[10] That is: press the **Alt** key on the left-hand side of the **space bar** and the **F2** function key at the same time.

To edit and view files, use the text editor **nano**. Log files for the installation system can be found in the `/var/log` directory.

Notă

Although you can do basically anything in a shell that the available commands allow you to do, the option to use a shell is really only there in case something goes wrong and for debugging.

Doing things manually from the shell may interfere with the installation process and result in errors or an incomplete installation. In particular, you should always use let the installer activate your swap partition and not do this yourself from a shell.

6.3.10 Installation Over the Network

One of the more interesting components is *network-console*. It allows you to do a large part of the installation over the network via SSH. The use of the network implies you will have to perform the first steps of the installation from the console, at least to the point of setting up the networking. (Although you can automate that part with Secțiune 4.6.)

This component is not loaded into the main installation menu by default, so you have to explicitly ask for it. If you are installing from optical media, you need to boot with medium priority or otherwise invoke the main installation menu and choose Load installer components from installation media and from the list of additional components select network-console: Continue installation remotely using SSH. Successful load is indicated by a new menu entry called Continue installation remotely using SSH.

After selecting this new entry, you will be asked for a new password to be used for connecting to the installation system and for its confirmation. That's all. Now you should see a screen which instructs you to login remotely as the user *installer* with the password you just provided. Another important detail to notice on this screen is the fingerprint of this system. You need to transfer the fingerprint securely to the person who will continue the installation remotely.

Should you decide to continue with the installation locally, you can always press **Enter**, which will bring you back to the main menu, where you can select another component.

Now let's switch to the other side of the wire. As a prerequisite, you need to configure your terminal for UTF-8 encoding, because that is what the installation system uses. If you do not, remote installation will be still possible, but you may encounter strange display artefacts like destroyed dialog borders or unreadable non-ascii characters. Establishing a connection with the installation system is as simple as typing:

```
$ ssh -l installer install_host
```

Where *install_host* is either the name or IP address of the computer being installed. Before the actual login the fingerprint of the remote system will be displayed and you will have to confirm that it is correct.

Notă

The **ssh** server in the installer uses a default configuration that does not send keep-alive packets. In principle, a connection to the system being installed should be kept open indefinitely. However, in some situations — depending on your local network setup — the connection may be lost after some period of inactivity. One common case where this can happen is when there is some form of Network Address Translation (NAT) somewhere between the client and the system being installed. Depending on at which point of the installation the connection was lost, you may or may not be able to resume the installation after reconnecting.

You may be able to avoid the connection being dropped by adding the option **-o ServerAliveInterval=*value*** when starting the **ssh** connection, or by adding that option in your **ssh** configuration file. Note however that in some cases adding this option may also *cause* a connection to be dropped (for example if keep-alive packets are sent during a brief network outage, from which **ssh** would otherwise have recovered), so it should only be used when needed.

Notă

If you install several computers in turn and they happen to have the same IP address or hostname, **ssh** will refuse to connect to such host. The reason is that it will have different fingerprint, which is usually a sign of a spoofing attack. If you are sure this is not the case, you will need to delete the relevant line from `~/.ssh/known_hosts`[a] and try again.

[a] The following command will remove an existing entry for a host: **ssh-keygen -R <*hostname*|*IP address*>**.

After the login you will be presented with an initial screen where you have two possibilities called Start menu and Start shell. The former brings you to the main installer menu, where you can continue with the installation as usual. The latter starts a shell from which you can examine and possibly fix the remote system. You should only start one SSH session for the installation menu, but may start multiple sessions for shells.

AVERTISMENT

After you have started the installation remotely over SSH, you should not go back to the installation session running on the local console. Doing so may corrupt the database that holds the configuration of the new system. This in turn may result in a failed installation or problems with the installed system.

6.4 Loading Missing Firmware

As described in Secțiune 2.2, some devices require firmware to be loaded. In most cases the device will not work at all if the firmware is not available; sometimes basic functionality is not impaired if it is missing and the firmware is only needed to enable additional features.

If a device driver requests firmware that is not available, `debian-installer` will display a dialog offering to load the missing firmware. If this option is selected, `debian-installer` will scan available devices for either loose firmware files or packages containing firmware. If found, the firmware will be copied to the correct location (`/lib/firmware`) and the driver module will be reloaded.

Notă

Which devices are scanned and which file systems are supported depends on the architecture, the installation method and the stage of the installation. Especially during the early stages of the installation, loading the firmware is most likely to succeed from a FAT-formatted USB stick. On i386 and amd64 firmware can also be loaded from an MMC or SD card.

Note that it is possible to skip loading the firmware if you know the device will also function without it, or if the device is not needed during the installation.

`debian-installer` only prompts for firmware needed by kernel modules loaded during the installation. Not all drivers are included in `debian-installer`, in particular radeon is not, so this implies that the capabilities of some devices may be no different at the end of the installation from what they were at the beginning. Consequently, some of your hardware may not be being used to its full potential. If you suspect this is the case, or are just curious, it is not a bad idea to check the output of the **dmesg** command on the newly booted system and search for „firmware”.

6.4.1 Preparing a medium

Official installation images do not include non-free firmware. The most common method to load such firmware is from some removable medium such as a USB stick. Alternatively, unofficial installation images containing non-

free firmware can be found at https://cdimage.debian.org/cdimage/unofficial/non-free/cd-including-firmware/. To prepare a USB stick (or other medium like a hard drive partition), the firmware files or packages must be placed in either the root directory or a directory named /firmware of the file system on the medium. The recommended file system to use is FAT as that is most certain to be supported during the early stages of the installation.

Tarballs and zip files containing current packages for the most common firmware are available from:

- https://cdimage.debian.org/cdimage/unofficial/non-free/firmware/

Just download the tarball or zip file for the correct release and unpack it to the file system on the medium.

If the firmware you need is not included in the tarball, you can also download specific firmware packages from the (non-free section of the) archive. The following overview should list most available firmware packages but is not guaranteed to be complete and may also contain non-firmware packages:

- https://packages.debian.org/search?keywords=firmware

It is also possible to copy individual firmware files to the medium. Loose firmware could be obtained for example from an already installed system or from a hardware vendor.

6.4.2 Firmware and the Installed System

Any firmware loaded during the installation will be copied automatically to the installed system. In most cases this will ensure that the device that requires the firmware will also work correctly after the system is rebooted into the installed system. However, if the installed system runs a different kernel version from the installer there is a slight chance that the firmware cannot be loaded due to version skew.

If the firmware was loaded from a firmware package, debian-installer will also install this package for the installed system and will automatically add the non-free section of the package archive in APT's sources.list. This has the advantage that the firmware should be updated automatically if a new version becomes available.

If loading the firmware was skipped during the installation, the relevant device will probably not work with the installed system until the firmware (package) is installed manually.

Notă

If the firmware was loaded from loose firmware files, the firmware copied to the installed system will *not* be automatically updated unless the corresponding firmware package (if available) is installed after the installation is completed.

Capitolul 7

Booting Into Your New Debian System

7.1 The Moment of Truth

Your system's first boot on its own power is what electrical engineers call the „smoke test".

If you did a default installation, the first thing you should see when you boot the system is the menu of the `grub` or possibly the `lilo` bootloader. The first choices in the menu will be for your new Debian system. If you had any other operating systems on your computer (like Windows) that were detected by the installation system, those will be listed lower down in the menu.

If the system fails to start up correctly, don't panic. If the installation was successful, chances are good that there is only a relatively minor problem that is preventing the system from booting Debian. In most cases such problems can be fixed without having to repeat the installation. One available option to fix boot problems is to use the installer's built-in rescue mode (see Secțiune 8.6).

If you are new to Debian and Linux, you may need some help from more experienced users. For direct on-line help you can try the IRC channels #debian or #debian-boot on the OFTC network. Alternatively you can contact the debian-user mailing list. You can also file an installation report as described in Secțiune 5.4.7. Please make sure that you describe your problem clearly and include any messages that are displayed and may help others to diagnose the issue.

If you had any other operating systems on your computer that were not detected or not detected correctly, please file an installation report.

7.2 Mounting encrypted volumes

If you created encrypted volumes during the installation and assigned them mount points, you will be asked to enter the passphrase for each of these volumes during the boot.

For partitions encrypted using dm-crypt you will be shown the following prompt during the boot:

```
Starting early crypto disks... part_crypt(starting)
Enter LUKS passphrase:
```

In the first line of the prompt, *part* is the name of the underlying partition, e.g. sda2 or md0. You are now probably wondering *for which volume* you are actually entering the passphrase. Does it relate to your `/home`? Or to `/var`? Of course, if you have just one encrypted volume, this is easy and you can just enter the passphrase you used when setting up this volume. If you set up more than one encrypted volume during the installation, the notes you wrote down as the last step in Secțiune 6.3.4.6 come in handy. If you did not make a note of the mapping between *part*`_crypt` and the mount points before, you can still find it in `/etc/crypttab` and `/etc/fstab` of your new system.

The prompt may look somewhat different when an encrypted root file system is mounted. This depends on which initramfs generator was used to generate the initrd used to boot the system. The example below is for an initrd generated using `initramfs-tools`:

```
Begin: Mounting root file system... ...
Begin: Running /scripts/local-top ...
Enter LUKS passphrase:
```

No characters (even asterisks) will be shown while entering the passphrase. If you enter the wrong passphrase, you have two more tries to correct it. After the third try the boot process will skip this volume and continue to mount the next filesystem. Please see Secțiune 7.2.1 for further information.

After entering all passphrases the boot should continue as usual.

7.2.1 Troubleshooting

If some of the encrypted volumes could not be mounted because a wrong passphrase was entered, you will have to mount them manually after the boot. There are several cases.

- The first case concerns the root partition. When it is not mounted correctly, the boot process will halt and you will have to reboot the computer to try again.
- The easiest case is for encrypted volumes holding data like `/home` or `/srv`. You can simply mount them manually after the boot.

  However for dm-crypt this is a bit tricky. First you need to register the volumes with device mapper by running:

  ```
  # /etc/init.d/cryptdisks start
  ```

  This will scan all volumes mentioned in `/etc/crypttab` and will create appropriate devices under the `/dev` directory after entering the correct passphrases. (Already registered volumes will be skipped, so you can repeat this command several times without worrying.) After successful registration you can simply mount the volumes the usual way:

  ```
  # mount /mount_point
  ```

- If any volume holding noncritical system files could not be mounted (`/usr` or `/var`), the system should still boot and you should be able to mount the volumes manually like in the previous case. However, you will also need to (re)start any services usually running in your default runlevel because it is very likely that they were not started. The easiest way is to just reboot the computer.

7.3 Log In

Once your system boots, you'll be presented with the login prompt. Log in using the personal login and password you selected during the installation process. Your system is now ready for use.

If you are a new user, you may want to explore the documentation which is already installed on your system as you start to use it. There are currently several documentation systems, work is proceeding on integrating the different types of documentation. Here are a few starting points.

Documentation accompanying programs you have installed can be found in `/usr/share/doc/`, under a subdirectory named after the program (or, more precise, the Debian package that contains the program). However, more extensive documentation is often packaged separately in special documentation packages that are mostly not installed by default. For example, documentation about the package management tool **apt** can be found in the packages `apt-doc` or `apt-howto`.

In addition, there are some special folders within the `/usr/share/doc/` hierarchy. Linux HOWTOs are installed in *.gz* (compressed) format, in `/usr/share/doc/HOWTO/en-txt/`. After installing `dhelp`, you will find a browsable index of documentation in `/usr/share/doc/HTML/index.html`.

One easy way to view these documents using a text based browser is to enter the following commands:

```
$ cd /usr/share/doc/
$ w3m .
```

The dot after the **w3m** command tells it to show the contents of the current directory.

If you have a graphical desktop environment installed, you can also use its web browser. Start the web browser from the application menu and enter **/usr/share/doc/** in the address bar.

You can also type **info *command*** or **man *command*** to see documentation on most commands available at the command prompt. Typing **help** will display help on shell commands. And typing a command followed by **--help** will usually display a short summary of the command's usage. If a command's results scroll past the top of the screen, type **| more** after the command to cause the results to pause before scrolling past the top of the screen. To see a list of all commands available which begin with a certain letter, type the letter and then two tabs.

Capitolul 8

Next Steps and Where to Go From Here

8.1 Shutting down the system

To shut down a running Debian GNU/Linux system, you must not reboot with the reset switch on the front or back of your computer, or just turn off the computer. Debian GNU/Linux should be shut down in a controlled manner, otherwise files might get lost and/or disk damage might occur. If you run a desktop environment, there is usually an option to „log out" available from the application menu that allows you to shutdown (or reboot) the system.

Alternatively you can press the key combination Ctrl-Alt-Del . If the key combinations do not work, a last option is to log in as root and type the necessary commands. Use **reboot** to reboot the system. Use **halt** to halt the system without powering it off [1]. To power off the machine, use **poweroff** or **shutdown -h now**. The systemd init system provides additional commands that perform the same functions; for example **systemctl reboot** or **systemctl poweroff**.

8.2 Orienting Yourself to Debian

Debian is a little different from other distributions. Even if you're familiar with Linux in other distributions, there are things you should know about Debian to help you to keep your system in a good, clean state. This chapter contains material to help you get oriented; it is not intended to be a tutorial for how to use Debian, but just a very brief glimpse of the system for the very rushed.

8.2.1 Debian Packaging System

The most important concept to grasp is the Debian packaging system. In essence, large parts of your system should be considered under the control of the packaging system. These include:

- `/usr` (excluding `/usr/local`)
- `/var` (you could make `/var/local` and be safe in there)
- `/bin`
- `/sbin`
- `/lib`

For instance, if you replace `/usr/bin/perl`, that will work, but then if you upgrade your `perl` package, the file you put there will be replaced. Experts can get around this by putting packages on „hold" in **aptitude**.

