

debian

Ghidul de referință Debian

Osamu Aoki

Copyright © 2013-2024 Osamu Aoki

Acest Ghid de referință Debian (versiunea 2.152) (2026-10-10 14:56:25 UTC) are scopul de a oferi o imagine de ansamblu asupra sistemului Debian, ca ghid de utilizare post-instalare. Acesta acoperă multe aspecte ale administrării sistemului prin exemple de comenzi shell pentru utilizatorii care nu sunt dezvoltatori.

Cuprins

1 Tutoriale GNU/Linux **1**

1.1 Noțiuni de bază despre consolă . . . 1

1.1.1 Promptul shell . . . 1

1.1.2 Promptul shell în GUI . . . 2

1.1.3 Contul root . . . 2

1.1.4 Promptul shell-ului root . . . 3

1.1.5 Instrumente grafice de administrare a sistemului . . . 3

1.1.6 Console virtuale . . . 3

1.1.7 Cum să ieșiți din promptul liniei de comandă . . . 4

1.1.8 Cum să opriți sistemul . . . 4

1.1.9 Recuperarea unei console funcționale . . . 4

1.1.10 Sugestii de pachete suplimentare pentru începători . . . 4

1.1.11 Un cont de utilizator suplimentar . . . 5

1.1.12 Configurarea «sudo» . . . 5

1.1.13 Ora de joacă . . . 6

1.2 Sistem de fișiere de tip Unix . . . 6

1.2.1 Noțiuni de bază despre fișierele Unix . . . 7

1.2.2 Elementele interne ale sistemului de fișiere . . . 8

1.2.3 Permisiuni ale sistemului de fișiere . . . 8

1.2.4 Controlul permisiunilor pentru fișierele nou create: umask . . . 11

1.2.5 Permisiuni pentru grupuri de utilizatori (grup) . . . 11

1.2.6 Marcaje de timp . . . 13

1.2.7 Legături . . . 14

1.2.8 Conducte cu nume (FIFO) . . . 15

1.2.9 Socluri . . . 15

1.2.10 Fișiere de dispozitive . . . 16

1.2.11 Fișiere de dispozitive speciale . . . 16

1.2.12 procfs și sysfs . . . 17

1.2.13 tmpfs . . . 17

1.3 Midnight Commander (MC) . . . 18

1.3.1 Personalizarea MC . . . 18
1.3.2 Lansarea MC . . . 18
1.3.3 Gestionarul de fișiere din MC . . . 18
1.3.4 Trucuri din linia de comandă în MC . . . 19
1.3.5 Editorul intern din MC . . . 19
1.3.6 Vizorul intern din MC . . . 20
1.3.7 Funcții de pornire automată ale MC . . . 20
1.3.8 Sistemul de fișiere virtual al MC . . . 20
1.4 Mediul de lucru de bază de tip Unix . . . 20
1.4.1 Shell-ul de autentificare . . . 20
1.4.2 Personalizarea bash . . . 21
1.4.3 Combinații speciale de taste . . . 22
1.4.4 Operații cu mouse-ul . . . 22
1.4.5 Paginatorul . . . 23
1.4.6 Editorul de text . . . 23
1.4.7 Definirea unui editor de text implicit . . . 24
1.4.8 Utilizarea editorului vim . . . 24
1.4.9 Înregistrarea activităților shell-ului . . . 25
1.4.10 Comenzi Unix de bază . . . 25
1.5 Comanda simplă de shell . . . 26
1.5.1 Executarea comenzilor și variabilele de mediu . . . 28
1.5.2 Variabila „$LANG” . . . 28
1.5.3 Variabila „$PATH” . . . 29
1.5.4 Variabila „$HOME” . . . 29
1.5.5 Opțiuni în linia de comandă . . . 30
1.5.6 Facilitatea glob a shell-ului . . . 30
1.5.7 Valoarea returnată de comandă . . . 31
1.5.8 Secvențe tipice de comenzi și redirecționarea shell-ului . . . 31
1.5.9 Alias de comandă . . . 33
1.6 Procesarea textului în stilul Unix . . . 34
1.6.1 Instrumente pentru text în Unix . . . 34
1.6.2 Expresii regulate . . . 35
1.6.3 Expresii de înlocuire . . . 36
1.6.4 Înlocuire globală cu expresii regulate . . . 37
1.6.5 Extragerea datelor din tabelul fișierului text . . . 37
1.6.6 Fragmente de script pentru comenzi de direcționare . . . 39

2 Gestionarea pachetelor Debian 40

2.1 Cerințe preliminare pentru gestionarea pachetelor Debian . . . 40
2.1.1 Sistemul de gestionare a pachetelor Debian . . . 40
2.1.2 Configurația pachetului . . . 41
2.1.3 Precauții de bază . . . 41
2.1.4 O viață cu îmbunătățiri veșnice . . . 42
2.1.5 Noțiuni de bază despre arhiva Debian . . . 43
2.1.6 Debian este software 100% liber . . . 47
2.1.7 Dependențele pachetelor . . . 48
2.1.8 Fluxul evenimentelor din gestionarea pachetelor . . . 49
2.1.9 Primul răspuns la problemele legate de gestionarea pachetelor . . . 50
2.1.10 Cum să selectați pachetele Debian . . . 51
2.1.11 Cum să faceți față cerințelor contradictorii . . . 51
2.2 Operații de bază de gestionare a pachetelor . . . 52
2.2.1 `apt` vs. `apt-get` / `apt-cache` vs. `aptitude` . . . 52
2.2.2 Operații de bază de gestionare a pachetelor din linia de comandă . . . 53
2.2.3 Utilizarea interactivă a «aptitudie» . . . 53
2.2.4 Combinații de taste pentru aptitude . . . 55
2.2.5 Vizualizarea de pachete în aptitude . . . 55
2.2.6 Opțiuni pentru metoda de căutare cu aptitude . . . 57
2.2.7 Formula expresiei regulate pentru aptitudine . . . 57
2.2.8 Rezolvarea dependențelor de către aptitude . . . 59
2.2.9 Jurnale de activitate ale pachetelor . . . 59
2.3 Exemple de operații cu aptitudine . . . 59
2.3.1 Căutarea de pachete interesante . . . 59
2.3.2 Listarea pachetelor care se potrivesc cu expresia regulată în numele pachetelor . . . 59
2.3.3 Navigarea prin rezultatul potrivirilor expresiei regulate . . . 59
2.3.4 Stergerea definitivă a pachetelor eliminate . . . 60
2.3.5 Reorganizarea stării instalării automate/manuale . . . 60
2.3.6 Actualizare la nivel de sistem . . . 61
2.4 Operații avansate de gestionare a pachetelor . . . 62
2.4.1 Operații avansate de gestionare a pachetelor din linia de comandă . . . 62
2.4.2 Verificarea fișierelor pachetului instalat . . . 64
2.4.3 Protejarea împotriva problemelor legate de pachete . . . 64
2.4.4 Căutarea în metadatele pachetului . . . 64
2.5 Detalii interne privind gestionarea pachetelor Debian . . . 64
2.5.1 Metadate de arhivă . . . 64
2.5.2 Fișierul „Release” de nivel superior și autenticitatea . . . 65
2.5.3 Fișiere „Release” la nivel de arhivă . . . 66

2.5.4 Preluarea metadatelor pentru pachet . . . 67
2.5.5 Starea pachetului pentru APT . . . 67
2.5.6 Starea pachetului pentru aptitude . . . 67
2.5.7 Copiile locale ale pachetelor descărcate . . . 67
2.5.8 Numele fișierelor pachetelor Debian . . . 68
2.5.9 Comanda «dpkg» . . . 68
2.5.10 Comanda «update-alternatives» . . . 69
2.5.11 Comanda «dpkg-statoverride» . . . 70
2.5.12 Comanda «dpkg-divert» . . . 70
2.6 Recuperarea dintr-un sistem defect . . . 70
2.6.1 Incompatibilitate cu configurația veche a utilizatorului . . . 70
2.6.2 Erori de stocare în cache ale datelor pachetului . . . 71
2.6.3 Recuperarea cu comanda dpkg . . . 71
2.6.4 Instalare eșuată din cauza dependențelor lipsă . . . 71
2.6.5 Pachete diferite cu fișiere suprapuse . . . 72
2.6.6 Remedierea scriptului pachetului defect . . . 72
2.6.7 Recuperarea datelor privind selecția pachetelor . . . 72
2.7 Sfaturi pentru gestionarea pachetelor . . . 73
2.7.1 Cine a încărcat pachetul? . . . 73
2.7.2 Limitarea lățimii de bandă pentru descărcare pentru APT . . . 73
2.7.3 Descărcarea și actualizarea automată a pachetelor . . . 73
2.7.4 Actualizări și retro-adaptări (versiuni de software migrate din ramura principală de dezvoltare și adaptate pentru a funcționa cu această versiune) . . . 74
2.7.5 Arhive de pachete externe . . . 74
2.7.6 Pachete din surse mixte de arhive fără **apt-pinning** . . . 75
2.7.7 Ajustarea versiunii candidate cu **apt-pinning** . . . 76
2.7.8 Blocarea pachetelor instalate de „Recommends” . . . 77
2.7.9 Urmărirea suitei `testing` cu unele pachete din `unstable` . . . 77
2.7.10 Urmărirea suitei `unstable` cu unele pachete din `experimental` . . . 79
2.7.11 Retrogradarea de urgență . . . 79
2.7.12 Pachetul equivs . . . 80
2.7.13 Adaptarea unui pachet la sistemul stabil . . . 81
2.7.14 Server proxy pentru APT . . . 81
2.7.15 Mai multe informații despre gestionarea pachetelor . . . 82

3 Inițializarea sistemului **83**
3.1 O prezentare generală a procesului de inițializare . . . 83
3.1.1 Etapa 1: UEFI . . . 84
3.1.2 Etapa 2: încărcătorul de pornire . . . 84
3.1.3 Etapa 3: sistemul mini-Debian . . . 85
3.1.4 Etapa 4: sistemul Debian normal . . . 86
3.2 Sistemul de recuperare . . . 87
3.2.1 Sistem de recuperare GRUB UEFI pe USB . . . 88
3.2.2 Sistem de recuperare Linux live pe USB . . . 88
3.2.3 Sistemul de recuperare live Linux din GRUB . . . 89
3.3 Systemd . . . 89
3.3.1 Init systemd . . . 89
3.3.2 Autentificarea cu systemd . . . 91
3.4 Mesajele nucleului . . . 91
3.5 Mesajele sistemului . . . 91
3.6 Gestionarea sistemului . . . 92
3.7 Alte monitoare de sistem . . . 92
3.8 Configurația sistemului . . . 92
3.8.1 Numele gazdei . . . 92
3.8.2 Sistemul de fișiere . . . 92
3.8.3 Inițializarea interfeței de rețea . . . 94
3.8.4 Inițializarea sistemului cloud . . . 94
3.8.5 Exemplu de personalizare pentru ajustarea serviciului sshd . . . 94
3.9 Sistemul udev . . . 95
3.10 Inițializarea modulelor de nucleu . . . 96

4 Autentificare și controale de acces **97**
4.1 Autentificare normală Unix . . . 97
4.2 Gestionarea informațiilor privind contul și parola . . . 99
4.3 Parolă bună . . . 99
4.4 Crearea unei parole criptate . . . 100
4.5 PAM și NSS . . . 100
4.5.1 Fișierele de configurare accesate de PAM și NSS . . . 101
4.5.2 Sistemul modern de gestionare centralizată . . . 102
4.5.3 „De ce comanda «su» a GNU nu acceptă grupul wheel” . . . 102
4.5.4 Reguli mai stricte privind parolele . . . 103
4.6 Securitatea autentificării . . . 103
4.6.1 Parolă sigură în Internet . . . 103
4.6.2 Shell securizat . . . 104

4.6.3 Măsuri suplimentare de securitate pentru Internet . . . 104
4.6.4 Securizarea parolei root . . . 104
4.7 Alte controale de acces . . . 105
4.7.1 Liste de control al accesului (ACL) . . . 105
4.7.2 sudo . . . 106
4.7.3 PolicyKit . . . 106
4.7.4 Restricționarea accesului la anumite servicii ale serverului . . . 106
4.7.5 Caracteristici de securitate Linux . . . 107

5 Configurarea rețelei 109
5.1 Infrastructura de bază a rețelei . . . 109
5.1.1 Rezoluția numelui de gazdă . . . 109
5.1.2 Numele interfeței de rețea . . . 111
5.1.3 Intervalul de adrese pentru rețeaua locală (LAN) . . . 112
5.1.4 Suportul pentru dispozitivele de rețea . . . 112
5.2 Configurația modernă a rețelei pentru mediul de birou . . . 112
5.2.1 Instrumente cu interfață grafică pentru configurarea rețelei . . . 113
5.3 Configurația modernă a rețelei fără interfața grafică . . . 113
5.4 Configurația modernă a rețelei pentru cloud . . . 114
5.4.1 Configurația modernă a rețelei pentru cloud cu DHCP . . . 114
5.4.2 Configurația modernă a rețelei pentru cloud cu adresă IP statică . . . 114
5.4.3 Configurația modernă a rețelei pentru cloud cu Network Manager . . . 114
5.5 Configurația rețelei de nivel inferior . . . 115
5.5.1 Comenzi «iproute2» . . . 115
5.5.2 Operații de rețea sigure la nivel scăzut . . . 115
5.6 Optimizarea rețelei . . . 116
5.6.1 Găsirea MTU optime . . . 116
5.6.2 Optimizare WAN TCP . . . 117
5.7 Infrastructura netfilter . . . 117

6 Aplicații de rețea 120
6.1 Navigatoare Web . . . 120
6.1.1 Falsificarea șirului User-Agent . . . 120
6.1.2 Extensie pentru navigator . . . 121
6.2 Sistemul de poștă electronică . . . 121
6.2.1 Noțiuni de bază despre poșta electronică . . . 121
6.2.2 Limita serviciilor poștale moderne . . . 122
6.2.3 Așteptări istorice privind serviciul poștal . . . 123
6.2.4 Agentul de transport al poștei electronice („Mail transport agent”: MTA) . . . 123

6.2.4.1 Configurarea exim4 . . . 123
6.2.4.2 Configurarea postfix cu SASL . . . 125
6.2.4.3 Configurarea adresei de poștă electronică . . . 126
6.2.4.4 Operații MTA de bază . . . 127
6.3 Serverul și instrumentele de acces la distanță (SSH) . . . 128
6.3.1 Noțiuni de bază despre SSH . . . 128
6.3.2 Numele de utilizator pe gazda la distanță . . . 129
6.3.3 Connectarea fără parole la distanță . . . 129
6.3.4 Gestionarea clienților SSH străini . . . 129
6.3.5 Configurarea ssh-agent . . . 130
6.3.6 Trimiterea unui mesaj de la o gazdă de la distanță . . . 130
6.3.7 Redirecționarea porturilor pentru tunelarea SMTP/POP3 . . . 130
6.3.8 Cum să opriți sistemul de la distanță pe SSH . . . 130
6.3.9 Soluționarea problemelor SSH . . . 131
6.4 Serverul de imprimare și utilitățile . . . 131
6.5 Alte servere de aplicații de rețea . . . 131
6.6 Alți clienți de aplicații de rețea . . . 132
6.7 Diagnosticul demonilor sistemului . . . 132

7 Sistemul de interfață grafică 134
7.1 Mediul de birou cu interfață grafică . . . 134
7.2 Protocolul de comunicare al interfeței grafice . . . 135
7.3 Infrastructura de interfață grafică . . . 136
7.4 Aplicații cu interfață grafică . . . 137
7.5 Directoarele utilizatorilor . . . 137
7.6 Tipuri de litere . . . 137
7.6.1 Fonturi de bază . . . 137
7.6.2 Conversia și redarea fonturilor . . . 140
7.7 Cutia cu nisip (sandbox) . . . 140
7.8 Mediu de birou la distanță . . . 142
7.9 Conexiune la serverul X . . . 142
7.9.1 Conexiune locală la serverul X . . . 142
7.9.2 Conexiune de la distanță la serverul X . . . 143
7.9.3 Conexiune chroot la serverul X . . . 143
7.10 Clipboard . . . 143

8 I18N și L10N 145

8.1 Configurația regională . . . 145
8.1.1 Justificarea pentru utilizearea UTF-8 în configurația lingvistică . . . 145
8.1.2 Reconfigurarea configurației regionale . . . 146
8.1.3 Codificarea numelor de fișiere . . . 147
8.1.4 Mesaje în limba maternă și documentație tradusă . . . 147
8.1.5 Efectele configurației regionale . . . 148
8.2 Introducere de la tastatură . . . 148
8.2.1 Introducerea de la tastatură pentru consola Linux și X Window . . . 149
8.2.2 Introducerea de la tastatură pentru Wayland . . . 149
8.2.3 Suportul pentru metoda de introducere cu IBus . . . 149
8.2.4 Suportul pentru metoda de introducere cu Fcitx . . . 151
8.2.5 Un exemplu pentru japoneză . . . 151
8.3 Afișarea ieșirii . . . 152
8.3.1 Configurația terminalului . . . 152
8.3.2 Caractere cu lățime de aspect ambiguu din Asia de Est . . . 153

9 Sfaturi privind sistemul 154

9.1 Sfaturi pentru consolă . . . 154
9.1.1 Înregistrarea activităților shell în mod curat . . . 154
9.1.2 Programul «screen» . . . 155
9.1.3 Navigarea între directoare . . . 156
9.1.4 Readline wrapper (învăluitorul «readline») . . . 156
9.1.5 Scanarea arborelui codului sursă . . . 156
9.2 Personalizarea vim . . . 157
9.2.1 Personalizarea vim cu caracteristicile interne . . . 157
9.2.2 Personalizarea vim cu pachete externe . . . 159
9.3 Înregistrarea și prezentarea datelor . . . 160
9.3.1 Demonul de jurnalizare . . . 160
9.3.2 Analizator de jurnale . . . 160
9.3.3 Afișarea personalizată a datelor textuale . . . 161
9.3.4 Afișare personalizată a orei și datei . . . 161
9.3.5 Ecou colorat al shell-ului . . . 162
9.3.6 Comenzi colorate . . . 162
9.3.7 Înregistrarea activităților editorului pentru repetări complexe . . . 163
9.3.8 Înregistrarea imaginii grafice a unei aplicații X . . . 163
9.3.9 Înregistrarea modificărilor din fișierele de configurare . . . 163
9.4 Monitorizarea, controlul și inițierea activităților programului . . . 164
9.4.1 Cronometrarea unui proces . . . 164

9.4.2 Prioritatea de planificare . . . 164
9.4.3 Comanda «ps» . . . 165
9.4.4 Comanda «top» . . . 165
9.4.5 Listarea fișierelor deschise de un proces . . . 165
9.4.6 Urmărirea activităților programului . . . 165
9.4.7 Identificarea proceselor care utilizează fișiere sau socluri . . . 166
9.4.8 Repetarea unei comenzi la intervale constante . . . 166
9.4.9 Repetarea unei comenzi care parcurge fișierele . . . 166
9.4.10 Pornirea unui program din interfața grafică . . . 167
9.4.11 Personalizarea programului care urmează să fie pornit . . . 168
9.4.12 Omorârea unui proces . . . 169
9.4.13 Programarea sarcinilor o singură dată . . . 169
9.4.14 Programarea regulată a sarcinilor . . . 169
9.4.15 Programarea sarcinilor la eveniment . . . 171
9.4.16 Tasta Alt-SysRq . . . 171
9.5 Sfaturi pentru întreținerea sistemului . . . 172
9.5.1 Cine este în sistem? . . . 172
9.5.2 Avertisment pentru toată lumea . . . 172
9.5.3 Identificarea hardware-ului . . . 172
9.5.4 Configurația hardware . . . 172
9.5.5 Ora sistemului și ora hardware-ului . . . 174
9.5.6 Infrastructura de sunet . . . 174
9.5.7 Dezactivarea protectorului de ecran . . . 175
9.5.8 Dezactivarea bipurilor sonore . . . 175
9.5.9 Utilizare memorie . . . 175
9.5.10 Verificarea securității și integrității sistemului . . . 176
9.6 Sfaturi pentru stocarea datelor . . . 177
9.6.1 Utilizarea spațiului pe disc . . . 177
9.6.2 Configurarea partițiilor de disc . . . 177
9.6.3 Accesarea partiției folosind UUID . . . 178
9.6.4 LVM2 . . . 178
9.6.5 Configurarea sistemului de fișiere . . . 179
9.6.6 Crearea sistemului de fișiere și verificarea integrității . . . 179
9.6.7 Optimizarea sistemului de fișiere prin opțiuni de montare . . . 181
9.6.8 Optimizarea sistemului de fișiere prin super-bloc . . . 181
9.6.9 Optimizarea discului dur . . . 181
9.6.10 Optimizarea discului cu stare solidă . . . 182
9.6.11 Utilizarea SMART pentru a prezice defectarea discului dur . . . 182
9.6.12 Specificați directorul de stocare temporară prin $TMPDIR . . . 182

9.6.13 Extinderea spațiului de stocare utilizabil prin LVM . . . 182
9.6.14 Extinderea spațiului de stocare utilizabil prin montarea unei alte partiții . . . 183
9.6.15 Extinderea spațiului de stocare utilizabil prin montarea unui alt director . . . 183
9.6.16 Extinderea spațiului de stocare utilizabil prin montarea suprapusă a unui alt director . . . 183
9.6.17 Extinderea spațiului de stocare utilizabil folosind legături simbolice . . . 183
9.7 Imaginea discului . . . 184
9.7.1 Crearea fișierului imagine de disc . . . 184
9.7.2 Scrierea direct pe disc . . . 184
9.7.3 Montarea fișierului imagine disc . . . 185
9.7.4 Curățarea unui fișier imagine de disc . . . 186
9.7.5 Crearea fișierului imagine de disc gol . . . 187
9.7.6 Crearea fișierului imagine ISO9660 . . . 187
9.7.7 Scrierea în mod direct pe CD/DVD-R/RW . . . 188
9.7.8 Montarea fișierului imagine ISO9660 . . . 188
9.8 Datele binare . . . 189
9.8.1 Vizualizarea și editarea datelor binare . . . 189
9.8.2 Manipularea fișierelor fără montarea discului . . . 189
9.8.3 Redundanța datelor . . . 189
9.8.4 Recuperarea fișierelor de date și analiza tehnico-criminalistică . . . 191
9.8.5 Împărțirea unui fișier mare în fișiere mici . . . 191
9.8.6 Ștergerea conținutului fișierului . . . 191
9.8.7 Fișiere fictive . . . 191
9.8.8 Ștergerea întregului disc dur . . . 192
9.8.9 Ștergerea zonei neutilizate a unui disc dur . . . 192
9.8.10 Recuperarea fișierelor șterse, dar încă deschise . . . 192
9.8.11 Căutarea tuturor legăturilor dure . . . 193
9.8.12 Consum invizibil de spațiu pe disc . . . 193
9.9 Sfaturi pentru criptarea datelor . . . 194
9.9.1 Criptarea discurilor amovibile cu dm-crypt/LUKS . . . 194
9.9.2 Montarea discului criptat cu dm-crypt/LUKS . . . 195
9.10 Nucleul . . . 195
9.10.1 Parametrii nucleului . . . 195
9.10.2 Antetele nucleului . . . 196
9.10.3 Compilarea nucleului și a modulelor aferente . . . 196
9.10.4 Compilarea sursei nucleului: Recomandarea echipei Debian Kernel . . . 197
9.10.5 Controlori hardware și firmware . . . 197
9.11 Sistem virtualizat . . . 198
9.11.1 Instrumente de virtualizare și emulare . . . 198
9.11.2 Fluxul de lucru pentru virtualizare . . . 200
9.11.3 Montarea fișierului imagine al discului virtual . . . 200
9.11.4 Sistemul chroot . . . 201
9.11.5 Mai multe sisteme de medii de birou . . . 202

10 Gestionarea datelor 203

10.1 Partajarea, copierea şi arhivarea . . . 203
10.1.1 Instrumente de arhivare şi comprimare . . . 204
10.1.2 Instrumente de copiere şi sincronizare . . . 204
10.1.3 Expresii idiomatice pentru arhivă . . . 204
10.1.4 Expresii idiomatice pentru copiere . . . 206
10.1.5 Expresii idiomatice pentru selectarea fişierelor . . . 207
10.1.6 Suportul de arhivare . . . 208
10.1.7 Dispozitive de stocare amovibile (detaşabile) . . . 209
10.1.8 Alegerea sistemului de fişiere pentru partajarea datelor . . . 210
10.1.9 Partajarea datelor prin reţea . . . 212
10.2 Copia de rezervă şi recuperarea . . . 212
10.2.1 Politica de copie de rezervă şi recuperare . . . 212
10.2.2 Suite de instrumente de copie de rezervă . . . 214
10.2.3 Sfaturi pentru copierea de rezervă . . . 215
10.2.3.1 Copierea de rezervă în interfaţa grafică . . . 215
10.2.3.2 Copiere de rezervă declanşată de evenimentul de montare . . . 216
10.2.3.3 Copiere de rezervă declanşată de evenimentul programatorului de sarcini . . . 216
10.3 Infrastructura de securitate a datelor . . . 217
10.3.1 Gestionarea cheilor pentru GnuPG . . . 218
10.3.2 Utilizarea GnuPG pe fişiere . . . 219
10.3.3 Utilizarea GnuPG cu Mutt . . . 219
10.3.4 Utilizarea GnuPG cu Vim . . . 219
10.3.5 Suma de control MD5 . . . 221
10.3.6 Inel de chei pentru parole . . . 221
10.4 Instrumente de îmbinare a codului sursă . . . 221
10.4.1 Extragerea diferenţelor pentru fişierele sursă . . . 221
10.4.2 Fuzionarea actualizărilor pentru fişierele sursă . . . 223
10.4.3 Îmbinare/fuzionare interactivă . . . 223
10.5 Git . . . 223
10.5.1 Configurarea clientului Git . . . 223
10.5.2 Comenzi Git de bază . . . 225
10.5.3 Sfaturi Git . . . 225
10.5.4 Referinţe Git . . . 226
10.5.5 Alte sisteme de control al versiunii . . . 228

11 Conversia datelor **229**
11.1 Instrumente de conversie a datelor textuale . . . 229
11.1.1 Conversia unui fișier text cu iconv . . . 229
11.1.2 Verificarea fișierului pentru a fi UTF-8 cu iconv . . . 231
11.1.3 Conversia numelor de fișiere cu iconv . . . 231
11.1.4 Conversie a sfârșitului de linie (EOL) . . . 232
11.1.5 Conversie a tabulatoarelor (TAB) . . . 232
11.1.6 Editori cu conversie automată . . . 232
11.1.7 Extragerea textului simplu . . . 233
11.1.8 Evidențierea și formatarea datelor în text simplu . . . 233
11.2 Date XML . . . 233
11.2.1 Sfaturi de bază pentru XML . . . 235
11.2.2 Procesarea XML . . . 236
11.2.3 Extragerea datelor XML . . . 236
11.2.4 Analizarea statică (lint) a datelor XML . . . 237
11.3 Tehnoredactarea (formatarea și pregătirea textului) . . . 237
11.3.1 Tehnoredactare roff . . . 237
11.3.2 TeX/LaTeX . . . 238
11.3.3 Imprimarea paginii de manual într-un format atractiv . . . 239
11.3.4 Crearea unei pagini de manual . . . 239
11.4 Date imprimabile . . . 239
11.4.1 Ghostscript . . . 240
11.4.2 Fuzionarea a două fișiere PS sau PDF . . . 240
11.4.3 Instrumente pentru date imprimabile . . . 240
11.4.4 Imprimarea cu CUPS . . . 240
11.5 Conversia datelor de poștă electronică . . . 242
11.5.1 Noțiuni de bază privind datele de poștă electronică . . . 242
11.6 Instrumente pentru date grafice . . . 243
11.6.1 Instrumente pentru date grafice (metapachet) . . . 243
11.6.2 Instrumente pentru date grafice (cu interfață grafică) . . . 244
11.6.3 Instrumente pentru date grafice (cu interfață de linie de comandă) . . . 244
11.7 Conversii diverse de date . . . 244

12 Programare **247**
12.1 Scriptul Shell . . . 247
12.1.1 Compatibilitate cu shell-ul POSIX . . . 248
12.1.2 Parametri shell . . . 248
12.1.3 Condiționale shell . . . 250
12.1.4 Bucle shell . . . 251

12.1.5 Variabilele de mediu ale shell . . . 251
12.1.6 Secvența de procesare a liniei de comandă shell . . . 251
12.1.7 Programe utile pentru scripturile shell . . . 252
12.2 Programarea în limbaje interpretate . . . 253
12.2.1 Depanarea codurilor în limbaje interpretate . . . 253
12.2.2 Program cu interfață grafică (GUI) și script de shell . . . 254
12.2.3 Acțiuni personalizate pentru gestionarul de fișiere cu interfață grafică . . . 255
12.2.4 Nebunia scripturilor scurte în Perl . . . 255
12.3 Programarea în limbaje compilate . . . 255
12.3.1 C . . . 256
12.3.2 Program simplu în limbajul C (gcc) . . . 256
12.3.3 Flex — o versiune îmbunătățită a Lex . . . 257
12.3.4 Bison — o versiune îmbunătățită a Yacc . . . 257
12.4 Instrumente de analiză statică a codului . . . 259
12.5 Depanare . . . 259
12.5.1 Executarea de bază a gdb . . . 259
12.5.2 Depanarea pachetului Debian . . . 261
12.5.3 Obținerea urmăririi traseului execuției . . . 262
12.5.4 Comenzi avansate gdb . . . 262
12.5.5 Verificați dependențele de biblioteci . . . 263
12.5.6 Instrumente de urmărire dinamică a apelurilor . . . 263
12.5.7 Depanarea erorilor X . . . 263
12.5.8 Instrumente de detectare a „scurgerilor” de memorie . . . 263
12.5.9 Dezasamblarea unui fișier binar . . . 263
12.6 Instrumentele de construcție . . . 264
12.6.1 Make . . . 264
12.6.2 Autotools . . . 265
12.6.2.1 Compilarea și instalarea unui program . . . 265
12.6.2.2 Dezinstalarea unui program . . . 265
12.6.3 Meson . . . 266
12.7 Web . . . 266
12.8 Conversia codului sursă . . . 266
12.9 Crearea pachetului Debian . . . 267

A Anexa **268**

A.1 Labirintul Debian . . . 268
A.2 Istoricul drepturilor de autor . . . 268
A.3 Formatul documentului . . . 269

Listă de tabele

1.1 Lista pachetelor de programe interesante în modul text . . . 5
1.2 Lista pachetelor de documentație utilă . . . 5
1.3 Lista directoarelor cheie și descrierea utilizării acestora . . . 8
1.4 Lista descriptivă a primului caracter din ieșirea comenziil «`ls -l`» . . . 9
1.5 Modul numeric pentru permisiunile de fișiere în comenzile chmod(1) . . . 10
1.6 Exemple de valori pentru **umask** . . . 11
1.7 Lista grupurilor notabile furnizate de sistem pentru accesul la fișiere . . . 12
1.8 Lista grupurilor notabile furnizate de sistem pentru executarea anumitor comenzi . . . 13
1.9 Lista tipurilor de marcaje de timp . . . 13
1.10 Lista fișierelor speciale de dispozitive . . . 17
1.11 Tastele de comenzi rapide ale MC . . . 19
1.12 Reacția la apăsarea tastei «Enter» în MC . . . 20
1.13 Lista programelor shell . . . 21
1.14 Lista tastelor de comenzi rapide pentru bash . . . 22
1.15 Lista operațiilor mouse-ului și acțiunile tastelor asociate în Debian . . . 23
1.16 Lista combinațiilor de taste de bază din Vim . . . 25
1.17 Lista comenzilor Unix de bază . . . 27
1.18 Cele 3 părți ale valorii configurației regionale . . . 28
1.19 Lista recomandărilor privind configurația regională . . . 29
1.20 Lista valorilor variabilei „`$HOME`” . . . 29
1.21 Modele globale shell . . . 30
1.22 Coduri de ieșire ale comenzii . . . 31
1.23 Expresii idiomatice ale comenzii shell . . . 32
1.24 Descriptori de fișiere predefiniți . . . 33
1.25 Metacaractere pentru BRE și ERE . . . 35
1.26 Expresia de înlocuire . . . 36
1.27 Lista fragmentelor de script pentru comenzi de direcționare . . . 39

2.1 Lista instrumentelor de gestionare a pachetelor Debian . . . 41
2.2 Lista siturilor de arhivă Debian . . . 44

2.3 Lista secțiunilor de arhivă Debian . . . 45
2.4 Relația dintre versiune și numele în cod . . . 46
2.5 Lista siturilor web importante pentru rezolvarea problemelor legate de un pachet specific . . . 50
2.6 Operații de bază de gestionare a pachetelor din linia de comandă folosind apt(8), aptitude(8) și apt-get(8) /apt-cache(8) . . . 54
2.7 Opțiuni de comandă demne de menționat pentru aptitude(8) . . . 54
2.8 Lista combinațiilor de taste pentru «aptitude» . . . 55
2.9 Lista vizualizărilor pentru aptitude . . . 56
2.10 Clasificarea vizualizărilor standard ale pachetelor . . . 56
2.11 Lista formulelor de expresii regulate pentru aptitude . . . 58
2.12 Fișierele jurnal pentru activitățile pachetului . . . 59
2.13 Lista operațiilor avansate de gestionare a pachetelor . . . 63
2.14 Conținutul metadatelor arhivei Debian . . . 65
2.15 Structura numelor pachetelor Debian . . . 68
2.16 Caracterele utilizabile pentru fiecare componentă din numele pachetelor Debian . . . 68
2.17 Fișierele importante create de `dpkg` . . . 69
2.18 Lista valorilor notabile (celor mai importante) ale priorității Pin pentru tehnica **apt-pinning** . . . 77
2.19 Lista instrumentelor proxy special pentru arhiva Debian . . . 81

3.1 Lista încărcătorilor de pornire . . . 84
3.2 Semnificația intrării din meniul din partea de sus a `/boot/grub/grub.cfg` . . . 85
3.3 Lista instrumentelor de pornire pentru sistemul Debian . . . 87
3.4 Lista nivelurilor de eroare ale nucleului . . . 91
3.5 Lista fragmentelor tipice de comandă `journalctl` . . . 92
3.6 Lista fragmentelor tipice de comenzi `systemctl` . . . 93
3.7 Lista altor fragmente de comenzi de monitorizare sub `systemd` . . . 94

4.1 3 fișiere de configurare importante pentru pam_unix(8) . . . 97
4.2 Al doilea conținut al intrării „`/etc/passwd`” . . . 98
4.3 Lista comenzilor pentru gestionarea informațiilor contului . . . 99
4.4 Lista instrumentelor pentru generarea parolei . . . 100
4.5 Lista pachetelor principale ale sistemelor PAM ți NSS . . . 101
4.6 Lista fișierelor de configurare accesate de PAM și NSS . . . 101
4.7 Lista serviciilor și porturilor nesigure și sigure . . . 103
4.8 Lista instrumentelor care oferă măsuri suplimentare de securitate . . . 104

5.1 Lista instrumentelor de configurare a rețelei . . . 110
5.2 Lista intervalelor de adrese de rețea . . . 112
5.3 Tabel de corespondență între comenzile învechite `net-tools` și noile comenzi `iproute2` . . . 115
5.4 Lista comenzilor de rețea de nivel inferior . . . 115

5.5 Instrumente de optimizare a rețelei . . . 116
5.6 Reguli de bază pentru valoarea optimă a MTU . . . 117
5.7 Lista instrumentelor de paravan de protecție . . . 118

6.1 Lista navigatoarelor web . . . 121
6.2 Lista agenților de utilizator de poștă electronică („Mail User Agent”: MUA) . . . 122
6.3 Lista pachetelor bazice legate de agentul de transport al poștei . . . 124
6.4 Lista paginilor importante din manualul postfix . . . 125
6.5 Lista fișierelor de configurare legate de adresele de poștă electronică . . . 126
6.6 Lista operațiilor de bază ale MTA . . . 127
6.7 Lista serverelor și instrumentelor de acces la distanță . . . 128
6.8 Lista fișierelor de configurare SSH . . . 129
6.9 Listă de exemple de pornire a clientului SSH . . . 129
6.10 Lista clienților SSH liberi pentru alte platforme . . . 130
6.11 Lista serverelor de imprimare și a utilităților . . . 131
6.12 Lista altor servere de aplicații de rețea . . . 132
6.13 Lista clienților de aplicații de rețea . . . 133
6.14 Lista RFC-urilor populare . . . 133

7.1 Lista mediilor grafice de birou . . . 134
7.2 Lista pachetelor importante de infrastructură de interfață grafică . . . 136
7.3 Lista aplicațiilor cu interfață grafică notabile . . . 138
7.4 Lista fonturilor notabile TrueType și OpenType . . . 139
7.5 Lista mediilor de fonturi notabile și a pachetelor asociate . . . 140
7.6 Lista mediilor sandbox notabile și a pachetelor asociate . . . 141
7.7 Lista serverelor notabile de acces la distanță . . . 142
7.8 Lista metodelor de conectare la serverul X . . . 142
7.9 Lista programelor legate de manipularea clipboardului de caractere . . . 144

8.1 Lista pachetelor IBus și a modulelor sale . . . 150
8.2 Lista Fcitx5 și a pachetelor sale de module . . . 151

9.1 Lista programelor ce ajută în activitățile cu consola . . . 154
9.2 Lista combinațiilor de taste pentru «screen» . . . 156
9.3 Informații despre inițializarea `vim` . . . 160
9.4 Lista analizatoarelor de jurnale de sistem . . . 161
9.5 Exemplele de afișare a datei și orei pentru comanda „`ls -l`” cu **valoarea stilului de afișare a datei și orei** . . . 162
9.6 Lista instrumentelor de manipulare a imaginilor grafice . . . 163
9.7 Lista pachetelor care pot înregistra istoricul configurațiilor . . . 163
9.8 Lista instrumentelor pentru monitorizarea și controlul activităților programului . . . 164

9.9 Lista valorilor de curtoazie pentru prioritatea de programare . . . 165
9.10 Lista stilurilor de comenzi ps . . . 165
9.11 Lista semnalelor utilizate frecvent pentru comanda «kill» . . . 170
9.12 Lista tastelor de comandă SAK importante . . . 171
9.13 Lista instrumentelor de identificare a hardware-ului . . . 173
9.14 Lista instrumentelor de configurare hardware . . . 173
9.15 Lista pachetelor de sunet . . . 175
9.16 Lista comenzilor pentru dezactivarea protectorului de ecran . . . 175
9.17 Lista dimensiunilor memoriei raportate . . . 176
9.18 Lista instrumentelor pentru verificarea securității și integrității sistemului . . . 177
9.19 Lista pachetelor de gestionare a partițiilor de disc . . . 178
9.20 Lista pachetelor de gestionare a sistemului de fișiere . . . 180
9.21 Lista pachetelor care vizualizează și editează date binare . . . 189
9.22 Lista pachetelor pentru manipularea fișierelor fără montarea discului . . . 189
9.23 Lista instrumentelor pentru adăugarea redundanței datelor la fișiere . . . 190
9.24 Lista pachetelor pentru recuperarea fișierelor de date și analiza tehnico-criminalistică . . . 190
9.25 Lista instrumentelor de criptare a datelor . . . 194
9.26 Lista pachetelor cheie care trebuie instalate pentru recompilarea nucleului pe sistemul Debian . . . 196
9.27 Lista instrumentelor de virtualizare . . . 199

10.1 Lista instrumentelor de arhivare și comprimare . . . 205
10.2 Lista instrumentelor de copiere și sincronizare . . . 206
10.3 Lista opțiunilor de sistem de fișiere pentru dispozitive de stocare amovibile cu scenarii tipice de utilizare 211
10.4 Lista serviciilor de rețea din care puteți alege, cu scenariul tipic de utilizare . . . 212
10.5 Lista suitelor de instrumente de copiere de rezervă . . . 214
10.6 Lista instrumentelor de infrastructură pentru securitatea datelor . . . 217
10.7 Lista comenzilor GNU Privacy Guard pentru gestionarea cheilor . . . 218
10.8 Lista semnificațiilor codului de încredere . . . 218
10.9 Lista comenzilor GNU Privacy Guard pentru fișiere . . . 220
10.10 Lista instrumentelor de îmbinare a codului sursă . . . 222
10.11 Lista pachetelor și comenzilor legate de git . . . 224
10.12 Comenzile git principale . . . 226
10.13 Sfaturi Git . . . 227
10.14 Lista altor instrumente pentru sistemul de control al versiunilor . . . 228

11.1 Lista instrumentelor de conversie a datelor textuale . . . 229
11.2 Lista valorilor de codificare și utilizarea acestora . . . 230
11.3 Lista stilurilor EOL pentru diferite platforme . . . 232
11.4 Lista comenzilor de conversie a tabulatoarelor (TAB) din pachetele `bsdmainutils` și `coreutils` . . . 232

11.5 Lista instrumentelor pentru extragerea datelor în text simplu . . . 234
11.6 Lista instrumentelor pentru evidențierea datelor în text simplu . . . 234
11.7 Lista entităților predefinite pentru XML . . . 235
11.8 Lista instrumentelor XML . . . 236
11.9 Lista instrumentelor DSSSL . . . 236
11.10 Lista instrumentelor de extragere a datelor XML . . . 237
11.11 Lista instrumentelor de imprimare elegantă XML . . . 237
11.12 Lista instrumentelor de tehnoredactare . . . 238
11.13 Lista pachetelor utile pentru crearea paginilor de manual . . . 239
11.14 Lista interpreților Ghostscript PostScript . . . 240
11.15 Lista instrumentelor pentru date imprimabile . . . 241
11.16 Lista pachetelor utile pentru conversia datelor de poștă electronică . . . 242
11.17 Lista instrumentelor pentru date grafice (metapachet) . . . 243
11.18 Lista instrumentelor pentru date grafice (cu interfață grafică) . . . 244
11.19 Lista instrumentelor pentru date grafice (cu interfață de linie de comandă) . . . 245
11.20 Lista instrumentelor diverse de conversie a datelor . . . 246

12.1 Lista expresiilor tipice din Bash („bashisme”) . . . 248
12.2 Lista parametrilor shell-ului . . . 249
12.3 Lista expansiunilor parametrilor shell-ului . . . 249
12.4 Lista substituțiilor parametrilor cheie ai shell-ului . . . 249
12.5 Lista operatorilor de comparare a fișierelor în expresia condițională . . . 250
12.6 Lista operatorilor de comparare a șirurilor în expresia condițională . . . 251
12.7 Lista pachetelor care conțin mici programe utilitare pentru scripturi shell . . . 253
12.8 Lista pachetelor legate de interpret . . . 253
12.9 Lista programelor de dialog . . . 254
12.10 Lista pachetelor legate de compilator . . . 256
12.11 Lista generatoarelor de analizare LALR compatibile cu Yacc . . . 257
12.12 Lista instrumentelor pentru analiza statică a codului . . . 260
12.13 Lista pachetelor de depanare . . . 260
12.14 Lista comenzilor avansate gdb . . . 262
12.15 Lista instrumentelor de detectare a „scurgerilor” de memorie . . . 263
12.16 Lista pachetelor de instrumente de construire . . . 264
12.17 Lista variabilelor automate «make» . . . 264
12.18 Lista expansiunilor variabilelor «make» . . . 265
12.19 Lista instrumentelor de conversie a codului sursă . . . 267

Rezumat

Această carte este gratuită; o puteți redistribui și/sau modifica în conformitate cu termenii Licenței Publice Generale GNU, în orice versiune compatibilă cu Ghidul Debian pentru Software Liber (DFSG).

Prefață

Acest Ghid de referință Debian (versiunea 2.152) (2026-10-10 14:56:25 UTC) are scopul de a oferi o imagine de ansamblu asupra administrării sistemului Debian, ca ghid de utilizare post-instalare.

Cititorul vizat este cineva care dorește să învețe scripturi shell, dar care nu este pregătit să citească toate sursele C pentru a înțelege cum funcționează sistemul GNU/Linux.

Pentru instrucțiuni de instalare, consultați:

- Ghidul de instalare Debian GNU/Linux pentru versiunea stabilă actuală
- Ghidul de instalare Debian GNU/Linux pentru versiunea de testare actuală

Declinare de responsabilitate

Toate garanțiile sunt declinate. Toate mărcile comerciale sunt proprietatea deținătorilor respectivi ai mărcilor comerciale.

Sistemul Debian în sine este un obiectiv în continuă schimbare. Acest lucru face ca documentația sa să fie dificil de actualizat și corectat. Deși versiunea actuală de testare `testing` a sistemului Debian a fost utilizată ca bază pentru redactarea acestui document, este posibil ca unele informații să fie deja depășite în momentul în care citiți aceste rânduri.

Vă rugăm să tratați acest document ca referință secundară. Acest document nu înlocuiește niciun ghid oficial. Autorul și colaboratorii nu își asumă responsabilitatea pentru consecințele erorilor, omisiunilor sau ambiguităților din acest document.

Ce este Debian

Proiectul Debian este o asociație de persoane care s-au unit pentru a crea un sistem de operare liber. Distribuția sa se caracterizează prin următoarele.

- Angajamentul față de libertatea software-ului: Contractul social Debian și Ghidul Debian pentru software-ul liber (DFSG)
- Distribuirea rezultatului muncii voluntare neremunerate în Internet: https://www.debian.org
- Un număr mare de pachete software precompilate de înaltă calitate
- Accent pe stabilitate și securitate, cu acces facil la actualizările de securitate
- Accent pe actualizarea fără probleme la cele mai recente pachete software din arhivele `testing`
- Oferă suport pentru un număr mare de arhitecturi hardware

Componentele software-ului liber din Debian provin din GNU, Linux, BSD, X, ISC, Apache, Ghostscript, Common Unix Printing System , Samba, GNOME, KDE, Mozilla, LibreOffice, Vim, TeX, LaTeX, DocBook, Perl, Python, Tcl, Java, Ruby, PHP, Berkeley DB, MariaDB, PostgreSQL, SQLite, Exim, Postfix, Mutt, FreeBSD, OpenBSD, Plan 9 și din multe alte proiecte independente de software liber. Debian integrează această diversitate de software liber într-un singur sistem.

Despre acest document

Reguli

La întocmirea acestui document s-au respectat următoarele reguli.

- Furnizează o imagine de ansamblu și omite cazurile speciale. (**Imaginea de ansamblu**)
- Menținerea acestuia într-o formă scurtă și simplă. (aplicarea principiului **KISS**)
- Aplicarea principiului „Nu reinventa roata!”. (Folosește trimiteri către **referințele existente**)
- Pune accentul pe instrumentele non-GUI și console. (Folosește **exemple shell**)
- Să fie obiectiv. (Folosește popcon etc.)

Indicație
S-a încercat să se elucideze aspectele ierarhice și nivelurile inferioare ale sistemului.

Cerințe prealabile



Avertisment
Se așteaptă ca dumneavoastră să depuneți eforturi susținute pentru a găsi răspunsuri pe cont propriu, dincolo de această documentație. Acest document oferă doar puncte de plecare eficiente.

Trebuie să căutați soluții pe cont propriu, din surse primare.

- Situl Debian la https://www.debian.org pentru informații generale
- Documentația din directorul „`/usr/share/doc/`*`nume-pachet`*”
- **Pagina de manual (manpage)** în stil Unix: «`dpkg -L` *`nume-pachet`* `|grep '/man/man.*/'`»
- **Pagina de informații (info page)** în stil Unix: «`dpkg -L` *`nume-pachet`* `|grep '/info/'`»
- Raportul de eroare: https://bugs.debian.org/*nume-pachet*
- Wiki-ul Debian la https://wiki.debian.org/ pentru subiecte specifice și în mișcare
- Specificația UNIX unică (Single UNIX Specification) de la Open Group The UNIX System Home Page
- Enciclopedia liberă Wikipedia la https://www.wikipedia.org/
- Manualul administratorului Debian
- Ghidurile HOWTO de la Proiectul de documentație Linux (TLDP)

Notă
Pentru documentație detaliată, poate fi necesar să instalați pachetul de documentație corespunzător, al cărui nume este numele pachetului cu sufixul „`-doc`”.

Convenții

Acest document oferă informații prin intermediul următorului stil de prezentare simplificat, cu exemple de comenzi shell bash(1).

```
# command-in-root-account
$ command-in-user-account
```

Aceste prompturi shell disting contul utilizat și corespund variabilelor de mediu definite astfel: „PS1='\$'” și „PS2='>'”. Aceste valori sunt alese pentru a facilita citirea acestui document și nu sunt tipice pentru sistemul instalat efectiv.

Toate exemplele de comenzi sunt rulate în limba engleză „LANG=en_US.UTF8”. Vă rugăm să nu vă așteptați ca șirurile de caractere de substituție, cum ar fi *command-in-root-account* (comandă-în-contul-root) și *command-in-user-ac* (comandă-în-contul-utilizatorului), să fie traduse în exemplele de comenzi. Aceasta este o alegere intenționată pentru a menține toate exemplele traduse la zi.

Notă
Consultați semnificația variabilelor de mediu „$PS1” și „$PS2” în bash(1).

Acțiunea necesară din partea administratorului de sistem este scrisă într-o propoziție imperativă, de exemplu „Apăsați tasta Enter după ce introduceți fiecare șir de comenzi în shell”.

Coloana **descriere** și coloanele similare din tabel pot conține o **fraza nominală** care urmează convenția de descriere scurtă a pachetului care omite articolele introductive precum „un” și „o”. Alternativ, ele pot conține o frază infinitivă ca **fraza nominală** fără „la” la început, urmând convenția de descriere scurtă a comenzilor din paginile man. Acestea pot părea ciudate pentru unii, dar sunt alegeri stilistice intenționate ale mele pentru a păstra această documentație cât mai simplă posibil. Aceste **fraze nominale** nu încep cu majusculă și nu se termină cu punct, conform convenției de descriere scurtă.

Notă
Numele proprii, inclusiv numele comenzilor, își păstrează majusculele/minusculele indiferent de locația lor.

Un **fragment de comandă** citat într-un paragraf de text este indicat prin fontul de mașină de scris între ghilimele franceze, cum ar fi «`aptitude safe-upgrade`».

Un **text de date** dintr-un fișier de configurare citat într-un paragraf de text este indicat prin fontul de mașină de scris între ghilimele, cum ar fi „`deb-src`”.

O **comandă** este menționată prin numele său în fontul de mașină de scris, urmată opțional de numărul secțiunii din pagina de manual între paranteze, cum ar fi bash(1). Vă recomandăm să obțineți informații tastând următoarele.

```
$ man 1 bash
```

O **pagină de manual** este indicată prin numele său scris cu fontul de mașină de scris, urmat de numărul secțiunii paginii de manual între paranteze, cum ar fi sources.list(5). Vă recomandăm să obțineți informații tastând următoarele.

```
$ man 5 sources.list
```

O **pagină de informații** este indicată prin fragmentul de comandă scris cu fontul de mașină de scris între ghilimele, cum ar fi „`info make`”. Vă încurajăm să obțineți informații tastând următoarele.

```
$ info make
```

Un **nume de fișier** este indicat prin fontul de mașină de scris între ghilimele, cum ar fi „`/etc/passwd`”. Pentru fișierele de configurare, vă recomandăm să obțineți informații tastând următoarele.

```
$ sensible-pager "/etc/passwd"
```

Un **nume de director** este indicat prin fontul de mașină de scris între ghilimele, cum ar fi „`/etc/apt/`”. Vă încurajăm să explorați conținutul acestuia tastând următoarele.

```
$ mc "/etc/apt/"
```

Un **nume de pachet** este menționat prin numele său în fontul de mașină de scris, cum ar fi `vim`. Vă încurajăm să obțineți informații tastând următoarele.

```
$ dpkg -L vim
$ apt-cache show vim
$ aptitude show vim
```

O **documentație** poate indica locația sa prin numele fișierului în fontul de mașină de scris între ghilimele, cum ar fi „`/usr/share/doc/base-passwd/users-and-groups.txt.gz`” și „`/usr/share/doc/base-passwd/users-and-` sau prin adresa URL, cum ar fi https://www.debian.org. Vă încurajăm să citiți documentația tastând următoarele.

```
$ zcat "/usr/share/doc/base-passwd/users-and-groups.txt.gz" | sensible-pager
$ sensible-browser "/usr/share/doc/base-passwd/users-and-groups.html"
$ sensible-browser "https://www.debian.org"
```

O **variabilă de mediu** este indicată prin numele său precedat de „$” în fontul de mașină de scris între ghilimele, cum ar fi „`$TERM`”. Vă recomandăm să obțineți valoarea sa curentă tastând următoarea comandă.

```
$ echo "$TERM"
```

Statistici de utilizare (popcon)

Datele popcon sunt prezentate ca măsură obiectivă a popularității fiecărui pachet. Acestea au fost descărcate la data de 2026-10-07 05:10:21 UTC și conțin totalul de 291550 de raportări referitoare la 224298 de pachete binare și 25 de arhitecturi.

Notă
Vă rugăm să rețineți că arhiva `amd64 unstable` conține în prezent numai pachete 78072. Datele popcon conțin rapoarte de la multe instalări de sistem vechi.

Numărul popcon precedat de „V:” pentru „voturi” se calculează astfel: „1000 * (numărul de trimiteri popcon pentru pachetul executat recent pe PC)/(numărul total de trimiteri popcon)”.

Numărul popcon precedat de „I:” pentru „instalări” se calculează astfel: „1000 * (numărul de rapoarte popcon pentru pachetul instalat în PC)/(numărul total de rapoarte popcon)”.

Notă
Cifrele popcon nu trebuie considerate ca fiind măsuri absolute ale importanței pachetelor. Există mulți factori care pot denatura statisticile. De exemplu, unele sisteme care participă la popcon pot avea directoare montate, cum ar fi „`/usr/bin`” cu opțiunea „`noatime`” pentru îmbunătățirea performanței sistemului și au dezactivat efectiv „votul” din astfel de sisteme.

Dimensiunea pachetului

Datele privind dimensiunea pachetului sunt, de asemenea, prezentate ca măsură obiectivă pentru fiecare pachet. Aceasta se bazează pe „`Installed-Size:`” raportată de comanda «`apt-cache show`» sau «`aptitude show`» (în prezent pe arhitectura `amd64` pentru versiunea `unstable`). Dimensiunea raportată este exprimată în Kio (Kibi-octet = unitate pentru 1024 octeți).

Notă
Un pachet cu o dimensiune numerică mică poate indica faptul că pachetul din versiunea `unstable` este un pachet fictiv care instalează alte pachete cu conținut semnificativ prin dependență. Pachetul fictiv permite o tranziție lină sau divizarea pachetului.

Notă
O dimensiune a pachetului urmată de „(*)” indică faptul că pachetul din versiunea `unstable` lipsește și că se utilizează în schimb dimensiunea pachetului pentru versiunea `experimental`.

Rapoarte de erori privind acest document

Vă rugăm să raportați erorile din pachetul `debian-reference` folosind reportbug(1) dacă găsiți probleme în acest document. Vă rugăm să includeți sugestii de corectare folosind «`diff -u`» în versiunea text simplu sau în sursă.

Sfaturi pentru utilizatorii noi

Iată câteva sfaturi pentru utilizatorii noi:

- Faceți o copie de rezervă a datelor
  - A se vedea Secțiune 10.2.
- Protejați-vă parola și cheile de securitate
- KISS (keep it simple stupid) - păstrează lucrurile simple, prostule
  - Nu complicați excesiv sistemul
- Citiți fișierele jurnal
  - **PRIMA** eroare este cea care contează
- RTFM: „Read the fucking manual” - (Citește naibii manualul!)
- Caută în Internet înainte de a pune întrebări
- Nu fi root când nu este necesar să fii
- Nu te juca cu sistemul de gestionare a pachetelor
- Nu tasta nimic din ceea ce nu înțelegi
- Nu schimba permisiunile fișierului (înainte de revizuirea completă a securității)
- Nu părăsi shell-ul root până nu **TESTEZI** modificările
- Aveți întotdeauna la îndemână un suport de pornire alternativ (unitate flash USB, CD-ROM, …)

Câteva citate pentru utilizatorii noi

Iată câteva citate interesante din lista de discuții Debian care pot ajuta la lămurirea noilor utilizatori.

- „Acesta este Unix. Îți oferă suficientă funie pentru a te spânzura.” --- Miquel van Smoorenburg <miquels at cistron.nl>
- „Unix este prietenos cu utilizatorul... Doar că este selectiv în ceea ce privește prietenii pe care și-i alege.” --- Tollef Fog Heen <tollef at add.no>

Wikipedia are un articol intitulat „Unix philosophy” (Filosofia Unix) care conține citate interesante.

Capitolul 1

Tutoriale GNU/Linux

Cred că învățarea unui sistem informatic este similară cu învățarea unei limbi străine. Deși manualele și documentația sunt utile, trebuie să exersați singur. Pentru a vă ajuta să începeți fără probleme, voi detalia câteva puncte de bază.

Construcția robustă a Debian GNU/Linux provine din sistemul de operare Unix, adică un sistem de operare multi-utilizator și multi-sarcină. Trebuie să învățați să profitați de forța acestor caracteristici și similitudini între Unix și GNU/Linux.

Nu vă feriti de textele orientate spre Unix și nu vă bazați exclusiv pe textele GNU/Linux, deoarece astfel veți pierde multe informații utile.

Notă
Dacă utilizați de ceva timp un sistem de tip Unix cu instrumente de linie de comandă, probabil că știți deja tot ce explic aici. Vă rugăm să utilizați acest articol ca o verificare a realității și o reîmprospătare a cunoștințelor.

1.1 Noțiuni de bază despre consolă

1.1.1 Promptul shell

La pornirea sistemului, vi se afișează ecranul de autentificare bazat pe caractere dacă nu ați instalat niciun mediu GUI precum GNOME sau KDE. Să presupunem că numele gazdei dvs. este `foo`, promptul de autentificare arată astfel.

Dacă ați instalat un mediu GUI, puteți accesa în continuare promptul de autentificare bazat pe caractere apăsând «Ctrl-Alt-F3» și puteți reveni la mediul GUI apăsând «Ctrl-Alt-F2» (pentru mai multe informații, a se vedea Secțiune 1.1.6 mai jos).

```
foo login:
```

La solicitarea de autentificare, introduceți numele de utilizator, de exemplu `penguin`, și apăsați tasta «Enter», apoi introduceți parola și apăsați din nou tasta «Enter».

Notă
Conform tradiției Unix, numele de utilizator și parola sistemului Debian sunt sensibile la majuscule și minuscule. Numele de utilizator este ales de obicei numai din litere mici. Primul cont de utilizator este creat de obicei în timpul instalării. Conturi de utilizator suplimentare pot fi create cu adduser(8) de către root.

Sistemul pornește cu mesajul de salut stocat în „`/etc/motd`” („Message Of The Day” - Mesajul zilei) și afișează o linie de comandă.

```
Debian GNU/Linux 12 foo tty3

foo login: penguin
Password:

Linux foo 6.5.0-0.deb12.4-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.5.10-1~bpo12+1 (2023-11-23) ↩
     x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Last login: Wed Dec 20 09:39:00 JST 2023 on tty3
foo:~$
```

Acum vă aflați în shell. Shell-ul interpretează comenzile dvs.

1.1.2 Promptul shell în GUI

Dacă ați instalat un mediu GUI în timpul instalării, la pornirea sistemului vi se va afișa ecranul grafic de autentificare. Introduceți numele de utilizator și parola pentru a vă autentifica în contul de utilizator fără privilegii. Utilizați tasta «Tab» pentru a naviga între numele de utilizator și parolă sau utilizați butonul principal al mouse-ului.

Puteți accesa promptul shell în mediul GUI pornind un program `x-terminal-emulator` precum gnome-terminal(1), rxvt(1) sau xterm(1). În mediul grafic de birou GNOME, apăsați tasta «SUPER» (tasta «Windows») și tastați „terminal” în câmpul de căutare.

În cazul altor sisteme grafice de birou (cum ar fi `fluxbox`), este posibil să nu existe un punct de pornire evident pentru meniu. Dacă se întâmplă acest lucru, încercați să faceți clic (dreapta) pe fundalul imaginii de birou și așteptați să apară un meniu.

1.1.3 Contul root

Contul root este denumit și super-utilizator sau utilizator privilegiat. Din acest cont, puteți efectua următoarele sarcini de administrare a sistemului.

- Citește, scrie și elimină orice fișiere din sistem, indiferent de permisiunile acestor fișiere.
- Stabilește drepturile de proprietate și permisiunile pentru orice fișier din sistem
- Stabilește parola pentru orice utilizator fără privilegii din sistem
- Se conectează la orice cont fără a introduce parola

Această putere nelimitată a contului root necesită să fiți atent și responsabil atunci când îl utilizați.



Avertisment
Nu divulgați niciodată parola de root altor persoane.

Notă
Permisiunile unui fișier (inclusiv dispozitive hardware precum CD-ROM etc., care sunt doar un alt fișier pentru sistemul Debian) pot face ca acesta să fie inutilizabil sau inaccesibil pentru utilizatorii care nu sunt root. Deși utilizarea contului root este o modalitate rapidă de a testa acest tip de situație, rezolvarea acesteia trebuie făcută prin stabilirea corespunzătoare a permisiunilor fișierului și a apartenenței utilizatorului la grup (a se vedea Secțiune 1.2.3).

1.1.4 Promptul shell-ului root

Iată câteva metode de bază pentru a obține promptul shell root utilizând parola root.

- Tastați `root` la promptul de autentificare bazat pe caractere.
- Tastați „`su -l`” din orice prompt al shell-ului utilizatorului.
  - Aceasta nu păstrează mediul utilizatorului curent.
- Tastați „`su`” din orice prompt al shell-ului utilizatorului.
  - Aceasta păstrează o parte din mediul utilizatorului curent.

1.1.5 Instrumente grafice de administrare a sistemului

Când meniul mediului grafic de birou nu pornește automat instrumentele de administrare a sistemului GUI cu privilegiile corespunzătoare, le puteți porni din promptul shell-ului root al emulatorului de terminal, cum ar fi gnome-terminal(1), rxvt(1) sau xterm(1). Consultați Secțiune 1.1.4 și Secțiune 7.9.



Avertisment
Nu porniți niciodată administratorul de afișare/sesiune cu interfață grafică sub contul root introducând `root` la promptul administratorului de afișare, cum ar fi gdm3(1).
Nu rulați niciodată programe cu interfață grafică la distanță care nu sunt de încredere sub X Window atunci când sunt afișate informații critice, deoarece acestea pot intercepta ecranul X.

1.1.6 Console virtuale

În sistemul Debian implicit, există șase console de caractere comutabile de tip VT100 disponibile pentru a porni shell-ul de comandă direct pe gazda Linux. Cu excepția cazului în care vă aflați într-un mediu cu interfață grafică, puteți comuta între consolele virtuale apăsând simultan tasta `Alt-stânga` și una dintre tastele `F1` — `F6`. Fiecare consolă de caractere permite conectarea independentă la cont și oferă un mediu multiutilizator. Acest mediu multiutilizator este o caracteristică excelentă a Unix și foarte captivantă.

Dacă vă aflați în mediul cu interfață grafică, puteți accesa consola de caractere 3 apăsând tastele `Ctrl-Alt-F3`, adică apăsând simultan `tasta Ctrl-stânga`, tasta `Alt-stânga` și tasta `F3`. Puteți reveni la mediul cu interfață grafică, care rulează în mod normal pe consola virtuală 2, apăsând `Alt-F2`.

Alternativ, puteți trece la o altă consolă virtuală, de exemplu la consola 3, din linia de comandă.

```
# chvt 3
```

1.1.7 Cum să ieșiți din promptul liniei de comandă

Tastați `Ctrl-D`, adică apăsați simultan tasta `Ctrl-stânga` și tasta d în linia de comandă pentru a închide activitatea shell-ului. Dacă vă aflați în consola de caractere, veți reveni la linia de autentificare. Chiar dacă aceste caractere de control sunt denumite «control D» cu majusculă, nu este necesar să apăsați tasta «Shift». Expresia prescurtată, `^D`, este utilizată de asemenea pentru `Ctrl-D`. Alternativ, puteți tasta „exit”.

Dacă vă aflați în x-terminal-emulator(1), puteți închide fereastra `x-terminal-emulator` cu această comandă.

1.1.8 Cum să opriți sistemul

La fel ca orice alt sistem de operare modern în care operațiile cu fișiere implică stocarea datelor în memoria cache pentru o performanță îmbunătățită, sistemul Debian necesită o procedură de oprire corespunzătoare înainte ca alimentarea să poată fi oprită în siguranță. Acest lucru are scopul de a menține integritatea fișierelor, forțând toate modificările din memorie să fie scrise pe disc. Dacă este disponibilă funcția de control al alimentării software, procedura de oprire oprește automat alimentarea sistemului. (În caz contrar, poate fi necesar să apăsați butonul de alimentare timp de câteva secunde după procedura de oprire.)

Puteți opri sistemul în modul multiutilizator normal din linia de comandă.

```
# shutdown -h now
```

Puteți opri sistemul în modul utilizator unic din linia de comandă.

```
# poweroff -i -f
```

Consultați Secțiune 6.3.8.

1.1.9 Recuperarea unei console funcționale

Când ecranul devine nebun după ce faceți unele lucruri ciudate, cum ar fi „`cat` *`vreun-fișier-binar`*”, tastați «`reset`» în promptul de comandă. Este posibil să nu puteți vedea comanda introdusă pe măsură ce o tastați. De asemenea, puteți tasta «`clear`» pentru a curăța ecranul.

1.1.10 Sugestii de pachete suplimentare pentru începători

Deși chiar și instalarea minimă a sistemului Debian fără niciun mediu grafic de birou oferă funcționalitatea de bază Unix, este o idee bună să instalați câteva pachete suplimentare pentru terminalul caracter bazat pe linia de comandă și curses, cum ar fi `mc` și `vim` cu apt-get(8) pentru începători, pentru a începe cu următoarele.

```
# apt-get update
 ...
# apt-get install mc vim sudo aptitude
 ...
```

Dacă aveți deja aceste pachete instalate, nu se instalează pachete noi.

Ar fi o idee bună să citiți câteva documente informative.

Puteți instala unele dintre aceste pachete urmând pașii de mai jos.

```
# apt-get install package_name
```

pachet	popcon(popularitate) și dimensiune		descriere
mc	V:40, I:176	1590	Un gestionar de fișiere în modul text pe ecran complet
sudo	V:761, I:869	6814	Un program care permite utilizatorilor privilegii root limitate
vim	V:83, I:338	4164	Editorul de text Unix Vi IMproved, un editor de text pentru programatori (versiunea standard)
vim-tiny	V:59, I:980	1867	Editorul de text Unix Vi IMproved, un editor de text pentru programatori (versiunea compactă)
emacs-nox	V:3, I:12	46564	Proiectul GNU Emacs, editorul de text extensibil bazat pe Lisp
w3m	V:10, I:129	2853	Navigatoare web în modul text
gpm	V:8.3, I:9.0	524	Copierea și lipirea în stil Unix în consola text (demon, simulând un mouse)

Tabela 1.1: Lista pachetelor de programe interesante în modul text

pachet	popcon(popularitate) și dimensiune		descriere
doc-debian	I:886	185	Documentația proiectului Debian (Întrebări frecvente despre Debian) și alte documente
debian-policy	I:6.8	5147	Manualul de politici Debian și documentele conexe
developers-reference	V:0.2, I:2.2	2657	Ghiduri și informații pentru dezvoltatorii Debian
debmake-doc	I:0.33	11803	Ghid pentru responsabilii cu întreținerea pachetelor Debian
debian-history	I:0.50	6513	Istoria proiectului Debian
debian-faq	I:885	791	Întrebări frecvente despre Debian (FAQ)

Tabela 1.2: Lista pachetelor de documentație utilă

1.1.11 Un cont de utilizator suplimentar

Dacă nu doriți să utilizați contul principal de utilizator pentru următoarele activități de instruire, puteți crea un cont de utilizator pentru instruire, de exemplu `fish`, urmând pașii de mai jos.

```
# adduser fish
```

Răspundeți la toate întrebările.

Aceasta creează un cont nou numit `fish`. După ce ați exersat, puteți șterge acest cont de utilizator și directorul său principal urmând pașii de mai jos.

```
# deluser --remove-home fish
```

În sistemele non-Debian și Debian specializate, activitățile de mai sus trebuie să utilizeze în schimb instrumentele de nivel inferior useradd(8) și userdel(8).

1.1.12 Configurarea «sudo»

Pentru stațiile de lucru tipice cu un singur utilizator, cum ar fi sistemul Debian cu mediu grafic de birou pe laptop, este obișnuit să se implementeze o configurație simplă a sudo(8) după cum urmează, pentru a permite utilizatorului fără privilegii, de exemplu `penguin`, să obțină privilegii administrative doar cu parola sa de utilizator, fără a fi nevoie de parola root.

```
# echo "penguin  ALL=(ALL) ALL" >> /etc/sudoers
```

Alternativ, este de asemenea obișnuit să se procedeze după cum urmează pentru a permite utilizatorului fără privilegii, de exemplu `penguin`, să obțină privilegii administrative fără nicio parolă.

```
# echo "penguin  ALL=(ALL) NOPASSWD:ALL" >> /etc/sudoers
```

Această truc ar trebui folosit numai pentru stația de lucru cu un singur utilizator pe care o administrați și unde sunteți singurul utilizator.



Avertisment
Nu configurați conturi de utilizatori obișnuiți pe stații de lucru multiutilizator de acest tip, deoarece acest lucru ar afecta grav securitatea sistemului.



Atenție
Parola și contul `penguin` din exemplul de mai sus necesită aceeași protecție ca și parola root și contul root. În acest context, privilegiul administrativ aparține unei persoane autorizate să îndeplinească sarcini de administrare a sistemului pe stația de lucru. Nu acordați niciodată un astfel de privilegiu unui responsabil din departamentul administrativ al companiei dvs. sau șefului dvs., cu excepția cazului în care aceștia sunt autorizați și capabili să îndeplinească astfel de sarcini.

Notă
Pentru a acorda privilegii de acces la dispozitive și fișiere limitate, ar trebui să luați în considerare utilizarea **grupului** pentru a acorda acces limitat, în loc să utilizați privilegiul `root` prin intermediul sudo(8).
Cu o configurare mai atentă și mai grijulie, sudo(8) poate acorda privilegii administrative limitate altor utilizatori dintr-un sistem partajat, fără a partaja parola root. Acest lucru poate ajuta la asigurarea responsabilității în cazul gazdelor cu mai mulți administratori, astfel încât să puteți ști cine a făcut ce. Pe de altă parte, este posibil să nu doriți ca altcineva să aibă astfel de privilegii.

1.1.13 Ora de joacă

Acum sunteți gata să utilizați sistemul Debian fără riscuri, atâta timp cât utilizați contul de utilizator fără privilegii.

Acest lucru se datorează faptului că sistemul Debian, chiar și după instalarea implicită, este configurat cu permisiuni de fișiere adecvate, care împiedică utilizatorii fără privilegii să deterioreze sistemul. Desigur, pot exista încă unele vulnerabilități care pot fi exploatate, dar cei care sunt preocupați de aceste probleme nu ar trebui să citească această secțiune, ci ar trebui să citească Manualul de securizare Debian.

Vom învăța sistemul Debian ca un sistem de tip Unix cu ajutorul următoarelor:

- Secțiune 1.2 (conceptul de bază)
- Secțiune 1.3 (metoda de supraviețuire)
- Secțiune 1.4 (metoda de bază)
- Secțiune 1.5 (mecanismul shell)
- Secțiune 1.6 (metoda de procesare a textului)

1.2 Sistem de fișiere de tip Unix

În GNU/Linux și alte sisteme de operare de tip Unix, fișierele sunt organizate în directoare. Toate fișierele și directoarele sunt aranjate într-o singură structură arborescentă cu rădăcina în „/”. Se numește structură arborescentă deoarece, dacă desenezi sistemul de fișiere, acesta arată ca un arbore, dar cu rădăcina în sus.

Aceste fișiere și directoare pot fi distribuite pe mai multe dispozitive. mount(8) servește la atașarea sistemului de fișiere găsit pe un dispozitiv la arborele de fișiere mare. În schimb, umount(8) îl detașează din nou. Pe nucleele Linux

recente, mount(8) cu anumite opțiuni poate lega o parte a arborelui de fișiere în altă parte sau poate monta sistemul de fișiere ca fiind partajat, privat, secundar sau nelegat. Opțiunile de montare acceptate pentru fiecare sistem de fișiere sunt disponibile în „`/usr/share/doc/linux-doc-*/Documentation/filesystems/`”.

Directoarele din sistemele Unix sunt denumite **dosare** în alte sisteme. De asemenea, rețineți că nu există conceptul de **unitate** precum „`A:`” pe niciun sistem Unix. Există un singur sistem de fișiere și totul este inclus. Acesta este un avantaj enorm în comparație cu Windows.

1.2.1 Noțiuni de bază despre fișierele Unix

Iată câteva noțiuni de bază despre fișierele Unix.

- Numele fișierelor sunt **sensibile la majuscule și minuscule**. Adică, „`FIȘIERULMEU`” și „`FișierulMEU`” sunt fișiere diferite.
- **Directorul rădăcină** înseamnă rădăcina sistemului de fișiere, denumită simplu „`/`”. Nu confundați acest lucru cu directorul de bază al utilizatorului rădăcină: „`/root`”.
- Fiecare director are un nume care poate conține orice litere sau simboluri **cu excepția „`/`”**. Directorul rădăcină este o excepție; numele său este „`/`” (pronunțat „bară oblică” sau „directorul rădăcină”) și nu poate fi redenumit.
- Fiecare fișier sau director este desemnat printr-un **nume de fișier complet calificat**, **nume de fișier absolut** sau **rută**, indicând secvența de directoare care trebuie parcurse pentru a ajunge la acesta. Cei trei termeni sunt sinonimi.
- Toate **numele de fișiere complet calificate** încep cu directorul „`/`” și există o „`/`” între fiecare director sau fișier din numele fișierului. Primul „`/`” este directorul de nivel superior, iar celelalte „`/`” separă subdirectoarele succesive, până când ajungem la ultima intrare, care este numele fișierului propriu-zis. Cuvintele folosite aici pot fi confuze. Luați ca exemplu următorul **nume de fișier complet calificat**: „`/usr/share/keytables/us.map.gz`”. Cu toate acestea, oamenii se referă la numele său de bază „`us.map.gz`” ca fiind numele fișierului.
- Directorul rădăcină are o serie de ramuri, cum ar fi „`/etc/`” și „`/usr/`”. Aceste subdirectoare se ramifică la rândul lor în și mai multe subdirectoare, cum ar fi „`/etc/systemd/`” și „`/usr/local/`”. Ansamblul este denumit **arborele de directoare**. Puteți considera un nume de fișier absolut ca o rută de la baza arborelui („`/`”) până la capătul unei ramuri (un fișier). De asemenea, veți auzi oamenii vorbind despre arborele de directoare ca și cum ar fi un **arborele genealogic** care cuprinde toți descendenții direcți ai unei singure entități numite directorul rădăcină („`/`”): astfel, subdirectoarele au **părinți**, iar o rută arată ascendența completă a unui fișier. Există, de asemenea, rute relative care încep în alt loc decât directorul rădăcină. Trebuie să rețineți că directorul „`../`” se referă la directorul părinte. Această terminologie se aplică și altor structuri de tip director, cum ar fi structurile de date ierarhice.
- Nu există nicio componentă specială a numelui rutei de director care să corespundă unui dispozitiv fizic, cum ar fi discul dur. Acest lucru diferă de RT-11, CP/M, OpenVMS, MS-DOS, AmigaOS și Microsoft Windows, unde ruta conține un nume de dispozitiv, cum ar fi „`C:\`”. (Cu toate acestea, există intrări de director care se referă la dispozitive fizice ca parte a sistemului de fișiere normal. Consultați Secțiune 1.2.2.)

Notă
Deși **puteți** utiliza aproape orice litere sau simboluri într-un nume de fișier, în practică nu este recomandat să faceți acest lucru. Este mai bine să evitați orice caractere care au adesea semnificații speciale în linia de comandă, inclusiv spații, tabulatoare, linii noi și alte caractere speciale: `{ } ( ) [ ] ' ` " \ / > < | ; ! # & ^ * % @ $`. Dacă doriți să separați cuvintele dintr-un nume, alegeri bune sunt punctul, cratima și linia de subliniere. De asemenea, puteți scrie fiecare cuvânt cu majusculă la început, „`CaAcesta`”. Utilizatorii experimentați de Linux tind să evite spațiile în numele fișierelor.

Notă
Cuvântul „root” poate însemna fie „utilizatorul root”, fie „directorul root”. Contextul utilizării lor ar trebui să clarifice acest lucru.

Notă
Cuvântul **rută (path)** este folosit nu doar pentru **nume de fișier complet calificat**, așa cum s-a menționat mai sus, ci și pentru **ruta de căutare a comenzii**. Semnificația intenționată este de obicei clară din context.

Cele mai bune practici detaliate pentru ierarhia fișierelor sunt descrise în Filesystem Hierarchy Standard -- Standardul de ierarhie a sistemului de fișiere („`/usr/share/doc/debian-policy/fhs/fhs-2.3.txt.gz`” și hier(7)). Ar trebui să rețineți următoarele aspecte ca punct de plecare.

director	utilizarea directorului
`/`	directorul rădăcină
`/etc/`	fișierele de configurare la nivel de sistem
`/var/log/`	fișierele de jurnal ale sistemului
`/home/`	toate directoarele personale (acasă) pentru toți utilizatorii fără privilegii

Tabela 1.3: Lista directoarelor cheie și descrierea utilizării acestora

1.2.2 Elementele interne ale sistemului de fișiere

Urmând **tradiția Unix**, sistemul Debian GNU/Linux oferă sistemul de fișiere în care se află datele fizice de pe discurile dure și alte dispozitive de stocare, iar interacțiunea cu dispozitivele hardware, cum ar fi ecranele consolei și consolele seriale la distanță, este reprezentată într-o manieră unificată sub „`/dev/`”.

Fiecare fișier, director, conductă cu nume (o modalitate prin care două programe pot partaja date) sau dispozitiv fizic dintr-un sistem Debian GNU/Linux are o structură de date numită nod-i care descrie atributele asociate, cum ar fi utilizatorul care îl deține (proprietarul), grupul căruia îi aparține, ora ultimei accesări etc. Ideea de a reprezenta aproape totul în sistemul de fișiere a fost o inovație Unix, iar nucleele Linux moderne au dezvoltat această idee și mai mult. Acum, chiar și informațiile despre procesele care rulează în calculator pot fi găsite în sistemul de fișiere.

Această reprezentare abstractă și unificată a entităților fizice și a proceselor interne este foarte puternică, deoarece ne permite să folosim aceeași comandă pentru același tip de operație pe multe dispozitive total diferite. Este chiar posibil să se modifice modul de funcționare al nucleului prin scrierea de date în fișiere speciale care sunt legate de procesele în execuție.

Indicație
Dacă trebuie să identificați corespondența dintre arborele de fișiere și entitatea fizică, executați mount(8) fără argumente.

1.2.3 Permisiuni ale sistemului de fișiere

Permisiunile sistemului de fișiere ale unui sistem precum Unix sunt definite pentru trei categorii de utilizatori afectați.

- **Utilizatorul** care deține fișierul (**u**)
- Alți utilizatori din **grupul** căruia aparține fișierul (**g**)
- Toți **ceilalți** utilizatori (**o**), denumiți și „lumea” și „toată lumea”

Pentru fișier, fiecare permisiune corespunzătoare permite următoarele acțiuni.

- Permisiunea de **citire - (read)** (**r**) permite proprietarului să examineze conținutul fișierului.

- Permisiunea de **scriere - (write)** (**w**) permite proprietarului să modifice fișierul.
- Permisiunea de **executare - (execute)** (**x**) permite proprietarului să execute fișierul ca o comandă.

Pentru director, fiecare permisiune corespunzătoare permite următoarele acțiuni.

- Permisiunea de **citire - (read)** (**r**) permite proprietarului să afișeze conținutul directorului.
- Permisiunea de **scriere - (write)** (**w**) permite proprietarului să adauge sau să elimine fișiere din director.
- Permisiunea de **executare - (execute)** (**x**) permite proprietarului să acceseze fișierele din director.

Aici, permisiunea de **executare - (execute)** asupra unui director înseamnă nu numai să permită citirea fișierelor din acel director, ci și să permită vizualizarea atributelor acestora, cum ar fi dimensiunea și data modificării.

Instrumentul ls(1) este utilizat pentru a afișa informații privind permisiunile (și altele) pentru fișiere și directoare. Când este invocat cu opțiunea „`-l`”, afișează următoarele informații în ordinea indicată.

- **Tipul fișierului** (primul caracter)
- **Permisiunea** de acces la fișier (nouă caractere, compuse din câte trei caractere pentru utilizator, grup și altele, în această ordine)
- **Numărul de legături dure** către fișier
- Numele **utilizatorului** care deține fișierul
- Numele **grupului** căruia îi aparține fișierul
- **Dimensiunea** fișierului în caractere (octeți)
- **Data și ora** fișierului (mtime)
- **Numele** fișierului

caracter	**semnificație**
-	fișier normal
d	director
l	legătură simbolică
c	nod de dispozitiv de caractere
b	nod de dispozitiv de blocuri
p	conductă cu nume
s	soclu

Tabela 1.4: Lista descriptivă a primului caracter din ieșirea comenzii «`ls -l`»

chown(1) este utilizat din contul root pentru a schimba proprietarul fișierului. chgrp(1) este utilizat din contul proprietarului fișierului sau din contul root pentru a schimba grupul fișierului. chmod(1) este utilizat din contul proprietarului fișierului sau din contul root pentru a schimba permisiunile de acces la fișiere și directoare. Sintaxa de bază pentru manipularea unui fișier `foo` este următoarea.

```
# chown newowner foo
# chgrp newgroup foo
# chmod  [ugoa][+-=][rwxXst][,...] foo
```

De exemplu, puteți crea o structură de directoare care să aparțină utilizatorului `foo` și să fie partajată de grupul `bar` folosind următoarea comandă.

```
# cd /some/location/
# chown -R foo:bar .
# chmod -R ug+rwX,o=rX .
```

Există încă trei biți de permisiune specială.

- Bit-ul **set user ID** (**s** sau **S** în loc de **x** al utilizatorului)
- Bit-ul **set group ID** (**s** sau **S** în loc de **x** al grupului)
- Bit-ul **sticky** (**t** sau **T** în loc de **x** al celorlalți)

Aici, rezultatul comenzii „`ls -l`” pentru acești biți este **cu majuscule** dacă biții de execuție ascunși de aceste rezultate sunt **neactivați** .

Activarea bitului **set user ID** pe un fișier executabil permite utilizatorului să execute fișierul executabil cu ID-ul proprietarului fișierului (de exemplu **root**). În mod similar, activarea bitului **set group ID** pentru un fișier executabil permite utilizatorului să execute fișierul executabil cu ID-ul de grup al fișierului (de exemplu, **root**). Deoarece activările acestor biți pot genera riscuri de securitate, activarea lor necesită o atenție sporită.

Activarea bitului **set group ID** pentru un director activează schema de creare a fișierelor tip BSD în care toate fișierele create în director aparțin **grupului** directorului.

Activarea bitului **sticky** pe un director împiedică ștergerea unui fișier din director de către un utilizator care nu este proprietarul fișierului. Pentru a securiza conținutul unui fișier din directoare care pot fi scrise de toată lumea, cum ar fi „`/tmp`” sau din directoare care pot fi scrise de grup, nu trebuie doar să redefiniți permisiunea de **scriere** pentru fișier, ci și să activați bitul **sticky** pe director. În caz contrar, fișierul poate fi șters și poate fi creat un fișier nou cu același nume de către orice utilizator care are acces de scriere la director.

Iată câteva exemple interesante de permisiuni pentru fișiere.

```
$ ls -l /etc/passwd /etc/shadow /dev/ppp /usr/sbin/exim4
crw------T 1 root root   108, 0 Oct 16 20:57 /dev/ppp
-rw-r--r-- 1 root root     2761 Aug 30 10:38 /etc/passwd
-rw-r----- 1 root shadow   1695 Aug 30 10:38 /etc/shadow
-rwsr-xr-x 1 root root   973824 Sep 23 20:04 /usr/sbin/exim4
$ ls -ld /tmp /var/tmp /usr/local /var/mail /usr/src
drwxrwxrwt 14 root root  20480 Oct 16 21:25 /tmp
drwxrwsr-x 10 root staff  4096 Sep 29 22:50 /usr/local
drwxr-xr-x 10 root root   4096 Oct 11 00:28 /usr/src
drwxrwsr-x  2 root mail   4096 Oct 15 21:40 /var/mail
drwxrwxrwt  3 root root   4096 Oct 16 21:20 /var/tmp
```

Există un mod numeric alternativ pentru a descrie permisiunile fișierelor cu chmod(1). Acest mod numeric utilizează numere octale (în bază=8) cu 3 până la 4 cifre.

cifra	**semnificație**
prima cifră (opțională)	suma dintre bitul **set user ID** (=4), bitul **set group ID** (=2) și bitul **sticky** (=1)
a doua cifră	suma permisiunilor **citire** (=4), **scriere** (=2) și **executare** (=1) pentru **utilizator**
a treia cifră	la fel ca a doua cifră, dar pentru **grup**
a patra cifră	la fel ca a doua cifră, dar pentru **ceilalți**

Tabela 1.5: Modul numeric pentru permisiunile de fișiere în comenzile chmod(1)

Sună complicat, dar de fapt este destul de simplu. Dacă vă uitați la primele câteva coloane (2-10) din „`ls -l`” și le citiți ca o reprezentare binară (bază=2) a permisiunilor de fișier („-” fiind „0” și „rwx” fiind „1”), ultimele 3 cifre ale valorii numerice ale modului ar trebui să aibă sens ca o reprezentare octală (bază=8) a permisiunilor de fișier.

De exemplu, încercați următoarele

```
$ touch foo bar
$ chmod u=rw,go=r foo
$ chmod 644 bar
$ ls -l foo bar
-rw-r--r-- 1 penguin penguin 0 Oct 16 21:39 bar
-rw-r--r-- 1 penguin penguin 0 Oct 16 21:35 foo
```

Indicație
Dacă trebuie să accesați informațiile afișate de „`ls -l`” într-un script shell, trebuie să utilizați comenzi relevante, cum ar fi test(1), stat(1) și readlink(1). Se pot utiliza și comenzi încorporate în shell, cum ar fi „`[`” sau „`test`”.