One of the best installation methods is apt. You can use the command line version of **apt** as well as tools like aptitude or synaptic (which are just graphical frontends for **apt**). Note that apt will also let you merge main, contrib, and non-free so you can have restricted packages (strictly speaking not belonging to Debian) as well as packages from Debian GNU/Linux at the same time.

[1] Under the SysV init system **halt** had the same effect as **poweroff**, but with systemd as init system (the default since jessie) their effects are different.

8.2.2 Additional Software Available for Debian

There are official and unofficial software repositories that are not enabled in the default Debian install. These contain software which many find important and expect to have. Information on these additional repositories can be found on the Debian Wiki page titled The Software Available for Debian's Stable Release.

8.2.3 Application Version Management

Alternative versions of applications are managed by update-alternatives. If you are maintaining multiple versions of your applications, read the update-alternatives man page.

8.2.4 Cron Job Management

Any jobs under the purview of the system administrator should be in `/etc`, since they are configuration files. If you have a root cron job for daily, weekly, or monthly runs, put them in `/etc/cron.{daily,weekly,monthly}`. These are invoked from `/etc/crontab`, and will run in alphabetic order, which serializes them.

On the other hand, if you have a cron job that (a) needs to run as a special user, or (b) needs to run at a special time or frequency, you can use either `/etc/crontab`, or, better yet, `/etc/cron.d/whatever`. These particular files also have an extra field that allows you to stipulate the user account under which the cron job runs.

In either case, you just edit the files and cron will notice them automatically. There is no need to run a special command. For more information see cron(8), crontab(5), and `/usr/share/doc/cron/README.Debian`.

8.3 Further Reading and Information

The Debian web site contains a large quantity of documentation about Debian. In particular, see the Debian GNU/-Linux FAQ and the Debian Reference. An index of more Debian documentation is available from the Debian Documentation Project. The Debian community is self-supporting; to subscribe to one or more of the Debian mailing lists, see the Mail List Subscription page. Last, but not least, the Debian Mailing List Archives contain a wealth of information on Debian.

If you need information about a particular program, you should first try **`man` *`program`***, or **`info` *`program`***.

There is lots of useful documentation in `/usr/share/doc` as well. In particular, `/usr/share/doc/HOWTO` and `/usr/share/doc/FAQ` contain lots of interesting information. To submit bugs, look at `/usr/share/doc/debian/bug*`. To read about Debian-specific issues for particular programs, look at `/usr/share/doc/(package name)/README.Debian`.

A general source of information on GNU/Linux is the Linux Documentation Project. There you will find the HOWTOs and pointers to other very valuable information on parts of a GNU/Linux system.

Linux is an implementation of Unix. The Linux Documentation Project (LDP) collects a number of HOWTOs and online books relating to Linux.

If you are new to Unix, you probably should go out and buy some books and do some reading. This list of Unix FAQs contains a number of UseNet documents which provide a nice historical reference.

8.4 Setting Up Your System To Use E-Mail

Today, email is an important part of many people's life. As there are many options as to how to set it up, and as having it set up correctly is important for some Debian utilities, we will try to cover the basics in this section.

There are three main functions that make up an e-mail system. First there is the *Mail User Agent* (MUA) which is the program a user actually uses to compose and read mails. Then there is the *Mail Transfer Agent* (MTA) that takes care of transferring messages from one computer to another. And last there is the *Mail Delivery Agent* (MDA) that takes care of delivering incoming mail to the user's inbox.

These three functions can be performed by separate programs, but they can also be combined in one or two programs. It is also possible to have different programs handle these functions for different types of mail.

On Linux and Unix systems **mutt** is historically a very popular MUA. Like most traditional Linux programs it is text based. It is often used in combination with **exim** or **sendmail** as MTA and **procmail** as MDA.

With the increasing popularity of graphical desktop systems, the use of graphical e-mail programs like GNOME's **evolution**, KDE's **kmail** or Mozilla's **thunderbird** has becoming more popular. These programs combine the function of a MUA, MTA and MDA, but can — and often are — also be used in combination with the traditional Linux tools.

8.4.1 Default E-Mail Configuration

Even if you are planning to use a graphical mail program, it would be useful, to have a traditional MTA/MDA installed and correctly set up on your Debian GNU/Linux system. Reason is that various utilities running on the system[2] can send important notices by e-mail to inform the system administrator of (potential) problems or changes.

For this you can install `exim4` and `mutt` with **apt install exim4 mutt**. `exim4` is a combination MTA/MDA that is relatively small but very flexible. By default it will be configured to only handle e-mail local to the system itself and e-mails addressed to the system administrator (root account) will be delivered to the regular user account created during the installation[3].

When system e-mails are delivered they are added to a file in `/var/mail/account_name`. The e-mails can be read using **mutt**.

8.4.2 Sending E-Mails Outside The System

As mentioned earlier, the installed Debian system is only set up to handle e-mail local to the system, not for sending mail to others nor for receiving mail from others.

If you would like `exim4` to handle external e-mail, please refer to the next subsection for the basic available configuration options. Make sure to test that mail can be sent and received correctly.

If you intend to use a graphical mail program and use a mail server of your Internet Service Provider (ISP) or your company, there is not really any need to configure `exim4` for handling external e-mail. Just configure your favorite graphical mail program to use the correct servers to send and receive e-mail (how is outside the scope of this manual).

However, in that case you may need to configure individual utilities to correctly send e-mails. One such utility is **reportbug**, a program that facilitates submitting bug reports against Debian packages. By default it expects to be able to use `exim4` to submit bug reports.

To correctly set up **reportbug** to use an external mail server, please run the command **reportbug --configure** and answer „no” to the question if an MTA is available. You will then be asked for the SMTP server to be used for submitting bug reports.

8.4.3 Configuring the Exim4 Mail Transport Agent

If you would like your system to also handle external e-mail, you will need to reconfigure the `exim4` package[4]:

```
# dpkg-reconfigure exim4-config
```

After entering that command (as root), you will be asked if you want split the configuration into small files. If you are unsure, select the default option.

Next you will be presented with several common mail scenarios. Choose the one that most closely resembles your needs.

internet site Your system is connected to a network and your mail is sent and received directly using SMTP. On the following screens you will be asked a few basic questions, like your machine's mail name, or a list of domains for which you accept or relay mail.

mail sent by smarthost In this scenario your outgoing mail is forwarded to another machine, called a „smarthost”, which takes care of sending the message on to its destination. The smarthost also usually stores incoming mail addressed to your computer, so you don't need to be permanently online. That also means you have to download your mail from the smarthost via programs like fetchmail.

In a lot of cases the smarthost will be your ISP's mail server, which makes this option very suitable for dial-up users. It can also be a company mail server, or even another system on your own network.

mail sent by smarthost; no local mail This option is basically the same as the previous one except that the system will not be set up to handle mail for a local e-mail domain. Mail on the system itself (e.g. for the system administrator) will still be handled.

local delivery only This is the option your system is configured for by default.

[2] Examples are: **cron**, **quota**, **logcheck**, **aide**, …

[3] The forwarding of mail for root to the regular user account is configured in `/etc/aliases`. If no regular user account was created, the mail will of course be delivered to the root account itself.

[4] You can of course also remove `exim4` and replace it with an alternative MTA/MDA.

no configuration at this time Choose this if you are absolutely convinced you know what you are doing. This will leave you with an unconfigured mail system — until you configure it, you won't be able to send or receive any mail and you may miss some important messages from your system utilities.

If none of these scenarios suits your needs, or if you need a finer grained setup, you will need to edit configuration files under the `/etc/exim4` directory after the installation is complete. More information about `exim4` may be found under `/usr/share/doc/exim4`; the file `README.Debian.gz` has further details about configuring `exim4` and explains where to find additional documentation.

Note that sending mail directly to the Internet when you don't have an official domain name, can result in your mail being rejected because of anti-spam measures on receiving servers. Using your ISP's mail server is preferred. If you still do want to send out mail directly, you may want to use a different e-mail address than is generated by default. If you use `exim4` as your MTA, this is possible by adding an entry in `/etc/email-addresses`.

8.5 Compiling a New Kernel

Why would someone want to compile a new kernel? It is most probably not necessary since the default kernel shipped with Debian handles almost all configurations.

If you want to compile your own kernel nevertheless, this is of course possible and we recommend the use of the „make deb-pkg" target. For more information read the Debian Linux Kernel Handbook.

8.6 Recovering a Broken System

Sometimes, things go wrong, and the system you've carefully installed is no longer bootable. Perhaps the boot loader configuration broke while trying out a change, or perhaps a new kernel you installed won't boot, or perhaps cosmic rays hit your disk and flipped a bit in `/sbin/init`. Regardless of the cause, you'll need to have a system to work from while you fix it, and rescue mode can be useful for this.

To access rescue mode, select **`rescue`** from the boot menu, type **`rescue`** at the `boot:` prompt, or boot with the **`rescue/enable=true`** boot parameter. You'll be shown the first few screens of the installer, with a note in the corner of the display to indicate that this is rescue mode, not a full installation. Don't worry, your system is not about to be overwritten! Rescue mode simply takes advantage of the hardware detection facilities available in the installer to ensure that your disks, network devices, and so on are available to you while repairing your system.

Instead of the partitioning tool, you should now be presented with a list of the partitions on your system, and asked to select one of them. Normally, you should select the partition containing the root file system that you need to repair. You may select partitions on RAID and LVM devices as well as those created directly on disks.

If possible, the installer will now present you with a shell prompt in the file system you selected, which you can use to perform any necessary repairs. For example, if you need to reinstall the GRUB boot loader into the master boot record of the first hard disk, you could enter the command **`grub-install '(hd0)'`** to do so.

If the installer cannot run a usable shell in the root file system you selected, perhaps because the file system is corrupt, then it will issue a warning and offer to give you a shell in the installer environment instead. You may not have as many tools available in this environment, but they will often be enough to repair your system anyway. The root file system you selected will be mounted on the `/target` directory.

In either case, after you exit the shell, the system will reboot.

Finally, note that repairing broken systems can be difficult, and this manual does not attempt to go into all the things that might have gone wrong or how to fix them. If you have problems, consult an expert.

Anexa A

Installation Howto

This document describes how to install Debian GNU/Linux bullseye for the 64-bit PC („amd64”) with the new `debian-installer`. It is a quick walkthrough of the installation process which should contain all the information you will need for most installs. When more information can be useful, we will link to more detailed explanations in other parts of this document.

A.1 Preliminaries

The debian-installer is still in a beta state. If you encounter bugs during your install, please refer to Secțiune 5.4.7 for instructions on how to report them. If you have questions which cannot be answered by this document, please direct them to the debian-boot mailing list (debian-boot@lists.debian.org) or ask on IRC (#debian-boot on the OFTC network).

A.2 Booting the installer

For some quick links to installation images, check out the `debian-installer` home page. The debian-cd team provides builds of installation images using `debian-installer` on the Debian CD/DVD page. For more information on where to get installation images, see Secțiune 4.1.

Some installation methods require other images than those for optical media. The `debian-installer` home page has links to other images. Secțiune 4.2.1 explains how to find images on Debian mirrors.

The subsections below will give the details about which images you should get for each possible means of installation.

A.2.1 Optical disc

The netinst CD image is a popular image which can be used to install bullseye with the `debian-installer`. This installation method is intended to boot from the image and install additional packages over a network; hence the name „netinst”. The image has the software components needed to run the installer and the base packages to provide a minimal bullseye system. If you'd rather, you can get a full size CD/DVD image which will not need the network to install. You only need the first image of such set.

Download whichever type you prefer and burn it to an optical disc. To boot the disc, you may need to change your BIOS configuration, as explained in Secțiune 3.6.1.

A.2.2 USB memory stick

It's also possible to install from removable USB storage devices. For example a USB keychain can make a handy Debian install medium that you can take with you anywhere.

The easiest way to prepare your USB memory stick is to download any Debian CD or DVD image that will fit on it, and write the image directly to the memory stick. Of course this will destroy anything already on the stick. This works because Debian CD/DVD images are „isohybrid” images that can boot both from optical and USB drives.

There are other, more flexible ways to set up a memory stick to use the debian-installer, and it's possible to get it to work with smaller memory sticks. For details, see Secțiune 4.3.

Some BIOSes can boot USB storage directly, and some cannot. You may need to configure your BIOS to enable „USB legacy support”. The boot device selection menu should show „removable drive” or „USB-HDD” to get it to boot from the USB device. For helpful hints and details, see Secțiune 5.1.1.

A.2.3 Booting from network

It's also possible to boot `debian-installer` completely from the net. The various methods to netboot depend on your architecture and netboot setup. The files in `netboot/` can be used to netboot `debian-installer`.

The easiest thing to set up is probably PXE netbooting. Untar the file `netboot/pxeboot.tar.gz` into `/srv/tftp` or wherever is appropriate for your tftp server. Set up your DHCP server to pass filename `pxelinux.0` to clients, and with luck everything will just work. For detailed instructions, see Secțiune 4.5.

A.2.4 Booting from hard disk

It's possible to boot the installer using no removable media, but just an existing hard disk, which can have a different OS on it. Download `hd-media/initrd.gz`, `hd-media/vmlinuz`, and a Debian CD/DVD image to the top-level directory of the hard disk. Make sure that the image has a filename ending in `.iso`. Now it's just a matter of booting linux with the initrd. Secțiune 5.1.5 explains one way to do it.

A.3 Installation

Once the installer starts, you will be greeted with an initial screen. Press **Enter** to boot, or read the instructions for other boot methods and parameters (see Secțiune 5.3).

After a while you will be asked to select your language. Use the arrow keys to pick a language and press **Enter** to continue. Next you'll be asked to select your country, with the choices including countries where your language is spoken. If it's not on the short list, a list of all the countries in the world is available.

You may be asked to confirm your keyboard layout. Choose the default unless you know better.

Now sit back while debian-installer detects some of your hardware, and loads the rest of the installation image.

Next the installer will try to detect your network hardware and set up networking by DHCP. If you are not on a network or do not have DHCP, you will be given the opportunity to configure the network manually.

Setting up the network is followed by the creation of user accounts. By default you are asked to provide a password for the „root” (administrator) account and information necessary to create one regular user account. If you do not specify a password for the „root” user, this account will be disabled but the **sudo** package will be installed later to enable administrative tasks to be carried out on the new system. By default, the first user created on the system will be allowed to use the **sudo** command to become root.

The next step is setting up your clock and time zone. The installer will try to contact a time server on the Internet to ensure the clock is set correctly. The time zone is based on the country selected earlier and the installer will only ask to select one if a country has multiple zones.

Now it is time to partition your disks. First you will be given the opportunity to automatically partition either an entire drive, or available free space on a drive (see Secțiune 6.3.4.2). This is recommended for new users or anyone in a hurry. If you do not want to autopartition, choose Manual from the menu.

If you have an existing DOS or Windows partition that you want to preserve, be very careful with automatic partitioning. If you choose manual partitioning, you can use the installer to resize existing FAT or NTFS partitions to create room for the Debian install: simply select the partition and specify its new size.

On the next screen you will see your partition table, how the partitions will be formatted, and where they will be mounted. Select a partition to modify or delete it. If you did automatic partitioning, you should just be able to choose Finish partitioning and write changes to disk from the menu to use what it set up. Remember to assign at least one partition for swap space and to mount a partition on `/`. For more detailed information on how to use the partitioner, please refer to Secțiune 6.3.4; the appendix Anexa C has more general information about partitioning.

Now `debian-installer` formats your partitions and starts to install the base system, which can take a while. That is followed by installing a kernel.

The base system that was installed earlier is a working, but very minimal installation. To make the system more functional the next step allows you to install additional packages by selecting tasks. Before packages can be installed `apt` needs to be configured as that defines from where the packages will be retrieved. The „Standard system” task will be selected by default and should normally be installed. Select the „Desktop environment” task if you would like to have a graphical desktop after the installation. See Secțiune 6.3.6.2 for additional information about this step.

The last step is to install a boot loader. If the installer detects other operating systems on your computer, it will add them to the boot menu and let you know. By default GRUB will be installed to the master boot record of the

first harddrive, which is generally a good choice. You'll be given the opportunity to override that choice and install it elsewhere.

`debian-installer` will now tell you that the installation has finished. Remove the cdrom or other boot media and hit **Enter** to reboot your machine. It should boot up into the newly installed system and allow you to log in. This is explained in Cap. 7.

If you need more information on the install process, see Cap. 6.

A.4 Send us an installation report

If you successfully managed an installation with `debian-installer`, please take time to provide us with a report. The simplest way to do so is to install the reportbug package (**apt install reportbug**), configure `reportbug` as explained in Secțiune 8.4.2, and run **reportbug installation-reports**.

If you did not complete the install, you probably found a bug in debian-installer. To improve the installer it is necessary that we know about them, so please take the time to report them. You can use an installation report to report problems; if the install completely fails, see Secțiune 5.4.6.

A.5 And finally…

We hope that your Debian installation is pleasant and that you find Debian useful. You might want to read Cap. 8.

Anexa B

Automating the installation using preseeding

This appendix explains how to preseed answers to questions in `debian-installer` to automate your installation.

The configuration fragments used in this appendix are also available as an example preconfiguration file from `https://d-i.debian.org/manual/example-preseed.txt`.

B.1 Introduction

Preseeding provides a way to set answers to questions asked during the installation process, without having to manually enter the answers while the installation is running. This makes it possible to fully automate most types of installation and even offers some features not available during normal installations.

Preseeding is not required. If you use an empty preseed file, the installer will behave just the same way as in a normal manual installation. Each question you preseed will (if you got it right!) modify the installation in some way from that baseline.

B.1.1 Preseeding methods

There are three methods that can be used for preseeding: *initrd*, *file* and *network*. Initrd preseeding will work with any installation method and supports preseeding of more things, but it requires the most preparation. File and network preseeding each can be used with different installation methods.

The following table shows which preseeding methods can be used with which installation methods.

Installation method	**initrd**	**file**	**network**
CD/DVD/USB	yes	yes	yes[1]
netboot	yes	no	yes
hd-media (including usb-stick)	yes	yes	yes[1]

An important difference between the preseeding methods is the point at which the preconfiguration file is loaded and processed. For initrd preseeding this is right at the start of the installation, before the first question is even asked. Preseeding from the kernel command line happens just after. It is thus possible to override configuration set in the initrd by editing the kernel command line (either in the bootloader configuration or manually at boot time for bootloaders that allow it). For file preseeding this is after the installation image has been loaded. For network preseeding it is only after the network has been configured.

[1] but only if you have network access, and set `preseed/url` appropriately

IMPORTANT

Obviously, any questions that have been processed before the preconfiguration file is loaded cannot be preseeded (this will include questions that are only displayed at medium or low priority, like the first hardware detection run). A not so convenient way to avoid these questions from being asked is to preseed them through the boot parameters, as described in Secțiune B.2.2.

In order to easily avoid the questions that would normally appear before the preseeding occurs, you can start the installer in „auto" mode. This delays questions that would normally be asked too early for preseeding (i.e. language, country and keyboard selection) until after the network comes up, thus allowing them to be preseeded. It also runs the installation at critical priority, which avoids many unimportant questions. See Secțiune B.2.3 for details.

B.1.2 Limitations

Although most questions used by `debian-installer` can be preseeded using this method, there are some notable exceptions. You must (re)partition an entire disk or use available free space on a disk; it is not possible to use existing partitions.

B.2 Using preseeding

You will first need to create a preconfiguration file and place it in the location from where you want to use it. Creating the preconfiguration file is covered later in this appendix. Putting it in the correct location is fairly straightforward for network preseeding or if you want to read the file off a usb-stick. If you want to include the file in an installation ISO image, you will have to remaster the image. How to get the preconfiguration file included in the initrd is outside the scope of this document; please consult the developers' documentation for `debian-installer`.

An example preconfiguration file that you can use as basis for your own preconfiguration file is available from `https://d-i.debian.org/manual/example-preseed.txt`. This file is based on the configuration fragments included in this appendix.

B.2.1 Loading the preconfiguration file

If you are using initrd preseeding, you only have to make sure a file named `preseed.cfg` is included in the root directory of the initrd. The installer will automatically check if this file is present and load it.

For the other preseeding methods you need to tell the installer what file to use when you boot it. This is normally done by passing the kernel a boot parameter, either manually at boot time or by editing the bootloader configuration file (e.g. `syslinux.cfg`) and adding the parameter to the end of the append line(s) for the kernel.

If you do specify the preconfiguration file in the bootloader configuration, you might change the configuration so you don't need to hit enter to boot the installer. For syslinux this means setting the timeout to `1` in `syslinux.cfg`.

To make sure the installer gets the right preconfiguration file, you can optionally specify a checksum for the file. Currently this needs to be a md5sum, and if specified it must match the preconfiguration file or the installer will refuse to use it.

```
Boot parameters to specify:
- if you're netbooting:
  preseed/url=http://host/path/to/preseed.cfg
  preseed/url/checksum=5da499872becccfeda2c4872f9171c3d
- or
  preseed/url=tftp://host/path/to/preseed.cfg
  preseed/url/checksum=5da499872becccfeda2c4872f9171c3d

- if you're booting a remastered installation image:
  preseed/file=/cdrom/preseed.cfg
  preseed/file/checksum=5da499872becccfeda2c4872f9171c3d

- if you're installing from USB media (put the preconfiguration file in the
```

```
  toplevel directory of the USB stick):
  preseed/file=/hd-media/preseed.cfg
  preseed/file/checksum=5da499872becccfeda2c4872f9171c3d
```

Note that `preseed/url` can be shortened to just `url`, `preseed/file` to just `file` and `preseed/file/checksum` to just `preseed-md5` when they are passed as boot parameters.

B.2.2 Using boot parameters to preseed questions

If a preconfiguration file cannot be used to preseed some steps, the install can still be fully automated, since you can pass preseed values on the command line when booting the installer.

Boot parameters can also be used if you do not really want to use preseeding, but just want to provide an answer for a specific question. Some examples where this can be useful are documented elsewhere in this manual.

To set a value to be used inside `debian-installer`, just pass ***path/to/variable=value*** for any of the preseed variables listed in the examples in this appendix. If a value is to be used to configure packages for the target system, you will need to prepend the *owner*[2] of the variable as in ***owner:path/to/variable=value***. If you don't specify the owner, the value for the variable will not be copied to the debconf database in the target system and thus remain unused during the configuration of the relevant package.

Normally, preseeding a question in this way will mean that the question will not be asked. To set a specific default value for a question, but still have the question asked, use „?=" instead of „=" as operator. See also Secțiune B.5.2.

Note that some variables that are frequently set at the boot prompt have a shorter alias. If an alias is available, it is used in the examples in this appendix instead of the full variable. The `preseed/url` variable for example has been aliased as `url`. Another example is the `tasks` alias, which translates to `tasksel:tasksel/first`.

A „---" in the boot options has special meaning. Kernel parameters that appear after the last „---" may be copied into the bootloader configuration for the installed system (if supported by the installer for the bootloader). The installer will automatically filter out any options (like preconfiguration options) that it recognizes.

> Notă
>
> 
>
> Current linux kernels (2.6.9 and later) accept a maximum of 32 command line options and 32 environment options, including any options added by default for the installer. If these numbers are exceeded, the kernel will panic (crash). (For earlier kernels, these numbers were lower.)

For most installations some of the default options in your bootloader configuration file, like `vga=normal`, may be safely removed which may allow you to add more options for preseeding.

> Notă
>
> 
>
> It may not always be possible to specify values with spaces for boot parameters, even if you delimit them with quotes.

B.2.3 Auto mode

There are several features of Debian Installer that combine to allow fairly simple command lines at the boot prompt to result in arbitrarily complex customized automatic installs.

This is enabled by using the `Automated install` boot choice, also called `auto` for some architectures or boot methods. In this section, `auto` is thus not a parameter, it means selecting that boot choice, and appending the following boot parameters on the boot prompt. See Secțiune 5.1.7 for information on how to add a boot parameter.

To illustrate this, here are some examples that can be used at the boot prompt:

```
auto url=autoserver
```

[2] The owner of a debconf variable (or template) is normally the name of the package that contains the corresponding debconf template. For variables used in the installer itself the owner is „d-i". Templates and variables can have more than one owner which helps to determine whether they can be removed from the debconf database if the package is purged.

This relies on there being a DHCP server that will get the machine to the point where `autoserver` can be resolved by DNS, perhaps after adding the local domain if that was provided by DHCP. If this was done at a site where the domain is `example.com`, and they have a reasonably sane DHCP setup, it would result in the preseed file being retrieved from `http://autoserver.example.com/d-i/bullseye/./preseed.cfg`.

The last part of that url (`d-i/bullseye/./preseed.cfg`) is taken from `auto-install/defaultroot`. By default this includes the directory `bullseye` to allow future versions to specify their own codename and let people migrate forwards in a controlled manner. The `/./` bit is used to indicate a root, relative to which subsequent paths can be anchored (for use in preseed/include and preseed/run). This allows files to be specified either as full URLs, paths starting with / that are thus anchored, or even paths relative to the location where the last preseed file was found. This can be used to construct more portable scripts where an entire hierarchy of scripts can be moved to a new location without breaking it, for example copying the files onto a USB stick when they started out on a web server. In this example, if the preseed file sets `preseed/run` to `/scripts/late_command.sh` then the file will be fetched from `http://autoserver.example.com/d-i/bullseye/./scripts/late_command.sh`.

If there is no local DHCP or DNS infrastructure, or if you do not want to use the default path to `preseed.cfg`, you can still use an explicit url, and if you don't use the `/./` element it will be anchored to the start of the path (i.e. the third `/` in the URL). Here is an example that requires minimal support from the local network infrastructure:

```
auto url=http://192.168.1.2/path/to/mypreseed.file
```

The way this works is that:

- if the URL is missing a protocol, http is assumed,
- if the hostname section contains no periods, it has the domain derived from DHCP appended to it, and
- if there's no /'s after the hostname, then the default path is added.

In addition to specifying the url, you can also specify settings that do not directly affect the behavior of `debian-installer` itself, but can be passed through to scripts specified using `preseed/run` in the loaded preseed file. At present, the only example of this is `auto-install/classes`, which has an alias `classes`. This can be used thus:

```
auto url=example.com classes=class_A;class_B
```

The classes could for example denote the type of system to be installed, or the localization to be used.

It is of course possible to extend this concept, and if you do, it is reasonable to use the auto-install namespace for this. So one might have something like `auto-install/style` which is then used in your scripts. If you feel the need to do this, please mention it on the debian-boot@lists.debian.org mailing list so that we can avoid namespace conflicts, and perhaps add an alias for the parameter for you.

The `auto` boot choice is not yet defined on all arches. The same effect may be achieved by simply adding the two parameters `auto=true priority=critical` to the kernel command line. The `auto` kernel parameter is an alias for `auto-install/enable` and setting it to `true` delays the locale and keyboard questions until after there has been a chance to preseed them, while `priority` is an alias for `debconf/priority` and setting it to `critical` stops any questions with a lower priority from being asked.