1.2.4 Controlul permisiunilor pentru fișierele nou create: umask

Permisiunile aplicate unui fișier sau director nou creat sunt restricționate de comanda încorporată în shell `umask`. Consultați dash(1), bash(1) și builtins(7).

```
(file permissions) = (requested file permissions) & ~(umask value)
```

umask	permisiunile fișierelor create	permisiunile directoarelor create	utilizare
`0022`	`-rw-r--r--`	`-rwxr-xr-x`	poate fi scris numai de către utilizator
`0002`	`-rw-rw-r--`	`-rwxrwxr-x`	poate fi scris de grup

Tabela 1.6: Exemple de valori pentru **umask**

Sistemul Debian utilizează în mod implicit o schemă de grupuri private de utilizatori („user private group”: UPG). Un UPG este creat de fiecare dată când un nou utilizator este adăugat în sistem. Un UPG are același nume ca utilizatorul pentru care a fost creat, iar acel utilizator este singurul membru al UPG. Schema UPG face ca stabilirea umask la `0002` să fie sigură, deoarece fiecare utilizator are propriul grup privat; (în unele variante Unix, este destul de obișnuit să se configureze toți utilizatorii normali aparținând unui singur grup **`users`** și este o idee bună să se stabilească umask la `0022` pentru securitate în astfel de cazuri).

Indicație
Activați UPG introducând „`umask 002`” în fișierul `~/.bashrc`.

1.2.5 Permisiuni pentru grupuri de utilizatori (grup)



Avertisment
Asigurați-vă că salvați modificările nesalvate înainte de a reporni sau de a efectua acțiuni similare.

Puteți adăuga un utilizator `penguin` la un grup `bird` în doi pași:

- Modificați configurația grupului utilizând una dintre următoarele opțiuni:
  - Executați «`sudo usermod -aG bird penguin`».
  - Executați «`sudo adduser penguin bird`» (numai pe sistemele Debian tipice)
  - Executați «`sudo vigr`» pentru `/etc/group` și «`sudo vigr -s`» pentru `/etc/gshadow` pentru a adăuga `penguin` în linia pentru `bird`.
- Aplicați configurația utilizând una dintre următoarele opțiuni:
  - Reporniți sistemul și conectați-vă (cea mai bună opțiune)
  - Ieșire din sesiune prin intermediul meniului interfeței grafice și autentificare; (acest lucru poate să nu funcționeze în mediul grafic de birou modern).

Puteți elimina un utilizator `penguin` dintr-un grup `bird` în doi pași:

- Modificați configurația grupului utilizând una dintre următoarele opțiuni:
  - Executați «`sudo usermod -rG bird penguin`».
  - Executați «`sudo deluser penguin bird`» (numai pe sistemele Debian tipice)
  - Executați «`sudo vigr`» pentru `/etc/group` și «`sudo vigr -s`» pentru `/etc/gshadow` pentru a elimina `penguin` din linia pentru `bird`.
- Aplicați configurația utilizând una dintre următoarele opțiuni:
  - Reporniți sistemul și conectați-vă (cea mai bună opțiune)
  - Executați «`kill -TERM -1`» și efectuați câteva acțiuni de remediere, cum ar fi «`systemctl restart NetworkMana`
  - Deconectarea prin meniul interfeței grafice nu este o opțiune pentru mediul grafic de birou Gnome.

Orice încercare de repornire la cald este o înlocuire fragilă a repornirii la rece reale în cadrul sistemului de birou modern.

Notă
Alternativ, puteți adăuga dinamic utilizatori la grupuri în timpul procesului de autentificare, adăugând linia „`auth optional pam_group.so`” la „`/etc/pam.d/common-auth`” și configurând „`/etc/security/group.conf`”; (consultați Cap. 4).

Dispozitivele fizice sunt doar un alt tip de fișier în sistemul Debian. Dacă aveți probleme cu accesarea dispozitivelor precum unitatea flash USB și unitatea CD-ROM dintr-un cont de utilizator, ar trebui să adăugați utilizatorul respectiv ca membru al grupului relevant.

Unele grupuri importante (notabile) furnizate de sistem permit membrilor lor să acceseze anumite fișiere și dispozitive fără privilegii `root`.

grupul	**descrierea fișierelor și dispozitivelor accesibile**
`dialout`	acces complet și direct la porturile seriale („`/dev/ttyS[0-3]`”)
`dip`	acces limitat la porturile seriale pentru conexiunea **Dialup IP** la mașini de încredere
`cdrom`	unități CD-ROM, DVD+/-RW
`audio`	dispozitiv audio
`video`	dispozitiv video
`scanner`	scaner(e)
`adm`	jurnalele de monitorizare a sistemului
`staff`	unele directoare pentru lucrări administrative minore: „`/usr/local`”, „`/home`”

Tabela 1.7: Lista grupurilor notabile furnizate de sistem pentru accesul la fișiere

Indicație
Trebuie să aparțineți grupului `dialout` pentru a reconfigura modemul, a forma orice număr etc. Dar dacă `root` creează fișiere de configurare predefinite pentru partenerii (mașinile) de încredere în „`/etc/ppp/peers/`”, trebuie doar să aparțineți grupului `dip` pentru a crea o conexiune **Dialup IP** la acele mașini de încredere folosind comenzile pppd(8), pon(1) și poff(1).

Unele grupuri notabile furnizate de sistem permit membrilor lor să execute anumite comenzi fără privilegii `root`.

Pentru lista completă a utilizatorilor și grupurilor furnizate de sistem, consultați versiunea recentă a documentului „Utilizatori și grupuri” din „`/usr/share/doc/base-passwd/users-and-groups.html`” furnizat de pachetul `base-passwd`.

Consultați passwd(5), group(5), shadow(5), newgrp(1), vipw(8), vigr(8) și pam_group(8) pentru comenzile de gestionare a utilizatorilor și grupurilor din sistem.

grupul	comenzi disponibile
`sudo`	execută orice comandă cu privilegii de superutilizator
`lpadmin`	execută comenzi pentru a adăuga, modifica și elimina imprimante din bazele de date ale imprimantelor

Tabela 1.8: Lista grupurilor notabile furnizate de sistem pentru executarea anumitor comenzi

1.2.6 Marcaje de timp

Există trei tipuri de marcaje de timp pentru un fișier GNU/Linux.

tipul	semnificația (definiția istorică Unix)
mtime	momentul modificării fișierului (`ls -l`)
ctime	momentul modificării stării fișierului (`ls -lc`)
atime	ultima dată când a fost accesat fișierul (`ls -lu`)

Tabela 1.9: Lista tipurilor de marcaje de timp

Notă
ctime nu este ora creării fișierului.

Notă
Valoarea reală a **atime** pe sistemul GNU/Linux poate fi diferită de cea din definiția istorică Unix.

- Suprascrierea unui fișier modifică toate atributele **mtime**, **ctime** și **atime** ale fișierului.
- Schimbarea proprietarului sau a permisiunilor unui fișier modifică atributele **ctime** și **atime** ale fișierului.
- Citirea unui fișier modifică atributul **atime** al fișierului în sistemul Unix clasic.
- Citirea unui fișier modifică atributul **atime** al fișierului pe sistemul GNU/Linux dacă sistemul său de fișiere este montat cu „`strictatime`”.
- Citirea unui fișier pentru prima dată sau după o zi modifică atributul **atime** al fișierului pe sistemul GNU/Linux dacă sistemul său de fișiere este montat cu „`relatime`”; comportamentul implicit începând cu Linux 2.6.30).
- Citirea unui fișier nu modifică atributul **atime** al fișierului pe sistemul GNU/Linux dacă sistemul său de fișiere este montat cu „`noatime`”.

Notă
Opțiunile de montare „`noatime`” și „`relatime`” sunt introduse pentru a îmbunătăți performanța de citire a sistemului de fișiere în condiții normale de utilizare. Operația simplă de citire a fișierelor în cadrul opțiunii „`strictatime`” însoțește operația de scriere care consumă mult timp pentru a actualiza atributul **atime**. Dar atributul **atime** este rar utilizat, cu excepția fișierului mbox(5). Consultați mount(8).

Utilizați comanda touch(1) pentru a modifica marcajele temporale ale fișierelor existente.

Pentru marcajele temporale, comanda `ls` afișează șiruri traduse în limbi diferite de engleză („`fr_FR.UTF-8`”).

```
$ LANG=C  ls -l foo
-rw-rw-r-- 1 penguin penguin 0 Oct 16 21:35 foo
$ LANG=en_US.UTF-8  ls -l foo
-rw-rw-r-- 1 penguin penguin 0 Oct 16 21:35 foo
$ LANG=fr_FR.UTF-8  ls -l foo
-rw-rw-r-- 1 penguin penguin 0 oct. 16 21:35 foo
```

Indicație
Consultați Secțiune 9.3.4 pentru a personaliza ieșirea comenzii «`ls -l`».

1.2.7 Legături

Există două metode de asociere a unui fișier „`foo`” cu un nume de fișier diferit „`bar`”.

- Hard link
  - Nume duplicat pentru un fișier existent
  - ”`ln foo bar`”
- Legătură simbolică sau symlink
  - Fișier special care indică un alt fișier după nume
  - ”`ln -s foo bar`”

A se vedea exemplul următor pentru modificările numărului de legături și diferențele subtile în rezultatul comenzii «`rm`».

```
$ umask 002
$ echo "Original Content" > foo
$ ls -li foo
1449840 -rw-rw-r-- 1 penguin penguin 17 Oct 16 21:42 foo
$ ln foo bar     # hard link
$ ln -s foo baz  # symlink
$ ls -li foo bar baz
1449840 -rw-rw-r-- 2 penguin penguin 17 Oct 16 21:42 bar
1450180 lrwxrwxrwx 1 penguin penguin  3 Oct 16 21:47 baz -> foo
1449840 -rw-rw-r-- 2 penguin penguin 17 Oct 16 21:42 foo
$ rm foo
$ echo "New Content" > foo
$ ls -li foo bar baz
1449840 -rw-rw-r-- 1 penguin penguin 17 Oct 16 21:42 bar
1450180 lrwxrwxrwx 1 penguin penguin  3 Oct 16 21:47 baz -> foo
1450183 -rw-rw-r-- 1 penguin penguin 12 Oct 16 21:48 foo
$ cat bar
Original Content
$ cat baz
New Content
```

Legătura dură poate fi creată în cadrul aceluiași sistem de fișiere și are același număr de nod-i, pe care îl afișează opțiunea „`-i`” cu ls(1) .

Legătura simbolică are întotdeauna permisiuni nominale de acces la fișiere de tipul „`rwxrwxrwx`”, așa cum se arată în exemplul de mai sus, permisiunile efective de acces fiind dictate de permisiunile fișierului către care face trimitere.



Atenție
În general, este o idee bună să nu creați legături simbolice sau legături dure complicate, cu excepția cazului în care aveți un motiv foarte bun. Acest lucru poate provoca probleme în care combinația logică a legăturilor simbolice duce la bucle în sistemul de fișiere.

Notă
În general, este preferabil să utilizați legături simbolice în locul legăturilor dure, cu excepția cazului în care aveți un motiv întemeiat pentru a utiliza o legătură dură.

Directorul „`.`” conține legături către directorul în care apare, astfel încât numărul de legături ale oricărui director nou începe de la 2. Directorul „`..`” conține legături către directorul părinte, astfel încât numărul de legături ale directorului crește odată cu adăugarea de noi subdirectoare.

Dacă tocmai treceți de la Windows la Linux, veți observa rapid cât de bine concepută este legarea fișierelor în Unix, în comparație cu echivalentul Windows al „scurtăturilor”. Deoarece este implementată în sistemul de fișiere, aplicațiile nu pot vedea nicio diferență între un fișier legat și original. În cazul legăturilor dure, nu există nicio diferență.

1.2.8 Conducte cu nume (FIFO)

O conductă cu nume este un fișier care funcționează ca o conductă. Introduceți ceva în fișier și acesta iese la celălalt capăt. De aceea se numește FIFO, sau First-In-First-Out (primul intrat, primul ieșit): primul lucru pe care îl introduceți în conductă este primul lucru care iese la celălalt capăt.

Dacă scrieți într-o conductă cu nume, procesul care scrie în conductă nu se termină până când informațiile scrise nu sunt citite din conductă. Dacă citiți dintr-o conductă cu nume, procesul de citire așteaptă până când nu mai este nimic de citit înainte de a se termina. Dimensiunea conductei este întotdeauna zero --- nu stochează date, ci doar leagă două procese, similar funcționalității oferite de sintaxa shell „`|`”. Cu toate acestea, deoarece această conductă are un nume, cele două procese nu trebuie să se afle pe aceeași linie de comandă și nici măcar să fie rulate de același utilizator. Conductele au reprezentat o inovație foarte influentă a sistemului Unix.

De exemplu, încercați următoarele

```
$ cd; mkfifo mypipe
$ echo "hello" >mypipe & # put into background
[1] 8022
$ ls -l mypipe
prw-rw-r-- 1 penguin penguin 0 Oct 16 21:49 mypipe
$ cat mypipe
hello
[1]+  Done                    echo "hello" >mypipe
$ ls mypipe
mypipe
$ rm mypipe
```

1.2.9 Socluri

Soclurile sunt utilizate pe scară largă de toate comunicațiile pe Internet, bazele de date și sistemul de operare în sine. Este similar cu conducta cu nume (FIFO) și permite proceselor să schimbe informații chiar și între calculatoare diferite. Pentru soclu, aceste procese nu trebuie să ruleze în același timp și nici să ruleze ca procese secundare ale aceluiași proces ancestral. Acesta este punctul final pentru comunicarea între procese (IPC). Schimbul de informații poate avea loc prin rețea între diferite gazde. Cele mai comune două sunt soclul Internet și soclul de domeniu Unix.

Indicație
«`ss -t -u -a`» (din pachetul `iproute2`) oferă o imagine de ansamblu foarte utilă asupra soclurilor deschise pe un anumit sistem.

1.2.10 Fișiere de dispozitive

Fișierele de dispozitive se referă la dispozitivele fizice sau virtuale din sistemul dvs., cum ar fi discul dur, placa video, ecranul sau tastatura. Un exemplu de dispozitiv virtual este consola, reprezentată de „`/dev/console`”.

Există 2 tipuri de fișiere de dispozitiv.

- **Dispozitiv de caractere**
  - accesat de câte un caracter pe rând
  - 1 caracter = 1 octet
  - De exemplu, tastatura, portul serial, …
- **Dispozitiv de blocuri**
  - accesat în unități mai mari numite blocuri
  - 1 bloc > 1 octet
  - De exemplu, discul dur, …

Puteți citi și scrie fișiere de dispozitiv, deși fișierul poate conține date binare care pot fi incomprehensibile pentru oameni. Scrierea datelor direct în aceste fișiere este uneori utilă pentru depanarea conexiunilor hardware. De exemplu, puteți descărca un fișier text pe dispozitivul de imprimare „`/dev/lp0`” sau puteți trimite comenzi modem către portul serial corespunzător „`/dev/ttyS0`”. Dar, dacă nu se face cu atenție, acest lucru poate provoca o catastrofă majoră. Așadar, fiți precauți.

Notă
Pentru accesul normal la o imprimantă, utilizați lp(1).

Numărul nodului dispozitivului este afișat prin executarea comenzii ls(1), după cum urmează.

```
$ ls -l /dev/sda /dev/sr0 /dev/ttyS0 /dev/zero
brw-rw---T  1 root disk     8,  0 Oct 16 20:57 /dev/sda
brw-rw---T+ 1 root cdrom   11,  0 Oct 16 21:53 /dev/sr0
crw-rw---T  1 root dialout  4, 64 Oct 16 20:57 /dev/ttyS0
crw-rw-rw-  1 root root     1,  5 Oct 16 20:57 /dev/zero
```

- „`/dev/sda`” are numărul major al dispozitivului 8 și numărul minor al dispozitivului 0. Acesta este accesibil în mod citire/scriere pentru utilizatorii care aparțin grupului `disk`.
- „`/dev/sr0`” are numărul major al dispozitivului 11 și numărul minor al dispozitivului 0. Acesta este accesibil în mod citire/scriere pentru utilizatorii care aparțin grupului `cdrom`.
- „`/dev/ttyS0`” are numărul major al dispozitivului 4 și numărul minor al dispozitivului 64. Acesta este accesibil în mod citire/scriere pentru utilizatorii care aparțin grupului `dialout`.
- „`/dev/zero`” are numărul major al dispozitivului 1 și numărul minor al dispozitivului 5. Acesta poate fi citit/scris de către oricine.

În sistemul Linux modern, sistemul de fișiere din „`/dev/`” este completat automat de mecanismul udev(7).

1.2.11 Fișiere de dispozitive speciale

Există câteva fișiere speciale pentru dispozitive.

Acestea sunt utilizate frecvent împreună cu redirecționarea shell-ului (a se vedea Secțiune 1.5.8).

fișierul de dispozitiv	acțiunea	descrierea răspunsului
`/dev/null`	citire	returnează „caracterul de sfârșit de fișier (EOF)”
`/dev/null`	scriere	nu returnează nimic (o groapă fără fund pentru stocarea datelor)
`/dev/zero`	citire	returnează „caracterul `\0` (NUL)” (nu este același cu numărul zero ASCII)
`/dev/random`	citire	returnează caractere aleatorii dintr-un generator de numere aleatorii reale, oferind entropie reală (lent)
`/dev/urandom`	citire	returnează caractere aleatorii dintr-un generator de numere pseudoaleatorii securizat criptografic
`/dev/full`	scriere	returnează eroarea de disc plin (ENOSPC)

Tabela 1.10: Lista fișierelor speciale de dispozitive

1.2.12 procfs și sysfs

procfs și sysfs montate pe „`/proc`” și „`/sys`” sunt pseudo-sisteme de fișiere și expun structurile de date interne ale nucleului către spațiul utilizatorului. Cu alte cuvinte, aceste intrări sunt virtuale, ceea ce înseamnă că acționează ca o fereastră convenabilă către funcționarea sistemului de operare.

Directorul „`/proc`” conține (printre altele) un subdirector pentru fiecare proces care rulează în sistem, numit după ID-ul procesului (PID). Utilitarele de sistem care accesează informații despre procese, cum ar fi ps(1), obțin informațiile din această structură de directoare.

Directoarele din „`/proc/sys/`” conțin interfețe pentru modificarea anumitor parametri ai nucleului în timpul rulări; (puteți face același lucru prin comanda specializată sysctl(8) sau prin fișierul său de preîncărcare/configurare „`/etc/sysctl.d/`

Oamenii intră adesea în panică când observă un anumit fișier - „`/proc/kcore`” - care este, în general, foarte mare. Acesta este (mai mult sau mai puțin) o copie a conținutului memoriei calculatorului dvs. Este utilizat pentru depanarea nucleului. Este un fișier virtual care indică memoria calculatorului, așa că nu vă faceți griji în privința dimensiunii sale.

Directoarele din „`/sys`” conține structuri de date exportate ale nucleului, atributele acestora și legăturile dintre ele. De asemenea, conține interfețe pentru modificarea anumitor parametri ai nucleului în timpul rulării.

A se vedea „`proc.txt(.gz)`”, „`sysfs.txt(.gz)`” și alte documente conexe din documentația nucleului Linux („`/usr/share/doc/linux-doc-*/Documentation/filesystems/*`”) furnizate de pachetul `linux-doc-*`.

1.2.13 tmpfs

tmpfs este un sistem de fișiere temporar care păstrează toate fișierele în memoria virtuală. Datele din tmpfs din cache-ul paginii din memorie pot fi transferate în spațiul de interschimb (swap) de pe disc, după cum este necesar.

Directorul „`/run`” este montat ca tmpfs în procesul de pornire inițială. Acest lucru permite scrierea în el chiar și atunci când directorul „`/`” este montat ca fiind numai pentru citire. Aceasta este noua locație pentru stocarea fișierelor de stare tranzitorie și înlocuiește mai multe locații descrise în Filesystem Hierarchy Standard versiunea 2.3:

- „`/var/run`” → „`/run`”
- „`/var/lock`” → „`/run/lock`”
- ”`/dev/shm`” → ”`/run/shm`”

A se vedea „`tmpfs.txt(.gz)`” în documentația nucleului Linux („`/usr/share/doc/linux-doc-*/Documentation/f`
furnizată de pachetul `linux-doc-*`.

1.3 Midnight Commander (MC)

Midnight Commander (MC) este un „briceag elvețian” GNU pentru consola Linux și alte medii terminale. Acesta oferă începătorilor o experiență de consolă bazată pe meniuri, mult mai ușor de învățat decât comenzile standard Unix.

Este posibil să fie necesar să instalați pachetul Midnight Commander, denumit „`mc`”, urmând pașii de mai jos.

```
$ sudo apt-get install mc
```

Utilizați comanda mc(1) pentru a explora sistemul Debian. Acesta este cel mai bun mod de a învăța. Explorați câteva locații interesante folosind tastele săgeții și tasta Enter.

- „`/etc`” și subdirectoarele acestuia
- „`/var/log`” și subdirectoarele acestuia
- „`/usr/share/doc`” și subdirectoarele acestuia
- „`/usr/sbin`” și „`/usr/bin`”

1.3.1 Personalizarea MC

Pentru a face ca MC să schimbe directorul de lucru la ieșire și după aceea a executa `cd` către director, sugerez să modificați „`~/.bashrc`” pentru a include un script furnizat de pachetul `mc`.

```
. /usr/lib/mc/mc.sh
```

Vedeți mc(1) (sub opțiunea „`-P`”) pentru a afla motivul. Dacă nu înțelegeți exact despre ce vorbesc aici, puteți face acest lucru mai târziu.

1.3.2 Lansarea MC

MC poate fi lansat astfel.

```
$ mc
```

MC se ocupă de toate operațiile cu fișiere prin intermediul meniului său, necesitând un efort minim din partea utilizatorului. Apăsați tasta F1 pentru a afișa ecranul de ajutor. Puteți utiliza MC apăsând tastele cursorului și tastele de funcții.

Notă
În unele console, cum ar fi gnome-terminal(1), apăsările tastelor de funcții pot fi interceptate de programul consolei. Puteți dezactiva aceste funcții în meniul „Preferințe” → „General” și „Comenzi rapide” pentru `gnome-terminal`.

Dacă întâmpinați probleme de codificare a caracterelor care afișează caractere neinteligibile, adăugarea „`-a`” la linia de comandă MC poate ajuta la prevenirea problemelor.

Dacă acest lucru nu rezolvă problemele de afișare cu MC, consultați Secțiune 8.3.1.

1.3.3 Gestionarul de fișiere din MC

Implicit sunt două panouri de directoare care conțin liste de fișiere. Un alt mod util este configurarea ferestrei din dreapta pe „informații” pentru a vedea informații despre privilegiile de acces la fișiere etc. În continuare sunt prezentate câteva comenzi esențiale. Cu demonul gpm(8) în funcțiune, se poate utiliza mouse-ul și pe consolele de caractere Linux. (Asigurați-vă că apăsați tasta Shift pentru a obține comportamentul normal al funcțiilor de tăiere și lipire în MC.)

tasta	comanda/funcția asociată
F1	meniul de ajutor
F3	vizorul de fișiere intern
F4	editorul intern
F9	activează meniul derulant
F10	ieșire din Midnight Commander
Tab	deplasare între două ferestre
Insert sau Ctrl-T	marchează fișierul pentru o operație cu mai multe fișiere, cum ar fi copierea
Del	șterge fișierul (aveți grijă --- configurați MC în modul de ștergere sigură)
Tastele cursor	auto-explicative

Tabela 1.11: Tastele de comenzi rapide ale MC

1.3.4 Trucuri din linia de comandă în MC

- Comanda cd modifică directorul afișat pe ecranul selectat.
- Ctrl-Enter sau Alt-Enter copiază un nume de fișier în linia de comandă. Utilizați această comandă împreună cu comenzile cp(1) și mv(1) și cu editarea liniei de comandă.
- Alt-Tab afișează opțiunile de extindere a numelor de fișiere din shell.
- Se poate specifica directorul de pornire pentru ambele ferestre ca argumente pentru MC; de exemplu, „mc /etc /root”.
- Esc + n-key → Fn (adică, Esc + 1 → F1, etc.; Esc + 0 → F10)
- Apăsarea tastei Esc înaintea tastei are același efect ca și apăsarea simultană a tastelor Alt și ; Adică, tastați Esc + c pentru Alt-C. Esc se numește meta-tastă și uneori este notată ca „M-”.

1.3.5 Editorul intern din MC

Editorul intern are o schemă interesantă de copiere și lipire. Apăsarea tastei F3 marchează începutul unei selecții, o a doua apăsare a tastei F3 marchează sfârșitul selecției și evidențiază selecția. Apoi puteți muta cursorul. Dacă apăsați tasta «F6», zona selectată este mutată la locația cursorului. Dacă apăsați tasta «F5», zona selectată este copiată și inserată la locația cursorului. F2 salvează fișierul. F10 vă scoate din program. Majoritatea tastelor cursorului funcționează intuitiv.

Acest editor poate fi pornit direct pe un fișier folosind una dintre următoarele comenzi.

```
$ mc -e filename_to_edit
```

```
$ mcedit filename_to_edit
```

Acesta nu este un editor cu ferestre multiple, dar se pot utiliza mai multe console Linux pentru a obține același efect. Pentru a copia între ferestre, utilizați tastele Alt-F*n* pentru a comuta între consolele virtuale și utilizați „File→Insert file” (Fișier→Inserați fișier) sau „File→Copy to file” (Fișier→Copiați în fișier) pentru a muta o parte dintr-un fișier într-un alt fișier.

Acest editor intern poate fi înlocuit cu orice editor extern la alegere.

De asemenea, multe programe utilizează variabilele de mediu „$EDITOR” sau „$VISUAL” pentru a decide ce editor să utilizeze. Dacă nu vă simțiți confortabil cu vim(1) sau nano(1) la început, puteți defini aceste variabile ca „mcedit” adăugând următoarele linii în „~/.bashrc”.

```
export EDITOR=mcedit
export VISUAL=mcedit
```

Recomand să le definiți ca „vim”, dacă este posibil.

Dacă nu vă simțiți confortabil cu vim(1), puteți continua să utilizați mcedit(1) pentru majoritatea sarcinilor de întreținere a sistemului.

1.3.6 Vizorul intern din MC

MC este un vizor foarte inteligent. Este un instrument excelent pentru căutarea cuvintelor în documente. Îl folosesc întotdeauna pentru fișierele din directorul „/usr/share/doc”. Este cea mai rapidă modalitate de a răsfoi cantități mari de informații despre Linux. Acest vizor poate fi pornit direct folosind una dintre următoarele comenzi.

```
$ mc -v path/to/filename_to_view
```

```
$ mcview path/to/filename_to_view
```

1.3.7 Funcții de pornire automată ale MC

Apăsați tasta «Enter» pe un fișier, iar programul corespunzător va gestiona conținutul fișierului (consultați Secțiune 9.4.11). Aceasta este o funcție MC foarte convenabilă.

tipul fișierului	**reacția la apăsarea tastei «Enter»**
fișier executabil	execută comanda
fișier de pagină de manual	direcționează conținutul către software-ul de vizualizare
fișier html	direcționează conținutul către navigatorul web
fișiere „*.tar.gz” și „*.deb”	răsfoiește conținutul său ca și cum ar fi un subdirector

Tabela 1.12: Reacția la apăsarea tastei «Enter» în MC

Pentru ca aceste funcții de vizualizare și fișiere virtuale să funcționeze, fișierele care pot fi vizualizate nu trebuie definite ca fiind executabile. Modificați-le starea folosind chmod(1) sau prin meniul fișier MC.

1.3.8 Sistemul de fișiere virtual al MC

MC poate fi utilizat pentru a accesa fișiere prin Internet. Accesați meniul apăsând F9, „Enter” și „h” pentru a activa sistemul de fișiere Shell. Introduceți o adresă URL în formatul „sh://[utilizator@]mașină[:opțiuni]/[director-l
care recuperează un director la distanță care apare ca unul local utilizând ssh.

1.4 Mediul de lucru de bază de tip Unix

Deși MC vă permite să faceți aproape totul, este foarte important să învățați cum să utilizați instrumentele liniei de comandă invocate din promptul shell și să vă familiarizați cu mediul de lucru de tip Unix.

1.4.1 Shell-ul de autentificare

Deoarece shell-ul de autentificare poate fi utilizat de unele programe de inițializare a sistemului, este prudent să îl păstrați ca bash(1) și să evitați schimbarea shell-ului de autentificare cu chsh(1).

Dacă doriți să utilizați un prompt interactiv diferit, definiți-l din configurația emulatorului de terminal cu interfață grafică sau porniți-l din ~/.bashrc, de exemplu, introducând „exec /usr/bin/zsh -i -l” sau „exec /usr/bin/fish -i -l”.

pachet	popcon(popularitate)	dimensiune	shell POSIX	descriere
bash	V:886, I:999	7308	Da	Bash: GNU Bourne Again SHell (standard de facto)
bash-completion	V:37, I:962	1952	N/D	completare programabilă pentru shell-ul bash
dash	V:929, I:999	207	Da	Debian Almquist Shell, bun pentru scripturi shell
zsh	V:39, I:68	2571	Da	Z shell: shell-ul standard cu numeroase îmbunătățiri
tcsh	V:4, I:14	1366	Nu	TENEX C Shell: o versiune îmbunătățită a Berkeley csh
mksh	V:4.7, I:7.5	7713	Da	O versiune a Korn shell
csh	V:1.0, I:5.2	348	Nu	OpenBSD C Shell, o versiune a Berkeley csh
sash	V:0.3, I:4.7	1245	Da	Shell autonom cu comenzi încorporate (nu este destinat pentru standardul „/usr/bin/sh”)
ksh	V:0.3, I:7.9	65	Da	Versiunea reală, AT&T a Korn shell
rc	V:0.07, I:0.70	182	Nu	Implementarea AT&T Plan 9 rc shell
posh	V:0.02, I:0.22	195	Da	SHell obișnuit conform cu politica (derivat din pdksh)

Tabela 1.13: Lista programelor shell

Indicație
Deși shell-urile de tip POSIX au aceeași sintaxă de bază, ele pot diferi în ceea ce privește comportamentul în aspecte fundamentale, precum variabilele shell și expansiunile globale (cu caractere joker). Vă rugăm să consultați documentația acestora pentru detalii.

În acest capitol al tutorialului, shell-ul interactiv înseamnă întotdeauna bash.

1.4.2 Personalizarea bash

Puteți personaliza comportamentul bash(1) prin „~/.bashrc”.

De exemplu, încercați următoarele.

```
# enable bash-completion
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

# CD upon exiting MC
. /usr/lib/mc/mc.sh

# set CDPATH to a good one
CDPATH=.:/usr/share/doc:~:~/Desktop:~
export CDPATH

PATH="${PATH+$PATH:}/usr/sbin:/sbin"
# set PATH so it includes user's private bin if it exists
if [ -d ~/bin ] ; then
```

```
  PATH="~/bin${PATH+:$PATH}"
fi
export PATH

EDITOR=vim
export EDITOR
```

Indicație
Puteți găsi mai multe sfaturi de personalizare `bash`, cum ar fi Secțiune 9.3.6, în Cap. 9.

Indicație
Pachetul `bash-completion` permite completarea programabilă pentru `bash`.

1.4.3 Combinații speciale de taste

În mediul de tip Unix există câteva combinații de taste care au semnificații speciale. Rețineți că pe o consolă Linux normală, numai tastele `Ctrl` și `Alt` din partea stângă funcționează așa cum vă așteptați. Iată câteva combinații de taste importante de reținut.

tasta	descrierea acțiunii efectuate
`Ctrl-U`	șterge linia din fața cursorului
`Ctrl-H`	șterge un caracter înaintea cursorului
`Ctrl-D`	termină introducerea („ieșire din shell” dacă utilizați shell)
`Ctrl-C`	termină un program în execuție
`Ctrl-Z`	oprește temporar programul mutându-l în fundal
`Ctrl-S`	oprește afișarea pe ecran
`Ctrl-Q`	reactivează ieșirea pe ecran
`Ctrl-Alt-Del`	repornește/oprește sistemul, a se vedea inittab(5)
`Tasta Alt stânga` (opțional, `Tasta Windows`)	meta-tastă pentru Emacs și interfețe similare
`Up-arrow`	pornește căutarea istoricului comenzilor sub `bash`
`Ctrl-R`	pornește căutarea incrementală în istoricul comenzilor sub `bash`
`Tab`	completează introducerea numelui fișierului în linia de comandă sub `bash`
`Ctrl-V Tab`	introduce `Tab` fără extindere în linia de comandă sub `bash`

Tabela 1.14: Lista tastelor de comenzi rapide pentru bash

Indicație
Funcția terminalului `Ctrl-S` poate fi dezactivată folosind stty(1).

1.4.4 Operații cu mouse-ul

Operațiile cu mouse-ul pentru text în sistemul Debian combină două stiluri cu câteva modificări:

- Operații tradiționale ale mouse-ului în stil Unix:
  - utilizează 3 butoane (clic)

  - utilizare PRIMARĂ
  - utilizată de aplicații X precum `xterm` și aplicații text în consola Linux
- Operații moderne cu mouse-ul în stil de interfață grafică:
  - utilizează 2 butoane (glisare + clic)
  - utilizare PRIMARĂ și CLIPBOARD
  - utilizată în aplicații cu interfață grafică modernă, cum ar fi `gnome-terminal`

acțiunea	**răspunsul**
Clic stânga și glisare cu mouse-ul	selectează intervalul ca selecție PRIMARĂ
Clic stânga	selectează începutul intervalului pentru selecția PRIMARĂ
Clic dreapta (tradițional)	selectează sfârșitul intervalului pentru selecția PRIMARĂ
Clic dreapta (modern)	meniu dependent de context (tăiere/copiere/lipire)
Clic pe butonul din mijloc sau Shift-`Ins`	inserează selecția PRIMARĂ la poziția cursorului
`Ctrl-X`	taie selecția PRIMARĂ în CLIPBOARD
`Ctrl-C` (`Shift-Ctrl-C` în terminal)	copiază selecția PRIMARĂ în CLIPBOARD
`Ctrl-V`	lipește conținutul CLIPBOARDULUI la poziția cursorului

Tabela 1.15: Lista operațiilor mouse-ului și acțiunile tastelor asociate în Debian

Aici, selecția PRIMARĂ este intervalul de text evidențiat. În cadrul programului de terminal, se utilizează în schimb `Shift-Ctrl-C` pentru a evita închiderea unui program în execuție.

Rotița centrală a mouse-ului modern cu rotiță este considerată butonul din mijloc al mouse-ului și poate fi utilizată pentru clic-mijloc. Apăsarea simultană a butoanelor stânga și dreapta ale mouse-ului servește ca clic-mijloc în cazul unui mouse cu 2 butoane.

Pentru a utiliza un mouse în consolele de caractere Linux, trebuie să aveți gpm(8) rulând ca demon.

1.4.5 Paginatorul

Comanda less(1) este un paginator îmbunătățit (permite navigarea prin conținutul fișierelor). Citește fișierul specificat prin argumentul comenzii sau prin intrarea standard. Apăsați „h” dacă aveți nevoie de ajutor în timp ce navigați cu comanda `less`. Poate face mult mai mult decât more(1) și poate fi supraalimentat prin executarea „`eval $(lesspipe)`” sau „`eval $(lessfile)`” în scriptul de pornire al shell-ului. Pentru mai multe informații, consultați „`/usr/share/doc/less/LESSOPEN`”. Opțiunea „`-R`” permite afișarea caracterelor brute și activează secvențele de eludare a culorilor ANSI. Consultați less(1).

Indicație
În comanda `less`, tastați „h” pentru a afișa ecranul de ajutor, tastați „`/`” sau „?” pentru a căuta un șir de caractere și tastați „`-i`” pentru a schimba respectarea/ignorarea diferenței dintre majuscule și minuscule.

1.4.6 Editorul de text

Ar trebui să deveniți expert într-una dintre variantele programelor Vim sau Emacs, care sunt populare în sistemele de tip Unix.

Cred că este bine să vă obișnuiți cu comenzile Vim, deoarece editorul Vi este mereu prezent în lumea Linux/Unix; (de fapt, programele originale `vi` sau noul `nvi` se găsesc peste tot. Am ales Vim pentru începători, deoarece oferă ajutor prin tasta `F1`, fiind suficient de similar și mai avansat).

Dacă ați ales Emacs sau XEmacs ca editor, aceasta este o altă alegere bună, în special pentru programare. Emacs are o mulțime de alte caracteristici, inclusiv funcționarea ca cititor de știri, editor de directoare, program de poștă

electronică etc. Când este utilizat pentru programare sau editarea scripturilor shell, recunoaște în mod inteligent formatul asupra căruia lucrați și încearcă să vă ofere asistență. Unii oameni susțin că singurul program de care au nevoie pe Linux este Emacs. Zece minute de învățare a Emacs acum pot economisi ore mai târziu. Este foarte recomandat să aveți manualul GNU Emacs ca referință atunci când învățați Emacs.

Toate aceste programe vin de obicei cu un program de tutoriat pentru a le învăța prin practică. Porniți Vim tastând „`vim`” și apăsați tasta F1. Ar trebui să citiți cel puțin primele 35 de linii. Apoi urmați cursul de formare pe internet mutând cursorul la „`|tutor|`” și apăsând `Ctrl-]`.

Notă
Editorii buni, precum Vim și Emacs, pot gestiona corect textele codificate în UTF-8 și alte codificări exotice. Este o idee bună să utilizați mediul de interfață grafică cu configurația regională pentru text, UTF-8 și să instalați programele și fonturile necesare în acesta. Editorii au opțiuni pentru a configura codificarea fișierelor independent de mediul de interfață grafică. Vă rugăm să consultați documentația acestora referitoare la textul multi-octet.

1.4.7 Definirea unui editor de text implicit

Debian vine cu o serie de editori diferiți. Vă recomandăm să instalați pachetul `vim`, așa cum s-a menționat mai sus.

Debian oferă acces unificat la editorul implicit al sistemului prin comanda „`/usr/bin/editor`”, astfel încât alte programe (de exemplu, reportbug(1)) să îl poată invoca. Îl puteți modifica după cum urmează.

```
$ sudo update-alternatives --config editor
```

Alegerea „`/usr/bin/vim.basic`” în locul „`/usr/bin/vim.tiny`” este recomandarea mea pentru începători, deoarece oferă suport pentru evidențierea sintaxei.

Indicație
Multe programe utilizează variabilele de mediu „`$EDITOR`” sau „`$VISUAL`” pentru a decide ce editor să utilizeze (consultați Secțiune 1.3.5 și Secțiune 9.4.11). Pentru consecvență în sistemul Debian, definiți aceste variabile la „`/usr/bin/editor`”; (în trecut, „`$EDITOR`” era „`ed`” și „`$VISUAL`” era „`vi`”).

1.4.8 Utilizarea editorului vim

Versiunea recentă vim(1) pornește în mod implicit cu opțiunea „`nocompatible`” și intră în modul `NORMAL`.[1]

Vă rugăm să utilizați programul „`vimtutor`” pentru a învăța `vim` printr-un curs tutorial interactiv.

Programul `vim` își modifică comportamentul în funcție de tastele apăsate, pe baza **modului**. Tastarea în tampon se face în principal în modul `INSERARE` și în modul `ÎNLOCUIRE`. Deplasarea cursorului se face în principal în modul `NORMAL`. Selecția interactivă se face în modul `VIZUAL` . Tastarea „`:`” în modul `NORMAL` schimbă **modul** în modul `Ex` . Modul `Ex` acceptă comenzi.

Indicație
Vim vine cu pachetul **Netrw**. Netrw permite citirea și scrierea fișierelor, navigarea în directoare prin rețea și navigarea locală! Încercați Netrw cu „`vim .`” (un punct ca argument) și citiți manualul său la „`:help netrw`”.

Pentru configurarea avansată a `vim`, consultați Secțiune 9.2.

[1] Chiar și versiunea mai veche `vim` poate porni în modul „`nocompatible`” prin opțiunea „`-N`”.

modul	combinația de taste	acțiunea
NORMAL	`:help|only`	afișează fișierul de ajutor
NORMAL	`:e filename.ext`	deschide un nou spațiu tampon în memorie pentru a edita `filename.ext`
NORMAL	`:w`	suprascrie fișierul original cu conținutul tamponului curent
NORMAL	`:w filename.ext`	scrie tamponul curent în `filename.ext`
NORMAL	`:q`	ieșire din `vim`
NORMAL	`:q!`	forțează ieșirea din `vim`
NORMAL	`:only`	închide toate celelalte ferestre deschise
NORMAL	`:set nocompatible?`	verifică dacă `vim` se află în modul `nocompatible`
NORMAL	`:set nocompatible`	stabilește `vim` în modul de operare `nocompatible`
NORMAL	`i`	intră în modul `INSERARE`
NORMAL	`R`	intră în modul `ÎNLOCUIRE`
NORMAL	`v`	intră în modul `VIZUAL`
NORMAL	`V`	intră în modul `VIZUAL` pe linii
NORMAL	Ctrl-V	intrați în modul `VISUAL` pe blocuri
cu excepția `TERMINAL-JOB`	ESC-key	intră în modul `NORMAL`
NORMAL	`:term`	intră în modul `TERMINAL-JOB`
`TERMINAL-NORMAL`	`i`	intră în modul `TERMINAL-JOB`
`TERMINAL-JOB`	Ctrl-`W N` (sau Ctrl-`\` Ctrl-`N`)	intră în modul `TERMINAL-NORMAL`
`TERMINAL-JOB`	Ctrl-`W :`	intră în modul `Ex` din modul `TERMINAL-NORMAL`

Tabela 1.16: Lista combinațiilor de taste de bază din Vim

1.4.9 Înregistrarea activităților shell-ului

Rezultatul comenzii shell poate să nu mai apară pe ecran și să se piardă definitiv. Este recomandat să înregistrați activitățile shell în fișier pentru a le putea revizui ulterior. Acest tip de înregistrare este esențial atunci când efectuați orice sarcini de administrare a sistemului.

Indicație
Noul Vim (versiunea>=8.2) poate fi utilizat pentru a înregistra activitățile shell-ului în mod curat folosind modul `TERMINAL-JOB`. A se vedea Secțiune 1.4.8.

Metoda de bază pentru înregistrarea activității shell-ului este rularea acestuia sub script(1).

De exemplu, încercați următoarele

```
$ script
Script started, file is typescript
```

Executați orice comenzi shell sub `script`.

Apăsați `Ctrl-D` pentru a ieși din `script`.

```
$ vim typescript
```

Consultați Secțiune 9.1.1 .

1.4.10 Comenzi Unix de bază

Să învățăm comenzile de bază ale Unix. Aici folosesc „Unix” în sensul său generic. Orice sistem de operare clonă Unix oferă de obicei comenzi echivalente. Sistemul Debian nu face excepție. Nu vă faceți griji dacă unele comenzi

nu funcționează așa cum doriți acum. Dacă `alias` este utilizat în shell, ieșirile corespunzătoare ale comenzii sunt diferite. Aceste exemple nu sunt menite să fie executate în această ordine.

Încercați toate comenzile următoare din contul de utilizator fără privilegii.

Notă
Unix are tradiția de a ascunde numele fișierelor care încep cu „`.`”. Acestea sunt, de obicei, fișiere care conțin informații de configurare și preferințele utilizatorului.
Pentru comanda `cd`, consultați builtins(7).
Paginatorul implicit al sistemului Debian de bază este more(1), care nu permite derularea înapoi. Prin instalarea pachetului `less` folosind linia de comandă „`apt-get install less`”, less(1) devine paginatorul implicit și puteți derula înapoi cu tastele cursorului.
„`[`” și „`]`” din expresia regulată a comenzii „`ps aux | grep -e "[e]xim4*"`” de mai sus permit `grep` să evite potrivirea cu sine însuși. „`4*`” din expresia regulată înseamnă 0 sau mai multe repetări ale caracterului „4”, permițând astfel lui `grep` să potrivească atât cu „`exim`”, cât și cu „`exim4`”. Deși „`*`” este utilizat în globalizarea numelui de fișier shell și în expresia regulată, semnificațiile lor sunt diferite. Aflați mai multe despre expresia regulată din grep(1).

Vă rugăm să parcurgeți directoarele și să aruncați o privire în sistem folosind comenzile de mai sus ca exercițiu. Dacă aveți întrebări cu privire la oricare dintre comenzile consolei, vă rugăm să citiți pagina manualului.

De exemplu, încercați următoarele

```
$ man man
$ man bash
$ man builtins
$ man grep
$ man ls
```

Stilul paginilor de manual poate fi puțin dificil de asimilat, deoarece acestea sunt destul de concise, în special cele mai vechi, foarte tradiționale. Dar, odată ce vă obișnuiți cu ele, veți aprecia concisitatea lor.

Rețineți că multe comenzi de tip Unix, inclusiv cele din GNU și BSD, afișează informații succinte de ajutor dacă le invocați într-unul din următoarele moduri (sau fără argumente, în unele cazuri).

```
$ commandname --help
$ commandname -h
```

1.5 Comanda simplă de shell

Acum aveți o idee despre cum se utilizează sistemul Debian. Să analizăm în detaliu mecanismul de execuție a comenzilor în sistemul Debian. Aici, am simplificat realitatea pentru începători. Consultați bash(1) pentru explicații exacte.

O comandă simplă este o secvență de componente.

1. Atribuiri de variabile (opțional)
2. Numele comenzii
3. Argumente (opțional)
4. Redirecționări (opțional: `>` , `>>` , `<` , `<<` , etc.)
5. Operator de control (opțional: `&&` , `||` , *newline* , `;` , `&` , `(` , `)`)

comandă	**descriere**
`pwd`	afișează numele directorului curent/de lucru
`whoami`	afișează numele utilizatorului curent
`id`	afișează identitatea utilizatorului curent (nume, uid, gid și grupurile asociate)
`file` *foo*	afișează tipul de fișier pentru fișierul „*foo*”
`type -p` *commandname*	afișează locația fișierului comenzii „*nume-comandă*”
`which` *commandname*	, ,
`type` *commandname*	afișează informații despre comanda „*nume-comandă*”
`apropos` *key-word*	găsește comenzi legate de „*cuvânt-cheie*”
`man -k` *key-word*	, ,
`whatis` *commandname*	afișează o explicație pe o singură linie privind comanda „*nume-comandă*”
`man -a` *commandname*	afișează explicația privind comanda „*nume-comandă*” (stil Unix)
`info` *commandname*	afișează o explicație destul de lungă privind comanda „*nume-comandă*” (stil GNU)
`ls`	listează conținutul directorului (fișiere și directoare fără punct)
`ls -a`	listează conținutul directorului (toate fișierele și directoarele)
`ls -A`	listează conținutul directorului (aproape toate fișierele și directoarele, adică omite „`..`” și „`.`”)
`ls -la`	listează tot conținutul directorului cu informații detaliate
`ls -lai`	listează tot conținutul directorului cu numărul de nod-i și informații detaliate
`ls -d`	listează toate directoarele din directorul curent
`tree`	afișează conținutul arborelui de fișiere
`lsof` *foo*	afișează starea deschisă a fișierului „*foo*”
`lsof -p` *pid*	listează fișierele deschise de ID-ul procesului: „*pid*”
`mkdir` *foo*	creează un nou director „*foo*” în directorul curent
`rmdir` *foo*	elimină directorul „*foo*” din directorul curent
`cd` *foo*	schimbă directorul în directorul „*foo*” din directorul curent sau din directorul listat în variabila „`$CDPATH`”
`cd /`	schimbă directorul la directorul rădăcină
`cd`	schimbă directorul la directorul personal al utilizatorului curent
`cd /`*foo*	schimbă directorul la directorul cu ruta absolută „*/foo*”
`cd ..`	schimbă directorul la directorul părinte
`cd ~`*foo*	schimbă directorul la directorul personal al utilizatorului „*foo*”
`cd -`	schimbă directorul la directorul anterior
`</etc/motd pager`	afișează conținutul fișierului „`/etc/motd`” folosind paginatorul implicit
`touch` *junkfile*	creați un fișier gol „*fișier-test*”
`cp` *foo bar*	copiază un fișier existent „*foo*” într-un fișier nou „*bar*”
`rm` *junkfile*	elimină fișierul „*fișier-test*”
`mv` *foo bar*	redenumește un fișier existent „*foo*” într-un nou nume „*bar*” („*bar*” nu trebuie să existe)
`mv` *foo bar*	mută un fișier existent „*foo*” într-o nouă locație „*bar/foo*” (directorul „*bar*” trebuie să existe)
`mv` *foo bar/baz*	mută un fișier existent „*foo*” într-o nouă locație cu un nou nume „*bar/baz*” (directorul „*bar*” trebuie să existe, dar directorul „*bar/baz*” nu trebuie să existe)
`chmod 600` *foo*	face ca un fișier existent „*foo*” să fie necitibil și neinscriptibil de către alte persoane (neexecutabil pentru toți)
`chmod 644` *foo*	face ca un fișier existent „*foo*” să fie citibil, dar neinscriptibil de către alte persoane (neexecutabil pentru toți)
`chmod 755` *foo*	faceți ca un fișier existent „*foo*” să fie citibil, dar neinscriptibil de către alte persoane (executabil pentru toți)
`find . -name` *pattern*	găsește numele de fișiere ce coincid folosind shell-ul „*model*” (mai lent)
`locate -d . pattern`	găsește numele de fișiere ce coincid folosind shell-ul „*model*” (mai rapid folosind baza de date generată regulat)
`grep -e "`*pattern*`" *.html`	găsește un „*model*” în toate fișierele care se termină cu „`.html`” din directorul curent și le afișează pe toate
`top`	afișează informații despre proces pe ecran complet, tastați „q” pentru a ieși
`ps aux | pager`	afișează informații despre toate procesele care rulează utilizând

1.5.1 Executarea comenzilor și variabilele de mediu

Valorile unor variabile de mediu modifică comportamentul unor comenzi Unix.

Valorile implicite ale variabilelor de mediu sunt stabilite inițial de sistemul PAM, iar unele dintre ele pot fi reajustate de anumite programe aplicaționale.

- Sistemul PAM, cum ar fi `pam_env`, poate defini variabile de mediu prin `/etc/pam.conf`„, ”`/etc/environment`„ și ”`/etc/default/locale`”.
- Mediul grafic modern de birou, rulat din unitatea systemd din modul utilizator (a se vedea systemd.unit(5)), își configurează variabilele de mediu printr-un fișier din directorul „`~/.config/environment.d/`”.
- Inițializarea programului specific utilizatorului poate redefini variabilele de mediu prin „`~/.profile`”, „`~/.bash_profile`” și „`~/.bashrc`”.

1.5.2 Variabila „`$LANG`”

Configurația regională implicită este definită în variabila de mediu „`$LANG`” și este configurată ca „`LANG=xx_YY.UTF-8`” de către programul de instalare sau de către configurația ulterioară a interfeței grafice, de exemplu, „Configurări” → „Regiune și limbă” → „Limbă” / „Formate” pentru GNOME.

Notă
Vă recomand să configurați mediul de sistem doar cu variabila „`$LANG`” pentru moment și să evitați variabilele „`$LC_*`”, cu excepția cazului în care este absolut necesar.

Valoarea completă a configurației regionale atribuită variabilei „`$LANG`” este formată din 3 părți: „`xx_YY.ZZZZ`”.

valoarea configurației regionale	semnificație
`xx`	codurile de limbă ISO 639 (cu litere mici), cum ar fi „ro”
`YY`	codurile de țară ISO 3166 (majuscule), cum ar fi „RO”
`ZZZZ`	codificarea setului de caractere, definită întotdeauna la „UTF-8”

Tabela 1.18: Cele 3 părți ale valorii configurației regionale

Executarea tipică a comenzii utilizează o secvență de linii shell, după cum urmează.

```
$ echo $LANG
en_US.UTF-8
$ date -u
Wed 19 May 2021 03:18:43 PM UTC
$ LANG=fr_FR.UTF-8 date -u
mer. 19 mai 2021 15:19:02 UTC
```

Aici, programul date(1) este executat cu valori de configurație regională diferite.

- Pentru prima comandă, „`$LANG`” este definită la valoarea implicită a sistemului pentru configurația regională „`en_US.UTF-8`”
- Pentru a doua comandă, „`$LANG`” este definită la valoarea configurației regionale franceze UTF-8 „`fr_FR.UTF-8`”.

Majoritatea execuțiilor de comenzi nu au, de obicei, o definiție a variabilei de mediu precedentă. Pentru exemplul de mai sus, puteți executa alternativ următoarele.

```
$ LANG=fr_FR.UTF-8
$ date -u
mer. 19 mai 2021 15:19:24 UTC
```

recomandare configurație regională	limba (zona teritorială)
en_US.UTF-8	engleză (SUA)
en_GB.UTF-8	engleză (Marea Britanie)
fr_FR.UTF-8	franceză (Franța)
de_DE.UTF-8	germană (Germania)
it_IT.UTF-8	italiană (Italia)
es_ES.UTF-8	spaniolă (Spania)
ca_ES.UTF-8	catalană (Spania)
sv_SE.UTF-8	suedeză (Suedia)
pt_BR.UTF-8	portugheză (Brazilia)
ru_RU.UTF-8	rusă (Rusia)
zh_CN.UTF-8	chineză (R.P. Chiina)
zh_TW.UTF-8	chineză (Taiwan)
ja_JP.UTF-8	japoneză (Japonia)
ko_KR.UTF-8	coreeană (Republica Coreea)
vi_VN.UTF-8	vietnameză (Vietnam)

Tabela 1.19: Lista recomandărilor privind configurația regională

Indicație
Când raportați o eroare, este recomandat să rulați și să verificați comanda în configurația regională „en_US.UTF-8” dacă utilizați un mediu non-englez.

Pentru detalii precise privind configurarea parametrilor configurației regionale, consultați Secțiune 8.1.

1.5.3 Variabila „$PATH”

Când introduceți o comandă în shell, shell-ul caută comanda în lista de directoare conținute în variabila de mediu „$PATH”. Valoarea variabilei de mediu „$PATH” este denumită și ruta de căutare a shell-ului.

În instalarea implicită Debian, variabila de mediu „$PATH” a conturilor de utilizator poate să nu includă „/usr/sbin” și „/usr/sbin”. De exemplu, comanda ifconfig trebuie emisă cu ruta completă „/usr/sbin/ifconfig”; (comanda similară ip se află în „/usr/bin”).

Puteți modifica variabila de mediu „$PATH” a shell-ului Bash prin fișierele „~/.bash_profile” sau „~/.bashrc”.

1.5.4 Variabila „$HOME”

Multe comenzi stochează configurația specifică utilizatorului în directorul personal al acestuia și modifică comportamentul acestora în funcție de conținutul lor. Directorul personal este identificat de variabila de mediu „$HOME”.

valoarea variabilei „$HOME”	situația execuției programului
/	programul este rulat de procesul init (demon)
/root	programul rulează din shell-ul root normal
/home/*normal_user*	programul rulează din shell-ul utilizatorului normal
/home/*normal_user*	programul rulează din meniul mediului grafic de birou al utilizatorului normal
/home/*normal_user*	programul rulează ca root cu «sudo program»
/root	programul rulează ca root cu «sudo -H program»

Tabela 1.20: Lista valorilor variabilei „$HOME”

Indicație
Shell-ul extinde „~/” la directorul personal al utilizatorului curent, adică „$HOME/”. Shell-ul extinde „~foo/” la directorul personal al lui foo, adică „/home/foo/”.

Consultați Secțiune 12.1.5 dacă $HOME nu este disponibil pentru programul dvs.

1.5.5 Opțiuni în linia de comandă

Unele comenzi acceptă argumente. Argumentele care încep cu „-” sau „--” sunt numite opțiuni și controlează comportamentul comenzii.

```
$ date
Thu 20 May 2021 01:08:08 AM JST
$ date -R
Thu, 20 May 2021 01:08:12 +0900
```

Aici, argumentul liniei de comandă „-R” modifică comportamentul date(1) pentru a genera un șir de caractere compatibil cu RFC2822.

1.5.6 Facilitatea glob a shell-ului

Adesea, doriți ca o comandă să funcționeze cu un grup de fișiere fără a le introduce pe toate. Modelul de extindere a numelor de fișiere utilizând facilitatea **glob** a shell-ului (denumită uneori **wildcards**) facilitează această necesitate.

modele globale shell	descrierea regulii de potrivire
*	numele fișierului nu începe cu „.”
.*	numele fișierului a început cu „.” (în Bash 5.2 și versiunile ulterioare, „.” și „..” sunt excluse în mod implicit)
?	exact un caracter
[…]	exact un caracter cu orice caracter între paranteze
[a-z]	exact un caracter cu orice caracter între „a” și „z”
[!…]	exact un caracter, altul decât orice caracter inclus între paranteze (cu excepția „!”)

Tabela 1.21: Modele globale shell

Notă
Un element „/” dintr-o rută nu poate fi identificat de niciunul dintre modelele globale ale shell-ului enumerate în tabelul de mai sus.

De exemplu, încercați următoarele

```
$ mkdir foo; cd foo; touch b1.txt b2.txt b3.c b4.h .b5.txt ..b6.txt
$ mkdir bar; cd bar; touch b1.txt b2.txt b3.c b4.h .b5.txt ..b6.txt ; cd -
$ echo b*.txt
b1.txt b2.txt
$ echo b*
b1.txt b2.txt b3.c b4.h bar
$ echo */*.txt
bar/b1.txt bar/b2.txt
$ echo *.[hc]
b3.c b4.h
```

```
$ echo .*
.b5.txt ..b6.txt
$ echo b[!1-3]*
b4.h bar
$ echo b*/b[!1-3]*
bar/b4.h
$ cd ..; rm -rf junk
```

A se vedea glob(7).

Notă
Spre deosebire de extinderea normală a numelor de fișiere de către shell, modelul shell „*” testat în find(1) cu testul „-name” etc., se potrivește cu „.” inițial al numelui fișierului; (noua caracteristică POSIX).

Notă
BASH poate fi modificat pentru a-i schimba comportamentul glob cu opțiunile sale încorporate shopt, cum ar fi „dotglob”, „noglob”, „nocaseglob”, „nullglob”, „extglob” etc. Consultați bash(1).

1.5.7 Valoarea returnată de comandă

Fiecare comandă returnează starea sa de ieșire (variabila: „$?”) ca valoare de returnare.

starea de ieșire a comenzii	valoare numerică returnată	valoarea logică returnată
succes	zero, 0	**TRUE**
eroare	diferită de zero, -1	**FALSE**

Tabela 1.22: Coduri de ieșire ale comenzii

De exemplu, încercați următoarele.

```
$ [ 1 = 1 ] ; echo $?
0
$ [ 1 = 2 ] ; echo $?
1
```

Notă
Vă rugăm să rețineți că, în contextul logic pentru shell, **succesul** este tratat ca **TRUE** logic, care are valoarea 0 (zero). Acest lucru este oarecum neintuitiv și trebuie reamintit aici.

1.5.8 Secvențe tipice de comenzi și redirecționarea shell-ului

Să încercăm să reținem următoarele expresii ale comenzii shell tastate într-o singură linie ca parte a comenzii shell.

Sistemul Debian este un sistem multi-sarcini. Sarcinile de fundal permit utilizatorilor să ruleze mai multe programe într-un singur shell. Gestionarea proceselor în fundal implică utilizarea comenzilor încorporate ale shell-ului: `jobs`, `fg`, `bg` și `kill`. Vă rugăm să citiți secțiunile din bash(1) sub „SEMNALE” și „CONTROLUL SARCINILOR” și builtins(1).

De exemplu, încercați următoarele

```
$ </etc/motd pager
```

expresia idiomatică a comenzii	**descriere**
`command &`	**execuție în fundal** a `comenzii` în subshell
`command1 | command2`	**redirecționează** ieșirea standard a `comenzii1` către intrarea standard a `comenzii2` (execuție **concomitentă**)
`command1 2>&1 | command2`	**redirecționează** atât ieșirea standard, cât și eroarea standard a `comenzii1` către intrarea standard a `comenzii2` (execuție **concomitentă**)
`command1 ; command2`	execută `comanda1` și `comanda2` **secvențial**
`command1 && command2`	execută `comanda1`; dacă are succes, execută `comanda2` **secvențial** (returnează succes dacă atât `comanda1` **cât și** `comanda2` au succes)
`command1 || command2`	execută `comanda1`; dacă nu are succes, execută `comanda2` **secvențial** (returnează succes dacă `comanda1` **sau** `comanda2` au succes)
`command > foo`	redirecționează ieșirea standard a comenzii `comanda` către un fișier `foo` (suprascrie)
`command 2> foo`	redirecționează ieșirea de eroare standard a comenzii `comanda` către un fișier `foo` (suprascrie)
`command >> foo`	redirecționează ieșirea standard a comenzii `comanda` către un fișier `foo` (adăugare)
`command 2>> foo`	redirecționează ieșirea de eroare standard a comenzii `command` către un fișier `foo` (adăugare)
`command > foo 2>&1`	redirecționează atât ieșirea standard, cât și ieșirea de eroare standard a comenzii `comanda` către un fișier `foo`
`command < foo`	redirecționează intrarea standard a comenzii `comanda` către un fișier `foo`
`command << delimiter`	redirecționează intrarea standard a comenzii `comanda` către următoarele linii până când se întâlnește „`delimitatorul`” (aici, document)
`command <<- delimiter`	redirecționează intrarea standard a comenzii `comanda` către următoarele linii până când se întâlnește „`delimitatorul`” (aici document, caracterele de tabulare din fața liniilor de intrare sunt eliminate)

Tabela 1.23: Expresii idiomatice ale comenzii shell

```
$ pager </etc/motd
```

```
$ pager /etc/motd
```

```
$ cat /etc/motd | pager
```

Deși toate cele 4 exemple de redirecționări shell afișează același lucru, ultimul exemplu execută o comandă suplimentară `cat` și irosește resurse fără niciun motiv.

Shell-ul vă permite să deschideți fișiere folosind comanda internă `exec` cu un descriptor de fișier arbitrar.

```
$ echo Hello >foo
$ exec 3<foo 4>bar  # open files
$ cat <&3 >&4       # redirect stdin to 3, stdout to 4
$ exec 3<&- 4>&-    # close files
$ cat bar
Hello
```

Descriptorii de fișiere 0-2 sunt predefiniți.

dispozitiv	descriere	descriptor de fișier
`stdin`	intrarea standard	0
`stdout`	ieșirea standard	1
`stderr`	ieșirea de eroare standard	2

Tabela 1.24: Descriptori de fișiere predefiniți

1.5.9 Alias de comandă

Puteți defini un alias pentru comanda utilizată frecvent.

De exemplu, încercați următoarele

```
$ alias la='ls -la'
```

Acum, «`la`» funcționează ca o prescurtare pentru «`ls -la`», care listează toate fișierele în formatul de listare lungă.

Puteți lista orice alias existente prin `alias` (consultați bash(1) la «COMENZILE INTERNE ALE SHELL-ului»).

```
$ alias
...
alias la='ls -la'
```

Puteți identifica ruta exactă sau identitatea comenzii prin `type` (consultați bash(1) la «COMENZILE INTERNE ALE SHELL-ului»).

De exemplu, încercați următoarele

```
$ type ls
ls is hashed (/bin/ls)
$ type la
la is aliased to ls -la
$ type echo
echo is a shell builtin
$ type file
file is /usr/bin/file
```

Aici, «ls» a fost căutat recent, în timp ce «file» nu a fost, astfel încât «ls» este indexat, adică shell-ul are o înregistrare internă pentru accesul rapid la locația comenzii «ls».

Indicație
A se vedea Secțiune 9.3.6.

1.6 Procesarea textului în stilul Unix

În mediul de lucru de tip Unix, procesarea textului se realizează prin transferul textului prin intermediul unor lanțuri de instrumente standard de procesare a textului. Aceasta a fost o altă inovație crucială a sistemului Unix.

1.6.1 Instrumente pentru text în Unix

Există câteva instrumente standard de procesare a textului care sunt utilizate foarte des în sistemele de tip Unix.

- Nu se utilizează nicio expresie regulată:
  - cat(1) concatenează fișiere și afișează întregul conținut.
  - tac(1) concatenează fișierele și le afișează în ordine inversă.
  - cut(1) selectează părți din linii și le afișează.
  - head(1) afișează prima parte a fișierelor.
  - tail(1) afișează ultima parte a fișierelor.
  - sort(1) sortează liniile din fișierele text.
  - uniq(1) elimină liniile duplicate dintr-un fișier sortat.
  - tr(1) traduce sau șterge caractere.
  - diff(1) compară fișierele linie cu linie.
- Expresia regulată de bază („Basic regular expression”: **BRE**) este utilizată în mod implicit:
  - ed(1) este un editor de linii primitiv.
  - sed(1) este un editor de flux.
  - grep(1) potrivește textul cu modelele.
  - vim(1) este un editor de ecran.
  - emacs(1) este un editor de ecran. (oarecum extins cu **BRE**)
- Se utilizează expresia regulată extinsă („Extended regular expression”: **ERE**):
  - awk(1) efectuează procesarea simplă a textului.
  - egrep(1) potrivește textul cu modelele.
  - `tcl`(3tcl) poate efectua orice operație de procesare a textului: consultați re_syntax(3). Se utilizează adesea împreună cu `tk`(3tk).
  - perl(1) poate efectua orice operație de procesare a textului imaginabilă. Consultați perlre(1).
  - pcre2grep(1) din pachetul `pcre2-util` potrivește textul cu modelul Perl Compatible Regular Expressions (PCRE).
  - python(1) cu modulul `re` poate efectua orice procesare de text imaginabilă. Consultați „`/usr/share/doc/python/htm`

Dacă nu sunteți sigur ce fac exact aceste comenzi, vă rugăm să utilizați «`man comanda`» pentru a afla singur.

Notă
Ordinea de sortare și expresia intervalului depind de configurația regională. Dacă doriți să obțineți comportamentul tradițional pentru o comandă, utilizați parametrul de configurare regională **C** sau **C.UTF-8** în locul celor normale **UTF-8** (consultați Secțiune 8.1).

Notă
Expresiile regulate Perl(perlre(1)), Expresii regulate compatibile cu Perl („Perl Compatible Regular Expressions”: PCRE) și expresiile regulate Python oferite de modulul `re` au multe extensii comune cu **ERE** normal.

1.6.2 Expresii regulate

Expresiile regulate sunt utilizate în multe instrumente de procesare a textului. Acestea sunt similare cu modelele globale din shell, dar sunt mai complicate și mai puternice.

Expresia regulată descrie modelul de potrivire și este alcătuită din caractere text și **metacaractere**.

Un **metacaracter** este doar un caracter cu o semnificație specială. Există două stiluri principale, „Basic Regular Expressions”: **BRE** (expresii regulate de bază) și „Extended Regular Expressions”: **ERE** (expresii regulate extinse), în funcție de instrumentele de text descrise mai sus.

BRE	**ERE**	**descrierea expresiei regulate**
`\ . [ ] ^ $ *`	`\ . [ ] ^ $ *`	**metacaractere** comune
`\+ \? \( \) \{ \} |`		doar în BRE, **metacaractere** eludate „\”
	`+ ? ( ) { } |`	doar în ERE, **metacaractere** fără caracterul de eludare „\”
c	c	potrivește cu un **non-metacaracter** „c”
`\c`	`\c`	potrivește un caracter literal „c” chiar dacă „c” este de fapt un **metacaracter**
`.`	`.`	potrivește orice caracter, inclusiv cel de linie nouă
`^`	`^`	poziția la începutul unui șir de caractere
$	$	poziția la sfârșitul unui șir de caractere
`\<`	`\<`	poziția la începutul unui cuvânt
`\>`	`\>`	poziția la sfârșitul unui cuvânt
`[abc…]`	`[abc…]`	potrivește orice caractere din „`abc…`”
`[^abc…]`	`[^abc…]`	potrivește orice caractere, cu excepția celor din „`abc…`”
`r*`	`r*`	potrivește zero sau mai multe expresii regulate identificate prin „`r`”
`r\+`	`r+`	potrivește una sau mai multe expresii regulate identificate prin „`r`”
`r\?`	`r?`	potrivește zero sau o expresie regulată identificată prin „`r`”
`r1|r2`	`r1|r2`	potrivește una dintre expresiile regulate identificate prin „`r1`” sau „`r2`”
`\(r1|r2\)`	`(r1|r2)`	potrivește una dintre expresiile regulate identificate prin „`r1`” sau „`r2`” și o tratează ca o expresie regulată **încadrată între paranteze**

Tabela 1.25: Metacaractere pentru BRE și ERE

Expresia regulată **`emacs`** este în esență **BRE**, dar a fost extinsă pentru a trata „+” și „?” ca **metacaractere** ca în **ERE**. Astfel, nu este necesar să le eludați cu „\” în expresia regulată a `emacs`.

grep(1) poate fi utilizat pentru a efectua căutarea textului folosind o expresie regulată.

De exemplu, încercați următoarele

```
$ egrep 'GNU.*LICENSE|Yoyodyne' /usr/share/common-licenses/GPL
GNU GENERAL PUBLIC LICENSE
GNU GENERAL PUBLIC LICENSE
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
```

Indicație
A se vedea Secțiune 9.3.6.

1.6.3 Expresii de înlocuire

Pentru expresia de înlocuire, unele caractere au semnificații speciale.

expresie de înlocuire	descrierea textului care va înlocui expresia de înlocuire
&	ceea ce a fost găsit de expresia regulată (utilizați \& în emacs)
\n	cu ce s-a potrivit a n-a expresie regulată **între paranteze** (unde „n” este un număr)

Tabela 1.26: Expresia de înlocuire

Pentru înlocuirea șirurilor în Perl, se utilizează „$&” în loc de „&” și „$n” în loc de „\n”.

De exemplu, încercați următoarele

```
$ echo zzz1abc2efg3hij4 | \
sed -e 's/\(1[a-z]*\)[0-9]*\(.*\)$/=&=/'
zzz=1abc2efg3hij4=
$ echo zzz1abc2efg3hij4 | \
sed -E -e 's/(1[a-z]*)[0-9]*(.*)$/=&=/'
zzz=1abc2efg3hij4=
$ echo zzz1abc2efg3hij4 | \
perl -pe 's/(1[a-z]*)[0-9]*(.*)$/=$&=/'
zzz=1abc2efg3hij4=
$ echo zzz1abc2efg3hij4 | \
sed -e 's/\(1[a-z]*\)[0-9]*\(.*\)$/\2===\1/'
zzzefg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
sed -E -e 's/(1[a-z]*)[0-9]*(.*)$/\2===\1/'
zzzefg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
perl -pe 's/(1[a-z]*)[0-9]*(.*)$/$2===$1/'
zzzefg3hij4===1abc
```

Aici, vă rugăm să acordați o atenție deosebită stilului expresiei regulate **în paranteze** și modului în care șirurile potrivite sunt utilizate în procesul de înlocuire a textului în diferite instrumente.

Aceste expresii regulate pot fi utilizate și pentru mișcările cursorului și acțiunile de înlocuire a textului în unele editoare.

Bară oblică inversă „\” la sfârșitul liniei în linia de comandă a shell-ului eludează caracterul de linie nouă ca caracter spațiu alb și continuă introducerea liniei de comandă a shell-ului în linia următoare.

Vă rugăm să citiți toate paginile de manual aferente pentru a învăța aceste comenzi.

1.6.4 Înlocuire globală cu expresii regulate

Comanda ed(1) poate înlocui toate aparițiile „FROM_REGEX” cu „TO_TEXT” în „file”.

```
$ ed file <<EOF
,s/FROM_REGEX/TO_TEXT/g
w
q
EOF
```

Comanda sed(1) poate înlocui toate aparițiile „FROM_REGEX” cu „TO_TEXT” în „file”.

```
$ sed -i -e 's/FROM_REGEX/TO_TEXT/g' file
```

Comanda vim(1) poate înlocui toate aparițiile „FROM_REGEX” cu „TO_TEXT” în „file” folosind comenzi ex(1).

```
$ vim '+%s/FROM_REGEX/TO_TEXT/gc' '+update' '+q' file
```

Indicație
Indicatorul „c” din exemplul de mai sus asigură confirmarea interactivă pentru fiecare substituție.

Mai multe fișiere („file1”, „file2” și „file3”) pot fi procesate cu expresii regulate în mod similar cu vim(1) sau perl(1).

```
$ vim '+argdo %s/FROM_REGEX/TO_TEXT/gce|update' '+q' file1 file2 file3
```

Indicație
Indicatorul „e” din exemplul de mai sus împiedică eroarea „No match” (fără corespondență) să întrerupă o căutare.

```
$ perl -i -p -e 's/FROM_REGEX/TO_TEXT/g;' file1 file2 file3
```

În exemplul perl(1), „-i” este pentru editarea pe loc (fără folosirea unui tampon/fișier temporal pentru aceasta) a fiecărui fișier țintă, iar „-p” este pentru bucla implicită peste toate fișierele date.

Indicație
Utilizarea argumentului „-i.bak” în locul „-i” păstrează fiecare fișier original prin adăugarea „.bak” la numele fișierului. Acest lucru facilitează recuperarea în cazul erorilor pentru substituții complexe.

Notă
ed(1) și vim(1) sunt **BRE**; perl(1) este **ERE**.

1.6.5 Extragerea datelor din tabelul fișierului text

Să luăm în considerare un fișier text numit „DPL” în care sunt enumerate numele unor lideri ai proiectului Debian dinainte de 2004 și data inițierii lor, într-un format separat prin spații.

```
Ian     Murdock   August  1993
Bruce   Perens    April   1996
Ian     Jackson   January 1998
Wichert Akkerman  January 1999
Ben     Collins   April   2001
Bdale   Garbee    April   2002
Martin  Michlmayr March   2003
```

Indicație
Consultați „O scurtă istorie a Debian” pentru cea mai recentă istorie a conducerii Debian.

Awk este frecvent utilizat pentru extragerea datelor din aceste tipuri de fișiere.

De exemplu, încercați următoarele

```
$ awk '{ print $3 }' <DPL                  # month started
August
April
January
January
April
April
March
$ awk '($1=="Ian") { print }' <DPL         # DPL called Ian
Ian     Murdock   August  1993
Ian     Jackson   January 1998
$ awk '($2=="Perens") { print $3,$4 }' <DPL # When Perens started
April 1996
```

Shell-uri precum Bash pot fi, de asemenea, utilizate pentru a analiza acest tip de fișiere.

De exemplu, încercați următoarele

```
$ while read first last month year; do
    echo $month
  done <DPL
... same output as the first Awk example
```

Aici, comanda internă `read` utilizează caracterele din „`$IFS`” („internal field separators” - separatoare interne de câmpuri) pentru a împărți liniile în cuvinte.

Dacă schimbați „`$IFS`” cu „`:`”, puteți analiza „`/etc/passwd`” cu shell-ul fără probleme.

```
$ oldIFS="$IFS"   # save old value
$ IFS=':'
$ while read user password uid gid rest_of_line; do
    if [ "$user" = "bozo" ]; then
      echo "$user's ID is $uid"
    fi
  done < /etc/passwd
bozo's ID is 1000
$ IFS="$oldIFS"   # restore old value
```

(Dacă se utilizează Awk pentru a face același lucru, utilizați „`FS=':'`” pentru a defini separatorul de câmpuri.)

IFS este utilizat și de shell pentru a împărți rezultatele expansiunii parametrilor, substituției comenzilor și expansiunii aritmetice. Acestea nu apar în cuvintele între ghilimele duble sau simple. Valoarea implicită a IFS este *spațiu*, *tabulator* și *linie nouă* combinate.

Aveți grijă când utilizați aceste trucuri IFS ale shell-ului. Pot apărea lucruri ciudate atunci când shell-ul interpretează anumite părți ale scriptului ca fiind **intrare**.

```
$ IFS=":,"                          # use ":" and "," as IFS
$ echo IFS=$IFS,   IFS="$IFS"       # echo is a Bash builtin
IFS=  , IFS=:,
$ date -R                           # just a command output
Sat, 23 Aug 2003 08:30:15 +0200
$ echo $(date -R)                   # sub shell --> input to main shell
Sat  23 Aug 2003 08 30 36 +0200
$ unset IFS                         # reset IFS to the default
$ echo $(date -R)
Sat, 23 Aug 2003 08:30:50 +0200
```

1.6.6 Fragmente de script pentru comenzi de direcționare

Următoarele scripturi fac lucruri interesante ca parte a unei conducte (direcționări).

fragment de script (introduceți într-o singură linie)	**efectul comenzii**
`find /usr -print`	găsește toate fișierele din „`/usr`”
`seq 1 100`	imprimă numere de la 1 până la 100
`| xargs -n 1 command`	rulează comanda în mod repetat cu fiecare element din conductă ca argument al său
`| xargs -n 1 echo`	împarte elementele separate prin spații albe din conductă în linii
`| xargs echo`	fuzionează toate liniile din conductă într-o singură linie
`| grep -e regex_pattern`	extrage linii din conductă care conțin *model_expresie-regulată*
`| grep -v -e regex_pattern`	extrage liniile din conductă care nu conțin *model_expresie-regulată*
`| cut -d: -f3 -`	extrage al treilea câmp din conductă separat prin „`:`” (fișier passwd etc.)
`| awk '{ print $3 }'`	extrage al treilea câmp din șirul separat prin spații
`| awk -F'\t' '{ print $3 }'`	extrage al treilea câmp din șirul separat prin tabulator
`| col -bx`	elimină „backspace” (retururile de cărucior) și transformă tabulatoarele în spații
`| expand -`	transformă tabulatoarele în spații
`| sort| uniq`	sortează și elimină duplicatele
`| tr 'A-Z' 'a-z'`	convertește majusculele în minuscule
`| tr -d '\n'`	concatenează liniile într-o singură linie
`| tr -d '\r'`	elimină CR (retururile de cărucior)
`| sed 's/^/# /'`	adaugă „#” la începutul fiecărei linii
`| sed 's/\.ext//g'`	elimină „`.ext`”
`| sed -n -e 2p`	imprimă a doua linie
`| head -n 2 -`	imprimă primele 2 linii
`| tail -n 2 -`	imprimă ultimele 2 linii

Tabela 1.27: Lista fragmentelor de script pentru comenzi de direcționare

Un script shell de o singură linie poate parcurge mai multe fișiere folosind find(1) și xargs(1) pentru a efectua sarcini destul de complicate. Vedeți Secțiune 10.1.5 și Secțiune 9.4.9.

Când utilizarea modului interactiv al shell-ului devine prea complicată, luați în considerare scrierea unui script shell (consultați Secțiune 12.1).

Capitolul 2

Gestionarea pachetelor Debian

Notă
Acest capitol este scris presupunând că cea mai recentă versiune stabilă este nume în cod: `trixie`.
Sursa de date a sistemului APT este denumită în mod colectiv **lista surselor** în acest document . Aceasta poate fi definită oriunde în fișierul „`/etc/apt/sources.list`”, în fișierele „`/etc/apt/sources.list.d/*.list`” sau în fișierele „`/etc/apt/sources.list.d/*.sources`”.

2.1 Cerințe preliminare pentru gestionarea pachetelor Debian

2.1.1 Sistemul de gestionare a pachetelor Debian

Debian este o organizație voluntară care creează distribuții **consistente** de pachete binare precompilate de software liber și le distribuie din arhiva sa.

Arhiva Debian este oferită de multe situri oglindă la distanță pentru acces prin metode HTTP și FTP. Este disponibilă și pe CD-ROM/DVD.

Sistemul actual de gestionare a pachetelor Debian care poate utiliza toate aceste resurse este Advanced Packaging Tool (APT).

Sistemul de gestionare a pachetelor Debian, **atunci când este utilizat corect**, oferă utilizatorului posibilitatea de a instala **seturi consistente de pachete binare** în sistem din arhivă. În prezent, există 78072 de pachete disponibile pentru arhitectura amd64.

Sistemul de gestionare a pachetelor Debian are o istorie bogată și oferă numeroase opțiuni pentru programul de interfață cu utilizatorul final și metoda de acces la arhiva care urmează să fie utilizată. În prezent, recomandăm următoarele.

- apt(8) pentru toate operațiile interactive din linia de comandă, inclusiv instalarea, eliminarea și actualizarea pachetelor.
- apt-get(8) pentru apelarea sistemului de gestionare a pachetelor Debian din scripturi. Este, de asemenea, o opțiune de rezervă atunci când `apt` nu este disponibil (adesea în cazul sistemelor Debian mai vechi).
- aptitude(8) pentru o interfață text interactivă pentru gestionarea pachetelor instalate și căutarea pachetelor disponibile.

pachet	popcon(popularitate) și dimensiune	dimensiune instalată	descriere
dpkg	V:906, I:1000	4944	sistem de gestionare a pachetelor de nivel scăzut pentru Debian (bazat pe fișiere)
apt	V:895, I:1000	4956	interfață APT pentru gestionarea pachetelor cu CLI:(interfață de linie de comandă) apt/apt-get/apt-cache
aptitude	V:31, I:160	4621	interfață APT pentru gestionarea interactivă a pachetelor cu consolă pe ecran complet: aptitude(8)
tasksel	V:38, I:985	351	interfață APT pentru instalarea sarcinilor selectate: tasksel(8)
unattended-upgrades	V:130, I:182	320	pachet de îmbunătățiri pentru APT pentru a permite instalarea automată a actualizărilor de securitate
gnome-software	V:158, I:263	4732	Centru software pentru GNOME (interfață grafică pentru APT)
synaptic	V:31, I:269	7264	administrator grafic de pachete (interfață GTK APT)
apt-utils	V:420, I:998	1181	programe utilitare APT: apt-extracttemplates(1), apt-ftparchive(1) și apt-sortpkgs(1)
apt-listchanges	V:403, I:894	562	instrument de notificare a istoricului modificărilor pachetelor
apt-listbugs	V:4.7, I:6.5	512	listează erorile critice înainte de fiecare instalare APT
apt-file	V:14, I:53	76	instrument de căutare a pachetelor APT — interfață linie de comandă
apt-rdepends	V:0.4, I:4.2	39	listează recursiv dependențele pachetului

Tabela 2.1: Lista instrumentelor de gestionare a pachetelor Debian

2.1.2 Configurația pachetului

Iată câteva puncte cheie pentru configurarea pachetelor în sistemul Debian.

- Pentru mediul grafic de birou modern, repornirea sistemului după schimbarea configurației pachetelor și actualizarea pachetelor este o idee bună pentru a asigura funcționarea corectă a sistemului.
- Configurația manuală efectuată de administratorul de sistem este respectată. Cu alte cuvinte, sistemul de configurare a pachetelor nu efectuează nicio configurare intruzivă din motive de comoditate.
- Fiecare pachet vine cu propriul script de configurare cu interfață standardizată numită debconf(7) pentru a ajuta în procesul de instalare inițială a pachetului.
- Dezvoltatorii Debian fac tot posibilul pentru a vă oferi o experiență de actualizare fără probleme, cu ajutorul scripturilor de configurare a pachetelor.
- Administratorul de sistem are la dispoziție toate funcționalitățile pachetului software. Însă cele care prezintă riscuri de securitate sunt dezactivate în instalarea implicită.
- Dacă activați manual un serviciu care prezintă anumite riscuri de securitate, sunteți responsabil pentru limitarea riscurilor.
- Configurația ezoterică poate fi activată manual de administratorul de sistem. Acest lucru poate crea interferențe cu programele generice populare de asistență pentru configurarea sistemului.

2.1.3 Precauții de bază



Avertisment
Nu instalați pachete dintr-un amestec aleatoriu de suite. Probabil că acest lucru va afecta coerența pachetului, ceea ce necesită cunoștințe aprofundate de administrare a sistemului, cum ar fi compilatorul ABI, versiunea bibliotecii, caracteristicile interpretului etc.

Administratorul de sistem Debian începător ar trebui să rămână la versiunea **`stabilă`** a Debian, aplicând doar actualizările de securitate. Până când nu înțelegeți foarte bine sistemul Debian, ar trebui să urmați următoarele precauții.

- Nu includeți suitele **`testing`** (de testare) sau **`unstable`** (instabilă) în **lista sursă**.
- Nu amestecați Debian standard cu alte arhive non-Debian, cum ar fi Ubuntu, în **lista de surse**.
- Nu creați „`/etc/apt/preferences`”.
- Nu modificați comportamentul implicit al instrumentelor de gestionare a pachetelor prin fișiere de configurare fără a cunoaște impactul complet al acestora.
- Nu instalați pachete aleatorii folosind comanda „`dpkg -i` *`pachet_aleatoriu`*”.
- Nu instalați niciodată pachete aleatorii folosind comanda „`dpkg --force-all -i` *`pachet_aleatoriu`*”.
- Nu ștergeți și nu modificați fișierele din „`/var/lib/dpkg/`”.
- Nu suprascrieți fișierele de sistem prin instalarea programelor software compilate direct din sursă.
  - Instalați-le în „`/usr/local`” sau „`/opt`”, dacă este necesar.

Efectele incompatibile cauzate de încălcarea precauțiilor de mai sus asupra sistemului de gestionare a pachetelor Debian pot face sistemul dvs. inutilizabil.

Administratorii de sistem Debian serioși, care gestionează servere critice, ar trebui să ia măsuri de precauție suplimentare.

- Nu instalați niciun pachet, inclusiv actualizările de securitate de la Debian, fără a le testa temeinic cu configurația dvs. specifică, în condiții de siguranță.
  - În calitate de administrator de sistem, dumneavoastră sunteți responsabil în ultimă instanță pentru sistemul dumneavoastră.
  - Istoria îndelungată de stabilitate a sistemului Debian nu constituie o garanție în sine.

2.1.4 O viață cu îmbunătățiri veșnice



Atenție
Pentru **serverul de producție**, se recomandă suita `stabilă` cu actualizările de securitate. Același lucru se poate spune și despre calculatoarele stații de lucru pe care puteți depune eforturi administrative limitate.

În ciuda avertismentelor mele de mai sus, știu că mulți cititori ai acestui document ar putea dori să ruleze noile suite `testing` sau `unstable`.

Iluminarea cu următoarele salvează o persoană de lupta eternă karmică a ascensiunii iadului și îi permite să atingă nirvana Debian.

Această listă este destinată mediului de birou **autoadministrat**.

- Utilizați suita `testing`, deoarece aceasta este practic versiunea continuă gestionată automat de infrastructura QA a arhivei Debian, cum ar fi integrarea continuă Debian, practicile de încărcare numai a surselor și urmărirea tranziției bibliotecilor. Pachetele din suita `testing` sunt actualizate suficient de frecvent pentru a oferi toate funcționalitățile cele mai recente.
- Stabiliți numele de cod corespunzător suitei `testing` („`forky`” în timpul ciclului de lansare `trixie-ca-stable`) în **lista surselor** .