Additional options that may be of interest while attempting to automate an install while using DHCP are: `interface=auto netcfg/dhcp_timeout=60` which makes the machine choose the first viable NIC and be more patient about getting a reply to its DHCP query.

> **INDICAȚIE**
>
> 
>
> An extensive example of how to use this framework, including example scripts and classes, can be found on the website of its developer. The examples available there also show many other nice effects that can be achieved by creative use of preconfiguration.

B.2.4 Aliases useful with preseeding

The following aliases can be useful when using (auto mode) preseeding. Note that these are simply short aliases for question names, and you always need to specify a value as well: for example, `auto=true` or `interface=eth0`.

priority	debconf/priority

fb	debian-installer/framebuffer
language	debian-installer/language
country	debian-installer/country
locale	debian-installer/locale
theme	debian-installer/theme
auto	auto-install/enable
classes	auto-install/classes
file	preseed/file
url	preseed/url
domain	netcfg/get_domain
hostname	netcfg/get_hostname
interface	netcfg/choose_interface
protocol	mirror/protocol
suite	mirror/suite
modules	anna/choose_modules
recommends	base-installer/install-recommends
tasks	tasksel:tasksel/first
desktop	tasksel:tasksel/desktop
dmraid	disk-detect/dmraid/enable
keymap	keyboard-configuration/xkb-keymap
preseed-md5	preseed/file/checksum

B.2.5 Using a DHCP server to specify preconfiguration files

It's also possible to use DHCP to specify a preconfiguration file to download from the network. DHCP allows specifying a filename. Normally this is a file to netboot, but if it appears to be an URL then installation media that support network preseeding will download the file from the URL and use it as a preconfiguration file. Here is an example of how to set it up in the dhcpd.conf for version 3 of the ISC DHCP server (the isc-dhcp-server Debian package).

```
if substring (option vendor-class-identifier, 0, 3) = "d-i" {
    filename "http://host/preseed.cfg";
}
```

Note that the above example limits this filename to DHCP clients that identify themselves as „d-i", so it will not affect regular DHCP clients, but only the installer. You can also put the text in a stanza for only one particular host to avoid preseeding all installs on your network.

A good way to use the DHCP preseeding is to only preseed values specific to your network, such as the Debian mirror to use. This way installs on your network will automatically get a good mirror selected, but the rest of the installation can be performed interactively. Using DHCP preseeding to fully automate Debian installs should only be done with care.

B.3 Creating a preconfiguration file

The preconfiguration file is in the format used by the **debconf-set-selections** command. The general format of a line in a preconfiguration file is:

```
<owner> <question name> <question type> <value>
```

The file should start with `#_preseed_V1`
There are a few rules to keep in mind when writing a preconfiguration file.

- Put only a single space or tab between type and value: any additional whitespace will be interpreted as belonging to the value.
- A line can be split into multiple lines by appending a backslash („\") as the line continuation character. A good place to split a line is after the question name; a bad place is between type and value. Split lines will be joined into a single line with all leading/trailing whitespace condensed to a single space.
- For debconf variables (templates) used only in the installer itself, the owner should be set to „d-i"; to preseed variables used in the installed system, the name of the package that contains the corresponding debconf template

should be used. Only variables that have their owner set to something other than „d-i" will be propagated to the debconf database for the installed system.

- Most questions need to be preseeded using the values valid in English and not the translated values. However, there are some questions (for example in `partman`) where the translated values need to be used.
- Some questions take a code as value instead of the English text that is shown during installation.
- Start with `#_preseed_V1`

The easiest way to create a preconfiguration file is to use the example file linked in Secțiune B.4 as basis and work from there.

An alternative method is to do a manual installation and then, after rebooting, use the **debconf-get-selections** from the `debconf-utils` package to dump both the debconf database and the installer's cdebconf database to a single file:

```
$ echo "#_preseed_V1" > file
$ debconf-get-selections --installer >> file
$ debconf-get-selections >> file
```

However, a file generated in this manner will have some items that should not be preseeded, and the example file is a better starting place for most users.

Notă

This method relies on the fact that, at the end of the installation, the installer's cdebconf database is saved to the installed system in `/var/log/installer/cdebconf`. However, because the database may contain sensitive information, by default the files are only readable by root.

The directory `/var/log/installer` and all files in it will be deleted from your system if you purge the package `installation-report`.

To check possible values for questions, you can use **nano** to examine the files in `/var/lib/cdebconf` while an installation is in progress. View `templates.dat` for the raw templates and `questions.dat` for the current values and for the values assigned to variables.

To check if the format of your preconfiguration file is valid before performing an install, you can use the command **debconf-set-selections -c *preseed.cfg***.

B.4 Contents of the preconfiguration file (for bullseye)

The configuration fragments used in this appendix are also available as an example preconfiguration file from https://d-i.debian.org/manual/example-preseed.txt.

Note that this example is based on an installation for the Intel x86 architecture. If you are installing a different architecture, some of the examples (like keyboard selection and bootloader installation) may not be relevant and will need to be replaced by debconf settings appropriate for your architecture.

Details on how the different Debian Installer components actually work can be found in Secțiune 6.3.

B.4.1 Localization

During a normal install the questions about localization are asked first, so these values can only be preseeded via the initrd or kernel boot parameter methods. Auto mode (Secțiune B.2.3) includes the setting of `auto-install/enable=true` (normally via the `auto` preseed alias). This delays the asking of the localisation questions, so that they can be preseeded by any method.

The locale can be used to specify both language and country and can be any combination of a language supported by `debian-installer` and a recognized country. If the combination does not form a valid locale, the installer will automatically select a locale that is valid for the selected language. To specify the locale as a boot parameter, use **`locale=en_US`**.

Although this method is very easy to use, it does not allow preseeding of all possible combinations of language, country and locale[3]. So alternatively the values can be preseeded individually. Language and country can also be specified as boot parameters.

```
# Preseeding only locale sets language, country and locale.
d-i debian-installer/locale string en_US

# The values can also be preseeded individually for greater flexibility.
#d-i debian-installer/language string en
#d-i debian-installer/country string NL
#d-i debian-installer/locale string en_GB.UTF-8
# Optionally specify additional locales to be generated.
#d-i localechooser/supported-locales multiselect en_US.UTF-8, nl_NL.UTF-8
```

Keyboard configuration consists of selecting a keymap and (for non-latin keymaps) a toggle key to switch between the non-latin keymap and the US keymap. Only basic keymap variants are available during installation. Advanced variants are available only in the installed system, through **dpkg-reconfigure keyboard-configuration**.

```
# Keyboard selection.
d-i keyboard-configuration/xkb-keymap select us
# d-i keyboard-configuration/toggle select No toggling
```

To skip keyboard configuration, preseed `keymap` with **`skip-config`**. This will result in the kernel keymap remaining active.

B.4.2 Network configuration

Of course, preseeding the network configuration won't work if you're loading your preconfiguration file from the network. But it's great when you're booting from optical disc or USB stick. If you are loading preconfiguration files from the network, you can pass network config parameters by using kernel boot parameters.

If you need to pick a particular interface when netbooting before loading a preconfiguration file from the network, use a boot parameter such as **`interface=`*`eth1`***.

Although preseeding the network configuration is normally not possible when using network preseeding (using „preseed/url"), you can use the following hack to work around that, for example if you'd like to set a static address for the network interface. The hack is to force the network configuration to run again after the preconfiguration file has been loaded by creating a „preseed/run" script containing the following commands:

```
kill-all-dhcp; netcfg
```

The following debconf variables are relevant for network configuration.

```
# Disable network configuration entirely. This is useful for cdrom
# installations on non-networked devices where the network questions,
# warning and long timeouts are a nuisance.
#d-i netcfg/enable boolean false

# netcfg will choose an interface that has link if possible. This makes it
# skip displaying a list if there is more than one interface.
d-i netcfg/choose_interface select auto

# To pick a particular interface instead:
#d-i netcfg/choose_interface select eth1

# To set a different link detection timeout (default is 3 seconds).
# Values are interpreted as seconds.
#d-i netcfg/link_wait_timeout string 10

# If you have a slow dhcp server and the installer times out waiting for
# it, this might be useful.
#d-i netcfg/dhcp_timeout string 60
#d-i netcfg/dhcpv6_timeout string 60
```

[3] Preseeding `locale` to **`en_NL`** would for example result in `en_US.UTF-8` as default locale for the installed system. If e.g. `en_GB.UTF-8` is preferred instead, the values will need to be preseeded individually.

```
# If you prefer to configure the network manually, uncomment this line and
# the static network configuration below.
#d-i netcfg/disable_autoconfig boolean true

# If you want the preconfiguration file to work on systems both with and
# without a dhcp server, uncomment these lines and the static network
# configuration below.
#d-i netcfg/dhcp_failed note
#d-i netcfg/dhcp_options select Configure network manually

# Static network configuration.
#
# IPv4 example
#d-i netcfg/get_ipaddress string 192.168.1.42
#d-i netcfg/get_netmask string 255.255.255.0
#d-i netcfg/get_gateway string 192.168.1.1
#d-i netcfg/get_nameservers string 192.168.1.1
#d-i netcfg/confirm_static boolean true
#
# IPv6 example
#d-i netcfg/get_ipaddress string fc00::2
#d-i netcfg/get_netmask string ffff:ffff:ffff:ffff::
#d-i netcfg/get_gateway string fc00::1
#d-i netcfg/get_nameservers string fc00::1
#d-i netcfg/confirm_static boolean true

# Any hostname and domain names assigned from dhcp take precedence over
# values set here. However, setting the values still prevents the questions
# from being shown, even if values come from dhcp.
d-i netcfg/get_hostname string unassigned-hostname
d-i netcfg/get_domain string unassigned-domain

# If you want to force a hostname, regardless of what either the DHCP
# server returns or what the reverse DNS entry for the IP is, uncomment
# and adjust the following line.
#d-i netcfg/hostname string somehost

# Disable that annoying WEP key dialog.
d-i netcfg/wireless_wep string
# The wacky dhcp hostname that some ISPs use as a password of sorts.
#d-i netcfg/dhcp_hostname string radish

# If non-free firmware is needed for the network or other hardware, you can
# configure the installer to always try to load it, without prompting. Or
# change to false to disable asking.
#d-i hw-detect/load_firmware boolean true
```

Please note that **netcfg** will automatically determine the netmask if `netcfg/get_netmask` is not preseeded. In this case, the variable has to be marked as `seen` for automatic installations. Similarly, **netcfg** will choose an appropriate address if `netcfg/get_gateway` is not set. As a special case, you can set `netcfg/get_gateway` to „none” to specify that no gateway should be used.

B.4.3 Network console

```
# Use the following settings if you wish to make use of the network-console
# component for remote installation over SSH. This only makes sense if you
# intend to perform the remainder of the installation manually.
#d-i anna/choose_modules string network-console
#d-i network-console/authorized_keys_url string http://10.0.0.1/openssh-key
#d-i network-console/password password r00tme
#d-i network-console/password-again password r00tme
```

B.4.4 Mirror settings

Depending on the installation method you use, a mirror may be used to download additional components of the installer, to install the base system, and to set up the `/etc/apt/sources.list` for the installed system.

The parameter `mirror/suite` determines the suite for the installed system.

The parameter `mirror/udeb/suite` determines the suite for additional components for the installer. It is only useful to set this if components are actually downloaded over the network and should match the suite that was used to build the initrd for the installation method used for the installation. Normally the installer will automatically use the correct value and there should be no need to set this.

```
# If you select ftp, the mirror/country string does not need to be set.
#d-i mirror/protocol string ftp
d-i mirror/country string manual
d-i mirror/http/hostname string http.us.debian.org
d-i mirror/http/directory string /debian
d-i mirror/http/proxy string

# Suite to install.
#d-i mirror/suite string testing
# Suite to use for loading installer components (optional).
#d-i mirror/udeb/suite string testing
```

B.4.5 Account setup

The password for the root account and name and password for a first regular user's account can be preseeded. For the passwords you can use either clear text values or crypt(3) *hashes*.

AVERTISMENT

Be aware that preseeding passwords is not completely secure as everyone with access to the preconfiguration file will have the knowledge of these passwords. Storing hashed passwords is considered secure unless a weak hashing algorithm like DES or MD5 is used which allow for bruteforce attacks. Recommended password hashing algorithms are SHA-256 and SHA512.

```
# Skip creation of a root account (normal user account will be able to
# use sudo).
#d-i passwd/root-login boolean false
# Alternatively, to skip creation of a normal user account.
#d-i passwd/make-user boolean false

# Root password, either in clear text
#d-i passwd/root-password password r00tme
#d-i passwd/root-password-again password r00tme
# or encrypted using a crypt(3)  hash.
#d-i passwd/root-password-crypted password [crypt(3) hash]

# To create a normal user account.
#d-i passwd/user-fullname string Debian User
#d-i passwd/username string debian
# Normal user's password, either in clear text
#d-i passwd/user-password password insecure
#d-i passwd/user-password-again password insecure
# or encrypted using a crypt(3) hash.
#d-i passwd/user-password-crypted password [crypt(3) hash]
# Create the first user with the specified UID instead of the default.
#d-i passwd/user-uid string 1010

# The user account will be added to some standard initial groups. To
# override that, use this.
#d-i passwd/user-default-groups string audio cdrom video
```

The `passwd/root-password-crypted` and `passwd/user-password-crypted` variables can also be preseeded with „!” as their value. In that case, the corresponding account is disabled. This may be convenient for the root account, provided of course that an alternative method is set up to allow administrative activities or root login (for instance by using SSH key authentication or **sudo**).