- Actualizați manual acest nume de cod în **lista surselor** cu cel nou numai după ce evaluați situația pe cont propriu timp de aproximativ o lună după lansarea suitei majore. Lista de discuții a utilizatorilor și dezvoltatorilor Debian este, de asemenea, o sursă bună de informații în acest sens.

Utilizarea suitei `unstable` nu este recomandată. Suita `unstable` este **potrivită pentru depanarea pachetelor** ca dezvoltator, dar tinde să vă expună la riscuri inutile pentru utilizarea normală a stației de lucru. Chiar dacă suita `unstable` a sistemului Debian pare foarte stabilă în majoritatea cazurilor, au existat unele probleme cu pachetele, iar câteva dintre ele nu au fost atât de ușor de rezolvat.

Iată câteva idei de măsuri de precauție de bază pentru a asigura recuperarea rapidă și ușoară în cazul apariției erorilor în pachetele Debian.

- Faceți disponibil sistemul de recuperare urmând instrucțiunile din secțiunea Secțiune 3.2.
- Faceți sistemul să fie cu **pornire dual**ă instalând suita `stable` a sistemului Debian pe o altă partiție
- Luați în considerare instalarea `apt-listbugs` pentru a verifica informațiile din Sistemul de urmărire a erorilor Debian (BTS) înainte de actualizare.
- Învățați infrastructura sistemului de pachete suficient pentru a rezolva problema



Atenție
Dacă nu puteți lua niciuna dintre aceste măsuri de precauție, probabil că nu sunteți pregătit pentru suitele `testing` (de testare) și `unstable` (instabilă).

2.1.5 Noțiuni de bază despre arhiva Debian

Indicație
Politica oficială a arhivei Debian este definită în Manualul de politici Debian, Capitolul 2 - Arhiva Debian.

Să analizăm arhiva Debian din perspectiva unui utilizator de sistem.

Pentru un utilizator al sistemului, arhiva Debian este accesată folosind sistemul APT.

Sistemul APT specifică sursa de date ca fiind **lista surselor** și este descris în sources.list(5).

Pentru sistemul `trixie` cu acces HTTP tipic, **lista surselor** este furnizat în stilul modern deb822 în „`/etc/apt/sources.l`
ca mai jos:

```
Types: deb deb-src
URIs: http://deb.debian.org/debian/
Suites: trixie trixie-updates
Components: main non-free-firmware contrib non-free

Types: deb deb-src
URIs: http://security.debian.org/debian-security/
Suites: trixie-security
Components: main non-free-firmware contrib non-free
```

Indicație
Dacă **lista surselor** este furnizată în stilul vechi, depreciat, pe o singură linie, în fișierele „`/etc/apt/sources.list`” sau „`/etc/apt/sources.list.d/*.list`”, actualizați-le cu:

```
$ sudo apt modernize-sources
```

Punctele cheie ale **listei surselor** în stilul deb822 sunt următoarele.

- Fișierele sale de definiție se află în fișierele „`/etc/apt/sources.list.d/*.sources`”.
- Fiecare bloc de linii separate printr-o linie goală definește sursa de date pentru sistemul APT.
- Strofa „`Types:`” definește lista tipurilor, cum ar fi „`deb`” și „`deb-src`”.
- Strofa „`URIs:`” definește lista adreselor URI rădăcină ale arhivei Debian.
- Strofa „`Suites:`” definește lista numelor de distribuție utilizând fie numele suitei, fie numele în cod.
- Strofa „`Components:`” definește lista numelor valide ale zonelor de arhivă din arhiva Debian.

Definiția pentru „`deb-src`” poate fi omisă în siguranță dacă este doar pentru `aptitude`, care nu accesează metadatele legate de sursă. Aceasta accelerează actualizările metadatelor arhivei.

Adresa URL poate fi de tipul „`https://`”, „`http://`”, „`ftp://`”, „`file://`”, ….

Liniile care încep cu „#” sunt comentarii și sunt ignorate.

Aici, tind să folosesc numele în cod „`trixie`” sau „`forky`” în loc de numele suitei „`stable`” sau „`testing`” pentru a evita surprizele atunci când va fi lansată următoarea versiune `stable`.

Indicație
Dacă în exemplul de mai sus se utilizează „`sid`” în loc de „`trixie`”, nu este necesară linia „`deb: http://security.debian.org/ …`” sau conținutul său echivalent deb822 pentru actualizările de securitate din **lista surselor** nu este necesar. Acest lucru se datorează faptului că nu există arhivă de actualizări de securitate pentru „`sid`” (`unstable`).

Iată lista adreselor URL ale siturilor de arhivă Debian și numele suitei sau numele de cod utilizate în fișierul de configurare după versiunea `trixie`.

adresa URL a arhivei	numele suitei	nume în cod	scopul depozitului
http://deb.debian.org/debian/	`stable`	`trixie`	Publicare `stabilă` cvasi-statică după verificări ample
http://deb.debian.org/debian/	`stable-updates`	`trixie-updates`	Actualizări urgente pentru versiunea `stabilă` (important)
http://security.debian.org/debian-security/	`stable-security`	`trixie-security`	Actualizări de securitate pentru versiunea `stabilă` (important)
http://deb.debian.org/debian/	`testing`	`forky`	Publicare dinamică a suitei de `testare` după verificări adecvate și perioade scurte de așteptare
http://deb.debian.org/debian/	`unstable`	`sid`	Publicare dinamică a suitei `instabile` după verificări minime și fără așteptări
http://deb.debian.org/debian/	`experimental`	N/D	Experimentări pre-publicare efectuate de dezvoltatori (opțional, numai pentru dezvoltatori)
http://deb.debian.org/debian/	`stable-proposed-updates`	`trixie-proposed-updates`	Actualizări propuse pentru următoarea versiune `stabilă` (opțional)
http://deb.debian.org/debian/	`stable-backports`	`trixie-backports`	Colecție aleatorie de pachete recompilate, în mare parte din versiunea `testing` (opțional)
http://security.debian.org/debian-security/	`testing-security`	`forky-security`	Aceasta nu este susținută în mod activ și nici utilizată de echipa de securitate

Tabela 2.2: Lista siturilor de arhivă Debian



Atenție
Numai versiunea pură **`stabilă`** cu actualizări de securitate oferă cea mai bună stabilitate. Rularea în principal a versiunii **`stable`** combinată cu unele pachete din versiunea **`testing`** sau **`unstable`** este mai riscantă decât rularea versiunii pure **`unstable`** din cauza incompatibilității versiunilor bibliotecilor etc. Dacă aveți cu adevărat nevoie de cea mai recentă versiune a unor programe din versiunea **`stable`**, vă rugăm să utilizați pachetele din serviciile stable-updates și backports (a se vedea Secțiune 2.7.4). Aceste servicii trebuie utilizate cu precauție sporită.



Atenție
În principiu, ar trebui să enumerați doar una dintre suitele `stable`, `testing` sau `unstable` în linia „deb”. Dacă listați orice combinație de suite `stable`, `testing` și `unstable` în linia „deb”, programele APT vor funcționa mai lent, iar numai arhiva cea mai recentă va fi eficientă. Listarea multiplă are sens pentru acestea atunci când fișierul „`/etc/apt/preferences`” este utilizat cu obiective clare (a se vedea Secțiune 2.7.7).

Indicație
Pentru sistemul Debian cu suita `stable`, este recomandat să includeți conținutul cu „`http://security.debian.org/`” în **lista de surse** pentru a activa actualizările de securitate, așa cum se arată în exemplul de mai sus.

Notă
Erorile/problemele de securitate pentru arhiva `stable` sunt remediate de echipa de securitate Debian. Această activitate a fost destul de riguroasă și fiabilă. Cele pentru arhiva `testing` pot fi remediate de echipa de securitate Debian testing. Din mai multe motive, această activitate nu este la fel de riguroasă ca cea pentru `stable` și este posibil să fie necesar să așteptați migrarea pachetelor `unstable` remediate către arhiva `testing`. Cele pentru arhiva `unstable` sunt remediate de către administratorul individual. Pachetele `unstable` întreținute activ sunt de obicei într-o stare destul de bună, beneficiind de cele mai recente remedieri de securitate din amonte. Consultați FAQ-ul de securitate Debian pentru a afla cum gestionează Debian erorile/problemele de securitate.

secțiunea	**numărul de pachete**	**criterii privind componentele pachetului**
`main`	76587	conform cu DFSG și fără dependență de `non-free`
`non-free-firmware`	78	nu este conform cu DFSG, necesită firmware pentru o experiență rezonabilă de instalare a sistemului
`contrib`	362	conform cu DFSG, dar cu dependență de `non-free`
`non-free`	1045	nu este conform cu DFSG și nu se află în `non-free-firmware`

Tabela 2.3: Lista secțiunilor de arhivă Debian

Aici numărul de pachete din cele de mai sus este pentru arhitectura amd64. Secțiunea `main` furnizează sistemul Debian (vedeți Secțiune 2.1.6).

Organizarea arhivei Debian poate fi studiată cel mai bine accesând în navigatorul dvs. fiecare adresă URL a arhivei la care se adaugă `dists` sau `pool`.

Distribuția este menționată în două moduri, versiunea sau nume în cod. Termenul „distribuție” este utilizat alternativ ca sinonim pentru „suită” în multe documentații. Relația dintre versiune și numele în cod poate fi rezumată după cum urmează.

Istoria numelor în cod este descrisă în Debian FAQ: 6.2.1 Ce alte nume în cod au fost folosite în trecut?

Calendarul	versiunea = `stable`	versiunea = `testing`	versiunea = `unstable`
după publicarea versiunii `trixie`	nume în cod = `trixie`	nume în cod = `forky`	nume în cod = `sid`
după publicarea versiunii `forky`	nume în cod = `forky`	nume în cod = `duke`	nume în cod = `sid`

Tabela 2.4: Relația dintre versiune și numele în cod

În terminologia mai strictă a arhivei Debian, cuvântul „secțiune” este utilizat în mod specific pentru clasificarea pachetelor în funcție de domeniul de aplicare. (Cu toate acestea, cuvântul „secțiune principală” poate fi utilizat uneori pentru a descrie zona arhivei Debian denumită „main”.)

De fiecare dată când un dezvoltator Debian (DD) efectuează o nouă încărcare în arhiva `unstable` (prin procesarea incoming), DD trebuie să se asigure că pachetele încărcate sunt compatibile cu cel mai recent set de pachete din cea mai recentă arhivă `unstable`.

Dacă DD încalcă intenționat această compatibilitate pentru actualizări importante ale bibliotecii etc., de obicei se face un anunț pe lista de discuții debian-devel etc.

Înainte ca un set de pachete să fie mutat de scriptul de întreținere a arhivei Debian din arhiva `unstable` în arhiva `testing`, scriptul de întreținere a arhivei nu numai că verifică maturitatea (aproximativ 2-10 zile) și starea rapoartelor de erori RC pentru pachete, dar încearcă și să se asigure că acestea sunt compatibile cu cel mai recent set de pachete din arhiva `testing`. Acest proces face ca arhiva `testing` să fie foarte actuală și utilizabilă.

Prin procesul gradual de înghețare a arhivei condus de echipa de lansare, arhiva `testing` este maturizată pentru a deveni complet consistentă și fără erori, cu câteva intervenții manuale. Apoi, noua versiune `stable` este creată prin atribuirea numelui în cod al vechii arhive `testing` noii arhive `stable` și crearea unui nou nume în cod pentru noua arhivă `testing`. Conținutul inițial al noii arhive `testing` este exact același cu cel al arhivei `stable` recent lansate.

Atât arhiva `unstable` cât și arhiva `testing` pot suferi defecțiuni temporare din cauza mai multor factori.

- Încărcare pachet defect în arhivă (în special pentru `unstable`)
- Întârzierea acceptării noilor pachete în arhivă (în special pentru `unstable`)
- Problemă de sincronizare a arhivei (atât pentru `testing` cât și pentru `unstable`)
- Intervenția manuală asupra arhivei, cum ar fi eliminarea pachetelor (mai mult pentru `testing`) etc.

Deci, dacă vreodată decideți să utilizați aceste arhive, ar trebui să puteți remedia sau să rezolvați acest tip de probleme.



Atenție
Timp de aproximativ câteva luni după lansarea unei noi versiuni `stable` (stabilă), majoritatea utilizatorilor de stații de lucru ar trebui să utilizeze arhiva `stable` (stabilă) cu actualizările sale de securitate, chiar dacă de obicei utilizează arhivele `unstable` (instabilă) sau `testing` (în testare). În această perioadă de tranziție, arhivele `unstable` și `testing` nu sunt potrivite pentru majoritatea utilizatorilor. Este dificil să mențineți sistemul în stare bună de funcționare cu arhiva `unstable`, deoarece aceasta suferă actualizări majore pentru pachetele de bază. Arhiva `testing` nu este utilă nici ea, deoarece conține în mare parte același conținut ca arhiva `stable`, fără suportul de securitate (Debian testing-security-announce 2008-12). După aproximativ o lună, arhivele `unstable` sau `testing` pot deveni utile dacă aveți grijă.

Indicație
Când se urmărește arhiva `testing`, o problemă cauzată de un pachet eliminat este de obicei rezolvată prin instalarea pachetului corespunzător din arhiva `unstable`, care este încărcată pentru remedierea erorii.

Consultați Manualul de politici Debian pentru definițiile arhivelor.

- "Secțiuni"
- "Priorități"
- "Sistem de bază"
- "Pachete esențiale"

2.1.6 Debian este software 100% liber

Debian este software 100% liber datorită următoarelor aspecte:

- Debian instalează în mod implicit numai software liber, pentru a respecta libertățile utilizatorilor.
- Debian oferă numai software liber în `main`.
- Debian recomandă rularea numai a software-ului liber din `main`.
- Niciun pachet din `main` nu depinde și nu recomandă pachete din `non-free`, `non-free-firmware` sau `contrib`.

Unii oameni se întreabă dacă următoarele două fapte se contrazic sau nu.

- „Debian va rămâne 100% gratuit". (Primul termen din Contractul social Debian)
- Serverele Debian găzduiesc unele pachete `non-free-firmware`, `non-free` și `contrib`.

Acestea nu se contrazic, din următoarele motive.

- Sistemul Debian este 100% liber, iar pachetele sale sunt găzduite de serverele Debian în secțiunea `main`.
- Pachetele din afara sistemului Debian sunt găzduite de serverele Debian în zonele `non-free`, `non-free-firmware` și `contrib`.

Acestea sunt explicate în detaliu în termenii 4 și 5 din Contractul social Debian:

- Prioritățile noastre sunt utilizatorii și software-ul liber
  - Ne vom ghida după nevoile utilizatorilor noștri și ale comunității de software liber. Vom pune interesele lor pe primul loc în lista noastră de priorități. Vom sprijini nevoile utilizatorilor noștri de a opera în multe tipuri diferite de medii informatice. Nu ne vom opune lucrărilor care nu sunt libere și care sunt destinate utilizării pe sistemele Debian și nu vom încerca să percepem taxe persoanelor care creează sau utilizează astfel de lucrări. Vom permite altora să creeze distribuții care conțin atât sistemul Debian, cât și alte lucrări, fără a percepe nicio taxă din partea noastră. În vederea promovării acestor obiective, vom furniza un sistem integrat de materiale de înaltă calitate, fără restricții legale care ar împiedica astfel de utilizări ale sistemului.
- Lucrările care nu respectă standardele noastre privind software-ul liber
  - Recunoaștem că unii dintre utilizatorii noștri necesită utilizarea unor lucrări care nu sunt conforme cu Ghidul Debian pentru software liber. Am creat secțiunile „`non-free`", „`non-free-firmware`" și „`contrib`" în arhiva noastră pentru aceste lucrări. Pachetele din aceste secțiuni nu fac parte din sistemul Debian, deși au fost configurate pentru a fi utilizate cu Debian. Încurajăm producătorii de CD-uri să citească licențele pachetelor din aceste secțiuni și să determine dacă pot distribui pachetele pe CD-urile lor. Astfel, deși lucrările non-free nu fac parte din Debian, noi susținem utilizarea lor și oferim infrastructura necesară pentru pachetele non-free (cum ar fi sistemul nostru de urmărire a erorilor și listele de discuții). Suporturile media oficiale Debian pot include firmware care altfel nu face parte din sistemul Debian, pentru a permite utilizarea Debian cu hardware care necesită un astfel de firmware.

Notă
Textul actual al celui de-al cincilea termen din actualul Contract social Debian 1.2 este ușor diferit de textul de mai sus. Această abatere editorială este intenționată, pentru a asigura coerența acestui document destinat utilizatorilor, fără a modifica conținutul real al Contractului social.

Utilizatorii trebuie să fie conștienți de riscurile utilizării pachetelor din secțiunile `non-free`, `non-free-firmware` și `contrib`:

- lipsa de libertate pentru astfel de pachete software
- lipsa de asistență din partea Debian pentru astfel de pachete software (Debian nu poate oferi asistență adecvată pentru software fără a avea acces la codul sursă al acestuia)
- contaminarea sistemului Debian 100% liber

Ghidul Debian pentru software-ul liber reprezintă standardele de software liber pentru Debian. Debian interpretează „software-ul” în sensul cel mai larg, incluzând documentele, firmware-ul, logo-ul și datele grafice din pachet. Acest lucru face ca standardele Debian pentru software-ul liber să fie foarte stricte.

Pachetele tipice `non-free`, `non-free-firmware` și `contrib` includ pachete distribuibile liber de următoarele tipuri:

- Pachete de documente sub GNU Free Documentation License cu secțiuni invariabile, cum ar fi cele pentru GCC și Make. (se găsesc în principal în secțiunea `non-free/doc`.)
- Pachete firmware care conțin date binare fără sursă, cum ar fi cele enumerate în Secțiune 9.10.5 ca `non-free-firmware` (se găsesc în principal în secțiunea `non-free-firmware/kernel`).
- Pachete de jocuri și fonturi cu restricții privind utilizarea comercială și/sau modificarea conținutului.

Vă rugăm să rețineți că numărul pachetelor `non-free`, `non-free-firmware` și `contrib` este mai mic de 2% din numărul pachetelor `main`. Activarea accesului la secțiunile `non-free`, `non-free-firmware` și `contrib` nu ascunde sursa pachetelor. Utilizarea interactivă pe ecran complet a aptitude(8) vă oferă vizibilitate și control deplin asupra pachetelor instalate din fiecare secțiune, pentru a vă menține sistemul atât de liber pe cât doriți.

2.1.7 Dependențele pachetelor

Sistemul Debian oferă un set consistent de pachete binare prin mecanismul său de declarare a dependențelor binare cu versiuni în câmpurile fișierului de control. Iată o definiție puțin simplificată pentru acestea.

- „Depends” (depinde de)
  - Aceasta declară o dependență absolută și toate pachetele enumerate în acest câmp trebuie instalate în același timp sau în prealabil.
- „Pre-Depends” (pre-depinde de)
  - Este similar cu Depends, cu excepția faptului că necesită instalarea completă în prealabil a pachetelor enumerate.
- „Recommends” (recomandate)
  - Aceasta declară o dependență puternică, dar nu absolută. Majoritatea utilizatorilor nu ar dori pachetul decât dacă toate pachetele enumerate în acest câmp sunt instalate.
- „Suggest” (sugerate)
  - Aceasta declară o dependență slabă. Mulți utilizatori ai acestui pachet pot beneficia de instalarea pachetelor enumerate în acest câmp, dar pot avea funcții rezonabile și fără acestea.

- „Enhances” (îmbunătățește)
  - Aceasta declară o dependență slabă, similară cu Suggests, dar funcționează în sens invers.
- „Breaks” (deteriorează)
  - Aceasta declară o incompatibilitate a pachetului, de obicei cu o anumită specificație de versiune. În general, soluția este să actualizați toate pachetele enumerate în acest câmp.
- „Conflicts” (conflicte)
  - Aceasta declară o incompatibilitate absolută. Toate pachetele enumerate în acest câmp trebuie eliminate pentru a instala acest pachet.
- „Replaces” (înlocuiește)
  - Aceasta se declară atunci când fișierele instalate de acest pachet înlocuiesc fișierele din pachetele enumerate.
- „Provides” (furnizează)
  - Acest lucru este declarat atunci când acest pachet furnizează toate fișierele și funcționalitățile din pachetele enumerate.

Notă
Vă rugăm să rețineți că definirea simultană a „Provides”, „Conflicts” și „Replaces” pentru un pachet virtual este configurația corectă. Aceasta asigură că doar un singur pachet real care furnizează acest pachet virtual poate fi instalat la un moment dat.

Definiția oficială, inclusiv dependența de sursă, poate fi găsită în Manualul de politici: Capitolul 7 - Declararea relațiilor dintre pachete.

2.1.8 Fluxul evenimentelor din gestionarea pachetelor

Iată un rezumat al fluxului simplificat al evenimentelor din gestionarea pachetelor de către APT.

- **Actualizare metadate** („`apt update`”, „`aptitude update`” sau „`apt-get update`”):
  1. Preluarea metadatelor din arhiva de la distanță
  2. Reconstruirea și actualizarea metadatelor locale pentru utilizarea de către APT
- **Modernizare (actualizarea la ultima versiune a pachetelor)** („`apt upgrade`” și „`apt full-upgrade`” sau „`aptitude safe-upgrade`” și „`aptitude full-upgrade`”, sau „`apt-get upgrade`” și „`apt-get dist-upgrade`”
  1. Alegeți versiunea candidată, care este de obicei cea mai recentă versiune disponibilă pentru toate pachetele instalate (a se vedea Secțiune 2.7.7 pentru excepții)
  2. Rezolvarea dependențelor pachetelor
  3. Preluarea pachetelor binare selectate din arhiva de la distanță dacă versiunea candidată este diferită de versiunea instalată
  4. Despachetarea pachetelor binare descărcate
  5. Rularea scriptului **preinst**
  6. Instalarea fișierelor binare
  7. Rularea scriptului **postinst**
- **Instalare** („`apt install …`”, `aptitude install …`” sau „`apt-get install …`”):
  1. Alegerea pachetelor listate în linia de comandă

2. Rezolvarea dependențelor pachetelor
3. Preluarea pachetelor binare selectate din arhiva de la distanță
4. Despachetarea pachetelor binare descărcate
5. Rularea scriptului **preinst**
6. Instalarea fișierelor binare
7. Rularea scriptului **postinst**

- **Eliminare** („`apt remove …`”, „`aptitude remove …`” sau „`apt-get remove …`”):
  1. Alegerea pachetelor listate în linia de comandă
  2. Rezolvarea dependențelor pachetelor
  3. Rularea scriptului **prerm**
  4. Eliminarea fișierelor instalate **cu excepția** fișierelor de configurare
  5. Rularea scriptului **postrm**
- **Purjare (curățare)** («`apt purge`», «`aptitude purge …`» sau «`apt-get purge …`»):
  1. Alegerea pachetelor listate în linia de comandă
  2. Rezolvarea dependențelor pachetelor
  3. Rularea scriptului **prerm**
  4. Eliminarea fișierelor instalate **inclusiv** fișierele de configurare
  5. Rularea scriptului **postrm**

Aici, am omis intenționat detaliile tehnice pentru a prezenta imaginea de ansamblu.

2.1.9 Primul răspuns la problemele legate de gestionarea pachetelor

Ar trebui să citiți documentația oficială detaliată. Primul document pe care trebuie să-l citiți este cel specific Debian „`/usr/share/doc/`*`nume_pachet`*`/README.Debian`”. Ar trebui consultată și altă documentație din „`/usr/share/doc/`*`n`*
Dacă configurați shell-ul ca în Secțiune 1.4.2, tastați următoarele.

```
$ cd package_name
$ pager README.Debian
$ mc
```

Pentru informații detaliate, poate fi necesar să instalați pachetul de documentație corespunzător, denumit cu sufixul „`-doc`”.

Dacă întâmpinați probleme cu un pachet specific, verificați mai întâi siturile sistemului de urmărire a erorilor Debian (BTS).

situl web	**comandă**
Pagina principală a sistemului de urmărire a erorilor Debian (BTS)	`sensible-browser "https://bugs.debian.org/"`
Raportul de eroare al unui pachet cunoscut	`sensible-browser "https://bugs.debian.org/`*`package_name`*`"`
Raportul de eroare al erorii cunoscute cu numărul	`sensible-browser "https://bugs.debian.org/`*`bug_number`*`"`

Tabela 2.5: Lista siturilor web importante pentru rezolvarea problemelor legate de un pachet specific

Căutați cu Google cu cuvinte cheie precum „`site:debian.org`”, „`site:wiki.debian.org`”, „`site:lists.debian.or`
etc.

Când raportați o eroare, vă rugăm să utilizați comanda reportbug(1).

2.1.10 Cum să selectați pachetele Debian

Când întâlniți mai mult de două pachete similare și vă întrebați pe care să îl instalați fără a recurge la „încercări și erori”, ar trebui să folosiți **bunul simț**. Consider că următoarele puncte sunt indicii bune pentru alegerea pachetelor preferate.

- Esențial: da > nu
- Secțiunea: main > contrib > non-free
- Prioritate: necesar > important> standard> opțional> suplimentar
- Sarcini: pachete enumerate în sarcini precum „Mediu de birou”
- Pachetele selectate de pachetul de dependență (de exemplu, `gcc-16` de către `gcc`)
- Popcon (popularitate): număr mai mare de voturi și instalări
- Istoric modificări: actualizări periodice de către responsabilul pachetului
- BTS: Fără erori RC (fără erori critice, grave sau serioase)
- BTS: responsabil de pachet receptiv la rapoartele de erori
- BTS: număr mai mare de erori remediate recent
- BTS: număr mai mic de erori rămase care nu sunt incluse ca listă de dorințe (wishlist)

Debian fiind un proiect de voluntariat cu un model de dezvoltare distribuit, arhiva sa conține multe pachete cu orientări și calități diferite. Trebuie să decideți singur ce să faceți cu ele.

2.1.11 Cum să faceți față cerințelor contradictorii

Indiferent de suita sistemului Debian pe care decideți să o utilizați, este posibil să doriți să rulați versiuni ale programelor care nu sunt disponibile în acea suită. Chiar dacă găsiți pachete binare ale acestor programe în alte suite Debian sau în alte resurse non-Debian, cerințele lor pot intra în conflict cu sistemul Debian actual.

Deși puteți modifica sistemul de gestionare a pachetelor cu tehnica **apt-pinning** etc., așa cum este descris în Secțiune 2.7.7 pentru a instala astfel de pachete binare nesincronizate, astfel de abordări de modificare au doar cazuri de utilizare limitate, deoarece pot afecta programele respective și sistemul dvs.

Înainte de a instala în mod brutal astfel de pachete nesincronizate, ar trebui să căutați toate soluțiile tehnice alternative mai sigure disponibile, care sunt compatibile cu sistemul Debian actual.

- Instalați astfel de programe utilizând pachetele binare corespunzătoare din amonte (vezi Secțiune 7.7).
  - Multe programe cu interfață grafică, precum LibreOffice și aplicațiile GNOME, sunt disponibile sub formă de pachete Flatpak, Snap sau AppImage.
- Creați un mediu chroot sau similar și rulați astfel de programe în acesta (consultați Secțiune 9.11).
  - Comenzile CLI pot fi executate cu ușurință în cadrul chroot compatibil (vedeți Secțiune 9.11.4).
  - Mai multe medii grafice de birou complete pot fi încercate cu ușurință fără a reporni sistemul (a se vedea Secțiune 9.11.5).
- Construiți singur versiunile dorite ale pachetelor binare care sunt compatibile cu sistemul Debian actual.
  - Aceasta este o sarcină non-trivială (a se vedea Secțiune 2.7.13).

2.2 Operații de bază de gestionare a pachetelor

Operațiile de gestionare a pachetelor bazate pe depozit în sistemul Debian pot fi efectuate de numeroase instrumente de gestionare a pachetelor bazate pe APT disponibile în sistemul Debian. Aici, explicăm 3 instrumente de bază pentru gestionarea pachetelor: `apt`, `apt-get` / `apt-cache` și `aptitude`.

Pentru operația de gestionare a pachetelor care implică instalarea pachetelor sau actualizarea metadatelor pachetelor, trebuie să aveți privilegii de root.

2.2.1 `apt` vs. `apt-get` / `apt-cache` vs. `aptitude`

Deși `aptitude` este un instrument interactiv foarte bun, pe care autorul îl folosește în principal, trebuie să știți câteva lucruri importante:

- Comanda `aptitude` nu este recomandată pentru actualizarea sistemului de la o versiune la alta pe sistemul Debian `stabil` după lansarea noii versiuni.
  - Pentru aceasta se recomandă utilizarea „`apt full-upgrade`” sau „`apt-get dist-upgrade`”. Consultați Eroarea #411280.
- Comanda `aptitude` sugerează uneori eliminarea în masă a pachetelor pentru actualizarea sistemului pe sistemul Debian `testing` sau `unstable`.
  - Această situație a speriat mulți administratori de sistem. Nu intrați în panică.
  - Aceasta pare să fie cauzată în principal de diferențele de versiune între pachetele dependente sau recomandate de un meta-pachet precum `gnome-core`.
  - Această problemă poate fi rezolvată selectând „Anulează acțiunile în așteptare” din meniul de comenzi `aptitude`, ieșind din `aptitude` și utilizând „`apt full-upgrade`”.

Comenzile `apt-get` și `apt-cache` sunt cele mai **elementare** instrumente de gestionare a pachetelor bazate pe APT.

- `apt-get` și `apt-cache` oferă doar interfața de utilizare a liniei de comandă.
- `apt-get` este cel mai potrivit pentru **actualizări majore ale sistemului** între versiuni etc.
- `apt-get` oferă un rezolvator de dependențe de pachete **robust**.
- `apt-get` solicită mai puține resurse hardware. Consumă mai puțină memorie și rulează mai rapid.
- `apt-cache` oferă o căutare **standard** bazată pe expresii regulate pentru numele și descrierea pachetului.
- `apt-get` și `apt-cache` pot gestiona mai multe versiuni de pachete folosind `/etc/apt/preferences`, dar este destul de complicat.

Comanda `apt` este o interfață de linie de comandă de nivel înalt pentru gestionarea pachetelor. Este practic un înveliș pentru `apt-get`, `apt-cache` și comenzi similare, concepută inițial ca interfață pentru utilizatorul final și care activează în mod implicit unele opțiuni mai potrivite pentru utilizarea interactivă.

- `apt` oferă o bară de progres intuitivă atunci când instalați pachete folosind `apt install`.
- `apt` va **elimina** pachetele `.deb` stocate în cache în mod implicit după instalarea cu succes a pachetelor descărcate.

Indicație
Utilizatorilor li se recomandă să utilizeze noua comandă apt(8) pentru utilizarea **interactivă** și să utilizeze comenzile apt-get(8) și apt-cache(8) într-un script shell.

Comanda `aptitude` este cel mai **versatil** instrument de gestionare a pachetelor bazat pe APT.

- `aptitude` oferă interfața text interactivă cu utilizatorul pe ecran complet.
- `aptitude` oferă și interfața de utilizare a liniei de comandă.
- `aptitude` este cel mai potrivit pentru **gestionarea interactivă zilnică a pachetelor**, cum ar fi inspectarea pachetelor instalate și căutarea pachetelor disponibile.
- `aptitude` este mai exigent în ceea ce privește resursele hardware. Consumă mai multă memorie și rulează mai lent.
- `aptitude` oferă o căutare **îmbunătățită** bazată pe expresii regulate pentru toate metadatele pachetului.
- `aptitude` poate gestiona mai multe versiuni ale pachetelor fără a utiliza `/etc/apt/preferences` și este destul de intuitiv.

2.2.2 Operații de bază de gestionare a pachetelor din linia de comandă

Iată operațiile de bază pentru gestionarea pachetelor din linia de comandă, folosind apt(8), aptitude(8) și apt-get(8) /apt-cache(8).

`apt` / `apt-get` și `aptitude` pot fi combinate fără probleme majore.

„`aptitude why` *`expresie-regulată`*” poate afișa mai multe informații prin „`aptitude -v why` *`expresie-regulat`*
Informații similare pot fi obținute prin „`apt rdepends` *`pachet`*” sau „`apt-cache rdepends` *`pachet`*”.

Când comanda `aptitude` este pornită în modul linie de comandă și întâmpină unele probleme, cum ar fi conflicte între pachete, puteți trece la modul interactiv pe ecran complet apăsând tasta «e» în prompt.

Notă
Deși comanda `aptitude` vine cu funcții bogate, cum ar fi rezolvatorul de pachete îmbunătățit, această complexitate a cauzat (sau poate încă cauzează) unele regresii, cum ar fi Eroarea #411123, Eroarea #514930 și Eroarea #570377. În caz de îndoială, vă rugăm să utilizați comenzile `apt`, `apt-get` și `apt-cache` în locul comenzii `aptitude`.

Puteți furniza opțiuni de comandă imediat după „`aptitude`”.

Pentru mai multe informații, consultați aptitude(8) și „Manualul utilizatorului aptitude” la „`/usr/share/doc/aptitude/REA`

2.2.3 Utilizarea interactivă a «aptitudie»

Pentru gestionarea interactivă a pachetelor, porniți `aptitude` în modul interactiv din promptul consolei shell, după cum urmează.

```
$ sudo aptitude -u
Password:
```

Aceasta actualizează copia locală a informațiilor din arhivă și afișează lista pachetelor pe ecranul complet cu meniu. Aptitude plasează configurația sa în „`~/.aptitude/config`”.

Indicație
Dacă doriți să utilizați configurația root în locul celei a utilizatorului, utilizați „`sudo -H aptitude` …” în locul „`sudo aptitude` …” în expresia de mai sus.

Indicație
`Aptitude` stabilește automat **acțiunile în așteptare** atunci când este pornit interactiv. Dacă nu vă place, puteți să-l reinițializați din meniu: „Acțiune” → „Anulează acțiunile în așteptare”.

---

sintaxa `apt`	sintaxa `aptitude`	sintaxa `apt-get`/`apt-cache`	descriere
`apt update`	`aptitude update`	`apt-get update`	actualizează metadatele arhivei pachetelor
`apt install foo`	`aptitude install foo`	`apt-get install foo`	instalează versiunea candidată a pachetului „`foo`” împreună cu dependențele sale
`apt upgrade`	`aptitude safe-upgrade`	`apt-get upgrade`	instalează versiunea candidată a pachetelor instalate fără a elimina alte pachete
`apt full-upgrade`	`aptitude full-upgrade`	`apt-get dist-upgrade`	instalați versiunea candidată a pachetelor instalate, eliminând alte pachete dacă este necesar
`apt remove foo`	`aptitude remove foo`	`apt-get remove foo`	eliminați pachetul „`foo`” lăsând în același timp fișierele sale de configurare
`apt autoremove`	N/D	`apt-get autoremove`	elimină pachetele instalate automat care nu mai sunt necesare
`apt purge foo`	`aptitude purge foo`	`apt-get purge foo`	șterge pachetul „`foo`” împreună cu fișierele sale de configurare
`apt clean`	`aptitude clean`	`apt-get clean`	șterge complet depozitul local de fișiere ale pachetelor preluate (descărcate)
`apt autoclean`	`aptitude autoclean`	`apt-get autoclean`	șterge din depozitul local fișierele pachetelor descărcate pentru pachetele învechite
`apt show foo`	`aptitude show foo`	`apt-cache show foo`	afișează informații detaliate despre pachetul „`foo`”
`apt search` *`regex`*	`aptitude search` *`regex`*	`apt-cache search` *`regex`*	căută pachetele care corespund cu *expresia-regulată*
N/D	`aptitude why` *`regex`*	N/D	explică motivul pentru care ar trebui instalate pachetele care corespund cu *expresia-regulată*
N/D	`aptitude why-not` *`regex`*	N/D	explică motivul pentru care pachetele care corespund cu *expresia-regulată* nu pot fi instalate
`apt list --manual-installed`	`aptitude search '~i!~M'`	`apt-mark showmanual`	listează pachetele instalate manual

Tabela 2.6: Operații de bază de gestionare a pachetelor din linia de comandă folosind apt(8), aptitude(8) și apt-get(8) /apt-cache(8)

opțiunea de comandă	descriere
`-s`	simulează rezultatul comenzii
`-d`	doar descărcare, fără instalare/actualizare
`-D`	afișează explicații succinte înainte de instalările și dezinstalările automate

Tabela 2.7: Opțiuni de comandă demne de menționat pentru aptitude(8)

2.2.4 Combinații de taste pentru aptitude

Tastele demne de menționat pentru a naviga în starea pachetelor și pentru a defini „acțiunea planificată” pentru acestea în acest mod ecran complet sunt următoarele.

tasta	**comanda/funcția asociată**
`F10` sau `Ctrl-t`	meniu
`?`	afișează **ajutorul** pentru apăsarea tastei (listă mai completă)
`F10` → Ajutor → Manualul utilizatorului	Afișează manualul utilizatorului
`u`	actualizează informațiile referitoare la arhiva pachetelor
`+`	marchează pachetul pentru **actualizare** sau **instalare**
`-`	marcați pachetul pentru **eliminare** (păstrează fișierele de configurare)
`_`	marchează pachetul pentru **ștergere** (elimină fișierele de configurare)
`=`	pune pachetul în **așteptare**
`U`	marchează toate pachetele care pot fi actualizate (funcționează ca **full-upgrade**)
`g`	începe **descărcarea** și **instalarea** pachetelor selectate
`q`	iese din ecranul curent și salvează modificările
`x`	iese din ecranul curent și înlătură modificările
`Enter`	afișează informații despre un pachet
`C`	afișează jurnalul de modificări al unui pachet
`l`	modifică limita pentru pachetele afișate
`/`	caută prima potrivire
`\`	repetă ultima căutare

Tabela 2.8: Lista combinațiilor de taste pentru «aptitude»

Specificația numelui fișierului din linia de comandă și promptul meniului după apăsarea „`l`” și „`//`” utilizează expresia regulată aptitude, așa cum este descrisă mai jos. Expresia regulată aptitude poate potrivi în mod explicit un nume de pachet utilizând un șir care începe cu „~n” și este urmat de numele pachetului.

Indicație
Trebuie să apăsați tasta „U” pentru a actualiza toate pachetele instalate la **versiunea candidată** în interfața vizuală. În caz contrar, numai pachetele selectate și anumite pachete cu dependențe de versiune față de acestea vor fi actualizate la **versiunea candidată**.

2.2.5 Vizualizarea de pachete în aptitude

În modul interactiv pe ecran complet al aptitude(8), pachetele din lista de pachete sunt afișate conform exemplului următor.

```
idA   libsmbclient                              -2220kB 3.0.25a-1  3.0.25a-2
```

Aici, această linie înseamnă de la stânga către dreapta, după cum urmează:

- Indicatorul „starea curentă” (prima literă)
- Indicatorul „acțiunea planificată” (a doua literă)
- Indicatorul „automat” (a treia literă)
- Numele pachetului

- Modificarea utilizării spațiului pe disc atribuită „acțiunii planificate”
- Versiunea actuală a pachetului
- Versiunea candidată a pachetului

Indicație
Lista completă a indicatorilor este afișată în partea de jos a ecranului **Ajutor**, care apare apăsând «?».

Versiunea candidată este aleasă în funcție de preferințele locale curente (vezi apt_preferences(5) și Secțiune 2.7.7).

Mai multe tipuri de vizualizări ale pachetelor sunt disponibile în meniul „`Vizualizări`”.

vizualizarea	**descrierea vizualizării**
`Package View`	vedeți Tabel 2.10 (implicit)
`Audit Recommendations`	lista pachetelor recomandate de unele pachete instalate, dar care nu sunt încă instalate
`Flat Package List`	lista pachetelor fără categorizare (pentru utilizare cu expresii regulate)
`Debtags Browser`	lista pachetelor clasificate în funcție de intrările lor debtags
`Source Package View`	lista pachetelor grupate după pachetele sursă

Tabela 2.9: Lista vizualizărilor pentru aptitude

Notă
Vă rugăm să ne ajutați să îmbunătățim etichetarea pachetelor cu debtags!

Standardul „`Vizualizare pachete`” clasifică pachetele într-un mod similar cu `dselect`, cu câteva caracteristici suplimentare.

categoria	**descrierea vizualizării**
`Upgradable Packages`	lista pachetele organizate ca `secțiune` → `zonă` → `pachet`
`New Packages`	,,
`Installed Packages`	,,
`Not Installed Packages`	,,
`Obsolete and Locally Created Packages`	,,
`Virtual Packages`	lista pachetelor cu aceeași funcție
`Tasks`	lista pachetelor cu diferite funcții necesare în general pentru o sarcină

Tabela 2.10: Clasificarea vizualizărilor standard ale pachetelor

Indicație
Vizualizarea `Sarcini` poate fi utilizată pentru a selecta pachetele potrivite pentru sarcina dvs.

2.2.6 Opțiuni pentru metoda de căutare cu aptitude

Aptitude oferă mai multe opțiuni pentru căutarea pachetelor folosind formula de expresie regulată.

- Linia de comandă shell:
  - «`aptitude search '`*`expresie-regulată_aptitude`*`'`» pentru a afișa starea instalării, numele pachetului și o scurtă descriere a pachetelor care corespund căutării
  - «`aptitude show '`*`nume_pachet`*`'`» pentru a afișa descrierea detaliată a pachetului
- Modul interactiv pe ecran complet:
  - „`l`” pentru a limita vizualizarea pachetelor la pachetele care se potrivesc căutării
  - „`/`” pentru căutarea unui pachet care corespunde modelului de căutare
  - „`\`” pentru căutarea înapoi a unui pachet care corespunde modelului de căutare
  - „`n`” pentru căutarea următoarei coincidențe
  - „`N`” pentru căutarea următoarei coincidențe (înapoi)

Indicație
Șirul pentru *nume_pachet* este tratat ca fiind exact identic cu numele pachetului, cu excepția cazului în care începe în mod explicit cu „~” pentru a fi considerat formulă de expresie regulată.

2.2.7 Formula expresiei regulate pentru aptitudine

Formula expresiei regulate aptitude este similară cu cea extinsă mutt **ERE** (a se vedea Secțiune 1.6.2), iar semnificațiile extensiilor speciale ale regulilor de potrivire specifice `aptitude` sunt următoarele.

- Partea expresie-regulată este aceeași **ERE** ca cea utilizată în instrumentele text tipice de tip Unix care utilizează „^”, „`.*`”, „$” etc., ca în egrep(1), awk(1) și perl(1).
- *tipul* de dependență este unul dintre (depends, predepends, recommends, suggests, conflicts, replaces, provides) care specifică interrelația dintre pachete.
- *Tipul* implicit de dependență este „depends”.

Indicație
Când *modelul_expresie-regulată* este un șir nul, plasați „~T” imediat după comandă.

Iată câteva scurtături.

- „`~P`*`termen`*” == „`~Dprovides:`*`termen`*”
- „`~C`*`termen`*” == „`~Dconflicts:`*`termen`*”
- „…`~W termen`” == „`(…|termen)`”

Utilizatorii familiarizați cu `mutt` învață repede, deoarece mutt a fost sursa de inspirație pentru sintaxa expresiilor. Consultați „CĂUTARE, LIMITARE ȘI EXPRESII - (SEARCHING, LIMITING, AND EXPRESSIONS)” în „Manualul utilizatorului” „`/usr/share/doc/aptitude/README`”.

Notă
Cu versiunea `lenny` a aptitude(8), noua sintaxă de **formă lungă**, cum ar fi „`?broken`”, poate fi utilizată pentru potrivirea expresiilor regulate în locul vechii sale echivalente de **formă scurtă** „~b”. Acum, caracterul spațiu „ ” este considerat unul dintre caracterele de terminare a expresiilor regulate, pe lângă caracterul tilde „~”. Consultați „Manualul utilizatorului” pentru noua sintaxă de **formă lungă**.

---

descrierea regulii extinse de potrivire	**formula expresiei regulate**
potrivire cu numele pachetului	`~n`*`regex_name`*
potrivire cu descrierea	`~d`*`regex_description`*
potrivire cu numele sarcinii	`~t`*`regex_task`*
potrivire cu debtag	`~G`*`regex_debtag`*
potrivire cu resposabilul pachetului	`~m`*`regex_maintainer`*
potrivire cu secțiunea pachetului	`~s`*`regex_section`*
potrivire cu versiunea pachetului	`~V`*`regex_version`*
potrivire cu distribuția	`~A{trixie,forky,sid}`
potrivire cu originea	`~O{debian,…}`
coincidență prioritate	`~p{extra,important,optional,required,standard}`
corespunde cu pachete esențiale	`~E`
corespunde cu pachete virtuale	`~v`
corespunde cu pachete noi	`~N`
potrivire cu acțiunea în așteptare	`~a{install,upgrade,downgrade,remove,purge,hold,keep}`
potrivire cu pachete instalate	`~i`
corespunde pachetelelor instalate cu marca **A** (pachete instalate automat)	`~M`
corespunde pachetelelor instalate fără marca **A** (pachete selectate de administrator)	`~i!~M`
potrivire cu pachete instalate și actualizabile	`~U`
potrivire cu pachete eliminate, dar nu șterse	`~c`
potrivire cu pachete eliminate, șterse sau care pot fi eliminate	`~g`
potrivire cu pachete care declară o dependență ruptă	`~b`
potrivire cu pachete care declară o dependență ruptă de *tipul*	`~B`*`type`*
corespunde cu *modelul* de pachete care declară o dependență de *tipul*	`~D[`*`type`*`:]`*`pattern`*
corespunde cu *modelul* de pachete care declară o dependență ruptă de *tipul*	`~DB[`*`type`*`:]`*`pattern`*
corespunde cu pachetele pentru care *modelul* de potrivire corespunde pachetelor care declară o dependență de *tipul*	`~R[`*`type`*`:]`*`pattern`*
corespunde cu pachetele pentru care *modelul* de potrivire corespunde pachetelor care declară o dependență ruptă de *tipul*	`~RB[`*`type`*`:]`*`pattern`*
corespunde cu pachetele de care depind alte pachete instalate	`~R~i`
corespunde cu pachetele de care nu depind alte pachete instalate	`!~R~i`
corespunde cu pachetele de care depind sau pe care le recomandă alte pachete instalate	`~R~i|~Rrecommends:~i`
potrivire cu *modelul* de pachet filtrat după versiune	`~S filter `*`pattern`*
corespunde cu toate pachetele (adevărat)	`~T`
nu corespunde cu niciun pachet (fals)	`~F`

Tabela 2.11: Lista formulelor de expresii regulate pentru aptitude

2.2.8 Rezolvarea dependențelor de către aptitude

Selectarea unui pachet în `aptitude` nu numai că extrage pachetele definite în lista „`Depends:`”, ci și pe cele definite în lista „`Recommends:`”, dacă meniul „`F10` → Opțiuni → Preferințe → Gestionarea dependențelor” este configurat corespunzător. Aceste pachete instalate automat sunt eliminate automat dacă nu mai sunt necesare în `aptitude`.

Indicatorul care controlează comportamentul „instalării automate” al comenzii `aptitude` poate fi, de asemenea, manipulat folosind comanda apt-mark(8) din pachetul `apt`.

2.2.9 Jurnale de activitate ale pachetelor

Puteți verifica istoricul activității pachetului în fișierele jurnal.

fișier	conținut
`/var/log/dpkg.log`	Jurnalul activității la nivel `dpkg` pentru activitățile tuturor pachetelor
`/var/log/apt/term.log`	Jurnalul activității APT generice
`/var/log/aptitude`	Jurnalul activității comenzii `aptitude`

Tabela 2.12: Fișierele jurnal pentru activitățile pachetului

În realitate, nu este atât de ușor să obții rapid informații relevante din aceste jurnale. Vedeți Secțiune 9.3.9 pentru o metodă mai simplă.

2.3 Exemple de operații cu aptitudine

Iată câteva exemple de operații cu aptitude(8).

2.3.1 Căutarea de pachete interesante

Puteți căuta pachete care să vă satisfacă nevoile cu ajutorul `aptitude` din descrierea pachetului sau din lista aflată sub „Sarcini”.

2.3.2 Listarea pachetelor care se potrivesc cu expresia regulată în numele pachetelor

Următoarea comandă listează pachetele cu potriviri de expresie regulată în numele pachetelor.

```
$ aptitude search '~n(pam|nss).*ldap'
p libnss-ldap - NSS module for using LDAP as a naming service
p libpam-ldap - Pluggable Authentication Module allowing LDAP interfaces
```

Acest lucru este foarte util pentru a găsi numele exact al unui pachet.

2.3.3 Navigarea prin rezultatul potrivirilor expresiei regulate

Expresia regulată „`~dipv6`” din vizualizarea „Listă nouă plană de pachete” cu promptul „`l`” limitează vizualizarea la pachetele cu descrierea care coincide cu expresia și vă permite să răsfoiți informațiile acestora în mod interactiv.

2.3.4 Stergerea definitivă a pachetelor eliminate

Puteți șterge toate fișierele de configurare rămase ale pachetelor eliminate.

Verificați rezultatele următoarei comenzi.

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

Dacă considerați că pachetele listate pot fi șterse, executați următoarea comandă.

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

Puteți face același lucru în modul interactiv pentru un control mai precis.

Introduceți expresia regulată „~c” în vizualizarea «Vizualizare pachet nou» cu promptul „l”. Aceasta limitează vizualizarea pachetului numai la pachetele care corespund expresiei regulate, adică „eliminate, dar nu șterse”. Toate aceste pachete care corespund expresiei regulate pot fi afișate apăsând „[” în titlurile de nivel superior.

Apoi apăsați „_” în titlurile de nivel superior, cum ar fi „Pachete neinstalate”. Numai pachetele care corespund expresiei regulate din titlu sunt marcate pentru a fi șterse. Puteți exclude unele pachete de la ștergere apăsând „=” în mod interactiv pentru fiecare dintre ele.

Această tehnică este destul de utilă și funcționează pentru multe alte taste de comandă.

2.3.5 Reorganizarea stării instalării automate/manuale

Iată cum ordonez starea instalării automate/manuale pentru pachete (după utilizarea unui program de instalare a pachetelor non-aptitude etc.).

1. Lansați `aptitude` în modul interactiv ca root.
2. Tastați „u”, „U”, „f” și „g” pentru a actualiza și a îmbunătăți lista de pachete și pachetele.
3. Tastați „l” pentru a introduce limita de afișare a pachetelor ca „`~i(~R~i|~Rrecommends:~i)`” și tastați „M” peste „`Pachete instalate`” ca instalate automat.
4. Tastați „l” pentru a introduce limita de afișare a pachetelor ca „`~required|~pimportant|~pstandard|~E`” și tastați „m” peste „`Pachete instalate`” ca instalate manual.
5. Tastați „l” pentru a introduce limita de afișare a pachetelor ca „`~i!~M`” și eliminați pachetele neutilizate tastând „-” peste fiecare dintre ele după ce le-ați expus tastând „[” peste „`Pachete instalate`”.
6. Tastați „l” pentru a introduce limita de afișare a pachetelor ca „`~i`”; apoi tastați „m” peste „`Sarcini`” pentru a marca pachetele respective ca fiind instalate manual.
7. Ieșiți din `aptitude`.
8. Lansați „`apt-get -s autoremove|less`” ca root pentru a verifica ce nu este utilizat.
9. Reporniți `aptitude` în modul interactiv și marcați pachetele necesare ca „m”.
10. Reporniți „`apt-get -s autoremove|less`” ca root pentru a verifica din nou că REMOVED conține numai pachetele așteptate.
11. Porniți „`apt-get autoremove|less`” ca root pentru a elimina automat pachetele neutilizate.

Acțiunea „m” asupra „`Tasks`” este opțională, pentru a preveni situații de eliminare în masă a pachetelor în viitor.

2.3.6 Actualizare la nivel de sistem

Notă
Când treceți la o nouă versiune etc., ar trebui să luați în considerare efectuarea unei instalări curate a noului sistem, chiar dacă Debian poate fi actualizat așa cum este descris mai jos. Acest lucru vă oferă șansa de a elimina fișierele inutile acumulate și vă expune la cea mai bună combinație de pachete recente. Desigur, ar trebui să faceți o copie de rezervă completă a sistemului într-un loc sigur (consultați Secțiune 10.2) înainte de a face acest lucru. Vă recomand să configurați un sistem de pornire dual utilizând partiții diferite pentru a avea o tranziție cât mai lină.

Puteți efectua o actualizare la nivel de sistem la o versiune mai nouă modificând conținutul **listei de surse** pentru a indica o nouă versiune și executând comanda „`apt update; apt dist-upgrade`”.

Pentru a actualiza de la `stable` la `testing` sau `unstable` în timpul ciclului de lansare `trixie`-ca-`stable`, înlocuiți „`trixie `” din exemplul **listei sursă** din Secțiune 2.1.5 cu „`forky`” sau „`sid`”.

În realitate, este posibil să întâmpinați unele complicații din cauza unor probleme legate de tranziția pachetelor, în principal din cauza dependențelor pachetelor. Cu cât diferența dintre versiuni este mai mare, cu atât este mai probabil să întâmpinați probleme mai mari. Pentru tranziția de la vechiul `stable` la noul `stable` după lansarea acestuia, puteți citi noile Note de lansare și să urmați procedura exactă descrisă în acestea pentru a minimiza problemele.

Când decideți să treceți de la `stable` la `testing` înainte de lansarea oficială, nu există Note de lansare care să vă ajute. Diferența dintre `stable` și `testing` ar putea fi destul de mare după lansarea anterioară a `stable`, ceea ce complică situația actualizării.

Ar trebui să luați măsuri de precauție pentru actualizarea completă, în timp ce colectați cele mai recente informații din lista de distribuție și folosiți bunul simț.

1. Citiți „Notele de lansare” anterioare.
2. Faceți o copie de rezervă a întregului sistem (în special a datelor și informațiilor de configurare).
3. Aveți la îndemână un suport de pornire pentru cazul în care se strică încărcătorul de pornire.
4. Informați utilizatorii cu privire la sistem cu suficient timp înainte.
5. Înregistrați activitatea de actualizare cu script(1).
6. Aplicați „unmarkauto” pachetelor necesare, de exemplu, „`aptitude unmarkauto vim`”, pentru a preveni eliminarea.
7. Reduceți la minimum pachetele instalate pentru a diminua riscul de conflicte între pachete, de exemplu, eliminați pachetele de sarcini de medii grafice de birou.
8. Eliminați fișierul „`/etc/apt/preferences`” (dezactivați **apt-pinning**).
9. Încercați să faceți actualizarea pas cu pas: `oldstable` → `stable` → `testing` → `unstable`.
10. Actualizați **lista surselor** pentru a indica numai noua arhivă și rulați „`aptitude update`”.
11. Instalați, opțional, mai întâi noile **pachete de bază**, de exemplu, «`aptitude install perl`».
12. Rulați comanda «`apt-get -s dist-upgrade`» pentru a evalua impactul.
13. În cele din urmă, executați comanda «`apt-get dist-upgrade`».



Atenție
Nu este recomandat să săriți peste versiunile majore Debian atunci când faceți actualizări între versiunile `stable` (stabile).



Atenție
În „Notele de lansare” anterioare, GCC, Linux Kernel, initrd-tools, Glibc, Perl, lanțul de instrumente APT etc. au necesitat o atenție specială pentru actualizarea la nivel de sistem.



Atenție
„Notele de lansare” pot să nu acopere toate cazurile posibile. Dacă modificați configurațiile de nivel inferior, următoarea actualizare poate eșua grav, generând „... eroare de segmentare după actualizare ...”.

Pentru actualizări zilnice în `unstable`, consultați Secțiune 2.4.3.

2.4 Operații avansate de gestionare a pachetelor

2.4.1 Operații avansate de gestionare a pachetelor din linia de comandă

Iată o listă cu alte opereții de gestionare a pachetelor pentru care `aptitude` este prea complex sau nu dispune de funcționalitățile necesare.

Notă
Pentru un pachet cu caracteristica multi-arch, poate fi necesar să specificați numele arhitecturii pentru unele comenzi. De exemplu, utilizați «`dpkg -L libglib2.0-0:amd64`» pentru a afișa conținutul pachetului `libglib2.0-0` pentru arhitectura `amd64`.



Atenție
Instrumentele de nivel inferior, precum „`dpkg -i ...`” și „`debi ...`”, trebuie utilizate cu precauție de către administratorul de sistem. Acestea nu se ocupă automat de dependențele necesare ale pachetelor. Opțiunile liniei de comandă Dpkg „`--force-all`” și similare (vezi dpkg(1)) sunt destinate utilizării numai de către experți. Utilizarea acestora fără a înțelege pe deplin efectele lor poate duce la deteriorarea întregului sistem.

Vă rugăm să rețineți următoarele:

- Toate comenzile de configurare și instalare a sistemului trebuie rulate din contul root.
- Spre deosebire de `aptitude`, care utilizează expresii regulate (a se vedea Secțiune 1.6.2), alte comenzi de gestionare a pachetelor utilizează modele precum modelele globale de shell „shell glob” (a se vedea Secțiune 1.5.6).
- apt-file(1) furnizat de pachetul `apt-file` trebuie să execute în prealabil comanda «`apt-file update`».
- configure-debian(8) furnizat de pachetul `configure-debian` rulează dpkg-reconfigure(8) ca motor.
- dpkg-reconfigure(8) rulează scripturi de pachete folosind debconf(1) ca motor.
- Comenzile «`apt-get build-dep`», «`apt-get source`» și «`apt-cache showsrc`» necesită introducerea unei intrări „`deb-src`” în **lista surselor**.
- dget(1), debuild(1) și debi(1) necesită pachetul `devscripts`.
- Consultați procedura de (re)împachetare folosind «`apt-get source`» în Secțiune 2.7.13.
- Comanda `make-kpkg` necesită pachetul `kernel-package` (a se vedea Secțiune 9.10).
- Consultați Secțiune 12.9 pentru informații generale despre împachetare.

comandă	**acțiunea**
`COLUMNS=120 dpkg -l` *`package_name_pattern`*	afișează starea unui pachet instalat pentru raportul de eroare
`dpkg -L` *`package_name`*	afișează conținutul unui pachet instalat
`dpkg -L` *`package_name`* `| egrep '/usr/share/man/man.*/.+'`	listează paginile de manual pentru un pachet instalat
`dpkg -S` *`file_name_pattern`*	listează pachetele instalate care au numele de fișier ce coincide cu modelul
`apt-file search` *`file_name_pattern`*	listează pachetele din arhivă care au numele de fișier ce coincide cu modelul
`apt-file list` *`package_name_pattern`*	listează conținutul pachetelor din arhivă ce coincide cu modelul
`dpkg-reconfigure` *`package_name`*	reconfigurează pachetul specificat
`dpkg-reconfigure -plow` *`package_name`*	reconfigurează pachetul specificat cu nivelul de chestionare cel mai detaliat (ridicat)
`configure-debian`	reconfigurează pachetele selectate din meniul pe ecran complet
`dpkg --audit`	sistemul de auditare pentru pachete instalate parțial
`dpkg --configure -a`	configurează toate pachetele instalate parțial
`apt-cache policy` *`binary_package_name`*	afișează versiunea disponibilă, prioritatea și informațiile de arhivă ale unui pachet binar
`apt-cache madison` *`package_name`*	afișează versiunea disponibilă, informații privind arhiva unui pachet
`apt-cache showsrc` *`binary_package_name`*	afișează informațiile despre pachetul sursă al unui pachet binar
`apt-get build-dep` *`package_name`*	instalează pachete necesare pentru a construi pachetul
`aptitude build-dep` *`package_name`*	instalează pachete necesare pentru a construi pachetul
`apt-get source` *`package_name`*	descarcă pachetul sursă (din arhiva standard)
`dget` *`URL for dsc file`*	descarcă pachetele sursă (din altă arhivă)
`dpkg-source -x` *`package_name_version-debian.revision`*`.dsc`	construiește un arbore sursă dintr-un set de pachete sursă („`*.orig.tar.gz`” și „`*.debian.tar.gz`”/„`*.diff.gz`”)
`debuild binary`	construiește pachete dintr-un arbore sursă local
`make-kpkg kernel_image`	construiește un pachet de nucleu dintr-un arbore sursă de nucleu
`make-kpkg --initrd kernel_image`	construiește un pachet de nucleu dintr-un arbore sursă de nucleu cu initramfs activat
`dpkg -i` *`package_name_version-debian.revision_arch`*`.deb`	instalează un pachet local în sistem
`apt install /path/to/`*`package_filename`*`.deb`	instalează un pachet local în sistem, încercând în același timp să rezolve automat dependențele
`debi` *`package_name_version-debian.revision_arch`*`.dsc`	instalează pachetul (pachetele) local(e) în sistem
`dpkg --get-selections '*' >selection.txt`	salvează informații despre starea selecției pachetului la nivel `dpkg`
`dpkg --set-selections <selection.txt`	definește informațiile privind starea selecției pachetelor la nivel `dpkg`
`echo` *`package_name`* `hold | dpkg --set-selections`	definește starea de selecție a pachetului la nivel `dpkg` pentru un pachet la **hold** (echivalent cu „`aptitude hold` *`nume_pachet`*”)

Tabela 2.13: Lista operațiilor avansate de gestionare a pachetelor

2.4.2 Verificarea fișierelor pachetului instalat

Instalarea `debsums` permite verificarea fișierelor pachetelor instalate în raport cu valorile MD5sum din fișierul „`/var/lib/dp` cu debsums(1). Consultați Secțiune 10.3.5 pentru a afla cum funcționează MD5sum.

Notă
Deoarece baza de date MD5sum poate fi modificată de către intrus, debsums(1) are o utilitate limitată ca instrument de securitate. Este util doar pentru verificarea modificărilor locale efectuate de administrator sau a daunelor cauzate de erori ale suportului media.

2.4.3 Protejarea împotriva problemelor legate de pachete

Mulți utilizatori preferă să urmeze versiunile **de testare** „testing” (sau **instabile** „unstable”) ale sistemului Debian pentru noile sale caracteristici și pachete. Acest lucru face ca sistemul să fie mai predispus la erori critice ale pachetelor.

Instalarea pachetului `apt-listbugs` vă protejează sistemul împotriva erorilor critice, verificând automat Debian BTS pentru erori critice atunci când actualizați cu sistemul APT.

Instalarea pachetului `apt-listchanges` oferă știri importante din „`NEWS.Debian`” atunci când se efectuează actualizarea cu sistemul APT.

2.4.4 Căutarea în metadatele pachetului

Deși vizitarea sitului Debian https://packages.debian.org/ facilitează în prezent căutarea ușoară a metadatelor pachetelor, să analizăm câteva metode mai tradiționale.

Comenzile grep-dctrl(1), grep-status(1) și grep-available(1) pot fi utilizate pentru a căuta orice fișier care are formatul general al unui fișier de control al pachetului Debian.

„`dpkg -S` *`model_nume-fișier`*” poate fi utilizat pentru a căuta numele pachetelor care conțin fișiere cu numele corespunzător instalate de `dpkg`. Dar acest lucru ignoră fișierele create de scripturile administratorului.

Dacă trebuie să efectuați o căutare mai detaliată în metadatele dpkg, trebuie să rulați comanda «`grep -e model_expresi *`» în directorul „`/var/lib/dpkg/info/`”. Astfel, veți căuta cuvintele menționate în scripturile pachetului și în textele de interogare ale instalării.

Dacă doriți să căutați recursiv dependențele pachetelor, trebuie să utilizați apt-rdepends(8).

2.5 Detalii interne privind gestionarea pachetelor Debian

Să învățăm cum funcționează sistemul de gestionare a pachetelor Debian la nivel intern. Acest lucru ar trebui să vă ajute să creați propria soluție pentru unele probleme legate de pachete.

2.5.1 Metadate de arhivă

Fișierele cu metadate pentru fiecare distribuție sunt stocate în „`dist/`*`codename`*” pe fiecare sit oglindă Debian, de exemplu „`http://deb.debian.org/debian/`”. Structura arhivei poate fi răsfoită cu ajutorul navigatorului web. Există 6 tipuri de metadate cheie.

În arhiva recentă, aceste metadate sunt stocate sub formă de fișiere comprimate și diferențiale pentru a reduce traficul de rețea.

fișier	locație	conținut
`Release`	partea superioară a distribuției	descrierea arhivei și informații privind integritatea
`Release.gpg`	partea superioară a distribuției	fișierul de semnătură pentru fișierul „`Release`” semnat cu cheia arhivei
`Contents-`*`architecture`*	partea superioară a distribuției	lista tuturor fișierelor pentru toate pachetele din arhiva relevantă
`Release`	partea superioară a fiecărei combinații distribuție/secțiune/arhitectură	descrierea arhivei utilizată pentru regula apt_preferences(5)
`Packages`	partea superioară a fiecărei combinații distribuție/secțiune/arhitectură binară	`debian/control` concatenat pentru pachetele binare
`Sources`	partea superioară a fiecărei combinații distribuție/secțiune/sursă	`debian/control` concatenat pentru pachetele sursă

Tabela 2.14: Conținutul metadatelor arhivei Debian

2.5.2 Fișierul „Release” de nivel superior și autenticitatea

Indicație
Fișierul de nivel superior „`Release`” este utilizat pentru semnarea arhivei în cadrul sistemului **APT securizat**.

Fiecare suită din arhiva Debian are un fișier de nivel superior „`Release`”, de exemplu „`http://deb.debian.org/debian/`” după cum urmează.

```
Origin: Debian
Label: Debian
Suite: unstable
Codename: sid
Date: Sat, 14 May 2011 08:20:50 UTC
Valid-Until: Sat, 21 May 2011 08:20:50 UTC
Architectures: alpha amd64 armel hppa hurd-i386 i386 ia64 kfreebsd-amd64 kfreebsd-i386 mips ↩
    mipsel powerpc s390 sparc
Components: main contrib non-free
Description: Debian x.y Unstable - Not Released
MD5Sum:
 bdc8fa4b3f5e4a715dd0d56d176fc789 18876880 Contents-alpha.gz
 9469a03c94b85e010d116aeeab9614c0 19441880 Contents-amd64.gz
 3d68e206d7faa3aded660dc0996054fe 19203165 Contents-armel.gz
...
```

Notă
Aici puteți găsi motivele pentru care folosesc termenii „suită” și „nume în cod” în Secțiune 2.1.5. Termenul „distribuție” este folosit atunci când se face referire atât la „suită”, cât și la „numele în cod”. Toate numele „secțiunilor” oferite de arhivă sunt listate la „Componente”.

Integritatea fișierului de nivel superior „`Release`” este verificată de infrastructura criptografică numită secure apt (apt securizat), așa cum este descrisă în apt-secure(8).

- Fișierul cu semnătura criptografică „`Release.gpg`” este creat din fișierul autentic de nivel superior „`Release`” și cheia secretă a arhivei Debian.

- Cheile publice ale arhivei Debian sunt instalate local de cel mai recent pachet `debian-archive-keyring`.
- Sistemul **secure APT** verifică automat integritatea fișierului de nivel superior „`Release`” descărcat, prin criptare, cu ajutorul fișierului „`Release.gpg`” și al cheilor publice ale arhivei Debian instalate local.
- Integritatea tuturor fișierelor „`Packages`” și „`Sources`” este verificată utilizând valorile MD5sum din fișierul de nivel superior „`Release`”. Integritatea tuturor fișierelor pachetului este verificată utilizând valorile MD5sum din fișierele „`Packages`” și „`Sources`”. Consultați debsums(1) și Secțiune 2.4.2.
- Deoarece verificarea semnăturii criptografice este un proces mult mai intens pentru CPU decât calcularea valorii MD5sum, utilizarea valorii MD5sum pentru fiecare pachet în timp ce se utilizează semnătura criptografică pentru fișierul de nivel superior „`Release`” oferă o bună securitate cu performanță (a se vedea Secțiune 10.3).

Dacă intrarea **listei sursă** specifică opțiunea „`signed-by`”, integritatea fișierului „`Release`” de nivel superior descărcat este verificată folosind cheia publică specificată. Acest lucru este util atunci când **lista sursă** conține arhive non-Debian.

Indicație
Utilizarea comenzii apt-key(8) pentru gestionarea cheilor APT este depreciată.

De asemenea, puteți verifica manual integritatea fișierului „`Release`” cu fișierul „`Release.gpg`” și cheia publică a arhivei Debian publicată pe ftp-master.debian.org folosind `gpg`.

2.5.3 Fișiere „Release” la nivel de arhivă

Indicație
Fișierele „`Release`” la nivel de arhivă sunt utilizate pentru regula apt_preferences(5).

Există fișiere „`Release`” la nivel de arhivă pentru toate locațiile de arhivă specificate de **lista sursă**, cum ar fi „`http://deb.debian.org/debian/dists/unstable/main/binary-amd64/Release`” sau „`http://deb.debia`
după cum urmează.

```
Archive: unstable
Origin: Debian
Label: Debian
Component: main
Architecture: amd64
```



Atenție
Pentru strofa „`Archive:`”, numele suitei („`stable`”, „`testing`”, „`unstable`”, ...) sunt utilizate în arhiva Debian, în timp ce numele în cod („`trusty`”, „`xenial`”, „`artful`”, ...) sunt utilizate în arhiva Ubuntu.

Pentru unele arhive, cum ar fi `experimental` și `trixie -backports`, care conțin pachete care nu trebuie instalate automat, există o linie suplimentară, de exemplu, „`http://deb.debian.org/debian/dists/experimental/main/b`
după cum urmează.

```
Archive: experimental
Origin: Debian
Label: Debian
NotAutomatic: yes
Component: main
Architecture: amd64
```

Vă rugăm să rețineți că pentru arhivele normale fără „`NotAutomatic: yes`”, valoarea implicită a Pin-Priority este 500, în timp ce pentru arhivele speciale cu „`NotAutomatic: yes`”, valoarea implicită a Pin-Priority este 1 (consultați apt_preferences(5) și Secțiune 2.7.7).

2.5.4 Preluarea metadatelor pentru pachet

Când se utilizează instrumentele APT, cum ar fi `aptitude`, `apt-get`, `synaptic`, `apt-file`, `auto-apt`, ..., trebuie să actualizăm copiile locale ale metadatelor care conțin informațiile din arhiva Debian. Aceste copii locale au următoarele nume de fișiere corespunzătoare numelor specificate `distribuție`, `secțiune` și `arhitectură` din **lista surselor** (a se vedea Secțiune 2.1.5).

- „`/var/lib/apt/lists/deb.debian.org_debian_dists_`*`distribuția`*`_Release`”
- „`/var/lib/apt/lists/deb.debian.org_debian_dists_`*`distribuția`*`_Release.gpg`”
- „`/var/lib/apt/lists/deb.debian.org_debian_dists_`*`distribuția_secțiunea`*`_binary-`*`arhitectura_`*
- „`/var/lib/apt/lists/deb.debian.org_debian_dists_`*`distribuția_secțiunea`*`_source_Sources`”
- „`/var/cache/apt/apt-file/deb.debian.org_debian_dists_`*`distribuția`*`_Contents-`*`arhitectura`*`.gz`” (pentru `apt-file`)

Primele 4 tipuri de fișiere sunt comune tuturor comenzilor APT relevante și sunt actualizate din linia de comandă prin „`apt-get update`” sau „`aptitude update`”. Metadatele „`Packages`” sunt actualizate dacă „`deb`” este specificat în **lista de surse**. Metadatele „`Sources`” sunt actualizate dacă „`deb-src`” este specificat în **lista de surse**.

Metadatele „`Packages`” și „`Sources`” conțin „`Filename:`” care indică locația fișierului pachetelor binare și sursă. În prezent, aceste pachete se află în arborele de directoare „`pool/`” pentru o tranziție îmbunătățită între versiuni.

Copiile locale ale metadatelor „`Packages`” pot fi căutate interactiv cu ajutorul `aptitude`. Comanda de căutare specializată grep-dctrl(1) poate căuta copii locale ale metadatelor „`Packages`” și „`Sources`”.

Copia locală a metadatelor „`Contents-`*`architecture`*” poate fi actualizată cu „`apt-file update`”, iar locația acesteia este diferită de celelalte 4. Consultați apt-file(1). (`auto-apt` utilizează o locație diferită pentru copia locală a „`Contents-`*`architecture`*`.gz`” ca locație implicită.)

2.5.5 Starea pachetului pentru APT

În plus față de metadatele preluate de la distanță, instrumentul APT după `lenny` stochează informațiile despre starea instalării generate local în „`/var/lib/apt/extended_states`”, care este utilizat de toate instrumentele APT pentru a urmări toate pachetele instalate automat.

2.5.6 Starea pachetului pentru aptitude

În plus față de metadatele preluate de la distanță, comanda `aptitude` stochează informațiile despre starea instalării generate local în „`/var/lib/aptitude/pkgstates`”, care este utilizat numai de aceasta.

2.5.7 Copiile locale ale pachetelor descărcate

Toate pachetele descărcate de la distanță prin mecanismul APT sunt stocate în „`/var/cache/apt/archives`” până când sunt șterse.

Această politică de curățare a fișierelor cache pentru `aptitude` poate fi configurată în „`Opțiuni`” → „`Preferințe`” și poate fi forțată prin meniul „`Curăță cache-ul pachetelor`” sau „`Curăță fișierele învechite`” din „`Acțiuni`”.

tipul pachetului	structura numelui
Pachetul binar (cunoscut și sub numele de deb)	*package-name_upstream-version-debian.revision_architec*
Pachetul binar pentru debian-installer (cunoscut și sub numele de udeb)	*package-name_upstream-version-debian.revision_architec*
Pachetul sursă (sursă upstream)	*package-name_upstream-version-debian.revision*.orig.tar
Pachetul sursă 1.0 (modificări Debian)	*package-name_upstream-version-debian.revision*.diff.gz
Pachetul sursă 3.0 (quilt) (modificări Debian)	*package-name_upstream-version-debian.revision*.debian.ta
Pachetul sursă (descrierea)	*package-name_upstream-version-debian.revision*.dsc

Tabela 2.15: Structura numelor pachetelor Debian

2.5.8 Numele fișierelor pachetelor Debian

Fișierele pachetelor Debian au structuri de nume specifice.

Indicație
Aici sunt descrise doar formatele de bază ale pachetelor sursă. Pentru mai multe informații, consultați dpkg-source(1).

numele componentei	caractere utilizabile (expresie regulată ERE)	existență
package-name	[a-z0-9][-a-z0-9.+]+	cerut
epoch:	[0-9]+:	opțional
upstream-version	[-a-zA-Z0-9.+:]+	cerut
debian.revision	[a-zA-Z0-9.+~]+	opțional

Tabela 2.16: Caracterele utilizabile pentru fiecare componentă din numele pachetelor Debian

Notă
Puteți verifica ordinea versiunilor pachetelor cu ajutorul comenzii dpkg(1), de exemplu, «dpkg --compare-versions 7.0 gt 7.~pre1 ; echo $?».

Notă
Programul de instalare Debian «debian-installer» (d-i) utilizează extensia de fișier udeb pentru pachetul său binar, în loc de extensia normală deb. Un pachet udeb este un pachet deb simplificat, din care sunt eliminate câteva conținuturi neesențiale, cum ar fi documentația, pentru a economisi spațiu și a relaxa cerințele politicii privind pachetele. Atât pachetele deb, cât și cele udeb au aceeași structură. Litera „u” înseamnă micro.

2.5.9 Comanda «dpkg»

dpkg(1) este instrumentul de nivel inferior pentru gestionarea pachetelor Debian. Acesta este foarte puternic și trebuie utilizat cu precauție.

În timpul instalării pachetului numit „*nume-pachet*”, dpkg îl procesează în următoarea ordine.

1. Despachetează fișierul deb (echivalentul „ar -x”)

2. Execută „*nume-pachet*`.preinst`” folosind debconf(1)
3. Instalează conținutul pachetului în sistem (echivalentul „`tar -x`”)
4. Execută „*nume-pachet*`.postinst`” folosind debconf(1)

Sistemul `debconf` oferă interacțiune standardizată cu utilizatorul, cu suport pentru I18N și L10N (Cap. 8).

fișier	descrierea conținutului
`/var/lib/dpkg/info/`*package_name*`.conffiles`	lista fișierelor de configurare. (modificabile de utilizator)
`/var/lib/dpkg/info/`*package_name*`.list`	lista fișierelor și directoarelor instalate de pachet
`/var/lib/dpkg/info/`*package_name*`.md5sums`	lista valorilor sumelor de control MD5 pentru fișierele instalate de pachet
`/var/lib/dpkg/info/`*package_name*`.preinst`	scriptul pachetului care trebuie rulat înainte de instalarea pachetului
`/var/lib/dpkg/info/`*package_name*`.postinst`	scriptul pachetului care trebuie rulat după instalarea pachetului
`/var/lib/dpkg/info/`*package_name*`.prerm`	scriptul pachetului care trebuie rulat înainte de eliminarea pachetului
`/var/lib/dpkg/info/`*package_name*`.postrm`	scriptul pachetului care trebuie rulat după eliminarea pachetului
`/var/lib/dpkg/info/`*package_name*`.config`	scriptul pachetului pentru sistemul `debconf`
`/var/lib/dpkg/alternatives/`*package_name*	informațiile alternative utilizate de comanda `update-alternatives`
`/var/lib/dpkg/available`	informațiile privind disponibilitatea tuturor pachetelor
`/var/lib/dpkg/diversions`	informațiile de redirecționare utilizate de dpkg(1) și stabilite de dpkg-divert(8)
`/var/lib/dpkg/statoverride`	informațiile de suprascriere a stării utilizate de dpkg(1) și stabilite de dpkg-statoverride(8)
`/var/lib/dpkg/status`	informațiile privind starea tuturor pachetelor
`/var/lib/dpkg/status-old`	copia de rezervă de prima generație a fișierului „`var/lib/dpkg/status`”
`/var/backups/dpkg.status*`	copia de rezervă de a doua generație și cele mai vechi ale fișierului „`var/lib/dpkg/status`”

Tabela 2.17: Fișierele importante create de `dpkg`

Fișierul „`status`” este utilizat și de instrumente precum dpkg(1), „`dselect update`” și „`apt-get -u dselect-upgrade`”.

Comanda de căutare specializată grep-dctrl(1) poate căuta în copiile locale ale metadatelor „`status`” și „`available`”.

Indicație
În mediul debian-installer, comanda `udpkg` este utilizată pentru a deschide pachetele `udeb`. Comanda `udpkg` este o versiune simplificată a comenzii `dpkg`.

2.5.10 Comanda «update-alternatives»

Sistemul Debian dispune de un mecanism care permite instalarea fără probleme a programelor care se suprapun într-o oarecare măsură, folosind update-alternatives(1). De exemplu, puteți face ca comanda `vi` să selecteze rularea `vim` în timp ce instalați atât pachetul `vim`, cât și pachetul `nvi`.

```
$ ls -l $(type -p vi)
lrwxrwxrwx 1 root root 20 2007-03-24 19:05 /usr/bin/vi -> /etc/alternatives/vi
$ sudo update-alternatives --display vi
...
$ sudo update-alternatives --config vi
  Selection    Command
```

```
 ----------------------------------------------
      1        /usr/bin/vim
*+    2        /usr/bin/nvi

Enter to keep the default[*], or type selection number: 1
```

Sistemul Debian alternatives păstrează selecția sa sub formă de legături simbolice în „`/etc/alternatives/`”. Procesul de selecție utilizează fișierul corespunzător din „`/var/lib/dpkg/alternatives/`”.

2.5.11 Comanda «dpkg-statoverride»

Suprascrierile de stare furnizate de comanda dpkg-statoverride(8) sunt o modalitate de a indica dpkg(1) să utilizeze un proprietar sau un mod diferit pentru un **fișier** atunci când un pachet este instalat. Dacă se specifică „`--update`” și fișierul există, acesta este configurat imediat cu noul proprietar și mod.



Atenție
Modificarea directă a proprietarului sau a modului pentru un **fișier** deținut de pachet folosind comenzile `chmod` sau `chown` de către administratorul de sistem este reinițializată la următoarea actualizare a pachetului.

Notă
Folosesc aici cuvântul **fișier**, dar în realitate acesta poate fi orice obiect al sistemului de fișiere pe care `dpkg` îl gestionează, inclusiv directoare, dispozitive etc.

2.5.12 Comanda «dpkg-divert»

Redirecționările de fișiere furnizate de comanda dpkg-divert(8) sunt o modalitate de a forța dpkg(1) să nu instaleze un fișier în locația sa implicită, ci într-o locație **redirecționată**. Utilizarea `dpkg-divert` este destinată scripturilor de întreținere a pachetelor. Utilizarea sa ocazională de către administratorul de sistem este depreciată.

2.6 Recuperarea dintr-un sistem defect

Când rulează un sistem `de testare` sau `instabil`, administratorul trebuie să remedieze situația de gestionare defectuoasă a pachetelor.



Atenție
Unele metode descrise aici sunt acțiuni cu risc ridicat. Ați fost avertizați!

2.6.1 Incompatibilitate cu configurația veche a utilizatorului

Dacă un program cu interfață grafică pentru mediul grafic de birou a prezentat instabilitate după o actualizare semnificativă a versiunii din amonte, ar trebui să suspectați o interferență cu fișierele de configurare locale vechi create de acesta. Dacă este stabil sub un cont de utilizator nou creat, această ipoteză este confirmată; (aceasta este o eroare de împachetare și, de obicei, este evitată de către responsabilul Debian al pachetului).

Pentru a restabili stabilitatea, trebuie să mutați fișierele de configurare locale corespunzătoare și să reporniți programul cu interfață grafică. Este posibil să fie necesar să citiți conținutul fișierelor de configurare vechi pentru a recupera informațiile de configurare ulterior. (Nu le ștergeți prea repede.)

2.6.2 Erori de stocare în cache ale datelor pachetului

Erorile de stocare în cache a datelor pachetului provoacă erori intrigante, cum ar fi „GPG error: ... invalid: BADSIG ..” cu APT.

Trebuie să ștergeți toate datele din cache cu comanda „`sudo rm -rf /var/lib/apt/* `” și să încercați din nou. (Dacă se utilizează `apt-cacher-ng`, trebuie să executați și comanda „`sudo rm -rf /var/cache/apt-cacher-ng/* `”.)

2.6.3 Recuperarea cu comanda dpkg

Deoarece `dpkg` este un instrument de pachete de nivel foarte scăzut, acesta poate funcționa fără conexiune la rețea.

Să presupunem că pachetul `foo` era deteriorat și trebuie reparat.

Este posibil să găsiți încă copii cache ale versiunii mai vechi, fără erori, a pachetului `foo` în directorul cache al pachetului: „`/var/cache/apt/archives/`”. (Dacă nu, îl puteți descărca din arhiva https://snapshot.debian.org/ sau îl puteți copia din cache-ul pachetelor unei mașini funcționale.)

Dacă puteți porni sistemul, îl puteți instala folosind următoarea comandă:

```
# dpkg -i /path/to/foo_old_version_arch.deb
```

Dacă încercarea de a instala un pachet în acest mod eșuează din cauza unor încălcări ale dependențelor și trebuie neapărat să faceți acest lucru ca ultimă soluție, puteți ignora dependențele folosind opțiunile „`--ignore-depends`”, „`--force-depends`” și alte opțiuni ale `dpkg`. Dacă faceți acest lucru, trebuie să depuneți eforturi serioase pentru a restabili dependența corespunzătoare ulterior. Consultați dpkg(8) pentru detalii.

Notă
Dacă sistemul dvs. este grav defect, ar trebui să faceți o copie de rezervă completă a sistemului într-un loc sigur (consultați Secțiune 10.2) și să efectuați o instalare curată. Aceasta consumă mai puțin timp și produce rezultate mai bune în final.

Indicație
Dacă defecțiunea sistemului este minoră, puteți alternativ să faceți retrogradarea întregului sistem ca în Secțiune 2.7.11 folosind sistemul APT de nivel superior.

2.6.4 Instalare eșuată din cauza dependențelor lipsă

Dacă forțați instalarea unui pachet prin „`sudo dpkg -i ...`” într-un sistem în care nu sunt instalate toate pachetele dependente, instalarea pachetului va eșua, fiind instalat parțial.

Trebuie să instalați toate pachetele dependente utilizând în mod repetat «`sudo dpkg -i ...`» sau utilizând:

```
# apt --fix-broken install
```

Apoi, configurați toate pachetele instalate parțial cu următoarea comandă.

```
# dpkg --configure -a
```

2.6.5 Pachete diferite cu fișiere suprapuse

Sistemele de gestionare a pachetelor la nivel de arhivă, precum aptitude(8) sau apt-get(1), nici măcar nu încearcă să instaleze pachete cu fișiere suprapuse utilizând dependențele pachetelor (a se vedea Secțiune 2.1.7).

Erorile comise de responsabilul cu întreținerea pachetului sau implementarea unor surse de arhive mixte inconsistente (a se vedea Secțiune 2.7.6) de către administratorul de sistem pot crea o situație în care dependențele pachetului sunt definite incorect. Când instalați un pachet cu fișiere suprapuse folosind aptitude(8) sau apt-get(1) într-o astfel de situație, dpkg(1), care despachetează pachetul, se asigură că returnează o eroare programului care l-a apelat, fără a suprascrie fișierele existente.



Atenție
Utilizarea pachetelor terțe introduce riscuri semnificative pentru sistem prin intermediul scripturilor responsabilului cu întreținerea pachetului care sunt rulate cu privilegii de root și pot face orice în sistemul dvs. Comanda dpkg(1) protejează doar împotriva suprascrierii prin despachetare.

Puteți remedia o astfel de instalare defectuoasă eliminând mai întâi pachetul vechi care cauzează problema, *pachetul-vec*

```
$ sudo dpkg -P old-package
```

2.6.6 Remedierea scriptului pachetului defect

Când o comandă din scriptul pachetului returnează o eroare din anumite motive și scriptul se închide cu o eroare, sistemul de gestionare a pachetelor întrerupe acțiunea și pachetele rămân parțial instalate. Când un pachet conține erori în scripturile sale de eliminare, pachetul poate deveni imposibil de eliminat și destul de neplăcut.

Pentru problema scriptului pachetului „*nume-pachet*”, ar trebui să consultați următoarele scripturi ale pachetului.

- „`/var/lib/dpkg/info/`*nume-pachet*`.preinst`”
- „`/var/lib/dpkg/info/`*nume-pachet*`.postinst`”
- „`/var/lib/dpkg/info/`*nume-pachet*`.prerm`”
- „`/var/lib/dpkg/info/`*nume-pachet*`.postrm`”

Editați scriptul pachetului problematic din directorul rădăcină folosind următoarele tehnici:

- dezactivați linia problematică precedând-o cu „#”
- forțați returnarea succesului prin adăugarea la linia problematică a „`|| true`”

Apoi, configurați toate pachetele instalate parțial cu următoarea comandă.

```
# dpkg --configure -a
```

2.6.7 Recuperarea datelor privind selecția pachetelor

Dacă „`/var/lib/dpkg/status`” se corupe din orice motiv, sistemul Debian pierde datele de selecție a pachetelor și suferă grav. Căutați vechiul fișier „`/var/lib/dpkg/status`” în „`/var/lib/dpkg/status-old`” sau „`/var/backups/dpkg.status.*`”.

Păstrarea „`/var/backups/`” într-o partiție separată poate fi o idee bună, deoarece acest director conține multe date importante ale sistemului.

În cazul unei defecțiuni grave, recomand să reinstalați sistemul după ce ați făcut o copie de rezervă a acestuia. Chiar dacă totul din „/var/” a dispărut, puteți recupera unele informații din directoarele din „/usr/share/doc/” pentru a vă ghida în noua instalare.

Reinstalați sistemul minimal (mediul de birou).

```
# mkdir -p /path/to/old/system
```

Montați sistemul vechi la „/path/to/old/system/ - (/ruta/către/sistemul/vechi/)”.

```
# cd /path/to/old/system/usr/share/doc
# ls -1 >~/ls1.txt
# cd /usr/share/doc
# ls -1 >>~/ls1.txt
# cd
# sort ls1.txt | uniq | less
```

Apoi vi se prezintă numele pachetelor care trebuie instalate. (Pot exista și unele nume care nu sunt ale pachetelor, cum ar fi „texmf”.)

2.7 Sfaturi pentru gestionarea pachetelor

Pentru simplitate, exemplele **listei surselor** din această secțiune sunt prezentate ca „/etc/apt/sources.list” într-un stil pe o singură linie după versiunea trixie.

2.7.1 Cine a încărcat pachetul?

Deși numele responsabilului cu întreținerea pachetului listat în „/var/lib/dpkg/available” și „/usr/share/doc/pack
oferă unele informații despre „cine se află în spatele activității de împachetare”, persoana care a încărcat efectiv pachetul este oarecum necunoscută. who-uploads(1) din pachetul devscripts identifică persoana care a încărcat efectiv pachetele sursă Debian.

2.7.2 Limitarea lățimii de bandă pentru descărcare pentru APT

Dacă doriți să limitați lățimea de bandă de descărcare pentru APT la, de exemplu, 800 Kio/sec (=100 kio/sec), trebuie să configurați APT cu parametrul său de configurare după cum urmează.

```
APT::Acquire::http::Dl-Limit "800";
```

2.7.3 Descărcarea și actualizarea automată a pachetelor

Pachetul apt vine cu propriul script cron „/etc/cron.daily/apt” pentru a sprijini descărcarea automată a pachetelor. Acest script poate fi îmbunătățit pentru a efectua actualizarea automată a pachetelor prin instalarea pachetului unattended-upgrades. Acestea pot fi personalizate prin parametrii din „/etc/apt/apt.conf.d/02backup” și „/etc/apt/apt.conf.d/50unattended-upgrades”, așa cum este descris în „/usr/share/doc/unattended-upgr

Pachetul unattended-upgrades este destinat în principal actualizării de securitate pentru sistemul stabil. Dacă riscul de a afecta un sistem stabil existent prin actualizarea automată este mai mic decât riscul ca sistemul să fie afectat de un intrus care utilizează o breșă de securitate care a fost remediată prin actualizarea de securitate, ar trebui să luați în considerare utilizarea acestei actualizări automate cu parametrii de configurare următori.

```
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Download-Upgradeable-Packages "1";
APT::Periodic::Unattended-Upgrade "1";
```

Dacă utilizați un sistem `testing` sau `unstable`, nu este recomandat să utilizați actualizarea automată, deoarece aceasta va duce cu siguranță la defectarea sistemului într-o zi. Chiar și în cazul unui astfel de sistem `testing` sau `unstable`, este posibil să doriți să descărcați pachetele în avans pentru a economisi timp pentru actualizarea interactivă cu parametrii de configurare, după cum urmează.

```
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Download-Upgradeable-Packages "1";
APT::Periodic::Unattended-Upgrade "0";
```

2.7.4 Actualizări și retro-adaptări (versiuni de software migrate din ramura principală de dezvoltare și adaptate pentru a funcționa cu această versiune)

Există actualizări stabile („`trixie`-updates" în timpul ciclului de lansare `trixie` -ca-`stable`) și arhive backports.debian.org care oferă pachete de actualizare pentru `stable`.

Pentru a utiliza aceste arhive, listați toate arhivele necesare în fișierul „`/etc/apt/sources.list`" după cum urmează.

```
deb http://deb.debian.org/debian/ trixie main non-free-firmware contrib non-free
deb http://security.debian.org/debian-security trixie-security main non-free-firmware ↩
    contrib non-free
deb http://deb.debian.org/debian/ trixie-updates main non-free-firmware contrib non-free
deb http://deb.debian.org/debian/ trixie-backports main non-free-firmware contrib non-free
```

Nu este necesar să stabiliți explicit valoarea Pin-Priority în fișierul „`/etc/apt/preferences`". Când apar pachete mai noi, configurația implicită oferă cele mai rezonabile actualizări (consultați Secțiune 2.5.3).

- Toate pachetele vechi instalate sunt actualizate la versiuni mai noi din `trixie-updates`.
- Numai pachetele mai vechi instalate manual din `trixie-backports` sunt actualizate la versiuni mai noi din `trixie-backports`.

Ori de câte ori doriți să instalați manual un pachet numit „*`nume-pachet`*" cu dependența sa din arhiva `trixie-backports`, utilizați următoarea comandă în timp ce comutați versiunea țintă cu opțiunea „`-t`".

```
$ sudo apt-get install -t trixie-backports package-name
```



Avertisment
Nu instalați prea multe pachete din arhivele backports.debian.org. Acest lucru poate cauza complicații legate de dependențele pachetelor. Consultați Secțiune 2.1.11 pentru soluții alternative.

2.7.5 Arhive de pachete externe



Avertisment
Trebuie să știți că pachetul extern obține privilegii de root asupra sistemului dvs. Trebuie să utilizați numai arhiva de pachete externe de încredere. Consultați Secțiune 2.1.11 pentru soluții alternative.

Puteți utiliza APT securizat cu arhiva de pachete externe compatibilă cu Debian adăugând-o la **lista de surse** și fișierul cheie al arhivei în directorul „`/etc/apt/trusted.gpg.d/`". Consultați sources.list(5), apt-secure(8) și apt-key(8).

2.7.6 Pachete din surse mixte de arhive fără apt-pinning



Atenție
Instalarea pachetelor din surse mixte de arhive nu este acceptată de distribuția oficială Debian, cu excepția combinațiilor de arhive acceptate oficial, cum ar fi `stable` cu actualizări de securitate și stable-updates.

Iată un exemplu de operații pentru a include pachete specifice din versiuni upstream mai noi găsite în `unstable` în timp ce se urmărește `testing` pentru o singură ocazie.

1. Modificați temporar fișierul „`/etc/apt/sources.list`” pentru a include o singură intrare „`unstable`”.
2. Rulați „`aptitude update`”.
3. Rulați „`aptitude install` *`nume-pachet`*”.
4. Recuperați fișierul original „`/etc/apt/sources.list`” pentru `testing` .
5. Rulați „`aptitude update`”.

Nu creați fișierul „`/etc/apt/preferences`” și nici nu trebuie să vă faceți griji cu privire la **apt-pinning** cu această abordare manuală. Dar acest lucru este foarte greoi.



Atenție
Când utilizați surse mixte de arhive, trebuie să vă asigurați singuri de compatibilitatea pachetelor, deoarece Debian nu o garantează. Dacă există incompatibilități între pachete, puteți deteriora sistemul. Trebuie să fiți capabili să evaluați aceste cerințe tehnice. Utilizarea surselor mixte de arhive aleatorii este o operație complet opțională și nu vă încurajez să o utilizați.

Regulile generale pentru instalarea pachetelor din diferite arhive sunt următoarele.

- Pachetele non-binare („`Architecture: all`”) sunt **mai sigure** de instalat.
  - pachete de documentație: fără cerințe speciale
  - pachete de programe interpret: trebuie să fie disponibil un interpret compatibil
- Pachetele binare (care nu sunt „`Architecture: all`”) se confruntă de obicei cu multe obstacole și sunt **nesigure** de instalat.
  - compatibilitatea versiunii bibliotecii (inclusiv „`libc`”)
  - compatibilitatea versiunii programului utilitar asociat
  - compatibilitatea cu ABI a nucleului
  - compatibilitatea cu ABI C++
  - …

Notă
Pentru a face un pachet mai **sigur** de instalat, unele pachete comerciale de programe binare ne-libere pot fi furnizate cu biblioteci complet legate static. Ar trebui totuși să verificați problemele de compatibilitate ABI etc. pentru acestea.

Notă
Cu excepția cazului în care doriți să evitați deteriorarea pachetului pe termen scurt, instalarea pachetelor binare din arhive non-Debian este, în general, o idee proastă. Ar trebui să căutați toate soluțiile tehnice alternative mai sigure disponibile, care sunt compatibile cu sistemul Debian actual (consultați Secțiune 2.1.11).

2.7.7 Ajustarea versiunii candidate cu apt-pinning



Avertisment
Utilizarea tehnicii **apt-pinning** de către un utilizator începător poate cauza probleme majore. Trebuie să evitați utilizarea acestei tehnici, cu excepția cazurilor în care este absolut necesar. Consultați Secțiune 2.1.11 pentru soluții alternative.

Fără fișierul „`/etc/apt/preferences`”, sistemul APT alege cea mai recentă versiune disponibilă ca **versiune candidată** utilizând șirul de versiune. Aceasta este starea normală și utilizarea cea mai recomandată a sistemului APT. Toate combinațiile de arhive acceptate oficial nu necesită fișierul „`/etc/apt/preferences`”, deoarece unele arhive care nu ar trebui utilizate ca sursă automată de actualizări sunt marcate ca **NotAutomatic** și tratate corespunzător.

Indicație
Regula de comparare a șirurilor de versiuni poate fi verificată, de exemplu, cu „`dpkg --compare-versions ver1.1 gt ver1.1~1; echo $?`” (a se vedea dpkg(1)).

Când instalați regulat pachete din surse mixte de arhive (a se vedea Secțiune 2.7.6), puteți automatiza aceste operații complicate creând fișierul „`/etc/apt/preferences`” cu intrările corespunzătoare și modificând regula de selecție a pachetelor pentru **versiunea candidată**, așa cum este descris în apt_preferences(5). Aceasta se numește **apt-pinning**.

Când utilizați **apt-pinning**, trebuie să vă asigurați singuri de compatibilitatea pachetelor, deoarece Debian nu o garantează. **apt-pinning** este o operație complet opțională și nu vă încurajez să o utilizați.

Fișierele Release la nivel de arhivă (vedeți Secțiune 2.5.3) sunt utilizate pentru regula apt_preferences(5). Astfel, **apt-pinning** funcționează numai cu numele „suitei” pentru arhivele Debian normale și arhivele Debian de securitate. (Acest lucru este diferit de arhivele Ubuntu.) De exemplu, puteți face „`Pin: release a=unstable`”, dar nu puteți face „`Pin: release a=sid`” în fișierul „`/etc/apt/preferences`”.

Când utilizați arhive non-Debian ca parte a **apt-pinning**, trebuie să verificați pentru ce sunt destinate și să verificați credibilitatea acestora. De exemplu, Ubuntu și Debian nu sunt destinate a fi combinate.

Notă
Chiar dacă nu creați fișierul „`/etc/apt/preferences`” , puteți efectua operații de sistem destul de complexe (consultați Secțiune 2.6.3 și Secțiune 2.7.6) fără **apt-pinning**.

Iată o explicație simplificată a tehnicii **apt-pinning**.

Sistemul APT alege pachetul cu cea mai mare prioritate Pin **upgrading** din sursele de pachete disponibile definite în fișierul „`/etc/apt/sources.list`” ca pachet **candidat**. Dacă prioritatea Pin a pachetului este mai mare de 1000, această restricție de versiune pentru **actualizare** este eliminată pentru a permite retrogradarea (vedeți Secțiune 2.7.11).

Valoarea „Pin-Priority” a fiecărui pachet este definită de intrările „Pin-Priority” din fișierul „`/etc/apt/preferences`” sau se utilizează valoarea implicită a acestuia.

Arhiva **versiune țintă** poate fi definită prin opțiunea liniei de comandă, de exemplu, „`apt-get install -t testing anumit-pachet`”

Arhiva **NotAutomatic** și **ButAutomaticUpgrades** este definită de serverul arhivei care are fișierul Release al nivelului arhivei (a se vedea Secțiune 2.5.3) conținând atât „`NotAutomatic: yes`” și „`ButAutomaticUpgrades: yes`”. Arhiva **NotAutomatic** este definită de serverul de arhivă care are fișierul Release la nivel de arhivă conținând doar „`NotAutomatic: yes`”.

Situația **apt-pinning** a *pachetului* din mai multe surse de arhivă este afișată prin „`apt-cache policy` *`pachet`*”.

Pin-Priority	efectele apt-pinning asupra pachetului
1001	instalează pachetul, chiar dacă acest lucru constituie o retrogradare a pachetului
990	utilizată ca valoare implicită pentru arhiva **versiunii țintă**
500	utilizată ca valoare implicită pentru arhiva **normală**
100	utilizată ca valoare implicită pentru arhiva **NotAutomatic** și **ButAutomaticUpgrades**
100	utilizată pentru pachetul **instalat**
1	utilizată ca valoare implicită pentru arhiva **NotAutomatic**
-1	**nu instalează niciodată** pachetul, chiar dacă este recomandat

Tabela 2.18: Lista valorilor notabile (celor mai importante) ale priorității Pin pentru tehnica **apt-pinning**.

- O linie care începe cu „`Package pin:`” listează versiunea pachetului **pin** dacă este definită doar asocierea cu un *pachet*, de exemplu „`Package pin: 0.190`”.
- Nu există nicio linie cu „`Package pin:`” dacă nu este definită nicio asociere doar cu un *pachet*.
- Valoarea Pin-Priority asociată doar cu *pachetul* este listată în partea dreaptă a tuturor șirurilor de versiune, de exemplu, „`0.181 700`”.
- „`0`” este afișat în partea dreaptă a tuturor șirurilor de versiune dacă nu este definită nicio asociere doar cu *pachetul*, de exemplu, „`0.181 0`”.
- Valorile Pin-Priority ale arhivelor (definite ca „`Package: *`” în fișierul „`/etc/apt/preferences`”) sunt listate în partea stângă a tuturor rutelor de arhivă, de exemplu, „`100 http://deb.debian.org/debian/ trixie-backports Packages`”.

2.7.8 Blocarea pachetelor instalate de „Recommends”



Avertisment
Utilizarea tehnicii **apt-pinning** de către un utilizator începător poate cauza probleme majore. Trebuie să evitați utilizarea acestei tehnici, cu excepția cazurilor în care este absolut necesar. Consultați Secțiune 2.1.11 pentru soluții alternative.

Dacă nu doriți să descărcați automat anumite pachete recomandate, trebuie să creați fișierul „`/etc/apt/preferences`” și să listați în mod explicit toate aceste pachete în partea de sus a acestuia, după cum urmează.

```
Package: package-1
Pin: version *
Pin-Priority: -1

Package: package-2
Pin: version *
Pin-Priority: -1
```

2.7.9 Urmărirea suitei `testing` cu unele pachete din `unstable`



Avertisment
Utilizarea tehnicii **apt-pinning** de către un utilizator începător poate cauza probleme majore. Trebuie să evitați utilizarea acestei tehnici, cu excepția cazurilor în care este absolut necesar. Consultați Secțiune 2.1.11 pentru soluții alternative.

Iată un exemplu de tehnică **apt-pinning** pentru a include pachete specifice mai noi din versiunea upstream găsite în `unstable` actualizate regulat în timp ce se urmărește `testing`. Enumerați toate arhivele necesare în fișierul „`/etc/apt/sources.list`” după cum urmează.

```
deb http://deb.debian.org/debian/ testing main contrib non-free
deb http://deb.debian.org/debian/ unstable main contrib non-free
deb http://security.debian.org/debian-security testing-security main contrib
```

Configurați fișierul „`/etc/apt/preferences`” după cum urmează.

```
Package: *
Pin: release a=unstable
Pin-Priority: 100
```

Când doriți să instalați un pachet numit „*`nume-pachet`*” cu dependențele sale din arhiva `unstable` în această configurație, executați următoarea comandă care comută versiunea țintă cu opțiunea „`-t`” (prioritatea Pin a `unstable` devine 990).

```
$ sudo apt-get install -t unstable package-name
```

Cu această configurație, executarea obișnuită a „`apt-get upgrade`” și „`apt-get dist-upgrade`” (sau „`aptitude safe-upgrade`” și „`aptitude full-upgrade`”) actualizează pachetele care au fost instalate din arhiva `testing` utilizând arhiva `testing` curentă și pachetele care au fost instalate din arhiva `unstable` utilizând arhiva `unstable` curentă.



Atenție
Aveți grijă să nu ștergeți intrarea „`testing`” din fișierul „`/etc/apt/sources.list`”. Fără intrarea „`testing`”, sistemul APT actualizează pachetele utilizând arhiva mai nouă `unstable`.

Indicație
De obicei, editez fișierul „`/etc/apt/sources.list`” pentru a comenta intrarea din arhivă „`unstable`” imediat după operația de mai sus. Astfel se evită încetinirea procesului de actualizare din cauza numărului prea mare de intrări din fișierul „`/etc/apt/sources.list`”, deși acest lucru împiedică actualizarea pachetelor instalate din arhiva `unstable` folosind arhiva `unstable` curentă.

Indicație
Dacă se utilizează „`Pin-Priority: 1`” în loc de „`Pin-Priority: 100`” în fișierul „`/etc/apt/preferences`” , pachetele deja instalate care au valoarea Pin-Priority de 100 nu sunt actualizate de arhiva `unstable`, chiar dacă intrarea „`testing`” din fișierul „`/etc/apt/sources.list`” este eliminată.

Dacă doriți să urmăriți automat anumite pachete din `unstable` fără instalarea inițială „`-t unstable`”, trebuie să creați fișierul „`/etc/apt/preferences`” și să enumerați în mod explicit toate aceste pachete în partea de sus a acestuia, după cum urmează.

```
Package: package-1
Pin: release a=unstable
Pin-Priority: 700

Package: package-2
Pin: release a=unstable
Pin-Priority: 700
```

Acestea stabilesc valoarea Pin-Priority pentru fiecare pachet specific. De exemplu, pentru a urmări cea mai recentă versiune `unstable` a acestei „Referințe Debian” în limba engleză, ar trebui să aveți următoarele intrări în fișierul „`/etc/apt/preferences`”.

```
Package: debian-reference-en
Pin: release a=unstable
Pin-Priority: 700

Package: debian-reference-common
Pin: release a=unstable
Pin-Priority: 700
```

Indicație
Această tehnică **apt-pinning** este valabilă chiar și atunci când urmăriți arhiva `stable`. Din experiența mea, până în prezent, pachetele de documentație au fost întotdeauna sigure de instalat din arhiva `unstable`.

2.7.10 Urmărirea suitei `unstable` cu unele pachete din `experimental`



Avertisment
Utilizarea tehnicii **apt-pinning** de către un utilizator începător poate cauza probleme majore. Trebuie să evitați utilizarea acestei tehnici, cu excepția cazurilor în care este absolut necesar. Consultați Secțiune 2.1.11 pentru soluții alternative.

Iată un alt exemplu de tehnică **apt-pinning** pentru a include pachete specifice mai noi din versiunea upstream găsite în `experimental` în timp ce se urmărește `unstable`. Enumerați toate arhivele necesare în fișierul „`/etc/apt/sources.li` după cum urmează.

```
deb http://deb.debian.org/debian/ unstable main contrib non-free
deb http://deb.debian.org/debian/ experimental main contrib non-free
deb http://security.debian.org/ testing-security main contrib
```

Valoarea implicită a Pin-Priority pentru arhiva `experimental` este întotdeauna 1 (<<100), deoarece este o arhivă **NotAutomatic** (a se vedea Secțiune 2.5.3). Nu este necesar să stabiliți explicit valoarea Pin-Priority în fișierul „`/etc/apt/preferences`” doar pentru a utiliza arhiva `experimental`, cu excepția cazului în care doriți să urmăriți automat anumite pachete din aceasta pentru următoarea actualizare.

2.7.11 Retrogradarea de urgență



Avertisment
Utilizarea tehnicii **apt-pinning** de către un utilizator începător poate cauza probleme majore. Trebuie să evitați utilizarea acestei tehnici, cu excepția cazurilor în care este absolut necesar. Consultați Secțiune 2.1.11 pentru soluții alternative.



Atenție
Retrogradarea nu este acceptată oficial de Debian. Ar trebui să fie făcută doar ca parte a procesului de recuperare de urgență. În ciuda acestei situații, se știe că funcționează bine în multe cazuri. Pentru sistemele critice, ar trebui să faceți o copie de rezervă a tuturor datelor importante din sistem după operația de recuperare și să reinstalați sistemul nou de la zero.

Puteți avea norocul să faceți retrogradarea de la arhiva mai nouă la arhiva mai veche pentru a recupera din actualizarea defectuoasă a sistemului prin manipularea **versiunii candidate** (vezi Secțiune 2.7.7). Aceasta este o alternativă mai ușoară la acțiunile obositoare ale multor comenzi „`dpkg -i pachet-defect_versiune-veche.deb`” (a se vedea Secțiune 2.6.3).

Căutați în fișierul „`/etc/apt/sources.list`” liniile care conțin `unstable`, după cum urmează.

```
deb http://deb.debian.org/debian/ sid main contrib non-free
```

Înlocuiți-le cu următorul text pentru a urmări suita `testing` (de testare).

```
deb http://deb.debian.org/debian/ forky main contrib non-free
```

Configurați fișierul „`/etc/apt/preferences`” după cum urmează.

```
Package: *
Pin: release a=testing
Pin-Priority: 1010
```

Rulați „`apt-get update; apt-get dist-upgrade`” pentru a forța retrogradarea pachetelor din întregul sistem.

Eliminați acest fișier special „`/etc/apt/preferences`” după această retrogradare de urgență.

Indicație
Este o idee bună să eliminați (nu să ștergeți!) cât mai multe pachete pentru a minimiza problemele de dependență. Este posibil să fie necesar să eliminați și să instalați manual unele pachete pentru a retrograda sistemul. Nucleul Linux, încărcătorul de pornire, udev, PAM, APT și pachetele legate de rețea, precum și fișierele lor de configurare necesită o atenție specială.

2.7.12 Pachetul equivs

Dacă doriți să compilați un program din sursă pentru a înlocui pachetul Debian, cel mai bine este să îl transformați într-un pachet „debianizat” local real (`*.deb`) și să utilizați arhiva privată.

Dacă ați ales să compilați un program din sursă și să îl instalați în „`/usr/local`”, este posibil să fie necesar să utilizați `equivs` ca ultimă soluție pentru a satisface dependența pachetului lipsă.

```
Package: equivs
Priority: optional
Section: admin
Description: Circumventing Debian package dependencies
 This package provides a tool to create trivial Debian packages.
 Typically these packages contain only dependency information, but they
 can also include normal installed files like other packages do.
 .
 One use for this is to create a metapackage: a package whose sole
 purpose is to declare dependencies and conflicts on other packages so
 that these will be automatically installed, upgraded, or removed.
 .
 Another use is to circumvent dependency checking: by letting dpkg
 think a particular package name and version is installed when it
 isn't, you can work around bugs in other packages' dependencies.
 (Please do still file such bugs, though.)
```

2.7.13 Adaptarea unui pachet la sistemul stabil



Atenție
Nu există nicio garanție că procedura descrisă aici va funcționa fără eforturi manuale suplimentare din cauza diferențelor dintre sisteme.

Pentru actualizări parțiale ale sistemului `stabil`, este recomandabil să se reconstruiască un pachet în mediul său utilizând pachetul sursă. Astfel se evită actualizări masive ale pachetelor din cauza dependențelor acestora.

Adăugați următoarele intrări în „`/etc/apt/sources.list`” al unui sistem `stabil`.

```
deb-src http://deb.debian.org/debian unstable  main contrib non-free
```

Instalați pachetele necesare pentru compilare și descărcați pachetul sursă după cum urmează.

```
# apt-get update
# apt-get dist-upgrade
# apt-get install fakeroot devscripts build-essential
# apt-get build-dep foo
$ apt-get source foo
$ cd foo*
```

Actualizați unele pachete din lanțul de instrumente, cum ar fi `dpkg` și `debhelper` din pachetele retro-adaptate „backport”, dacă acestea sunt necesare pentru retro-adatare „backporting”.

Executați următoarele.

```
$ dch -i
```

Incrementați versiunea pachetului, de exemplu, adăugând „`+bp1`” în „`debian/changelog`”

Construiți pachetele și instalați-le în sistem după cum urmează.

```
$ debuild
$ cd ..
# debi foo*.changes
```

2.7.14 Server proxy pentru APT

Deoarece oglindirea întregii subsecțiuni a arhivei Debian consumă spațiu pe disc și lățime de bandă de rețea, implementarea unui server proxy local pentru APT este de dorit atunci când administrați mai multe sisteme într-o rețea locală LAN. APT poate fi configurat pentru a utiliza servere proxy web (http) generice, cum ar fi `squid` (consultați Secțiune 6.5), așa cum este descris în apt.conf(5) și în „`/usr/share/doc/apt/examples/configure-index.gz`”. Variabila de mediu „`$http_proxy`” poate fi utilizată pentru a suprascrie configurația serverului proxy din fișierul „`/etc/apt/apt.conf`”.

Există instrumente proxy special pentru arhiva Debian. Ar trebui să verificați BTS înainte de a le utiliza.

pachet	popcon(popularitate) și descriere		
`apt-cacher`	V:0.28, I:0.33	265	proxy cache pentru pachetele Debian și fișierele sursă (program Perl)
`apt-cacher-ng`	V:3.9, I:4.1	2063	proxy cache pentru distribuirea pachetelor software (program compilat în C++)

Tabela 2.19: Lista instrumentelor proxy special pentru arhiva Debian



Atenție
Când Debian își reorganizează structura arhivei, aceste instrumente proxy specializate tind să necesite rescrierea codului de către administratorul pachetului și pot să nu funcționeze pentru o perioadă. Pe de altă parte, serverele proxy web (http) generice sunt mai robuste și mai ușor de adaptat la astfel de schimbări.

2.7.15 Mai multe informații despre gestionarea pachetelor

Puteți afla mai multe despre gestionarea pachetelor din următoarele documentații.

- Documentație primară privind gestionarea pachetelor:
  - aptitude(8), dpkg(1), tasksel(8), apt(8), apt-get(8), apt-config(8), apt-secure(8), sources.list(5), apt.conf(5), și apt_preferences(5);
  - „`/usr/share/doc/apt-doc/guide.html/index.html`” și „`/usr/share/doc/apt-doc/offline.html/inde` din pachetul `apt-doc`; și
  - „`/usr/share/doc/aptitude/html/en/index.html`” din pachetul `aptitude-doc-en`.
- Documentație oficială și detaliată despre arhiva Debian:
  - Manualul de politici Debian, Capitolul 2 - Arhiva Debian,
  - „Referința dezvoltatorului Debian, Capitolul 4 - Resurse pentru dezvoltatorii Debian 4.6 Arhiva Debian”, și
  - „Întrebări frecvente despre Debian GNU/Linux, Capitolul 6 – Arhivele FTP Debian”.
- Tutorial pentru crearea unui pachet Debian pentru utilizatorii Debian:
  - „Ghid pentru administratorii Debian”.

Capitolul 3

Inițializarea sistemului

Este recomandabil ca administratorul de sistem să cunoască în linii mari modul în care sistemul Debian este pornit și configurat. Deși detaliile exacte se găsesc în fișierele sursă ale pachetelor instalate și în documentația acestora, pentru majoritatea dintre noi acestea sunt puțin copleșitoare.

Iată o prezentare generală a punctelor cheie ale inițializării sistemului Debian. Deoarece sistemul Debian este o țintă în mișcare, ar trebui să consultați cea mai recentă documentație.

- Debian Linux Kernel Handbook - (Manualul nucleului Debian Linux) este sursa principală de informații despre nucleul Debian.
- bootup(7) descrie procesul de pornire a sistemului bazat pe `systemd` . (Debian recent)
- boot(7) descrie procesul de pornire a sistemului bazat pe UNIX System V Release 4. (Debian mai vechi)

3.1 O prezentare generală a procesului de inițializare

Sistemul informatic trece prin mai multe faze ale proceselor de inițializare de la momentul pornirii până când oferă utilizatorului un sistem de operare (SO) complet funcțional.

Pentru simplitate, voi limita discuția la platforma tipică de PC cu instalarea implicită.

Procesul tipic de inițializare este ca o rachetă cu patru trepte (în cazul nostru, etape). Fiecare treaptă a rachetei predă controlul sistemului treptei următoare.

- Secțiune 3.1.1
- Secțiune 3.1.2
- Secțiune 3.1.3
- Secțiune 3.1.4

Desigur, acestea pot fi configurate diferit. De exemplu, dacă ați compilat propriul nucleu, este posibil să săriți peste pasul cu sistemul mini-Debian. Așadar, vă rugăm să nu presupuneți că acesta este cazul pentru sistemul dvs. până nu verificați personal.

3.1.1 Etapa 1: UEFI

Interfața firmware extensibilă unificată (UEFI) definește un administrator de pornire ca parte a specificației UEFI. Când un calculator este pornit, administratorul de pornire este prima etapă a procesului de pornire, care verifică configurația de pornire și, pe baza parametrilor săi, execută încărcătorul de sistem de operare specificat sau nucleul sistemului de operare (de obicei încărcătorul de pornire). Configurația de pornire este definită de variabile stocate în NVRAM, inclusiv variabile care indică rutele sistemului de fișiere către încărcătoarele de sistem de operare sau nucleele sistemului de operare.

O partiție de sistem EFI (ESP) este o partiție a dispozitivului de stocare a datelor utilizată în calculatoarele care respectă specificațiile UEFI. Accesată de firmware-ul UEFI la pornirea calculatorului, aceasta stochează aplicațiile UEFI și fișierele necesare pentru rularea acestor aplicații, inclusiv programele de încărcare a sistemului de operare. (Pe sistemele PC vechi, se poate utiliza în schimb BIOS stocat în MBR.)

3.1.2 Etapa 2: încărcătorul de pornire

Încărcătorul de pornire este a doua etapă a procesului de pornire, care este inițiat de UEFI. Acesta încarcă imaginea nucleului sistemului și imaginea initrd în memorie și le transferă controlul. Această imagine initrd este imaginea sistemului de fișiere rădăcină, iar suportul acesteia depinde de încărcătorul de pornire utilizat.

Sistemul Debian utilizează în mod normal nucleul Linux ca nucleu implicit al sistemului. Imaginea initrd pentru nucleul Linux 5.x actual este, din punct de vedere tehnic, imaginea initramfs (sistem de fișiere RAM inițial).

Există multe opțiuni disponibile pentru încărcătoare de pornire și configurare.

pachet	popcon(popularitate)	dimensiune	initrd	încărcător de pornire	descriere
grub-efi-amd64	I:466	142	Compatibil	GRUB UEFI	Este suficient de inteligent pentru a înțelege partițiile de disc și sistemele de fișiere precum vfat, ext4, …. (UEFI)
grub-pc	V:16, I:512	479	Compatibil	GRUB 2	Este suficient de inteligent pentru a înțelege partițiile de disc și sistemele de fișiere precum vfat, ext4, …. (BIOS)
grub-rescue-pc	V:0.05, I:0.58	7269	Compatibil	GRUB 2	Acestea sunt imagini de recuperare ice pot fi pornite GRUB 2 (CD și dischetă) (versiunea PC/BIOS)
grml-rescueboot	V:0.1, I:1.0	39	N/D	modulul GRUB 2	grml-rescueboot adaugă imagini ISO Linux live la meniul de pornire grub2
syslinux	V:3, I:31	325	Compatibil	Isolinux	Acesta înțelege sistemul de fișiere ISO9660. Acesta este utilizat de CD-ul de pornire.
syslinux	V:3, I:31	325	Compatibil	Syslinux	Acesta înțelege sistemul de fișiere MSDOS (FAT). Acesta este utilizat de discheta de pornire.
loadlin	V:0.03, I:0.39	87	Compatibil	Loadlin	Noul sistem este pornit din sistemul FreeDOS/MSDOS.
mbr	V:0.2, I:2.7	47	Necompatibil	MBR de Neil Turton	Acesta este un software liber care înlocuiește MSDOS MBR. Acesta recunoaște numai partițiile de disc.

Tabela 3.1: Lista încărcătorilor de pornire

Pentru sistemul UEFI, GRUB2 citește mai întâi partiția ESP și utilizează UUID-ul specificat pentru `search.fs_uuid` în „`/boot/efi/EFI/debian/grub.cfg`” pentru a determina partiția fișierului de configurare a meniului GRUB2 „`/boot/grub/grub.cfg`”.

Partea esențială a fișierului de configurare a meniului GRUB2 arată astfel:

```
menuentry 'Debian GNU/Linux' ... {
        load_video
        insmod gzio
        insmod part_gpt
        insmod ext2
        search --no-floppy --fs-uuid --set=root fe3e1db5-6454-46d6-a14c-071208ebe4b1
        echo    'Loading Linux 5.10.0-6-amd64 ...'
        linux   /boot/vmlinuz-5.10.0-6-amd64 root=UUID=fe3e1db5-6454-46d6-a14c-071208ebe4b1 ↩
            ro quiet
        echo    'Loading initial ramdisk ...'
        initrd  /boot/initrd.img-5.10.0-6-amd64
}
```

Pentru această parte din `/boot/grub/grub.cfg`, această intrare din meniu înseamnă următoarele.

parametrul	**valoare**
module GRUB2 încărcate	`gzio`, `part_gpt`, `ext2`
partiția sistemului de fișiere rădăcină utilizată	partiția identificată prin `UUID=fe3e1db5-6454-46d6-a14c-071208ebe4b1`
ruta către imaginea nucleului în sistemul de fișiere rădăcină	`/boot/vmlinuz-5.10.0-6-amd64`
parametru de pornire al nucleului utilizat	”`root=UUID=fe3e1db5-6454-46d6-a14c-071208ebe4b1 ro quiet`”
ruta către imaginea „initrd” în sistemul de fișiere rădăcină	`/boot/initrd.img-5.10.0-6-amd64`

Tabela 3.2: Semnificația intrării din meniul din partea de sus a `/boot/grub/grub.cfg`

În sistemul Debian, „`/boot/grub/grub.cfg`” este gestionat de pachetul GRUB instalat (de exemplu, `grub-efi-amd64`) și modificarea directă a acestui fișier de către utilizator este depreciată. În schimb, ar trebui să personalizați fișierele de configurare din „`/etc/grub.d/`” și „`/etc/default/grub`”. Apoi configurați fișierele de configurare GRUB și actualizați variabilele NVRAM pentru a porni automat în Debian prin:

```
# dpkg-reconfigure grub-efi-amd64
```

Indicație
Puteți afișa mesajele jurnalului de pornire al nucleului eliminând „`quiet`” din valoarea opțiunii „`GRUB_CMDLINE_LINUX_DEFAULT`” din fișierul „`/etc/default/grub`”.

Indicație
Puteți adăuga fundalul GRUB de întâmpinare plasând fișierul imagine în „`/boot/grub/`”.

A se vedea „`info grub`”, grub-install(8), grub-mkconfig(8).

3.1.3 Etapa 3: sistemul mini-Debian

Sistemul mini-Debian este a treia etapă a procesului de pornire, care este inițiată de încărcătorul de pornire. Acesta rulează nucleul sistemului cu sistemul său de fișiere rădăcină în memorie. Aceasta este o etapă pregătitoare opțională a procesului de pornire.

Notă
Termenul „sistemul mini-Debian” este inventat de autor pentru a descrie această a treia etapă a procesului de pornire în acest document. Acest sistem este denumit în mod obișnuit initrd sau sistem initramfs. Un sistem similar în memorie este utilizat de programul de instalare Debian.

Programul „`/init`” este executat ca primul program din acest sistem de fișiere rădăcină din memorie. Este un program care inițializează nucleul în spațiul utilizatorului și transferă controlul către etapa următoare. Acest mini-sistem Debian oferă flexibilitate procesului de pornire, cum ar fi adăugarea de module de nucleu înainte de procesul principal de pornire sau montarea sistemului de fișiere rădăcină ca unul criptat.

- Programul „`/init`” este un program script shell dacă initramfs a fost creat de `initramfs-tools`.
  - Puteți întrerupe această parte a procesului de pornire pentru a obține shell-ul root, furnizând „`break=init`” etc. parametrului de pornire al nucleului. Consultați scriptul „`/init`” pentru mai multe condiții de întrerupere. Acest mediu shell este suficient de sofisticat pentru a efectua o inspecție bună a hardware-ului mașinii dvs.
  - Comenzile disponibile în acest mini-sistem Debian sunt simplificate și furnizate în principal de un instrument GNU numit busybox(1).
- Programul „`/init`” este un program binar `systemd` dacă initramfs a fost creat de `dracut`.
  - Comenzile disponibile în acest mini-sistem Debian sunt reduse la mediul systemd(1).

Atenție
Trebuie să utilizați opțiunea „`-n`” pentru comanda `mount` când vă aflați în sistemul de fișiere rădăcină numai pentru citire.

3.1.4 Etapa 4: sistemul Debian normal

Sistemul Debian normal este a patra etapă a procesului de pornire, care este inițiat de sistemul mini-Debian. Nucleul sistemului pentru sistemul mini-Debian continuă să ruleze în acest mediu. Sistemul de fișiere rădăcină este comutat de la cel din memorie la cel din dispozitivul de stocare fizic.

Programul init este executat ca primul program cu PID=1 pentru a efectua procesul principal de pornire a mai multor programe. Ruta implicită a fișierului pentru programul init este „`/usr/sbin/init`”, dar poate fi modificată prin parametrul de pornire al nucleului ca „`init=/ruta/către/programul-init`” .

„`/usr/sbin/init`” este legat simbolic de „`/lib/systemd/systemd`” după Debian 8 Jessie (lansat în 2015).

Indicație
Comanda init actuală din sistemul dvs. poate fi verificată cu comanda «`ps --pid 1 -f`».

Indicație
Consultați Debian wiki: BootProcessSpeedup pentru cele mai recente sfaturi privind accelerarea procesului de pornire.

pachet	popcon(popularitate) și	dimensiune	descriere
systemd	V:920, I:980	11386	demon bazat pe evenimente init(8) pentru execuție simultană (alternativă la sysvinit)
cloud-init	V:4.1, I:6.3	3267	sistem de inițializare pentru instanțele cloud de infrastructură
systemd-sysv	V:915, I:983	76	paginile de manual și legăturile necesare pentru ca systemd să înlocuiască sysvinit
init-system-helpers	V:723, I:988	132	instrumente auxiliare pentru comutarea între sysvinit și systemd
initscripts	V:15, I:54	197	scripturi pentru inițializarea și oprirea sistemului
sysvinit-core	V:3.1, I:3.6	365	instrumente System-V precum init(8)
sysv-rc	V:29, I:57	85	mecanism de schimbare a nivelului de rulare de tip System-V
sysvinit-utils	V:725, I:1000	100	instrumente de tip System-V (startpar(8), bootlogd(8), …)
lsb-base	V:205, I:300	12	Linux Standard Base 3.2 Funcționalitatea scriptului init
insserv	V:31, I:57	132	instrument pentru organizarea secvenței de pornire utilizând dependențele scriptului LSB init.d
kexec-tools	V:1.4, I:5.3	320	instrumentul kexec pentru reporniri kexec(8) (repornire la cald)
systemd-bootchart	V:0.17, I:0.52	131	analizator de performanță al procesului de pornire
mingetty	V:0.2, I:2.3	36	numai consolă getty(8)
mgetty	V:0.13, I:0.47	318	modem inteligent getty(8) înlocuitor

Tabela 3.3: Lista instrumentelor de pornire pentru sistemul Debian

3.2 Sistemul de recuperare



Avertisment
Nu efectuați sarcini de administrare a sistemului în jurul procesului de inițializare fără a avea un sistem de recuperare.

Disponibilitatea unui sistem de recuperare ne permite să efectuăm sarcini dificile, cum ar fi:

- Pornirea unui sistem dintr-o instalare deteriorată a încărcătorului de pornire
- Repararea unei instalări deteriorate a încărcătorului de pornire
- Extragerea datelor dintr-un sistem deteriorat care nu poate fi pornit
- Editarea sistemelor de fișiere, partițiilor de disc și volumelor LVM care implică sistemul de fișiere rădăcină

De obicei, un sistem de recuperare este furnizat sub forma unui fișier imagine ISO și scris pe un suport de stocare amovibil, cum ar fi:

- Unitate flash USB pregătită ca în Secțiune 9.7.2
- CD / DVD pregătit ca în Secțiune 9.7.7

Pentru simplitate, mai jos sunt menționate ca exemple cazurile de unități flash USB, dar pot fi utilizate și CD-uri sau DVD-uri.

Indicație
Este posibil să fie necesar să modificați unele variabile UEFI NVRAM pentru a porni încărcătoare de pornire arbitrare pe suporturile de stocare amovibile.

3.2.1 Sistem de recuperare GRUB UEFI pe USB

Sistemul de recuperare GRUB UEFI cu meniu poate fi pornit prin pornirea sistemului cu „Sistemul de recuperare GRUB UEFI pe USB” introdus.

Acest „sistem de recuperare GRUB UEFI pe USB” este pregătit prin scrierea imaginii ISO Super Grub2 Disk pe o unitate flash USB în prealabil.

În cazul în care configurația încărcătorului de pornire este deteriorată de instalarea unui alt sistem de operare etc., puteți remedia această problemă astfel:

- Porniți sistemul de recuperare GRUB UEFI pentru a descoperi automat sistemele instalate care pot fi pornite.
- Porniți sistemul Debian instalat din meniul GRUB.
- În promptul shell-ului root Linux:

```
# dpkg-reconfigure grub-efi-amd64
```

Notă
Imaginea ISO de recuperare GRUB care poate fi pornită poate fi generată și urmând instrucțiunile din „`info grub-mkrescue`”. Aceasta oferă un prompt CLI GRUB shell, dar nu oferă detectarea automată a sistemelor instalate care pot fi pornite.

3.2.2 Sistem de recuperare Linux live pe USB

Sistemul de recuperare Linux live poate fi pornit prin pornirea sistemului cu „sistemul de recuperare Linux live pe USB” introdus.

Acest „sistem de recuperare live Linux pe USB” poate fi pregătit prin scrierea în prealabil a uneia dintre imaginile ISO live Linux bazate pe Debian pe o unitate flash USB. Iată câteva exemple de astfel de imagini live Linux.

- imagini Debian live
- Kali Linux Live
- Grml Live Linux

Iată câteva exemple de utilizare a acestui sistem de recuperare live Linux:

- Remediați configurația defectuoasă a încărcătorului de pornire cauzată de instalarea unui alt sistem de operare etc.:
  - Porniți sistemul de recuperare live Linux.
  - Montați partiția care conține sistemul de fișiere rădăcină al sistemului Debian instalat care nu poate fi pornit în „`/mnt`”.
  - În promptul shell-ului root Linux:

```
# chroot /mnt; dpkg-reconfigure grub-efi-amd64
```

- Remediați pachetul „`dpkg`” deteriorat:
  - Porniți sistemul de recuperare live Linux.
  - Montați partiția care conține sistemul de fișiere rădăcină al sistemului Debian instalat cu pachetul `dpkg` deteriorat în „`/mnt`”.

  - În promptul shell-ului root Linux:

```
# dpkg --root /mnt -i /mnt/var/cache/apt/archives/dpkg_old_version_amd64.deb
```

- Efectuați modificări care sunt în mod normal interzise, cum ar fi operații asupra sistemului de fișiere al sistemului instalat (consultați Secțiune 9.6).



Avertisment
Ecranul interfaței grafice a sistemului Linux live poate fi blocat după o perioadă de inactivitate.

Indicație

- Este o idee bună să stabiliți parola dvs. imediat ce porniți un sistem Linux live.
- Puteți stabili parola, de exemplu, prin «`sudo passwd utilizator`» din promptul shell al utilizatorului pe consolele virtuale Linux (consultați Secțiune 1.1.6).
- Unele sisteme Linux live pot stabili implicit „*utilizatorul*/*parola*”: „Debian Live” = „user/live”, „Kali Linux Live” = „kali/kali”

3.2.3 Sistemul de recuperare live Linux din GRUB

Sistemul de recuperare live Linux poate fi pornit din meniul GRUB. Configurația GRUB pentru acesta este pregătită după cum urmează:

- Instalați pachetul `grml-rescueboot`.
- Găsiți o imagine ISO live Linux care conține „`/boot/grub/loopback.cfg`” în imaginea sa.
  - Imaginile ISO menționate în Secțiune 3.2.2 conțin „`/boot/grub/loopback.cfg`” în imaginea lor.
- Copiați vreuna dintre aceste imagini ISO live Linux în directorul „`/boot/grml/`”.
- Actualizați meniul GRUB astfel:

```
# dpkg-reconfigure grub-efi-amd64
```

La pornirea sistemului, GRUB afișează un meniu cu opțiuni pentru sistemele de recuperare Linux live.

3.3 Systemd

3.3.1 Init systemd

Când sistemul Debian pornește, `/usr/sbin/init` legat simbolic la `/usr/lib/systemd` este pornit ca proces al sistemului init (`PID=1`) deținut de root (`UID=0`). Vedeți systemd(1).

Procesul de inițializare `systemd` generează procese în paralel pe baza fișierelor de configurare a unităților (vezi systemd.unit(5)), care sunt scrise în stil declarativ, în loc de stilul procedural de tip SysV.

Procesele generate sunt plasate în grupuri de control Linux individuale, denumite după unitatea căreia aparțin în ierarhia privată systemd (a se vedea cgroups și Secțiune 4.7.5).

Unitățile pentru modul sistem sunt încărcate din „Ruta de căutare a unităților sistemului” descrisă în systemd.unit(5). Cele principale sunt următoarele, în ordinea priorității:

- „/etc/systemd/system/*”: Unități de sistem create de administrator
- „/run/systemd/system/*”: Unități de execuție
- „/lib/systemd/system/*”: Unități de sistem instalate de gestionarul de pachete al distribuției

Interdependențele lor sunt specificate de directivele „Wants=”, „Requires=”, „Before=”, „After=”, … (a se vedea „MAPPING OF UNIT PROPERTIES TO THEIR INVERSES” în systemd.unit(5)). De asemenea, sunt definite și controalele resurselor (a se vedea systemd.resource-control(5)).

Sufixul fișierului de configurare al unității codifică tipurile acestora astfel:

- ***.service** descrie procesul controlat și supravegheat de systemd. Consultați systemd.service(5).
- ***.device** descrie dispozitivul expus în sysfs(5) ca arbore de dispozitive udev(7). Consultați systemd.device(5).
- ***.mount** descrie punctul de montare al sistemului de fișiere controlat și supravegheat de systemd. Consultați systemd.mount(5).
- ***.automount** descrie punctul de montare automată a sistemului de fișiere controlat și supravegheat de systemd. Consultați systemd.automount(5).
- ***.swap** descrie dispozitivul sau fișierul spațiului de interschimb (swap) controlat și supravegheat de systemd. Consultați systemd.swap(5).
- ***.path** descrie ruta monitorizată de systemd pentru activarea bazată pe rută. Consultați systemd.path(5).
- ***.socket** descrie soclul controlat și supravegheat de systemd pentru activarea bazată pe soclu. Consultați systemd.socket(5).
- ***.timer** descrie temporizatorul controlat și supravegheat de systemd pentru activarea bazată pe temporizator. Consultați systemd.timer(5).
- ***.slice** gestionează resursele cu ajutorul cgroups(7). Consultați systemd.slice(5).
- ***.scope** este creat programatic folosind interfețele de magistrală ale systemd pentru a gestiona un set de procese de sistem. Consultați systemd.scope(5).
- ***.target** grupează alte fișiere de configurare ale unității pentru a crea punctul de sincronizare în timpul pornirii. Consultați systemd.target(5).

La pornirea sistemului (adică, init), procesul systemd încearcă să pornească „/lib/systemd/system/default.target” (în mod normal, legat simbolic la „graphical.target”). Mai întâi, unele unități țintă speciale (vezi systemd.special(7)), cum ar fi „local-fs.target”, „swap.target” și „cryptsetup.target”, sunt introduse pentru a monta sistemele de fișiere. Apoi, alte unități țintă sunt, de asemenea, introduse de dependențele unității țintă. Pentru detalii, citiți bootup(7).

systemd oferă funcții de compatibilitate cu versiunile anterioare. Scripturile de pornire în stil SysV din „/etc/init.d/rc[0 sunt în continuare analizate, iar telinit(8) este tradus în cereri de activare a unităților systemd.



Atenție
Nivelurile de execuție emulate 2 până la 4 sunt toate legate prin legături simbolice la aceeași țintă „multi-user.target”.

3.3.2 Autentificarea cu systemd

Când un utilizator se conectează la sistemul Debian prin gdm3(8), sshd(8) etc., `/lib/systemd/system --user` este pornit ca proces de gestionare a serviciilor utilizatorului, deținut de utilizatorul corespunzător. Consultați systemd(1).

Procesul de gestionare a serviciilor utilizatorului `systemd` generează procese în paralel pe baza fișierelor de configurare declarative ale unităților (consultați systemd.unit(5) și `user@.service`(5)).

Unitățile pentru modul utilizator sunt încărcate din „Ruta de căutare a unităților utilizatorului” descrisă în systemd.unit(5). Cele principale sunt următoarele, în ordinea priorității:

- „`~/.config/systemd/user/*`”: Unități de configurare utilizator
- „`/etc/systemd/user/*`”: Unități de utilizator create de administrator
- „`/run/systemd/user/*`”: Unități de execuție
- „`/lib/systemd/user/*`”: Unități utilizator instalate de gestionarul de pachete al distribuției

Acestea sunt gestionate în același mod ca Secțiune 3.3.1.

3.4 Mesajele nucleului

Mesajul de eroare al nucleului afișat pe consolă poate fi configurat prin definirea nivelului lui de prag.

```
# dmesg -n3
```

valoarea nivelului de eroare	numele nivelului de eroare	semnificație
0	KERN_EMERG	sistemul este inutilizabil
1	KERN_ALERT	trebuie să se ia imediat măsuri
2	KERN_CRIT	condiții critice
3	KERN_ERR	condiții de eroare
4	KERN_WARNING	condiții de avertizare
5	KERN_NOTICE	condiție normală, dar semnificativă
6	KERN_INFO	informativ
7	KERN_DEBUG	mesaje de nivel de depanare

Tabela 3.4: Lista nivelurilor de eroare ale nucleului

3.5 Mesajele sistemului

În cadrul `systemd`, atât mesajele nucleului, cât și cele ale sistemului sunt înregistrate de serviciul jurnal `systemd-journald` (cunoscut și sub numele de `journald`) fie într-un fișier binar persistent sub „`/var/log/journal`”, fie într-un fișier binar volatil sub „`/run/log/journal/`”. Aceste date binare din jurnal sunt accesate prin comanda journalctl(1). De exemplu, puteți afișa jurnalul de la ultima pornire astfel:

```
$ journalctl -b
```

În cadrul `systemd`, instrumentul de înregistrare a sistemului rsyslogd(8) poate fi dezinstalat. Dacă este instalat, acesta își modifică comportamentul pentru a citi datele volatile din jurnalul binar (în loc de „`/dev/log`” implicit înainte de systemd) și pentru a crea date tradiționale permanente ASCII din jurnalul sistemului. Acest lucru poate fi personalizat prin „`/etc/default/rsyslog`” și „`/etc/rsyslog.conf`” atât pentru fișierul jurnal, cât și pentru afișarea pe ecran. Consultați rsyslogd(8) și rsyslog.conf(5). Consultați și Secțiune 9.3.2.

Operația	Fragmente de comandă
Afişarea jurnalului pentru serviciile de sistem şi nucleu de la ultima pornire	"`journalctl -b --system`"
Afişarea jurnalului pentru serviciile utilizatorului curent de la ultima pornire	"`journalctl -b --user`"
Afişarea jurnalului de activitate al „`$unit`" de la ultima pornire	"`journalctl -b -u $unit`"
Afişarea jurnalului de activităţi al „`$unit`" (în stilul „`tail -f`") de la ultima pornire	"`journalctl -b -u $unit -f`"

Tabela 3.5: Lista fragmentelor tipice de comandă `journalctl`

3.6 Gestionarea sistemului

Sistemul `systemd` oferă nu numai sistemul init, ci și operații generice de gestionare a sistemului cu comanda systemctl(1).

Aici, „`$unit`" din exemplele de mai sus poate fi un singur nume de unitate (sufixele precum `.service` și `.target` sunt opționale) sau, în multe cazuri, mai multe specificații de unități (modele globale de tip shell „`*`", „?", „`[]`" folosind fnmatch(3) care vor fi comparate cu numele primare ale tuturor unităților aflate în prezent în memorie).

Comenzile de modificare a stării sistemului din exemplele de mai sus sunt precedate de obicei de „`sudo`" pentru a obține privilegiile administrative necesare.

Rezultatul comenzii „`systemctl status $unit|$PID|$device`" utilizează culoarea punctului („●") pentru a rezuma starea unității dintr-o privire.

- Culoarea albă a „●" indică starea „inactivă" sau „dezactivată".
- Culoarea roșie a „●" indică o stare „eșuată" sau „de eroare".
- Culoarea verde a „●" indică starea „activă", „reîncărcare" sau „de activare".

3.7 Alte monitoare de sistem

Iată o listă cu alte fragmente de comenzi de monitorizare sub `systemd`. Vă rugăm să citiți paginile de manual relevante, inclusiv cgroups(7).

3.8 Configurația sistemului

3.8.1 Numele gazdei

Nucleul menține **numele de gazdă** al sistemului. Unitatea de sistem pornită de `systemd-hostnamed.service` definește numele de gazdă al sistemului la pornire ca fiind numele stocat în „`/etc/hostname`". Acest fișier trebuie să conțină **numai** numele de gazdă al sistemului, nu un nume de domeniu complet calificat.

Pentru a imprima numele actual al gazdei, executați hostname(1) fără niciun argument.

3.8.2 Sistemul de fișiere

Opțiunile de montare ale sistemelor de fișiere normale de disc și de rețea sunt definite în „`/etc/fstab`". Consultați fstab(5) și Secțiune 9.6.7.

Operația	Fragmente de comandă
Listează toate tipurile de unități disponibile	"systemctl list-units --type=help"
Listează toate unitățile țintă din memorie	"systemctl list-units --type=target"
Afișează toate unitățile de serviciu din memorie	"systemctl list-units --type=service"
Listează toate unitățile de dispozitiv din memorie	"systemctl list-units --type=device"
Listează toate unitățile de montare din memorie	"systemctl list-units --type=mount"
Afișează toate unitățile de soclu din memorie	"systemctl list-sockets"
Afișează toate unitățile de temporizare din memorie	"systemctl list-timers"
Pornește „$unit"	"systemctl start $unit"
Oprește „$unit"	"systemctl stop $unit"
Reîncarcă configurația specifică serviciului	"systemctl reload $unit"
Oprește și pornește toate „$unit"	"systemctl restart $unit"
Pornește „$unit" și oprește toate celelalte unități	"systemctl isolate $unit"
Comută la „graphical" (sistem GUI, cu interfață grafică)	"systemctl isolate graphical"
Comută la „multi-user" (sistem CLI, cu interfață de linie de comandă)	"systemctl isolate multi-user"
Comută la „rescue" (sistem CLI, cu interfață de linie de comandă, mono-utilizator)	"systemctl isolate rescue"
Trimite semnalul de omorâre către „$unit"	"systemctl kill $unit"
Verifică dacă serviciul „$unit" este activ	"systemctl is-active $unit"
Verifică dacă serviciul „$unit" a eșuat	"systemctl is-failed $unit"
Verificăi starea „$unit\|$PID|dispozitiv"	"systemctl status $unit\|$PID|$device"
Afișează proprietățile „$unit\|$job"	"systemctl show $unit\|$job"
Reinițializează „$unit" eșuată	"systemctl reset-failed $unit"
Listează dependențele tuturor serviciilor de unitate	"systemctl list-dependencies --all"
Listează fișierele de unitate instalate în sistem	«systemctl list-unit-files»
Activează „$unit" (adaugă legătură simbolică)	"systemctl enable $unit"
Dezactivează „$unit" (elimină legătura simbolică)	"systemctl disable $unit"
Demască „$unit" (elimină legătura simbolică către „/dev/null")	"systemctl unmask $unit"
Maschează „$unit" (adaugă legătură simbolică la „/dev/null")	"systemctl mask $unit"
Obține configurația țintei implicite	"systemctl get-default"
Stabilește ținta implicită la „graphical" (sistem GUI, cu interfață grafică)	"systemctl set-default graphical"
Stabilește ținta implicită la „multi-user" (sistem CLI, cu interfață de linie de comandă)	"systemctl set-default multi-user"
Afișează mediul de lucru	"systemctl show-environment"
Definește „variabila" din mediul de lucru la „valoare"	"systemctl set-environment variable=value"
Elimină definiția (valoarea) „variabilei" din mediul de lucru	"systemctl unset-environment variable"
Reîncarcă toate fișierele de unitate și demonii	"systemctl daemon-reload"
Închide(oprește) sistemul	"systemctl poweroff"
Oprește și repornește sistemul	"systemctl reboot"
Suspendă sistemul	"systemctl suspend"
Hibernează sistemul	"systemctl hibernate"

Operația	Fragmente de comandă
Afișează timpul petrecut pentru fiecare etapă de inițializare	"`systemd-analyze time`"
Listează toate unitățile în funcție de timpul necesar pentru inițializare	"`systemd-analyze blame`"
Încarcă și detectează erorile din fișierul „`$unit`"	"`systemd-analyze verify $unit`"
Afișează informații succinte despre starea de rulare a utilizatorului sesiunii apelantului	"`loginctl user-status`"
Afișează informații succinte despre starea de rulare a sesiunii apelantului	"`loginctl session-status`"
Urmărește procesul de pornire prin cgroups	"`systemd-cgls`"
Urmărește procesul de pornire prin cgroups	"`ps xawf -eo pid,user,cgroup,args`"
Urmărește procesul de pornire prin cgroups	Citiți sysfs sub „`/sys/fs/cgroup/`"

Tabela 3.7: Lista altor fragmente de comenzi de monitorizare sub `systemd`

Configurația sistemului de fișiere criptat este definită în „`/etc/crypttab`". Consultați crypttab(5)

Configurația RAID software cu mdadm(8) este definită în „`/etc/mdadm/mdadm.conf`". Consultați mdadm.conf(5).



Avertisment
După montarea tuturor sistemelor de fișiere, fișierele temporare din „`/tmp`", „`/var/lock`" și „`/var/run`" sunt șterse la fiecare pornire.

3.8.3 Inițializarea interfeței de rețea

Interfețele de rețea sunt inițializate de obicei din „`networking.service`" pentru interfața `lo` și din „`NetworkManager.ser` pentru alte interfețe în sistemele Debian moderne sub `systemd`.

Consultați Cap. 5 pentru a afla cum să le configurați.

3.8.4 Inițializarea sistemului cloud

Instanța sistemului cloud poate fi lansată ca o clonă a „Imaginilor cloud oficiale Debian" sau a unor imagini similare. Pentru o astfel de instanță de sistem, caracteristici precum numele gazdei, sistemul de fișiere, rețeaua, configurația regională, cheile SSH, utilizatorii și grupurile pot fi configurate folosind funcționalitățile oferite de pachetele `cloud-init` și `netplan.io` cu mai multe surse de date, cum ar fi fișierele plasate în imaginea originală a sistemului și datele externe furnizate în timpul lansării sale. Aceste pachete permit configurarea declarativă a sistemului utilizând date YAML.

Pentru mai multe informații, consultați „Cloud Computing cu Debian și descendenții săi", „Documentația Cloud-init" și Secțiune 5.4.

3.8.5 Exemplu de personalizare pentru ajustarea serviciului sshd

Cu instalarea implicită, multe servicii de rețea (a se vedea Cap. 6) sunt pornite ca procese demon după `network.target` la pornire de către `systemd`. „`sshd`" nu face excepție. Să schimbăm acest lucru în pornirea la cerere a „`sshd`" ca exemplu de personalizare.

Mai întâi, dezactivați unitatea de serviciu instalată în sistem.

```
$ sudo systemctl stop sshd.service
$ sudo systemctl mask sshd.service
```

Sistemul de activare la cerere a soclurilor serviciilor Unix clasice se realiza prin intermediul superserverului `inetd` (sau `xinetd`). În cadrul `systemd`, echivalentul poate fi activat prin adăugarea fișierelor de configurare a unităților ***.socket** și ***.service**.

`sshd.socket` pentru specificarea unui soclu pe care să se asculte

```
[Unit]
Description=SSH Socket for Per-Connection Servers

[Socket]
ListenStream=22
Accept=yes

[Install]
WantedBy=sockets.target
```

`sshd@.service` ca fișier de serviciu corespunzător pentru `sshd.socket`

```
[Unit]
Description=SSH Per-Connection Server

[Service]
ExecStart=-/usr/sbin/sshd -i
StandardInput=socket
```

Apoi reîncărcați.

```
$ sudo systemctl daemon-reload
```

3.9 Sistemul udev

Sistemul udev oferă un mecanism pentru detectarea și inițializarea automată a hardware-ului (a se vedea udev(7)) începând cu nucleul Linux 2.6. La detectarea fiecărui dispozitiv de către nucleu, sistemul udev pornește un proces de utilizator care folosește informații din sistemul de fișiere sysfs (a se vedea Secțiune 1.2.12), încarcă modulele de nucleu necesare utilizând programul modprobe(8) (a se vedea Secțiune 3.10) și creează nodurile de dispozitiv corespunzătoare.

Indicație
Dacă „`/lib/modules/`*`versiunea-nucleului`*`/modules.dep`” nu a fost generat corect de depmod(8) din anumite motive, este posibil ca modulele să nu fie încărcate așa cum se așteaptă sistemul udev. Executați „`depmod -a`” pentru a remedia problema.
Pentru regulile de montare din „`/etc/fstab`”, nodurile dispozitivelor nu trebuie să fie statice. Puteți utiliza UUID pentru a monta dispozitive în locul numelor de dispozitive, cum ar fi „`/dev/sda`”. Consultați Secțiune 9.6.3.

Deoarece sistemul udev este o țintă în continuă schimbare, las detaliile pentru alte documentații și descriu aici doar informațiile minimale.



Avertisment
Nu încercați să rulați programe de lungă durată, cum ar fi scriptul de copie de rezervă, cu `RUN` în regulile udev, așa cum se menționează în udev(7). Creați în schimb un fișier systemd.service(5) adecvat și activați-l. Consultați Secțiune 10.2.3.2.

3.10 Inițializarea modulelor de nucleu

Programul modprobe(8) ne permite să configurăm nucleul Linux în execuție din procesul utilizatorului prin adăugarea și eliminarea modulelor nucleului. Sistemul udev (a se vedea Secțiune 3.9) automatizează invocarea acestuia pentru a ajuta la inițializarea modulului nucleului.

Există module non-hardware și module speciale de controlor hardware, precum cele enumerate mai jos, care trebuie preîncărcate prin listarea lor în fișierul „`/etc/modules`” (consultați modules(5)).

- Modulele TUN/TAP furnizează dispozitive de rețea virtuale punct-la-punct (TUN) și dispozitive de rețea Ethernet virtuale (TAP),
- modulele netfilter care oferă funcționalități de paravan de protecție netfilter (iptables(8), Secțiune 5.7), și
- și modulul controlorului watchdog timer.

Fișierele de configurare pentru programul modprobe(8) se află în directorul „`/etc/modprobes.d/`”, așa cum se explică în modprobe.conf(5). (Dacă doriți să evitați încărcarea automată a anumitor module ale nucleului, luați în considerare adăugarea acestora pe lista neagră din fișierul „`/etc/modprobes.d/blacklist`”.)

Fișierul „`/lib/modules/`*`versiune`*`/modules.dep`” generat de programul depmod(8) descrie dependențele modulelor utilizate de programul modprobe(8).

Notă
Dacă întâmpinați probleme la încărcarea modulelor la pornirea sistemului sau cu modprobe(8), comanda „`depmod -a`” poate rezolva aceste probleme prin reconstruirea fișierului „`modules.dep`”.

Programul modinfo(8) afișează informații despre un modul al nucleului Linux.

Programul lsmod(8) formatează conținutul „`/proc/modules`”, arătând ce module ale nucleului sunt încărcate în prezent.

Indicație
Puteți identifica hardware-ul exact din sistemul dvs. Consultați Secțiune 9.5.3.
Puteți configura hardware-ul la pornire pentru a activa funcțiile hardware dorite. Consultați Secțiune 9.5.4.
Probabil puteți adăuga suport pentru dispozitivul dvs. special prin recompilarea nucleului. Consultați Secțiune 9.10.

Capitolul 4

Autentificare și controale de acces

Când o persoană (sau un program) solicită acces la sistem, autentificarea confirmă identitatea ca fiind una de încredere.



Avertisment
Erorile de configurare ale PAM vă pot bloca accesul la propriul sistem. Trebuie să aveți la îndemână un CD de recuperare sau să configurați o partiție de pornire alternativă. Pentru a recupera sistemul, porniți-l cu ajutorul acestora și remediați problemele de acolo.

4.1 Autentificare normală Unix

Autentificarea normală Unix este asigurată de modulul pam_unix(8) din cadrul PAM (Pluggable Authentication Modules). Cele trei fișiere de configurare importante, cu intrări separate prin „`:`”, sunt următoarele.

fișier	permisiuni	utilizator	grupul	descriere
`/etc/passwd`	`-rw-r--r--`	`root`	`root`	informații despre contul utilizatorului (curățate)
`/etc/shadow`	`-rw-r-----`	`root`	`shadow`	informații securizate despre contul utilizatorului
`/etc/group`	`-rw-r--r--`	`root`	`root`	informații despre grup

Tabela 4.1: 3 fișiere de configurare importante pentru pam_unix(8)

„`/etc/passwd`” conține următoarele.

```
 ...
user1:x:1000:1000:User1 Name,,,:/home/user1:/bin/bash
user2:x:1001:1001:User2 Name,,,:/home/user2:/bin/bash
 ...
```

Așa cum se explică în passwd(5), fiecare intrare separată prin „`:`” din acest fișier înseamnă următoarele.

- Nume de autentificare
- Intrarea specificării parolei
- ID-ul numeric al utilizatorului

- ID-ul numeric al grupului
- Nume utilizator sau câmp comentariu
- Directorul personal al utilizatorului
- Interpret opțional de comenzi pentru utilizator

A doua intrare din „`/etc/passwd`” a fost utilizată pentru introducerea parolei criptate. După introducerea „`/etc/shadow`”, această intrare este utilizată pentru introducerea specificațiilor parolei.

conținut	**semnificație**
(gol)	cont fără parolă
x	parola criptată se află în „`/etc/shadow`”

Tabela 4.2: Al doilea conținut al intrării „`/etc/passwd`”

Fișierul „`/etc/shadow`” conține următoarele câmpuri.

```
 ...
user1:$1$Xop0FYH9$IfxyQwBe9b8tiyIkt2P4F/:13262:0:99999:7:::
user2:$1$vXGZLVbS$ElyErNf/agUDsm1DehJMS/:13261:0:99999:7:::
 ...
```

După cum se explică în shadow(5), fiecare intrare separată prin „`:`” din acest fișier înseamnă următoarele.

- Nume de autentificare
- Parolă criptată (inițialele „1” indică utilizarea criptării MD5. Simbolul „*” indică faptul că nu există date de conectare.)
- Data ultimei modificări a parolei, exprimată ca număr de zile de la 1 ianuarie 1970
- Numărul de zile pe care utilizatorul va trebui să le aștepte înainte de a i se permite să își schimbe din nou parola
- Numărul de zile după care utilizatorul va trebui să își schimbe parola
- Numărul de zile înainte de expirarea parolei în care utilizatorul trebuie avertizat
- Numărul de zile după expirarea parolei în care parola ar trebui să fie încă acceptată
- Data expirării contului, exprimată ca număr de zile de la 1 ianuarie 1970
- …

Fișierul „`/etc/group`” conține următoarele câmpuri.

```
group1:x:20:user1,user2
```

După cum se explică în group(5), fiecare intrare separată prin „`:`” din acest fișier are următoarea semnificație.

- Numele grupului
- Parolă criptată (nu este utilizată efectiv)
- ID-ul numeric al grupului
- Listă separată prin virgule a numelor de utilizatori

Notă
„`/etc/gshadow`” oferă o funcție similară cu „`/etc/shadow`” pentru „`/etc/group`”, dar nu este utilizat în mod real.

Notă
Apartenența efectivă a unui utilizator la un grup poate fi adăugată dinamic dacă se adaugă linia „`auth optional pam_group.so`” în „`/etc/pam.d/common-auth`” și se definește în „`/etc/security/group.conf`”. Consultați pam_group(8).

Notă
Pachetul `base-passwd` conține o listă oficială a utilizatorilor și grupurilor: „`/usr/share/doc/base-passwd/users-and-groups.html`”.

4.2 Gestionarea informațiilor privind contul și parola

Iată câteva comenzi importante pentru gestionarea informațiilor contului.

comandă	funcție
`getent passwd` *`user_name`*	răsfoiește informațiile contului „*`nume-utilizator`*”
`getent shadow` *`user_name`*	răsfoiește informațiile contului ascuns al „*`nume-utilizator`*”
`getent group` *`group_name`*	răsfoiește informațiile despre grupul „*`nume-grup`*”
`passwd`	gestionează parola pentru cont
`passwd -e`	definește o parolă unică pentru activarea contului
`chage`	gestionează informațiile privind expirarea parolelor

Tabela 4.3: Lista comenzilor pentru gestionarea informațiilor contului

Este posibil să aveți nevoie de privilegii de root pentru ca unele funcții să funcționeze. Consultați crypt(3) pentru criptarea parolelor și a datelor.

Notă
În sistemul configurat cu PAM și NSS ca mașină Debian salsa, conținutul local „`/etc/passwd`”, „`/etc/group`” și „`/etc/shadow`” poate să nu fie utilizat în mod activ de sistem. Comenzile de mai sus sunt valabile chiar și în astfel de medii.

4.3 Parolă bună

Când creați un cont în timpul instalării sistemului sau cu comanda passwd(1), ar trebui să alegeți o parolă bună care să conțină cel puțin 6-8 caractere, inclusiv unul sau mai multe caractere din fiecare dintre următoarele seturi, conform passwd(1).

- literele alfabetului în minusculă
- cifrele de la 0 la 9
- Semne de punctuație



Avertisment
Nu alegeți cuvinte ușor de ghicit pentru parolă. Numele contului, numărul de asigurare socială, numărul de telefon, adresa, data nașterii, numele membrilor familiei sau al animalelor de companie, cuvinte din dicționar, secvențe simple de caractere precum „12345” sau „qwerty” etc. sunt toate alegeri nepotrivite pentru parolă.

4.4 Crearea unei parole criptate

Există instrumente independente pentru a genera parole criptate cu „salt” (sare).

pachet	popcon(popularitate)	dimensiune	comandă	funcție
whois	V:23, I:203	384	mkpasswd	interfață cu funcții excesive pentru biblioteca crypt(3)
openssl	V:846, I:996	2564	openssl passwd	calculează rezumatele criptografice ale parolelor (OpenSSL). passwd(1ssl)

Tabela 4.4: Lista instrumentelor pentru generarea parolei

4.5 PAM și NSS

Sistemele moderne similare cu Unix precum sistemul Debian oferă PAM (Pluggable Authentication Modules -- Module de autentificare conectabile) și NSS (Name Service Switch -- Comutare servicii de nume) administratorului de sistem local pentru configurarea sistemului său. Rolul acestora poate fi rezumat după cum urmează.

- PAM oferă un mecanism flexibil de autentificare utilizat de software-ul aplicației, implicând astfel schimbul de date privind parolele.
- NSS oferă un mecanism flexibil de servicii de nume, care este frecvent utilizat de biblioteca standard C pentru a obține numele utilizatorului și al grupului pentru programe precum ls(1) și id(1).

Aceste sisteme PAM și NSS trebuie configurate în mod coerent.

Pachetele principale ale sistemelor PAM și NSS sunt următoarele.

- „The Linux-PAM System Administrators’ Guide -- Ghidul administratorilor de sistem Linux-PAM” din `libpam-doc` este esențial pentru învățarea configurării PAM.
- Secțiunea „System Databases and Name Service Switch -- Baze de date de sistem și comutator de servicii de nume” din `glibc-doc-reference` este esențială pentru învățarea configurației NSS.

Notă
Puteți vedea o listă mai extinsă și actualizată folosind comanda „`aptitude search «libpam-|libnss-»`”. Acronimul NSS poate însemna și „Network Security Service” (Serviciu de securitate rețea), care este diferit de „Name Service Switch” (Comutator serviciu de nume).

Notă
PAM este cea mai simplă metodă de inițializare a variabilelor de mediu pentru fiecare program cu valoarea implicită la nivel de sistem.

pachet	popcon(popularitate) și dimensiune	dimensiune	descriere
libpam-modules	V:942, I:1000	892	Module de autentificare conectabile (serviciul de bază)
libpam-ldapd	V:7, I:13	79	Modul de autentificare conectabil care permite interfețe LDAP
libpam-systemd	V:764, I:969	848	Modul de autentificare conectabil pentru înregistrarea sesiunilor utilizatorilor pentru `logind`
libpam-doc	I:7.5	1492	Module de autentificare conectabile (documentație în format html și text)
libc6	V:936, I:1000	5483	Biblioteca GNU C: Biblioteci partajate care oferă de asemenea serviciul „Name Service Switch”
glibc-doc	I:5.1	3927	Biblioteca GNU C: Pagini de manual
glibc-doc-reference	I:3.1	14764	Biblioteca GNU C: Manual de referință în format info, pdf și html (non-free)
libnss-mdns	V:299, I:507	142	Modul NSS pentru rezolvarea numelor DNS multicast
libnss-ldapd	V:8, I:16	131	Modul NSS pentru utilizarea LDAP ca serviciu de denumire

Tabela 4.5: Lista pachetelor principale ale sistemelor PAM ți NSS

În cadrul systemd, pachetul `libpam-systemd` este instalat pentru a gestiona autentificările utilizatorilor prin înregistrarea sesiunilor utilizatorilor în ierarhia grupului de control `systemd` pentru logind. Consultați systemd-logind(8), logind.conf(5) și pam_systemd(8).

4.5.1 Fișierele de configurare accesate de PAM și NSS

Iată câteva fișiere de configurare importante accesate de PAM și NSS.

fișier de configurare	funcție
`/etc/pam.d/`*`program_name`*	definește configurația PAM pentru programul „*`nume-program`*”; vedeți pam(7) și pam.d(5)
`/etc/nsswitch.conf`	definește configurația NSS cu intrarea pentru fiecare serviciu.; a se vedea nsswitch.conf(5)
`/etc/nologin`	limitează autentificarea utilizatorului prin modulul pam_nologin(8)
`/etc/securetty`	limitează tty pentru accesul root prin modulul pam_securetty(8)
`/etc/security/access.conf`	stabilește limita de acces prin modulul pam_access(8)
`/etc/security/group.conf`	definește restricția bazată pe grup prin modulul pam_group(8)
`/etc/security/pam_env.conf`	definește variabilele de mediu prin modulul pam_env(8)
`/etc/environment`	definește variabile de mediu suplimentare prin modulul pam_env(8) cu argumentul „`readenv=1`”
`/etc/default/locale`	definește configurația de limbă prin modulul pam_env(8) cu argumentul „`readenv=1 envfile=/etc/default/locale`” (Debian)
`/etc/security/limits.conf`	stabilește restricții de resurse (ulimit, core, …) prin modulul pam_linits(8)
`/etc/security/time.conf`	stabilește restricția de timp prin modulul pam_time(8)
`/etc/systemd/logind.conf`	definește configurația gestionarului de autentificare `systemd` (a se vedea logind.conf(5) și systemd-logind.service(8))

Tabela 4.6: Lista fișierelor de configurare accesate de PAM și NSS

Limitarea selecției parolei este implementată de modulele PAM, pam_unix(8) și pam_cracklib(8). Acestea pot fi configurate prin argumentele lor.

Indicație
Modulele PAM utilizează sufixul „`.so`” pentru numele fișierelor lor.

4.5.2 Sistemul modern de gestionare centralizată

Sistemul modern de gestionare centralizată poate fi implementat utilizând serverul centralizat Lightweight Directory Access Protocol (LDAP) pentru a administra numeroase sisteme de tip Unix și non-Unix din rețea. Implementarea de cod-sursă deschis a protocolului Lightweight Directory Access Protocol este OpenLDAP Software.

Serverul LDAP furnizează informațiile despre cont prin utilizarea PAM și NSS cu pachetele `libpam-ldapd` și `libnss-ldapd` pentru sistemul Debian. Pentru a activa această funcție sunt necesare mai multe acțiuni (nu am utilizat această configurare, iar informațiile următoare sunt pur secundare. Vă rugăm să citiți acest text în acest context.).

- Configurați un server LDAP centralizat rulând un program precum demonul LDAP autonom, slapd(8).
- Modificați fișierele de configurare PAM din directorul „`/etc/pam.d/`” pentru a utiliza „`pam_ldap.so`” în locul fișierului implicit „`pam_unix.so`”.
- Modificați configurația NSS din fișierul „`/etc/nsswitch.conf`” pentru a utiliza „`ldap`” în locul valorii implicite („`compat`” sau „`file`”).
- Trebuie să configurați `libpam-ldapd` pentru a utiliza conexiunea SSL (sau TLS) pentru securitatea parolei.
- Puteți configura `libnss-ldapd` să utilizeze conexiunea SSL (sau TLS) pentru a asigura integritatea datelor, cu prețul unei suprasarcini a rețelei LDAP.
- Ar trebui să rulați nscd(8) local pentru a stoca în cache orice rezultate ale căutării LDAP, în scopul de a reduce traficul de rețea LDAP.

Consultați documentația din nsswitch.conf(5), pam.conf(5), ldap.conf(5) și „`/usr/share/doc/libpam-doc/html/`” oferită de pachetul `libpam-doc` și „`info libc «Name Service Switch»`” oferită de pachetul `glibc-doc`.

În mod similar, puteți configura sisteme centralizate alternative cu alte metode.

- Integrarea utilizatorilor și grupurilor cu sistemul Windows.
  - Accesați serviciile domeniului Windows prin pachetele `winbind` și `libpam_winbind`.
  - Consultați winbindd(8) și Integrating MS Windows Networks with Samba -- Integrarea rețelelor MS Windows cu Samba.
- Integrarea utilizatorilor și grupurilor cu sistemul vechi de tip Unix.
  - Accesați NIS (denumit inițial YP) sau NIS+ prin pachetul `nis`.
  - Consultați The Linux NIS(YP)/NYS/NIS+ HOWTO.

4.5.3 „De ce comanda «su» a GNU nu acceptă grupul wheel”

Aceasta este celebra frază care apare în partea de jos a vechii pagini „`info su`” a lui Richard M. Stallman. Nu vă faceți griji: comanda actuală `su` din Debian utilizează PAM, astfel încât se poate restricționa utilizarea `su` la grupul `root` prin activarea liniei cu „`pam_wheel.so`” în „`/etc/pam.d/su`”.

4.5.4 Reguli mai stricte privind parolele

Instalarea pachetului `libpam-cracklib` vă permite să impuneți reguli mai stricte pentru parole.

Într-un sistem GNOME tipic, care instalează automat `libpam-gnome-keyring`, „`/etc/pam.d/common-password`” arată astfel:

```
# here are the per-package modules (the "Primary" block)
password  requisite     pam_cracklib.so retry=3 minlen=8 difok=3
password  [success=1 default=ignore]  pam_unix.so obscure use_authtok try_first_pass  ↩
    yescrypt
# here's the fallback if no module succeeds
password  requisite     pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
password  required      pam_permit.so
# and here are more per-package modules (the "Additional" block)
password  optional  pam_gnome_keyring.so
# end of pam-auth-update config
```

4.6 Securitatea autentificării

Notă
Informațiile prezentate aici **pot fi insuficiente** pentru nevoile dvs. de securitate, dar ar trebui să constituie un **bun început**.

4.6.1 Parolă sigură în Internet

Multe servicii ale stratului de transport populare comunică mesaje, inclusiv autentificarea parolei, în text simplu. Este o idee foarte proastă să transmiteți parola în text simplu pe Internet, unde poate fi interceptată. Puteți rula aceste servicii prin „Transport Layer Security” (TLS) sau predecesorul său, „Secure Sockets Layer” (SSL), pentru a securiza întreaga comunicare, inclusiv parola, prin criptare.

nume de serviciu nesigur	port	nume de serviciu sigur	port
`www (http)`	80	`https`	443
`smtp (mail)`	25	`ssmtp (smtps)`	465
`ftp-data`	20	`ftps-data`	989
`ftp`	21	`ftps`	990
`telnet`	23	`telnets`	992
`imap2`	143	`imaps`	993
`pop3`	110	`pop3s`	995
`ldap`	389	`ldaps`	636

Tabela 4.7: Lista serviciilor și porturilor nesigure și sigure

Criptarea consumă timp de procesare. Ca alternativă care nu solicită procesorul, puteți păstra comunicarea în text simplu, securizând doar parola cu un protocol de autentificare securizat, cum ar fi „Authenticated Post Office Protocol” (APOP) pentru POP și „Challenge-Response Authentication Mechanism MD5” (CRAM-MD5) pentru SMTP și IMAP. (Pentru trimiterea mesajelor de poștă electronică prin Internet către serverul dvs. de poștă electronică de la clientul dvs. de poștă electronică, este popular în ultima vreme să se utilizeze noul port de trimitere a mesajelor 587 în locul

portului SMTP tradițional 25, pentru a evita blocarea portului 25 de către furnizorul de rețea, în timp ce vă autentificați cu CRAM-MD5.)

Autentificarea tradițională bazată pe parolă prezintă în continuare probleme inerente de securitate, cum ar fi reutilizarea parolelor pe mai multe situri web. Consultați alte metode de autentificare pentru o soluție sigură: Parola unică bazată pe timp (TOTP), Al doilea factor universal (U2F), Autentificarea web (WebAuthn, denumită în mod obișnuit „passkey”)

4.6.2 Shell securizat

Programul Secure Shell (SSH) asigură comunicații criptate sigure între două gazde ce nu sunt de „încredere” într-o rețea nesigură, cu autentificare securizată. Acesta constă din clientul OpenSSH, ssh(1) și demonul OpenSSH, sshd(8). Acest SSH poate fi utilizat pentru a tunela o comunicare prin protocol nesigur, cum ar fi POP și X, în mod securizat pe Internet, cu ajutorul funcției de redirecționare a porturilor.

Clientul încearcă să se autentifice utilizând autentificarea bazată pe gazdă, autentificarea cu cheie publică, autentificarea cu provocare-răspuns sau autentificarea cu parolă. Utilizarea autentificării cu cheie publică permite conectarea la distanță fără parolă. Consultați Secțiune 6.3.

4.6.3 Măsuri suplimentare de securitate pentru Internet

Chiar și atunci când utilizați servicii securizate, cum ar fi Secure Shell (SSH) și Point-to-point tunneling protocol (PPTP), există în continuare posibilitatea unor intruziuni prin atacuri de tip „forță brută” (brute force) pentru ghicirea parolelor etc. din Internet. Utilizarea politicii paravanului de protecție (a se vedea Secțiune 5.7) împreună cu următoarele instrumente de securitate poate îmbunătăți situația de securitate.

pachet	popcon(popularitate)	dimensiune	descriere
knockd	V:0.7, I:1.6	110	mic demon port-knock knockd(1) și client knock(1)
fail2ban	V:101, I:111	2275	interzice adresele IP care provoacă erori multiple de autentificare
libpam-shield	V:0.04, I:0.05	115	blochează atacatorii de la distanță care încearcă să ghicească parola

Tabela 4.8: Lista instrumentelor care oferă măsuri suplimentare de securitate

4.6.4 Securizarea parolei root

Pentru a împiedica accesul persoanelor la mașina dvs. cu privilegii de root, trebuie să efectuați următoarele acțiuni.

- Împiedicați accesul fizic la dispozitivul de stocare al sistemului (HDD / SSD / ...)
- Blocați UEFI/BIOS și împiedicați pornirea de pe suportul amovibil
- Definiți o parolă pentru sesiunea interactivă GRUB
- Blocați meniul GRUB împotriva editării

Cu acces fizic la dispozitivul de stocare al sistemului, redefinirea parolei este relativ ușoară, urmând pașii de mai jos.

1. Mutați dispozitivul de stocare al sistemului pe un PC cu USB ce poate fi pornit UEFI/BIOS.
2. Porniți sistemul cu un suport de recuperare (consultați Secțiune 3.2.2).
3. Montați partiția rădăcină cu acces de citire/scriere.

4. Editați fișierul „/etc/passwd” din partiția rădăcină și lăsați necompletată a doua intrare pentru contul root.

Dacă aveți acces de editare la intrarea din meniul GRUB (consultați Secțiune 3.1.2) pentru grub-rescue-pc la pornire, este și mai ușor să urmați pașii de mai jos.

1. Porniți sistemul cu parametrul de nucleu modificat în ceva precum „root=/dev/sda6 rw init=/bin/sh”.
2. Editați fișierul „/etc/passwd” și lăsați goală a doua intrare pentru contul root.
3. Reporniți sistemul.

Shell-ul root al sistemului este acum accesibil fără parolă.

Notă
Odată ce cineva are acces la shell-ul root, poate accesa totul din sistem și poate reinițializa oricare dintre parolele din sistem. Mai mult, poate compromite parola pentru toate conturile de utilizator folosind instrumente de spargere a parolelor prin forță brută, cum ar fi pachetele john și crack (a se vedea Secțiune 9.5.10). Această parolă spartă poate duce la compromiterea altor sisteme.

Singura soluție software rezonabilă pentru a evita toate aceste probleme este utilizarea unei partiții rădăcină criptate software (sau partiția „/etc”) folosind dm-crypt și initramfs (vezi Secțiune 9.9). Totuși, aveți întotdeauna nevoie de parolă pentru a porni sistemul.

4.7 Alte controale de acces

Există controale de acces la sistem, altele decât autentificarea bazată pe parolă și permisiunile de fișiere.