The following command (available from the `whois` package) can be used to generate a SHA-512 based crypt(3) hash for a password:

```
mkpasswd -m sha-512
```

B.4.6 Clock and time zone setup

```
# Controls whether or not the hardware clock is set to UTC.
d-i clock-setup/utc boolean true

# You may set this to any valid setting for $TZ; see the contents of
# /usr/share/zoneinfo/ for valid values.
d-i time/zone string US/Eastern

# Controls whether to use NTP to set the clock during the install
d-i clock-setup/ntp boolean true
# NTP server to use. The default is almost always fine here.
#d-i clock-setup/ntp-server string ntp.example.com
```

B.4.7 Partitioning

Using preseeding to partition the harddisk is limited to what is supported by `partman-auto`. You can choose to partition either existing free space on a disk or a whole disk. The layout of the disk can be determined by using a predefined recipe, a custom recipe from a recipe file or a recipe included in the preconfiguration file.

Preseeding of advanced partition setups using RAID, LVM and encryption is supported, but not with the full flexibility possible when partitioning during a non-preseeded install.

The examples below only provide basic information on the use of recipes. For detailed information see the files `partman-auto-recipe.txt` and `partman-auto-raid-recipe.txt` included in the `debian-installer` package. Both files are also available from the debian-installer source repository. Note that the supported functionality may change between releases.

AVERTISMENT

The identification of disks is dependent on the order in which their drivers are loaded. If there are multiple disks in the system, make very sure the correct one will be selected before using preseeding.

B.4.7.1 Partitioning example

```
# If the system has free space you can choose to only partition that space.
# This is only honoured if partman-auto/method (below) is not set.
#d-i partman-auto/init_automatically_partition select biggest_free

# Alternatively, you may specify a disk to partition. If the system has only
# one disk the installer will default to using that, but otherwise the device
# name must be given in traditional, non-devfs format (so e.g. /dev/sda
# and not e.g. /dev/discs/disc0/disc).
# For example, to use the first SCSI/SATA hard disk:
#d-i partman-auto/disk string /dev/sda
# In addition, you'll need to specify the method to use.
# The presently available methods are:
# - regular: use the usual partition types for your architecture
# - lvm:     use LVM to partition the disk
```

```
# - crypto:  use LVM within an encrypted partition
d-i partman-auto/method string lvm

# You can define the amount of space that will be used for the LVM volume
# group. It can either be a size with its unit (eg. 20 GB), a percentage of
# free space or the 'max' keyword.
d-i partman-auto-lvm/guided_size string max

# If one of the disks that are going to be automatically partitioned
# contains an old LVM configuration, the user will normally receive a
# warning. This can be preseeded away...
d-i partman-lvm/device_remove_lvm boolean true
# The same applies to pre-existing software RAID array:
d-i partman-md/device_remove_md boolean true
# And the same goes for the confirmation to write the lvm partitions.
d-i partman-lvm/confirm boolean true
d-i partman-lvm/confirm_nooverwrite boolean true

# You can choose one of the three predefined partitioning recipes:
# - atomic: all files in one partition
# - home:   separate /home partition
# - multi:  separate /home, /var, and /tmp partitions
d-i partman-auto/choose_recipe select atomic

# Or provide a recipe of your own...
# If you have a way to get a recipe file into the d-i environment, you can
# just point at it.
#d-i partman-auto/expert_recipe_file string /hd-media/recipe

# If not, you can put an entire recipe into the preconfiguration file in one
# (logical) line. This example creates a small /boot partition, suitable
# swap, and uses the rest of the space for the root partition:
#d-i partman-auto/expert_recipe string                         \
#      boot-root ::                                            \
#              40 50 100 ext3                                  \
#                      $primary{ } $bootable{ }                \
#                      method{ format } format{ }              \
#                      use_filesystem{ } filesystem{ ext3 }    \
#                      mountpoint{ /boot }                     \
#              .                                               \
#              500 10000 1000000000 ext3                       \
#                      method{ format } format{ }              \
#                      use_filesystem{ } filesystem{ ext3 }    \
#                      mountpoint{ / }                         \
#              .                                               \
#              64 512 300% linux-swap                          \
#                      method{ swap } format{ }                \
#              .

# The full recipe format is documented in the file partman-auto-recipe.txt
# included in the 'debian-installer' package or available from D-I source
# repository. This also documents how to specify settings such as file
# system labels, volume group names and which physical devices to include
# in a volume group.

# This makes partman automatically partition without confirmation, provided
# that you told it what to do using one of the methods above.
d-i partman-partitioning/confirm_write_new_label boolean true
d-i partman/choose_partition select finish
d-i partman/confirm boolean true
d-i partman/confirm_nooverwrite boolean true

# When disk encryption is enabled, skip wiping the partitions beforehand.
#d-i partman-auto-crypto/erase_disks boolean false
```

B.4.7.2 Partitioning using RAID

You can also use preseeding to set up partitions on software RAID arrays. Supported are RAID levels 0, 1, 5, 6 and 10, creating degraded arrays and specifying spare devices.

If you are using RAID 1, you can preseed grub to install to all devices used in the array; see Secțiune B.4.11.

AVERTISMENT

This type of automated partitioning is easy to get wrong. It is also functionality that receives relatively little testing from the developers of `debian-installer`. The responsibility to get the various recipes right (so they make sense and don't conflict) lies with the user. Check `/var/log/syslog` if you run into problems.

```
# The method should be set to "raid".
#d-i partman-auto/method string raid
# Specify the disks to be partitioned. They will all get the same layout,
# so this will only work if the disks are the same size.
#d-i partman-auto/disk string /dev/sda /dev/sdb

# Next you need to specify the physical partitions that will be used.
#d-i partman-auto/expert_recipe string \
#      multiraid ::                                          \
#              1000 5000 4000 raid                           \
#                      $primary{ } method{ raid }            \
#              .                                             \
#              64 512 300% raid                              \
#                      method{ raid }                        \
#              .                                             \
#              500 10000 1000000000 raid                     \
#                      method{ raid }                        \
#              .

# Last you need to specify how the previously defined partitions will be
# used in the RAID setup. Remember to use the correct partition numbers
# for logical partitions. RAID levels 0, 1, 5, 6 and 10 are supported;
# devices are separated using "#".
# Parameters are:
# <raidtype> <devcount> <sparecount> <fstype> <mountpoint> \
#          <devices> <sparedevices>

#d-i partman-auto-raid/recipe string \
#    1 2 0 ext3 /                    \
#          /dev/sda1#/dev/sdb1       \
#    .                               \
#    1 2 0 swap -                    \
#          /dev/sda5#/dev/sdb5       \
#    .                               \
#    0 2 0 ext3 /home                \
#          /dev/sda6#/dev/sdb6       \
#    .

# For additional information see the file partman-auto-raid-recipe.txt
# included in the 'debian-installer' package or available from D-I source
# repository.

# This makes partman automatically partition without confirmation.
d-i partman-md/confirm boolean true
d-i partman-partitioning/confirm_write_new_label boolean true
d-i partman/choose_partition select finish
d-i partman/confirm boolean true
d-i partman/confirm_nooverwrite boolean true
```

B.4.7.3 Controlling how partitions are mounted

Normally, filesystems are mounted using a universally unique identifier (UUID) as a key; this allows them to be mounted properly even if their device name changes. UUIDs are long and difficult to read, so, if you prefer, the installer can mount filesystems based on the traditional device names, or based on a label you assign. If you ask the installer to mount by label, any filesystems without a label will be mounted using a UUID instead.

Devices with stable names, such as LVM logical volumes, will continue to use their traditional names rather than UUIDs.

AVERTISMENT

Traditional device names may change based on the order in which the kernel discovers devices at boot, which may cause the wrong filesystem to be mounted. Similarly, labels are likely to clash if you plug in a new disk or a USB drive, and if that happens your system's behaviour when started will be random.

```
# The default is to mount by UUID, but you can also choose "traditional" to
# use traditional device names, or "label" to try filesystem labels before
# falling back to UUIDs.
#d-i partman/mount_style select uuid
```

B.4.8 Base system installation

There is actually not very much that can be preseeded for this stage of the installation. The only questions asked concern the installation of the kernel.

```
# Configure APT to not install recommended packages by default. Use of this
# option can result in an incomplete system and should only be used by very
# experienced users.
#d-i base-installer/install-recommends boolean false

# The kernel image (meta) package to be installed; "none" can be used if no
# kernel is to be installed.
#d-i base-installer/kernel/image string linux-image-686
```

B.4.9 Apt setup

Setup of the `/etc/apt/sources.list` and basic configuration options is fully automated based on your installation method and answers to earlier questions. You can optionally add other (local) repositories.

```
# You can choose to install non-free and contrib software.
#d-i apt-setup/non-free boolean true
#d-i apt-setup/contrib boolean true
# Uncomment this if you don't want to use a network mirror.
#d-i apt-setup/use_mirror boolean false
# Select which update services to use; define the mirrors to be used.
# Values shown below are the normal defaults.
#d-i apt-setup/services-select multiselect security, updates
#d-i apt-setup/security_host string security.debian.org

# Additional repositories, local[0-9] available
#d-i apt-setup/local0/repository string \
#       http://local.server/debian stable main
#d-i apt-setup/local0/comment string local server
# Enable deb-src lines
#d-i apt-setup/local0/source boolean true
# URL to the public key of the local repository; you must provide a key or
# apt will complain about the unauthenticated repository and so the
# sources.list line will be left commented out.
#d-i apt-setup/local0/key string http://local.server/key
```

```
# If the provided key file ends in ".asc" the key file needs to be an
# ASCII-armoured PGP key, if it ends in ".gpg" it needs to use the
# "GPG key public keyring" format, the "keybox database" format is
# currently not supported.

# By default the installer requires that repositories be authenticated
# using a known gpg key. This setting can be used to disable that
# authentication. Warning: Insecure, not recommended.
#d-i debian-installer/allow_unauthenticated boolean true

# Uncomment this to add multiarch configuration for i386
#d-i apt-setup/multiarch string i386
```

B.4.10 Package selection

You can choose to install any combination of tasks that are available. Available tasks as of this writing include:

- **standard** (standard tools)
- **desktop** (graphical desktop)
- **gnome-desktop** (Gnome desktop)
- **xfce-desktop** (XFCE desktop)
- **kde-desktop** (KDE Plasma desktop)
- **cinnamon-desktop** (Cinnamon desktop)
- **mate-desktop** (MATE desktop)
- **lxde-desktop** (LXDE desktop)
- **web-server** (web server)
- **print-server** (print server)
- **ssh-server** (SSH server)

You can also choose to install no tasks, and force the installation of a set of packages in some other way. We recommend always including the **standard** task.

If you want to install some individual packages in addition to packages installed by tasks, you can use the parameter `pkgsel/include`. The value of this parameter can be a list of packages separated by either commas or spaces, which allows it to be used easily on the kernel command line as well.

```
#tasksel tasksel/first multiselect standard, web-server, kde-desktop

# Individual additional packages to install
#d-i pkgsel/include string openssh-server build-essential
# Whether to upgrade packages after debootstrap.
# Allowed values: none, safe-upgrade, full-upgrade
#d-i pkgsel/upgrade select none

# Some versions of the installer can report back on what software you have
# installed, and what software you use. The default is not to report back,
# but sending reports helps the project determine what software is most
# popular and should be included on the first CD/DVD.
#popularity-contest popularity-contest/participate boolean false
```

B.4.11 Boot loader installation

```
# Grub is the default boot loader (for x86). If you want lilo installed
# instead, uncomment this:
#d-i grub-installer/skip boolean true
# To also skip installing lilo, and install no bootloader, uncomment this
# too:
#d-i lilo-installer/skip boolean true


# This is fairly safe to set, it makes grub install automatically to the MBR
# if no other operating system is detected on the machine.
d-i grub-installer/only_debian boolean true

# This one makes grub-installer install to the MBR if it also finds some other
# OS, which is less safe as it might not be able to boot that other OS.
d-i grub-installer/with_other_os boolean true

# Due notably to potential USB sticks, the location of the MBR can not be
# determined safely in general, so this needs to be specified:
#d-i grub-installer/bootdev  string /dev/sda
# To install to the first device (assuming it is not a USB stick):
#d-i grub-installer/bootdev  string default

# Alternatively, if you want to install to a location other than the mbr,
# uncomment and edit these lines:
#d-i grub-installer/only_debian boolean false
#d-i grub-installer/with_other_os boolean false
#d-i grub-installer/bootdev  string (hd0,1)
# To install grub to multiple disks:
#d-i grub-installer/bootdev  string (hd0,1) (hd1,1) (hd2,1)

# Optional password for grub, either in clear text
#d-i grub-installer/password password r00tme
#d-i grub-installer/password-again password r00tme
# or encrypted using an MD5 hash, see grub-md5-crypt(8).
#d-i grub-installer/password-crypted password [MD5 hash]

# Use the following option to add additional boot parameters for the
# installed system (if supported by the bootloader installer).
# Note: options passed to the installer will be added automatically.
#d-i debian-installer/add-kernel-opts string nousb
```

An MD5 hash for a password for `grub` can be generated using **grub-md5-crypt**, or using the command from the example in Secțiune B.4.5.