Notă
Consultați Secțiune 9.4.16 pentru restricționarea caracteristicii nucleului tastă de atenție securizată „secure attention key”: (SAK).

4.7.1 Liste de control al accesului (ACL)

ACL-urile sunt un superset al permisiunilor obișnuite, așa cum se explică în Secțiune 1.2.3.

Întâlniți ACL-uri în acțiune în mediul grafic de birou modern. Când un dispozitiv de stocare USB formatat este montat automat ca, de exemplu, „/media/penguin/USBSTICK”, un utilizator normal penguin poate executa:

```
 $ cd /media/penguin
 $ ls -la
total 16
drwxr-x---+ 1 root    root    16 Jan 17 22:55 .
drwxr-xr-x  1 root    root    28 Sep 17 19:03 ..
drwxr-xr-x  1 penguin penguin 18 Jan  6 07:05 USBSTICK
```

„+” în coloana a 11-a indică faptul că ACL-urile sunt active. Fără ACL-uri, un utilizator normal penguin nu ar trebui să poată afișa această listă, deoarece penguin nu face parte din grupul root. Puteți vedea ACL-urile astfel:

```
 $ getfacl .
# file: .
# owner: root
# group: root
user::rwx
```

```
user:penguin:r-x
group::---
mask::r-x
other::---
```

Aici:

- „`user::rwx`”, „`group::---`” și „`other::---`” corespund permisiunilor obișnuite ale proprietarului, grupului și altor utilizatori.
- ACL „`user:penguin:r-x`” permite unui utilizator normal `penguin` să aibă permisiuni „`r-x`”. Acest lucru a permis „`ls -la`” să listeze conținutul directorului.
- ACL „`mask::r-x`” stabilește limita superioară a permisiunilor.

Pentru mai multe informații, consultați „POSIX Access Control Lists on Linux -- Liste de control al accesului POSIX în Linux”, acl(5), getfacl(1) și setfacl(1).

4.7.2 sudo

sudo(8) este un program conceput pentru a permite unui administrator de sistem să acorde privilegii root limitate utilizatorilor și să înregistreze activitatea root. `sudo` necesită doar parola unui utilizator obișnuit. Instalați pachetul `sudo` și activați-l configurând opțiunile din „`/etc/sudoers`”. Consultați exemplul de configurare din „`/usr/share/doc/sudo/exa` și Secțiune 1.1.12.

Utilizarea mea a `sudo` pentru sistemul cu un singur utilizator (a se vedea Secțiune 1.1.12) are scopul de a mă proteja de propria mea prostie. Personal, consider că utilizarea `sudo` este o alternativă mai bună decât utilizarea sistemului din contul root tot timpul. De exemplu, următoarea comandă schimbă proprietarul „*`some_file`*” în „*`my_name`*”.

```
$ sudo chown my_name some_file
```

Desigur, dacă cunoașteți parola root (așa cum o cunosc utilizatorii Debian care au instalat singuri sistemul), orice comandă poate fi executată sub root din contul oricărui utilizator folosind „`su -c`”.

4.7.3 PolicyKit

PolicyKit este o componentă a sistemului de operare pentru controlul privilegiilor la nivel de sistem în sistemele de operare de tip Unix.

Aplicațiile GUI mai noi nu sunt concepute pentru a rula ca procese privilegiate. Acestea comunică cu procesele privilegiate prin intermediul PolicyKit pentru a efectua operații administrative.

PolicyKit limitează astfel de operetii la conturile de utilizator care aparțin grupului `sudo` din sistemul Debian.

A se vedea polkit(8).

4.7.4 Restricționarea accesului la anumite servicii ale serverului

Pentru securitatea sistemului, este recomandat să dezactivați cât mai multe programe de server posibil. Acest lucru devine esențial pentru serverele de rețea. Serverele neutilizate, activate fie direct ca daemon, fie prin programul super-server, sunt considerate riscuri de securitate.

Multe programe, cum ar fi sshd(8), utilizează controlul accesului bazat pe PAM. Există multe modalități de a restricționa accesul la anumite servicii ale serverului.

- fișiere de configurare: „`/etc/default/`*`program_name`*”

- configurația unității de serviciu systemd pentru daemon
- modulele de autentificare conectabile PAM (Pluggable Authentication Modules)
- „`/etc/inetd.conf`” pentru super-server
- „`/etc/hosts.deny`” și „`/etc/hosts.allow`” pentru TCP wrapper, tcpd(8)
- „`/etc/rpc.conf`” pentru Sun RPC
- „`/etc/at.allow`” și „`/etc/at.deny`” pentru atd(8)
- „`/etc/cron.allow`” și „`/etc/cron.deny`” pentru crontab(1)
- paravanul de protecție al rețelei din infrastructura netfilter

Consultați Secțiune 3.6, Secțiune 4.5.1 și Secțiune 5.7.

Indicație
Serviciile Sun RPC trebuie să fie active pentru NFS și alte programe bazate pe RPC.

Indicație
Dacă aveți probleme cu accesul la distanță într-un sistem Debian recent, comentați configurația problematică, cum ar fi „`ALL: PARANOID`” din „`/etc/hosts.deny`”, dacă există; (dar trebuie să fiți atenți la riscurile de securitate implicate de acest tip de acțiune).

4.7.5 Caracteristici de securitate Linux

Nucleul Linux a evoluat și oferă caracteristici de securitate care nu se regăsesc în implementările tradiționale UNIX.

Linux acceptă atribute extinse care extind atributele UNIX tradiționale (a se vedea xattr(7)).

Linux împarte privilegiile asociate în mod tradițional cu superutilizatorul în unități distincte, cunoscute sub numele de `capacități`(7), care pot fi activate și dezactivate independent. Capacitățile sunt un atribut per fir de execuție începând cu versiunea 2.2 a nucleului.

Infrastructura modulelor de securitate Linux („Linux Security Module”: LSM) oferă un mecanism pentru diverse verificări de securitate care pot fi conectate la noile extensii ale nucleului. De exemplu:

- AppArmor
- Security-Enhanced Linux (SELinux)
- Smack (Simplified Mandatory Access Control Kernel)
- Tomoyo Linux

Deoarece aceste extensii pot restricționa modelul de privilegii mai strict decât politicile obișnuite ale modelului de securitate de tip Unix, chiar și privilegiile root pot fi restricționate. Vă recomandăm să citiți documentul infrastructurii „Linux Security Module (LSM)” la kernel.org.

Spatiile de nume Linux învăluiesc o resursă globală a sistemului într-o abstractizare care face ca procesele din cadrul spatiului de nume să pară că au propria instanță izolată a resursei globale. Modificările aduse resursei globale sunt vizibile pentru alte procese care sunt membre ale spațiului de nume, dar sunt invizibile pentru alte procese. Începând cu versiunea 5.6 a nucleului, există 8 tipuri de spații de nume (a se vedea namespaces(7), unshare(1), nsenter(1)).

Începând cu Debian 11 Bullseye (2021), Debian utilizează ierarhia cgroup unificată (cunoscută și sub numele de cgroups-v2).

Exemple de utilizare a spațiilor de nume cu cgroups pentru a izola procesele acestora și a permite controlul resurselor sunt:

- Systemd. A se vedea Secțiune 3.3.1.
- Mediul sandbox. A se vedea Secțiune 7.7.
- Containerele Linux precum Docker, LXC. A se vedea Secțiune 9.11.

Aceste funcționalități nu pot fi realizate prin Secțiune 4.1. Aceste subiecte avansate sunt în mare parte în afara domeniului de aplicare al acestui document introductiv.

Capitolul 5

Configurarea rețelei

Indicație
Pentru un ghid modern specific Debian privind rețelele, citiți Manualul administratorului Debian — Configurarea rețelei.

Indicație
În cadrul systemd, networkd poate fi utilizat pentru gestionarea rețelelor. A se vedea systemd-networkd(8).

5.1 Infrastructura de bază a rețelei

Să trecem în revistă infrastructura de rețea de bază a sistemului Debian modern.

5.1.1 Rezoluția numelui de gazdă

Rezolvarea numelui de gazdă este acceptată în prezent și de mecanismul NSS (Name Service Switch). Fluxul acestei rezolvări este următorul.

1. Fișierul „`/etc/nsswitch.conf`” cu secțiunea „`hosts: files dns`” dictează ordinea de rezolvare a numelor de gazdă; (aceasta înlocuiește vechea funcționalitate a secțiunii „`order`” din „`/etc/host.conf`”).
2. Metoda `files` este invocată prima. Dacă numele gazdei este găsit în fișierul „`/etc/hosts`”, aceasta returnează toate adresele valide pentru acesta și se închide; (fișierul „`/etc/host.conf`” conține „`multi on`”).
3. Se invocă metoda `dns`. Dacă numele gazdei este găsit prin interogarea către Sistemul de nume de domeniu Internet („Internet Domain Name System”: DNS) identificat de fișierul „`/etc/resolv.conf`”, acesta returnează toate adresele valide pentru acesta și iese.

O stație de lucru tipică poate fi instalată cu numele gazdei definit, de exemplu, la „*host_name*” și numele de domeniu opțional definit la un șir gol. Apoi, „`/etc/hosts`” arată astfel.

```
127.0.0.1 localhost
127.0.1.1 host_name

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

pachete	popcon(popularitate)	dimensiune	tipul	descriere
network-manager	V:427, I:474	7365	config::NM	NetworkManager (daemon): gestionează rețeaua în mod automat
network-manager-gnome	V:36, I:147	19	config::NM	NetworkManager (interfață GNOME)
netplan.io	V:2.1, I:8.4	340	config::NM+networkd	Netplan (generator): interfață unificată și declarativă pentru NetworkManager și modulele systemd-networkd
ifupdown	V:649, I:973	204	config::ifupdown	instrument standardizat pentru activarea și dezactivarea rețelei (specific Debian)
pppoeconf	V:0.2, I:3.9	176	config::helper	asistent de configurare pentru conexiunea PPPoE
wpasupplicant	V:421, I:521	3896	config::helper	client ce oferă suport pentru WPA și WPA2 (IEEE 802.11i)
wpagui	V:0.2, I:1.2	779	config::helper	client cu interfață grafică Qt pentru wpa_supplicant
wireless-tools	V:193, I:262	232	config::helper	instrumente pentru manipularea extensiilor „fără fir” Linux (Linux Wireless Extensions)
iw	V:40, I:478	332	config::helper	instrument pentru configurarea dispozitivelor wireless Linux
iproute2	V:781, I:987	4268	config::iproute2	iproute2, IPv6 și alte configurații avansate de rețea: ip(8), tc(8) etc.
iptables	V:381, I:635	2408	config::Netfilter	instrumente de administrare pentru filtrarea pachetelor și NAT (Netfilter)
nftables	V:266, I:880	189	config::Netfilter	instrumente de administrare pentru filtrarea pachetelor și NAT (Netfilter) (succesor al {ip,ip6,arp,eb}tables)
iputils-ping	V:206, I:998	188	test	testează accesibilitatea rețelei unei gazde aflată la distanță prin nume gazdă sau adresă IP (iproute2)
iputils-arping	V:2, I:17	53	test	testează accesibilitatea rețelei unei gazde de la distanță specificată de adresa ARP
iputils-tracepath	V:2, I:19	50	test	trasează ruta de rețea către o gazdă la distanță
ethtool	V:98, I:251	1093	test	afișează sau modifică configurarea dispozitivului Ethernet
mtr-tiny	V:4, I:38	182	test::low-level	urmărește ruta de rețea către o gazdă la distanță (curses)
mtr	V:4, I:40	231	test::low-level	urmărește ruta de rețea către o gazdă la distanță (curses și GTK)
gnome-nettool	V:0.5, I:8.4	2480	test::low-level	instrumente pentru operații comune de informații de rețea (GNOME)
nmap	V:25, I:182	4934	test::low-level	cartograf de rețea / scaner de porturi (Nmap, consolă)
tcpdump	V:20, I:177	1346	test::low-level	analizator de trafic de rețea (Tcpdump, consolă)
wireshark	V:4, I:39	17068	test::low-level	analizator de trafic de rețea (Wireshark, GTK)
tshark	V:2, I:23	433	test::low-level	analizator de trafic de rețea (consolă)
tcptrace	V:0.2, I:1.5	407	test::low-level	produce o sinteză a conexiunilor din ieșirea `tcpdump`
dnsutils	V:5, I:146	23	test::low-level	clienți de rețea furnizați cu BIND: nslookup(8), nsupdate(8), dig(8)
dlint	V:0.1, I:2.0	52	test::low-level	verifică informațiile zonei DNS utilizând căutări în serverul de nume
dnstracer	V:0.1, I:1.1	63	test::low-level	urmărește un lanț de servere DNS până la sursă

Tabela 5.1: Lista instrumentelor de configurare a rețelei

Fiecare linie începe cu adresa IP și este urmată de numele gazdei asociat.

Adresa IP `127.0.1.1` din a doua linie a acestui exemplu poate să nu fie găsită pe unele sisteme de tip Unix. Programul de instalare Debian creează această intrare pentru un sistem fără adresă IP permanentă, ca soluție pentru unele programe (de exemplu, GNOME), așa cum este documentat în eroarea #719621.

host_name corespunde numelui de gazdă definit în „`/etc/hostname`” (a se vedea Secțiune 3.8.1).

Pentru un sistem cu o adresă IP permanentă, această adresă IP permanentă trebuie utilizată aici în locul `127.0.1.1`.

Pentru un sistem cu o adresă IP permanentă și un nume de domeniu complet calificat (FQDN) furnizat de Sistemul de nume de domeniu (DNS), acel *host_name* canonic. *domain_name* ar trebui utilizat în locul *host_name*.

„`/etc/resolv.conf`” este un fișier static dacă pachetul `resolvconf` nu este instalat. Dacă este instalat, este o legătură simbolică. În ambele cazuri, acesta conține informații care inițializează rutinele de rezolvare. Dacă DNS-ul este găsit la IP=”`192.168.11.1`”, acesta conține următoarele.

```
nameserver 192.168.11.1
```

Pachetul `resolvconf` transformă acest „`/etc/resolv.conf`” într-o legătură simbolică și gestionează automat conținutul său prin scripturi-cârlig.

Pentru stația de lucru PC într-un mediu LAN ad-hoc tipic, numele gazdei poate fi rezolvat prin Multicast DNS (mDNS), pe lângă metodele de bază `files` și `dns`.

- Avahi oferă un cadru pentru descoperirea serviciilor DNS multicast în Debian.
- Acesta este echivalentul lui Apple Bonjour / Apple Rendezvous.
- Pachetul de module de extensie `libnss-mdns` asigură rezolvarea numelor de gazdă prin mDNS pentru funcționalitatea GNU Name Service Switch (NSS) a bibliotecii GNU C (glibc).
- Fișierul „`/etc/nsswitch.conf`” ar trebui să conțină o secțiune de tipul „`hosts: files mdns4_minimal [NOTFOUND dns`” (consultați `/usr/share/doc/libnss-mdns/README.Debian` pentru alte configurații).
- Un nume de gazdă cu sufixul ”. local„ domeniu pseudo-de nivel superior este rezolvat prin trimiterea unui mesaj de interogare mDNS într-un pachet UDP multicast utilizând adresa IPv4 ”`224.0.0.251`„ sau adresa IPv6 ”`FF02::FB`”.

Notă
Extinderea domeniilor generice de nivel superior (gTLD) în Sistemul de nume de domeniu este în curs de desfășurare. Aveți grijă la coliziunea numelor atunci când alegeți un nume de domeniu utilizat numai în cadrul LAN.

Notă
Utilizarea pachetelor precum `libnss-resolve` împreună cu `systemd-resolved`, sau `libnss-myhostname`, sau `libnss-mymachine`, cu listări corespunzătoare în linia „`hosts`” din fișierul „`/etc/nsswitch.conf`” poate suprascrie configurația tradițională de rețea discutată mai sus. Consultați nss-resolve(8), systemd-resolved(8), nss-myhostname(8) și nss-mymachines(8) pentru mai multe informații.

5.1.2 Numele interfeței de rețea

systemd utilizează „Nume predictibile ale interfețelor de rețea”, cum ar fi „`enp0s25`”.

Clasa	adrese de rețea	masca de rețea	masca de rețea /biți	numărul de su-brețele
A	10.x.x.x	255.0.0.0	/8	1
B	172.16.x.x — 172.31.x.x	255.255.0.0	/16	16
C	192.168.0.x — 192.168.255.x	255.255.255.0	/24	256

Tabela 5.2: Lista intervalelor de adrese de rețea

5.1.3 Intervalul de adrese pentru rețeaua locală (LAN)

Să ne reamintim intervalele de adrese IPv4 pe 32 de biți din fiecare clasă rezervate pentru utilizare în rețelele locale (LAN) de către rfc1918. Aceste adrese sunt garantate să nu intre în conflict cu niciuna dintre adresele din Internet.

Notă
Adresele IP scrise cu două puncte sunt adresa IPv6, de exemplu, „`::1`” pentru `localhost`.

Notă
Dacă una dintre aceste adrese este atribuită unei gazde, atunci gazda respectivă nu trebuie să acceseze Internetul direct, ci trebuie să îl acceseze printr-o poartă de acces care acționează ca un proxy pentru servicii individuale sau care efectuează Network Address Translation (NAT). Routerul de bandă largă efectuează de obicei NAT pentru mediul LAN al consumatorului.

5.1.4 Suportul pentru dispozitivele de rețea

Deși majoritatea dispozitivelor hardware sunt acceptate de sistemul Debian, există unele dispozitive de rețea care necesită firmware DFSG non-free pentru a le accepta. Vă rugăm să consultați Secțiune 9.10.5.

5.2 Configurația modernă a rețelei pentru mediul de birou

Interfețele de rețea sunt inițializate de obicei din „`networking.service`” pentru interfața `lo` și din „`NetworkManager.ser` pentru alte interfețe în sistemele Debian moderne sub `systemd`.

Debian poate gestiona conexiunea la rețea prin intermediul software-ului de gestionare daemon, cum ar fi Network-Manager (NM) (network-manager și pachetele asociate).

- Acestea vin cu propriile programe cu interfață grafică (GUI) și de linie de comandă ca interfețe de utilizator.
- Acestea vin cu propriul daemon ca motor al sistemului.
- Acestea permit conectarea ușoară a sistemului dvs. la Internet.
- Acestea permit gestionarea ușoară a configurației rețelelor cu fir și fără fir.
- Acestea ne permit să configurăm rețeaua independent de pachetul vechi `ifupdown`.

Notă
Nu utilizați aceste instrumente de configurare automată a rețelei pentru servere. Acestea sunt destinate în principal utilizatorilor de stații de lucru mobile pe laptopuri.

Aceste instrumente moderne de configurare a rețelei trebuie configurate corespunzător pentru a evita conflictele cu pachetul vechi `ifupdown` și fișierul său de configurare „`/etc/network/interfaces`”.

5.2.1 Instrumente cu interfață grafică pentru configurarea rețelei

Documentația oficială pentru NM în Debian este disponibilă în fișierul „`/usr/share/doc/network-manager/README.De`

În esență, configurarea rețelei pentru mediul grafic debirou se face după cum urmează.

1. Faceți ca utilizatorul mediului grafic de birou, de exemplu `foo`, să aparțină grupului „`netdev`” prin următoarea comandă (alternativ, faceți acest lucru automat prin intermediul D-bus în medii grafice moderne de birou, precum GNOME și KDE).

```
$ sudo usermod -a -G netdev foo
```

2. Păstrați configurația din fișierul „`/etc/network/interfaces`” cât mai simplă, după cum se arată mai jos.

```
auto lo
iface lo inet loopback
```

3. Reporniți NM în felul următor.

```
$ sudo systemctl restart NetworkManager
```

4. Configurați rețeaua prin intermediul interfeței grafice.

Notă
Numai interfețele care **nu** sunt listate în „`/etc/network/interfaces`” sunt gestionate de NM pentru a evita conflictul cu `ifupdown`.

Indicație
Dacă doriți să extindeți capacitățile de configurare a rețelei NM, vă rugăm să căutați module de extensie adecvate și pachete suplimentare, cum ar fi `network-manager-openconnect`, `network-manager-openvpn-gnome`, `network-manager-pptp-gnome`, `mobile-broadband-provider-info`, `gnome-bluetooth` etc.

5.3 Configurația modernă a rețelei fără interfața grafică

În systemd, rețeaua poate fi configurată în `/etc/systemd/network/`. Consultați systemd-resolved(8), resolved.conf(5) și systemd-networkd(8).

Acest lucru permite configurarea modernă a rețelei fără interfață grafică.

O configurație pentru clientul DHCP poate fi efectuată prin crearea fișierului „`/etc/systemd/network/dhcp.network`”. De exemplu:

```
[Match]
Name=en*

[Network]
DHCP=yes
```

O configurație de rețea statică poate fi efectuată prin crearea fișierului „`/etc/systemd/network/static.network`”. De exemplu:

```
[Match]
Name=en*

[Network]
Address=192.168.0.15/24
Gateway=192.168.0.1
```

5.4 Configurația modernă a rețelei pentru cloud

Configurația modernă de rețea pentru cloud poate utiliza pachetele `cloud-init` și `netplan.io` (consultați Secțiune 3.8.4).

Pachetul `netplan.io` acceptă `systemd-networkd` și `NetworkManager` ca sisteme de configurare a rețelei și permite configurarea declarativă a rețelei utilizând date YAML. Când modificați YAML:

- Rulați comanda „`netplan generate`” pentru a genera toate configurațiile necesare acestor sisteme din YAML.
- Rulați comanda „`netplan apply`” pentru a aplica configurația generată la sistemele de configurare a rețelei.

Consultați „Documentația Netplan”, netplan(5), netplan-generate(8) și netplan-apply(8).

A se vedea și „Documentația Cloud-init” (în special secțiunile „Surse de configurare” și „Netplan Passthrough”) pentru a afla cum `cloud-init` poate integra configurația `netplan.io` cu surse de date alternative.

5.4.1 Configurația modernă a rețelei pentru cloud cu DHCP

O configurație pentru clientul DHCP poate fi efectuată prin crearea unui fișier sursă de date „`/etc/netplan/50-dhcp.yaml`”:

```
network:
  version: 2
  ethernets:
    all-en:
      match:
        name: "en*"
      dhcp4: true
      dhcp6: true
```

5.4.2 Configurația modernă a rețelei pentru cloud cu adresă IP statică

O configurație de rețea statică poate fi efectuată prin crearea unui fișier sursă de date „`/etc/netplan/50-static.yaml`”:

```
network:
  version: 2
  ethernets:
    eth0:
      addresses:
        - 192.168.0.15/24
      routes:
        - to: default
          via: 192.168.0.1
```

5.4.3 Configurația modernă a rețelei pentru cloud cu Network Manager

Configurația clientului de rețea utilizând infrastructura Network Manager poate fi efectuată prin crearea unui fișier sursă de date „`/etc/netplan/00-network-manager.yaml`”:

```
network:
  version: 2
  renderer: NetworkManager
```

5.5 Configurația rețelei de nivel inferior

Pentru configurarea rețelei de nivel inferior în Linux, utilizați programele iproute2 (ip(8), …) .

5.5.1 Comenzi «iproute2»

Comenzile Iproute2 oferă funcționalități complete de configurare a rețelei la nivel inferior. Iată un tabel de conversie din comenzile învechite net-tools în noile comenzi iproute2 etc.

comenzile net-tools învechite	noile comenzi iproute2, etc.	utilitate
ifconfig(8)	`ip addr`	adresa de protocol (IP sau IPv6) pe un dispozitiv
route(8)	`ip route`	intrare în tabelul de direcționare
arp(8)	`ip neigh`	intrare în cache-ul ARP sau NDISC
`ipmaddr`	`ip maddr`	adresă multicast
`iptunnel`	`ip tunnel`	tunel peste IP
nameif(8)	ifrename(8)	numește interfețele de rețea pe baza adreselor MAC
mii-tool(8)	ethtool(8)	configurează dispozitivul Ethernet

Tabela 5.3: Tabel de corespondență între comenzile învechite `net-tools` și noile comenzi `iproute2`

A se vedea ip(8)și Linux Advanced Routing & Traffic Control.

5.5.2 Operații de rețea sigure la nivel scăzut

Puteți utiliza în siguranță comenzile de rețea de nivel inferior, deoarece acestea nu modifică configurația rețelei.

comandă	descriere
`ip addr show`	afișează starea legăturii și a adresei interfețelor active
`route -n`	afișează toate tabelele de direcționare în adrese numerice
`ip route show`	afișează toate tabelele de direcționare în adrese numerice
`arp`	afișează conținutul curent al tabelelor cache ARP
`ip neigh`	afișează conținutul curent al tabelelor cache ARP
`plog`	afișează jurnalul demonului ppp
`ping yahoo.com`	verifică conexiunea la Internet la „`yahoo.com`”
`whois yahoo.com`	verifică cine a înregistrat „`yahoo.com`” în baza de date a domeniilor
`traceroute yahoo.com`	urmărește conexiunea la Internet la „`yahoo.com`”
`tracepath yahoo.com`	urmărește conexiunea la Internet la „`yahoo.com`”
`mtr yahoo.com`	urmărește conexiunea la Internet la „`yahoo.com`” (în mod repetat)
`dig [@dns-server.com] example.com [{a|mx|any}]`	verifică înregistrările DNS ale „`example.com`” de către „`dns-server.com`” pentru o înregistrare „a”, „mx” sau „any”
`iptables -L -n`	verifică filtrul de pachete
`ss -a`	găsește toate porturile deschise
`ss -l`	găsește porturile de ascultare
`ss -l -t -n`	găsește porturile TCP de ascultare (numerice)
`dlint example.com`	verifică informațiile zonei DNS pentru „`example.com`”

Tabela 5.4: Lista comenzilor de rețea de nivel inferior

Indicație
Unele dintre aceste instrumente de configurare a rețelei de nivel inferior se află în „`/usr/sbin/`”. Este posibil să fie necesar să introduceți ruta completă a comenzii, cum ar fi „`/usr/sbin/ifconfig`”, sau să adăugați „`/usr/sbin`” la lista „$PATH” din „`~/.bashrc`”.

5.6 Optimizarea rețelei

Optimizarea generală a rețelei depășește sfera acestei documentații. Abordez doar subiecte relevante pentru conexiunile casnice.

pachete	popcon(popularitate) și dimensiune		descriere
`iftop`	V:6, I:85	93	afișează informații despre utilizarea lățimii de bandă pe o interfață de rețea
`iperf`	V:2, I:32	431	instrument de măsurare a lățimii de bandă a protocolului Internet
`ifstat`	V:0.5, I:5.3	52	monitorizarea statisticilor interfeței
`bmon`	V:2, I:19	154	monitor portabil de lățime de bandă și estimator de viteză
`ethstatus`	V:0.2, I:2.3	41	script care măsoară rapid rata de transfer a dispozitivelor de rețea
`bing`	V:0.08, I:0.47	80	tester empiric stocastic al lățimii de bandă
`bwm-ng`	V:1.0, I:9.1	95	monitor de lățime de bandă mic și simplu, bazat pe consolă
`ethstats`	V:0.05, I:0.36	21	monitor de statistici Ethernet bazat pe consolă
`ipfm`	V:0.04, I:0.09	78	instrument de analizare a lățimii de bandă

Tabela 5.5: Instrumente de optimizare a rețelei

5.6.1 Găsirea MTU optime

NM stabilește în mod normal automat unitatea maximă de transmisie („Maximum Transmission Unit”: MTU) optimă.

În unele cazuri, este posibil să doriți să stabiliți MTU manual după experimentarea cu ping(8) cu opțiunea „`-M do`” pentru a trimite un pachet ICMP cu diferite dimensiuni de pachete de date. MTU este dimensiunea maximă a pachetului de date fără fragmentare IP plus 28 de octeți pentru IPv4 și plus 48 de octeți pentru IPv6. De exemplu, următoarea comandă găsește MTU pentru conexiunea IPv4 ca fiind 1460 și MTU pentru conexiunea IPv6 ca fiind 1500.

```
$ ping -4 -c 1 -s $((1500-28)) -M do www.debian.org
PING  (149.20.4.15) 1472(1500) bytes of data.
ping: local error: message too long, mtu=1460

---  ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

$ ping -4 -c 1 -s $((1460-28)) -M do www.debian.org
PING  (130.89.148.77) 1432(1460) bytes of data.
1440 bytes from klecker-misc.debian.org (130.89.148.77): icmp_seq=1 ttl=50 time=325 ms

---  ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 325.318/325.318/325.318/0.000 ms
$ ping -6 -c 1 -s $((1500-48)) -M do www.debian.org
```

```
PING www.debian.org(mirror-csail.debian.org (2603:400a:ffff:bb8::801f:3e)) 1452 data bytes
1460 bytes from mirror-csail.debian.org (2603:400a:ffff:bb8::801f:3e): icmp_seq=1 ttl=47 ↩
    time=191 ms

--- www.debian.org ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 191.332/191.332/191.332/0.000 ms
```

Acest proces este descoperirea traseului MTU („Path MTU”: PMTU) (RFC1191), iar comanda tracepath(8) poate automatiza acest proces.

mediul de rețea	**MTU**	**justificare**
Legătură dial-up (IP: PPP)	576	standard
Legătură Ethernet (IP: DHCP sau fixă)	1500	standard și implicită

Tabela 5.6: Reguli de bază pentru valoarea optimă a MTU

În plus față de aceste reguli de bază, trebuie să știți următoarele.

- Orice utilizare a metodelor de tunelare (VPN etc.) poate reduce și mai mult MTU-ul optim din cauza suprasarcinii generate.
- Valoarea MTU nu trebuie să depășească valoarea PMTU determinată experimental.
- O valoare MTU mai mare este, în general, mai bună atunci când sunt îndeplinite alte limitări.

Dimensiunea maximă a segmentului (MSS) este utilizată ca măsură alternativă a dimensiunii pachetului. Relația dintre MSS și MTU este următoarea.

- MSS = MTU - 40 pentru IPv4
- MSS = MTU - 60 pentru IPv6

Notă
Optimizarea bazată pe iptables(8) (a se vedea Secțiune 5.7) poate limita dimensiunea pachetelor prin MSS și este utilă pentru router. A se vedea „TCPMSS” în iptables(8).

5.6.2 Optimizare WAN TCP

Debitul TCP poate fi maximizat prin ajustarea parametrilor dimensiunii memoriei tampon TCP, așa cum se arată în „TCP tuning -- ajustarea TCP” pentru rețelele WAN moderne cu lățime de bandă mare și latență ridicată. Până în prezent, configurația implicită actuală a Debian funcționează bine chiar și pentru rețeaua mea LAN conectată prin serviciul rapid FTTP de 1 Gbps.

5.7 Infrastructura netfilter

Netfilter oferă infrastructura pentru stateful firewall -- paravan de protecție dinamic (cu stări memorate) și traducerea adreselor de rețea („Network address translation”: NAT) cu module ale nucleului Linux (a se vedea Secțiune 3.10).

Consultați Proiectul „nftables” de pe netfilter.org pentru a afla mai multe despre noua comandă nft(8) din spațiul utilizatorului din pachetul `nftables`. (Nucleul Linux >= 3.13)

pachete	popcon(popularitate) și dimensiune	dimensiune instalată	descriere
nftables	V:266, I:880	189	instrumente de administrare pentru filtrarea pachetelor și NAT (Netfilter) (succesor al {ip,ip6,arp,eb}tables)
iptables	V:381, I:635	2408	instrumente de administrare pentru netfilter (iptables(8) pentru IPv4, ip6tables(8) pentru IPv6)
arptables	V:0.1, I:1.6	102	instrumente de administrare pentru netfilter (arptables(8) pentru ARP)
ebtables	V:14, I:23	276	instrumente de administrare pentru netfilter (ebtables(8) pentru punte Ethernet)
iptstate	V:0.1, I:1.6	121	monitorizează continuu starea netfilter (similar cu top(1))
ufw	V:83, I:109	862	Uncomplicated Firewall (UFW) -- (paravan de protecție fără complicații) este un program pentru gestionarea unui paravan de protecție netfilter
gufw	V:6, I:11	3682	interfață grafică cu utilizatorul pentru Uncomplicated Firewall (UFW)
firewalld	V:15, I:22	2581	firewalld este un program de paravan de protecție gestionat dinamic, cu suport pentru zone de rețea
firewall-config	V:0.8, I:3.2	1096	interfață grafică cu utilizatorul pentru firewalld
shorewall-init	V:0.17, I:0.33	88	inițializarea Shoreline Firewall (paravan de protecție al liniei de coastă)
shorewall	V:2.1, I:4.7	3090	Shoreline Firewall, generator de fișiere de configurare netfilter
shorewall-lite	V:0.03, I:0.06	71	Shoreline Firewall, generator de fișiere de configurare netfilter (versiune simplificată)
shorewall6	V:0.6, I:1.2	1334	Shoreline Firewall, generator de fișiere de configurare netfilter (versiune IPv6)
shorewall6-lite	V:0.01, I:0.02	71	Shoreline Firewall, generator de fișiere de configurare netfilter (versiune simplificată, IPv6)

Tabela 5.7: Lista instrumentelor de paravan de protecție

Consultați Proiectul „iptables” de pe netfilter.org pentru versiunea mai veche a iptables(8) din spațiul utilizatorului și pentru comenzi similare din pachetele `iptables` și altele asemenea. (Nucleul Linux >= 2.4.x)

Puteți configura manual netfilter în mod interactiv din shell, salva starea acestuia cu „`nft list ruleset > file`” sau iptables-save(8) și s-o restaurați prin scriptul init cu „`nft -f file`” sau iptables-restore(8) la repornirea sistemului.

Scripturile de asistență pentru configurare, precum shorewall, facilitează acest proces.

Consultați documentația la Documentația Netfilter (sau în „`/usr/share/doc/iptables/html/`”).

- Linux Networking-concepts HOWTO
- Linux 2.4 Packet Filtering HOWTO
- Linux 2.4 NAT HOWTO
- nftables HOWTO

Capitolul 6

Aplicații de rețea

După stabilirea conexiunii la rețea (a se vedea Cap. 5), puteți rula diverse aplicații de rețea.

Indicație
Pentru un ghid modern specific Debian privind infrastructura de rețea, citiți Manualul administratorului Debian — Infrastructura de rețea.

Indicație
Dacă ați activat „Verificarea în doi pași” cu un anumit ISP, trebuie să obțineți o parolă de aplicație pentru a accesa serviciile POP și SMTP din programul dvs. Este posibil să fie necesar să aprobați în prealabil adresa IP a gazdei.

6.1 Navigatoare Web

Există multe pachete de navigatoare web pentru accesarea conținutului la distanță cu ajutorul Protocolului de transfer hipertext (HTTP).

6.1.1 Falsificarea șirului User-Agent

Pentru a accesa unele siturile web excesiv de restrictive, poate fi necesar să falsificați șirul User-Agent returnat de programul de navigare web. A se vedea:

- MDN Web Docs: userAgent -- Documentație web MDN: userAgent
- Chrome Developers: Override the user agent string -- Dezvoltatori Chrome: Suprascrieți șirul agentului de utilizator
- How to change your user agent -- Cum să schimbați agentul de utilizator
- How to Change User-Agent in Chrome, Firefox, Safari, and more -- Cum să schimbați User-Agent în Chrome, Firefox, Safari și altele
- How to Change Your Browser’s User Agent Without Installing Any Extensions -- Cum să schimbați agentul de utilizator al navigatorului fără a instala extensii
- How to change the User Agent in Gnome Web (epiphany) -- Cum se schimbă agentul de utilizator în Gnome Web (epiphany)

pachet	popcon(popularitate)	dimensiune	tipul	descrierea navigatorului web
chromium	V:31, I:100	320285	X	Chromium, (navigator cu codul-sursă deschis de la Google)
firefox	V:15, I:21	308351	, ,	Firefox, (navigator cu codul-sursă deschis de la Mozilla, disponibil numai în Debian Unstable)
firefox-esr	V:186, I:425	299827	, ,	Firefox ESR, (versiune Firefox cu suport extins)
epiphany-browser	V:2, I:10	6735	, ,	GNOME, conform cu liniile directoare privind interfața umanăHuman Interface Guidelines: HIG, Epiphany
konqueror	V:26, I:111	8179	, ,	KDE, Konqueror
dillo	V:0.4, I:3.9	2203	, ,	Dillo, (navigator de dimensiuni reduse, bazat pe FLTK)
w3m	V:10, I:129	2853	text	w3m
lynx	V:29, I:455	2031	, ,	Lynx
elinks	V:2, I:15	1791	, ,	ELinks
links	V:2, I:20	2321	, ,	Links (text only)
links2	V:0.8, I:9.4	5466	grafică	Linkuri (grafică de consolă fără X)

Tabela 6.1: Lista navigatoarelor web

6.1.2 Extensie pentru navigator

Toate navigatoarele cu interfață grafică modernă acceptă extensii de navigator bazate pe cod sursă extensii de navigator și acestea sunt în curs de standardizare ca extensii web.

6.2 Sistemul de poștă electronică

Această secțiune se concentrează pe stațiile de lucru mobile tipice cu conexiuni la internet la nivel de utilizator casnic.

Atenție
Dacă doriți să configurați serverul de poștă electronică pentru a schimba mesaje direct cu Internetul, ar fi bine să citiți acest document elementar.

6.2.1 Noțiuni de bază despre poșta electronică

Un mesaj de poștă electronică este format din trei componente: plicul mesajului, antetul mesajului și corpul mesajului.

- Informațiile „Către (To)” și „De la (From)” din plicul mesajului sunt utilizate de SMTP pentru a livra mesajul electronic; (informațiile „De la” din plicul mesajului sunt denumite și adresă de returnare , From_ etc.).
- Informațiile „Către” și „De la” din antetul mesajului sunt afișate de clientul de poștă electronică; (deși cel mai frecvent aceste informații sunt identice cu cele din plicul mesajului, nu întotdeauna este așa).
- Formatul mesajului de poștă electronică care acoperă datele din antet și corp este extins de („Multipurpose Internet Mail Extensions”: MIME) -- extensiile cu scopuri multiple pentru poșta electronică din Internet, de la text ASCII simplu la alte codificări de caractere, precum și atașamente de fișiere audio, video, imagini și programe de aplicații.

Clienții de poștă electronică cu interfață grafică completă oferă toate funcțiile următoare, utilizând configurația intuitivă bazată pe interfața grafică.

- Acesta creează și interpretează antetul mesajului și datele din corp folosind Multipurpose Internet Mail Extensions (MIME) pentru a gestiona tipul de date și codificarea conținutului.
- Acesta se autentifică pe serverele SMTP și IMAP ale ISP-ului utilizând autentificarea de acces de bază învechită sau autentificarea OAuth 2.0 modernă; (pentru OAuth 2.0, configurați-l prin intermediul opțiunilor de configurare ale mediului de birou. De exemplu, „Configurări” -> „Conturi online”).
- Acesta trimite mesajul către serverul SMTP „smarthost” al ISP-ului care ascultă pe portul de trimitere a mesajelor (587).
- Primește mesajul stocat pe serverul ISP de la portul TLS/IMAP4 (993).
- Poate filtra mesajele după atributele lor.
- Poate oferi funcționalități suplimentare: Contacte, Calendar, Sarcini, Notițe.

pachet	popcon(popularitate) și dimensiunea		descrierea pe scurt a tipului
`evolution`	V:27, I:229	489	program cu interfață grafică X (GNOME, suită de programe de lucru în grup)
`thunderbird`	V:39, I:102	303329	program cu interfață grafică X (GTK, Mozilla Thunderbird)
`kmail`	V:43, I:103	25265	program cu interfață grafică X (KDE)
`mutt`	V:10, I:82	7334	program terminal de caractere probabil utilizat cu `vim`
`mew`	V:0.01, I:0.14	2319	program terminal de caractere sub `(x)emacs`

Tabela 6.2: Lista agenților de utilizator de poștă electronică („Mail User Agent”: MUA)

6.2.2 Limita serviciilor poștale moderne

Serviciile poștale moderne sunt supuse unor restricții pentru a minimiza expunerea la problemele legate de spam (mesaje electronice nedorite și nesolicitate).

- Nu este realist să rulezi un server SMTP pe o rețea de uz casnic pentru a trimite mesaje în mod fiabil direct către gazda la distanță.
- Un mesaj poate fi respins în mod discret de orice gazdă pe traseul către destinație, cu excepția cazului în care pare cât mai autentic posibil.
- Nu este realist să ne așteptăm ca un singur smarthost să trimită în mod fiabil mesaje de la adrese de poștă electronică fără legătură cu gazda la distanță.

Acest lucru se datorează faptului că:

- Conexiunile SMTP (portul 25) de la gazdele deservite de rețeaua de uz casnic către Internet sunt blocate.
- Conexiunile la portul SMTP (25) către gazdele deservite de rețeaua publică de pe Internet sunt blocate.
- Mesajele trimise de la gazdele deservite de rețeaua de uz casnic către Internet pot fi trimise numai prin portul de trimitere a mesajelor (587).
- Tehnici anti-spam precum DomainKeys Identified Mail (DKIM), Sender_Policy_Framework (SPF) și Domain-based Message Authentication, Reporting and Conformance (DMARC) sunt utilizate pe scară largă pentru filtrarea mesajelor de poștă electronică.
- Serviciul DomainKeys Identified Mail poate fi furnizat pentru mesajele dvs. trimise prin intermediul smarthost.
- Smarthostul poate rescrie adresa de poștă electronică sursă din antetul mesajului cu adresa contului dvs. de poștă electronică de pe smarthost, pentru a preveni falsificarea adresei de poștă electronică.

6.2.3 Așteptări istorice privind serviciul poștal

Unele programe din Debian se așteaptă să acceseze comanda `/usr/sbin/sendmail` pentru a trimite mesaje de poștă electronică ca opțiune implicită sau personalizată, deoarece serviciul de poștă electronică pe un sistem UNIX funcționa în mod tradițional astfel:

- Un mesaj de poștă electronică este creat ca fișier text.
- Mesajul de poștă electronică este transmis comenzii `/usr/sbin/sendmail`.
- Pentru adresa de destinație de pe același gazdă, comanda `/usr/sbin/sendmail` efectuează livrarea locală a mesajului prin adăugarea acestuia la fișierul `/var/mail/$username`.
  - Comenzi care necesită această funcție: `apt-listchanges`, `cron`, `at`, ...
- Pentru adresa de destinație de pe gazda la distanță, comanda `/usr/sbin/sendmail` efectuează transferul la distanță al mesajului electronic către gazda de destinație găsită de înregistrarea DNS MX utilizând SMTP.
  - Comenzi care așteaptă această funcție: `popcon`, `reportbug`, `bts`, ...

6.2.4 Agentul de transport al poștei electronice („Mail transport agent”: MTA)

Stațiile de lucru mobile Debian pot fi configurate doar cu clienți de poștă electronică cu interfață grafică completă clienți de poștă electronică fără programul agent de transfer de poștă electronică (MTA) după Debian 12 Bookworm.

Debian instala în mod tradițional un program MTA pentru a oferi suport programelor care așteptau comanda `/usr/sbin/sen`
Un astfel de MTA pe stațiile de lucru mobile trebuie să facă față Secțiune 6.2.2 și Secțiune 6.2.3.

Pentru stațiile de lucru mobile, alegerea tipică pentru MTA este fie `exim4-daemon-light`, fie `postfix`, cu opțiunea de instalare „Mail sent by smarthost; received via SMTP or fetchmail” (Mesaj trimis de smarthost; primit prin SMTP sau fetchmail) selectată. Acestea sunt MTA-uri de dimensiuni reduse, care respectă „`/etc/aliases`”.

Indicație
Configurarea `exim4` pentru a trimite corespondența electronică prin intermediul mai multor smarthosturi corespunzătoare pentru mai multe adrese de poștă electronică sursă nu este o operațiune simplă. Dacă aveți nevoie de această funcționalitate pentru anumite programe, configurați-le să utilizeze `msmtp`, care este ușor de configurat pentru mai multe adrese de poștă electronică sursă. Apoi, lăsați MTA principal doar pentru o singură adresă de poștă electronică.

6.2.4.1 Configurarea exim4

Pentru poșta electronică prin intermediul smarthost, (re)configurați pachetele `exim4-*` după cum urmează.

```
$ sudo systemctl stop exim4
$ sudo dpkg-reconfigure exim4-config
```

Selectați „e-mail trimis prin smarthost; primit prin SMTP sau fetchmail” pentru „Tipul general de configurare a poștei”.

Definiți „Numele de poștă al sistemului:” la valoarea implicită ca FQDN (consultați Secțiune 5.1.1).

Definiți „Adresele IP pe care să le monitorizeze pentru conexiunile SMTP primite:” la valoarea implicită „127.0.0.1 ; ::1”.

Eliminați conținutul din „Alte destinații pentru care se acceptă corespondența:”.

Eliminați conținutul din „Mașini pentru redirecționarea mesajelor către:”.

Definiți „Adresa IP sau numele gazdei smarthost-ului de ieșire:” la „smtp.hostname.dom:587”.

pachet	popcon(popularitate) și dimensiune		descriere
exim4-daemon-light	V:200, I:209	1776	agentul de transport de poștă Exim4 (MTA: implicit în Debian)
exim4-daemon-heavy	V:4.8, I:4.9	1954	agentul de transport de poștă Exim4 (MTA: alternativă flexibilă)
exim4-base	V:205, I:214	1685	documentația Exim4 (text) și fișiere comune
exim4-doc-html	I:0.98	3809	documentația Exim4 (html)
exim4-doc-info	I:0.55	649	documentația Exim4 (info)
postfix	V:102, I:107	4274	agentul de transport de poștă Postfix (MTA: alternativă securizată)
postfix-doc	I:4.3	5028	documentația Postfix (html+text)
sasl2-bin	V:4.5, I:9.9	381	implementarea API-ului SASL Cyrus (supliment postfix pentru SMTP AUTH)
cyrus-sasl2-doc	I:0.68	2140	Cyrus SASL - documentație
msmtp	V:9, I:15	822	MTA cu o dimensiune foarte mică
msmtp-mta	V:7.2, I:9.1	139	MTA cu o dimensiune foarte mică (extensie de compatibilitate cu sendmail pentru `msmtp`)
nullmailer	V:7.4, I:7.7	483	dezactivează MTA, fără poștă locală
ssmtp	V:3.9, I:5.9	134	dezactivează MTA, fără poștă locală
sendmail-bin	V:10, I:10	1954	MTA cu funcții complete (numai dacă sunteți deja familiarizat cu acesta)
git-email	V:0.3, I:9.3	1229	programul git-send-email(1) pentru trimiterea unei serii de mesaje electronice cu plasturi (corecții, așa numitele „patch”)

Tabela 6.3: Lista pachetelor bazice legate de agentul de transport al poștei

Selectați „Nu” pentru „Ascundeți numele adresei locale în mesajele trimise?”; (utilizați „`/etc/email-addresses`” ca în Secțiune 6.2.4.3, în schimb).

Răspundeți la „Mențineți numărul de interogări DNS la un nivel minim (Dial-on-Demand)?” cu una dintre următoarele opțiuni.

- „Nu” dacă sistemul este conectat la Internet în timpul pornirii.
- „Da” dacă sistemul **nu** este conectat la Internet în timpul pornirii.

Stabiliți „Metoda de livrare pentru poșta locală:” la „format mbox în /var/mail/”.

Selectați „Da” pentru „Împărțiți configurația în fișiere mici?:”.

Creați intrări de parolă pentru smarthost editând „`/etc/exim4/passwd.client`”.

```
$ sudo vim /etc/exim4/passwd.client
 ...
$ cat /etc/exim4/passwd.client
^smtp.*\.hostname\.dom:username@hostname.dom:password
```

Configurați exim4(8) cu „`QUEUERUNNER='queueonly'`”, „`QUEUERUNNER='nodaemon'`”, etc. în „`/etc/default/exim4`” pentru a minimiza utilizarea resurselor sistemului. (opțional)

Porniți `exim4` prin următoarea comandă.

```
$ sudo systemctl start exim4
```

Numele gazdei din „`/etc/exim4/passwd.client`” nu trebuie să fie cel al alias. Verificați numele real al gazdei folosind următoarea comandă.

```
$ host smtp.hostname.dom
smtp.hostname.dom is an alias for smtp99.hostname.dom.
smtp99.hostname.dom has address 123.234.123.89
```

Folosesc expresii regulate în „`/etc/exim4/passwd.client`” pentru a rezolva problema cu numele-alias. SMTP AUTH funcționează probabil chiar dacă ISP mută gazda indicată de numele-alias.

Puteți actualiza manual configurația `exim4` urmând pașii de mai jos:

- Actualizați fișierele de configurare `exim4` din „`/etc/exim4/`”.
  - creând fișierul „`/etc/exim4/exim4.conf.localmacros`” pentru a defini MACRO-urile și editând fișierul „`/etc/exi`
    (configurație nedivizată)
  - creând fișiere noi sau editând fișierele existente în subdirectoarele „`/etc/exim4/exim4.conf.d`”. (configurație divizată)
- Rulați „`systemctl reload exim4`”.

Atenție
Pornirea `exim4` durează mult timp dacă s-a ales „Nu” (valoare implicită) pentru întrebarea debconf „Păstrați numărul de interogări DNS la minimum (Dial-on-Demand)?” și sistemul **nu** este conectat la Internet în timpul pornirii.

Vă rugăm să citiți ghidul oficial la: „`/usr/share/doc/exim4-base/README.Debian.gz`” și update-exim4.conf(8).

Avertisment
Din considerente practice, utilizați SMTP cu STARTTLS pe portul 587 sau SMTPS (SMTP peste SSL) pe portul 465, în loc de SMTP simplu pe portul 25.

6.2.4.2 Configurarea postfix cu SASL

Pentru poșta electronică prin intermediul smarthost, ar trebui să citiți mai întâi documentația postfix și paginile cheie ale manualului.

comandă	funcție
postfix(1)	Controlul programului postfix
postconf(1)	Instrumentul de configurare postfix
postconf(5)	Parametrii de configurare postfix
postmap(1)	Administrarea tabelului de căutare postfix
postalias(1)	Administrarea bazei de date de alias postfix

Tabela 6.4: Lista paginilor importante din manualul postfix

(Re)configurați pachetele `postfix` și `sasl2-bin` după cum urmează.

```
$ sudo systemctl stop postfix
$ sudo dpkg-reconfigure postfix
```

Alegeți „Internet cu smarthost”.

Definiți „Numele-gazdei de releu SMTP (în alb pentru niciunul):” la „`[smtp.hostname.dom]:587`” și configurați-l după cum urmează.

```
$ sudo postconf -e 'smtp_sender_dependent_authentication = yes'
$ sudo postconf -e 'smtp_sasl_auth_enable = yes'
$ sudo postconf -e 'smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd'
$ sudo postconf -e 'smtp_sasl_type = cyrus'
$ sudo vim /etc/postfix/sasl_passwd
```

Creați intrări de parolă pentru smarthost.

```
$ cat /etc/postfix/sasl_passwd
[smtp.hostname.dom]:587     username:password
$ sudo postmap hush:/etc/postfix/sasl_passwd
```

Porniți `postfix` cu următoarele.

```
$ sudo systemctl start postfix
```

Aici, utilizarea „[" și „]" în dialogul `dpkg-reconfigure` și „`/etc/postfix/sasl_passwd`" asigură că nu se verifică înregistrarea MX, ci se utilizează direct numele exact al gazdei specificat. Consultați „Activarea autentificării SASL în clientul SMTP Postfix" în „`/usr/share/doc/postfix/html/SASL_README.html`".

6.2.4.3 Configurarea adresei de poștă electronică

Există câteva fișiere de configurare a adreselor de poștă pentru transportul, livrarea și agenții utilizatorilor de poștă electronică.

fișier	funcție	aplicație
`/etc/mailname`	numele gazdei implicite pentru corespondența (expediată)	Specific Debian, mailname(5)
`/etc/email-addresses`	falsificarea numelui gazdei pentru corespondența trimisă	specific exim(8), exim4-config_files(5)
`/etc/postfix/generic`	falsificarea numelui gazdei pentru corespondența trimisă	specific postfix(1), activat după executarea comenzii postmap(1).
`/etc/aliases`	alias al numelui contului pentru mesajele primite	general, activat după executarea comenzii newaliases(1)

Tabela 6.5: Lista fișierelor de configurare legate de adresele de poștă electronică

mailname din fișierul „`/etc/mailname`" este de obicei un nume de domeniu complet calificat (FQDN) care se rezolvă la una dintre adresele IP ale gazdei. Pentru stația de lucru mobilă care nu are un nume de gazdă cu adresă IP rezolvabilă, stabiliți acest **mailname** la valoarea „`hostname -f`". (Aceasta este o alegere sigură și funcționează atât pentru `exim4-*`, cât și pentru `postfix`.)

Indicație
Conținutul fișierului „`/etc/mailname`" este utilizat de multe programe non-MTA pentru comportamentul lor implicit. Pentru `mutt`, definiți variabilele „`hostname`" și „`from`" în fișierul `~/muttrc` pentru a suprascrie valoarea **mailname**. Pentru programele din pachetul `devscripts`, cum ar fi bts(1) și dch(1), exportați variabilele de mediu „`$DEBFULLNAME`" și „`$DEBEMAIL`" pentru a le suprascrie.

Indicație
Pachetul `popularity-contest` trimite în mod normal mesaje de poștă electronică din contul root cu FQDN. Trebuie să definiți `MAILFROM` în `/etc/popularity-contest.conf` așa cum este descris în fișierul `/usr/share/popularity-contest/default.conf`. În caz contrar, mesajul dvs. va fi respins de serverul SMTP smarthost. Deși este un proces tedios, această abordare este mai sigură decât rescrierea adresei sursă pentru toate mesajele de la root de către MTA și ar trebui utilizată pentru alți demoni și scripturi cron.

Când se definește **mailname** ca „`hostname -f`”, falsificarea adresei de poștă electronică sursă prin MTA poate fi realizată după cum urmează.

- Fișierul „`/etc/email-addresses`” pentru exim4(8), așa cum se explică în exim4-config_files(5)
- Fișierul „`/etc/postfix/generic`” pentru postfix(1), așa cum se explică în generic(5)

Pentru `postfix`, sunt necesare următoarele etape suplimentare.

```
# postmap hash:/etc/postfix/generic
# postconf -e 'smtp_generic_maps = hash:/etc/postfix/generic'
# postfix reload
```

Puteți testa configurația adresei de poștă electronică utilizând următoarele comenzi.

- exim(8) cu opțiunile `-brw, -bf, -bF, -bV, …`
- postmap(1) cu opțiunea `-q`.

Indicație
Exim vine cu mai multe programe auxiliare, cum ar fi exiqgrep(8) și exipick(8). Consultați „`dpkg -L exim4-base|grep man8/`” pentru comenzile disponibile.

6.2.4.4 Operații MTA de bază

Există mai multe operații MTA de bază. Unele pot fi efectuate prin intermediul interfeței de compatibilitate sendmail(1).

comanda «exim»	comanda «postfix»	descriere
`sendmail`	`sendmail`	citește mesajele din intrarea standard și organizează livrarea (`-bm`)
`mailq`	`mailq`	listează coada de mesaje cu starea și ID-ul cozii (`-bp`)
`newaliases`	`newaliases`	inițializează baza de date alias (`-I`)
`exim4 -q`	`postqueue -f`	elimină mesajele în așteptare (`-q`)
`exim4 -qf`	`postsuper -r ALL deferred; postqueue -f`	elimină toate mesajele
`exim4 -qff`	`postsuper -r ALL; postqueue -f`	elimină chiar și mesajele înghețate
`exim4 -Mg queue_id`	`postsuper -h queue_id`	îngheață un mesaj după ID-ul cozii sale
`exim4 -Mrm queue_id`	`postsuper -d queue_id`	elimină un mesaj după ID-ul cozii sale
N/D	`postsuper -d ALL`	elimină toate mesajele

Tabela 6.6: Lista operațiilor de bază ale MTA

Indicație
Ar fi o idee bună să eliminați toate mesajele prin intermediul unui script din „`/etc/ppp/ip-up.d/*`”.

6.3 Serverul și instrumentele de acces la distanță (SSH)

Secure SHell (SSH) este modalitatea **sigură** de conectare la Internet. O versiune gratuită a SSH numită OpenSSH este disponibilă sub forma pachetelor `openssh-client` și `openssh-server` în Debian.

Pentru utilizator, ssh(1) funcționează ca un telnet(1) mai inteligent și mai sigur. Spre deosebire de comanda `telnet`, comanda `ssh` nu se oprește la caracterul de eludare `telnet` (implicit CTRL-]).

pachet	popcon(popularitate)	dimensiune	instrument	descriere
`openssh-client`	V:682, I:997	3611	ssh(1)	client de shell securizat
`openssh-server`	V:774, I:820	3615	sshd(8)	server de shell securizat
`ssh-askpass`	I:15	103	ssh-askpass(1)	solicită utilizatorului o frază de acces pentru ssh-add (X simplu)
`ssh-askpass-gnome`	V:0.5, I:2.9	46	ssh-askpass-gnome(1)	solicită utilizatorului o frază de acces pentru ssh-add (GNOME)
`ssh-askpass-fullscreen`	V:0.09, I:0.40	41	ssh-askpass-fullscreen(1)	solicită utilizatorului o frază de acces pentru ssh-add (GNOME) cu un plus de atracție vizuală
`shellinabox`	V:0.61, I:0.94	528	shellinaboxd(1)	server web pentru emulator terminal VT100 accesibil din navigator

Tabela 6.7: Lista serverelor și instrumentelor de acces la distanță

Deși `shellinabox` nu este un program SSH, este menționat aici ca o alternativă interesantă pentru accesul la terminalul la distanță.

A se vedea de asemenea Secțiune 7.9 pentru conectarea la programe client X la distanță.



Atenție
Consultați Secțiune 4.6.3 dacă SSH-ul dvs. este accesibil din Internet.

Indicație
Vă rugăm să utilizați programul screen(1) pentru a permite procesului shell la distanță să supraviețuiască conexiunii întrerupte (consultați Secțiune 9.1.2).

6.3.1 Noțiuni de bază despre SSH

Demonul SSH al OpenSSH acceptă numai protocolul SSH 2.

Vă rugăm să citiți „`/usr/share/doc/openssh-client/README. Debian.gz`”, ssh(1), sshd(8), ssh-keygen(1), ssh-add(1) și ssh-agent(1).



Avertisment
„`/etc/ssh/sshd_not_to_be_run`” nu trebuie să fie prezent dacă se dorește rularea serverului OpenSSH.
Nu activați autentificarea bazată pe rhost (`HostbasedAuthentication` în `/etc/ssh/sshd_config`).

Următorul cod inițializează o conexiune ssh(1) de la un client.

fișier de configurare	descrierea fișierului de configurare
`/etc/ssh/ssh_config`	valorile implicite ale clientului SSH, consultați ssh_config(5)
`/etc/ssh/sshd_config`	valorile implicite ale serverului SSH, consultați sshd_config(5)
`~/.ssh/authorized_keys`	cheile SSH publice implicite pe care clienții le utilizează pentru a se conecta la acest cont pe acest server SSH
`~/.ssh/id_rsa`	cheia secretă SSH-2 RSA a utilizatorului
`~/.ssh/id_`*`key-type-name`*	cheie secretă SSH-2 *key-type-name* precum `ecdsa`, `ed25519`, ... a utilizatorului

Tabela 6.8: Lista fișierelor de configurare SSH

comandă	descriere
`ssh username@hostname.domain.ext`	se conectează cu modul implicit
`ssh -v username@hostname.domain.ext`	se conectează cu modul implicit cu mesaje de depanare
`ssh -o PreferredAuthentications=password username@hostname.domain.ext`	forțează utilizarea parolei cu SSH versiunea 2
`ssh -t username@hostname.domain.ext passwd`	rulează programul `passwd` pentru a actualiza parola pe o gazdă la distanță

Tabela 6.9: Listă de exemple de pornire a clientului SSH

6.3.2 Numele de utilizator pe gazda la distanță

Dacă utilizați același nume de utilizator pe gazda locală și pe cea la distanță, puteți elimina introducerea „`username@`”.

Chiar dacă utilizați nume de utilizator diferite pe gazda locală și pe cea la distanță, puteți elimina acest lucru utilizând „`~/.ssh/config`”. Pentru serviciul Debian Salsa cu numele de cont „`foo-guest`”, modificați „`~/.ssh/config`” astfel încât să conțină următoarele.

```
Host salsa.debian.org people.debian.org
User foo-guest
```

6.3.3 Connectarea fără parole la distanță

Se poate evita memorarea parolelor pentru sistemele la distanță utilizând „`PubkeyAuthentication`” (protocolul SSH-2).

Pe sistemul la distanță, definiți intrările respective, „`PubkeyAuthentication yes`”, în „`/etc/ssh/sshd_config`”.

Generați cheile de autentificare local și instalați cheia publică pe sistemul la distanță, urmând pașii de mai jos.

```
$ ssh-keygen -t rsa
$ cat .ssh/id_rsa.pub | ssh user1@remote "cat - >>.ssh/authorized_keys"
```

Puteți adăuga opțiuni la intrările din „`~/.ssh/authorized_keys`” pentru a limita gazdele și pentru a rula comenzi specifice. Consultați sshd(8) „FORMATUL FIȘIERULUI AUTHORIZED_KEYS”.

6.3.4 Gestionarea clienților SSH străini

Există câțiva clienți SSH liberi disponibili pentru alte platforme.

mediu	program SSH liber
Windows	puTTY (PuTTY: un client SSH și Telnet liber) (GPL)
Windows (cygwin)	SSH în cygwin (Cygwin: Simțiți atmosfera Linux - pe Windows) (GPL)
Mac OS X	OpenSSH; utilizează `ssh` în aplicația Terminal (GPL)

Tabela 6.10: Lista clienților SSH liberi pentru alte platforme

6.3.5 Configurarea ssh-agent

Este mai sigur să vă protejați cheile secrete de autentificare SSH cu o frază de acces. Dacă nu a fost stabilită o frază de acces, utilizați „`ssh-keygen -p`” pentru a o stabili.

Plasați cheia SSH publică (de exemplu, „`~/.ssh/id_rsa.pub`”) în „`~/.ssh/authorized_keys`” pe un host la distanță, utilizând o conexiune bazată pe parolă la hostul la distanță, așa cum este descris mai sus.

```
$ ssh-agent bash
$ ssh-add ~/.ssh/id_rsa
Enter passphrase for /home/username/.ssh/id_rsa:
Identity added: /home/username/.ssh/id_rsa (/home/username/.ssh/id_rsa)
```

De aici înainte nu mai este necesară o parolă de la distanță pentru următoarea comandă.

```
$ scp foo username@remote.host:foo
```

Apăsați ^D pentru a încheia sesiunea ssh-agent.

Pentru serverul X, scriptul normal de pornire Debian execută `ssh-agent` ca proces părinte. Deci, trebuie să executați `ssh-add` o singură dată. Pentru mai multe informații, citiți ssh-agent(1) și ssh-add(1).

6.3.6 Trimiterea unui mesaj de la o gazdă de la distanță

Dacă aveți un cont shell SSH pe un server cu setări DNS corespunzătoare, puteți trimite un mesaj generat pe stația dvs. de lucru ca un mesaj electronic trimis efectiv de pe serverul la distanță.

```
$ ssh username@example.org /usr/sbin/sendmail -bm -ti -f "username@example.org" < mail_data ↩
    .txt
```

6.3.7 Redirecționarea porturilor pentru tunelarea SMTP/POP3

Pentru a stabili o conexiune la portul 25 al serverului de la distanță `remote-server` de la portul 4025 al `localhost` și la portul 110 al serverului de la distanță `remote-server` de la portul 4110 al `localhost` prin `ssh`, executați pe gazda locală următoarea comandă.

```
# ssh -q -L 4025:remote-server:25 4110:remote-server:110 username@remote-server
```

Aceasta este o metodă sigură de a stabili conexiuni la serverele SMTP/POP3 prin Internet. Stabiliți intrarea „`AllowTcpForwa`
la „`yes`” în fișierul „`/etc/ssh/sshd_config`” al gazdei de la distanță.

6.3.8 Cum să opriți sistemul de la distanță pe SSH

Trebuie să protejați procesul care execută „`shutdown -h now`” (consultați Secțiune 1.1.8) împotriva terminării SSH utilizând comanda at(1) (consultați Secțiune 9.4.13) prin următoarele.

```
# echo "shutdown -h now" | at now
```

Executarea comenzii „`shutdown -h now`” în sesiunea screen(1) (vezi Secțiune 9.1.2) este o altă modalitate de a face același lucru.

6.3.9 Soluționarea problemelor SSH

Dacă aveți probleme, verificați permisiunile fișierelor de configurare și rulați `ssh` cu opțiunea „`-v`”.

Utilizați opțiunea „`-p`” dacă sunteți root și aveți probleme cu un paravan de protecție; aceasta evită utilizarea porturilor serverului 1 — 1023.

Dacă conexiunile `ssh` la un sit la distanță încetează brusc să funcționeze, este posibil ca acest lucru să fie rezultatul unei intervenții a administratorului de sistem, cel mai probabil o modificare a „`host_key`” în timpul întreținerii sistemului. După ce vă asigurați că acesta este cazul și că nimeni nu încearcă să falsifice gazda la distanță printr-un truc inteligent, puteți restabili conexiunea eliminând intrarea „`host_key`” din „`~/.ssh/known_hosts`” de pe gazda locală.

6.4 Serverul de imprimare și utilitățile

În vechiul sistem de tip Unix, BSD Line printer daemon (lpd) era standardul, iar formatul standard de imprimare al software-ului liber clasic era PostScript (PS). Unele sisteme de filtrare erau utilizate împreună cu Ghostscript pentru a permite imprimarea pe imprimante non-PostScript. A se vedea Secțiune 11.4.1.

În sistemul Debian modern, Common UNIX Printing System (CUPS) este standardul de facto, iar formatul standard de imprimare al software-ului liber modern este Portable Document Format (PDF).

CUPS utilizează Internet Printing Protocol (IPP). IPP este standardul de facto multiplatformă pentru imprimarea la distanță cu capacitate de comunicare bidirecțională.

Datorită funcției de conversie automată dependentă de formatul fișierului din sistemul CUPS, introducerea oricăror date în comanda `lpr` ar trebui să genereze rezultatul de imprimare așteptat; (în CUPS, `lpr` poate fi activat prin instalarea pachetului `cups-bsd`).

Sistemul Debian dispune de câteva pachete importante pentru serverele de imprimare și utilități.

pachet	popcon(popularitate)	dimensiune	port	descriere
`lpr`	V:1.9, I:2.2	378	imprimantă (515)	BSD lpr/lpd (Demonul de imprimare în linie)
`cups`	V:144, I:447	1088	IPP (631)	Serverul CUPS pentru imprimare prin Internet
`cups-client`	V:152, I:459	429	, ,	Comenzi imprimare System V pentru CUPS: lp(1), lpstat(1), lpoptions(1), cancel(1), lpmove(8), lpinfo(8), lpadmin(8), …
`cups-bsd`	V:34, I:182	131	, ,	Comenzi BSD pentru imprimare pentru CUPS: lpr(1), lpq(1), lprm(1), lpc(8)
`printer-driver-gutenprint`	V:11, I:52	1117	Nu este aplicabil	controlori de imprimantă pentru CUPS

Tabela 6.11: Lista serverelor de imprimare și a utilităților

Indicație
Puteți configura sistemul CUPS accesând adresa „http://localhost:631/” în navigatorul web.

6.5 Alte servere de aplicații de rețea

Iată alte servere de aplicații de rețea.

Protocolul CIFS (Common Internet File System Protocol) este același protocol ca Server Message Block (SMB) și este utilizat pe scară largă de Microsoft Windows.

pachet	popcon(popularitate)	număr port	protocol	descriere
telnetd	V:0.3, I:1.6	56	TELNET	server TELNET
nfs-kernel-server	V:47, I:54	874	NFS	partajarea fișierelor Unix
samba	V:110, I:123	5182	SMB	partajarea fișierelor și imprimantelor Windows
netatalk	V:0.62, I:0.85	933	ATP	partajarea fișierelor și imprimantelor Apple/Mac (AppleTalk)
proftpd-basic	V:3.3, I:8.4	452	FTP	descărcare fișier general
apache2	V:175, I:211	587	HTTP	server web general
squid	V:9.0, I:9.4	9583	, ,	server proxy web general
bind9	V:33, I:35	904	DNS	adresa IP pentru alte gazde
kea	I:0.51	245	DHCP	adresa IP a clientului însuși

Tabela 6.12: Lista altor servere de aplicații de rețea

Indicație
Consultați Secțiune 4.5.2 pentru integrarea sistemelor de servere.

Indicație
Rezoluția numelui de gazdă este furnizată de obicei de serverul DNS. Pentru adresa IP a gazdei atribuită dinamic de DHCP, Dynamic DNS poate fi configurat pentru rezolvarea numelui de gazdă utilizând `bind9` și `kea`, așa cum este descris în pagina DDNS din wiki-ul Debian.

Indicație
Utilizarea unui server proxy precum `squid` este mult mai eficientă pentru economisirea lățimii de bandă decât utilizarea unui server oglindă local cu conținutul complet al arhivei Debian.

6.6 Alți clienți de aplicații de rețea

Iată alți clienți de aplicații de rețea.

6.7 Diagnosticul demonilor sistemului

Programul `telnet` permite conectarea manuală la demonii sistemului și diagnosticarea acestora.

Pentru a testa serviciul simplu POP3, încercați următoarele

```
$ telnet mail.ispname.net pop3
```

Pentru a testa serviciul TLS/SSL activat POP3 de către unii furnizori de servicii Internet, aveți nevoie de un client `telnet` cu TLS/SSL activat prin pachetele `telnet-ssl` sau `openssl`.

```
$ telnet -z ssl pop.gmail.com 995
```

```
$ openssl s_client -connect pop.gmail.com:995
```

Următoarele RFC-uri furnizează cunoștințele necesare pentru fiecare demon de sistem.

Utilizarea porturilor este descrisă în „`/etc/services`”.

pachet	popcon(popularitate)	dimensiune	protocol	descriere
netcat-traditional	V:55, I:906	139	TCP/IP	cuțitul elvețian TCP/IP
netcat-openbsd	V:22, I:119	105	TCP/IP	cuțitul elvețian TCP/IP cu suport pentru IPv6, proxy-uri și socluri Unix
openssl	V:846, I:996	2564	SSL	binare Secure Socket Layer (SSL) și instrumente criptografice asociate
stunnel4	V:6.6, I:9.9	24	, ,	învăluitor SSL universal
telnet	V:10, I:190	56	TELNET	client TELNET
nfs-common	V:147, I:191	1158	NFS	partajarea fișierelor Unix
smbclient	V:25, I:206	2117	SMB	client de partajare a fișierelor și imprimantelor MS Windows
cifs-utils	V:32, I:118	368	, ,	comenzile de montare și demontare pentru fișiere MS Windows la distanță
wget	V:181, I:982	3784	HTTP și FTP	program de descărcare din rețea
curl	V:260, I:727	520	, ,	, ,
transmission-gtk	V:9, I:143	6259	BitTorrent	client BitTorrent (GTK)
transmission-qt	V:0.6, I:2.6	6213	, ,	client BitTorrent (Qt)
ktorrent	V:1.3, I:5.0	5031	, ,	client BitTorrent (Qt)
qbittorrent	V:8, I:23	15378	, ,	client BitTorrent (Qt)
bind9-host	V:124, I:940	136	DNS	host(1)din bind9, ”Priority: standard”
dnsutils	V:5, I:146	23	, ,	dig(1) din bind, ”Priority: standard”
ldap-utils	V:11, I:57	790	LDAP	obține date de la serverul LDAP

Tabela 6.13: Lista clienților de aplicații de rețea

RFC	descriere
rfc1939 și rfc2449	POP3 service
rfc3501	serviciul IMAP4
rfc2821 (rfc821)	serviciul SMTP
rfc2822 (rfc822)	formatul fișierului de poștă electronică
rfc2045	Multipurpose Internet Mail Extensions (MIME)
rfc819	serviciul DNS
rfc2616	serviciul HTTP
rfc2396	definiție URI

Tabela 6.14: Lista RFC-urilor populare

Capitolul 7

Sistemul de interfață grafică

7.1 Mediul de birou cu interfață grafică

Există mai multe opțiuni pentru mediul grafic de birou în sistemul Debian.

pachetul de sarcini	popcon(popularitate)	dimensiune	descriere
task-gnome-desktop	I:192	9	mediul grafic de birou GNOME
task-xfce-desktop	I:88	9	mediul grafic de birou Xfce
task-kde-desktop	I:94	6	mediul grafic de birou KDE Plasma
task-mate-desktop	I:30	9	mediul grafic de birou MATE
task-cinnamon-desktop	I:35	9	mediul grafic de birou Cinnamon
task-lxde-desktop	I:19	9	mediul grafic de birou LXDE
task-lxqt-desktop	I:16	9	mediul grafic de birou LXQt
task-gnome-flashback-desktop	I:9.6	6	mediul grafic de birou GNOME Flashback

Tabela 7.1: Lista mediilor grafice de birou

Indicație
Pachetele de dependențe selectate de un metapachet de sarcini pot fi nesincronizate cu cea mai recentă stare de tranziție a pachetului în mediul Debian `unstable`/`testing`. Pentru `task-gnome-desktop`, poate fi necesar să ajustați selecțiile de pachete după cum urmează:

- Porniți aptitude(8) ca `sudo aptitude -u`.
- Deplasați cursorul la „Sarcini” și apăsați tasta «Enter».
- Deplasați cursorul la „Utilizator” și apăsați tasta «Enter».
- Deplasați cursorul la „GNOME” și apăsați tasta «Enter».
- Deplasați cursorul la `task-gnome-desktop` și apăsați tasta «Enter».
- Deplasați cursorul la „Depinde” și apăsați tasta «m» (selectat manual).
- Deplasați cursorul la „Recomandă” și apăsați tasta «m» (selectat manual).
- Deplasați cursorul la `task-gnome-desktop` și apăsați tasta «-». (abandonare)
- Ajustați pachetele selectate, eliminând cele problematice care provoacă conflicte între pachete.
- Apăsați tasta «g» pentru a începe instalarea.

Acest capitol se va concentra în principal pe mediul grafic de birou implicit al Debian: `task-gnome-desktop` care oferă GNOME pe wayland.

7.2 Protocolul de comunicare al interfeței grafice

Protocolul de comunicare al interfeței grafice utilizat în mediul grafic de birou GNOME poate fi:

- Wayland (protocol server de afișare) (nativ)
- protocolul central al sistemului X Window (via `xwayland`)

Vă rugăm să consultați situl freedesktop.org pentru a afla în ce fel arhitectura Wayland diferă de arhitectura X Window.

Din perspectiva utilizatorului, diferențele pot fi rezumate în mod colocvial astfel:

- Wayland este un protocol de comunicare cu interfața grafică pe aceeași gazdă: nou, mai simplu, mai rapid, fără binar setuid root
- X Window este un protocol de comunicare cu interfața grafică cu capacitate de rețea: tradițional, complex, mai lent, binar setuid root

Pentru aplicațiile care utilizează protocolul Wayland, accesul la conținutul afișat pe ecranul acestora de la o gazdă la distanță este acceptat de VNC sau RDP. Consultați Secțiune 7.8

Serverele X moderne dispun de MIT Shared Memory Extension și comunică cu clienții X locali utilizând memoria partajată locală. Acest lucru ocolește canalul de comunicare interproces Xlib transparent pentru rețea și îmbunătățește performanța. Această situație a stat la baza creării Wayland ca protocol de comunicare cu interfața grafică de utilizator (GUI) numai local.

Folosind programul `xeyes` pornit din terminalul GNOME, puteți verifica protocolul de comunicare GUI utilizat de fiecare aplicație GUI.

```
$ xeyes
```

- Dacă cursorul mouse-ului se află pe o aplicație precum „GNOME terminal” care utilizează protocolul serverului de afișare Wayland, ochii nu se mișcă odată cu cursorul mouse-ului.
- Dacă cursorul mouse-ului se află pe o aplicație precum „xterm”, care utilizează protocolul de bază al sistemului X Window, ochii se mișcă odată cu cursorul mouse-ului, dezvăluind natura nu tocmai izolată a arhitecturii X Window.

Începând cu aprilie 2021, multe aplicații GUI populare, precum GNOME și LibreOffice (LO), au fost migrate la protocolul serverului de afișare Wayland. Văd că `xterm`, `gitk`, `chromium`, `firefox`, `gimp`, `dia` și aplicațiile KDE încă utilizează protocolul central al sistemului X Window.

Notă
Atât pentru xwayland pe Wayland, cât și pentru sistemul nativ X Window, vechiul fișier de configurare a serverului X „`/etc/X11/xorg.conf`” nu ar trebui să existe în sistem. Plăcile grafice și dispozitivele de intrare sunt acum configurate de către nucleu cu DRM, KMS și udev. Serverul X nativ a fost rescris pentru a le utiliza. Consultați „suportul modului video implicit modedb” în documentația nucleului Linux.

7.3 Infrastructura de interfață grafică

Iată câteva pachete importante de infrastructură de interfață grafică pentru mediul grafic de birou GNOME în Wayland.

pachet	popcon(popularitate)	dimensiunea pachetului	descriere
`mutter`	V:1, I:22	216	Administratorul de ferestre mutter al GNOME **[auto]**
`xwayland`	V:256, I:335	2652	Un server X care rulează pe wayland **[auto]**
`gnome-remote-desktop`	V:147, I:241	2743	Demon de mediu de birou la distanță pentru GNOME folosind PipeWire **[auto]**
`gnome-tweaks`	V:17, I:232	1145	Opțiuni avansate de configurare pentru GNOME
`gnome-shell-extension-prefs`	V:8, I:117	73	Instrument pentru activarea/dezactivarea extensiilor GNOME Shell

Tabela 7.2: Lista pachetelor importante de infrastructură de interfață grafică

Aici, „**[auto]**” înseamnă că aceste pachete sunt instalate automat atunci când este instalat pachetul `task-gnome-desktop`.

Indicație
`gnome-tweaks` este instrumentul de configurare indispensabil. De exemplu:

- Puteți forța „supra-amplificarea” volumului sunetului din „General”.
- Puteți forța „Caps” să devină „Esc” din «Tastatură și mouse» -> «Tastatură» -> «Opțiuni suplimentare de configurare».

Indicație
Caracteristicile detaliate ale mediului grafic de birou GNOME pot fi configurate cu ajutorul instrumentelor pornite prin tastarea „`settings`”, „`tweaks`” sau „`extensions`” după apăsarea tastei `Super`.

7.4 Aplicații cu interfață grafică

Multe aplicații utile cu interfață grafică sunt disponibile acum în Debian. Instalarea pachetelor software precum `scribus` (KDE) în mediul grafic de birou GNOME este acceptabilă, deoarece funcționalitatea corespunzătoare nu este disponibilă în mediul grafic de birou GNOME. Însă instalarea unui număr prea mare de pachete cu funcționalități duplicate poate aglomera sistemul.

Iată o listă cu aplicațiile cu interfață grafică care mi-au atras atenția.

7.5 Directoarele utilizatorilor

Numele implicite pentru directoarele utilizatorilor, cum ar fi „`~/Desktop`”, „`~/Documents`”, ..., utilizate de mediul grafic de birou, depind de configurația regională utilizată pentru instalarea sistemului. Le puteți reinițializa la cele în limba engleză astfel:

```
$ LANGUAGE=C xdg-user-dirs-update --force
```

Apoi mutați manual toate datele în directoarele mai noi. Consultați xdg-user-dirs-update(1).

De asemenea, le puteți atribui orice nume dorit, editând „`~/.config/user-dirs.dirs`”. Consultați user-dirs.dirs(5).

7.6 Tipuri de litere

Utilizatorii Debian au la dispoziție numeroase fonturi scalabile utile. Preocuparea utilizatorilor este cum să evite redundanța și cum să configureze dezactivarea anumitor fonturi instalate. În caz contrar, fonturile inutile pot aglomera meniurile aplicațiilor cu interfață grafică.

Sistemul Debian utilizează biblioteca FreeType 2.0 pentru a reproduce multe formate de fonturi scalabile pentru ecran și imprimare:

- Fonturi de tip 1 (PostScript) care utilizează curbe Bézier cubice (format aproape învechit)
- Fonturi TrueType care utilizează curbe Bézier cuadratice (format recomandat)
- Fonturi OpenType care utilizează curbe Bézier cubice (cel mai bun format)

7.6.1 Fonturi de bază

Tabelul următor a fost întocmit în speranța de a ajuta utilizatorii să aleagă fonturi scalabile adecvate, cu o înțelegere clară a compatibilității metrice și a acoperirii glifelor. Majoritatea fonturilor acoperă toate caracterele latine, grecești și chirilice. Alegerea finală a fonturilor activate poate fi influențată și de preferințele estetice. Aceste fonturi pot fi utilizate pentru afișarea pe ecran sau pentru imprimarea pe hârtie.

Aici:

- „MCM” înseamnă „compatibil metric cu fonturile furnizate de Microsoft”
- „MCMATC” înseamnă „compatibil metric cu fonturile furnizate de Microsoft: Arial, Times New Roman, Courier New”
- „MCAHTC” înseamnă „metric compatibil cu fonturile furnizate de Adobe: Helvetica, Times, Courier”
- Numerele din coloanele tipului de font reprezintă lățimea relativă aproximativă „M” pentru fontul cu aceeași dimensiune a punctului.

pachet	popcon(popularitate)	dimensiunea pachetului	tipul	descriere
evolution	V:27, I:229	489	GNOME	gestionarea informațiilor personale (software de lucru în grup și poșta electronică)
thunderbird	V:39, I:102	303329	GTK	client de poștă electronică (Mozilla Thunderbird)
kontact	V:1, I:10	2360	KDE	gestionarea informațiilor personale (software de lucru în grup și poșta electronică)
libreoffice-writer	V:105, I:422	34232	LO	procesor de text
abiword	V:0.8, I:4.7	3596	GNOME	procesor de text
calligrawords	V:0.2, I:2.7	7144	KDE	procesor de text
scribus	V:1, I:12	32942	KDE	editor de tehnoredactare computerizată pentru editarea fișierelor PDF
glabels	V:0.3, I:2.5	1283	GNOME	editor de etichete
libreoffice-calc	V:101, I:419	29620	LO	foaie de calcul
gnumeric	V:3, I:10	9945	GNOME	foaie de calcul
calligrasheets	V:0.1, I:1.7	14014	KDE	foaie de calcul
libreoffice-impress	V:88, I:417	2499	LO	prezentare
calligrastage	V:0.1, I:1.6	6237	KDE	prezentare
libreoffice-base	V:21, I:67	5130	LO	gestionare a bazei de date
kexi	V:0.05, I:0.80	7565	KDE	gestionare a bazei de date
libreoffice-draw	V:89, I:418	11418	LO	editor de grafică vectorială (desen)
inkscape	V:12, I:76	113353	GNOME	editor de grafică vectorială (desen)
karbon	V:0.1, I:2.1	4029	KDE	editor de grafică vectorială (desen)
dia	V:1, I:16	3846	GTK	editor de diagrame și organigrame
gimp	V:34, I:211	33315	GTK	editor grafic (pictură)
shotwell	V:14, I:246	6334	GTK	organizator de fotografii digitale
digikam	V:1.5, I:8.3	333	KDE	organizator de fotografii digitale
darktable	V:4, I:11	36092	GTK	masă de lumină și cameră obscură pentru fotografi
gnome-video-trimmer	V:0.1, I:1.0	1665	GNOME	tăierea fără pierderi a fișierelor video
kdenlive	V:3, I:21	22134	KDE	editor video neliniar
shotcut	V:1.0, I:7.5	8270	aplicație Qt	editor video neliniar
openshot-qt	V:0.9, I:8.8	196007	aplicație Qt	editor video neliniar
flowblade	V:0.2, I:2.2	42213	GTK	editor video neliniar
planner	V:0.2, I:4.2	1400	GNOME	gestionarea proiectelor
calligraplan	V:0.2, I:2.9	23545	KDE	gestionarea proiectelor
gnucash	V:2.0, I:6.7	29434	GNOME	contabilitate personală
homebank	V:0.3, I:1.5	3194	GTK	contabilitate personală
kmymoney	V:0.4, I:1.9	19297	KDE	contabilitate personală
frescobaldi	V:0.3, I:1.5	11102	aplicație Qt	editorul de text pentru partituri LilyPond
librecad	V:1, I:13	9164	aplicație Qt	sistem de proiectare asistată de calculator (CAD) (2D)
freecad	V:1, I:20	112	aplicație Qt	sistem de proiectare asistată de calculator (CAD) (3D)
kicad	V:3, I:14	224155	GTK	software pentru proiectarea schemelor electronice și a plăcilor cu circuite imprimate
xsane	V:9, I:128	1512	GTK	interfață scaner
libreoffice-math	V:84, I:420	1982	LO	editor de ecuații/formule matematice
calibre	V:7, I:24	71216	KDE	convertor de cărți electronice și gestionarea bibliotecii
fbreader	V:0.7, I:5.4	3827	GTK	cititor de cărți electronice
evince	V:76, I:284	953	GNOME	program de vizualizare a documentelor(pdf)
okular	V:42, I:130	4544	KDE	program de vizualizare a documentelor(pdf)
			aplicație X	

pachet	popcon(popularitate)	dimensiune	sans	serif	mono	informații despre font
fonts-cantarell	V:174, I:288	224	59	-	-	Cantarell (GNOME 3, afișare)
fonts-noto	I:150	30	61	63	40	Fonturi Noto (Google, multi-lingvistice cu CJK)
fonts-dejavu	I:385	34	58	68	40	DejaVu (GNOME 2, MCM:Verdana, extins Bitstream Vera)
fonts-liberation2	V:40, I:156	15	56	60	40	Fonturi Liberation pentru LibreOffice (Red Hat, MCMATC)
fonts-croscore	V:20, I:36	5260	56	60	40	Chrome OS: Arimo, Tinos și Cousine (Google, MCMATC)
fonts-crosextra-carlito	V:20, I:90	2696	57	-	-	Chrome OS: Carlito (Google, MCM:Calibri)
fonts-crosextra-caladea	V:12, I:83	347	-	55	-	Chrome OS: Caladea (Google, MCM:Cambria) (doar Latin)
fonts-freefont-ttf	V:78, I:203	14460	57	59	40	GNU FreeFont (URW Nimbus extins)
fonts-quicksand	V:210, I:452	392	56	-	-	Debian task-desktop, Quicksand (afișare, doar Latin)
fonts-hack	V:34, I:136	2509	-	-	40 P	Un tip de font conceput pentru codul sursă Hack (Facebook)
fonts-sil-gentiumplus	V:0, I:28	14345	-	54	-	Gentium SIL
fonts-sil-charis	V:1, I:28	6704	-	59	-	Charis SIL
fonts-urw-base35	V:187, I:534	15558	56	60	40	URW Nimbus (Nimbus Sans, Roman No. 9 L, Mono L, MCAHTC)
fonts-ubuntu	V:2.2, I:4.7	4339	58	-	33 P	Fonturi Ubuntu (afișare)
fonts-terminus	V:0.3, I:3.9	451	-	-	33	Cool retro terminal fonts
ttf-mscorefonts-installer	V:1, I:40	85	56?	60	40	Descărcător de fonturi Microsoft care nu sunt libere (vedeți mai jos)

Tabela 7.4: Lista fonturilor notabile TrueType și OpenType

- Litera „P” din coloanele cu font mono indică utilitatea acestuia pentru programare, având caractere „0”/„O” și „1”/„l”/„I” clar distincte.
- Pachetul `ttf-mscorefonts-installer` descarcă „Core fonts for the Web” și instalează Arial, Times New Roman, Courier New, Verdana, ... Aceste date de fonturi instalate nu sunt date libere.

Multe fonturi latine libere își au originea în familia URW Nimbus sau Bitstream Vera.

Indicație
Dacă configurația dvs. regională necesită fonturi care nu sunt acoperite corespunzător de fonturile de mai sus, vă rugăm să utilizați aptitude pentru a verifica pachetele de sarcini listate la „Sarcini” -> „Localizare”. Pachetele de fonturi listate ca „Depinde de:” sau „Recomandă:” în pachetele de sarcini de regionalizare sunt principalele candidate.

7.6.2 Conversia și redarea fonturilor

Debian utilizează FreeType pentru conversia și redarea fonturilor. Infrastructura sa de selectare a fonturilor este furnizată de biblioteca de configurare a fonturilor Fontconfig.

pachet	popcon(popularitate) și descriere		
`libfreetype6`	V:589, I:997	1050	FreeType bibliotecă de conversie și redare a fonturilor
`libfontconfig1`	V:577, I:808	344	Fontconfig bibliotecă de configurare a fonturilor
`fontconfig`	V:471, I:693	415	`fc-*`: comenzi în linia de comandă pentru Fontconfig
`font-manager`	V:2.3, I:6.6	1116	Font Manager: comenzi din interfață grafică pentru Fontconfig
`nautilus-font-manager`	V:0.7, I:0.38	38	Extensie Nautilus pentru Font Manager

Tabela 7.5: Lista mediilor de fonturi notabile și a pachetelor asociate

Indicație
Unele pachete de fonturi, cum ar fi `fonts-noto*`, instalează prea multe fonturi. Este posibil să doriți să păstrați unele pachete de fonturi instalate, dar dezactivate în condiții normale de utilizare. Mai multe glyphs sunt așteptate pentru unele puncte de cod Unicode din cauza unificării Han, iar glyphs nedorite pot fi alese de biblioteca Fontconfig neconfigurată. Unul dintre cele mai enervante cazuri este „U+3001 IDEOGRAPHIC COMMA” și „U+3002 IDEOGRAPHIC FULL STOP” în țările CJK. Puteți evita cu ușurință această situație problematică configurând disponibilitatea fonturilor folosind interfața grafică Font Manager (font-manager).

De asemenea, puteți afișa starea configurației fonturilor din linia de comandă.

- ”`fc-match(1)`” pentru fontul implicit al fontconfig
- ”`fc-list(1)`” pentru fonturile fontconfig disponibile

Puteți configura starea configurației fonturilor din editorul de text, dar acest lucru nu este trivial. Consultați fonts.conf(5).

7.7 Cutia cu nisip (sandbox)

Multe aplicații cu interfață grafică pentru Linux sunt disponibile în format binar din surse non-Debian.

- AppImage -- Aplicații Linux care rulează oriunde

- FLATHUB -- Aplicații pentru Linux, chiar aici
- snapcraft -- Magazinul de aplicații pentru Linux



Avertisment
Fișierele binare de pe aceste situri pot include pachete software proprietare care nu sunt libere.

Există o anumită rațiune de a fi pentru aceste distribuții în format binar pentru pasionații de software liber care utilizează Debian, deoarece acestea pot acomoda un set curat de biblioteci utilizate pentru fiecare aplicație de către dezvoltatorul din amonte, independent de cele furnizate de Debian.

Riscul inerent asociat rulării fișierelor binare externe poate fi redus prin utilizarea mediului sandbox, care exploatează caracteristicile moderne de securitate ale Linux (consultați Secțiune 4.7.5).

- Pentru fișierele binare din AppImage și unele situri upstream, rulați-le în firejail cu configurație manuală.
- Pentru fișierele binare din FLATHUB, rulați-le în Flatpak ; (nu este necesară configurarea manuală).
- Pentru fișierele binare din snapcraft, rulați-le în Snap ; (nu este necesară configurarea manuală. Compatibil cu programele demon).

Pachetul `xdg-desktop-portal` oferă o API standardizată pentru funcțiile comune ale mediului de birou. Consultați xdg-desktop-portal (flatpak) și xdg-desktop-portal (snap) .

pachet	popcon(popularitate) și dimensiune	descriere
`flatpak`	V:104, I:115	9293	Flatpak cadru de implementare a aplicațiilor pentru aplicații de birou
`gnome-software-plugin-flatpak`	V:29, I:41	291	Suport Flatpak pentru software-ul GNOME
`snapd`	V:58, I:62	62738	Demon și instrumente care abilitează pachetele snap
`gnome-software-plugin-snap`	V:1.4, I:2.2	144	Suport Snap pentru software-ul GNOME
`xdg-desktop-portal`	V:373, I:439	2291	portal de integrare a mediului grafic de birou pentru Flatpak și Snap
`xdg-desktop-portal-gtk`	V:337, I:438	715	serviciul xdg-desktop-portal pentru gtk (GNOME)
`xdg-desktop-portal-kde`	V:81, I:112	3333	serviciul xdg-desktop-portal pentru Qt (KDE)
`xdg-desktop-portal-wlr`	V:2.4, I:6.6	159	serviciul xdg-desktop-portal pentru wlroots (Wayland)
`firejail`	V:1.2, I:4.4	1827	un program sandbox de securitate SUID firejail pentru utilizare cu AppImage

Tabela 7.6: Lista mediilor sandbox notabile și a pachetelor asociate

Această tehnologie de mediu sandbox este foarte asemănătoare cu aplicațiile de pe sistemele de operare ale telefoanelor inteligente, unde aplicațiile sunt executate sub acces controlat la resurse.

Unele aplicații cu interfață grafică de mari dimensiuni, precum navigatoarele web din Debian, utilizează și ele tehnologia mediului sandbox la nivel intern pentru a le face mai sigure.

pachet	popcon(popularitate)	dimensiune	protocoale	descriere
gnome-remote-desktop	V:14, I:241	2743	RDP	serverul GNOME Remote Desktop
xrdp	V:28, I:31	4757	RDP	xrdp, serverul Remote Desktop Protocol (RDP)
x11vnc	V:9, I:46	1863	RFB (VNC)	x11vnc, serverul Remote Framebuffer Protocol (VNC)
tigervnc-standalone-server	V:5, I:15	3124	RFB (VNC)	TigerVNC, serverul Remote Framebuffer Protocol (VNC)
gnome-connections	V:6, I:149	1726	RDP, RFB (VNC)	client GNOME remote desktop
vinagre	V:1, I:21	4249	RDP, RFB (VNC), SPICE, SSH	Vinagre: client GNOME remote desktop
remmina	V:16, I:59	983	RDP, RFB (VNC), SPICE, SSH, ...	Remmina: client de mediu grafic de birou la distanță GTK
krdc	V:2, I:15	4144	RDP, RFB (VNC)	KRDC: client de Birou la distanță KDE
virt-viewer	V:4, I:37	1278	RFB (VNC), SPICE	clientul de afișare grafică al Virtual Machine Manager pentru sistemul de operare invitat

Tabela 7.7: Lista serverelor notabile de acces la distanță

7.8 Mediu de birou la distanță

7.9 Conexiune la serverul X

Există mai multe modalități de conectare de la o aplicație de pe o gazdă la distanță la serverul X, inclusiv `xwayland` din gazda locală.

pachet	popcon(popularitate)	dimensiune	comandă	descriere
openssh-server	V:774, I:820	3615	`sshd` cu opțiunea `X11-forwarding`	server SSH (securizat)
openssh-client	V:682, I:997	3611	`ssh -X`	client SSH (securizat)
xauth	V:192, I:974	81	`xauth`	utilitarul de fișier de autoritate X
x11-xserver-utils	V:373, I:533	559	`xhost`	controlul accesului la server pentru X

Tabela 7.8: Lista metodelor de conectare la serverul X

7.9.1 Conexiune locală la serverul X

Accesul la serverul X local de către aplicațiile locale care utilizează protocolul X core poate fi conectat local printr-un soclu de domeniu UNIX local. Acest lucru poate fi autorizat de fișierul de autoritate care conține cookie-ul de acces. Locația fișierului de autoritate este identificată de variabila de mediu „`$XAUTHORITY`”, iar afișajul X este identificat

de variabila de mediu „$DISPLAY”. Deoarece acestea sunt definite în mod normal automat, nu este necesară nicio acțiune specială, de exemplu „gitk” după cum urmează.

```
username $ gitk
```

Notă
Pentru xwayland, XAUTHORITY conține valori precum „/run/user/1000/.mutter-Xwaylandauth.YVSU30”.

7.9.2 Conexiune de la distanță la serverul X

Accesul la afișajul serverului X local din aplicațiile la distanță care utilizează protocolul central X este acceptat prin utilizarea funcției de redirecționare X11.

- Deschideți un gnome-terminal pe gazda locală.
- Rulați ssh(1) cu opțiunea -X pentru a stabili o conexiune cu situl la distanță, după cum urmează.

```
localname @ localhost $ ssh -q -X loginname@remotehost.domain
Password:
```

- Rulați o comandă a aplicației X, de exemplu „gitk”, pe situl la distanță, după cum urmează.

```
loginname @ remotehost $ gitk
```

Această metodă poate afișa ieșirea de la un client X la distanță ca și cum ar fi conectat local printr-un soclu de domeniu UNIX local.

Consultați Secțiune 6.3 pentru SSH/SSHD.

Avertisment
O conexiune la distanță TCP/IP la serverul X este dezactivată în mod implicit în sistemul Debian din motive de securitate. Nu o activați prin simpla configurare a „xhost +” și nici prin activarea conexiunii XDMCP, dacă puteți evita acest lucru.

7.9.3 Conexiune chroot la serverul X

Accesul la serverul X de către aplicațiile care utilizează protocolul central X și rulează pe aceeași gazdă, dar într-un mediu precum chroot, unde fișierul de autoritate nu este accesibil, poate fi autorizat în mod securizat cu xhost utilizând Accesul bazat pe utilizator, de exemplu „gitk”, după cum urmează.

```
username $ xhost + si:localuser:root ; sudo chroot /path/to
# cd /src
# gitk
# exit
username $ xhost -
```

7.10 Clipboard

Pentru a copia textul în clipboard, consultați Secțiune 1.4.4.

Pentru a copia elemente grafice în clipboard, consultați Secțiune 11.6.

Unele comenzi CLI pot manipula de asemenea clipboardul de caractere (PRIMARY și CLIPBOARD).

pachet	popcon(popularitate)	dimensiunea pachetului	ținta	descriere
xsel	V:8, I:42	55	X	interfață de linie de comandă pentru selecții X (clipboard)
xclip	V:18, I:78	62	X	interfață de linie de comandă pentru selecții X (clipboard)
wl-clipboard	V:10, I:27	174	Wayland	`wl-copy` `wl-paste`: interfață de linie de comandă pentru clipboardul Wayland
gpm	V:8.3, I:9.0	524	Consola Linux	un demon care capturează evenimentele mouse-ului din consola Linux

Tabela 7.9: Lista programelor legate de manipularea clipboardului de caractere

Capitolul 8

I18N și L10N

Multilingvismul (M17N) sau suportul pentru limba maternă pentru un software de aplicație se realizează în 2 pași.

- Internaționalizare (I18N): Pentru a face un software capabil să gestioneze mai multe limbi.
- Localizare (L10N): Pentru a face un software să gestioneze o anumită configurație regională.

Indicație
Există 17, 18 sau 10 litere între „m” și „n”, „i” și „n” sau „l” și „n” în multilingvism, internaționalizare și localizare, care corespund M17N, I18N și L10N. Pentru detalii, consultați Internaționalizare și localizare.

8.1 Configurația regională

Comportamentul programelor care acceptă internaționalizarea este configurat de variabila de mediu „`$LANG`” pentru a accepta localizarea(regionalizarea). Suportul efectiv al caracteristicilor dependente de configurația regională prin biblioteca `libc`, necesită instalarea pachetelor `locales` sau `locales-all`. Pachetul `locales` trebuie să fie inițializat adecvat.

Dacă nici pachetul `locales` și nici pachetul `locales-all` nu sunt instalate, suportul pentru caracteristicile regionle se pierde, iar sistemul utilizează mesaje în limba engleză americană și tratează datele ca **ASCII**. Acest comportament este identic cu cel al definirii variabilei „`$LANG`” prin „`LANG=`”, „`LANG=C`” sau „`LANG=POSIX`”.

Software-ul modern, precum GNOME și KDE, este multilingv. Acesta este internaționalizat prin gestionarea datelor UTF-8 și regionalizat prin furnizarea mesajelor traduse prin infrastructura gettext(1). Mesajele traduse pot fi furnizate ca pachete de limbă separate.

Sistemul de interfață grafică actual al mediului de birou Debian definește în mod normal configurația regională în mediul de interfață grafică ca „`LANG=xx_YY.UTF-8`”. Aici, „`xx`” este codul limbii ISO 639, iar „YY” este codul țării ISO 3166. Aceste valori sunt definite în dialogul grafic de configurare a biroului și modifică comportamentul programului. Vedeți Secțiune 1.5.2

8.1.1 Justificarea pentru utilizarea UTF-8 în configurația lingvistică

Cea mai simplă reprezentare a datelor textuale este **ASCII**, care este suficientă pentru limba engleză și utilizează mai puțin de 127 de caractere (reprezentabile cu 7 biți).

Chiar și textul în limba engleză simplă poate conține caractere non-ASCII, de exemplu ghilimelele ușor curbate la stânga și la dreapta nu sunt disponibile în ASCII.

```
b'‘b'double quoted textb'’b'' is not "double quoted ASCII"
b'‘b'single quoted textb'’b'' is not 'single quoted ASCII'
```

Pentru a oferi suport pentru mai multe caractere, au fost utilizate numeroase seturi de caractere și sisteme de codificare pentru a oferi suport pentru mai multe limbi (a se vedea Tabel 11.2).

Setul de caractere Unicode poate reprezenta practic toate caracterele cunoscute de om cu un interval de coduri de 21 de biți (adică de la 0 la 10FFFF în notație hexazecimală).

Sistemul de codificare a textului UTF-8 încadrează punctele de cod Unicode într-un flux de date sensibil de 8 biți, compatibil în mare parte cu sistemul de procesare a datelor ASCII. Acest lucru face ca **UTF-8** să fie alegerea preferată în prezent. **UTF** înseamnă Unicode Transformation Format (Format de transformare Unicode). Când datele text simplu ASCII sunt convertite în UTF-8, acestea au exact același conținut și aceeași dimensiune ca și cele ASCII originale. Așadar, nu pierdeți nimic prin implementarea parametrului lingvistic de codificare a textului UTF-8.

În cadrul configurației lingvistice UTF-8 compatibilă cu programul aplicației, puteți afișa și edita orice date text în limbi străine, atâta timp cât fonturile și metodele de introducere necesare sunt instalate și activate. De exemplu, în cadrul configurației regionale „`LANG=fr_FR.UTF-8`”, gedit(1) (editor de text pentru mediul grafic de birou GNOME) poate afișa și edita date text cu caractere chinezești, prezentând în același timp meniurile în limba franceză.

Indicație
Atât noul standard „`en_US.UTF-8`” cât și vechiul standard „`C`”/„`POSIX`” utilizează mesajul standard în limba engleză americană, dar prezintă diferențe subtile în ceea ce privește ordinea de sortare etc. Dacă doriți să gestionați nu numai caracterele ASCII, ci și toate caracterele codificate UTF-8, păstrând în același timp comportamentul local vechi „C”, utilizați configurația regională non-standard „`C.UTF-8`” în Debian.

Notă
Unele programe consumă mai multă memorie după ce acceptă I18N. Acest lucru se datorează faptului că sunt codificate pentru a utiliza UTF-32(UCS4) intern pentru a accepta Unicode în scopul optimizării vitezei și consumă 4 octeți pentru fiecare caracter ASCII, indiferent de configurația regională selectată. Din nou, nu pierdeți nimic prin implementarea configurației regionale UTF-8.

8.1.2 Reconfigurarea configurației regionale

Pentru ca sistemul să poată accesa o anumită configurație regională, datele configurației regionale trebuie compilate din baza de date a configurațiilor regionale.

Pachetul `locales` **nu** este livrat cu date de configurație regională precompilate. Trebuie să îl configurați astfel:

```
# dpkg-reconfigure locales
```

Acest proces implică 2 pași.

1. Selectați toate datele de configurație regională necesare pentru a fi compilate în format binar; (asigurați-vă că includeți cel puțin o configurație regională UTF-8).
2. Stabiliți valoarea configurației lingvistice implicite la nivel de sistem prin crearea fișierului „`/etc/default/locale`” pentru utilizarea de către PAM (consultați Secțiune 4.5).

Valoarea implicită a configurației lingvistice la nivel de sistem, definită în „`/etc/default/locale`”, poate fi suprascrisă de configurația grafică pentru aplicațiile cu interfață grafică.

Notă
Sistemul tradițional de codificare actual poate fi identificat prin „`/usr/share/i18n/SUPPORTED`”. Astfel, „`LANG=en_US`” este „`LANG=en_US.ISO-8859-1`”.

Pachetul `locales-all` vine cu date de configurație regională precompilate pentru toate datele de configurație regională. Deoarece nu creează „`/etc/default/locale`”, este posibil să fie necesar să instalați și pachetul `locales`.

Indicație
Pachetul `locales` al unor distribuții derivate din Debian vine cu date de configurație regională precompilate pentru toate datele locale. Trebuie să instalați atât pachetul `locales`, cât și pachetul `locales-all` pe Debian pentru a emula un astfel de mediu de sistem.

8.1.3 Codificarea numelor de fișiere

Pentru schimburile de date între platforme (a se vedea Secțiune 10.1.7), poate fi necesar să montați unele sisteme de fișiere cu codificări speciale. De exemplu, mount(8) pentru sistemul de fișiere vfat presupune CP437 dacă este utilizat fără opțiune. Trebuie să furnizați o opțiune de montare explicită pentru a utiliza UTF-8 sau CP932 pentru numele fișierelor.

Notă
Când se montează automat o unitate flash USB conectabilă la cald într-un mediu grafic de birou modern, cum ar fi GNOME, puteți furniza această opțiune de montare făcând clic dreapta pe pictograma de pe birou, făcând clic pe fila «Unitate», făcând clic pentru a extinde «Configurări» și introducând „utf8” în «Opțiuni de montare:». Data viitoare când acest stick de memorie va fi montat, montarea cu UTF-8 va fi activată.

Notă
Dacă actualizați sistemul sau mutați unitățile de disc de pe un sistem mai vechi care nu utilizează UTF-8, numele fișierelor cu caractere non-ASCII pot fi codificate în codificări istorice și învechite, cum ar fi ISO-8859-1 sau eucJP. Vă rugăm să solicitați ajutorul instrumentelor de conversie a textului pentru a le converti în UTF-8. Consultați Secțiune 11.1.

Samba utilizează Unicode pentru clienții mai noi (Windows NT, 200x, XP), dar utilizează CP850 pentru clienții mai vechi (DOS și Windows 9x/Me) ca valoare implicită. Această valoare implicită pentru clienții mai vechi poate fi modificată utilizând „`dos charset`” în fișierul „`/etc/samba/smb.conf`”, de exemplu, la CP932 pentru japoneză.

8.1.4 Mesaje în limba maternă și documentație tradusă

Există traduceri pentru multe dintre mesajele text și documentele afișate în sistemul Debian, cum ar fi mesajele de eroare, ieșirile standard ale programelor, meniurile și paginile de manual. Lanțul de instrumente GNU gettext(1) este utilizat ca instrument de bază pentru majoritatea activităților de traducere.

În secțiunea „Sarcini” → „Localizare” aptitude(8) oferă o listă extinsă de pachete binare utile care adaugă mesaje în limba maternă aplicațiilor și furnizează documentație tradusă.

De exemplu, puteți obține mesajul în limba maternă pentru pagina de manual instalând pachetul `manpages-`*`LANG`*. Pentru a citi pagina de manual în limba italiană pentru *programname* din „`/usr/share/man/it/`”, executați următoarea comandă.

```
LANG=it_IT.UTF-8 man programname
```

GNU gettext poate acomoda lista de priorități a limbilor de traducere cu variabila de mediu $LANGUAGE. De exemplu:

```
$ export LANGUAGE="pt:pt_BR:es:it:fr"
```

Pentru mai multe informații, consultați `info gettext` și citiți secțiunea „Variabila LANGUAGE”.

8.1.5 Efectele configurației regionale

Ordinea de sortare a caracterelor cu sort(1) și ls(1) este afectată de configurația regională. Exportarea `LANG=en_US.UTF-8` sortează în dicționarul `A->a->B->b... ->Z->z`, în timp ce exportarea `LANG=C.UTF-8` sortează în ordinea binară ASCII `A->B->...->Z->a->b...`.

Formatul datei ls(1) este influențat de configurația regională (a se vedea Secțiune 9.3.4).

Formatul datei date(1) este influențat de configurația regională. De exemplu:

```
 $ unset LC_ALL
 $ LANG=en_US.UTF-8 date
Thu Dec 24 08:30:00 PM JST 2023
 $ LANG=en_GB.UTF-8 date
Thu 24 Dec 20:30:10 JST 2023
 $ LANG=es_ES.UTF-8 date
jue 24 dic 2023 20:30:20 JST
 $ LC_TIME=en_DK.UTF-8 date
2023-12-24T20:30:30 JST
```

Punctuația numerelor diferă în funcție de configurația regională. De exemplu, în configurația regională engleză, o mie și unu se afișează ca „`1,000.1`”, în timp ce în configurația regională germană se afișează ca „`1.000,1`”. Puteți observa această diferență în programul de foi de calcul.

Fiecare caracteristică detaliată a variabilei de mediu „`$LANG`” poate fi suprascrisă prin definirea variabilelor „`$LC_*`”. Aceste variabile de mediu pot fi suprascrise din nou prin definirea variabilei „`$LC_ALL`”. Consultați pagina de manual locale(7) pentru detalii. Dacă nu aveți un motiv întemeiat pentru a crea o configurație complicată, vă rugăm să le evitați și să utilizați numai variabila „`$LANG`” definită la una dintre configurările lingvistice UTF-8.

8.2 Introducere de la tastatură

Tastatura sistemului poate fi configurată la diferite niveluri ale sistemului.

- Nucleul Linux: keyboard(5)
- X server: setxkbmap(1), xkeyboard-config(5), variabila de mediu `XMODIFIERS`
- Mediul grafic de birou: Cadrul pentru metode de introducere a datelor: `ibus`, `fcitx5`
- Aplicație: variabile de mediu pentru configurarea sursei de introducere: `GTK_IM_MODULE`, `QT_IM_MODULE`, `QT_IM_MODUL` ...

Cadrul metodei de introducere a datelor („Input method framework”: IM) cuprinde:

- Motorul metodei de introducere („Input method engine”: IME): Metoda de introducere actuală
- Configurare: Se ocupă de configurarea IBus și a altor servicii, precum modulele IME
- Panou: Interfață de utilizator, cum ar fi bara de limbă și tabelul de selecție a variantelor

Introducerea multilingvă în aplicație este procesată, în mare, după cum urmează:

```
Keyboard        UI panel    Configuration          Application
|                  ^ |           |                       ^  ^
v                  | v           v                       |  |
Linux kernel -> Input method engine (IME) -+-> Gtk, Qt ---+  |
                   | ^                     |                 |
                   | |                     +-> X11, Wayland -+
                   v |
                 IME plugin (ibus-mozc, ...)
```

8.2.1 Introducerea de la tastatură pentru consola Linux și X Window

Sistemul Debian poate fi configurat pentru a funcționa cu multe aranjamente internaționale ale tastaturii folosind pachetul `keyboard-configuration`.

```
# dpkg-reconfigure keyboard-configuration
```

Pentru consola Linux și sistemul X Window, acesta actualizează parametrii de configurare din „`/etc/default/keyboard`”. caractere non-ASCII, inclusiv caracterele accentuate utilizate de multe limbi europene, pot fi disponibile cu tasta moartă, tasta AltGr și tasta de compunere.

Notă
Dacă ibus este activ, configurația clasică a tastaturii X prin `setxkbmap` poate fi suprascrisă de `ibus` chiar și în mediul de birou clasic bazat pe X. Puteți dezactiva `ibus` instalat utilizând `im-config` pentru a stabili metoda de introducere la „Niciuna”. Pentru mai multe informații, consultați Debian Wiki despre tastatură.

8.2.2 Introducerea de la tastatură pentru Wayland

Spre deosebire de protocolul X Window, protocolul de bază Wayland nu acceptă nici măcar introducerea caracterelor accentuate. Compozitoarele populare pentru Wayland, precum Mutter pentru GNOME sau KWin pentru KDE, implementează protocoale de extensie precum `text-input-unstable-v3` pentru introducerea textului (vezi „protocoalele Wayland actuale și starea suportului acestora”).

Protocolul `text-input-unstable-v3` funcționează bine cu metodele de introducere a textului pentru Wayland (vezi „Analiza post-mortem a proiectului metodei de introducere a textului pentru Wayland – Wayland input method project post-mortem”).

- Majoritatea aplicațiilor cu interfață grafică sunt realizate cu ajutorul unor biblioteci GUI, precum GTK sau Qt, care oferă suport pentru acest `text-input-unstable-v3`.
- Motoarele populare de introducere a textului (IME), precum IBus sau Fcitx (versiunea 5), pot funcționa cu acest `text-input-unstable-v3`.
- IME-urile acceptă introducerea de text în numeroase limbi prin intermediul unor module.
- IME-urile recente integrează funcționalități ale X Keyboard Extension (XKB), precum `setxkbmap`, care erau furnizate anterior de X Window pentru a permite introducerea textului cu caractere accentuate în limbile europene în mediul Wayland.

8.2.3 Suportul pentru metoda de introducere cu IBus

Pentru GNOME, „`ibus`” este IME-ul implicit, care se instalează automat prin intermediul dependențelor pachetului său.

- Majoritatea funcțiilor de introducere a textului de la tastatură în mai multe limbi pot fi configurate din „Configurări GNOME” sau „Ajustări GNOME”.
- Unele caracteristici ale introducerii textului prin tastatură în mai multe limbi ar putea necesita configurarea prin comanda ibus-setup(1).
- Introducerea prin tastatură a emoji-urilor este disponibilă tastând „SUPER-.” (apăsați simultan tastele Windows și punct), urmată de un cuvânt-cheie pentru fiecare emoji și de tasta SPAȚIU.

Lista pachetelor IBus și a modulelor sale este următoarea.

pachet	popcon(popularitate) și dimensiune	dimensiune	configurația lingvistică acceptată
ibus	V:212, I:253	1884	infrastructura metodei de introducere a datelor utilizând dbus
ibus-mozc	V:2.1, I:3.8	1054	japoneză
ibus-anthy	V:0.5, I:1.1	8740	japoneză
ibus-skk	V:0.03, I:0.14	242	japoneză
ibus-kkc	V:0.04, I:0.19	206	japoneză
ibus-libpinyin	V:1.1, I:5.1	123	chineză (pentru zh_CN)
ibus-chewing	V:0.16, I:0.84	298	chineză (pentru zh_TW)
ibus-libzhuyin	I:0.10	41021	chineză (pentru zh_TW)
ibus-rime	V:0.29, I:0.51	76	chineză (pentru zh_CN/zh_TW)
ibus-cangjie	V:0.02, I:0.10	235	chineză (pentru zh_HK)
ibus-hangul	V:0.3, I:1.9	261	coreeană
ibus-libthai	V:0.00, I:0.05	84	thailandeză
ibus-table-thai	I:0.04	59	thailandeză
ibus-unikey	V:0.22, I:0.40	286	vietnameză
keyman	V:0.01, I:0.10	507	Multilingv: modulul Keyman pentru peste 2000 de limbi
ibus-table	V:0.04, I:0.92	2364	modul de tabele pentru IBus
ibus-m17n	V:0.1, I:1.8	373	multilingv: indic, arabă și altele

Tabela 8.1: Lista pachetelor IBus și a modulelor sale

8.2.4 Suportul pentru metoda de introducere cu Fcitx

Cadrul de introducere a textului Fcitx (versiunea 5) este popular în rândul utilizatorilor chinezi și este compatibil cu „`ibus`”.

Lista pachetelor „`fcitx5`” și a modulelor său este următoarea.

pachet	popcon(popularitate) și dimensiune	dimensiune	configurația lingvistică acceptată
fcitx5	V:8, I:13	712	cadru pentru metode de introducere a textului compatibil cu „`ibus`”
fcitx5-mozc	V:1.1, I:1.8	1328	japoneză
fcitx5-anthy	V:0.07, I:0.25	853	japoneză
fcitx5-skk	V:0.05, I:0.15	370	japoneză
fcitx5-kkc	V:0.01, I:0.06	418	japoneză
fcitx5-chinese-addons	I:9.2	17	chineză (metapachet pentru zh_*)
fcitx5-pinyin	V:4.0, I:9.7	1000	chineză (pentru zh_CN)
fcitx5-chewing	V:0.29, I:0.98	228	chineză (pentru zh_TW)
fcitx5-zhuyin	I:0.08	108535	chineză (pentru zh_TW)
fcitx5-rime	V:0.46, I:0.93	393	chineză (pentru zh_CN/zh_TW)
fcitx5-table-cangjie-large	I:0.13	1292	chineză (pentru zh_HK)
fcitx5-hangul	V:0.12, I:0.27	239	coreeană
fcitx5-libthai	V:0.00, I:0.04	119	thailandeză
fcitx5-table-thai	I:0.09	34	thailandeză
fcitx5-unikey	V:0.13, I:0.23	624	vietnameză
fcitx5-m17n	V:0.16, I:0.61	171	multilingv: indic, arabă și altele
fcitx5-table	V:0.3, I:9.6	428	modul de tabele pentru fcitx5
fcitx5-keyman	V:0.04, I:0.05	241	Multilingv: modulul Keyman pentru peste 2000 de limbi

Tabela 8.2: Lista Fcitx5 și a pachetelor sale de module

8.2.5 Un exemplu pentru japoneză

Consider că metoda de introducere a caracterelor japoneze pornită în mediul englez („`en_US.UTF-8`”) este foarte utilă. Iată cum am procedat cu IBus:

1. Instalați pachetul de instrumente de introducere a textului în limba japoneză `ibus-mozc` (sau `ibus-anthy`).
2. • Procedură generală: Executați comanda ibus-setup(1) → selectați „Metodă de introducere” → faceți clic pe „`Adăugare`” → „Japoneză” → „Mozc (sau Anthy)” → faceți clic pe „Adaugă”
   • GNOME: Selectați „Configurări” → „Tastatură” → „Surse de introducere” → faceți clic pe „+” în „Surse de introducere” → „Japoneză” → „Japoneză mozc (sau anthy)” și faceți clic pe „Adaugă”
3. Puteți alege câte surse de intrare doriți.

4. Conectați-vă din nou la contul dvs. de utilizator.
5. Configurați fiecare sursă de intrare făcând clic dreapta pe pictograma barei de instrumente din interfața grafică.
6. Comutați între sursele de intrare instalate cu ajutorul SUPER-SPACE; (SUPER este, de obicei, tasta Windows).

Indicație
Dacă doriți să aveți acces la un mediu de tastatură numai cu alfabetul, cu tastatura fizică japoneză pe care shift-2 are gravată " (ghilimele duble), selectați „Japoneză” în procedura de mai sus. Puteți introduce text în japoneză utilizând „Japanese mozc (sau anthy)” cu tastatura fizică „US” pe care shift-2 are gravat @ (semnul a-încercuitt).

- Pentru Wayland:
  - Pachetul `im-config` nu face nimic și poate fi eliminat fără probleme.
  - Probabil că nu este necesar să definiți variabile de mediu, cu excepția cazurilor legate de compatibilitatea cu versiunile anterioare etc.
  - Dacă trebuie să configurați variabile de mediu, creați un fișier cu numele „`~/.config/environment.d/50-input-me` pentru a le configura.
- Pentru X Window:
  - Instalați pachetul `im-config`.
  - Intrarea din meniul interfeței grafice pentru im-config(8) este „Metoda de introducere”.
  - Alternativ, executați „`im-config`” din shell-ul utilizatorului.
  - im-config(8) se comportă diferit dacă comanda este executată din contul root sau nu.
  - im-config(8) activează cea mai bună metodă de introducere a datelor din sistem ca metodă implicită, fără nicio acțiune din partea utilizatorului.

8.3 Afișarea ieșirii

Consola Linux poate afișa doar un număr limitat de caractere. (Pentru a afișa limbi non-europene pe consola fără interfață grafică, trebuie să utilizați un program terminal special, cum ar fi jfbterm(1).)

Mediul grafic (Cap. 7) poate afișa orice caractere în UTF-8, atâta timp cât fonturile necesare sunt instalate și activate; (codificarea datelor originale ale fonturilor este asigurată și transparentă pentru utilizator).

8.3.1 Configurația terminalului

Sistemul Debian poate fi configurat pentru a funcționa cu multe aranjamente internaționale ale tastaturii pentru consolă folosind pachetul `console-setup`.

```
# dpkg-reconfigure console-setup
```

Pentru consola Linux și sistemul X Window, acesta actualizează parametrii de configurare din „`/etc/default/console-s` astfel încât să poată fi afișate numeroase caractere non-ASCII, inclusiv caracterele accentuate utilizate în multe limbi europene.

Există mai multe componente pentru configurarea consolei de caractere și a caracteristicilor sistemului ncurses(3).

- Fișierul „`/etc/terminfo/*/*`” (terminfo(5))
- Variabila de mediu „`$TERM`” (term(7))

- setterm(1), stty(1), tic(1), și toe(1)

Dacă intrarea `terminfo` pentru `xterm` nu funcționează cu un `xterm` non-Debian, schimbați tipul terminalului, „`$TERM`”, din „`xterm`” într-una dintre versiunile cu funcționalități limitate, cum ar fi „`xterm -r6`” atunci când vă conectați la un sistem Debian de la distanță. Consultați „`/usr/share/doc/libncurses5/FAQ`” pentru mai multe informații. „`dumb`” este cel mai mic numitor comun pentru „`$TERM`”.

8.3.2 Caractere cu lățime de aspect ambiguu din Asia de Est

În configurația regională Asia de Est, caracterele din caseta de desen, grecești și chirilice pot fi afișate mai late decât lățimea dorită, ceea ce poate cauza o aliniere incorectă a ieșirii din terminal (a se vedea Anexa standard Unicode nr. 11, 4.2 Caractere ambigue).

Puteți rezolva această problemă:

- `gnome-terminal`: Preferințe → Profiluri → *Numele profilului* → Compatibilitate → Caractere cu lățime imprecisă → Înguste
- `ncurses`: Definiți mediul `export NCURSES_NO_UTF8_ACS=0`.

Capitolul 9

Sfaturi privind sistemul

Aici, descriu sfaturi de bază pentru configurarea și gestionarea sistemelor, în principal din consolă.

9.1 Sfaturi pentru consolă

Există câteva programe utilitare care vă ajută în activitățile cu consola.

pachet	popcon(popularitate) și dimensiune		descriere
mc	V:40, I:176	1590	consultați Secțiune 1.3
bsdutils	V:421, I:999	338	script(1) comandă pentru înregistrarea sesiunii terminalului
screen	V:50, I:191	1007	multiplexor de terminal cu emulare de terminal VT100/ANSI
tmux	V:106, I:167	1423	alternativă multiplexor de terminal (utilizați «Control-B» în loc de acesta)
ripgrep	V:17, I:48	5284	căutare rapidă recursivă de șiruri în arborele codului sursă cu filtrare automată
fzf	V:10, I:36	5677	căutare text aproximativ
fzy	V:0.07, I:0.34	59	căutare text aproximativ
rlwrap	V:1, I:11	328	învăluitor de linie de comandă pentru funcționalitatea readline
ledit	V:0.5, I:7.3	375	învăluitor de linie de comandă pentru funcționalitatea readline
rlfe	V:0.02, I:0.48	45	învăluitor de linie de comandă pentru funcționalitatea readline

Tabela 9.1: Lista programelor ce ajută în activitățile cu consola

9.1.1 Înregistrarea activităților shell în mod curat

Utilizarea simplă a script(1) (vezi Secțiune 1.4.9) pentru a înregistra activitatea shell-ului produce un fișier cu caractere de control. Acest lucru poate fi evitat utilizând col(1) după cum urmează.

```
$ script
Script started, file is typescript
```

Faceți orice ... și apăsați `Ctrl-D` pentru a ieși din `script`.

```
$ col -bx < typescript > cleanedfile
$ vim cleanedfile
```

Există metode alternative pentru înregistrarea activităților din shell:

- Utilizați `tee` (utilizabil în timpul procesului de pornire din initramfs):

```
$ sh -i 2>&1 | tee typescript
```

- Utilizați `gnome-terminal` cu tamponul de linie extins pentru derulare înapoi.
- Utilizați `screen` cu „`^A H`” (consultați Secțiune 9.1.2) pentru a efectua înregistrarea consolei.
- Utilizați `vim` cu „`:terminal`” pentru a intra în modul terminal. Utilizați „`Ctrl-W N`” pentru a ieși din modul terminal și a reveni la modul normal. Utilizați „`:w typescript`” pentru a scrie conținutul memoriei tampon într-un fișier.
- Utilizați `emacs` cu „`M-x shell`”, „`M-x eshell`” sau „`M-x term`” pentru a intra în consola de înregistrare. Utilizați „`C-x C-w`” pentru a scrie conținutul memoriei tampon într-un fișier.

9.1.2 Programul «screen»

screen(1) nu numai că permite unei ferestre de terminal să lucreze cu mai multe procese, dar permite și **procesului shell de la distanță să supraviețuiască conexiunilor întrerupte**. Iată un scenariu tipic de utilizare a screen(1).

1. Vă conectați la o mașină de la distanță.
2. Porniți `screen` pe o singură consolă.
3. Executați mai multe programe în ferestrele `screen` create cu `^A c` («Control-A» urmat de «c»).
4. Comutați între mai multe ferestre `screen` folosind comanda `^A n` («Control-A» urmat de «n»).
5. Dintr-o dată trebuie să părăsiți terminalul, dar nu doriți să pierdeți munca activă menținând conexiunea.
6. Puteți **detașa** sesiunea `screen` prin orice metodă.
   - Deconectați brutal conexiunea la rețea
   - Tastați `^A d` («Control-A» urmat de «d») și deconectați-vă manual de la conexiunea de la distanță
   - Tastați `^A DD` («Control-A» urmat de «DD») pentru a detașa `screen` și a vă deconecta
7. Vă conectați din nou la aceeași mașină de la distanță (chiar și de la un terminal diferit).
8. Porniți `screen` ca „`screen -r`”.
9. `screen` **reatașează** în mod magic toate ferestrele `screen` anterioare cu toate programele care rulează în mod activ.

Indicație
Puteți economisi costurile de conectare cu `screen` pentru conexiunile de rețea contorizate, cum ar fi cele dial-up și cele pe pachete, deoarece puteți lăsa un proces activ în timp ce sunteți deconectat și apoi îl puteți reatașa mai târziu, când vă conectați din nou.

Într-o sesiune `screen`, toate introducerile de la tastatură sunt trimise către fereastra curentă, cu excepția tastelor de comandă. Toate tastele de comandă `screen` sunt introduse tastând `^A` («Control-A») plus o singură tastă [plus orice parametri]. Iată câteva dintre cele mai importante de reținut.

Pentru detalii, consultați screen(1).

Consultați tmux(1) pentru funcționalitățile comenzii alternative.

comanda/funcția asociată	semnificație
^A ?	afișează un ecran de ajutor (afișează combinațiile de taste)
^A c	crează o fereastră nouă și comută la ea
^A n	trece la fereastra următoare
^A p	trece la fereastra anterioară
^A 0	trece la fereastra numărul 0
^A 1	trece la fereastra numărul 1
^A w	afișează o listă de ferestre
^A a	trimite «Ctrl-A» către fereastra curentă ca intrare de la tastatură
^A h	scrie o copie fizică a ferestrei curente într-un fișier
^A H	începe/încheie înregistrarea ferestrei curente în fișier
^A ^X	blochează terminalul (protejat cu parolă)
^A d	detașează sesiunea «screen» de terminal
^A DD	detașează sesiunea «screen» și termină sesiunea

Tabela 9.2: Lista combinațiilor de taste pentru «screen»

9.1.3 Navigarea între directoare

În Secțiune 1.4.2 sunt descrise două sfaturi care permit navigarea rapidă între directoare: `$CDPATH` și `mc`.

Dacă utilizați un program de filtrare a textului aproximativ, puteți renunța la introducerea rutei exacte. Pentru `fzf`, includeți următoarele în `~/.bashrc`.

```
FZF_KEYBINDINGS_PATH=/usr/share/doc/fzf/examples/key-bindings.bash
if [ -f $FZF_KEYBINDINGS_PATH ]; then
  . $FZF_KEYBINDINGS_PATH
fi
```

De exemplu:

- Puteți salta într-un subdirector foarte adânc cu eforturi minime. Mai întâi tastați „`cd **`” și apăsați `Tab`. Apoi vi se vor afișa rutele candidate. Tastând șiruri parțiale de rută, de exemplu, `s/d/b foo`, veți restrânge rutele candidate. Selectați ruta care va fi utilizată de `cd` cu ajutorul tastelor cursor și «Enter».
- Puteți selecta o comandă din istoricul comenzilor mai eficient, cu eforturi minime. Apăsați `Ctrl-R` la promptul de comandă. Apoi vi se vor afișa comenzile candidate. Tastând șiruri parțiale de comenzi, de exemplu `vim d`, veți restrânge numărul de comenzi candidate. Selectați comanda dorită cu ajutorul cursorului și tastelor de returnare.

9.1.4 Readline wrapper (învăluitorul «readline»)

Unele comenzi, cum ar fi `/usr/bin/dash`, care nu dispun de funcția de editare a istoricului liniei de comandă, pot adăuga această funcționalitate în mod transparent prin rularea sub `rlwrap` sau echivalenții săi.

```
$ rlwrap dash -i
```

Aceasta oferă o platformă convenabilă pentru testarea punctelor subtile pentru `dash` într-un mediu prietenos, similar cu `bash`.

9.1.5 Scanarea arborelui codului sursă

Comanda rg(1) din pachetul `ripgrep` oferă o alternativă mai rapidă la comanda grep(1) pentru scanarea arborelui codului sursă în situații tipice. Aceasta profită de avantajele procesoarelor moderne multi-core și aplică automat filtre rezonabile pentru a omite anumite fișiere.

9.2 Personalizarea vim

După ce învățați noțiunile de bază ale vim(1) prin Secțiune 1.4.8, vă rugăm să citiți articolul lui Bram Moolenaar „Seven habits of effective text editing (2000)” (Șapte obiceiuri pentru o editare eficientă a textului) pentru a înțelege cum trebuie utilizat `vim`.

9.2.1 Personalizarea vim cu caracteristicile interne

Comportamentul `vim` poate fi modificat semnificativ prin activarea caracteristicilor sale interne prin intermediul comenzilor Ex-mode, cum ar fi „`set ...`” pentru a defini opțiunile vim.

Aceste comenzi în modul Ex pot fi incluse în fișierul vimrc al utilizatorului, tradiționalul „`~/.vimrc`” sau „`~/.vim/vimrc`”, compatibil cu git. Iată un exemplu foarte simplu 1:

```
""" Generic baseline Vim and Neovim configuration (~/.vimrc)
"""   - For NeoVim, use "nvim -u ~/.vimrc [filename]"
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
let mapleader = ' '             " :h mapleader
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
set nocompatible                " :h 'cp -- sensible (n)vim mode
syntax on                       " :h :syn-on
filetype plugin indent on       " :h :filetype-overview
set encoding=utf-8              " :h 'enc (default: latin1) -- sensible encoding
""" current vim option value can be verified by :set encoding?
set backspace=indent,eol,start  " :h 'bs (default: nobs) -- sensible BS
set statusline=%<%f%m%r%h%w%=%y[U+%04B]%2l/%2L=%P,%2c%V
set listchars=eol:¶,tab:b''▫b''\ ,extends:b''▫b'',precedes:b''▫b'',nbsp:b''▫b''
set viminfo=!,'100,<5000,s100,h " :h 'vi -- bigger copy buffer etc.
""" Pick "colorscheme" from blue darkblue default delek desert elflord evening
""" habamax industry koehler lunaperche morning murphy pablo peachpuff quiet ron
""" shine slate torte zellner
colorscheme industry
""" don't pick "colorscheme" as "default" which may kill SpellUnderline settings
set scrolloff=5                 " :h 'scr -- show 5 lines around cursor
set laststatus=2                " :h 'ls (default 1)  k
""" boolean options can be unset by prefixing "no"
set ignorecase                  " :h 'ic
set smartcase                   " :h 'scs
set autoindent                  " :h 'ai
set smartindent                 " :h 'si
set nowrap                      " :h 'wrap
"set list                        " :h 'list (default nolist)
set noerrorbells                " :h 'eb
set novisualbell                " :h 'vb
set t_vb=                       " :h 't_vb -- termcap visual bell
set spell                       " :h 'spell
set spelllang=en_us,cjk         " :h 'spl -- english spell, ignore CJK
set clipboard=unnamedplus       " :h 'cb -- cut/copy/paste with other app
set hidden                      " :h 'hid
set autowrite                   " :h 'aw
set timeoutlen=300              " :h 'tm
```

Harta tastelor din `vim` poate fi modificată în fișierul vimrc al utilizatorului. De exemplu:



Atenție
Nu încercați să modificați combinațiile de taste implicite fără motive întemeiate.

1Exemple de personalizare mai elaborate: „Vim Galore”, „sensible.vim”, ...

```
""" Popular mappings (imitating LazyVim etc.)
""" Window moves without using CTRL-W which is dangerous in INSERT mode
nnoremap <C-H> <C-W>h
nnoremap <C-J> <C-W>j
nnoremap <C-K> <C-W>k
silent! nnoremap <C-L> <C-W>l
""" Window resize
nnoremap <C-LEFT> <CMD>vertical resize -2<CR>
nnoremap <C-DOWN> <CMD>resize -2<CR>
nnoremap <C-UP> <CMD>resize +2<CR>
nnoremap <C-RIGHT> <CMD>vertical resize +2<CR>
""" Clear hlsearch with <ESC> (<C-L> is mapped as above)
nnoremap <ESC> <CMD>noh<CR><ESC>
inoremap <ESC> <CMD>noh<CR><ESC>
""" center after jump next
nnoremap n nzz
nnoremap N Nzz
""" fast "jk" to get out of INSERT mode (<ESC>)
inoremap  jk <CMD>noh<CR><ESC>
""" fast "<ESC><ESC>" to get out of TERM mode (CTRL-\ CTRL-N)
tnoremap <ESC><ESC> <C-\><C-N>
""" fast "jk" to get out of TERM mode (CTRL-\ CTRL-N)
tnoremap jk <C-\><C-N>
""" previous/next trouble/quickfix item
nnoremap [q <CMD>cprevious<CR>
nnoremap ]q <CMD>cnext<CR>
""" buffers
nnoremap <S-H> <CMD>bprevious<CR>
nnoremap <S-L> <CMD>bnext<CR>
nnoremap [b <CMD>bprevious<CR>
nnoremap ]b <CMD>bnext<CR>
""" Add undo break-points
inoremap  , ,<C-G>u
inoremap  . .<C-G>u
inoremap  ; ;<C-G>u
""" save file
inoremap <C-S> <CMD>w<CR><ESC>
xnoremap <C-S> <CMD>w<CR><ESC>
nnoremap <C-S> <CMD>w<CR><ESC>
snoremap <C-S> <CMD>w<CR><ESC>
""" better indenting
vnoremap < <gv
vnoremap > >gv
""" terminal (Somehow under Linux, <C-/> becomes <C-_> in Vim)
nnoremap <C-_> <CMD>terminal<CR>
"nnoremap <C-/> <CMD>terminal<CR>
""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
if ! has('nvim')
""" Toggle paste mode with <SPACE>p for Vim (no need for Nvim)
set pastetoggle=<leader>p
""" nvim default mappings for Vim.  See :h default-mappings in nvim
""" copy to EOL (no delete) like D for d
noremap Y y$
""" sets a new undo point before deleting
inoremap <C-U> <C-G>u<C-U>
inoremap <C-W> <C-G>u<C-W>
""" <C-L> is re-purposed as above
""" execute the previous macro recorded with Q
nnoremap Q @@
""" repeat last substitute and *KEEP* flags
```

```
nnoremap & :&&<CR>
""" search visual selected string for visual mode
xnoremap * y/\V<C-R>"<CR>
xnoremap # y?\V<C-R>"<CR>
endif
```

Pentru ca combinațiile de taste de mai sus să funcționeze corect, programul de terminal trebuie configurat să genereze „ASCII DEL” pentru tasta `Backspace` și „Secvență de eludare” pentru tasta `Delete`.

Alte configurații diverse pot fi modificate în fișierul vimrc al utilizatorului. De exemplu:

```
""" Use faster 'rg' (ripgrep package) for :grep
if executable("rg")
  set grepprg=rg\ --vimgrep\ --smart-case
  set grepformat=%f:%l:%c:%m
endif
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
""" Retain last cursor position :h '"
augroup RetainLastCursorPosition
  autocmd!
  autocmd BufReadPost *
    \ if line("'\"") > 0 && line ("'\"") <= line("$") |
    \   exe "normal! g'\"" |
    \ endif
augroup END
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
""" Force to use underline for spell check results
augroup SpellUnderline
  autocmd!
  autocmd ColorScheme * highlight SpellBad term=Underline gui=Undercurl
  autocmd ColorScheme * highlight SpellCap term=Underline gui=Undercurl
  autocmd ColorScheme * highlight SpellLocal term=Underline gui=Undercurl
  autocmd ColorScheme * highlight SpellRare term=Underline gui=Undercurl
augroup END
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
""" highlight tailing spaces except when typing as red (set after colorscheme)
highlight TailingWhitespaces ctermbg=red guibg=red
""" \s\+     1 or more whitespace character: <Space> and <Tab>
""" \%#\@<!  Matches with zero width if the cursor position does NOT match.
match TailingWhitespaces /\s\+\%#\@<!$/
```

9.2.2 Personalizarea vim cu pachete externe

Se pot găsi pachete de module de extensii externe interesante:

- Vim - the ubiquitous text editor -- Situl oficial al Vim și scripturilor vim
- VimAwsome -- Lista modulelor de extensie Vim
- vim-scripts -- Pachet Debian: o colecție de scripturi vim

Pachetele de module de extensie din pacheul vim-scripts pot fi activate folosind fișierul vimrc al utilizatorului. De exemplu:

```
packadd! secure-modelines
packadd! winmanager
" IDE-like UI for files and buffers with <space>w
nnoremap <leader>w         :WMToggle<CR>
```

Noul sistem nativ de pachete Vim funcționează bine cu „`git`” și „`git submodule`”. Un astfel de exemplu de configurare poate fi găsit în depozitul meu git: dot-vim. Acesta face în esență următoarele:

- Folosind „`git`” și „`git submodule`”, cele mai recente pachete externe, precum „*nume*”, sunt plasate în `~/.vim/pack/*/` și similare.
- Prin adăugarea liniei `:packadd! nume` în fișierul vimrc al utilizatorului, aceste pachete sunt plasate în `runtimepath`.
- Vim încarcă aceste pachete în `runtimepath` în timpul inițializării.
- La finalul inițializării, etichetele pentru documentele instalate sunt actualizate cu „`helptags ALL`”.

Pentru mai multe informații, porniți `vim` cu „`vim --startuptime vimstart.log`” pentru a verifica secvența reală de execuție și timpul utilizat pentru fiecare etapă.

Este destul de confuz să vezi prea multe modalități[2] de a gestiona și încărca aceste pachete externe în `vim`. Verificarea informațiilor originale este cea mai bună soluție.

combinația de taste	**informații**
`:help package`	explicație privind mecanismul pachetului vim
`:help runtimepath`	explicație privind mecanismul `runtimepath`
`:version`	stări interne, inclusiv candidați pentru fișierul vimrc
`:echo $VIM`	variabila de mediu „`$VIM`” utilizată pentru localizarea fișierului vimrc
`:set runtimepath?`	lista directoarelor care vor fi căutate pentru toate fișierele de suport pentru rulare
`:echo $VIMRUNTIME`	variabila de mediu „`$VIMRUNTIME`” utilizată pentru localizarea diverselor fișiere de suport pentru rulare furnizate de sistem

Tabela 9.3: Informații despre inițializarea `vim`

9.3 Înregistrarea și prezentarea datelor

9.3.1 Demonul de jurnalizare

Multe programe tradiționale înregistrează activitățile lor în format fișier text în directorul „`/var/log/`”.

logrotate(8) este utilizat pentru a simplifica administrarea fișierelor jurnal pe un sistem care generează o mulțime de fișiere jurnal.

Multe programe noi își înregistrează activitățile în formatul de fișier binar utilizând serviciul de jurnal systemd-journald(8) în directorul „`/var/log/journal`”.

Puteți înregistra date în jurnalul systemd-journald(8) dintr-un script shell utilizând comanda systemd-cat(1).

Consultați Secțiune 3.5 și Secțiune 3.4.

9.3.2 Analizator de jurnale

Iată câteva analizatoare de jurnale notabile („`~Gsecurity::log-analyzer`” în aptitude(8)).

Notă
CRM114 oferă infrastructura lingvistică necesară pentru a scrie filtre **fuzzy** (aproximative) cu ajutorul bibliotecii TRE regex. Este utilizat în mod obișnuit pentru filtrarea mesajelor spam, dar poate fi folosit și ca analizor de jurnale.

2vim-pathogen a fost popular.

pachet	popcon(popularitate) și dimensiune	dimensiune	descriere
fail2ban	V:101, I:111	2275	interzice adresele IP care provoacă erori multiple de autentificare
logwatch	V:9, I:10	2440	analizator de jurnale cu ieșire formatată plăcut, scris în Perl
awstats	V:5.1, I:8.0	6934	analizator de jurnale de server web, performant și cu multe funcții
analog	V:3, I:83	3739	analizator de jurnale de server web
sarg	V:0.65, I:0.71	863	generator de rapoarte de analizare squid
pflogsumm	V:1.5, I:3.5	173	program de sinteză a intrărilor din jurnalul Postfix
fwlogwatch	V:0.10, I:0.16	487	analizator de jurnale de paravan de protecție
squidview	V:0.04, I:0.41	224	monitorizează și analizează fișierele squid access.log
swatch	V:0.07, I:0.30	99	vizor de fișiere jurnal cu potrivire de expresii regulate, evidențiere și cârlige
crm114	V:0.06, I:0.32	1371	Filtru de spam și filtru cu expresii regulate controlabile („Controllable Regex Mutilator”: CRM114)
icmpinfo	V:0.01, I:0.35	42	interpretează mesajele ICMP

Tabela 9.4: Lista analizatoarelor de jurnale de sistem

9.3.3 Afișarea personalizată a datelor textuale

Deși instrumentele de paginare precum more(1) și `less` (1) (a se vedea Secțiune 1.4.5) și instrumentele personalizate pentru evidențiere și formatare (a se vedea Secțiune 11.1.8) pot afișa datele de text într-un mod plăcut, editorii de uz general (a se vedea Secțiune 1.4.6) sunt cei mai versatili și cei care pot fi cel mai mult personalizați.

Indicație
Pentru vim(1) și pseudonimul său pentru modul de paginator view(1), „`:set hls`” activează căutarea evidențiată.

9.3.4 Afișare personalizată a orei și datei

Formatul implicit de afișare a orei și datei de către comanda „`ls -l`” depinde de **configurația regională** (vedeți Secțiune 1.2.6 pentru valoare). Variabila „`$LANG`” este consultată prima și poate fi suprascrisă de variabilele de mediu exportate „`$LC_TIME`” sau „`$LC_ALL`”.

Formatul de afișare implicit pentru fiecare configurație regională depinde de versiunea bibliotecii standard C (pachetul `libc6`) utilizată. Adică, versiunile diferite ale Debian aveau configurații implicite diferite. Pentru formatele iso, consultați ISO 8601.

Dacă doriți cu adevărat să personalizați acest format de afișare a datei și orei dincolo de **configurația regională**, trebuie să definiți **valoarea stilului de afișare a datei și orei** prin argumentul „`--time-style`” sau prin valoarea „`$TIME_STYLE`” (consultați ls(1), date(1), «`info coreutils 'ls invocation'`»).

Indicație
Puteți elimina introducerea opțiunilor lungi în linia de comandă utilizând un alias de comandă (consultați Secțiune 1.5.9):

```
alias ls='ls --time-style=+%d.%m.%y %H:%M'
```

valoarea stilului de afișare a datei și orei	configurația regională	afișarea datei și orei
`iso`	oricare	`01-19 00:15`
`long-iso`	oricare	`2009-01-19 00:15`
`full-iso`	oricare	`2009-01-19 00:15:16.000000000 +0900`
`locale`	`C`	`Jan 19 00:15`
`locale`	`en_US.UTF-8`	`Jan 19 00:15`
`locale`	`es_ES.UTF-8`	`ene 19 00:15`
`+%d.%m.%y %H:%M`	oricare	`19.01.09 00:15`
`+%d.%b.%y %H:%M`	C sau `en_US.UTF-8`	`19.Jan.09 00:15`
`+%d.%b.%y %H:%M`	`es_ES.UTF-8`	`19.ene.09 00:15`

Tabela 9.5: Exemplele de afișare a datei și orei pentru comanda „`ls -l`” cu **valoarea stilului de afișare a datei și orei**

9.3.5 Ecou colorat al shell-ului

Ecoul Shell pentru majoritatea terminalelor moderne poate fi colorat folosind codul de eludare ANSI (vedeți „`/usr/share/do`

De exemplu, încercați următoarele

```
$ RED=$(printf "\x1b[31m")
$ NORMAL=$(printf "\x1b[0m")
$ REVERSE=$(printf "\x1b[7m")
$ echo "${RED}RED-TEXT${NORMAL} ${REVERSE}REVERSE-TEXT${NORMAL}"
```

9.3.6 Comenzi colorate

Comenzile colorate sunt utile pentru inspectarea rezultatelor acestora în mediul interactiv. Includ următoarele în fișierul meu „`~/.bashrc`”.

```
if [ "$TERM" != "dumb" ]; then
    eval "`dircolors -b`"
    alias ls='ls --color=always'
    alias ll='ls --color=always -l'
    alias la='ls --color=always -A'
    alias less='less -R'
    alias ls='ls --color=always'
    alias grep='grep --color=always'
    alias egrep='egrep --color=always'
    alias fgrep='fgrep --color=always'
    alias zgrep='zgrep --color=always'
else
    alias ll='ls -l'
    alias la='ls -A'
fi
```

Utilizarea unui alias limitează efectele de culoare la utilizarea interactivă a comenzii. Are avantajul față de exportarea variabilei de mediu „`export GREP_OPTIONS='--color=auto'`”, deoarece culoarea poate fi văzută în programele de paginare, cum ar fi less(1). Dacă doriți să suprimați culoarea atunci când redirecționați către alte programe, utilizați „`--color=auto`” în locul exemplului de mai sus pentru „`~/.bashrc`”.

Indicație
Puteți dezactiva aceste nume-alias de colorare în mediul interactiv apelând shell cu „`TERM=dumb bash`”.

9.3.7 Înregistrarea activităților editorului pentru repetări complexe

Puteți înregistra activitățile editorului pentru repetări complexe.

Pentru Vim, procedați după cum urmează.

- «qa»: începe înregistrarea caracterelor tastate în registrul numit „a”.
- … activități ale editorului
- «q»: încheie înregistrarea caracterelor tastate.
- «@a»: execută conținutul registrului „a”.

Pentru Emacs, procedați după cum urmează.

- «C-x (»: începe definirea unei macrocomenzi pentru tastatură.
- … activități ale editorului
- «C-x)»: încheie definirea macrocomenzii pentru tastatură.
- «C-x e»: execută o macrocomandă pentru tastatură.

9.3.8 Înregistrarea imaginii grafice a unei aplicații X

Există câteva modalități de a înregistra imaginea grafică a unei aplicații X, inclusiv un afișaj `xterm`.

pachet	popcon(popularitate)și	dimensiune	ecran
gnome-screenshot	V:12, I:94	1115	Wayland
flameshot	V:7, I:17	3542	Wayland
gimp	V:34, I:211	33315	Wayland + X
x11-apps	V:31, I:443	2461	X
imagemagick	V:7, I:272	80	X
scrot	V:4, I:48	144	X

Tabela 9.6: Lista instrumentelor de manipulare a imaginilor grafice

9.3.9 Înregistrarea modificărilor din fișierele de configurare

Există instrumente specializate pentru înregistrarea modificărilor din fișierele de configurare cu ajutorul DVCS și pentru realizarea de instantanee ale sistemului pe Btrfs.

pachet	popcon(popularitate)și	dimensiune	descriere
etckeeper	V:24, I:27	157	stochează fișierele de configurare și metadatele acestora cu Git (implicit), Mercurial sau GNU Bazaar
timeshift	V:7, I:13	4481	utiilitate de restaurare a sistemului folosind rsync sau instantanee BTRFS
snapper	V:6.1, I:8.3	2396	instrument de gestionare a instantaneelor sistemului de fișiere Linux

Tabela 9.7: Lista pachetelor care pot înregistra istoricul configurațiilor

Puteți lua în considerare și abordarea scriptului local Secțiune 10.2.3.

9.4 Monitorizarea, controlul și inițierea activităților programului

Activitățile programului pot fi monitorizate și controlate folosind instrumente specializate.

pachet	popcon(popularitate) și dimensiune	dimensiune	descriere
coreutils	V:912, I:1000	17994	nice(1): rulează un program cu prioritate de planificare modificată
bsdutils	V:421, I:999	338	renice(1): modifică prioritatea de planificare a unui proces în execuție
procps	V:831, I:997	2691	Utilități ale sistemului de fișiere „/proc”: ps(1), top(1), kill(1), watch(1), …
psmisc	V:400, I:715	950	Utilități ale sistemului de fișiere „/proc”: killall(1), fuser(1), peekfd(1), pstree(1)
time	V:6, I:76	148	time(1): rulează un program pentru a raporta utilizarea resurselor sistemului în funcție de timp
sysstat	V:115, I:164	1904	sar(1), iostat(1), mpstat(1), …: instrumente de performanță a sistemului pentru Linux
isag	V:0.1, I:3.1	109	Grafic interactiv al activității sistemului pentru sysstat
lsof	V:456, I:950	492	lsof(8): listează fișierele deschise de un proces în execuție folosind opțiunea „-p”
strace	V:10, I:100	3209	strace(1): urmărește apelurile de sistem și semnalele
ltrace	V:1, I:11	420	ltrace(1): trasează apelurile către bibliotecă
xtrace	V:0.03, I:0.69	342	xtrace(1): trasează comunicarea între clientul X11 și server
powertop	V:36, I:230	696	powertop(1): informații despre consumul de energie al sistemului
cron	V:923, I:997	248	rulează procesele conform unei programări în fundal din cron(8) daemon
anacron	V:423, I:481	110	programator de comenzi de tip cron pentru sisteme care nu funcționează 24 de ore pe zi
at	V:69, I:91	158	at(1) sau batch(1): execută o sarcină la o oră specificată sau sub un anumit nivel de încărcare

Tabela 9.8: Lista instrumentelor pentru monitorizarea și controlul activităților programului

Indicație
Pachetele procps oferă elementele de bază pentru monitorizarea, controlul și pornirea activităților programelor. Ar trebui să le învățați pe toate.

9.4.1 Cronometrarea unui proces

Afișează timpul utilizat de procesul invocat de comandă.

```
# time some_command >/dev/null
real    0m0.035s        # time on wall clock (elapsed real time)
user    0m0.000s        # time in user mode
sys     0m0.020s        # time in kernel mode
```

9.4.2 Prioritatea de planificare

O valoare potrivită(de curtoazie) este utilizată pentru a controla prioritatea de planificare a procesului.

valoarea de curtoazie	prioritatea de planificare
19	proces cu prioritate minimă (nice - curtoazie)
0	proces cu prioritate foarte mare pentru utilizator
-20	proces cu prioritate foarte mare pentru root (fără curtoazie)

Tabela 9.9: Lista valorilor de curtoazie pentru prioritatea de programare

```
# nice  -19 top                                    # very nice
# nice --20 wodim -v -eject speed=2 dev=0,0 disk.img # very fast
```

Uneori, o valoare extrem de mare poate face mai mult rău decât bine sistemului. Utilizați această comandă cu precauție.

9.4.3 Comanda «ps»

Comanda ps(1) pe un sistem Debian acceptă atât caracteristicile BSD, cât și SystemV și ajută la identificarea statică a activității procesului.

stil	comandă tipică	funcția
BSD	`ps aux`	afișează %CPU %MEM
System V	`ps -efH`	afișează PPID

Tabela 9.10: Lista stilurilor de comenzi ps

Procesele-copil zombie (inactive), le puteți opri utilizând ID-ul procesului părinte identificat în câmpul „`PPID`”.

Comanda pstree(1) afișează arborele proceselor.

9.4.4 Comanda «top»

top(1) în sistemul Debian are funcții bogate și ajută la identificarea proceselor care se comportă ciudat în mod dinamic.

Este un program interactiv pe ecran complet. Puteți obține ajutorul de utilizare apăsând tasta «h» și îl puteți închide apăsând tasta «q».

9.4.5 Listarea fișierelor deschise de un proces

Puteți lista toate fișierele deschise de un proces cu un ID de proces (PID), de exemplu 1, folosind următoarea comandă.

```
$ sudo lsof -p 1
```

PID=1 este de obicei programul `init`.

9.4.6 Urmărirea activităților programului

Puteți urmări activitatea programului cu strace(1), ltrace(1) sau xtrace(1) pentru apeluri și semnale de sistem, apeluri de bibliotecă sau comunicarea între clientul și serverul X11.

Puteți urmări apelurile de sistem ale comenzii `ls` după cum urmează.

```
$ sudo strace ls
```

Indicație
Utilizați scriptul **strace-graph** din **/usr/share/doc/strace/examples/** pentru a crea o vizualizare arborescentă plăcută

9.4.7 Identificarea proceselor care utilizează fișiere sau socluri

De asemenea, puteți identifica procesele utilizând fișiere prin fuser(1), de exemplu pentru „`/var/log/mail.log`” prin următoarea comandă.

```
$ sudo fuser -v /var/log/mail.log
                     USER        PID ACCESS COMMAND
/var/log/mail.log:   root       2946 F.... rsyslogd
```

Puteți vedea că fișierul „`/var/log/mail.log`” este deschis pentru scriere de comanda rsyslogd(8).

De asemenea, puteți identifica procesele care utilizează socluri cu ajutorul comenzii fuser(1), de exemplu pentru „`smtp/tcp`” cu ajutorul următoarei comenzi.

```
$ sudo fuser -v smtp/tcp
                     USER        PID ACCESS COMMAND
smtp/tcp:            Debian-exim   3379 F.... exim4
```

Acum știți că sistemul dvs. rulează exim4(8) pentru a gestiona conexiunile TCP la portul SMTP (25).

9.4.8 Repetarea unei comenzi la intervale constante

watch(1) execută un program în mod repetat, la intervale constante, afișând rezultatul pe ecranul complet.

```
$ watch w
```

Afișează cine este conectat la sistem, actualizat la fiecare 2 secunde.

9.4.9 Repetarea unei comenzi care parcurge fișierele

Există mai multe moduri de a repeta o comandă care parcurge fișierele care corespund unei anumite condiții, de exemplu, care corespund modelului global (cu caracter joker) „`*.ext`”.

- Metoda Shell for-loop (a se vedea Secțiune 12.1.4):

```
for x in *.ext; do if [ -f "$x"]; then command "$x" ; fi; done
```

- combinația find(1) și xargs(1):

```
find . -type f -maxdepth 1 -name '*.ext' -print0 | xargs -0 -n 1 command
```

- find(1) cu opțiunea „`-exec`” cu o comandă:

```
find . -type f -maxdepth 1 -name '*.ext' -exec command '{}' \;
```

- find(1) cu opțiunea „`-exec`” cu un script shell scurt:

```
find . -type f -maxdepth 1 -name '*.ext' -exec sh -c "command '{}' && echo 'successful'" \;
```

Exemplele de mai sus sunt scrise pentru a asigura gestionarea corectă a numelor de fișiere ciudate, cum ar fi cele care conțin spații. Consultați Secțiune 10.1.5 pentru utilizări mai avansate ale find(1).

9.4.10 Pornirea unui program din interfața grafică

Pentru interfața de linie de comandă („command-line interface”: CLI), se execută primul program cu numele corespunzător găsit în directoarele specificate în variabila de mediu `$PATH`. Consultați Secțiune 1.5.3.

Pentru interfața grafică cu utilizatorul („graphical user interface”: GUI) conformă cu standardele freedesktop.org, fișierele `*.desktop` din directorul `/usr/share/applications/` furnizează atributele necesare pentru afișarea meniului GUI al fiecărui program. Fiecare pachet care este compatibil cu sistemul de meniuri xdg al Freedesktop.org instalează datele de meniu furnizate de „*.desktop” în „/usr/share/applications/”. Mediile grafice de birou moderne care sunt compatibile cu standardul Freedesktop.org utilizează aceste date pentru a genera meniul lor folosind pachetul xdg-utils. Consultați „/usr/share/doc/xdg-utils/README”.

De exemplu, fișierul `chromium.desktop` definește atributele pentru „Navigatorul Web Chromium”, cum ar fi „Name” pentru numele programului, „Exec” pentru ruta de execuție și argumentele programului, „Icon” pentru pictograma utilizată etc. (consultați Descriptorul de intrare desktop) după cum urmează:

```
[Desktop Entry]
Version=1.0
Name=Chromium Web Browser
GenericName=Web Browser
Comment=Access the Internet
Comment[fr]=Explorer le Web
Exec=/usr/bin/chromium %U
Terminal=false
X-MultipleArgs=false
Type=Application
Icon=chromium
Categories=Network;WebBrowser;
MimeType=text/html;text/xml;application/xhtml_xml;x-scheme-handler/http;x-scheme-handler/ ↩
    https;
StartupWMClass=Chromium
StartupNotify=true
```

Aceasta este o descriere simplificată. Fișierele `*.desktop` sunt scanate după cum urmează.

Mediul de birou definește variabilele de mediu `$XDG_DATA_HOME` și `$XDG_DATA_DIR`. De exemplu, în GNOME:

- variabila `$XDG_DATA_HOME` nu este definită; (se utilizează valoarea implicită `$HOME/.local/share`)
- variabila `$XDG_DATA_DIRS` este definită ca `/usr/share/gnome:/usr/local/share/:/usr/share/`

Astfel, directoarele de bază (a se vedea XDG Base Directory Specification) și directoarele de `aplicații` sunt următoarele.

- `$HOME/.local/share/` → `$HOME/.local/share/applications/`
- `/usr/share/gnome/` → `/usr/share/gnome/applications/`

- `/usr/local/share/` → `/usr/local/share/applications/`
- `/usr/share/` → `/usr/share/applications/`

Fișierele `*.desktop` sunt scanate în aceste directoare de `aplicații` în această ordine.

Indicație
O intrare personalizată în meniul GUI poate fi creată prin adăugarea unui fișier `*.desktop` în directorul `$HOME/.local/share/applications/`.

Indicație
Linia „`Exec=...`” nu este analizată de shell. Utilizați comanda env(1) dacă trebuie să fie definite variabile de mediu.

Indicație
În mod similar, dacă un fișier `*.desktop` este creat în directorul `autostart` din aceste directoare de bază, programul specificat în fișierul `*.desktop` este executat automat la pornirea mediului de birou. Consultați Desktop Application Autostart Specification.

Indicație
În mod similar, dacă un fișier `*.desktop` este creat în directorul `$HOME/Desktop` și mediul grafic de birou este configurat să accepte funcția de lansare a pictogramelor de pe birou, programul specificat în acesta este executat la efectuarea unui clic cu mouse-ul pe pictogramă. Rețineți că numele real al directorului `$HOME/Desktop` depinde de configurația regională. Consultați xdg-user-dirs-update(1).

9.4.11 Personalizarea programului care urmează să fie pornit

Unele programe pornesc automat un alt program. Iată câteva puncte de verificare pentru personalizarea acestui proces.

- Meniul de configurare a aplicației:
  - mediul grafic de birou GNOME: «Configurări» → «Aplicații» → «Aplicații implicite»
  - mediul grafic de birou KDE: «Meniu aplicații» → «Configurări de sistem» → «Aplicații implicite»
  - navigatorul web Iceweasel: «Editare» → «Preferințe» → «Aplicații»
  - mc(1): ”`/etc/mc/mc.ext`”
- Variabile de mediu precum „`$BROWSER`”, „`$EDITOR`”, „`$VISUAL`” și „`$PAGER`” (a se vedea environ(7))
- Sistemul update-alternatives(1) pentru programe precum „`editor`”, „`view`”, „`x-www-browser`”, „`gnome-www-browser`” și „`www-browser`” (a se vedea Secțiune 1.4.7)
- Conținutul fișierelor „`~/.mailcap`” și „`/etc/mailcap`” care asociază tipul MIME cu programul (a se vedea mailcap(5))
- Conținutul fișierelor „`~/.mime.types`” și „`/etc/mime.types`” care asociază extensia numelui fișierului cu tipul MIME (a se vedea run-mailcap(1))

Indicație
update-mime(8) actualizează fișierul „`/etc/mailcap`” folosind fișierul „`/etc/mailcap.order`” (a se vedea mailcap.order(5)).

Indicație
Pachetul `debianutils` oferă sensible-browser(1), sensible-editor(1) și sensible-pager(1), care iau decizii rezonabile cu privire la editorul, paginatorul și navigatorul web care trebuie apelate. Vă recomand să citiți aceste scripturi shell.

Indicație
Pentru a rula o aplicație de consolă precum `mutt` sub GUI ca aplicație preferată, trebuie să creați o aplicație cu interfață grafică după cum urmează și să definiți „`/usr/local/bin/mutt-term`” ca aplicație preferată care să fie pornită conform descrierii.

```
# cat /usr/local/bin/mutt-term <<EOF
#!/bin/sh
gnome-terminal -e "mutt \$@"
EOF
# chmod 755 /usr/local/bin/mutt-term
```

9.4.12 Omorârea unui proces

Utilizați kill(1) pentru a omorî (sau a trimite un semnal către) un proces prin ID-ul procesului.

Utilizați killall(1) sau pkill(1) pentru a face același lucru folosind numele comenzii procesului și alte atribute.

9.4.13 Programarea sarcinilor o singură dată

Rulează comanda at(1) pentru a programa o sarcină unică după cum urmează.

```
$ echo 'command -args'| at 3:40 monday
```

9.4.14 Programarea regulată a sarcinilor

Utilizați cron(8) pentru a programa sarcini în mod regulat. Consultați crontab(1) și crontab(5).

Puteți programa rularea proceselor ca utilizator normal, de exemplu `foo`, creând un fișier crontab(5) ca „`/var/spool/cron/`” cu comanda „`crontab -e`”.

Iată un exemplu de fișier crontab(5).

```
# use /usr/bin/sh to run commands, no matter what /etc/passwd says
SHELL=/bin/sh
# mail any output to paul, no matter whose crontab this is
MAILTO=paul
# Min Hour DayOfMonth Month DayOfWeek command (Day... are OR'ed)
# run at 00:05, every day
5  0  *  * *   $HOME/bin/daily.job >> $HOME/tmp/out 2>&1
# run at 14:15 on the first of every month -- output mailed to paul
15 14 1  * *   $HOME/bin/monthly
# run at 22:00 on weekdays(1-5), annoy Joe. % for newline, last % for cc:
0 22 *   * 1-5 mail -s "It's 10pm" joe%Joe,%%Where are your kids?%.%%
23 */2 1 2 *   echo "run 23 minutes after 0am, 2am, 4am ..., on Feb 1"
5  4 *   * sun echo "run at 04:05 every Sunday"
# run at 03:40 on the first Monday of each month
40 3 1-7 * *   [ "$(date +%a)" == "Mon" ] && command -args
```

valoare semnal	nume semnal	acțiunea	notă
0	---	nu se trimite niciun semnal (vedeți kill(2))	verifică dacă procesul rulează
1	SIGHUP	termină procesul	terminal deconectat (semnal de suspendare)
2	SIGINT	termină procesul	întrerupere de la tastatură (`CTRL-C`)
3	SIGQUIT	termină procesul și efectuează operația dump core (descarcă conținutul memoriei)	ieșire de la tastatură (`CTRL-\`)
9	SIGKILL	termină procesul	semnal de omorâre de neblocat
15	SIGTERM	termină procesul	semnal de terminare blocabil

Tabela 9.11: Lista semnalelor utilizate frecvent pentru comanda «kill»

Indicație
Pentru ca sistemul să nu funcționeze continuu, instalați pachetul `anacron` pentru a programa comenzi periodice la intervale specificate, pe cât posibil în funcție de timpul de funcționare al mașinii. Consultați anacron(8) și anacrontab(5).

Indicație
Pentru scripturile de întreținere programată a sistemului, le puteți rula periodic din contul root, plasând astfel de scripturi în „`/etc/cron.hourly/`”, „`/etc/cron.daily/`”, „`/etc/cron.weekly/`” sau „`/etc/cron.monthly/`”. Momentele de execuție ale acestor scripturi pot fi personalizate prin „`/etc/crontab`” și „`/etc/anacrontab`”.

Systemd are capacitatea de nivel inferior de a programa rularea programelor fără demonul `cron`. De exemplu, `/lib/systemd/system/apt-daily.timer` și `/lib/systemd/system/apt-daily.service` configurează activitățile zilnice de descărcare apt. Consultați systemd.timer(5) .

9.4.15 Programarea sarcinilor la eveniment

Systemd poate programa programul nu numai pe evenimentul cronometrat, ci și pe evenimentul de montare. Consultați Secțiune 10.2.3.3 și Secțiune 10.2.3.2 pentru exemple.

9.4.16 Tasta Alt-SysRq

Apăsarea tastelor Alt-SysRq (PrtScr) urmată de o singură tastă face minunea de a recupera controlul asupra sistemului.

tasta ce urmează după Alt-SysRq	**descrierea acțiunii**
k	**k**ill (omoară) toate procesele din **consola virtuală curentă** (SAK)
s	**s**incronizează toate sistemele de fișiere montate pentru a evita coruperea datelor
u	remontează toate sistemele de fișiere montate în mod citire-numai (**u**mount)
r	restabilește tastatura din modul **r**aw (brut) după blocarea X

Tabela 9.12: Lista tastelor de comandă SAK importante

Vedeți mai multe în Linux kernel user’s and administrator’s guide » Linux Magic System Request Key Hacks

Indicație
Din terminalul SSH etc., puteți utiliza funcția Alt-SysRq scriind în „`/proc/sysrq-trigger`”. De exemplu, „`echo s > /proc/sysrq-trigger; echo u > /proc/sysrq-trigger`” din promptul shell-ului root **s**incronizează și **u**demontează toate sistemele de fișiere montate.

Nucleul Linux Debian amd64 actual (2021) are `/proc/sys/kernel/sysrq=438=0b110110110`:

- 2 = 0x2 - activează controlul nivelului de înregistrare în jurnalul consolei (ACTIVAT)
- 4 = 0x4 - activează controlul tastaturii (SAK, unraw) (ACTIVAT)
- 8 = 0x8 - activează dump-urile de depanare ale proceselor etc. (DEZACTIVAT)

- 16 = 0x10 - activează comanda de sincronizare (ACTIVAT)
- 32 = 0x20 - activează remontarea numai pentru citire (ACTIVAT)
- 64 = 0x40 - activează semnalizarea proceselor (term, kill, oom-kill) (dezactivat)
- 128 = 0x80 - permite repornirea/oprirea (ACTIVAT)
- 256 = 0x100 - permite optimizarea tuturor sarcinilor RT (ACTIVAT)

9.5 Sfaturi pentru întreținerea sistemului

9.5.1 Cine este în sistem?

Puteți verifica cine este conectat la sistem urmând pașii de mai jos.

- who (1) arată cine este conectat.
- w(1) arată cine este conectat și ce face.
- last(1) afișează lista ultimilor utilizatori conectați.
- lastb(1) afișează lista ultimilor utilizatori care s-au conectat incorect.

Indicație
„`/var/run/utmp`” și „`/var/log/wtmp`” conțin astfel de informații despre utilizatori. Consultați login(1) și utmp(5).

9.5.2 Avertisment pentru toată lumea

Puteți trimite mesaje tuturor celor care sunt conectați la sistem cu wall(1) în felul următor.

```
$ echo "We are shutting down in 1 hour" | wall
```

9.5.3 Identificarea hardware-ului

Pentru dispozitivele de tip PCI (AGP, PCI-Express, CardBus, ExpressCard etc.), lspci(8) (probabil cu opțiunea „`-nn`”) este un bun punct de plecare pentru identificarea hardware-ului.

Alternativ, puteți identifica hardware-ul citind conținutul „`/proc/bus/pci/devices`” sau răsfoind arborele de directoare din „`/sys/bus/pci`” (consultați Secțiune 1.2.12).

9.5.4 Configurația hardware

Deși majoritatea configurațiilor hardware ale sistemelor grafice de birou moderne, precum GNOME și KDE, pot fi gestionate prin intermediul instrumentelor de configurare grafice însoțitoare, este recomandabil să cunoașteți câteva metode de bază pentru configurarea acestora.

Pentru configurarea tastaturii și a terminalului, consultați Secțiune 8.2 și Secțiune 8.3.

Aici, ACPI este un cadru mai nou pentru sistemul de gestionare a energiei decât APM.

Indicație
Scalarea frecvenței procesorului pe sistemele moderne este controlată de module ale nucleului, precum `acpi_cpufreq`.

---

pachet	popcon(popularitate) și dimensiune	dimensiune instalată	descriere
pciutils	V:264, I:994	289	utilități PCI Linux: lspci(8)
usbutils	V:90, I:892	322	utilități USB Linux: lsusb(8)
nvme-cli	V:27, I:36	2222	utilități NVMe pentru Linux: nvme(1)
pcmciautils	V:4.1, I:6.1	92	utilități PCMCIA pentru Linux: pccardctl(8)
scsitools	V:0.2, I:2.1	261	colecție de instrumente pentru gestionarea hardware-ului SCSI: lsscsi(8)
procinfo	V:0.4, I:5.8	149	informații despre sistem obținute din „/proc”: lsdev(8)
lshw	V:12, I:88	971	informații despre configurația hardware: lshw(1)
discover	V:24, I:561	81	sistem de identificare hardware: discover (8)

Tabela 9.13: Lista instrumentelor de identificare a hardware-ului

pachet	popcon(popularitate) și dimensiune	dimensiune instalată	descriere
console-setup	V:81, I:976	422	utilități pentru fonturi și tabele de taste pentru consola Linux
keyd	V:1.5, I:1.7	1437	demon de reasociere a tastelor de pe tastatură care utilizează primitive de intrare la nivel de nucleu (evdev, uinput)
keyd-application-mapper	V:0.01, I:0.08	34	demon de reasociere a tastelor de la tastatură – program de reasociere specific aplicației
x11-xserver-utils	V:373, I:533	559	utilități server X: xset(1), xmodmap(1)
acpid	V:52, I:78	158	demon pentru gestionarea evenimentelor transmise de Interfața avansată de configurare și alimentare (ACPI)
acpi	V:6, I:73	49	instrument pentru afișarea informațiilor despre dispozitivele ACPI
sleepd	V:0.07, I:0.10	84	demon pentru a pune laptopul în stare de repaus în timpul inactivității
hdparm	V:110, I:200	246	optimizarea accesului la discurile dure (a se vedea Secțiune 9.6.9)
smartmontools	V:242, I:270	2455	control and monitor storage systems using S.M.A.R.T.
setserial	V:3.0, I:4.9	104	colecție de instrumente pentru gestionarea porturilor seriale
memtest86+	V:1, I:19	12483	colecție de instrumente pentru gestionarea hardware-ului de memorie
scsitools	V:0.2, I:2.1	261	colecție de instrumente pentru gestionarea hardware-ului SCSI
setcd	V:0.05, I:0.61	33	optimizarea accesului la unitatea de disc compact
big-cursor	I:0.67	26	cursoare de mouse mai mari pentru X

Tabela 9.14: Lista instrumentelor de configurare hardware

9.5.5 Ora sistemului și ora hardware-ului

Următoarele stabilesc ora sistemului și a hardware-ului la MM/DD hh:mm, CCYY.

```
# date MMDDhhmmCCYY
# hwclock --utc --systohc
# hwclock --show
```

Ora este afișată în mod normal în ora locală pe sistemul Debian, dar hardware-ul și ora sistemului utilizează de obicei UTC(GMT).

Dacă ora hardware este stabilită la UTC, modificați configurația la „`UTC=yes`” în „`/etc/default/rcS`”.

Următoarele reconfigurează fusul orar utilizat de sistemul Debian.