B.4.12 Finishing up the installation

```
# During installations from serial console, the regular virtual consoles
# (VT1-VT6) are normally disabled in /etc/inittab. Uncomment the next
# line to prevent this.
#d-i finish-install/keep-consoles boolean true

# Avoid that last message about the install being complete.
d-i finish-install/reboot_in_progress note

# This will prevent the installer from ejecting the CD during the reboot,
# which is useful in some situations.
#d-i cdrom-detect/eject boolean false

# This is how to make the installer shutdown when finished, but not
# reboot into the installed system.
#d-i debian-installer/exit/halt boolean true
```

```
# This will power off the machine instead of just halting it.
#d-i debian-installer/exit/poweroff boolean true
```

B.4.13 Preseeding other packages

```
# Depending on what software you choose to install, or if things go wrong
# during the installation process, it's possible that other questions may
# be asked. You can preseed those too, of course. To get a list of every
# possible question that could be asked during an install, do an
# installation, and then run these commands:
#   debconf-get-selections --installer > file
#   debconf-get-selections >> file
```

B.5 Advanced options

B.5.1 Running custom commands during the installation

A very powerful and flexible option offered by the preconfiguration tools is the ability to run commands or scripts at certain points in the installation.

When the filesystem of the target system is mounted, it is available in `/target`. If an installation CD is used, when it is mounted it is available in `/cdrom`.

```
# d-i preseeding is inherently not secure. Nothing in the installer checks
# for attempts at buffer overflows or other exploits of the values of a
# preconfiguration file like this one. Only use preconfiguration files from
# trusted locations! To drive that home, and because it's generally useful,
# here's a way to run any shell command you'd like inside the installer,
# automatically.

# This first command is run as early as possible, just after
# preseeding is read.
#d-i preseed/early_command string anna-install some-udeb
# This command is run immediately before the partitioner starts. It may be
# useful to apply dynamic partitioner preseeding that depends on the state
# of the disks (which may not be visible when preseed/early_command runs).
#d-i partman/early_command \
#       string debconf-set partman-auto/disk "$(list-devices disk | head -n1)"
# This command is run just before the install finishes, but when there is
# still a usable /target directory. You can chroot to /target and use it
# directly, or use the apt-install and in-target commands to easily install
# packages and run commands in the target system.
#d-i preseed/late_command string apt-install zsh; in-target chsh -s /bin/zsh
```

B.5.2 Using preseeding to change default values

It is possible to use preseeding to change the default answer for a question, but still have the question asked. To do this the *seen* flag must be reset to „false” after setting the value for a question.

```
d-i foo/bar string value
d-i foo/bar seen false
```

The same effect can be achieved for *all* questions by setting the parameter `preseed/interactive=true` at the boot prompt. This can also be useful for testing or debugging your preconfiguration file.

Note that the „d-i” owner should only be used for variables used in the installer itself. For variables belonging to packages installed on the target system, you should use the name of that package instead. See the footnote to Secțiune B.2.2.

If you are preseeding using boot parameters, you can make the installer ask the corresponding question by using the „?=” operator, i.e. ***foo/bar?=value*** (or ***owner:foo/bar?=value***). This will of course only have effect for parameters that correspond to questions that are actually displayed during an installation and not for „internal” parameters.

For more debugging information, use the boot parameter `DEBCONF_DEBUG=5`. This will cause `debconf` to print much more detail about the current settings of each variable and about its progress through each package's installation scripts.

B.5.3 Chainloading preconfiguration files

It is possible to include other preconfiguration files from a preconfiguration file. Any settings in those files will override pre-existing settings from files loaded earlier. This makes it possible to put, for example, general networking settings for your location in one file and more specific settings for certain configurations in other files.

```
# More than one file can be listed, separated by spaces; all will be
# loaded. The included files can have preseed/include directives of their
# own as well. Note that if the filenames are relative, they are taken from
# the same directory as the preconfiguration file that includes them.
#d-i preseed/include string x.cfg

# The installer can optionally verify checksums of preconfiguration files
# before using them. Currently only md5sums are supported, list the md5sums
# in the same order as the list of files to include.
#d-i preseed/include/checksum string 5da499872becccfeda2c4872f9171c3d

# More flexibly, this runs a shell command and if it outputs the names of
# preconfiguration files, includes those files.
#d-i preseed/include_command \
#      string if [ "`hostname`" = bob ]; then echo bob.cfg; fi

# Most flexibly of all, this downloads a program and runs it. The program
# can use commands such as debconf-set to manipulate the debconf database.
# More than one script can be listed, separated by spaces.
# Note that if the filenames are relative, they are taken from the same
# directory as the preconfiguration file that runs them.
#d-i preseed/run string foo.sh
```

It is also possible to chainload from the initrd or file preseeding phase, into network preseeding by setting preseed/url in the earlier files. This will cause network preseeding to be performed when the network comes up. You need to be careful when doing this, since there will be two distinct runs at preseeding, meaning for example that you get another chance to run the preseed/early command, the second one happening after the network comes up.

Anexa C

Partitioning for Debian

C.1 Deciding on Debian Partitions and Sizes

At a bare minimum, GNU/Linux needs one partition for itself. You can have a single partition containing the entire operating system, applications, and your personal files. Most people feel that a separate swap partition is also a necessity, although it's not strictly true. „Swap" is scratch space for an operating system, which allows the system to use disk storage as „virtual memory". By putting swap on a separate partition, Linux can make much more efficient use of it. It is possible to force Linux to use a regular file as swap, but it is not recommended.

Most people choose to give GNU/Linux more than the minimum number of partitions, however. There are two reasons you might want to break up the file system into a number of smaller partitions. The first is for safety. If something happens to corrupt the file system, generally only one partition is affected. Thus, you only have to replace (from the backups you've been carefully keeping) a portion of your system. At a bare minimum, you should consider creating what is commonly called a „root partition". This contains the most essential components of the system. If any other partitions get corrupted, you can still boot into GNU/Linux to fix the system. This can save you the trouble of having to reinstall the system from scratch.

The second reason is generally more important in a business setting, but it really depends on your use of the machine. For example, a mail server getting spammed with e-mail can easily fill a partition. If you made `/var/mail` a separate partition on the mail server, most of the system will remain working even if you get spammed.

The only real drawback to using more partitions is that it is often difficult to know in advance what your needs will be. If you make a partition too small then you will either have to reinstall the system or you will be constantly moving things around to make room in the undersized partition. On the other hand, if you make the partition too big, you will be wasting space that could be used elsewhere. Disk space is cheap nowadays, but why throw your money away?

C.2 The Directory Tree

Debian GNU/Linux adheres to the Filesystem Hierarchy Standard for directory and file naming. This standard allows users and software programs to predict the location of files and directories. The root level directory is represented simply by the slash `/`. At the root level, all Debian systems include these directories:

Directory	**Content**
`bin`	Essential command binaries
`boot`	Static files of the boot loader
`dev`	Device files
`etc`	Host-specific system configuration
`home`	User home directories
`lib`	Essential shared libraries and kernel modules
`media`	Contains mount points for replaceable media
`mnt`	Mount point for mounting a file system temporarily
`proc`	Virtual directory for system information
`root`	Home directory for the root user
`run`	Run-time variable data
`sbin`	Essential system binaries
`sys`	Virtual directory for system information
`tmp`	Temporary files

Directory	Content
usr	Secondary hierarchy
var	Variable data
srv	Data for services provided by the system
opt	Add-on application software packages

The following is a list of important considerations regarding directories and partitions. Note that disk usage varies widely given system configuration and specific usage patterns. The recommendations here are general guidelines and provide a starting point for partitioning.

- The root partition / must always physically contain /etc, /bin, /sbin, /lib and /dev, otherwise you won't be able to boot. Typically 250–350MB is needed for the root partition.
- /usr: contains all user programs (/usr/bin), libraries (/usr/lib), documentation (/usr/share/doc), etc. This is the part of the file system that generally takes up most space. You should provide at least 500MB of disk space. This amount should be increased depending on the number and type of packages you plan to install. A generous workstation or server installation should allow 4–6GB.
- It is now recommended to have /usr on the root partition /, otherwise it could cause some trouble at boot time. This means that you should provide at least 600–750MB of disk space for the root partition including /usr, or 5–6GB for a workstation or a server installation.
- /var: variable data like news articles, e-mails, web sites, databases, the packaging system cache, etc. will be placed under this directory. The size of this directory depends greatly on the usage of your system, but for most people will be dictated by the package management tool's overhead. If you are going to do a full installation of just about everything Debian has to offer, all in one session, setting aside 2 or 3 GB of space for /var should be sufficient. If you are going to install in pieces (that is to say, install services and utilities, followed by text stuff, then X, ...), you can get away with 300–500 MB. If hard drive space is at a premium and you don't plan on doing major system updates, you can get by with as little as 30 or 40 MB.
- /tmp: temporary data created by programs will most likely go in this directory. 40–100MB should usually be enough. Some applications — including archive manipulators, CD/DVD authoring tools, and multimedia software — may use /tmp to temporarily store image files. If you plan to use such applications, you should adjust the space available in /tmp accordingly.
- /home: every user will put his personal data into a subdirectory of this directory. Its size depends on how many users will be using the system and what files are to be stored in their directories. Depending on your planned usage you should reserve about 100MB for each user, but adapt this value to your needs. Reserve a lot more space if you plan to save a lot of multimedia files (pictures, MP3, movies) in your home directory.

C.3 Recommended Partitioning Scheme

For new users, personal Debian boxes, home systems, and other single-user setups, a single / partition (plus swap) is probably the easiest, simplest way to go. The recommended partition type is ext4.

For multi-user systems or systems with lots of disk space, it's best to put /var, /tmp, and /home each on their own partitions separate from the / partition.

You might need a separate /usr/local partition if you plan to install many programs that are not part of the Debian distribution. If your machine will be a mail server, you might need to make /var/mail a separate partition. If you are setting up a server with lots of user accounts, it's generally good to have a separate, large /home partition. In general, the partitioning situation varies from computer to computer depending on its uses.

For very complex systems, you should see the Multi Disk HOWTO. This contains in-depth information, mostly of interest to ISPs and people setting up servers.

With respect to the issue of swap partition size, there are many views. One rule of thumb which works well is to use as much swap as you have system memory. It also shouldn't be smaller than 512MB, in most cases. Of course, there are exceptions to these rules.

As an example, an older home machine might have 512MB of RAM and a 20GB SATA drive on /dev/sda. There might be a 8GB partition for another operating system on /dev/sda1, a 512MB swap partition on /dev/sda3 and about 11.4GB on /dev/sda2 as the Linux partition.

For an idea of the space taken by tasks you might be interested in adding after your system installation is complete, check Secțiune D.2.

C.4 Device Names in Linux

Linux disks and partition names may be different from other operating systems. You need to know the names that Linux uses when you create and mount partitions. Here's the basic naming scheme:

- The first hard disk detected is named `/dev/sda`.
- The second hard disk detected is named `/dev/sdb`, and so on.
- The first SCSI CD-ROM is named `/dev/scd0`, also known as `/dev/sr0`.

The partitions on each disk are represented by appending a decimal number to the disk name: `sda1` and `sda2` represent the first and second partitions of the first SCSI disk drive in your system.

Here is a real-life example. Let's assume you have a system with 2 SCSI disks, one at SCSI address 2 and the other at SCSI address 4. The first disk (at address 2) is then named `sda`, and the second `sdb`. If the `sda` drive has 3 partitions on it, these will be named `sda1`, `sda2`, and `sda3`. The same applies to the `sdb` disk and its partitions.

Note that if you have two SCSI host bus adapters (i.e., controllers), the order of the drives can get confusing. The best solution in this case is to watch the boot messages, assuming you know the drive models and/or capacities.

Linux represents the primary partitions as the drive name, plus the numbers 1 through 4. For example, the first primary partition on the first drive is `/dev/sda1`. The logical partitions are numbered starting at 5, so the first logical partition on that same drive is `/dev/sda5`. Remember that the extended partition, that is, the primary partition holding the logical partitions, is not usable by itself.

C.5 Debian Partitioning Programs

Several varieties of partitioning programs have been adapted by Debian developers to work on various types of hard disks and computer architectures. Following is a list of the program(s) applicable for your architecture.

partman Recommended partitioning tool in Debian. This Swiss army knife can also resize partitions, create filesystems („format" in Windows speak) and assign them to the mountpoints.

fdisk The original Linux disk partitioner, good for gurus.

Be careful if you have existing FreeBSD partitions on your machine. The installation kernels include support for these partitions, but the way that **fdisk** represents them (or not) can make the device names differ. See the Linux+FreeBSD HOWTO.

cfdisk A simple-to-use, full-screen disk partitioner for the rest of us.

Note that **cfdisk** doesn't understand FreeBSD partitions at all, and, again, device names may differ as a result.

One of these programs will be run by default when you select Partition disks (or similar). It may be possible to use a different partitioning tool from the command line on VT2, but this is not recommended.

Remember to mark your boot partition as „Bootable".

C.5.1 Partitioning for 64-bit PC

If you have an existing other operating system such as DOS or Windows and you want to preserve that operating system while installing Debian, you may need to resize its partition to free up space for the Debian installation. The installer supports resizing of both FAT and NTFS filesystems; when you get to the installer's partitioning step, select the option Manual and then simply select an existing partition and change its size.

The PC BIOS generally adds additional constraints for disk partitioning. There is a limit to how many „primary" and „logical" partitions a drive can contain. Additionally, with pre 1994–98 BIOSes, there are limits to where on the drive the BIOS can boot from. More information can be found in the Linux Partition HOWTO, but this section will include a brief overview to help you plan most situations.

„Primary" partitions are the original partitioning scheme for PC disks. However, there can only be four of them. To get past this limitation, „extended" and „logical" partitions were invented. By setting one of your primary partitions as an extended partition, you can subdivide all the space allocated to that partition into logical partitions. You can create up to 60 logical partitions per extended partition; however, you can only have one extended partition per drive.

Linux limits the partitions per drive to 255 partitions for SCSI disks (3 usable primary partitions, 252 logical partitions), and 63 partitions on an IDE drive (3 usable primary partitions, 60 logical partitions). However the normal Debian GNU/Linux system provides only 20 devices for partitions, so you may not install on partitions higher than 20 unless you first manually create devices for those partitions.

If you have a large IDE disk, and are using neither LBA addressing, nor overlay drivers (sometimes provided by hard disk manufacturers), then the boot partition (the partition containing your kernel image) must be placed within the first 1024 cylinders of your hard drive (usually around 524 megabytes, without BIOS translation).

This restriction doesn't apply if you have a BIOS newer than around 1995–98 (depending on the manufacturer) that supports the „Enhanced Disk Drive Support Specification". Both Lilo, the Linux loader, and Debian's alternative **mbr** must use the BIOS to read the kernel from the disk into RAM. If the BIOS int 0x13 large disk access extensions are found to be present, they will be utilized. Otherwise, the legacy disk access interface is used as a fall-back, and it cannot be used to address any location on the disk higher than the 1023rd cylinder. Once Linux is booted, no matter what BIOS your computer has, these restrictions no longer apply, since Linux does not use the BIOS for disk access.

If you have a large disk, you might have to use cylinder translation techniques, which you can set from your BIOS setup program, such as LBA (Logical Block Addressing) or CHS translation mode („Large"). More information about issues with large disks can be found in the Large Disk HOWTO. If you are using a cylinder translation scheme, and the BIOS does not support the large disk access extensions, then your boot partition has to fit within the *translated* representation of the 1024th cylinder.

The recommended way of accomplishing this is to create a small (25–50MB should suffice) partition at the beginning of the disk to be used as the boot partition, and then create whatever other partitions you wish to have, in the remaining area. This boot partition *must* be mounted on `/boot`, since that is the directory where the Linux kernel(s) will be stored. This configuration will work on any system, regardless of whether LBA or large disk CHS translation is used, and regardless of whether your BIOS supports the large disk access extensions.

Anexa D

Random Bits

D.1 Linux Devices

In Linux various special files can be found under the directory `/dev`. These files are called device files and behave unlike ordinary files. The most common types of device files are for block devices and character devices. These files are an interface to the actual driver (part of the Linux kernel) which in turn accesses the hardware. Another, less common, type of device file is the named *pipe*. The most important device files are listed in the tables below.

`sda`	First hard disk
`sdb`	Second hard disk
`sda1`	First partition of the first hard disk
`sdb7`	Seventh partition of the second hard disk

`sr0`	First CD-ROM
`sr1`	Second CD-ROM

`ttyS0`	Serial port 0, COM1 under MS-DOS
`ttyS1`	Serial port 1, COM2 under MS-DOS
`psaux`	PS/2 mouse device
`gpmdata`	Pseudo device, repeater data from GPM (mouse) daemon

`cdrom`	Symbolic link to the CD-ROM drive
`mouse`	Symbolic link to the mouse device file

`null`	Anything written to this device will disappear
`zero`	One can endlessly read zeros out of this device

D.1.1 Setting Up Your Mouse

The mouse can be used in both the Linux console (with gpm) and the X window environment. Normally, this is a simple matter of installing `gpm` and the X server itself. Both should be configured to use `/dev/input/mice` as the mouse device. The correct mouse protocol is named **`exps2`** in gpm, and **`ExplorerPS/2`** in X. The respective configuration files are `/etc/gpm.conf` and `/etc/X11/xorg.conf`.

Certain kernel modules must be loaded in order for your mouse to work. In most cases the correct modules are autodetected, but not always for old-style serial and bus mice[1], which are quite rare except on very old computers. Summary of Linux kernel modules needed for different mouse types:

[1] Serial mice usually have a 9-hole D-shaped connector; bus mice have an 8-pin round connector, not to be confused with the 6-pin round connector of a PS/2 mouse or the 4-pin round connector of an ADB mouse.

Module	Description
psmouse	PS/2 mice (should be autodetected)
usbhid	USB mice (should be autodetected)
sermouse	Most serial mice
logibm	Bus mouse connected to Logitech adapter card
inport	Bus mouse connected to ATI or Microsoft InPort card

To load a mouse driver module, you can use the **modconf** command (from the package with the same name) and look in the category **kernel/drivers/input/mouse**.

D.2 Disk Space Needed for Tasks

A standard installation for the amd64 architecture, including all standard packages and using the default kernel, takes up 933MB of disk space. A minimal base installation, without the „Standard system" task selected, will take 701MB.

IMPORTANT

In both cases this is the actual disk space used *after* the installation is finished and any temporary files deleted. It also does not take into account overhead used by the file system, for example for journal files. This means that significantly more disk space is needed both *during* the installation and for normal system use.

The following table lists sizes reported by aptitude for the tasks listed in tasksel. Note that some tasks have overlapping constituents, so the total installed size for two tasks together may be less than the total obtained by adding up the numbers.

By default the installer will install the GNOME desktop environment, but alternative desktop environments can be selected either by using one of the special installation images, or by specifying the desired desktop environment during installation (see Secțiune 6.3.6.2).

Note that you will need to add the sizes listed in the table to the size of the standard installation when determining the size of partitions. Most of the size listed as „Installed size" will end up in `/usr` and in `/lib`; the size listed as „Download size" is (temporarily) required in `/var`.

Task	Installed size (MB)	Download size (MB)	Space needed to install (MB)
Desktop environment			
• GNOME (default)	2724	785	3509
• KDE Plasma	4244	1280	5524
• Xfce	2342	657	2999
• LXDE	2486	707	3193
• MATE	2857	757	3614
• Cinnamon	3824	1102	4926
Web server	44	11	55
Print server	440	91	531
SSH server	2	0	2

If you install in a language other than English, **tasksel** may automatically install a *localization task*, if one is available for your language. Space requirements differ per language; you should allow up to 350MB in total for download and installation.

D.3 Installing Debian GNU/Linux from a Unix/Linux System

This section explains how to install Debian GNU/Linux from an existing Unix or Linux system, without using the menu-driven installer as explained in the rest of the manual. This „cross-install" HOWTO has been requested by users switching to Debian GNU/Linux from Red Hat, Mandriva, and SUSE. In this section some familiarity with

entering *nix commands and navigating the file system is assumed. In this section, `$` symbolizes a command to be entered in the user's current system, while `#` refers to a command entered in the Debian chroot.

Once you've got the new Debian system configured to your preference, you can migrate your existing user data (if any) to it, and keep on rolling. This is therefore a „zero downtime" Debian GNU/Linux install. It's also a clever way for dealing with hardware that otherwise doesn't play friendly with various boot or installation media.

Notă

As this is a mostly manual procedure, you should bear in mind that you will need to do a lot of basic configuration of the system yourself, which will also require more knowledge of Debian and of Linux in general than performing a regular installation. You cannot expect this procedure to result in a system that is identical to a system from a regular installation. You should also keep in mind that this procedure only gives the basic steps to set up a system. Additional installation and/or configuration steps may be needed.

D.3.1 Getting Started

With your current *nix partitioning tools, repartition the hard drive as needed, creating at least one filesystem plus swap. You need around 701MB of space available for a console only install, or about 2486MB if you plan to install X (more if you intend to install desktop environments like GNOME or KDE Plasma).

Next, create file systems on the partitions. For example, to create an ext3 file system on partition `/dev/sda6` (that's our example root partition):

```
# mke2fs -j /dev/sda6
```

To create an ext2 file system instead, omit **`-j`**.

Initialize and activate swap (substitute the partition number for your intended Debian swap partition):

```
# mkswap /dev/sda5
# sync
# swapon /dev/sda5
```

Mount one partition as `/mnt/debinst` (the installation point, to be the root (`/`) filesystem on your new system). The mount point name is strictly arbitrary, it is referenced later below.

```
# mkdir /mnt/debinst
# mount /dev/sda6 /mnt/debinst
```

Notă

If you want to have parts of the filesystem (e.g. /usr) mounted on separate partitions, you will need to create and mount these directories manually before proceding with the next stage.

D.3.2 Install debootstrap

The utility used by the Debian installer, and recognized as the official way to install a Debian base system, is **debootstrap**. It uses **wget** and **ar**, but otherwise depends only on `/bin/sh` and basic Unix/Linux tools[2]. Install **wget** and **ar** if they aren't already on your current system, then download and install **debootstrap**.

Or, you can use the following procedure to install it manually. Make a work folder for extracting the .deb into:

```
# mkdir work
# cd work
```

[2] These include the GNU core utilities and commands like **sed**, **grep**, **tar** and **gzip**.

The **debootstrap** binary is located in the Debian archive (be sure to select the proper file for your architecture). Download the **debootstrap** .deb from the pool, copy the package to the work folder, and extract the files from it. You will need to have root privileges to install the files.

```
# ar -x debootstrap_0.X.X_all.deb
# cd /
# zcat /full-path-to-work/work/data.tar.gz | tar xv
```

D.3.3 Run debootstrap

debootstrap can download the needed files directly from the archive when you run it. You can substitute any Debian archive mirror for **http.us.debian.org/debian** in the command example below, preferably a mirror close to you network-wise. Mirrors are listed at http://www.debian.org/mirror/list.

If you have a bullseye Debian GNU/Linux installation image mounted at /cdrom, you could substitute a file URL instead of the http URL: **file:/cdrom/debian/**

Substitute one of the following for *ARCH* in the **debootstrap** command: **amd64**, **arm64**, **armel**, **armhf**, **i386**, **mips**, **mips64el**, **mipsel**, **powerpc**, **ppc64el**, or **s390x**.

```
# /usr/sbin/debootstrap --arch ARCH bullseye \
     /mnt/debinst http://ftp.us.debian.org/debian
```

If the target architecture is different than the host, you should add the **--foreign** option.

D.3.4 Configure The Base System

Now you've got a real Debian system, though rather lean, on disk. **chroot** into it:

```
# LANG=C.UTF-8 chroot /mnt/debinst /bin/bash
```

If the target architecture is different from the host, you will need to first copy qemu-user-static to the new host:

```
# cp /usr/bin/qemu-ARCH-static /mnt/debinst/usr/bin
# LANG=C.UTF-8 chroot /mnt/debinst qemu-ARCH-static /bin/bash
```

After chrooting you may need to set the terminal definition to be compatible with the Debian base system, for example:

```
# export TERM=xterm-color
```

Depending on the value of TERM, you may have to install the ncurses-term package to get support for it.

If the target architecture is different from the host, you need to finish the multi-stage boot strap:

```
/debootstrap/debootstrap --second-stage
```

D.3.4.1 Create device files

At this point /dev/ only contains very basic device files. For the next steps of the installation additional device files may be needed. There are different ways to go about this and which method you should use depends on the host system you are using for the installation, on whether you intend to use a modular kernel or not, and on whether you intend to use dynamic (e.g. using udev) or static device files for the new system.

A few of the available options are:

- install the makedev package, and create a default set of static device files using (after chrooting)

  ```
  # apt install makedev
  # mount none /proc -t proc
  # cd /dev
  # MAKEDEV generic
  ```

- manually create only specific device files using **MAKEDEV**
- bind mount /dev from your host system on top of /dev in the target system; note that the postinst scripts of some packages may try to create device files, so this option should only be used with care

D.3.4.2 Mount Partitions

You need to create `/etc/fstab`.

```
# editor /etc/fstab
```

Here is a sample you can modify to suit:

```
# /etc/fstab: static file system information.
#
# file system    mount point   type    options                  dump pass
/dev/XXX         /             ext3    defaults                 0    1
/dev/XXX         /boot         ext3    ro,nosuid,nodev          0    2

/dev/XXX         none          swap    sw                       0    0
proc             /proc         proc    defaults                 0    0

/dev/cdrom       /media/cdrom  iso9660 noauto,ro,user,exec      0    0

/dev/XXX         /tmp          ext3    rw,nosuid,nodev          0    2
/dev/XXX         /var          ext3    rw,nosuid,nodev          0    2
/dev/XXX         /usr          ext3    rw,nodev                 0    2
/dev/XXX         /home         ext3    rw,nosuid,nodev          0    2
```

Use **`mount -a`** to mount all the file systems you have specified in your `/etc/fstab`, or, to mount file systems individually, use:

```
# mount /path   # e.g.: mount /usr
```

Current Debian systems have mountpoints for removable media under `/media`, but keep compatibility symlinks in `/`. Create these as as needed, for example:

```
# cd /media
# mkdir cdrom0
# ln -s cdrom0 cdrom
# cd /
# ln -s media/cdrom
```

You can mount the proc file system multiple times and to arbitrary locations, though `/proc` is customary. If you didn't use **`mount -a`**, be sure to mount proc before continuing:

```
# mount -t proc proc /proc
```

The command **`ls /proc`** should now show a non-empty directory. Should this fail, you may be able to mount proc from outside the chroot:

```
# mount -t proc proc /mnt/debinst/proc
```

D.3.4.3 Setting Timezone

Setting the third line of the file `/etc/adjtime` to „UTC" or „LOCAL" determines whether the system will interpret the hardware clock as being set to UTC respective local time. The following command allows you to set that.

```
# editor /etc/adjtime
```

Here is a sample:

```
0.0 0 0.0
0
UTC
```

The following command allows you to choose your timezone.

```
# dpkg-reconfigure tzdata
```

D.3.4.4 Configure Networking

To configure networking, edit `/etc/network/interfaces`, `/etc/resolv.conf`, `/etc/hostname` and `/etc/hosts`.

```
# editor /etc/network/interfaces
```

Here are some simple examples from `/usr/share/doc/ifupdown/examples`:

```
######################################################################
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)
# See the interfaces(5) manpage for information on what options are
# available.
######################################################################

# The loopback interface isn't really required any longer, but can be used
# if needed.
#
# auto lo
# iface lo inet loopback

# To use dhcp:
#
# auto eth0
# iface eth0 inet dhcp

# An example static IP setup: (network, broadcast and gateway are optional)
#
# auto eth0
# iface eth0 inet static
#     address 192.168.0.42
#     network 192.168.0.0
#     netmask 255.255.255.0
#     broadcast 192.168.0.255
#     gateway 192.168.0.1
```

Enter your nameserver(s) and search directives in `/etc/resolv.conf`:

```
# editor /etc/resolv.conf
```

A simple example `/etc/resolv.conf`:

```
search example.com
nameserver 10.1.1.36
nameserver 192.168.9.100
```

Enter your system's host name (2 to 63 characters):

```
# echo DebianHostName > /etc/hostname
```

And a basic `/etc/hosts` with IPv6 support:

```
127.0.0.1 localhost
127.0.1.1 DebianHostName

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

If you have multiple network cards, you should arrange the names of driver modules in the `/etc/modules` file into the desired order. Then during boot, each card will be associated with the interface name (eth0, eth1, etc.) that you expect.

D.3.4.5 Configure Apt

Debootstrap will have created a very basic `/etc/apt/sources.list` that will allow installing additional packages. However, you may want to add some additional sources, for example for source packages and security updates:

```
deb-src http://ftp.us.debian.org/debian bullseye main

deb http://security.debian.org/ bullseye/updates main
deb-src http://security.debian.org/ bullseye/updates main
```

Make sure to run **`apt update`** after you have made changes to the sources list.

D.3.4.6 Configure Locales and Keyboard

To configure your locale settings to use a language other than English, install the `locales` support package and configure it. Currently the use of UTF-8 locales is recommended.