```
# dpkg-reconfigure tzdata
```

Dacă doriți să actualizați ora sistemului prin rețea, luați în considerare utilizarea serviciului NTP cu pachete precum `ntp`, `ntpdate` și `chrony`.

Indicație
În systemd, utilizați în schimb `systemd-timesyncd` pentru sincronizarea orei în rețea. Consultați systemd-timesyncd(8).

Consultați următoarele.

- Managing Accurate Date and Time HOWTO
- NTP Public Services Project
- Pachetul `ntp-doc`

Indicație
ntptrace(8) din pachetul `ntp` poate urmări un lanț de servere NTP până la sursa primară.

9.5.6 Infrastructura de sunet

Controlorii pentru plăcile de sunet pentru versiunea actuală de Linux sunt furnizați de Advanced Linux Sound Architecture (ALSA). ALSA oferă modul de emulare pentru Open Sound System (OSS) anterior, pentru compatibilitate.

Aplicațiile software pot fi configurate nu numai pentru a accesa direct dispozitivele audio, ci și pentru a le accesa prin intermediul unui sistem standardizat de servere audio. În prezent, PulseAudio, JACK și PipeWire sunt utilizate ca sisteme de servere audio. Consultați Debian wiki page on Sound -- pagina wiki Debian despre sunet pentru a afla ultimele noutăți.

De obicei, există un motor de sunet comun pentru fiecare mediu grafic de birou popular. Fiecare motor de sunet utilizat de aplicație poate alege să se conecteze la diferite servere de sunet.

Indicație
Utilizați „`cat /dev/urandom > /dev/audio`” sau speaker-test(1) pentru a testa difuzorul (^C pentru a opri).

Indicație
Dacă nu auziți sunetul, este posibil ca difuzorul să fie conectat la o ieșire mută. Sistemele audio moderne au mai multe ieșiri. alsamixer(1) din pachetul `alsa-utils` este util pentru configurarea volumului și a opțiunilor de dezactivare a sunetului.

pachet	popcon(popularitate) și dimensiune	dimensiune	descriere
alsa-utils	V:342, I:460	2700	Instrumente pentru configurarea și utilizarea ALSA
oss-compat	V:0.6, I:9.2	21	compatibilitatea OSS sub ALSA previne erorile „/dev/dsp not found”
pipewire	V:326, I:370	136	audio and video processing engine multimedia server - metapachet
pipewire-bin	V:333, I:370	2251	audio and video processing engine multimedia server - server audio și programe CLI
pipewire-alsa	V:172, I:241	194	audio and video processing engine multimedia server - server audio pentru a înlocui ALSA
pipewire-pulse	V:298, I:347	53	audio and video processing engine multimedia server - server audio pentru a înlocui PulseAudio
pulseaudio	V:147, I:171	6616	server PulseAudio
libpulse0	V:443, I:579	984	bibliotecă client pentru PulseAudio
jackd	V:2, I:14	8	server (latență redusă) JACK Audio Connection Kit. (JACK)
libjack0	V:1.8, I:8.6	328	bibliotecă (latență redusă) pentru JACK Audio Connection Kit. (JACK)
libphonon4qt5-4	V:18, I:42	572	Phonon: motorul de sunet KDE

Tabela 9.15: Lista pachetelor de sunet

9.5.7 Dezactivarea protectorului de ecran

Pentru a dezactiva protectorul de ecran, utilizați următoarele comenzi.

mediu	comandă
Consola Linux	setterm -powersave off
Sistemul X Window (dezactivarea protectorului de ecran)	xset s off
Sistemul X Window (dezactivarea dpms)	xset -dpms
Sistemul X Window (intrerfața grafigă de configurare a protectorului de ecran)	xscreensaver-command -prefs

Tabela 9.16: Lista comenzilor pentru dezactivarea protectorului de ecran

9.5.8 Dezactivarea bipurilor sonore

Se poate întotdeauna deconecta difuzorul PC-ului pentru a dezactiva sunetele bip. Eliminarea modulului nucleului `pcspkr` face acest lucru pentru dvs.

Următorul cod împiedică programul readline(3) utilizat de bash(1) să emită un semnal sonor atunci când întâlnește un caracter de alertă (ASCII=7).

```
$ echo "set bell-style none">> ~/.inputrc
```

9.5.9 Utilizare memorie

Există 2 resurse disponibile pentru a afla situația utilizării memoriei.

- Mesajul de pornire al nucleului din „/var/log/dmesg” conține dimensiunea totală exactă a memoriei disponibile.

- free(1) și top(1) afișează informații despre resursele de memorie ale sistemului în funcțiune.

Iată un exemplu.

```
# grep '\] Memory' /var/log/dmesg
[    0.004000] Memory: 990528k/1016784k available (1975k kernel code, 25868k reserved, 931k ↩
    data, 296k init)
$ free -k
             total       used       free     shared    buffers     cached
Mem:        997184     976928      20256          0     129592     171932
-/+ buffers/cache:     675404     321780
Swap:      4545576          4    4545572
```

Poate vă întrebați „dmesg îmi arată că sunt disponibili 990 Mo, iar «free -k» arată că sunt disponibili 320 Mo. Lipsește mai mult de 600 Mo...”.

Nu vă faceți griji cu privire la dimensiunea mare a „`utilizată`” și dimensiunea mică a „`liberă`” din linia „`Mem:`”, ci citiți cea de sub ele (675404 și 321780 în exemplul de mai sus) și relaxați-vă.

Pentru MacBook-ul meu cu 1 Go = 1048576 k DRAM (sistemul video consumă o parte din această memorie), observ următoarele.

raportare	dimensiune
Dimensiunea totală în dmesg	1016784k = 1Go - 31792k
Liberă în dmesg	990528k
Total sub shell	997184k
Liberă sub shell	20256k (dar efectiv 321780k)

Tabela 9.17: Lista dimensiunilor memoriei raportate

9.5.10 Verificarea securității și integrității sistemului

O întreținere deficitară a sistemului poate expune sistemul la exploatare externă.

Pentru verificarea securității și integrității sistemului, ar trebui să începeți cu următoarele.

- Pachetul `debsums`, consultați debsums(1) și Secțiune 2.5.2.
- Pachetul `chkrootkit`, consultați chkrootkit(1).
- Familia de pachete `clamav`, consultați clamscan(1) și freshclam(1).
- Debian security FAQ.
- Securing Debian Manual.

Iată un script simplu pentru a verifica permisiunile incorecte tipice ale fișierelor care pot fi scrise de oricine.

```
# find / -perm 777 -a \! -type s -a \! -type l -a \! \( -type d -a -perm 1777 \)
```



Atenție
Deoarece pachetul `debsums` utilizează sumele de control MD5 stocate local, acesta nu poate fi considerat pe deplin fiabil ca instrument de auditare a securității sistemului împotriva atacurilor rău intenționate.

pachet	popcon(popularitate) și dimensiune	dimensiune	descriere
logcheck	V:4.4, I:5.4	120	demon pentru a trimite prin poșta electronică anomaliile din fișierele jurnal ale sistemului către administrator
debsums	V:4, I:29	108	ustensilă pentru verificarea fișierelor pachetelor instalate în raport cu sumele de control MD5
chkrootkit	V:10, I:14	987	detector de rootkit
clamav	V:8, I:38	19877	ustensilă antivirus pentru Unix - interfață de linie de comandă
tiger	V:1.0, I:1.3	7800	raportează vulnerabilitățile de securitate ale sistemului
tripwire	V:1.1, I:1.4	5060	verificator de integritate a fișierelor și directoarelor
john	V:1.1, I:7.3	469	instrument activ de spargere a parolelor
aide	V:2.2, I:2.6	328	mediu avansat de detectare a intruziunilor - binar static
integrit	V:0.09, I:0.17	2759	program de verificare a integrității fișierelor
crack	V:0.06, I:0.55	151	program de ghicire a parolelor

Tabela 9.18: Lista instrumentelor pentru verificarea securității și integrității sistemului

9.6 Sfaturi pentru stocarea datelor

Pornirea sistemului ca sistem Linux live (consultați Secțiune 3.2.2 și Secțiune 3.2.3) vă facilitează reconfigurarea stocării datelor pe sistemul instalat.

Notă
Declarațiile referitoare la discul dur (HDD) se aplică și altor dispozitive de stocare, cum ar fi SSD / unitate flash USB / card de memorie / ... . Înlocuiți numele dispozitivelor din exemple, cum ar fi `/dev/sda`, cu numele dispozitivelor aplicabile `/dev/nvme0`, `/dev/mmcblk0`, ...

Este posibil să fie necesar să demontați cu umount(8) unele dispozitive manual din linia de comandă înainte de a le utiliza, dacă acestea sunt montate automat de sistemul grafic de birou.

9.6.1 Utilizarea spațiului pe disc

Utilizarea spațiului pe disc poate fi evaluată cu ajutorul programelor furnizate de pachetele `mount`, `coreutils` și `xdu`:

- mount(8) raportează toate sistemele de fișiere montate (= discuri).
- df(1) raportează utilizarea spațiului pe disc pentru sistemul de fișiere.
- du(1) raportează utilizarea spațiului pe disc pentru arborele de directoare.

Indicație
Puteți introduce rezultatul comenzii du(8) în xdu(1) pentru a obține o prezentare grafică și interactivă cu „`du -k . |xdu`”, „`sudo du -k -x / |xdu`” etc.

9.6.2 Configurarea partițiilor de disc

Pentru configurarea partiției discului, deși fdisk(8) a fost considerat standard, parted(8) merită o anumită atenție. „Date de partiționare a discului”, „tabel de partiții”, „hartă de partiții” și „etichetă de disc” sunt toate sinonime.

PC-urile mai vechi utilizează schema clasică Master Boot Record (MBR) pentru a stoca datele de partiționare a discului în primul sector, adică sectorul 0 LBA (512 octeți).

PC-urile recente cu Unified Extensible Firmware Interface (UEFI), inclusiv Mac-urile bazate pe Intel, utilizează schema GUID Partition Table (GPT) pentru a stoca datele de partiționare a discului care nu se află în primul sector.

Deși fdisk(8) a fost standardul pentru instrumentul de partiționare a discurilor, parted(8) îl înlocuiește.

pachet	popcon(popularitate)	dimensiune	descriere
util-linux	V:913, I:1000	4826	utilitare de sistem diverse, inclusiv fdisk(8) și cfdisk(8)
parted	V:449, I:569	126	programul GNU Parted pentru redimensionarea partițiilor de disc
gparted	V:12, I:83	2313	editor de partiții GNOME bazat pe libparted
gdisk	V:19, I:253	967	editor de partiții pentru discul hibrid GPT/MBR
kpartx	V:15, I:24	78	program pentru crearea hărților dispozitivelor pentru partiții

Tabela 9.19: Lista pachetelor de gestionare a partițiilor de disc

Atenție
Deși parted(8) pretinde că creează și redimensionează sistemul de fișiere, este mai sigur să faceți astfel de lucruri folosind instrumente specializate bine întreținute, cum ar fi mkfs (8) (mkfs.msdos(8), mkfs.ext2(8), mkfs.ext3(8), mkfs.ext4(8), …) și resize2fs(8).

Notă
Pentru a comuta între GPT și MBR, trebuie să ștergeți direct primele câteva blocuri din conținutul discului (consultați Secțiune 9.8.6) și să utilizați „`parted /dev/sdx mklabel gpt`” sau „`parted /dev/sdx mklabel msdos`” pentru a-l pune în loc. Rețineți că „`msdos`” este utilizat aici pentru MBR.

9.6.3 Accesarea partiției folosind UUID

Deși reconfigurarea partiției sau ordinea de activare a mediilor de stocare amovibile poate genera denumiri diferite pentru partiții, puteți accesa aceste partiții în mod consecvent. Acest lucru este util și în cazul în care aveți mai multe discuri, iar BIOS/UEFI nu le atribuie denumiri consecvente.

- mount(8) cu opțiunea „`-U`” poate monta un dispozitiv de blocuri folosind UUID, în loc să folosească numele fișierului, cum ar fi „`/dev/sda3`”.
- „`/etc/fstab`” (a se vedea fstab(5)) poate utiliza UUID.
- Încărcătoarele de pornire (Secțiune 3.1.2) pot utiliza de asemenea UUID.

Indicație
Puteți verifica UUID-ul unui dispozitiv special de blocuri cu blkid(8).
De asemenea, puteți verifica UUID-ul și alte informații cu „`lsblk -f`”.

9.6.4 LVM2

LVM2 este un manager de volume logice pentru nucleul Linux. Cu LVM2, partițiile de disc pot fi create pe volume logice în loc de discuri dure fizice.

LVM necesită următoarele.

- suport pentru «device-mapper» în nucleul Linux (implicit pentru nucleele Debian)
- biblioteca de suport pentru cartografierea dispozitivelor în spațiul utilizatorului (pachetul `libdevmapper*`)
- instrumentele LVM2 din spațiul utilizatorului (pachetul `lvm2`)

Vă rugăm să începeți să învățați LVM2 din următoarele pagini de manual.

- lvm(8): noțiuni de bază despre mecanismul LVM2 (lista tuturor comenzilor LVM2)
- lvm.conf(5): fișierul de configurare pentru LVM2
- lvs(8): raportează informații despre volumele logice
- vgs(8): raportează informații despre grupurile de volume
- pvs(8): raportează informații despre volumele fizice

9.6.5 Configurarea sistemului de fișiere

Pentru sistemul de fișiere ext4, pachetul `e2fsprogs` oferă următoarele.

- mkfs.ext4(8) pentru a crea un nou sistem de fișiere ext4
- fsck.ext4(8) pentru a verifica și repara sistemul de fișiere ext4 existent
- tune2fs(8) pentru a configura superblocul sistemului de fișiere ext4
- debugfs(8) pentru a depana interactiv sistemul de fișiere ext4; acesta are comanda `undel` pentru a recupera fișierele șterse).

Comenzile mkfs(8) și fsck(8) sunt furnizate de pachetul `e2fsprogs` ca interfețe pentru diverse programe dependente de sistemul de fișiere (`mkfs.fstype` și `fsck.fstype`). Pentru sistemul de fișiere ext4, acestea sunt mkfs.ext4(8) și fsck.ext4(8) (acestea sunt legate simbolic de mke2fs(8) și e2fsck(8)).

Comenzi similare sunt disponibile pentru fiecare sistem de fișiere acceptat de Linux.

Indicație
Sistemul de fișiere Ext4 este sistemul de fișiere implicit pentru sistemul Linux și se recomandă insistent utilizarea acestuia, cu excepția cazului în care aveți motive specifice pentru a nu o face.
Starea Btrfs poate fi găsită la Debian wiki on btrfs și kernel.org wiki on btrfs. Se preconizează că acesta va fi următorul sistem de fișiere implicit după sistemul de fișiere ext4.
Unele instrumente permit accesul la sistemul de fișiere fără suportulasigurat de nucleul Linux (a se vedea Secțiune 9.8.2).

9.6.6 Crearea sistemului de fișiere și verificarea integrității

Comanda mkfs(8) creează sistemul de fișiere într-un sistem Linux. Comanda fsck(8) asigură verificarea integrității și repararea sistemului de fișiere într-un sistem Linux.

Debian nu mai execută implicit comanda `fsck` după crearea sistemului de fișiere.



Atenție
În general, nu este sigur să rulați `fsck` pe **sisteme de fișiere montate**.

---

pachet	popcon(popularitate) și	dimensiune	descriere
e2fsprogs	V:821, I:997	1560	ustensie pentru sistemele de fișiere ext2/ext3/ext4
btrfs-progs	V:52, I:79	5255	ustensile pentru sistemul de fișiere Btrfs
reiserfsprogs	V:11, I:21	473	ustensile pentru sistemul de fișiere Reiserfs
zfsutils-linux	V:32, I:33	1905	ustensile pentru sistemul de fișiere OpenZFS
dosfstools	V:265, I:568	310	ustensile pentru sistemul de fișiere FAT (Microsoft: MS-DOS, Windows)
exfatprogs	V:36, I:466	580	ustensile pentru sistemul de fișiere exFAT menținut de Samsung.
exfat-fuse	V:2, I:40	73	citește/scrie sistemul de fișiere exFAT controlor (Microsoft) pentru FUSE.
xfsprogs	V:42, I:84	4651	ustensile pentru sistemul de fișiere XFS (SGI: IRIX)
ntfs-3g	V:208, I:514	1507	citește/scrie sistemul de fișiere NTFS (Microsoft: Windows NT, …) controlor pentru FUSE.
jfsutils	V:0.4, I:6.7	1104	ustensile pentru sistemul de fișiere JFS. (IBM: AIX, OS/2)
xorriso	V:15, I:65	378	ustensile pentru sistemul de fișiere ISO-9660 și scrierea CD-urilor/DVD-urilor din libburnia
wodim	V:7, I:83	898	instrument de scriere CD/DVD din linia de comandă din pachetul cdrkit
genisoimage	V:17, I:153	1567	instrument pentru sistemul de fișiere ISO-9660 din linia de comandă din pachetul cdrkit
reiser4progs	V:0.1, I:1.1	1367	ustensile pentru sistemul de fișiere Reiser4
hfsprogs	V:0.3, I:3.1	394	ustensile pentru sistemul de fișiere HFS și HFS Plus (Apple: Mac OS)
zerofree	V:7, I:116	30	program pentru a șterge blocurile libere din sistemele de fișiere ext2/3/4

Tabela 9.20: Lista pachetelor de gestionare a sistemului de fișiere

Indicație
Puteți rula comanda fsck(8) în siguranță pe toate sistemele de fișiere, inclusiv pe sistemul de fișiere rădăcină la repornire, definind „`enable_periodic_fsck`” în „`/etc/mke2fs.conf`” și numărul maxim de montări la 0 folosind „`tune2fs -c0 /dev/`*`partition_name`*”. Consultați mke2fs.conf(5) și tune2fs(8).
Verificați fișierele din „`/var/log/fsck/`” pentru rezultatul comenzii fsck(8) executată din scriptul de pornire.

9.6.7 Optimizarea sistemului de fișiere prin opțiuni de montare

Configurația statică de bază a sistemului de fișiere este dată de „`/etc/fstab`”. De exemplu,

```
«file system»                    «mount point» «type» «options»    «dump» «pass»
proc                                    /proc proc   defaults          0 0
UUID=709cbe4c-80c1-56db-8ab1-dbce3146d2f7 /     ext4   errors=remount-ro 0 1
UUID=817bae6b-45d2-5aca-4d2a-1267ab46ac23 none  swap   sw                0 0
/dev/scd0                        /media/cdrom0  udf,iso9660 user,noauto  0 0
```

Indicație
UUID (vedeți Secțiune 9.6.3) poate fi utilizat pentru a identifica un dispozitiv de blocuri în locul numelor normale ale dispozitivelor de bloc, cum ar fi „`/dev/sda1`”, „`/dev/sda2`”, …

Începând cu Linux 2.6.30, nucleul utilizează în mod implicit comportamentul oferit de opțiunea „`relatime`”.

Consultați fstab(5) și mount(8).

9.6.8 Optimizarea sistemului de fișiere prin super-bloc

Caracteristicile unui sistem de fișiere pot fi optimizate prin intermediul super-blocului său, utilizând comanda tune2fs(8).

- Executarea comenzii „`sudo tune2fs -l /dev/sda1`” afișează conținutul super-blocului sistemului de fișiere pe „`/dev/sda1`”.
- Executarea comenzii „`sudo tune2fs -c 50 /dev/sda1`” modifică frecvența verificărilor sistemului de fișiere (executarea comenzii „`fsck`” în timpul pornirii) la fiecare 50 de porniri pe „`/dev/sda1`”.
- Executarea comenzii „`sudo tune2fs -j /dev/sda1`” adaugă capacitatea de jurnalizare la sistemul de fișiere, adică conversia sistemului de fișiere de la ext2 la ext3 pe „`/dev/sda1`”; (efectuați această operațiune pe sistemul de fișiere nemontat).
- Executarea comenzii „`sudo tune2fs -O extents,uninit_bg,dir_index /dev/sda1 && fsck -pf /dev/sd`” convertește partiția din ext3 în ext4 pe „`/dev/sda1`”; (faceți acest lucru pe sistemul de fișiere nemontat).

Indicație
În ciuda numelui său, tune2fs(8) funcționează nu numai pe sistemul de fișiere ext2, ci și pe sistemele de fișiere ext3 și ext4.

9.6.9 Optimizarea discului dur



Avertisment
Vă rugăm să verificați hardware-ul și să citiți pagina de manual a hdparm(8) înainte de a modifica configurația discului dur, deoarece acest lucru poate fi destul de periculos pentru integritatea datelor.

Puteți testa viteza de acces a unui disc dur, de exemplu „`/dev/sda`”, cu „`hdparm -tT /dev/sda`”. Pentru unele discuri dure conectate cu (E)IDE, puteți accelera viteza cu „`hdparm -q -c3 -d1 -u1 -m16 /dev/sda`” activând „(E) IDE 32-bit I/O support”, activând „fanionul using_dma”, activând „fanionul interrupt-unmask” și definind „multiple 16 sector I/O” (periculos!).

Puteți testa funcția de memorie cache de scriere a unui disc dur, de exemplu „`/dev/sda`”, cu „`hdparm -W /dev/sda`”. Puteți dezactiva funcția de memorie cache de scriere cu „`hdparm -W 0 /dev/sda`”.

Este posibil să puteți citi CD-ROM-urile preconfigurate defectuos pe unitățile CD-ROM moderne de mare viteză, încetinind-o cu „`setcd -x 2`”.

9.6.10 Optimizarea discului cu stare solidă

Discul/unitatea cu stare solidă („Solid State Drive”: SSD) este detectată automat acum.

Reduceți accesările inutile la disc pentru a preveni uzura discului prin montarea „`tmpfs`” în ruta de date volatile din `/etc/fstab`.

9.6.11 Utilizarea SMART pentru a prezice defectarea discului dur

Puteți monitoriza și înregistra discul dvs. dur compatibil cu SMART cu ajutorul demonului smartd(8).

1. Activați funcția SMART în BIOS.
2. Instalați pachetul `smartmontools`.
3. Identificați unitățile de disc dur listându-le cu df(1).
   - Să presupunem că unitatea de disc dur care trebuie monitorizată este „`/dev/sda`”.
4. Verificați rezultatul comenzii „`smartctl -a /dev/sda`” pentru a vedea dacă funcția SMART este activată.
   - Dacă nu, activați-o cu „`smartctl -s on -a /dev/sda`”.
5. Activați demonul smartd(8) pentru a rula astfel.
   - decomentați „`start_smartd=yes`” din fișierul „`/etc/default/smartmontools`”.
   - reporniți demonul smartd(8) cu comanda „`sudo systemctl restart smartmontools`”.

Indicație
Demonul smartd(8) poate fi personalizat cu ajutorul fișierului `/etc/smartd.conf`, inclusiv modul în care se pot primi notificări de avertizare.

9.6.12 Specificați directorul de stocare temporară prin $TMPDIR

Aplicațiile creează fișiere temporare în mod normal în directorul de stocare temporară „`/tmp`”. Dacă „`/tmp`” nu oferă suficient spațiu, puteți specifica un astfel de director de stocare temporară prin variabila `$TMPDIR` pentru programele care o acceptă.

9.6.13 Extinderea spațiului de stocare utilizabil prin LVM

Partițiile create cu ajutorul Managerul de volume logice („Logical Volume Manager”: LVM) (funcționalitate Linux) în momentul instalării pot fi redimensionate cu ușurință prin concatenarea extinderilor pe acestea sau prin trunchierea extinderilor de pe acestea pe mai multe dispozitive de stocare, fără a fi necesară o reconfigurare majoră a sistemului.

9.6.14 Extinderea spațiului de stocare utilizabil prin montarea unei alte partiții

Dacă aveți o partiție goală (de exemplu, „/dev/sdx”), o puteți formata cu mkfs.ext4(1) și o puteți monta mount (8) într-un director în care aveți nevoie de mai mult spațiu; (trebuie să copiați conținutul original al datelor).

```
$ sudo mv work-dir old-dir
$ sudo mkfs.ext4 /dev/sdx
$ sudo mount -t ext4 /dev/sdx work-dir
$ sudo cp -a old-dir/* work-dir
$ sudo rm -rf old-dir
```

Indicație
Alternativ, puteți monta un fișier imagine disc gol (consultați Secțiune 9.7.5) ca dispozitiv loop (consultați Secțiune 9.7.3). Utilizarea efectivă a discului crește odată cu datele efectiv stocate.

9.6.15 Extinderea spațiului de stocare utilizabil prin montarea unui alt director

Dacă aveți un director gol (de exemplu, „/path/to/emp-dir”) pe o altă partiție cu spațiu utilizabil, îl puteți monta (8) cu opțiunea „--bind” într-un director (de exemplu, „work-dir”) unde aveți nevoie de mai mult spațiu.

```
$ sudo mount --bind /path/to/emp-dir work-dir
```

9.6.16 Extinderea spațiului de stocare utilizabil prin montarea suprapusă a unui alt director

Dacă aveți spațiu utilizabil într-o altă partiție (de exemplu, „/path/to/empty” și „/path/to/work”), puteți crea un director în aceasta și îl puteți stivui pe un director vechi (de exemplu, „/path/to/old ”) unde aveți nevoie de spațiu, utilizând OverlayFS pentru nucleul Linux 3.18 sau mai recent (Debian Stretch 9.0 sau mai recent).

```
$ sudo mount -t overlay overlay \
  -olowerdir=/path/to/old-dir,upperdir=/path/to/empty,workdir=/path/to/work
```

Aici, „/path/to/empty” și „/path/to/work” ar trebui să se afle pe partiția cu acces RW pentru a scrie pe „/path/to/old

9.6.17 Extinderea spațiului de stocare utilizabil folosind legături simbolice



Atenție
Aceasta este o metodă învechită. Unele programe software pot să nu funcționeze bine cu o „legătură simbolică către un director”. În schimb, utilizați metodele de „montare” descrise mai sus.

Dacă aveți un director gol (de exemplu, „/path/to/emp-dir”) într-o altă partiție cu spațiu utilizabil, puteți crea o legătură simbolică către director cu ln(8).

```
$ sudo mv work-dir old-dir
$ sudo mkdir -p /path/to/emp-dir
$ sudo ln -sf /path/to/emp-dir work-dir
$ sudo cp -a old-dir/* work-dir
$ sudo rm -rf old-dir
```



Avertisment
Nu utilizați o „legătură simbolică către un director” pentru directoarele gestionate de sistem, cum ar fi „`/opt`”. O astfel de legătură simbolică poate fi suprascrisă atunci când sistemul este actualizat.

9.7 Imaginea discului

Aici discutăm despre manipulările imaginii discului.

9.7.1 Crearea fișierului imagine de disc

Fișierul imagine disc, „`disk.img`”, al unui dispozitiv nemontat, de exemplu, al doilea disc SCSI sau serial ATA „`/dev/sdb`”, poate fi creat folosind unul dintre următoarele instrumente.

```
# dd if=/dev/sdb of=disk.img; sync
```

```
# cp /dev/sdb disk.img ; sync
```

```
# cat /dev/sdb > disk.img ; sync
```

Imaginea discului master boot record (MBR) (a se vedea Secțiune 9.6.2) al unui PC tradițional, care se află în primul sector al discului IDE primar, poate fi creată folosind dd(1) după cum urmează.

```
# dd if=/dev/sda of=mbr.img bs=512 count=1
# dd if=/dev/sda of=mbr-nopart.img bs=446 count=1
# dd if=/dev/sda of=mbr-part.img skip=446 bs=1 count=66
```

- ”`mbr.img`”: MBR-ul cu tabelul de partiții
- ”`mbr-nopart.img`”: MBR-ul fără tabelul de partiții
- ”`mbr-part.img`”: Doar tabelul de partiții al MBR

Dacă creați o imagine a unei partiții a discului original, înlocuiți „`/dev/sda`” cu „`/dev/sda1`” etc.

9.7.2 Scrierea direct pe disc

Fișierul imagine disc, „`disk.img`”, poate fi scris pe un dispozitiv nemontat, de exemplu, pe a doua unitate SCSI „`/dev/sdb`” cu dimensiunea corespunzătoare, utilizând una dintre următoarele comenzi.

```
# dd if=disk.img of=/dev/sdb ; sync
```

```
# cp disk.img /dev/sdb ; sync
```

```
# cat disk.img >/dev/sdb ; sync
```

În mod similar, fișierul imagine al partiției de disc, „`partition.img`”, poate fi scris pe o partiție nemontată, de exemplu, prima partiție a celui de-al doilea disc SCSI „`/dev/sdb1`” cu dimensiunea corespunzătoare, prin următoarea comandă.

```
# dd if=partition.img of=/dev/sdb1 ; sync
```

9.7.3 Montarea fișierului imagine disc

Imaginea de disc „`partition.img`” care conține o singură imagine de partiție poate fi montată și demontată utilizând dispozitivul de buclă loop device după cum urmează.

```
# losetup --show -f partition.img
/dev/loop0
# mkdir -p /mnt/loop0
# mount -t auto /dev/loop0 /mnt/loop0
...hack...hack...hack
# umount /dev/loop0
# losetup -d /dev/loop0
```

Acest lucru poate fi simplificat după cum urmează.

```
# mkdir -p /mnt/loop0
# mount -t auto -o loop partition.img /mnt/loop0
...hack...hack...hack
# umount partition.img
```

Fiecare partiție a imaginii discului „`disk.img`” care conține mai multe partiții poate fi montată utilizând dispozitivul de buclă loop device.

```
# losetup --show -f -P disk.img
/dev/loop0
# ls -l /dev/loop0*
brw-rw---- 1 root disk   7,  0 Apr  2 22:51 /dev/loop0
brw-rw---- 1 root disk 259, 12 Apr  2 22:51 /dev/loop0p1
brw-rw---- 1 root disk 259, 13 Apr  2 22:51 /dev/loop0p14
brw-rw---- 1 root disk 259, 14 Apr  2 22:51 /dev/loop0p15
# fdisk -l /dev/loop0
Disk /dev/loop0: 2 GiB, 2147483648 bytes, 4194304 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 6A1D9E28-C48C-2144-91F7-968B3CBC9BD1

Device         Start     End Sectors  Size Type
/dev/loop0p1  262144 4192255 3930112  1.9G Linux root (x86-64)
/dev/loop0p14   2048    8191    6144    3M BIOS boot
/dev/loop0p15   8192  262143  253952  124M EFI System

Partition table entries are not in disk order.
# mkdir -p /mnt/loop0p1
# mkdir -p /mnt/loop0p15
# mount -t auto /dev/loop0p1 /mnt/loop0p1
# mount -t auto /dev/loop0p15 /mnt/loop0p15
# mount |grep loop
/dev/loop0p1 on /mnt/loop0p1 type ext4 (rw,relatime)
/dev/loop0p15 on /mnt/loop0p15 type vfat (rw,relatime,fmask=0002,dmask=0002,allow_utime ↩
    =0020,codepage=437,iocharset=ascii,shortname=mixed,utf8,errors=remount-ro)
...hack...hack...hack
# umount /dev/loop0p1
# umount /dev/loop0p15
# losetup -d /dev/loop0
```

Alternativ, efecte similare pot fi obținute utilizând dispozitivele device mapper create de kpartx(8) din pachetul `kpartx`, după cum urmează.

```
# kpartx -a -v disk.img
add map loop0p1 (253:0): 0 3930112 linear 7:0 262144
```

```
add map loop0p14 (253:1): 0 6144 linear 7:0 2048
add map loop0p15 (253:2): 0 253952 linear 7:0 8192
# fdisk -l /dev/loop0
Disk /dev/loop0: 2 GiB, 2147483648 bytes, 4194304 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 6A1D9E28-C48C-2144-91F7-968B3CBC9BD1

Device         Start     End Sectors  Size Type
/dev/loop0p1  262144 4192255 3930112  1.9G Linux root (x86-64)
/dev/loop0p14   2048    8191    6144    3M BIOS boot
/dev/loop0p15   8192  262143  253952  124M EFI System

Partition table entries are not in disk order.
# ls -l /dev/mapper/
total 0
crw------- 1 root root 10, 236 Apr  2 22:45 control
lrwxrwxrwx 1 root root       7 Apr  2 23:19 loop0p1 -> ../dm-0
lrwxrwxrwx 1 root root       7 Apr  2 23:19 loop0p14 -> ../dm-1
lrwxrwxrwx 1 root root       7 Apr  2 23:19 loop0p15 -> ../dm-2
# mkdir -p /mnt/loop0p1
# mkdir -p /mnt/loop0p15
# mount -t auto /dev/mapper/loop0p1 /mnt/loop0p1
# mount -t auto /dev/mapper/loop0p15 /mnt/loop0p15
# mount |grep loop
/dev/loop0p1 on /mnt/loop0p1 type ext4 (rw,relatime)
/dev/loop0p15 on /mnt/loop0p15 type vfat (rw,relatime,fmask=0002,dmask=0002,allow_utime ↩
    =0020,codepage=437,iocharset=ascii,shortname=mixed,utf8,errors=remount-ro)
...hack...hack...hack
# umount /dev/mapper/loop0p1
# umount /dev/mapper/loop0p15
# kpartx -d disk.img
```

9.7.4 Curățarea unui fișier imagine de disc

Un fișier imagine de disc, „`disk.img`”, poate fi curățat de toate fișierele eliminate într-o imagine curată și de dimensiune mai redusă „`new.img`” prin următoarele.

```
# mkdir old; mkdir new
# mount -t auto -o loop disk.img old
# dd bs=1 count=0 if=/dev/zero of=new.img seek=5G
# mount -t auto -o loop new.img new
# cd old
# cp -a --sparse=always ./ ../new/
# cd ..
# umount new.img
# umount disk.img
```

Dacă „`disk.img`” este în ext2, ext3 sau ext4, puteți utiliza și zerofree(8) din pachetul `zerofree` după cum urmează.

```
# losetup --show -f disk.img
/dev/loop0
# zerofree /dev/loop0
# cp --sparse=always disk.img new.img
# losetup -d /dev/loop0
```

9.7.5 Crearea fișierului imagine de disc gol

Imaginea de disc goală „`disk.img`”, care poate crește până la 5 Gio, poate fi creată folosind dd(1) după cum urmează.

```
$ dd bs=1 count=0 if=/dev/zero of=disk.img seek=5G
```

În loc să se utilizeze dd(1), aici se poate utiliza fallocate(8) specializat.

Puteți crea un sistem de fișiere ext4 pe această imagine de disc „`disk.img`” utilizând dispozitivul de buclă loop device după cum urmează.

```
# losetup --show -f disk.img
/dev/loop0
# mkfs.ext4 /dev/loop0
...hack...hack...hack
# losetup -d /dev/loop0
$ du  --apparent-size -h disk.img
5.0G  disk.img
$ du -h disk.img
83M disk.img
```

Pentru „`disk.img`”, dimensiunea fișierului este de 5,0 Gio, iar utilizarea efectivă a discului este de doar 83 Mio. Această discrepanță este posibilă deoarece ext4 poate stoca fișiere disperse.

Indicație
Utilizarea efectivă a discului de către fișierul dispers crește odată cu datele care sunt scrise pe acesta.

Folosind o operație similară pe dispozitivele create de dispozitivul de buclă loop device sau device mapper ca în Secțiune 9.7.3, puteți partiționa această imagine de disc „`disk.img`” folosind parted(8) sau fdisk(8) și puteți crea un sistem de fișiere pe acesta folosind mkfs.ext4(8), mkswap(8) etc.

9.7.6 Crearea fișierului imagine ISO9660

Indicație
Atât genisoimage(1) furnizat de cdrkit, cât și xorrisofs(1) furnizat de Libburnia au aceeași sintaxă de comandă, cu excepția numelui comenzii.

Fișierul imagine ISO9660 „`cd.iso`” din arborele directorului sursă „`source_directory`” poate fi creată folosind genisoimage(1) furnizat de cdrkit după cum urmează.

```
#  genisoimage -r -J -T -V volume_id -o cd.iso source_directory
```

În mod similar, fișierul imagine ISO9660 ce poate fi pornit, „`cdboot.iso`”, poate fi creat din `debian-installer`, ca arbore de directoare în „`source_directory`”, prin următoarele.

```
#  genisoimage -r -o cdboot.iso -V volume_id \
   -b isolinux/isolinux.bin -c isolinux/boot.cat \
   -no-emul-boot -boot-load-size 4 -boot-info-table source_directory
```

Aici încărcătorul de pornire Isolinux Isolinux boot loader (a se vedea Secțiune 3.1.2) este utilizat pentru pornire.

Puteți calcula valoarea md5sum și crea imaginea ISO9660 direct de pe dispozitivul CD-ROM, după cum urmează.

```
$ isoinfo -d -i /dev/cdrom
CD-ROM is in ISO 9660 format
...
Logical block size is: 2048
Volume size is: 23150592
...
# dd if=/dev/cdrom bs=2048 count=23150592 conv=notrunc,noerror | md5sum
# dd if=/dev/cdrom bs=2048 count=23150592 conv=notrunc,noerror > cd.iso
```



Avertisment
Pentru a obține rezultatul corect, trebuie să evitați cu atenție eroarea de citire anticipată a sistemului de fișiere ISO9660 din Linux, așa cum este descris mai sus.

9.7.7 Scrierea în mod direct pe CD/DVD-R/RW

Indicație
DVD-ul este doar un CD mare pentru wodim(1) furnizat de cdrkit și xorrecord(1) furnizat de Libburnia. Aceste comenzi au aceeași sintaxă, cu excepția numelui comenzii.

Puteți găsi un dispozitiv utilizabil după cum urmează.

```
# wodim --devices
```

Apoi, CD-ul gol este introdus în unitatea CD, iar fișierul imagine ISO9660, „`cd.iso`”, este scris pe acest dispozitiv, de exemplu „`/dev/sda`”, folosind wodim(1) după cum urmează.

```
# wodim -v -eject dev=/dev/sda cd.iso
```

Dacă se utilizează CD-RW în loc de CD-R, procedați în felul următor.

```
# wodim -v -eject blank=fast dev=/dev/sda cd.iso
```

Indicație
Dacă sistemul dvs. de birou montează automat CD-urile, demontați-l cu comanda „`sudo umount /dev/sda`” din consolă înainte de a utiliza wodim(1).

9.7.8 Montarea fișierului imagine ISO9660

Dacă „`cd.iso`” conține o imagine ISO9660, atunci următoarea comandă o montează manual în „`/cdrom`”.

```
# mount -t iso9660 -o ro,loop cd.iso /cdrom
```

Indicație
Sistemele grafice de birou moderne pot monta automat suporturi amovibile, cum ar fi CD-urile formatate ISO9660 (vedeți Secțiune 10.1.7).

9.8 Datele binare

Aici discutăm despre manipularea directă a datelor binare pe suporturile de stocare.

9.8.1 Vizualizarea și editarea datelor binare

Cea mai simplă metodă de vizualizare a datelor binare este utilizarea comenzii „`od -t x1`”.

pachet	popcon(popularitate) și dimensiune	descriere
`coreutils`	V:912, I:1000	17994 pachetul de bază care are od(1) pentru a descărca fișiere (HEX, ASCII, OCTAL, …)
`bsdmainutils`	V:4, I:121	17 pachet de ustensiler care are hd(1) pentru a descărca fișiere (HEX, ASCII, OCTAL, …)
`hexedit`	V:0.8, I:7.5	70 editor și vizor binar (HEX, ASCII)
`bless`	V:0.1, I:1.2	924 editor hexazecimal cu funcții avansate (GNOME)
`okteta`	V:1, I:11	1591 editor hexazecimal cu funcții avansate (KDE4)
`ncurses-hexedit`	V:0.1, I:1.1	130 editor și vizor binar (HEX, ASCII, EBCDIC)
`beav`	V:0.04, I:0.26	137 editor și vizor binar (HEX, ASCII, EBCDIC, OCTAL, …)

Tabela 9.21: Lista pachetelor care vizualizează și editează date binare

Indicație
HEX este utilizat ca acronim pentru formatul hexazecimal cu baza numerică 16. OCTAL este pentru formatul octal cu baza numerică 8. ASCII este pentru Codul standard american pentru schimbul de informații, adică codul normal al textului în limba engleză. EBCDIC este pentru Extended Binary Coded Decimal Interchange Code utilizat pe sistemele de operare IBM mainframe.

9.8.2 Manipularea fișierelor fără montarea discului

Există instrumente pentru citirea și scrierea fișierelor fără montarea discului.

pachet	popcon(popularitate) și dimensiune	descriere
`mtools`	V:7, I:52	400 instrumente pentru fișiere MSDOS fără a le monta
`hfsutils`	V:0.1, I:2.5	178 instrumente pentru fișiere HFS și HFS+ fără a le monta

Tabela 9.22: Lista pachetelor pentru manipularea fișierelor fără montarea discului

9.8.3 Redundanța datelor

Sistemele software RAID oferite de nucleul Linux asigură redundanța datelor la nivelul sistemului de fișiere al nucleului pentru a obține un nivel ridicat de fiabilitate a stocării.

Există instrumente pentru a adăuga redundanță de date la fișiere la nivel de program de aplicație, pentru a obține niveluri ridicate de fiabilitate a stocării.

pachet	popcon(popularitate) și dimensiune		descriere
par2	V:9, I:116	330	set de volume de arhivă Parity (Parity Archive Volume Set), pentru verificarea și repararea fișierelor
dvdisaster	V:0.1, I:1.1	1422	protecție împotriva pierderii datelor/zgârieturilor/îmbătrânirii pentru suporturile CD/DVD
dvbackup	V:0.01, I:0.03	413	instrument de copie de rezervă utilizând camere video MiniDV (furnizând rsbep(1))

Tabela 9.23: Lista instrumentelor pentru adăugarea redundanței datelor la fișiere

pachet	popcon(popularitate) și dimensiune		descriere
testdisk	V:2, I:25	528	ustensile pentru scanarea partițiilor și recuperarea discurilor
magicrescue	V:0.2, I:1.7	257	ustensilă pentru recuperarea fișierelor prin căutarea de octeți magici
scalpel	V:0.2, I:2.2	89	economic, cu performanțe ridicate în prelucrarea fișierelor pentru recuperarea acestora
myrescue	V:0.2, I:1.9	81	recuperează date de pe discuri dure deteriorate
extundelete	V:0.5, I:7.3	152	instrument pentru recuperarea fișierelor șterse din sistemul de fișiere ext3/4
ext4magic	V:0.2, I:3.5	235	instrument pentru recuperarea fișierelor șterse din sistemul de fișiere ext3/4
ext3grep	V:0.2, I:1.8	299	instrument pentru recuperarea fișierelor șterse din sistemul de fișiere ext3
scrounge-ntfs	V:0.1, I:1.6	49	program de recuperare a datelor pentru sisteme de fișiere NTFS
gzrt	V:0.02, I:0.43	34	set de instrumente de recuperare gzip
sleuthkit	V:3, I:25	1119	instrumente pentru analiza tehnico-criminalistică. (Sleuthkit -- Kit de detectiv)
autopsy	V:0.1, I:1.1	1026	interfață grafică pentru SleuthKit
foremost	V:0.3, I:3.8	101	aplicație tehnico-criminalistică pentru recuperarea datelor
guymager	V:0.08, I:0.71	1049	instrument de procesare a imaginilor de fișiere tehnico-criminalistică bazat pe Qt
dcfldd	V:0.3, I:2.6	113	versiune îmbunătățită a dd pentru analiză tehnico-criminalistică și securitate

Tabela 9.24: Lista pachetelor pentru recuperarea fișierelor de date și analiza tehnico-criminalistică

9.8.4 Recuperarea fișierelor de date și analiza tehnico-criminalistică

Există instrumente pentru recuperarea fișierelor de date și analiza tehnico-criminalistică.

Indicație
Puteți recupera fișierele șterse din sistemul de fișiere ext2 folosind comenzile `list_deleted_inodes` și `undel` din debugfs(8) din pachetul `e2fsprogs`.

9.8.5 Împărțirea unui fișier mare în fișiere mici

Când un fișier este prea mare pentru a fi copiat ca un singur fișier, puteți copia conținutul acestuia după ce îl împărțiți în bucăți de, de exemplu, 2000 Mio și apoi puteți reuni aceste bucăți în fișierul original.

```
$ split -b 2000m large_file
$ cat x* >large_file
```



Atenție
Asigurați-vă că nu aveți fișiere care încep cu „x” pentru a evita conflictele de nume.

9.8.6 Ștergerea conținutului fișierului

Pentru a șterge conținutul unui fișier, cum ar fi un fișier jurnal, nu utilizați rm(1) pentru a șterge fișierul și apoi creați un fișier gol nou, deoarece fișierul poate fi accesat în intervalul dintre comenzi. Următoarea este metoda sigură de a șterge conținutul fișierului.

```
$ :>file_to_be_cleared
```

9.8.7 Fișiere fictive

Următoarele comenzi creează fișiere fictive sau goale.

```
$ dd if=/dev/zero    of=5kb.file bs=1k count=5
$ dd if=/dev/urandom of=7mb.file bs=1M count=7
$ touch zero.file
$ : > alwayszero.file
```

Ar trebui să găsiți următoarele fișiere.

- „`5kb.file`” ce conține 5 Ko de zerouri.
- „`7mb.file`” ce conține 7 Mo de date aleatorii.
- ”`zero.file`” poate fi un fișier de 0 octeți. Dacă a existat, `mtime` este actualizat, în timp ce conținutul și lungimea acestuia sunt păstrate.
- ”`alwayszero.file`” este întotdeauna un fișier de 0 octeți. Dacă exista, `mtime` este actualizat și conținutul său este reinițializat.

9.8.8 Ștergerea întregului disc dur

Există mai multe modalități de a șterge complet datele de pe un dispozitiv precum un disc dur, de exemplu, un stick de memorie USB la „/dev/sda”.



Atenție
Verificați mai întâi locația stick-ului de memorie USB cu mount(8) înainte de a executa comenzile de aici. Dispozitivul indicat de „/dev/sda” poate fi un disc dur SCSI sau un disc dur serial-ATA pe care se află întregul sistem.

Ștergeți tot conținutul discului punând datele la 0 cu următoarea comandă.

```
# dd if=/dev/zero of=/dev/sda
```

Ștergeți totul suprascriind cu date aleatorii, ca mai jos.

```
# dd if=/dev/urandom of=/dev/sda
```

Ștergeți totul suprascriind cu date aleatorii în mod foarte eficient, în felul următor.

```
# shred -v -n 1 /dev/sda
```

Alternativ, puteți utiliza badblocks(8) cu opțiunea `-t random`.

Deoarece dd(1) este disponibil din shell-ul multor CD-uri Linux care pot fi pornite, cum ar fi CD-ul de instalare Debian, puteți șterge complet sistemul instalat executând o comandă de ștergere de pe un astfel de suport pe discul dur al sistemului, de exemplu, „/dev/sda”, „/dev/sda” etc.

9.8.9 Ștergerea zonei neutilizate a unui disc dur

Zona neutilizată de pe un disc dur (sau stick de memorie USB), de exemplu „/dev/sdb1”, poate conține încă date șterse, deoarece acestea sunt doar deconectate de la sistemul de fișiere. Acestea pot fi șterse prin suprascriere.

```
# mount -t auto /dev/sdb1 /mnt/foo
# cd /mnt/foo
# dd if=/dev/zero of=junk
dd: writing to `junk': No space left on device
...
# sync
# umount /dev/sdb1
```



Avertisment
De obicei, acest lucru este suficient pentru stick-ul de memorie USB. Dar nu este perfect. Majoritatea părților din numele fișierelor șterse și atributele acestora pot fi ascunse și pot rămâne în sistemul de fișiere.

9.8.10 Recuperarea fișierelor șterse, dar încă deschise

Chiar dacă ați șters accidental un fișier, atâta timp cât acel fișier este încă utilizat de o aplicație (în modul citire sau scriere), este posibil să recuperați un astfel de fișier.

De exemplu, încercați următoarele

```
$ echo foo > bar
$ less bar
$ ps aux | grep ' less[ ]'
bozo    4775  0.0  0.0  92200   884 pts/8    S+   00:18   0:00 less bar
$ rm bar
$ ls -l /proc/4775/fd | grep bar
lr-x------ 1 bozo bozo 64 2008-05-09 00:19 4 -> /home/bozo/bar (deleted)
$ cat /proc/4775/fd/4 >bar
$ ls -l
-rw-r--r-- 1 bozo bozo 4 2008-05-09 00:25 bar
$ cat bar
foo
```

Executați pe un alt terminal (când aveți pachetul `lsof` instalat) următoarele.

```
$ ls -li bar
2228329 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:02 bar
$ lsof |grep bar|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/bar
$ rm bar
$ lsof |grep bar|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/bar (deleted)
$ cat /proc/4775/fd/4 >bar
$ ls -li bar
2228302 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:05 bar
$ cat bar
foo
```

9.8.11 Căutarea tuturor legăturilor dure

Fișierele cu legături dure pot fi identificate prin „`ls -li`”.

```
$ ls -li
total 0
2738405 -rw-r--r-- 1 root root 0 2008-09-15 20:21 bar
2738404 -rw-r--r-- 2 root root 0 2008-09-15 20:21 baz
2738404 -rw-r--r-- 2 root root 0 2008-09-15 20:21 foo
```

Atât „`baz`” cât și „`foo`” au un număr de legături de „2” (>1), ceea ce arată că au legături dure. Numărul lor de nod-i este comune „2738404”. Aceasta înseamnă că sunt același fișier legat dur. Dacă nu găsiți toate fișierele legate dur din întâmplare, le puteți căuta după nodul-i, de exemplu „2738404”, în felul următor.

```
# find /path/to/mount/point -xdev -inum 2738404
```

9.8.12 Consum invizibil de spațiu pe disc

Toate fișierele șterse, dar deschise, ocupă spațiu pe disc, deși nu sunt vizibile din du(1) normal. Acestea pot fi listate împreună cu dimensiunea lor folosind următoarea comandă.

```
# lsof -s -X / |grep deleted
```

9.9 Sfaturi pentru criptarea datelor

Având acces fizic la calculatorul dvs., oricine poate obține cu ușurință privilegii de root și acces la toate fișierele de pe calculatorul dvs. (consultați Secțiune 4.6.4). Aceasta înseamnă că sistemul de parole de autentificare nu poate proteja datele dvs. private și sensibile împotriva unui eventual furt al calculatorului. Pentru a face acest lucru, trebuie să implementați tehnologia de criptare a datelor. Deși GNU privacy guard (a se vedea Secțiune 10.3) poate cripta fișiere, acest lucru necesită eforturi din partea utilizatorului.

Dm-crypt facilitează criptarea automată a datelor prin intermediul modulelor native ale nucleului Linux, cu eforturi minime din partea utilizatorului, utilizând device-mapper.

pachet	popcon(popularitate) și dimensiune	descriere
`cryptsetup`	V:34, I:77	469 instrumente pentru dispozitivul de blocuri criptat (dm-crypt / LUKS)
`cryptmount`	V:1.8, I:2.3	236 instrumente pentru dispozitivul de blocuri criptat (dm-crypt / LUKS) cu accent pe montarea/demontarea de către utilizatorii normali
`fscrypt`	V:0.4, I:1.1	6844 instrumente pentru criptarea sistemelor de fișiere Linux (fscrypt)
`libpam-fscrypt`	V:0.31, I:0.34	6941 modul PAM pentru criptarea sistemului de fișiere Linux (fscrypt)

Tabela 9.25: Lista instrumentelor de criptare a datelor



Atenție
Criptarea datelor costă timp CPU etc. Datele criptate devin inaccesibile în cazul în care parola lor este pierdută. Vă rugăm să evaluați beneficiile și costurile.

Notă
Întregul sistem Debian poate fi instalat pe un disc criptat de debian-installer (lenny sau mai nou) folosind dm-crypt/LUKS și initramfs.

Indicație
Consultați Secțiune 10.3 pentru instrumentul utilitar de criptare a spațiului utilizatorului: GNU Privacy Guard.

9.9.1 Criptarea discurilor amovibile cu dm-crypt/LUKS

Puteți cripta conținutul dispozitivelor de mare volum amovibile, de exemplu un stick de memorie USB pe „`/dev/sdx`”, utilizând dm-crypt/LUKS. Pur și simplu îl formatați după cum urmează.

```
# fdisk /dev/sdx
... "n" "p" "1" "return" "return" "w"
# cryptsetup luksFormat /dev/sdx1
...
# cryptsetup open /dev/sdx1 secret
...
# ls -l /dev/mapper/
total 0
crw-rw---- 1 root root  10, 60 2021-10-04 18:44 control
lrwxrwxrwx 1 root root       7 2021-10-04 23:55 secret -> ../dm-0
# mkfs.vfat /dev/mapper/secret
```

```
...
# cryptsetup close secret
```

Apoi, acesta poate fi montat la fel ca unul normal pe „`/media/username/disk_label`”, cu excepția solicitării parolei (a se vedea Secțiune 10.1.7) în mediul grafic de birou modern utilizând pachetul `udisks2`. Diferența este că toate datele scrise pe acesta sunt criptate. Introducerea parolei poate fi automatizată folosind «keyring» (a se vedea Secțiune 10.3.6).

În mod alternativ, puteți formata suportul în alt tip de sistem de fișiere, de exemplu, ext4 cu „`mkfs.ext4 /dev/mapper/sdx`”. Dacă se utilizează în schimb btrfs, trebuie instalat pachetul `udisks2-btrfs`. Pentru aceste sisteme de fișiere, este posibil să fie necesară configurarea drepturilor de proprietate și a permisiunilor pentru fișiere.

9.9.2 Montarea discului criptat cu dm-crypt/LUKS

De exemplu, o partiție de disc criptată creată cu dm-crypt/LUKS pe „`/dev/sdc5`” de către programul de instalare Debian poate fi montată pe „`/mnt`” astfel:

```
$ sudo cryptsetup open /dev/sdc5 ninja --type luks
Enter passphrase for /dev/sdc5: ****
$ sudo lvm
lvm> lvscan
  inactive          '/dev/ninja-vg/root' [13.52 GiB] inherit
  inactive          '/dev/ninja-vg/swap_1' [640.00 MiB] inherit
  ACTIVE            '/dev/goofy/root' [180.00 GiB] inherit
  ACTIVE            '/dev/goofy/swap' [9.70 GiB] inherit
lvm> lvchange -a y /dev/ninja-vg/root
lvm> exit
  Exiting.
$ sudo mount /dev/ninja-vg/root /mnt
```

9.10 Nucleul

Debian distribuie Nucleul Linux modularizat ca pachete pentru arhitecturile acceptate.

Dacă citiți această documentație, probabil că nu trebuie să compilați singur nucleul Linux.

9.10.1 Parametrii nucleului

Multe caracteristici Linux sunt configurabile prin intermediul parametrilor nucleului, după cum urmează.

- Parametrii nucleului inițializați de încărcătorul de pornire (a se vedea Secțiune 3.1.2)
- Parametrii nucleului modificați de sysctl(8) în timpul execuției pentru cei accesibili prin sysfs (a se vedea Secțiune 1.2.12)
- Parametrii modulului stabiliți prin argumentele lui modprobe(8) atunci când un modul este activat (a se vedea Secțiune 9.7.3)

Consultați „The Linux kernel user’s and administrator’s guide » The kernel’s command-line parameters” pentru detalii.

9.10.2 Antetele nucleului

Majoritatea **programelor normale** nu au nevoie de antete ale nucleului și, de fapt, se pot întrerupe dacă le utilizați direct pentru compilare. Acestea ar trebui compilate pe baza antetelor din „`/usr/include/linux`” și „`/usr/include/asm`” furnizate de pachetul `libc6-dev` (creat din pachetul sursă `glibc`) pe sistemul Debian.

Notă
Pentru compilarea unor programe specifice nucleului, cum ar fi modulele nucleului din sursă externă și demonul automounter (`amd`), trebuie să includeți în linia de comandă ruta către antetele nucleului corespunzător, de exemplu „`-I/usr/src/linux-particular-version/include/`”.

9.10.3 Compilarea nucleului și a modulelor aferente

Debian are propria sa metodă de compilare a nucleului și a modulelor aferente.

pachet	popcon(popularitate) și dimensiune		descriere
`build-essential`	V:20, I:510	17	pachete esențiale pentru construirea pachetelor Debian: `make`, `gcc`, …
`bzip2`	V:163, I:973	113	instrumente de comprimare și decomprimare pentru fișiere bz2
`libncurses5-dev`	I:31	6	biblioteci pentru dezvoltatori și documente pentru ncurses
`git`	V:422, I:625	52512	git: sistem distribuit de control al reviziei utilizat de nucleul Linux
`fakeroot`	V:32, I:511	225	furnizează mediul fakeroot pentru construirea pachetului ca non-root
`initramfs-tools`	V:495, I:989	49	instrument pentru a construi un initramfs (specific Debian)
`dkms`	V:73, I:147	235	dynamic kernel module support (DKMS) (generic)
`module-assistant`	V:1, I:12	395	instrument de ajutor pentru crearea pachetului de module (specific Debian)
`devscripts`	V:5, I:31	2783	scripturi de ajutor pentru un întreținător de pachete Debian (specific Debian)

Tabela 9.26: Lista pachetelor cheie care trebuie instalate pentru recompilarea nucleului pe sistemul Debian

Dacă utilizați `initrd` în Secțiune 3.1.2, asigurați-vă că citiți informațiile aferente în initramfs-tools(8), update-initramfs(8), mkinitramfs(8) și initramfs.conf(5).



Avertisment
Nu puneți legături simbolice către directoarele din arborele sursă (de exemplu, „`/usr/src/linux*`”) din „`/usr/include/linux`” și „`/usr/include/asm`” atunci când compilați sursa nucleului Linux; (unele documente învechite sugerează acest lucru).

Notă
La compilarea celui mai recent nucleu Linux pe sistemul Debian `stable`, poate fi necesară utilizarea celor mai recente instrumente retro-adaptate din Debian `unstable`.
module-assistant(8) (sau forma sa scurtă `m-a`) ajută utilizatorii să construiască și să instaleze cu ușurință pachetul (pachetele) de module pentru unul sau mai multe nuclee personalizate.
dynamic kernel module support (DKMS) este un nou cadru independent de distribuție conceput pentru a permite actualizarea modulelor individuale ale nucleului fără schimbarea întregului nucleu. Acest lucru este utilizat pentru întreținerea modulelor din afara arborelui. De asemenea, acest lucru facilitează foarte mult reconstruirea modulelor pe măsură ce actualizați nucleele.

9.10.4 Compilarea sursei nucleului: Recomandarea echipei Debian Kernel

Pentru a construi pachete binare de nucleu personalizate din sursa de nucleu din amonte, trebuie să utilizați ținta „`deb-pkg`” furnizată de acesta.

```
$ sudo apt-get build-dep linux
$ cd /usr/src
$ wget https://mirrors.edge.kernel.org/pub/linux/kernel/v7.x/linux-7.2.8.tar.xz
$ tar --xz -xvf linux-7.2.8.tar.xz
$ cd linux-7.2.8
$ cp /boot/config-7.2.8.config
$ make menuconfig
 ...
$ make deb-pkg
```

Indicație
Pachetul linux-source-*version* furnizează sursa nucleului Linux cu plasturi(corecții) Debian ca „`/usr/src/linux-`*`version`*`.tar.bz2`”.

Pentru a construi pachete binare specifice din pachetul sursă al nucleului Debian, trebuie să utilizați obiectivele „`binary-arch_`*`architecture_featureset_flavour`*” din „`debian/rules.gen`”.

```
$ sudo apt-get build-dep linux
$ apt-get source linux
$ cd linux-7.*
$ fakeroot make -f debian/rules.gen binary-arch_i386_none_686
```

Mai multe informații:

- Debian Wiki: KernelFAQ
- Debian Wiki: DebianKernel
- Debian Linux Kernel Handbook: https://kernel-handbook.debian.net

9.10.5 Controlori hardware și firmware

Controlorul hardware este codul care rulează pe CPU-urile principale ale sistemului țintă. Majoritatea controlorilor hardware sunt disponibili acum ca software liber și sunt incluși în pachetele normale de nucleu Debian în zona `main`.

- Controlor GPU
  - Controlor GPU Intel (`main`)
  - Controlor GPU AMD/ATI (`main`)
  - Controlor GPU NVIDIA (`main` pentru controlorul nouveau și `non-free` pentru controlorii numai-binari asigurați de producător.)

Firmware-ul este codul sau datele încărcate pe dispozitivul atașat sistemului țintă (de exemplu, microcode CPU, codul de redare care rulează pe GPU sau FPGA / CPLD date, ...). Unele pachete firmware sunt disponibile ca software gratuit, dar multe pachete firmware nu sunt disponibile ca software gratuit deoarece conțin date binare fără sursă. Instalarea acestor date de firmware este esențială pentru ca dispozitivul să funcționeze conform așteptărilor.

- Pachetele de date firmware care conțin date încărcate în memoria volatilă a dispozitivului țintă.
  - `firmware-linux-free` (`main`)

  - `firmware-linux-nonfree` (`non-free-firmware`)
  - `firmware-linux-*` (`non-free-firmware`)
  - `*-firmware` (`non-free-firmware`)
  - `intel-microcode` (`non-free-firmware`)
  - `amd64-microcode` (`non-free-firmware`)
- Pachetele de programe de actualizare a firmware-ului care actualizează datele din memoria nevolatilă a dispozitivului țintă.
  - fwupd (`main`): Demon de actualizare a firmware-ului care descarcă datele firmware de la Linux Vendor Firmware Service.
  - gnome-firmware (`main`): interfață grafică GTK pentru «fwupd»
  - plasma-discover-backend-fwupd (`main`): interfață grafică Qt pentru «fwupd»

Vă rugăm să rețineți că accesul la pachetele `non-free-firmware` sunt furnizate de mediile de instalare oficiale pentru a oferi utilizatorului o experiență de instalare funcțională începând cu Debian 12 Bookworm. Zona `non-free-firmware` este descrisă în Secțiune 2.1.5.

Vă rugăm să rețineți, de asemenea, că datele firmware descărcate de fwupd de la Linux Vendor Firmware Service și încărcate în nucleul Linux care rulează pot fi `non-free`.

9.11 Sistem virtualizat

Utilizarea sistemului virtualizat ne permite să rulăm mai multe instanțe ale sistemului simultan pe un singur hardware.

Indicație
Consultați Debian wiki on SystemVirtualization.

9.11.1 Instrumente de virtualizare și emulare

Există mai multe platforme de virtualizare și instrumente de emulare.

- Pachete complete de emulare a hardware-ului „hardware emulation” precum cele instalate de metapachetul games-emulator
- În principal emulație la nivel de CPU cu unele emulări de dispozitive de In/Ieș, cum ar fi QEMU
- În principal virtualizare la nivel de CPU cu unele emulări de dispozitive de In/Ieș, cum ar fi Kernel-based Virtual Machine (KVM)
- Virtualizarea containerelor la nivel de sistem de operare cu suport la nivel de nucleu, cum ar fi LXC (Linux Containers), Docker, systemd-nspawn(1), ...
- Virtualizarea accesului la sistemul de fișiere la nivel de sistem de operare cu suprascrierea apelului bibliotecii de sistem pe ruta fișierului, cum ar fi chroot
- Virtualizarea accesului la sistemul de fișiere la nivel de sistem de operare cu suprascrierea apelului bibliotecii de sistem privind proprietatea asupra fișierului, cum ar fi fakeroot
- Emularea API a sistemului de operare, cum ar fi Wine
- Virtualizarea la nivel de interpret cu selectarea executabilului și suprascrierea bibliotecilor în timp de execuție, cum ar fi virtualenv și venv pentru Python

pachet	popcon(popularitate) și dimensiune	dimensiune instalată	descriere
coreutils	V:912, I:1000	17994	Instrumente GNU de bază care conțin chroot(8)
util-linux	V:913, I:1000	4826	utiități de sistem diverse care conțin unshare(1)
systemd-container	V:74, I:77	2906	instrumente systemd container/nspawn care conțin systemd-nspawn(1)
schroot	V:7.2, I:9.1	2222	instrument specializat pentru executarea pachetelor binare Debian în chroot
sbuild	V:3.9, I:6.9	155	instrument pentru construirea de pachete binare Debian din surse Debian
debootstrap	V:4, I:44	332	creează și pornește un sistem Debian de bază (scris în sh)
mmdebstrap	V:8, I:12	592	creează și pornește un sistem Debian (scris în Perl)
cdebootstrap	V:0.1, I:1.3	114	creează și pornește un sistem Debian (scris în C)
cloud-image-utils	V:1, I:14	52	utilități de gestionare a imaginilor în cloud
cloud-guest-utils	V:5, I:23	67	utilități pentru invitați în cloud
virt-manager	V:12, I:48	2309	Gestionar mașină virtuală: aplicație de mediu grafic de birou pentru gestionarea mașinilor virtuale
libvirt-clients	V:45, I:71	1168	programe pentru biblioteca libvirt
docker.io	V:50, I:53	80705	docker: mediu de execuție de containere Linux
podman	V:34, I:38	85864	podman: motor pentru a rula containere bazate pe OCI în Pods
podman-docker	V:2.8, I:3.4	271	motor pentru a rula containere bazate pe OCI în Pods - învăluitor pentru docker
incus	V:0.8, I:3.4	24	Incus: container de sistem și gestionar de mașini virtuale
games-emulator	I:0.19	21	games-emulator: emulatoare Debian pentru jocuri
bochs	V:0.04, I:0.59	8469	Bochs: emulator PC IA-32
qemu-system	I:20	69	QEMU: binare de emulare completă a sistemului
qemu-user	V:5.6, I:9.3	472213	QEMU: binare de emulare în modul utilizator
qemu-utils	V:14, I:114	12643	QEMU: utilități
qemu-system-x86	V:52, I:98	69257	KVM: virtualizare completă pe hardware x86 cu hardware-assisted virtualization
virtualbox	V:2.9, I:3.7	151079	VirtualBox: soluție de virtualizare x86 pe i386 și amd64
gnome-boxes	V:1.2, I:6.1	7074	Boxes: aplicație GNOME simplă pentru accesarea sistemelor virtuale
xen-tools	V:0.1, I:1.4	719	instrumente pentru gestionarea serverului virtual debian XEN
wine	V:13, I:54	204	Wine: implementarea API Windows (suita standard)
dosbox	V:1, I:11	2697	DOSBox: emulator x86 cu grafică Tandy/Herc/CGA/EGA/VGA/SVGA, sunet și DOS
lxc	V:10, I:13	1681	Linux containers user space tools
python3-venv	V:8, I:159	6	venv pentru crearea de medii virtuale python (biblioteca de sistem)
python3-virtualenv	V:7, I:37	374	virtualenv pentru crearea de medii python virtuale izolate
pipx	V:8, I:50	4412	pipx pentru instalarea aplicațiilor python în medii izolate

Tabela 9.27: Lista instrumentelor de virtualizare

Virtualizarea containerelor utilizează Secțiune 4.7.5 și este tehnologia de bază a Secțiune 7.7.

Iată câteva pachete care vă vor ajuta să configurați sistemul virtualizat.

Consultați articolul Wikipedia Compararea platformelor de mașini virtuale pentru o comparație detaliată a diferitelor soluții de virtualizare a platformelor.

9.11.2 Fluxul de lucru pentru virtualizare

Notă
Nucleele Debian implicite acceptă KVM de la `lenny`.

Fluxul de lucru tipic pentru virtualizare implică mai multe etape.

- Creați un sistem de fișiere gol (un arbore de fișiere sau o imagine de disc).
  - Arborele de fișiere poate fi creat prin „`mkdir -p /ruta/la/chroot`”.
  - Fișierul imagine de disc brut poate fi creat cu dd(1) (a se vedea Secțiune 9.7.1 și Secțiune 9.7.5).
  - qemu-img(1) poate fi utilizat pentru a crea și converti fișiere imagine de disc acceptate de QEMU.
  - Formatul de fișier brut și VMDK poate fi utilizat ca format comun între instrumentele de virtualizare.
- Montați imaginea discului cu mount(8) pe sistemul de fișiere (opțional).
  - Pentru fișierul imagine de disc brut, montați-l ca dispozitiv loop device sau dispozitiv device mapper (a se vedea Secțiune 9.7.3).
  - Pentru imaginile de disc acceptate de QEMU, montați-le ca network block device (vedeți Secțiune 9.11.3).
- Populați sistemul de fișiere țintă cu datele de sistem necesare.
  - Utilizarea unor programe precum `debootstrap` și `cdebootstrap` ajută la acest proces (a se vedea Secțiune 9.11.4).
  - Utilizați programe de instalare ale sistemelor de operare sub emulația sistemului complet.
- Rulați un program într-un mediu virtualizat.
  - chroot oferă un mediu virtualizat de bază suficient pentru a compila programe, a rula aplicații de consolă și a rula demoni în el.
  - QEMU oferă emulare CPU multi-platformă.
  - QEMU cu KVM oferă emulare completă a sistemului prin virtualizare asistată de hardware hardware-assisted virtualization.
  - VirtualBox oferă emulare completă a sistemului pe i386 și amd64 cu sau fără virtualizare asistată de hardware hardware-assisted virtualization.

9.11.3 Montarea fișierului imagine al discului virtual

Pentru fișierul imagine de disc brut, vedeți Secțiune 9.7.

Pentru alte fișiere imagine de disc virtual, puteți utiliza qemu-nbd(8) pentru a le exporta utilizând protocolul network block device (dispozitiv de blocuri din rețea) și a le monta utilizând modulul nucleului `nbd`.

qemu-nbd(8) acceptă formatele de disc acceptate de QEMU: raw, qcow2, qcow, vmdk, vdi, bochs, cow (user-mode Linux copy-on-write), parallels, dmg, cloop, vpc, vvfat (VFAT virtual) și host_device.

Dispozitivul network block device poate accepta partiții în același mod ca dispozitivul loop device (consultați Secțiune 9.7.3). Puteți monta prima partiție din „`disk.img`” după cum urmează.

```
# modprobe nbd max_part=16
# qemu-nbd -v -c /dev/nbd0 disk.img
...
# mkdir /mnt/part1
# mount /dev/nbd0p1 /mnt/part1
```

Indicație
Puteți exporta numai prima partiție din „`disk.img`” utilizând opțiunea „`-P 1`” la qemu-nbd(8).

9.11.4 Sistemul chroot

Dacă doriți să încercați un nou mediu Debian de la o consolă terminal, vă recomand să utilizați chroot. Acest lucru vă permite să rulați aplicații de consolă Debian `unstable` și `testing` fără riscurile obișnuite asociate și fără repornire. chroot(8) este cea mai de bază modalitate.



Atenție
Exemplele de mai jos presupun că atât sistemul părinte, cât și sistemul chroot au aceeași arhitectură CPU `amd64`.

Deși puteți crea manual un mediu chroot(8) utilizând debootstrap(1), acest lucru necesită eforturi netriviale.

Pachetul sbuild pentru a construi pachete Debian din sursă utilizează mediul chroot gestionat de pachetul schroot. Acesta vine cu scriptul ajutător sbuild-createchroot(1). Să învățăm cum funcționează executându-l după cum urmează.

```
$ sudo mkdir -p /srv/chroot
$ sudo sbuild-createchroot -v --include=eatmydata,ccache unstable /srv/chroot/unstable- ↩
    amd64-sbuild http://deb.debian.org/debian
 ...
```

Vedeți cum debootstrap(8) completează datele sistemului pentru mediul `unstable` sub „`/srv/chroot/unstable-amd64`” pentru un sistem de compilare minim.

Vă puteți conecta la acest mediu utilizând schroot(1).

```
$ sudo schroot -v -c chroot:unstable-amd64-sbuild
```

Vedeți cum este creat un shell de sistem care rulează în mediul `unstable`.

Notă
Fișierul „`/usr/sbin/policy-rc.d`” care iese întotdeauna cu 101 împiedică pornirea automată a programelor demon pe sistemul Debian. Consultați „`/usr/share/doc/init-system-helpers/README.policy-rc.d.gz`”.

Notă
Unele programe sub chroot pot necesita acces la mai multe fișiere din sistemul părinte pentru a funcționa decât oferă `sbuild-createchroot` conform celor de mai sus. De exemplu, „`/sys`”, „`/etc/passwd`”, „`/etc/group`”, „`/var/run/utmp`”, „`/var/log/wtmp`” etc. pot necesita să fie montatecu opțiunea „--bind” sau copiate.

Indicație
Pachetul `sbuild` ajută la construirea unui sistem chroot și construiește un pachet în chroot folosind `schroot` ca sistem de gestionare și manipulare a datelor. Este un sistem ideal pentru a verifica dependențele de construcție. Vedeți mai multe despre sbuild la Debian wiki și sbuild exemplu de configurare în „Guide for Debian Maintainers”.

Indicație
Comanda systemd-nspawn(1) ajută la rularea unei comenzi sau a unui sistem de operare într-un container de dimensiuni reduse în moduri similare cu `chroot`. Este mai puternic deoarece utilizează spații de nume pentru a virtualiza complet arborele de procese, IPC, numele de gazdă, numele de domeniu și, opțional, bazele de date de rețea și de utilizator. Consultați systemd-nspawn.

9.11.5 Mai multe sisteme de medii de birou

Dacă doriți să încercați un nou mediu grafic de birou al oricărui sistem de operare, vă recomand să utilizați QEMU sau KVM pe un sistem Debian `stable` pentru a rula mai multe sisteme de mediu de birou în siguranță utilizând virtualizare. Acestea vă permit să rulați orice aplicații de mediu de birou, inclusiv cele din Debian `unstable` și `testing`, fără riscurile obișnuite asociate cu acestea și fără repornire.

Deoarece QEMU pur este foarte lent, se recomandă accelerarea acestuia cu KVM atunci când sistemul gazdă îl acceptă.

Gestionarul de mașini virtuale cunoscut și ca `virt-manager` este un instrument cu interfață grafică, convenabil pentru gestionarea mașinilor virtuale KVM prin libvirt.

Imaginea discului virtual „`virtdisk.qcow2`” care conține un sistem Debian pentru QEMU poate fi creată utilizând debian-installer: Small CDs după cum urmează.

```
$ wget https://cdimage.debian.org/debian-cd/5.0.3/amd64/iso-cd/debian-503-amd64-netinst.iso
$ qemu-img create -f qcow2 virtdisk.qcow2 5G
$ qemu -hda virtdisk.qcow2 -cdrom debian-503-amd64-netinst.iso -boot d -m 256
...
```

Indicație
Rularea altor distribuții GNU/Linux precum Ubuntu și Fedora sub virtualizare este o modalitate excelentă de a învăța sfaturi de configurare. Alte sisteme de operare proprietare pot fi rulate foarte bine și sub această virtualizare GNU/Linux.

Vedeți mai multe sfaturi la Debian wiki: SystemVirtualization.

Capitolul 10

Gestionarea datelor

Sunt descrise instrumente și sfaturi pentru gestionarea datelor binare și textuale în sistemul Debian.

10.1 Partajarea, copierea și arhivarea



Avertisment
Accesul necoordonat la scriere la dispozitive și fișiere accesate activ din mai multe procese nu trebuie efectuat pentru a evita condiția de concurență. Pentru a evita acest lucru, se pot utiliza mecanisme de blocare a fișierelor folosind flock(1).

Securitatea datelor și partajarea controlată a acestora au mai multe aspecte.

- Crearea arhivei de date
- Accesul la stocare de la distanță
- Duplicarea
- Urmărirea istoricului modificărilor
- Facilitarea schimbului de date
- Prevenirea accesului neautorizat la fișiere
- Detectarea modificării neautorizate a fișierelor

Acestea pot fi realizate prin utilizarea unei combinații de instrumente.

- Instrumente de arhivare și comprimare
- Instrumente de copiere și sincronizare
- Sisteme de fișiere în rețea
- Suporturi de stocare amovibile
- Shell-ul securizat
- Sistemul de autentificare
- Instrumente pentru sistemul de control al versiunilor
- Instrumente de criptare criptografică și sume de control

10.1.1 Instrumente de arhivare și comprimare

Iată un rezumat al instrumentelor de arhivare și comprimare disponibile în sistemul Debian.



Avertisment
Nu definiți variabila „$TAPE” decât dacă știți la ce să vă așteptați. Aceasta modifică comportamentul tar(1).

- Arhiva tar comprimată cu tar(1) utilizează extensia de fișier „`.tgz`” sau „`.tar.gz`”.
- Arhiva tar(1) comprimată cu xz utilizează extensia de fișier „`.txz`” sau „`.tar.xz`”.
- Metoda populară de comprimare în instrumentele FOSS, cum ar fi tar(1), a evoluat după cum urmează: `gzip` → `bzip2` → `xz`
- cp(1), scp(1) și tar(1) pot avea unele limitări pentru fișiere speciale. cpio(1) este cel mai versatil.
- cpio(1) este conceput pentru a fi utilizat împreună cu find(1) și alte comenzi și este potrivit pentru crearea de scripturi de copiere de rezervă, deoarece partea de selectare a fișierelor din script poate fi testată independent.
- Structura internă a fișierelor de date Libreoffice este „`.jar`”, care poate fi deschisă și cu ajutorul comenzii `unzip`.
- Instrumentul de arhivare multiplatformă de facto este `zip`. Utilizați-l ca «`zip -rX`» pentru a obține compatibilitatea maximă. Utilizați și opțiunea „`-s`”, dacă dimensiunea maximă a fișierului este importantă.

10.1.2 Instrumente de copiere și sincronizare

Iată un rezumat al instrumentelor simple de copiere și copie de rezervă disponibile în sistemul Debian.

Copierea fișierelor cu rsync(8) oferă funcții mai avansate decât alte programe.

- algoritm de transfer delta care trimite numai diferențele dintre fișierele sursă și fișierele existente în destinație
- algoritm de verificare rapidă (implicit) care caută fișiere care au suferit modificări în ceea ce privește dimensiunea sau data ultimei modificări
- opțiunile „`--exclude`” și „`--exclude-from`” similare cu tar(1)
- sintaxa „o bară oblică la sfârșitul directorului sursă” care evită crearea unui nivel suplimentar de director la destinație.

Indicație
Instrumentele sistemului de control al versiunilor (VCS) din Tabel 10.14 pot funcționa ca instrumente de copiere și sincronizare în mai multe direcții.

10.1.3 Expresii idiomatice pentru arhivă

Iată câteva modalități de arhivare și dezarhivare a întregului conținut al directorului „`./source`” folosind diferite instrumente.

GNU tar(1):

```
$ tar -cvJf archive.tar.xz ./source
$ tar -xvJf archive.tar.xz
```

Alternativ, prin următoarele.

pachet	popcon(popularitate)	dimensiune	extensie	comandă	comentariu
tar	V:907, I:1000	3101	.tar	tar(1)	arhivatorul standard (standardul de facto)
cpio	V:352, I:999	1201	.cpio	cpio(1)	arhivator în stil Unix System V, se utilizează cu find(1)
binutils	V:191, I:631	1131	.ar	ar(1)	arhivator pentru crearea de biblioteci statice
fastjar	V:1.0, I:9.5	182	.jar	fastjar(1)	arhivator pentru Java (precum zip)
pax	V:5.1, I:9.0	167	.pax	pax(1)	nou arhivator standard POSIX, compromis între tar și cpio
gzip	V:895, I:1000	262	.gz	gzip(1), zcat(1), …	instrumentul de comprimare GNU LZ77 (standardul de facto)
bzip2	V:163, I:973	113	.bz2	bzip2(1), bzcat(1), …	instrumentul de comprimare prin sortare de blocuri Burrows-Wheeler cu un raport de comprimare mai mare decât gzip(1) (mai lent decât gzip cu sintaxă similară)
lzma	V:0.6, I:9.6	349	.lzma	lzma(1)	instrument de comprimare LZMA cu o raport de comprimare mai mare decât gzip(1) (depreciat)
xz-utils	V:310, I:981	1551	.xz	xz(1), xzdec(1), …	instrument de comprimare XZ cu un raport de comprimare mai mare decât bzip2(1) (mai lent decât gzip, dar mai rapid decât bzip2; înlocuitor pentru instrumentul de comprimare LZMA)
zstd	V:334, I:830	2308	.zstd	zstd(1), zstdcat(1), …	instrument de comprimare rapidă fără pierderi Zstandard
p7zip	V:5, I:172	8	.7z	7zr(1), p7zip(1)	arhivator de fișiere 7-Zip cu raport de comprimare ridicat (comprimare LZMA)
p7zip-full	V:17, I:199	12	.7z	7z(1), 7za(1)	arhivator de fișiere 7-Zip cu raport de comprimare ridicat (comprimare LZMA și altele)
lzop	V:14, I:129	164	.lzo	lzop(1)	instrument de comprimare LZO cu viteză de comprimare și decomprimare mai mare decât gzip(1) (raport de comprimare mai mic decât gzip cu sintaxă similară)
zip	V:48, I:355	635	.zip	zip(1)	InfoZIP: instrument de arhivare și comprimare DOS
unzip	V:123, I:744	399	.zip	unzip(1)	InfoZIP: instrument pentru dezarhivare și decomprimare DOS

Tabela 10.1: Lista instrumentelor de arhivare și comprimare

pachet	popcon(popularitate)	dimensiune	instrument	funcție
coreutils	V:912, I:1000	17994	GNU cp	copierea locală a fișierelor și directoarelor („-a” pentru recursiv)
openssh-client	V:682, I:997	3611	scp	copierea de la distanță a fișierelor și directoarelor (client, „-r” pentru recursiv)
openssh-server	V:774, I:820	3615	sshd	copierea de la distanță a fișierelor și directoarelor (server la distanță)
rsync	V:180, I:539	967		sincronizare și copie de rezervă la distanță într-un singur sens
unison	V:3, I:12	14		sincronizare și copie de rezervă la distanță bidirecțională

Tabela 10.2: Lista instrumentelor de copiere și sincronizare

```
$ find ./source -xdev -print0 | tar -cvJf archive.tar.xz --null -T -
```

cpio(1):

```
$ find ./source -xdev -print0 | cpio -ov --null > archive.cpio; xz archive.cpio
$ zcat archive.cpio.xz | cpio -i
```

10.1.4 Expresii idiomatice pentru copiere

Iată câteva modalități de a copia întregul conținut al directorului „`./source`” folosind diferite instrumente.

- Copie locală: directorul „`./source`” → directorul „`/dest`”
- Copie la distanță: directorul „`./source`” de pe gazda locală → directorul „`/dest`” de pe gazda „`user@host.dom`”

rsync(8):

```
# cd ./source; rsync -aHAXSv . /dest
# cd ./source; rsync -aHAXSv . user@host.dom:/dest
```

Alternativ, puteți utiliza sintaxa „o bară oblică la sfârșitul directorului sursă”.

```
# rsync -aHAXSv ./source/ /dest
# rsync -aHAXSv ./source/ user@host.dom:/dest
```

Alternativ, prin următoarele.

```
# cd ./source; find . -print0 | rsync -aHAXSv0 --files-from=- . /dest
# cd ./source; find . -print0 | rsync -aHAXSv0 --files-from=- . user@host.dom:/dest
```

GNU cp(1) și openSSH scp(1):

```
# cd ./source; cp -a . /dest
# cd ./source; scp -pr . user@host.dom:/dest
```

GNU tar(1):

```
# (cd ./source && tar cf - . ) | (cd /dest && tar xvfp - )
# (cd ./source && tar cf - . ) | ssh user@host.dom '(cd /dest && tar xvfp - )'
```

cpio(1):

```
# cd ./source; find . -print0 | cpio -pvdm --null --sparse /dest
```

Puteți înlocui „`.`” cu „`foo`” pentru toate exemplele care conțin „`.`” pentru a copia fișiere din directorul „`./source/foo`” în directorul „`/dest/foo`”.

Puteți înlocui „`.`” cu ruta absolută „`/ruta/la/source/foo`” pentru toate exemplele care conțin „`.`” pentru a elimina „`cd ./source;`”. Acestea copiază fișierele în locații diferite, în funcție de instrumentele utilizate, după cum urmează.

- ”`/dest/foo`”: rsync(8), GNU cp(1), și scp(1)
- ”`/dest/ruta/la/source/foo`”: GNU tar(1), și cpio(1)

Indicație
rsync(8) și GNU cp(1) au opțiunea „`-u`” pentru a omite fișierele care sunt mai noi pe receptor (la destinație).

10.1.5 Expresii idiomatice pentru selectarea fișierelor

find(1) este utilizat pentru a selecta fișiere pentru comenzile de arhivare și copiere (a se vedea Secțiune 10.1.3 și Secțiune 10.1.4) sau pentru `xargs` (1) (a se vedea Secțiune 9.4.9). Aceasta poate fi îmbunătățită prin utilizarea argumentelor sale de comandă.

Sintaxa de bază a find(1) poate fi rezumată după cum urmează.

- Argumentele sale condiționale sunt evaluate de la stânga la dreapta.
- Această evaluare se încheie odată ce rezultatul său este determinat.
- „**SAU** logic” (specificat prin „`-o`” între condiții) are prioritate mai mică decât „**ȘI** logic” (specificat prin „`-a`” sau nimic între condiții).
- „**NU** logic” (specificat prin „`!`” înaintea unei condiții) are prioritate mai mare decât „**ȘI** logic”.
- „`-prune`” returnează întotdeauna valoarea logică **ADEVĂRAT** și, dacă este vorba de un director, căutarea fișierului este oprită dincolo de acest punct.
- „`-name`” potrivește cu baza numelui fișierului cu expresiile regulate shell (a se vedea Secțiune 1.5.6), dar se potrivește de asemenea cu „`.`” inițial cu metacaractere precum „`*`” și „?”; (funcție nouă POSIX).
- „`-regex`” potrivește cu ruta completă în stilul emacs **BRE** (a se vedea Secțiune 1.6.2) ca valoare implicită.
- „`-size`” potrivește fișierul pe baza dimensiunii fișierului (valoare precedată de „+” pentru mai mare, precedată de „-” pentru mai mică)
- „`-newer`” potrivește cu fișierul mai nou decât cel specificat în argumentul său.
- „`-print0`” returnează întotdeauna **ADEVĂRAT** logic și afișează numele complet al fișierului (terminat cu caracterul null) la ieșirea standard.

find(1) este adesea utilizat într-un stil idiomatic, după cum urmează.