```
# apt install locales
# dpkg-reconfigure locales
```

To configure your keyboard (if needed):

```
# apt install console-setup
# dpkg-reconfigure keyboard-configuration
```

Note that the keyboard cannot be set while in the chroot, but will be configured for the next reboot.

D.3.5 Install a Kernel

If you intend to boot this system, you probably want a Linux kernel and a boot loader. Identify available pre-packaged kernels with:

```
# apt search linux-image
```

Then install the kernel package of your choice using its package name.

```
# apt install linux-image-arch-etc
```

D.3.6 Set up the Boot Loader

To make your Debian GNU/Linux system bootable, set up your boot loader to load the installed kernel with your new root partition. Note that **debootstrap** does not install a boot loader, but you can use **apt** inside your Debian chroot to do so.

Check **`info grub`** or **`man lilo.conf`** for instructions on setting up the bootloader. If you are keeping the system you used to install Debian, just add an entry for the Debian install to your existing grub2 `grub.cfg` or `lilo.conf`. For `lilo.conf`, you could also copy it to the new system and edit it there. After you are done editing, call **lilo** (remember it will use `lilo.conf` relative to the system you call it from).

Installing and setting up `grub2` is as easy as:

```
# apt install grub-pc
# grub-install /dev/sda
# update-grub
```

The second command will install **grub2** (in this case in the MBR of `sda`). The last command will create a sane and working `/boot/grub/grub.cfg`.

Note that this assumes that a `/dev/sda` device file has been created. There are alternative methods to install **grub2**, but those are outside the scope of this appendix.

Here is a basic `/etc/lilo.conf` as an example:

```
boot=/dev/sda6
root=/dev/sda6
install=menu
delay=20
lba32
image=/vmlinuz
```

```
initrd=/initrd.img
label=Debian
```

D.3.7 Remote access: Installing SSH and setting up access

In case you can login to the system via console, you can skip this section. If the system should be accessible via the network later on, you need to install SSH and set up access.

```
# apt install ssh
```

Root login with password is disabled by default, so setting up access can be done by setting a password and re-enable root login with password:

```
# passwd
# editor /etc/ssh/sshd_config
```

This is the option to be enabled:

```
PermitRootLogin yes
```

Access can also be set up by adding an ssh key to the root account:

```
# mkdir /root/.ssh
# cat << EOF > /root/.ssh/authorized_keys
ssh-rsa ....
EOF
```

Lastly, access can be set up by adding a non-root user and setting a password:

```
# adduser joe
# passwd joe
```

D.3.8 Finishing touches

As mentioned earlier, the installed system will be very basic. If you would like to make the system a bit more mature, there is an easy method to install all packages with „standard" priority:

```
# tasksel install standard
```

Of course, you can also just use **apt** to install packages individually.

After the installation there will be a lot of downloaded packages in `/var/cache/apt/archives/`. You can free up some diskspace by running:

```
# apt clean
```

D.4 Installing Debian GNU/Linux over Parallel Line IP (PLIP)

This section explains how to install Debian GNU/Linux on a computer without an Ethernet card, but with just a remote gateway computer attached via a Null-Modem cable (also called Null-Printer cable). The gateway computer should be connected to a network that has a Debian mirror on it (e.g. to the Internet).

In the example in this appendix we will set up a PLIP connection using a gateway connected to the Internet over a dial-up connection (ppp0). We will use IP addresses 192.168.0.1 and 192.168.0.2 for the PLIP interfaces on the target system and the source system respectively (these addresses should be unused within your network address space).

The PLIP connection set up during the installation will also be available after the reboot into the installed system (see Cap. 7).

Before you start, you will need to check the BIOS configuration (IO base address and IRQ) for the parallel ports of both the source and target systems. The most common values are `io=0x378`, `irq=7`.

D.4.1 Requirements

- A target computer, called *target*, where Debian will be installed.
- System installation media; see Secțiune 2.4.
- Another computer connected to the Internet, called *source*, that will function as the gateway.
- A DB-25 Null-Modem cable. See the PLIP-Install-HOWTO for more information on this cable and instructions how to make your own.

D.4.2 Setting up source

The following shell script is a simple example of how to configure the source computer as a gateway to the Internet using ppp0.

```
#!/bin/sh

# We remove running modules from kernel to avoid conflicts and to
# reconfigure them manually.
modprobe -r lp parport_pc
modprobe parport_pc io=0x378 irq=7
modprobe plip

# Configure the plip interface (plip0 for me, see dmesg | grep plip)
ifconfig plip0 192.168.0.2 pointopoint 192.168.0.1 netmask 255.255.255.255 up

# Configure gateway
modprobe iptable_nat
iptables -t nat -A POSTROUTING -o ppp0 -j MASQUERADE
echo 1 > /proc/sys/net/ipv4/ip_forward
```

D.4.3 Installing target

Boot the installation media. The installation needs to be run in expert mode; enter **`expert`** at the boot prompt. If you need to set parameters for kernel modules, you also need to do this at the boot prompt. For example, to boot the installer and set values for the „io” and „irq” options for the parport_pc module, enter the following at the boot prompt:

```
expert parport_pc.io=0x378 parport_pc.irq=7
```

Below are the answers that should be given during various stages of the installation.

1. Load installer components from installation media

   Select the **`plip-modules`** option from the list; this will make the PLIP drivers available to the installation system.

2. Detect network hardware

   - If target *does* have a network card, a list of driver modules for detected cards will be shown. If you want to force `debian-installer` to use plip instead, you have to deselect all listed driver modules. Obviously, if target doesn't have a network card, the installer will not show this list.
   - Because no network card was detected/selected earlier, the installer will ask you to select a network driver module from a list. Select the **`plip`** module.

3. Configure the network

   - Auto-configure network with DHCP: No
   - IP address: ***`192.168.0.1`***
   - Point-to-point address: ***`192.168.0.2`***
   - Name server addresses: you can enter the same addresses used on source (see `/etc/resolv.conf`)

D.5 Installing Debian GNU/Linux using PPP over Ethernet (PPPoE)

In some countries PPP over Ethernet (PPPoE) is a common protocol for broadband (ADSL or cable) connections to an Internet Service Provider. Setting up a network connection using PPPoE is not supported by default in the installer, but can be made to work very simply. This section explains how.

The PPPoE connection set up during the installation will also be available after the reboot into the installed system (see Cap. 7).

To have the option of setting up and using PPPoE during the installation, you will need to install using one of the CD-ROM/DVD images that are available. It is not supported for other installation methods (e.g. netboot).

Installing over PPPoE is mostly the same as any other installation. The following steps explain the differences.

- Boot the installer with the boot parameter **`modules=ppp-udeb`**[3]. This will ensure the component responsible for the setup of PPPoE (`ppp-udeb`) will be loaded and run automatically.
- Follow the regular initial steps of the installation (language, country and keyboard selection; the loading of additional installer components[4]).
- The next step is the detection of network hardware, in order to identify any Ethernet cards present in the system.
- After this the actual setup of PPPoE is started. The installer will probe all the detected Ethernet interfaces in an attempt to find a PPPoE concentrator (a type of server which handles PPPoE connections).

  It is possible that the concentrator will not to be found at the first attempt. This can happen occasionally on slow or loaded networks or with faulty servers. In most cases a second attempt to detect the concentrator will be successful; to retry, select Configure and start a PPPoE connection from the main menu of the installer.
- After a concentrator is found, the user will be prompted to type the login information (the PPPoE username and password).
- At this point the installer will use the provided information to establish the PPPoE connection. If the correct information was provided, the PPPoE connection should be configured and the installer should be able to use it to connect to the Internet and retrieve packages over it (if needed). If the login information is not correct or some error appears, the installer will stop, but the configuration can be attempted again by selecting the menu entry Configure and start a PPPoE connection.

[3] See Secțiune 5.1.7 for information on how to add a boot parameter.

[4] The `ppp-udeb` component is loaded as one of the additional components in this step. If you want to install at medium or low priority (expert mode), you can also manually select the `ppp-udeb` instead of entering the „modules” parameter at the boot prompt.

Anexa E

Administrivia

E.1 About This Document

This manual was created for Sarge's debian-installer, based on the Woody installation manual for boot-floppies, which was based on earlier Debian installation manuals, and on the Progeny distribution manual which was released under GPL in 2003.

This document is written in DocBook XML. Output formats are generated by various programs using information from the `docbook-xml` and `docbook-xsl` packages.

In order to increase the maintainability of this document, we use a number of XML features, such as entities and profiling attributes. These play a role akin to variables and conditionals in programming languages. The XML source to this document contains information for each different architecture — profiling attributes are used to isolate certain bits of text as architecture-specific.

E.2 Contributing to This Document

If you have problems or suggestions regarding this document, you should probably submit them as a bug report against the package `installation-guide`. See the `reportbug` package or read the online documentation of the Debian Bug Tracking System. It would be nice if you could check the open bugs against installation-guide to see whether your problem has already been reported. If so, you can supply additional corroboration or helpful information to *xxxx*@bugs.debian.org, where *xxxx* is the number for the already-reported bug.

Better yet, get a copy of the DocBook source for this document, and produce patches against it. The DocBook source can be found at the installation-guide project on salsa. If you're not familiar with DocBook, don't worry: there is a simple cheatsheet in the manuals directory that will get you started. It's like html, but oriented towards the meaning of the text rather than the presentation. Patches submitted to the debian-boot mailing list (see below) are welcomed. For instructions on how to check out the sources via git, see README from the source root directory.

Please do *not* contact the authors of this document directly. There is also a discussion list for `debian-installer`, which includes discussions of this manual. The mailing list is debian-boot@lists.debian.org. Instructions for subscribing to this list can be found at the Debian Mailing List Subscription page; or you can browse the Debian Mailing List Archives online.

E.3 Major Contributions

This document was originally written by Bruce Perens, Sven Rudolph, Igor Grobman, James Treacy, and Adam Di Carlo. Sebastian Ley wrote the Installation Howto.

Miroslav Kuře has documented a lot of the new functionality in Sarge's debian-installer. Frans Pop was the main editor and release manager during the Etch, Lenny and Squeeze releases.

Many, many Debian users and developers contributed to this document. Particular note must be made of Michael Schmitz (m68k support), Frank Neumann (original author of the Amiga install manual), Arto Astala, Eric Delaunay/Ben Collins (SPARC information), Tapio Lehtonen, and Stéphane Bortzmeyer for numerous edits and text. We have to thank Pascal Le Bail for useful information about booting from USB memory sticks.

Extremely helpful text and information was found in Jim Mintha's HOWTO for network booting (no URL available), the Debian FAQ, the Linux/m68k FAQ, the Linux for SPARC Processors FAQ, the Linux/Alpha FAQ, amongst others. The maintainers of these freely available and rich sources of information must be recognized.

The section on chrooted installations in this manual (Secțiune D.3) was derived in part from documents copyright Karsten M. Self.

The section on installations over plip in this manual (Secțiune D.4) was based on the PLIP-Install-HOWTO by Gilles Lamiral.

E.4 Trademark Acknowledgement

All trademarks are property of their respective trademark owners.

Anexa F

GNU General Public License

Version 2, June 1991

```
Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA.

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
```

F.1 Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the gnu General Public License is intended to guarantee your freedom to share and change free software — to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the gnu Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

F.2 GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The „Program", below, refers to any such program or work, and a „work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or

translated into another language. (Hereinafter, translation is included without limitation in the term „modification".) Each licensee is addressed as „you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a. You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b. You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c. If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a. Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b. Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c. Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and „any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM „AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL AND COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE

THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

F.3 How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the „copyright" line and a pointer to where the full notice is found.

```
one line to give the program's name and a brief idea of what it does.
Copyright (C) year  name of author

This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA  02110-1301, USA.
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with absolutely no warranty; for details
type show w. This is free software, and you are welcome
to redistribute it under certain conditions; type show c
for details.
```

The hypothetical commands „show w" and „show c" should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than „show w" and „show c"; they could even be mouse-clicks or menu items — whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a „copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program Gnomovision (which makes passes at compilers) written
by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.