```
# find /path/to \
    -xdev -regextype posix-extended \
    -type f -regex ".*\.cpio|.*~" -prune -o \
    -type d -regex ".*/\.git" -prune -o \
    -type f -size +99M -prune -o \
    -type f -newer /path/to/timestamp -print0
```

Aceasta înseamnă efectuarea următoarelor acțiuni.

1. Caută toate fișierele ce încep cu „`/path/to`”
2. Limitează global căutarea în sistemul de fișiere inițial și utilizează în schimb **ERE** (a se vedea Secțiune 1.6.2).
3. Exclude fișierele care corespund expresiei regulate „`.*\.cpio`” sau „`.*~`” din căutare prin oprirea procesării.
4. Exclude din căutare directoarele care corespund expresiei regulate „`.*/\.git`” prin oprirea procesării.
5. Exclude fișierele mai mari de 99 megaocteți (unități de 1048576 octeți) din căutare prin oprirea procesării.
6. Afişează numele fişierelor care îndeplinesc condiţiile de căutare de mai sus şi sunt mai recente decât „`/path/to/time`

Vă rugăm să rețineți utilizarea idiomatică a „`-prune -o`” pentru a exclude fișierele din exemplul de mai sus.

Notă
Pentru sistemele non-Debian de tip Unix, este posibil ca unele opțiuni să nu fie acceptate de find(1). În acest caz, vă rugăm să luați în considerare ajustarea metodelor de potrivire și înlocuirea „`-print0`” cu „`-print`”. Este posibil să fie necesară ajustarea și a comenzilor asociate.

10.1.6 Suportul de arhivare

Atunci când alegeți suporturi de stocare a datelor pe calculator pentru arhivarea datelor importante, trebuie să fiți atenți la limitările acestora. Pentru copia de rezervă a datelor personale de mici dimensiuni, eu folosesc CD-R și DVD-R de la companii de renume și le păstrez într-un mediu răcoros, umbrit, uscat și curat; (suporturile de arhivare pe bandă par să fie populare pentru uz profesional.)

Notă
Un seif rezistent la foc este destinat documentelor imprimate pe hârtie. Majoritatea suporturilor de stocare a datelor informatice au o toleranță la temperatură mai mică decât hârtia. De obicei, mă bazez pe mai multe copii criptate securizate, stocate în mai multe locații sigure.

Durata de viață optimistă a suporturilor de arhivare observată pe internet (în principal din informațiile furnizate de producători).

- mai mult de 100 de ani: hârtie fără acid cu cerneală
- 100 de ani: stocare optică (CD/DVD, CD/DVD-R)
- 30 de ani: stocare magnetică (bandă, dischetă)
- 20 de ani: stocare optică cu schimbare de stare (inscripționare-ștergere-reinscripționare -- CD-RW)

Acestea nu iau în calcul defecțiunile mecanice cauzate de manipulare, etc.

Numărul de cicluri de scriere optimist al mediilor de arhivare observat pe internet (în principal din informațiile furnizate de producători).

- mai mult de 250.000 de cicluri: disc dur
- mai mult de 10.000 de cicluri: memorie flash
- 1.000 de cicluri: CD/DVD-RW
- 1 cicluri: CD/DVD-R, hârtie



Atenție
Cifrele referitoare la durata de stocare și ciclul de scriere prezentate aici nu trebuie utilizate pentru luarea de decizii privind stocarea datelor critice. Vă rugăm să consultați informațiile specifice despre produs furnizate de producător.

Indicație
Deoarece CD-urile/DVD-urile-R și hârtia au un singur ciclu de scriere, acestea previn în mod inerent pierderea accidentală a datelor prin suprascriere. Acesta este un avantaj!

Indicație
Dacă aveți nevoie de copii de rezervă rapide și frecvente pentru cantități mari de date, un disc dur pe un server la distanță conectat printr-o rețea rapidă poate fi singura opțiune realistă.

Indicație
Dacă utilizați suporturi reinscriptibile pentru copiile de rezervă, ar fi o idee bună să utilizați un sistem de fișiere precum btrfs sau zfs, care acceptă instantanee numai pentru citire.

10.1.7 Dispozitive de stocare amovibile (detașabile)

Dispozitivele de stocare amovibile pot fi oricare dintre următoarele.

- memorie USB
- disc dur
- disc optic
- Aparat de fotografiat digital
- Aparat de redare muzică digitală

Acestea pot fi conectate prin oricare dintre următoarele metode.

- USB
- IEEE 1394 / FireWire

Mediile grafice de birou moderne, precum GNOME și KDE, pot monta automat aceste dispozitive amovibile fără o intrare corespunzătoare în „`/etc/fstab`”.

- Pachetul `udisks2` oferă un demon și instrumente asociate pentru montarea și demontarea acestor dispozitive.
- D-bus creează evenimente pentru a iniția procese automate.
- PolicyKit furnizează privilegiile necesare.

Indicație
Dispozitivele montate automat pot avea opțiunea de montare „`uhelper=`”, care este utilizată de umount(8).

Indicație
Montarea automată în mediul grafic de birou modern are loc numai atunci când aceste dispozitive media amovibile nu sunt listate în „`/etc/fstab`”.

Punctul de montare în mediul grafic de birou modern este ales ca „`/media/`*`nume-utilizator/etichetă-dispozitiv`*”, care poate fi personalizat după cum urmează.

- mlabel(1) pentru sistemul de fișiere FAT
- genisoimage(1) cu opțiunea „`-V`” pentru sistemul de fișiere ISO9660
- tune2fs(1) cu opțiunea „`-L`” pentru sistemul de fișiere ext2/ext3/ext4

Indicație
Este posibil să fie necesar să se specifice alegerea codificării ca opțiune de montare (a se vedea Secțiune 8.1.3).

Indicație
Utilizarea meniului interfeței grafice pentru a dezactiva un sistem de fișiere poate elimina nodul dispozitivului generat dinamic, cum ar fi „`/dev/sdc`”. Dacă doriți să păstrați nodul dispozitivului, dezactivați-l cu comanda umount(8) din promptul shell.

10.1.8 Alegerea sistemului de fișiere pentru partajarea datelor

Când partajați date cu alte sisteme prin intermediul unui dispozitiv de stocare amovibil, trebuie să îl formatați cu un sistem de fișiere comun acceptat de ambele sisteme. Iată o listă cu opțiunile de sisteme de fișiere.

Notă
Declarațiile referitoare la discul dur (HDD) se aplică și altor dispozitive de stocare, cum ar fi SSD / unitate flash USB / card de memorie / ... . Înlocuiți numele dispozitivelor din exemple, cum ar fi `/dev/sda`, cu numele dispozitivelor aplicabile `/dev/nvme0`, `/dev/mmcblk0`, ...

Indicație
Consultați Secțiune 9.9.1 pentru partajarea datelor între platforme utilizând criptarea la nivel de dispozitiv.

Sistemul de fișiere FAT este acceptat de aproape toate sistemele de operare moderne și este foarte util pentru schimbul de date prin intermediul unor suporturi de stocare amovibile, precum discurile dure.

Atunci când formatați dispozitive precum discurile dure amovibile pentru partajarea datelor între platforme cu sistemul de fișiere FAT, următoarele opțiuni ar trebui să fie sigure.

- Partiționați-le cu fdisk(8), cfdisk(8) sau parted(8) (consultați Secțiune 9.6.2) într-o singură partiție primară și marcați-o după cum urmează.
  - Tastați „6” pentru FAT16 pentru suporturi media mai mici de 2 Go.
  - Tastați „c” pentru FAT32 (LBA) pentru suporturi media mai mari.
- Formatarea partiției primare cu mkfs.vfat(8) cu următoarele.

numele sistemului de fișiere	scenariul de utilizare tipic
FAT12	partajarea datelor între platforme pe dischetă (<32MiB)
FAT16	partajarea datelor între platforme pe un dispozitiv precum un disc dur mic (<2Gio)
FAT32	partajarea datelor între platforme pe un dispozitiv precum un disc dur de mare capacitate (<8Tio, compatibil cu versiuni mai noi decât MS Windows95 OSR2)
exFAT	partajarea datelor între platforme pe un dispozitiv precum un disc dur de mare capacitate (<512 TiB, compatibil cu WindowsXP, Mac OS X Snow Leopard 10.6.5 și cu nucleul Linux începând cu versiunea 5.4)
NTFS	partajarea datelor între platforme pe un dispozitiv precum un disc dur de mare capacitate (cu suport nativ în MS Windows NT și versiunile ulterioare, și cu suport în NTFS-3G prin intermediul FUSE în Linux)
ISO9660	partajarea între platforme a datelor statice pe CD-R și DVD+/-R
UDF	scrierea incrementală a datelor pe CD-R și DVD+/-R (nou)
MINIX	stocare eficientă a datelor fișierelor Unix pe dischetă
ext2	partajarea datelor de pe un dispozitiv precum discul dur cu sisteme Linux mai vechi
ext3	partajarea datelor de pe un dispozitiv precum discul dur cu sisteme Linux mai vechi
ext4	partajarea datelor pe un dispozitiv precum discul dur cu sistemele Linux actuale
btrfs	partajarea datelor pe dispozitive precum discul dur cu sistemele Linux actuale cu instantanee numai pentru citire

Tabela 10.3: Lista opțiunilor de sistem de fișiere pentru dispozitive de stocare amovibile cu scenarii tipice de utilizare

- Doar numele dispozitivului, de exemplu „`/dev/sda1`” pentru FAT16
- Opțiunea explicită și numele dispozitivului, de exemplu „`-F 32 /dev/sda1`” pentru FAT32

Atunci când utilizați sistemele de fișiere FAT sau ISO9660 pentru partajarea datelor, trebuie să luați în considerare următoarele aspecte de siguranță.

- Arhivarea fișierelor într-un fișier arhivă utilizând mai întâi tar(1) sau cpio(1) pentru a păstra numele lung al fișierului, legătura simbolică, permisiunea originală a fișierului Unix și informațiile despre proprietar.
- Împărțirea fișierului arhivă în bucăți mai mici de 2 Gio cu comanda split(1) pentru a-l proteja de limitarea dimensiunii fișierului.
- Criptarea fișierului arhivă pentru a proteja conținutul acestuia împotriva accesului neautorizat.

Notă
Pentru sistemele de fișiere FAT, conform specificațiilor, dimensiunea maximă a fișierului este `(2^32 - 1) octeți = (4Gio - 1 octet)`. Pentru unele aplicații pe sistemele de operare mai vechi de 32 de biți, dimensiunea maximă a fișierului era chiar mai mică `(2^31 - 1) octeți = (2Gio - 1 octet)`. Debian nu are această problemă.

Notă
Microsoft însuși nu recomandă utilizarea FAT pentru unități sau partiții de peste 200 Mo. Microsoft subliniază deficiențele sale, cum ar fi utilizarea ineficientă a spațiului pe disc, în „Prezentare generală a sistemelor de fișiere FAT, HPFS și NTFS”. Desigur, în mod normal ar trebui să utilizăm sistemul de fișiere ext4 pentru Linux.

Indicație
Pentru mai multe informații despre sistemele de fișiere și accesarea acestora, citiți „Ghidul sistemelor de fișiere”.

10.1.9 Partajarea datelor prin rețea

Când partajați date cu alte sisteme prin rețea, ar trebui să utilizați un serviciu comun. Iată câteva sfaturi.

serviciul de rețea	**descrierea scenariului tipic de utilizare**
SMB/CIFS sistem de fișiere montat în rețea cu Samba	partajarea fișierelor prin „Rețea Microsoft Windows”, consultați smb.conf(5) și Ghidul oficial Samba 3.x.x HOWTO și ghidul de referință sau pachetul `samba-doc`
NFS sistem de fișiere montat în rețea cu nucleul Linux	partajarea fișierelor prin „Rețea Unix/Linux”, consultați exports(5) și Linux NFS-HOWTO
serviciul HTTP	partajarea fișierelor între serverul web/client
serviciul HTTPS	partajarea fișierelor între serverul web/client cu Secure Sockets Layer (SSL) criptat sau Transport Layer Security (TLS)
serviciul FTP	partajarea fișierelor între serverul/clientul FTP

Tabela 10.4: Lista serviciilor de rețea din care puteți alege, cu scenariul tipic de utilizare

Deși aceste sisteme de fișiere montate în rețea și metodele de transfer de fișiere prin rețea sunt destul de convenabile pentru partajarea datelor, acestea pot fi nesigure. Conexiunea lor la rețea trebuie securizată prin următoarele măsuri.

- Criptați-o cu SSL/TLS
- Plasați-o în tunel prin SSH
- Plasați-o în tunel prin VPN
- Limitați-o în spatele paravanului de protecție securizat

Consultă de asemenea Secțiune 6.5 și Secțiune 6.6.

10.2 Copia de rezervă și recuperarea

Știm cu toții că uneori calculatoarele se defechează sau că erorile umane pot provoca daune sistemului și datelor. Operațiile de copie de rezervă și recuperare sunt o parte esențială a administrării cu succes a sistemului. Toate modurile posibile de funcționare defectuoasă vă pot afecta într-o zi.

Indicație
Mențineți sistemul de copiere de rezervă simplu și faceți copii de rezervă ale sistemului în mod frecvent. Este mai important să aveți date de rezervă decât să dispuneți de o metodă de copiere de rezervă performantă din punct de vedere tehnic.

10.2.1 Politica de copie de rezervă și recuperare

Există 3 factori cheie care determină politica actuală de copie de rezervă și recuperare.

1. Să știi ce să copiezi și ce să recuperezi.
   - Fișiere de date create direct de dvs.: date în „`~/`”
   - Fișiere de date create de aplicațiile utilizate de dvs.: date în „`/var/`” (cu excepția „`/var/cache/`”, „`/var/run/`” și „`/var/tmp/`”)
   - Fișiere de configurare a sistemului: date în „`/etc/`”

- Programe locale: date în „`/usr/local/`” sau „`/opt/`”
- Informații privind instalarea sistemului: o notă în format text simplu privind etapele cheie (partiționare, …)
- Set de date verificat: confirmat în prealabil prin operații experimentale de recuperare
  - Sarcina cron ca proces de utilizator: fișiere în directorul „`/var/spool/cron/crontabs`” și repornirea cron(8). Consultați Secțiune 9.4.14 pentru cron(8) și crontab(1).
  - Sarcini programate systemd ca procese utilizator: fișiere în directorul „`~/.config/systemd/user`”. Consultați systemd.timer(5) și systemd.service(5).
  - Sarcini de pornire automată ca procese ale utilizatorului: fișiere din directorul „`~/.config/autostart`”. Consultați Specificații privind pornirea automată a aplicațiilor de birou.

2. Să știi cum să faci copii de rezervă și să recuperezi datele.
   - Stocarea sigură a datelor: protecție împotriva suprascrierii și a defecțiunilor sistemului
   - Copiere de rezervă frecventă: copie de rezervă programată
   - Copie de rezervă redundantă: oglindirea datelor
   - Proces fără greșeli: copie de rezervă ușoară cu o singură comandă
3. Evaluarea riscurilor și a costurilor implicate.
   - Riscul de pierdere a datelor
     - Datele trebuie să se afle cel puțin pe partiții de disc diferite, de preferință pe discuri și mașini diferite, pentru a rezista la coruperea sistemului de fișiere. Datele importante sunt stocate cel mai bine pe un sistem de fișiere numai pentru citire. 1
   - Riscul de încălcare a securității datelor
     - Date sensibile de identitate, cum ar fi „`/etc/ssh/ssh_host_*_key`”, „`~/.gnupg/*`”, „`~/.ssh/*`”, „`~/.local/`” „`/etc/passwd`”, „`/etc/shadow`”, „`popularity-contest.conf`”, „`/etc/ppp/pap-secrets`” și „`/etc/exim`” trebuie să fie copiate ca fișiere criptate. 2 (A se vedea Secțiune 9.9.)
     - Nu încorporați niciodată parola de conectare la sistem sau fraza de decriptare în niciun script, chiar și pe un sistem de încredere; (a se vedea Secțiune 10.3.6).
   - Modul de defectare și posibilitatea apariției acestuia
     - Hardware-ul (în special HDD-ul) se va defecta
     - Sistemul de fișiere poate fi corupt, iar datele din acesta pot fi pierdute.
     - Sistemul de stocare la distanță nu este de încredere din punct de vedere al securității
     - Protecția prin parolă slabă poate fi compromisă cu ușurință
     - Sistemul de permisiuni pentru fișiere poate fi compromis
   - Resurse necesare pentru copie de rezervă: resurse umane, hardware, software, …
     - Copiere de rezervă automată programată cu sarcină de «cron» (cron job) sau cu sarcină programată de systemd (systemd timer job)

Indicație
Puteți recupera datele de configurare debconf cu „`debconf-set-selections debconf-selections`” și datele de selecție dpkg cu „`dpkg --set-selection <dpkg-selections.list`”.

Notă
Nu faceți copii de rezervă pentru conținutul pseudo-sistemului de fișiere găsit în `/proc`, `/sys`, `/tmp` și `/run` (consultați Secțiune 1.2.12 și Secțiune 1.2.13). Dacă nu știți exact ce faceți, acestea sunt date inutile de dimensiuni enorme.

---

1 Un suport de stocare cu o singură scriere, cum ar fi CD/DVD-R, poate preveni accidentele de suprascriere. (Consultați Secțiune 9.8 pentru a afla cum se scrie pe suportul de stocare din linia de comandă a shell-ului. Mediul grafic de birou GNOME vă oferă acces ușor prin meniu: „Locuri → Creator CD/DVD”.)

2 Unele dintre aceste date nu pot fi regenerate introducând același șir de intrare în sistem.

Notă
Este posibil să doriți să opriți anumiți demoni de aplicații, cum ar fi MTA (consultați Secțiune 6.2.4), în timp ce faceți copii de rezervă ale datelor.

10.2.2 Suite de instrumente de copie de rezervă

Iată o listă selectivă a celor mai importante suite de instrumente de copiere de rezervă disponibile în sistemul Debian.

pachet	popcon(popularitate)	dimensiune	descriere
bacula-common	V:6.0, I:6.7	2501	Bacula: copiere de rezervă, recuperare și verificare în rețea - fișiere comune de asistență
bacula-client	V:0.2, I:1.8	199	Bacula: copiere de rezervă, recuperare și verificare în rețea - meta-pachet client
bacula-console	V:0.6, I:2.1	112	Bacula: copiere de rezervă, recuperare și verificare în rețea - consolă text
bacula-server	I:0.59	199	Bacula: copiere de rezervă, recuperare și verificare în rețea - meta-pachet server
amanda-common	V:0.6, I:2.1	9951	Amanda: „Advanced Maryland Automatic Network Disk Archiver” -- Arhivator automat avansat de discuri în rețea Maryland (biblioteci)
amanda-client	V:0.5, I:2.1	1109	Amanda: Arhivator automat avansat de discuri în rețea Maryland (client)
amanda-server	V:0.11, I:0.22	1095	Amanda: Arhivator automat avansat de discuri în rețea Maryland (server)
backuppc	V:1.4, I:1.6	3088	BackupPC X este un sistem performant, la nivel de întreprindere, pentru copierea de rezervă a calculatoarelor (pe disc).
duplicity	V:5, I:44	2662	copie de rezervă incrementală (la distanță)
deja-dup	V:28, I:41	5274	interfață grafică pentru duplicitate
borgbackup	V:13, I:28	3532	de duplicarea copiilor de rezervă (de la distanță)
borgmatic	V:3.2, I:4.4	946	asistent robotizat de copiere de rezervă
rdiff-backup	V:2.1, I:6.1	1169	copie de rezervă incrementală (la distanță)
restic	V:8, I:15	41617	copie de rezervă incrementală (la distanță)
backupninja	V:2.0, I:2.4	360	sistem de **meta-copiere de rezervă** ușor și extensibil
slbackup	V:0.10, I:0.13	147	copie de rezervă incrementală (la distanță)
backup-manager	V:0.41, I:0.70	573	instrument de copie de rezervă din linia de comandă
backup2l	V:0.35, I:0.46	110	instrument de copiere de rezervă/restaurare cu întreținere redusă pentru suporturi ce pot fi montate (pe bază de disc)

Tabela 10.5: Lista suitelor de instrumente de copiere de rezervă

Instrumentele de copie de rezervă au domenii de specializare specifice.

- Mondo Rescue este un sistem de copiere de rezervă care facilitează restaurarea rapidă a întregului sistem de pe CD-uri/DVD-uri de rezervă etc., fără a fi necesar să se parcurgă procesele normale de instalare a sistemului.
- Bacula, Amanda, și BackupPC sunt suite complete de instrumente de copiere de rezervă, axate pe copierea periodică de rezervă prin rețea.
- Duplicity, și Borg sunt instrumente de copiere de rezervă mai simple pentru stațiile de lucru obișnuite.

10.2.3 Sfaturi pentru copierea de rezervă

Pentru o stație de lucru personală, instrumentele complete de copiere de rezervă concepute pentru mediul server pot să nu fie adecvate. În același timp, instrumentele existente de copiere de rezervă pentru stațiile de lucru pot avea unele deficiențe.

Iată câteva sfaturi pentru a facilita realizarea copiilor de rezervă cu eforturi minime din partea utilizatorului. Aceste tehnici pot fi utilizate cu orice instrumente de copiere de rezervă.

În scop demonstrativ, să presupunem că numele utilizatorului și al grupului principal este `penguin` și să creăm un exemplu de script de copiere de rezervă și instantanee„`/usr/local/bin/bkss.sh`” după cum urmează:

```
#!/bin/sh -e
SRC="$1" # source data path
DSTFS="$2" # backup destination filesystem path
DSTSV="$3" # backup destination subvolume name
DSTSS="${DSTFS}/${DSTSV}-snapshot" # snapshot destination path
if [ "$(stat -f -c %T "$DSTFS")" != "btrfs" ]; then
  echo "E: $DESTFS needs to be formatted to btrfs" >&2 ; exit 1
fi
MSGID=$(notify-send -p "bkup.sh $DSTSV" "in progress ...")
if [ ! -d "$DSTFS/$DSTSV" ]; then
  btrfs subvolume create "$DSTFS/$DSTSV"
  mkdir -p "$DSTSS"
fi
rsync -aHxS --delete --mkpath "${SRC}/" "${DSTFS}/${DSTSV}"
btrfs subvolume snapshot -r "${DSTFS}/${DSTSV}" ${DSTSS}/$(date -u --iso=min)
notify-send -r "$MSGID" "bkup.sh $DSTSV" "finished!"
```

Aici, doar instrumentul de bază rsync(1) este utilizat pentru a facilita copierea de rezervă a sistemului, iar spațiul de stocare este utilizat în mod eficient de Btrfs.

Indicație
Pentru informație: autorul utilizează propriul script shell similar „bss: Btrfs Subvolume Snapshot Utility” pentru stația sa de lucru.

10.2.3.1 Copierea de rezervă în interfața grafică

Iată un exemplu pentru configurarea copierii de rezervă cu un singur clic în interfața grafică.

- Pregătiți un dispozitiv de stocare USB care va fi utilizat pentru copia de rezervă.
  - Formatați un dispozitiv de stocare USB cu o singură partiție în btrfs, cu numele de etichetă „BKUP”. Acesta poate fi criptat (consultați Secțiune 9.9.1).
  - Conectați-l în sistemul dvs. Sistemul grafic de birou ar trebui să îl monteze automat ca „`/media/penguin/BKUP`”.
  - Executați «`sudo chown penguin:penguin /media/penguin/BKUP`» pentru a-l face inscriptibil de către utilizator.
- Creați „`~/.local/share/applications/BKUP.desktop`” urmând tehnicile descrise în Secțiune 9.4.10 astfel:

```
[Desktop Entry]
Name=bkss
Comment=Backup and snapshot of ~/Documents
Exec=/usr/local/bin/bkss.sh /home/penguin/Documents /media/penguin/BKUP Documents
Type=Application
```

La fiecare clic în interfața grafică, datele dvs. sunt copiate din „`~/Documents`” pe un dispozitiv de stocare USB și se creează o instantanee numai pentru citire.

10.2.3.2 Copiere de rezervă declanșată de evenimentul de montare

Iată un exemplu de configurare pentru copierea automată de rezervă declanșată de evenimentul de montare.

- Pregătiți un dispozitiv de stocare USB care să fie utilizat pentru copie de rezervă, ca în Secțiune 10.2.3.1.
- Creați un fișier de unitate de serviciu systemd „`~/.config/systemd/user/back-BKUP.service`” astfel:

```
[Unit]
Description=USB Disk backup
Requires=media-%u-BKUP.mount
After=media-%u-BKUP.mount

[Service]
ExecStart=/usr/local/bin/bkss.sh %h/Documents /media/%u/BKUP Documents
StandardOutput=append:%h/.cache/systemd-snap.log
StandardError=append:%h/.cache/systemd-snap.log

[Install]
WantedBy=media-%u-BKUP.mount
```

- Activați această configurație a unității systemd cu următoarea comandă:

```
 $ systemctl --user enable bkup-BKUP.service
```

Pentru fiecare eveniment de montare, datele dvs. sunt copiate din „`~/Documents`” pe un dispozitiv de stocare USB și se creează o instantanee numai pentru citire.

Aici, numele unităților de montare systemd pe care systemd le are în prezent în memorie pot fi solicitate gestionarului de servicii al utilizatorului care efectuează apelul cu «`systemctl --user list-units --type=mount`».

10.2.3.3 Copiere de rezervă declanșată de evenimentul programatorului de sarcini

Iată un exemplu de configurare pentru copierea automată de rezervă declanșată de evenimentul programatorului de sarcini.

- Pregătiți un dispozitiv de stocare USB care să fie utilizat pentru copie de rezervă, ca în Secțiune 10.2.3.1.
- Creați un fișier de unitate de programare systemd „`~/.config/systemd/user/snap-Documents.timer`” astfel:

```
[Unit]
Description=Run btrfs subvolume snapshot on timer
Documentation=man:btrfs(1)

[Timer]
OnStartupSec=30
OnUnitInactiveSec=900

[Install]
WantedBy=timers.target
```

- Creați un fișier de unitate de serviciu systemd „`~/.config/systemd/user/snap-Documents.service`” astfel:

```
[Unit]
Description=Run btrfs subvolume snapshot
Documentation=man:btrfs(1)

[Service]
```

```
Type=oneshot
Nice=15
ExecStart=/usr/local/bin/bkss.sh %h/Documents /media/%u/BKUP Documents
IOSchedulingClass=idle
CPUSchedulingPolicy=idle
StandardOutput=append:%h/.cache/systemd-snap.log
StandardError=append:%h/.cache/systemd-snap.log
```

- Activați această configurație a unității systemd cu următoarea comandă:

```
$ systemctl --user enable snap-Documents.timer
```

Pentru fiecare eveniment programat, datele dvs. sunt copiate din „`~/Documents`” pe un dispozitiv de stocare USB și se creează o instantanee numai pentru citire.

Aici, numele unităților de utilizator ale programatorului systemd pe care systemd le are în prezent în memorie pot fi solicitate gestionatorului de servicii al utilizatorului care efectuează apelul cu «`systemctl --user list-units --type=timer`».

Pentru sistemele grafice de birou moderne, această abordare systemd poate oferi un control mai precis decât cele tradiționale Unix care utilizează at(1), cron(8) sau anacron(8).

10.3 Infrastructura de securitate a datelor

Infrastructura de securitate a datelor este asigurată de combinația dintre instrumentul de criptare a datelor, instrumentul de rezumat al mesajelor și instrumentul de semnătură.

pachet	popcon(popularitate)	dimensiune	comandă	descriere
gnupg	V:418, I:860	465	gpg(1)	GNU Privacy Guard - instrument de criptare și semnare OpenPGP
gpgv	V:203, I:923	577	gpgv(1)	GNU Privacy Guard - instrument de verificare a semnăturilor OpenPGP
sq	V:1, I:33	24447	sq(1)	Sequoia-PGP - instrument de criptare și semnare OpenPGP
sqv	V:479, I:554	1919	sqv(1)	Sequoia-PGP - instrument de verificare a semnăturilor OpenPGP
paperkey	V:1, I:12	58	paperkey(1)	extrage doar informațiile secrete din cheile secrete OpenPGP
cryptsetup	V:34, I:77	469	cryptsetup(8), …	ustensile pentru dm-crypt criptarea dispozitivelor bloc care acceptă LUKS
coreutils	V:912, I:1000	17994	md5sum(1)	calculează și verifică rezumatul mesajului MD5
coreutils	V:912, I:1000	17994	sha1sum(1)	calculează și verifică rezumatul mesajului SHA1
openssl	V:846, I:996	2564	`openssl`(1ssl)	calculează rezumatul mesajului cu „`openssl dgst`” (OpenSSL)
libsecret-tools	V:2, I:13	44	secret-tool(1)	stochează și recuperează parole (CLI)
seahorse	V:76, I:260	7999	seahorse(1)	instrument de gestionare a cheilor (GNOME)

Tabela 10.6: Lista instrumentelor de infrastructură pentru securitatea datelor

Consultați Secțiune 9.9 cu dm-crypt și fscrypt, care implementează infrastructura de criptare automată a datelor prin intermediul modulelor nucleului Linux.

10.3.1 Gestionarea cheilor pentru GnuPG

Iată comenzile GNU Privacy Guard pentru gestionarea cheilor de bază.

comandă	descriere
`gpg --gen-key`	generează o cheie nouă
`gpg --gen-revoke ID-ul_meu_de-utilizator`	generează cheia de revocare pentru ID-ul_meu_de-utilizator
`gpg --edit-key ID_utilizator`	editează cheia în mod interactiv, „help” pentru ajutor
`gpg -o fișier --export`	exportă toate cheile publice în fișier
`gpg -o fișier --export-secret-keys`	exportă toate cheile private în fișier
`gpg --import fișier`	importă toate cheile din fișier
`gpg --send-keys ID_utilizator`	trimite cheia pentru ID_utilizator către serverul de chei
`gpg --recv-keys ID_utilizator`	primește cheia pentru ID_utilizator de la serverul de chei
`gpg --list-keys ID_utilizator`	listează cheile pentru ID_utilizator
`gpg --list-sigs ID_utilizator`	listează semnăturile pentru ID_utilizator
`gpg --check-sigs ID_utilizator`	verifică semnăturile pentru ID_utilizator
`gpg --fingerprint user_ID`	verifică amprenta digitală pentru ID_utilizator
`gpg --refresh-keys`	actualizează inelul de chei local

Tabela 10.7: Lista comenzilor GNU Privacy Guard pentru gestionarea cheilor

Iată semnificația codului de încredere.

cod	descrierea încrederii
`-`	nu s-a atribuit nicio încredere proprietarului / nu s-a calculat încă
`e`	calcularea încrederii a eșuat
`q`	informații insuficiente pentru calcul
`n`	nu aveți încredere niciodată în această cheie
`m`	încredere marginală
`f`	încredere deplină
`u`	în cele din urmă, de încredere

Tabela 10.8: Lista semnificațiilor codului de încredere

Următoarrea comandă generează cheia mea „`9563FC29932C409F1A77F9C1AB8A1522D8234C6A`”.

```
$ gpg --gen-key
gpg (GnuPG) 2.4.7; Copyright (C) 2024 g10 Code GmbH
...
GnuPG needs to construct a user ID to identify your key.

Real name: Osamu Aoki
Email address: osamu@debian.org
You selected this USER-ID:
    "Osamu Aoki <osamu@debian.org>"

Change (N)ame, (E)mail, or (O)kay/(Q)uit? o
...
public and secret key created and signed.

pub   ed25519 2026-02-14 [SC] [expires: 2029-02-13]
      9563FC29932C409F1A77F9C1AB8A1522D8234C6A
uid                      Osamu Aoki <osamu@debian.org>
sub   cv25519 2026-02-14 [E] [expires: 2029-02-13]
```

Următoarea comandă încarcă cheia mea „9563FC29932C409F1A77F9C1AB8A1522D8234C6A” pe popularul server de chei „hkp://keys.gnupg.net”.

```
$ gpg --keyserver hkp://keys.gnupg.net --send-keys 9563FC29932C409F1A77F9C1AB8A1522D8234C6A
```

Un server de chei implicit bine configurat în „~/.gnupg/gpg.conf” (sau vechea locație „~/.gnupg/options”) conține următoarele.

```
keyserver hkp://keys.gnupg.net
```

Următoarea comandă obține cheile necunoscute de la serverul de chei.

```
$ gpg --list-sigs --with-colons | grep '^sig.*\[User ID not found\]' |\
          cut -d ':' -f 5| sort | uniq | xargs gpg --recv-keys
```

A existat o eroare în OpenPGP Public Key Server (versiunea anterioară 0.9.6) care a corupt cheia cu mai mult de 2 subchei. Pachetul mai nou gnupg (>1.2.1-2) poate gestiona aceste subchei corupte. Consultați gpg(1) la opțiunea „--repair-pks-subkey-bug”.

Utilizarea SHA-1 pentru algoritmul de rezumat a fost înlocuită. Dacă vechea cheie openPGP bazată pe RSA utilizează SHA-1 pentru algoritmul de rezumat, remediați problema utilizând FixKeyWithSha1.

Indicație
Comenzile sq(1) și sqv(1) sunt un set alternativ de comenzi openPGP. Consultați documentația utilizatorului sq și O introducere practică în utilizarea sq, CLI-ul Sequoia PGP.

10.3.2 Utilizarea GnuPG pe fișiere

Iată câteva exemple de utilizare a comenzilor GNU Privacy Guard pe fișiere.

10.3.3 Utilizarea GnuPG cu Mutt

Adăugați următoarele în „~/.muttrc” pentru a împiedica pornirea automată a GnuPG, permițând totuși utilizarea acestuia prin tastarea „S” în meniul index.

```
macro index S ":toggle pgp_verify_sig\n"
set pgp_verify_sig=no
```

10.3.4 Utilizarea GnuPG cu Vim

Modulul gnupg vă permite să rulați GnuPG în mod transparent pentru fișierele cu extensia „.gpg”, „.asc” și „.pgp”.[3]

```
$ sudo aptitude install vim-scripts
$ echo "packadd! gnupg" >> ~/.vim/vimrc
```

3Dacă utilizați „~/.vimrc” în loc de „~/.vim/vimrc”, faceți înlocuirea corespunzătoare.

comandă	descriere
`gpg -a -s fișier`	semnează fișierul în armură ASCII ca fișier.asc
`gpg --armor --sign fișier`	, ,
`gpg --clearsign fișier`	semnează în text clar mesajul
`gpg --clearsign fișier|mesaj foo@example.org`	trimite un mesaj semnat în text clar la `foo@example.org`
`gpg --clearsign --not-dash-escaped patchfile`	semnează în text clar fișierul.patch
`gpg --verify fișier`	verifică fișierul semnat în text clar
`gpg -o fișier.sig -b fișier`	creează o semnătură detașată
`gpg -o fișier.sig --detach-sign fișier`	, ,
`gpg --verify fișier.sig fișier`	verifică fișierul cu fișierul.sig
`gpg -o fișier_criptat.gpg -r nume -e fișier`	criptare cu cheie publică destinată numelui din fișierul binar fișier_criptat.gpg
`gpg -o fișier_criptat.gpg --recipient nume --encrypt fișier`	, ,
`gpg -o fișier_criptat.asc -a -r name -e fișier`	criptare cu cheie publică destinată numelui din fișier către fișierul_criptat.asc cu armură ASCII
`gpg -o fișier_criptat.gpg -c fișier`	criptare simetrică din fișier în fișier_criptat.gpg
`gpg -o fișier_criptat.gpg --symmetric fișier`	, ,
`gpg -o fișier_criptat.asc -a -c fișier`	criptare simetrică destinată numelui din fișier către fișierul cu armură ASCII, fișier_criptat.asc
`gpg -o fișier -d fișier_criptat.gpg -r nume`	decriptare
`gpg -o fișier --decrypt fișier_criptat.gpg`	, ,

Tabela 10.9: Lista comenzilor GNU Privacy Guard pentru fișiere

10.3.5 Suma de control MD5

md5sum(1) oferă utilitatea de a crea un fișier de rezumat (digest) folosind metoda din rfc1321 și de a verifica fiecare fișier cu ajutorul acestuia.

```
$ md5sum foo bar >baz.md5
$ cat baz.md5
d3b07384d113edec49eaa6238ad5ff00  foo
c157a79031e1c40f85931829bc5fc552  bar
$ md5sum -c baz.md5
foo: OK
bar: OK
```

Notă
Calculul pentru suma MD5 solicită mai puțin procesorul decât cel pentru semnătura criptografică realizată de GNU Privacy Guard (GnuPG). De obicei, numai fișierul de rezumat (digest) de nivel superior este semnat criptografic pentru a asigura integritatea datelor.

10.3.6 Inel de chei pentru parole

În sistemul GNOME, instrumentul cu interfață grafică seahorse(1) gestionează parolele și le stochează în siguranță în inelul de chei `~/.local/share/keyrings/*`.

secret-tool(1) poate stoca parola în inelul de chei din linia de comandă.

Să stocăm fraza de acces utilizată pentru imaginea discului criptat LUKS/dm-crypt

```
$ secret-tool store --label='LUKS passphrase for disk.img' LUKS my_disk.img
Password: ********
```

Această parolă stocată poate fi recuperată și introdusă în alte programe, de exemplu, cryptsetup(8).

```
$ secret-tool lookup LUKS my_disk.img | \
  cryptsetup open disk.img disk_img --type luks --keyring -
$ sudo mount /dev/mapper/disk_img /mnt
```

Indicație
Ori de câte ori trebuie să furnizați o parolă într-un script, utilizați `secret-tool` și evitați să codificați direct fraza de acces în acesta.

10.4 Instrumente de îmbinare a codului sursă

Există multe instrumente de îmbinare/fuzionare pentru codul sursă. Următoarele comenzi mi-au atras atenția.

10.4.1 Extragerea diferențelor pentru fișierele sursă

Următoarele proceduri extrag diferențele dintre două fișiere sursă și creează fișiere diff unificate „`file.patch0`” sau „`file.patch1`”, în funcție de locația fișierului.

```
$ diff -u file.old file.new > file.patch0
$ diff -u old/file new/file > file.patch1
```

pachet	popcon(popularitate)	linii	comandă	descriere
patch	V:102, I:727	242	patch(1)	aplică un fișier diff la un fișier original
vim	V:83, I:338	4164	vimdiff(1)	compară 2 fișiere unul lângă altul în vim
imediff	V:0.02, I:0.25	329	imediff(1)	instrument interactiv pe ecran complet de îmbinare/fuzionare a 2/3 fișiere
meld	V:5, I:23	3996	meld(1)	compară și fuzionează fișiere (GTK)
wiggle	V:0.02, I:0.12	175	wiggle(1)	aplică corecțiile (patch) respinse
diffutils	V:896, I:998	1768	diff(1)	compară fișiere linie cu linie
diffutils	V:896, I:998	1768	diff3(1)	compară și fuzionează trei fișiere linie cu linie
quilt	V:2, I:16	880	quilt(1)	gestionează serii de plasturi(corecții)
wdiff	V:5, I:38	580	wdiff(1)	afișează diferențele de cuvinte între fișierele text
diffstat	V:9, I:93	79	diffstat(1)	produce o histogramă a modificărilor prin diff
patchutils	V:11, I:92	269	combinediff(1)	creează un plasture cumulativ din doi plasturi incrementali
patchutils	V:11, I:92	269	dehtmldiff(1)	extrage un fișier diff dintr-o pagină HTML
patchutils	V:11, I:92	269	filterdiff(1)	extrage sau exclude diferențele dintr-un fișier diff
patchutils	V:11, I:92	269	fixcvsdiff(1)	corectează fișierele diff create de CVS pe care patch(1) le interpretează greșit
patchutils	V:11, I:92	269	flipdiff(1)	schimbă ordinea a două fișiere de corecții(plasturi)
patchutils	V:11, I:92	269	grepdiff(1)	afișează fișierele modificate de un patch care corespunde unei expresii regulate
patchutils	V:11, I:92	269	interdiff(1)	afișează diferențele dintre două fișiere diff unificate
patchutils	V:11, I:92	269	lsdiff(1)	afișează fișierele modificate de un plasture
patchutils	V:11, I:92	269	recountdiff(1)	recalculează numărările și decalajele într-un context unificat de diferențe
patchutils	V:11, I:92	269	rediff(1)	corectează decalajele și numărările unui fișier diff editat manual
patchutils	V:11, I:92	269	splitdiff(1)	separă plasturii(corecțiile) incrementale
patchutils	V:11, I:92	269	unwrapdiff(1)	repară plasturii(corecțiile) în care cuvintele au fost împărțite pe mai multe rânduri
dirdiff	V:0.1, I:1.3	167	dirdiff(1)	afișează diferențele și îmbină/fuzionează modificările între arbori de directoare
docdiff	V:0.01, I:0.16	554	docdiff(1)	compară două fișiere cuvânt cu cuvânt / caracter cu caracter
makepatch	V:0.01, I:0.14	99	makepatch(1)	generează fișiere .patch extinse
makepatch	V:0.01, I:0.14	99	applypatch(1)	aplică fișiere .patch extinse

Tabela 10.10: Lista instrumentelor de îmbinare a codului sursă

10.4.2 Fuzionarea actualizărilor pentru fișierele sursă

Fișierul diff (denumit alternativ fișier patch) este utilizat pentru a trimite o actualizare a programului. Partea destinatară aplică această actualizare unui alt fișier după cum urmează.

```
$ patch -p0 file < file.patch0
$ patch -p1 file < file.patch1
```

10.4.3 Îmbinare/fuzionare interactivă

Dacă aveți două versiuni ale unui cod sursă, puteți efectua o îmbinare bidirecțională în mod interactiv utilizând imediff(1) după cum urmează.

```
$ imediff -o file.merged file.old file.new
```

Dacă aveți trei versiuni ale unui cod sursă, puteți efectua o îmbinare interactivă în trei moduri utilizând imediff(1) după cum urmează.

```
$ imediff -o file.merged file.yours file.base file.theirs
```

10.5 Git

Git este instrumentul preferat în prezent pentru sistemul de control al versiunilor (VCS), deoarece Git poate face totul atât pentru gestionarea codului sursă local, cât și pentru cea de la distanță.

Debian furnizează servicii Git gratuite prin intermediul serviciului Debian Salsa. Documentația acestuia poate fi găsită la https://wiki.debian.org/Salsa .

Iată câteva pachete legate de Git.

10.5.1 Configurarea clientului Git

Puteți specifica mai multe configurații globale în „`~/.gitconfig`”, cum ar fi numele și adresa dvs. de poștă electronică utilizate de Git, după cum urmează.

```
$ git config --global user.name "Name Surname"
$ git config --global user.email yourname@example.com
```

De asemenea, puteți personaliza comportamentul implicit al Git în felul următor.

```
$ git config --global init.defaultBranch main
$ git config --global pull.rebase true
$ git config --global push.default current
```

Dacă sunteți prea obișnuit cu comenzile CVS sau Subversion, puteți crea mai multe alias de comenzi folosind următoarea comandă.

```
$ git config --global alias.ci "commit -a"
$ git config --global alias.co checkout
```

Puteți verifica configurația globală după cum urmează.

```
$ git config --global --list
```

pachet	popcon(popularitate)	dimensiune	comandă	descriere
git	V:422, I:625	52512	git(7)	Git, sistemul rapid, scalabil și distribuit de control al reviziilor
gitk	V:3, I:26	2045	gitk(1)	navigator cu interfață grafică pentru depozitul Git, cu istoric
qgit	V:0.2, I:1.9	1431	qgit(1)	navigator cu interfață grafică pentru depozitul Git, cu istoric
git-cola	V:0.8, I:4.5	5129	git-cola(1)	navigator cu interfață grafică pentru depozitul Git, cu istoric
tig	V:2, I:11	1245	tig(1)	navigator de depozite Git pentru consolă, cu istoric
lazygit	V:1.0, I:4.1	24167	lazygit(1)	navigator de depozite Git pentru consolă, cu istoric
git-gui	V:1, I:16	2549	git-gui(1)	interfață grafică pentru depozitul Git (fără istoric)
git-email	V:0.3, I:9.3	1229	git-send-email(1)	trimite o colecție de plasturi(corecții) prin poșta electronică din Git
git-buildpackage	V:1.2, I:6.8	2027	git-buildpackage(1)	automatizează împachetarea Debian cu Git
dgit	V:0.12, I:0.97	757	dgit(1)	interoperabilitatea git cu arhiva Debian
imediff	V:0.02, I:0.25	329	git-ime(1)	instrument interactiv de ajutor pentru divizarea modificărilor comise în depozitul git
stgit	V:0.08, I:0.50	6253	stg(1)	quilt peste git (Python)
git-doc	I:10	15192	N/D	documentația oficială pentru Git
gitmagic	I:0.48	721	N/D	„Git Magic”, ghid mai ușor de înțeles pentru Git

Tabela 10.11: Lista pachetelor și comenzilor legate de git

10.5.2 Comenzi Git de bază

Operarea cu Git implică mai multe date.

- Arborele de lucru care conține fișierele destinate utilizatorilor și în care efectuați modificări.
  - Modificările care trebuie înregistrate trebuie selectate în mod explicit și introduse în index. Acestea sunt comenzile `git add` și `git rm`.
- Indexul care conține fișierele stocate în etape.
  - Fișierele modificate într-o etapă vor fi înregistrate în depozitul local la următoarea solicitare. Aceasta este comanda `git commit`.
- Depozitul local care conține fișierele modificate.
  - Git înregistrează istoricul vinculat al datelor comise și le organizează ca ramuri în depozit.
  - Depozitul local poate trimite date către depozitul la distanță prin comanda `git push`.
  - Depozitul local poate primi date din depozitul de la distanță prin comenzile `git fetch` și `git pull`.
    * Comanda `git pull` execută comanda `git merge` sau `git rebase` după comanda `git fetch`.
    * Aici, `git merge` combină două ramuri separate ale istoricului la final într-un singur punct. (Aceasta este opțiunea implicită a `git pull` fără personalizare și poate fi utilă pentru persoanele din amonte care publică ramuri către multe persoane.)
    * Aici, `git rebase` creează o singură ramură a istoricului secvențial al ramurii la distanță unu, urmată de ramura locală unu. (Acesta este un caz de personalizare `pull.rebase true` și poate fi util pentru cei mai mulți dintre noi.)
- Depozitul de la distanță care conține fișierele cu modificările comise.
  - Comunicarea cu depozitul de la distanță utilizează protocoale de comunicare securizate, precum SSH sau HTTPS.

Arborele de lucru este format din fișierele din afara directorului `.git/`. Fișierele din directorul `.git/` conțin indexul, datele din depozitul local și câteva fișiere text de configurare pentru git.

Iată o prezentare generală a principalelor comenzi Git.

10.5.3 Sfaturi Git

Iată câteva sfaturi pentru Git.



Avertisment
Nu utilizați șirul de etichete care conține spații, chiar dacă unele instrumente, cum ar fi gitk(1), vă permit să îl utilizați. Acesta poate bloca alte comenzi `git`.



Atenție
Dacă o ramură locală care a fost introdusă în depozitul de la distanță este rebazată sau comprimată, introducerea acestei ramuri prezintă riscuri și necesită opțiunea `--force`. De obicei, acest lucru nu este acceptabil pentru ramura `main`, dar poate fi acceptabil pentru o ramură tematică înainte de fuzionarea cu ramura `main`.

Comanda git	funcție
`git init`	creează depozitul (local)
`git clone URL`	clonează depozitul de la distanță într-un depozit local cu arborele de lucru
`git pull origin main`	actualizează ramura locală `main` cu ajutorul depozitului de la distanță `origin`
`git add .`	adaugă fișierul (fișierele) din arborele de lucru la index pentru fișierele preexistente numai în index
`git add -A .`	adaugă fișierul (fișierele) din arborele de lucru la index pentru toate fișierele, inclusiv cele eliminate
`git rm nume-fișier`	elimină fișierul (fișierele) din arborele de lucru și din index
`git commit`	comite modificările pregătite în index către depozitul local
`git commit -a`	adaugă toate modificările din arborele de lucru la index și le salvează în depozitul local (adaugă + salvează)
`git push -u origin branch_name`	actualizează depozitul de la distanță `origin` cu ramura locală `branch_name` (invocare inițială)
`git push origin branch_name`	actualizează depozitul de la distanță `origin` cu ramura locală `branch_name` (invocare ulterioară)
`git diff treeish1 treeish2`	afișează diferența dintre modificarea-comisă *treeish1* și modificarea-comisă *treeish2*
`gitk`	afișează în interfața grafică arborele istoricului ramurilor din depozitul VCS

Tabela 10.12: Comenzile git principale



Atenție
Invocarea unei subcomenzi `git` direct ca „`git-xyz`” din linia de comandă a fost abandonată încă de la începutul anului 2006.

Indicație
Dacă există un fișier executabil `git-foo` în ruta specificată de `$PATH`, introducerea „`git foo`” fără cratimă în linia de comandă invocă acest `git-foo`. Aceasta este o caracteristică a comenzii `git`.

10.5.4 Referințe Git

Consultați următoarele.

- pagina de manual: git(1) (`/usr/share/doc/git-doc/git.html`)
- Manualul utilizatorului git (`/usr/share/doc/git-doc/user-manual.html`)
- Un tutorial introductiv în git (`/usr/share/doc/git-doc/gittutorial.html`)
- Un tutorial introductiv în git - partea a doua (`/usr/share/doc/git-doc/gittutorial-2.html`)
- GIT de zi cu zi cu aproximativ 20 de comenzi (`/usr/share/doc/git-doc/giteveryday.html`)
- Magia Git (`/usr/share/doc/gitmagic/html/index.html`)

Linia de comandă Git	**funcție**
`gitk --all`	arată istoricul complet Git și operează asupra lui, cum ar fi reinițializarea HEAD la o altă modificare-comisă, selectarea plasturilor(corecțiilor), crearea de etichete și ramuri ...
`git stash`	obține arborele de lucru curat fără a pierde date
`git remote -v`	verifică configurarea pentru depozitul de la distanță
`git branch -vv`	verifică configurarea pentru ramură
`git status`	afișează starea arborelui de lucru
`git config -l`	listează configurările git
`git reset --hard HEAD; git clean -x -d -f`	anulează toate modificările din arborele de lucru și le curăță complet
`git rm --cached nume-fișier`	anulează indexul etapizat modificat de `git add nume-fișier`
`git reflog`	obține jurnalul de referință (util pentru recuperarea modificărilor-comise din ramura eliminată)
`git branch new_branch_name HEAD@{6}`	creează o nouă ramură din informațiile jurnalului de referință
`git remote add new_remote URL`	adaugă un depozit la distanță `new_remote` indicat de URL
`git remote rename origin upstream`	redenumește numele depozitului de la distanță din `origin` în `upstream`
`git branch -u upstream/branch_name`	configurează urmărirea la distanță pentru depozitul de la distanță `upstream` și numele ramurii sale `branch_name`.
`git remote set-url origin https://foo/bar.git`	modifică adresa URL a `origin`
`git remote set-url --push upstream DISABLED`	dezactivează push către `upstream` (Editează `.git/config` pentru a reactiva)
`git remote update upstream`	preia actualizările tuturor ramurilor de la distanță din depozitul `upstream`
`git fetch upstream foo:upstream-foo`	creează o ramură locală (posibil orfană) `upstream-foo` ca copie a ramurii `foo` din depozitul `upstream`
`git checkout -b topic_branch ; git push -u topic_branch origin`	crează un nou `topic_branch` și îl introduce în `origin`
`git branch -m oldname newname`	redenumește numele ramurii locale
`git push -d origin branch_to_be_removed`	elimină ramura de la distanță (metodă nouă)
`git push origin :branch_to_be_removed`	elimină ramura de la distanță (metodă veche)
`git checkout --orphan unconnected`	creați o nouă ramură `neconectată`
`git rebase -i origin/main`	reordonează/șterge/comprimă modificările-comise din `origin/main` pentru a curăța istoricul ramurii
`git reset HEAD^; git commit --amend`	comprimă ultimele 2 modificări-comise într-una singură
`git checkout topic_branch ; git merge --squash topic_branch`	comprimă întregul `topic_branch` într-o modificare-comisă
`git fetch --unshallow --update-head-ok origin '+refs/heads/*:refs/heads/*'`	convertește o clonă superficială într-o clonă completă a tuturor ramurilor
`git ime`	împarte ultima modificare-comisă într-o serie de modificări-comise mai mici, fișier cu fișier etc. (este necesar pachetul `imediff`)
`git repack -a -d; git prune`	reîmpachetează depozitul local într-un singur pachet (acest lucru poate limita șansele de recuperare a datelor pierdute din ramura ștersă etc.)

Tabela 10.13: Sfaturi Git

10.5.5 Alte sisteme de control al versiunii

Sistemele de control al versiunilor (VCS) sunt uneori cunoscute sub denumirea de sisteme de control al reviziilor (RCS) sau sisteme de gestionare a configurației software (SCM).

Iată un rezumat al celorlalte sisteme de control al versiunilor (VCS) notabile, altele decât Git, din sistemul Debian.

pachet	popcon(popularitate) și instrument	dimensiune	instrument	tipul de VCS	comentariu
`mercurial`	V:3, I:25	1375	Mercurial	distribut	DCVS scris în Python și un pic în C
`darcs`	V:0.2, I:3.1	38856	Darcs	distribut	DVCS cu algebră inteligentă a plasturilor/corecțiilor (lent)
`tla`	V:0.03, I:0.67	1022	GNU arch	distribut	DVCS în principal de Tom Lord (istoric)
`bzr`	V:0.0, I:3.9	28	GNU Bazaar	distribut	DVCS influențat de `tla` scris în Python (istoric)
`subversion`	V:9, I:54	4856	Subversion	la distanță	„CVS făcut corect”, noul standard VCS la distanță (istoric)
`cvs`	V:3, I:23	4835	CVS	la distanță	standardul anterior VCS la distanță (istoric)
`tkcvs`	V:0.11, I:0.83	34	CVS, …	la distanță	interfață grafică de afișare a arborelui depozitului VCS (CVS, Subversion, RCS)
`rcs`	V:1.6, I:9.0	578	RCS	local	”Unix SCCS bine făcut” (istoric)
`cssc`	V:0.01, I:0.28	2044	CSSC	local	clonă de Unix SCCS (istoric)

Tabela 10.14: Lista altor instrumente pentru sistemul de control al versiunilor

Capitolul 11

Conversia datelor

Sunt descrise instrumente și sfaturi pentru conversia formatelor de date în sistemul Debian.

Instrumentele bazate pe standarde sunt în stare foarte bună, dar suportul pentru formatele de date proprietare este limitat.

11.1 Instrumente de conversie a datelor textuale

Următoarele pachete pentru conversia datelor în format text mi-au atras atenția.

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
`libc6`	V:936, I:1000	5483	set de caractere	convertor de codificare a textului între diferite configurații regionale prin iconv(1) (fundamental)
`recode`	V:2, I:12	528	set de caractere + eol	convertor de codificare a textului între diferite configurații regionale (versatil, mai multe nume alternative(alias) și caracteristici)
`konwert`	V:2, I:42	137	set de caractere	convertor de codificare a textului între diferite configurații regionale (elegant)
`nkf`	V:0.4, I:8.6	360	set de caractere	traducător de seturi de caractere pentru limba japoneză
`tcs`	V:0.01, I:0.14	518	set de caractere	traducător de seturi de caractere
`unaccent`	V:0.03, I:0.27	35	set de caractere	înlocuiește literele accentuate cu echivalentul lor neaccentuat
`tofrodos`	V:1, I:12	50	eol (final de linie)	convertor de format de text între DOS și Unix: fromdos(1) și todos(1)
`macutils`	V:0.04, I:0.39	319	eol (final de linie)	convertor de format de text între Macintosh și Unix: frommac(1) și tomac(1)

Tabela 11.1: Lista instrumentelor de conversie a datelor textuale

11.1.1 Conversia unui fișier text cu iconv

Indicație
iconv(1) este furnizat ca parte a pachetului `libc6` și este întotdeauna disponibil pe practic toate sistemele de tip Unix pentru a converti codificarea caracterelor.

Puteți converti codificările unui fișier text cu iconv(1) după cum urmează.

```
$ iconv -f encoding1 -t encoding2 input.txt >output.txt
```

Valorile de codificare nu țin cont de diferențele între majuscule și minuscule și ignoră „-” și „_” pentru potrivire. Codificările acceptate pot fi verificate cu comanda „`iconv -l`”.

valoarea codificării	utilizare
ASCII	American Standard Code for Information Interchange (Codul american standard pentru schimbul de informații), cod de 7 biți fără caractere accentuate
UTF-8	standardul multilingv actual pentru toate sistemele de operare moderne
ISO-8859-1	standardul vechi pentru limbile vest-europene, ASCII + caractere accentuate
ISO-8859-2	standardul vechi pentru limbile est-europene, ASCII + caractere accentuate
ISO-8859-15	standardul vechi pentru limbile vest-europene, ISO-8859-1 cu simbolul euro
CP850	pagina de coduri 850, caractere Microsoft DOS cu grafică pentru limbile vest-europene, variantă a ISO-8859-1
CP932	pagina de coduri 932, variantă a Shift-JIS în stil Microsoft Windows pentru limba japoneză
CP936	pagina de coduri 936„ varianta în stilul Microsoft Windows a GB2312, GBK sau GB18030 pentru chineză simplificată
CP949	pagina de coduri 949, variantă în stil Microsoft Windows a codificării EUC-KR sau Unified Hangul Code pentru limba coreeană
CP950	pagina de coduri 950, varianta în stilul Microsoft Windows a Big5 pentru limba chineză tradițională
CP1251	pagina de coduri 1251, codificare în stil Microsoft Windows pentru alfabetul chirilic
CP1252	pagina de coduri 1252, variantă în stilul Microsoft Windows a ISO-8859-15 pentru limbile vest-europene
KOI8-R	vechiul standard rus UNIX pentru alfabetul chirilic
ISO-2022-JP	codificarea standard pentru mesajele electronice în limba japoneză, care utilizează numai coduri de 7 biți
eucJP	vechiul cod standard japonez UNIX pe 8 biți, complet diferit de Shift-JIS
Shift-JIS	standardul JIS X 0208 anexa 1 pentru limba japoneză (a se vedea CP932)

Tabela 11.2: Lista valorilor de codificare și utilizarea acestora

Notă
Unele codificări sunt acceptate numai pentru conversia datelor și nu sunt utilizate ca valori pentru configurația regională (Secțiune 8.1).

Pentru seturile de caractere care se încadrează într-un singur octet, cum ar fi seturile de caractere ASCII și ISO-8859, codificarea caracterelor înseamnă aproape același lucru cu setul de caractere.

Pentru seturile de caractere cu multe caractere, cum ar fi JIS X 0213 pentru japoneză sau Universal Character Set (UCS, Unicode, ISO-10646-1) pentru practic toate limbile, există multe scheme de codificare pentru a le încadra în secvența de octeți de date.

- EUC și ISO/IEC 2022 (de asemenea cunoscut sub numele de JIS X 0202) pentru limba japoneză
- UTF-8, UTF-16/UCS-2 și UTF-32/UCS-4 pentru Unicode

Pentru acestea, există diferențe clare între setul de caractere și codificarea caracterelor.

Pagina de coduri este utilizată ca sinonim pentru tabelele de codificare a caracterelor pentru unele tabele specifice anumitor furnizori.

Notă
Vă rugăm să rețineți că majoritatea sistemelor de codificare utilizează același cod ASCII pentru caracterele de 7 biți. Există însă și câteva excepții. Dacă convertiți programe C japoneze vechi și date URL din formatul de codificare numit în mod obișnuit shift-JIS în format UTF-8, utilizați „`CP932`” ca nume de codificare în loc de „`shift-JIS`” pentru a obține rezultatele dorite: `0x5C` → „\” și `0x7E` → „~”. În caz contrar, acestea sunt convertite în caractere greșite.

Indicație
recode(1) poate fi utilizat și oferă mai multe funcționalități decât iconv(1), fromdos(1), todos(1), frommac(1) și tomac(1) combinate. Pentru mai multe informații, consultați „`info recode`”.

11.1.2 Verificarea fișierului pentru a fi UTF-8 cu iconv

Puteți verifica dacă un fișier text este codificat în UTF-8 cu iconv(1) după cum urmează.

```
$ iconv -f utf8 -t utf8 input.txt >/dev/null || echo "non-UTF-8 found"
```

Indicație
Utilizați opțiunea „`--verbose`” din exemplul de mai sus pentru a găsi primul caracter non-UTF-8.

11.1.3 Conversia numelor de fișiere cu iconv

Iată un exemplu de script pentru a converti codificarea numelor de fișiere create în sisteme de operare mai vechi în codificarea modernă UTF-8 într-un singur director.

```
#!/bin/sh
ENCDN=iso-8859-1
for x in *;
 do
 mv "$x" "$(echo "$x" | iconv -f $ENCDN -t utf-8)"
done
```

Variabila „`$ENCDN`” specifică codificarea originală utilizată pentru numele fișierelor în sistemele de operare mai vechi, ca în Tabel 11.2.

Pentru cazuri mai complicate, vă rugăm să montați un sistem de fișiere (de exemplu, o partiție pe un disc) care conține astfel de nume de fișiere cu codificare adecvată, utilizând opțiunea mount(8) (a se vedea Secțiune 8.1.3) și copiați întregul conținut al acestuia într-un alt sistem de fișiere montat ca UTF-8 cu comanda „`cp -a`”.

platforma	codul EOL	control	zecimal	hexazecimal
Debian (unix)	LF	^J	10	0A
MSDOS și Windows	CR-LF	^M^J	13 10	0D 0A
Apple's Macintosh	CR	^M	13	0D

Tabela 11.3: Lista stilurilor EOL pentru diferite platforme

11.1.4 Conversie a sfârșitului de linie (EOL)

Formatul fișierului text, în special codul de sfârșit de linie (EOL), depinde de platformă.

Programele de conversie a formatului EOL, fromdos(1), todos(1), frommac(1) și tomac(1), sunt destul de utile. recode(1) este, de asemenea, util.

Notă
Unele date din sistemul Debian, cum ar fi datele paginii wiki pentru pachetul `python-moinmoin`, utilizează codul EOL în stil MSDOS CR-LF. Prin urmare, regula de mai sus este doar o regulă generală.

Notă
Majoritatea editorilor (de exemplu, `vim`, `emacs`, `gedit`, …) pot gestiona în mod transparent fișierele în stil MSDOS EOL.

Indicație
Utilizarea „`sed -e '/\r$/!s/$/\r/'`" în locul todos(1) este mai potrivită atunci când doriți să unificați stilul EOL la stilul MSDOS din stilul mixt MSDOS și Unix. (de exemplu, după îmbinarea a două fișiere în stil MSDOS cu diff3(1).) Acest lucru se datorează faptului că `todos` adaugă CR la toate liniile.

11.1.5 Conversie a tabulatoarelor (TAB)

Există câteva programe specializate populare pentru conversia codurilor de tabulatoare.

funcție	`bsdmainutils`	`coreutils`
convertește tabulatorul în spații	"`col -x`"	`expand`
convertește spațiile în tabulator	"`col -h`"	`unexpand`

Tabela 11.4: Lista comenzilor de conversie a tabulatoarelor (TAB) din pachetele `bsdmainutils` și `coreutils`

indent(1) din pachetul `indent` reformatează complet spațiile albe din programul C.

Programele de editare precum `vim` și `emacs` pot fi utilizate și pentru conversia tabulatoarelor (TAB). De exemplu, cu `vim`, puteți extinde TAB cu secvența de comenzi „`:set expandtab`" și „`:%retab`". Puteți reveni la starea inițială cu secvența de comenzi „`:set noexpandtab`" și „`:%retab!`".

11.1.6 Editori cu conversie automată

Editorii moderni inteligenți, precum programul `vim`, sunt destul de inteligenți și se descurcă bine cu orice sistem de codificare și orice format de fișier. Pentru o compatibilitate optimă, ar trebui să utilizați acești editori în mediul UTF-8 din consola compatibilă cu UTF-8.

Un fișier text Unix vechi din Europa de Vest, „`u-file.txt`", stocat în codificarea latin1 (iso-8859-1), poate fi editat simplu cu `vim` după cum urmează.

```
$ vim u-file.txt
```

Acest lucru este posibil deoarece mecanismul de detectare automată a codificării fișierului în `vim` presupune mai întâi codificarea UTF-8 și, dacă aceasta eșuează, presupune că este latin1.

Un fișier text vechi Unix în poloneză, „`pu-file.txt`”, stocat în codificarea latin2 (iso-8859-2), poate fi editat cu `vim` după cum urmează.

```
$ vim '+e ++enc=latin2 pu-file.txt'
```

Un fișier text vechi UNIX în japoneză, „`ju-file.txt`”, stocat în codificarea eucJP, poate fi editat cu `vim` după cum urmează.

```
$ vim '+e ++enc=eucJP ju-file.txt'
```

Un fișier text vechi MS-Windows în japoneză„`jw-file.txt`”, stocat în așa-numita codificare shift-JIS (mai precis: CP932) poate fi editat cu `vim` după cum urmează.

```
$ vim '+e ++enc=CP932 ++ff=dos jw-file.txt'
```

Când un fișier este deschis cu opțiunile „`++enc`” și „`++ff`”, comanda „`:w`” din linia de comandă Vim îl salvează în formatul original și suprascrie fișierul original. De asemenea, puteți specifica formatul de salvare și numele fișierului în linia de comandă Vim, de exemplu, „`:w ++enc=utf8 new.txt`”.

Consultați mbyte.txt „suport pentru text multi-octeți” în ajutorul în Internet al `vim` și Tabel 11.2 pentru valorile configurației regionale utilizate cu „`++enc`”.

Familia de programe `emacs` poate îndeplini funcții echivalente.

11.1.7 Extragerea textului simplu

Următorul cod citește o pagină web într-un fișier text. Acest lucru este foarte util atunci când copiați configurații de pe web sau aplicați instrumente text Unix de bază, cum ar fi grep(1) pe pagina web.

```
$ w3m -dump https://www.remote-site.com/help-info.html >textfile
```

În mod similar, puteți extrage date text simplu din alte formate utilizând următoarele.

11.1.8 Evidențierea și formatarea datelor în text simplu

Puteți evidenția și formata datele în text simplu în felul următor.

11.2 Date XML

The Extensible Markup Language (XML) este un limbaj de marcare pentru documente care conțin informații structurate.

A se vedea informațiile introductive la XML.COM.

- „Ce este XML?”
- „Ce este XSLT?”
- „Ce este XSL-FO?”
- „Ce este XLink?”

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	funcție
w3m	V:10, I:129	2853	html→text	convertor HTML în text cu comanda „w3m -dump”
html2text	V:3, I:66	298	html→text	convertor avansat HTML în text (ISO 8859-1)
lynx	V:29, I:455	2031	html→text	convertor HTML în text cu comanda „lynx -dump”
elinks	V:2, I:15	1791	html→text	convertor HTML în text cu comanda „elinks -dump”
links	V:2, I:20	2321	html→text	convertor HTML în text cu comanda „links -dump”
links2	V:0.8, I:9.4	5466	html→text	convertor HTML în text cu comanda „links2 -dump”
catdoc	V:16, I:170	682	MSWord→text,TeX	convertește fișierele MSWord în text simplu sau TeX
antiword	V:0.9, I:6.3	596	MSWord→text,ps	convertește fișierele MSWord în text simplu sau ps
unhtml	V:0.04, I:0.49	40	html→text	elimină etichetele de marcaj dintr-un fișier HTML
odt2txt	V:1, I:19	60	odt→text	convertor din text OpenDocument în text

Tabela 11.5: Lista instrumentelor pentru extragerea datelor în text simplu

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
vim-runtime	V:17, I:355	38849	evidențiere	MACRO Vim pentru a converti codul sursă în HTML cu ”:source $VIMRUNTIME/syntax/html.vim”
cxref	V:0.01, I:0.21	1191	c→html	convertor pentru programul C în latex și HTML (limbajul C)
src2tex	V:0.02, I:0.16	1799	evidențiere	convertește multe coduri sursă în TeX (limbajul C)
source-highlight	V:0.4, I:2.9	2131	evidențiere	convertește multe coduri sursă în HTML, XHTML, LaTeX, Texinfo, secvențe de eludare culori ANSI și fișiere DocBook cu evidențiere (C++)
highlight	V:0.4, I:2.8	1412	evidențiere	convertește multe coduri sursă în fișiere HTML, XHTML, RTF, LaTeX, TeX sau XSL-FO cu evidențiere (C++)
grc	V:0.9, I:5.7	208	text→color	instrument generic de colorare pentru orice (Python)
pandoc	V:11, I:47	209559	text→any	convertor general de marcaj (Haskell)
python3-docutils	V:13, I:51	2013	text→any	instrument de formatare de documente de text ReStructurat în XML (Python)
markdown	V:0.4, I:5.7	56	text→html	instrument de formatare de documente text markdown în (X)HTML (Perl)
asciidoctor	V:0.5, I:4.6	101	text→any	instrument de formatare de documente text AsciiDoc în XML/HTML (Ruby)
python3-sphinx	V:7, I:27	3234	text→any	sistem de publicare a documentelor bazat pe text ReStructurat (Python)
hugo	V:0.8, I:5.0	66708	text→html	sistem de publicare a siturilor statice bazat pe markdown (Go)

Tabela 11.6: Lista instrumentelor pentru evidențierea datelor în text simplu

11.2.1 Sfaturi de bază pentru XML

Textul XML arată aproximativ precum HTML. Acesta ne permite să gestionăm mai multe formate de ieșire pentru un document. Un sistem XML simplu este pachetul `docbook-xsl`, care este utilizat aici.

Fiecare fișier XML începe cu o declarație XML standard, după cum urmează.

```
<?xml version="1.0" encoding="UTF-8"?>
```

Sintaxa de bază pentru un element XML este marcată după cum urmează.

```
<name attribute="value">content</name>
```

Elementul XML cu conținut gol este marcat în următoarea formă scurtă.

```
<name attribute="value" />
```

„`attribute="value"`” din exemplele de mai sus sunt opționale.

Secțiunea de comentarii în XML este marcată după cum urmează.

```
<!-- comment -->
```

În afară de adăugarea de marcaje, XML necesită o conversie minoră a conținutului utilizând entități predefinite pentru următoarele caractere.

entitate predefinită	caracterul care urmează să fie convertit în
`"`	" : ghilimele duble
`'`	' : apostrof
`<`	< : mai mic decât
`>`	> : mai mare decât
`&`	& : semnul &

Tabela 11.7: Lista entităților predefinite pentru XML



Atenție
„<” sau „&” nu pot fi utilizate în atribute sau elemente.

Notă
Când se utilizează entități definite de utilizator în stil SGML, de exemplu „`&anumită-etichetă;`”, prima definiție prevalează asupra celorlalte. Definiția entității este exprimată în „`<!ENTITY anumită-etichetă "valoare entitate">`”.

Notă
Atâta timp cât marcajul XML este realizat în mod consecvent cu un anumit set de nume de etichete (fie unele date ca conținut, fie valori de atribute), conversia într-un alt XML este o sarcină trivială folosind Extensible Stylesheet Language Transformations (XSLT).

11.2.2 Procesarea XML

Există multe instrumente disponibile pentru procesarea fișierelor XML, cum ar fi Extensible Stylesheet Language (XSL).

Practic, odată ce ați creat un fișier XML bine structurat, îl puteți converti în orice format folosind Extensible Stylesheet Language Transformations (XSLT).

Extensible Stylesheet Language for Formatting Objects (XSL-FO) este considerat a fi soluția pentru formatare. Pachetul `fop` este nou în arhiva Debian `main` datorită dependenței sale de limbajul de programare Java. Deci, codul LaTeX este de obicei generat din XML folosind XSLT, iar sistemul LaTeX este utilizat pentru a crea fișiere imprimabile, cum ar fi DVI, PostScript și PDF.

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
`docbook-xml`	V:15, I:407	2126	xml	definirea tipului de document XML („Document Type Definition”: DTD) pentru DocBook
`docbook-xsl`	V:15, I:144	14823	xml/xslt	foi de stil XSL pentru procesarea DocBook XML în diverse formate de ieșire cu XSLT
`xsltproc`	V:15, I:71	83	xslt	procesor de linie de comandă XSLT (XML → XML, HTML, text simplu etc.)
`xmlto`	V:0.5, I:8.0	124	xml/xslt	convertor XML-în-orice cu XSLT
`fop`	V:0.7, I:8.0	281	xml/xsl-fo	convertește fișiere Docbook XML în PDF
`dblatex`	V:0.9, I:5.4	4635	xml/xslt	convertește fișiere Docbook în documente DVI, PostScript, PDF cu XSLT
`dbtoepub`	V:0.04, I:0.54	37	xml/xslt	convertor DocBook XML în .epub

Tabela 11.8: Lista instrumentelor XML

Deoarece XML este un subset al Standard Generalized Markup Language (SGML), acesta poate fi procesat de instrumentele extinse disponibile pentru SGML, cum ar fi Document Style Semantics and Specification Language (DSSSL).

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
`openjade`	V:1, I:21	1066	`dsssl`	procesor DSSSL conform standardului ISO/IEC 10179:1996 (cea mai recentă versiune)
`docbook-dsssl`	V:0.5, I:7.4	2594	`xml/dsssl`	foi de stil DSSSL pentru procesarea DocBook XML în diverse formate de ieșire cu DSSSL
`docbook-utils`	V:0.3, I:5.2	287	`xml/dsssl`	instrumente pentru fișiere DocBook, inclusiv conversia în alte formate (HTML, RTF, PS, man, PDF) cu comenzi `docbook2*` cu DSSSL

Tabela 11.9: Lista instrumentelor DSSSL

Indicație
`yelp`-ul GNOME este uneori util pentru a citi direct fișiere DocBook, deoarece le redă decent în X.

11.2.3 Extragerea datelor XML

Puteți extrage date HTML sau XML din alte formate utilizând următoarele.

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
man2html	V:0.1, I:1.2	142	manpage→html	convertor dintr-o pagină de manual în HTML (suport CGI)
doclifter	V:0.00, I:0.05	487	troff→xml	convertor din troff în DocBook XML
texi2html	V:0.2, I:2.8	1847	texi→html	convertor din Texinfo în HTML
info2www	V:0.8, I:1.4	76	info→html	convertor din GNU info în HTML (suport CGI)
wv	V:0.2, I:2.3	733	MSWord→any	convertor de documente din Microsoft Word în HTML, LaTeX etc.
unrtf	V:0.3, I:2.8	159	rtf→html	convertor de documente din RTF în HTML etc.
wp2x	V:0.00, I:0.08	200	WordPerfect→any	fișiere WordPerfect 5.0 și 5.1 în TeX, LaTeX, troff, GML și HTML

Tabela 11.10: Lista instrumentelor de extragere a datelor XML

11.2.4 Analizarea statică (lint) a datelor XML

Fișierele HTML care nu sunt XML pot fi convertite în XHTML, care este un exemplu de XML bine format. XHTML poate fi procesat cu instrumente XML.

Sintaxa fișierelor XML și validitatea adreselor URL găsite în acestea pot fi verificate.

pachet	popcon(popularitate)	dimensiune	funcție	descriere
libxml2-utils	V:60, I:207	213	xml↔html↔xhtml	instrument XML pentru linia de comandă cu xmllint(1) (verificare sintactică, reformatare, analizare statică (lint), …)
tidy	V:0.9, I:6.9	79	xml↔html↔xhtml	verificator de sintaxă HTML și instrument de reformatare
weblint-perl	V:0.05, I:0.90	32	lint	verificator de sintaxă și de stil minimal pentru HTML
linklint	V:0.04, I:0.44	343	verificare legături	instrument rapid de verificare a legăturilor și întreținere a siturilor web

Tabela 11.11: Lista instrumentelor de imprimare elegantă XML

Odată ce s-a generat un fișier XML corespunzător, puteți utiliza tehnologia XSLT pentru a extrage date pe baza contextului de marcare etc.

11.3 Tehnoredactarea (formatarea și pregătirea textului)

Programul Unix troff, dezvoltat inițial de AT&T, poate fi folosit pentru formatarea simplă a textului. Acesta este utilizat de obicei pentru crearea paginilor de manual.

TeX, creat de Donald Knuth, este un instrument de tehnoredactare extrem de puternic și reprezintă standardul de facto. LaTeX, scris inițial de Leslie Lamport, permite accesul la funcționalitățile avansate ale TeX-ului.

11.3.1 Tehnoredactare roff

În mod tradițional, roff este principalul sistem de procesare a textului din Unix. A se vedea roff(7), groff(7), groff(1), grotty(1), troff(1), groff_mdoc(7), groff_man(7), groff_ms(7), groff_me(7), groff_mm(7), și "`info groff`".

Puteți citi sau imprima un tutorial și un ghid de referință util privind macrocomanda „`-me`" din directorul „`/usr/share/doc/gr` prin instalarea pachetului `groff`.

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
texlive	V:1, I:27	55	(La)TeX	sistemul TeX pentru tehnoredactare, previzualizare și imprimare a textului
lilypond	V:0.7, I:5.4	16924	(La)TeX	compozitor de partituri muzicale
groff	V:2, I:23	16536	troff	sistemul de formatare a textului GNU troff

Tabela 11.12: Lista instrumentelor de tehnoredactare

Indicație
„groff -Tascii -me -” generează o ieșire sub formă de text simplu cu cod de eludare ANSI. Dacă doriți să obțineți o ieșire similară cu pagina de manual, cu multe caractere „^H” și „_”, utilizați în schimb „GROFF_NO_SGR=1 groff -Tascii -me -”.

Indicație
Pentru a elimina caracterele „^H” și „_” dintr-un fișier text generat de groff, filtrați-l cu comanda „col -b -x”.

11.3.2 TeX/LaTeX

Distribuția software TeX Live oferă un sistem TeX complet. Metapachetul texlive pune la dispoziție o selecție adecvată de pachete TeX Live, care ar trebui să fie suficientă pentru cele mai comune sarcini.

Există numeroase referințe disponibile pentru TeX și LaTeX.

- The teTeX HOWTO: The Linux-teTeX Local Guide (Rețetarul teTeX: Ghidul local pentru Linux-teTeX), în engleză
- tex(1)
- latex(1)
- texdoc(1)
- texdoctk(1)
- „The TeXbook”, de Donald E. Knuth, (Addison-Wesley)
- „LaTeX – A Document Preparation System”, de Leslie Lamport, (Addison-Wesley)
- „The LaTeX Companion”, de Goossens, Mittelbach, Samarin, (Addison-Wesley)

Acesta este cel mai performant mediu de tehnoredactare. Multe procesoare SGML îl utilizează ca procesor de text de fundal. Lyx, furnizat de pachetul lyx, și GNU TeXmacs, furnizat de pachetul texmacs, oferă un mediu de editare WYSIWYG pentru LaTeX, în timp ce mulți folosesc Emacs și Vim ca editor de sursă.

Există numeroase resurse disponibile în Internet.

- The TEX Live Guide - TEX Live 2007 („/usr/share/doc/texlive-doc-base/english/texlive-en/live.html” din pachetul texlive-doc-base)
- A Simple Guide to Latex/Lyx (Un ghid simplu pentru Latex/Lyx), în engleză
- Word Processing Using LaTeX (Procesarea textelor cu ajutorul LaTeX), în engleză

Când documentele devin mai mari, uneori TeX poate genera erori. Pentru a remedia această problemă, trebuie să măriți dimensiunea memoriei tampon în fișierul „/etc/texmf/texmf.cnf” (sau, mai bine, să editați fișierul „/etc/texmf/texmf.d/95NonPath” și să rulați comanda update-texmf(8)).

Notă
Sursa TeX a cărții „The TeXbook” este disponibilă la www.ctan.org situl arhivei TeX pentru fișierul texbook.tex. Acest fișier conține majoritatea macrocomenzilor necesare. Am auzit că puteți procesa acest document cu tex(1) după ce comentați liniile 7-10 și adăugați „`\input manmac \proofmodefalse`”. Se recomandă cu tărie să cumpărați această carte (și toate celelalte cărți ale lui Donald E. Knuth) în loc să folosiți versiunea online, dar sursa este un exemplu excelent de introducere în TeX!

11.3.3 Imprimarea paginii de manual într-un format atractiv

Puteți imprima o pagină de manual în format PostScript folosind una dintre următoarele comenzi.

```
$ man -Tps some_manpage | lpr
```

11.3.4 Crearea unei pagini de manual

Deși este posibil să scrieți o pagină de manual (manpage) în formatul simplu troff, există puține pachete auxiliare care să vă ajute să o creați.

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
docbook-to-man	V:0.6, I:5.3	189	SGML → manpage	convertor din DocBook SGML în macrocomenzi roff man
help2man	V:0.5, I:6.1	542	text → manpage	generator automat de pagini de manual din „--help”
info2man	V:0.02, I:0.18	134	info → manpage	convertor de la GNU Info la POD sau pagini de manual
txt2man	V:0.04, I:0.61	112	text → manpage	convertește textul ASCII simplu în formatul paginilor de manual

Tabela 11.13: Lista pachetelor utile pentru crearea paginilor de manual

11.4 Date imprimabile

Datele destinate imprimării sunt exprimate în formatul PostScript în sistemul Debian. Common Unix Printing System (CUPS) utilizează Ghostscript ca program de procesare a datelor pentru imprimantele care nu acceptă PostScript.

Datele care pot fi imprimate pot fi expresate și în formatul PDF în sistemul Debian actual.

Fișierele PDF pot fi vizualizate, iar câmpurile formularelor din acestea pot fi completate folosind instrumente de vizualizare cu interfață grafică, precum Evince și Okular (a se vedea Secțiune 7.4); precum și în navigatoarele moderne, cum ar fi Chromium.

Fișierele PDF pot fi editate folosind anumite instrumente grafice, cum ar fi LibreOffice, Scribus și Inkscape (a se vedea Secțiune 11.6).

Indicație
Puteți citi un fișier PDF cu GIMP și îl puteți converti în format PNG folosind o rezoluție mai mare de 300 dpi. Acesta poate fi folosit ca imagine de fundal pentru LibreOffice pentru a obține o imprimare modificată conform dorințelor cu efort minim.

11.4.1 Ghostscript

Elementul central al procesării datelor destinate imprimării este interpretul Ghostscript PostScript (PS), care generează imagini-rastru.

pachet	popcon(popularitate)	dimensiune	descriere
ghostscript	V:165, I:559	177	interpretul GPL Ghostscript de PostScript/PDF
ghostscript-x	V:0, I:13	88	interpretul GPL Ghostscript de PostScript/PDF - suport pentru afișare în sistemul X
libpoppler162	V:14, I:22	5313	bibliotecă de redare PDF derivată din vizorul de fișiere PDF xpdf
libpoppler-glib8t64	V:77, I:323	612	bibliotecă de redare PDF (bibliotecă partajată bazată pe GLib)
poppler-data	V:161, I:582	13086	Biblioteca CMaps pentru redarea fișierelor PDF (pentru suport CJK: Adobe-*)

Tabela 11.14: Lista interpreților Ghostscript PostScript

Indicație
Comanda «gs -h» poate afișa configurația Ghostscript.

11.4.2 Fuzionarea a două fișiere PS sau PDF

Puteți fuziona două fișiere PostScript (PS) sau Portable Document Format (PDF) folosind gs(1) din Ghostscript.

```
$ gs -q -dNOPAUSE -dBATCH -sDEVICE=pswrite -sOutputFile=bla.ps -f foo1.ps foo2.ps
$ gs -q -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -sOutputFile=bla.pdf -f foo1.pdf foo2.pdf
```

Notă
Fișierul PDF, care este un format de date imprimabil, utilizat pe scară largă pe mai multe platforme, este, în esență, formatul PS comprimat, cu câteva caracteristici și extensii suplimentare.

Indicație
Pentru linia de comandă, psmerge(1) și alte comenzi din pachetul psutils sunt utile pentru manipularea documentelor PostScript. De asemenea, pdftk(1) din pachetul pdftk este util pentru manipularea documentelor PDF.

11.4.3 Instrumente pentru date imprimabile

Următoarele pachete pentru instrumente de date imprimabile mi-au atras atenția.

11.4.4 Imprimarea cu CUPS

Atât comanda lp(1), cât și comanda lpr(1), oferite de Common Unix Printing System (CUPS), oferă opțiuni pentru imprimarea personalizată a datelor destinate imprimării.

Puteți imprima 3 copii ale unui fișier folosind una dintre următoarele comenzi.

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
poppler-utils	V:169, I:476	804	pdf→ps,text	Ustensile PDF: pdftops, pdfinfo, pdfimages, pdftotext, pdffonts
psutils	V:4, I:48	34	ps→ps	Instrumente de conversie a documentelor PostScript
poster	V:0.1, I:1.6	57	ps→ps	creează postere mari din pagini PostScript
enscript	V:1, I:10	2138	text→ps, html, rtf	convertește textul ASCII în PostScript, HTML, RTF sau imprimare elegantă (pretty-print)
a2ps	V:0.7, I:6.7	4109	text→ps	convertor „orice în PostScript” și instrument de imprimare elegantă
pdftk	V:1, I:20	28	pdf→pdf	instrument de conversie a documentelor PDF: pdftk
html2ps	V:0.2, I:1.7	256	html→ps	convertor de la HTML la PostScript
gnuhtml2latex	V:0.03, I:0.57	26	html→latex	convertor de la HTML la LaTeX
latex2rtf	V:0.1, I:1.9	495	latex→rtf	convertește documentele din format LaTeX în format RTF, care pot fi citite în MS Word
ps2eps	V:2, I:31	95	ps→eps	convertor de la PostScript la EPS (Encapsulated PostScript)
e2ps	V:0.01, I:0.11	104	text→ps	convertor de text în PostScript cu suport pentru codificarea limbii japoneze
impose+	V:0.1, I:1.6	118	ps→ps	ustensile PostScript
trueprint	V:0.00, I:0.07	148	text→ps	imprimă elegant multe coduri sursă (C, C++, Java, Pascal, Perl, Pike, Sh și Verilog) în format PostScript. (Limbajul C)
pdf2svg	V:0.2, I:2.9	33	pdf→svg	convertor din format PDF în format Scalable vector graphics (SVG) -- Grafică vectorială scalabilă
pdftoipe	V:0.01, I:0.43	69	pdf→ipe	convertor din format PDF în formatul XML al IPE

Tabela 11.15: Lista instrumentelor pentru date imprimabile

```
$ lp -n 3 -o Collate=True filename
```

```
$ lpr -#3 -o Collate=True filename
```

Puteți personaliza și mai mult funcționarea imprimantei folosind opțiuni precum „`-o number-up=2`”, „`-o page-set=even`”, „`-o page-set=odd`”, „`-o scaling=200`”, „`-o natural-scaling=200`” etc., documentate în Imprimare din linia de comandă și opțiuni.

11.5 Conversia datelor de poștă electronică

Următoarele pachete pentru conversia datelor de poștă electronică mi-au atras atenția.

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
`sharutils`	V:2, I:27	1436	mail	shar(1), unshar(1), uuencode(1), uudecode(1)
`mpack`	V:0.7, I:7.4	109	`MIME`	codarea și decodarea mesajelor MIME: mpack(1) și munpack(1)
`tnef`	V:0.3, I:3.7	103	`ms-tnef`	despachetarea atașamentelor MIME de tip „application/ms-tnef”, care este un format specific Microsoft
`uudeview`	V:0.2, I:1.7	105	`mail`	codificator și decodificator pentru următoarele formate: uuencode, xxencode, BASE64, quoted printable și BinHex

Tabela 11.16: Lista pachetelor utile pentru conversia datelor de poștă electronică

Indicație
Serverul Internet Message Access Protocol versiunea 4 (IMAP4) poate fi utilizat pentru a transfera mesajele din sistemele de poștă electronică proprietare, cu condiția ca programul client de poștă electronică să poată fi configurat să utilizeze, la rândul său, un server IMAP4.

11.5.1 Noțiuni de bază privind datele de poștă electronică

Datele de poștă electronică (SMTP) trebuie să se limiteze la șiruri de date de 7 biți. Astfel, datele binare și datele text de 8 biți sunt codificate în format de 7 biți cu ajutorul Multipurpose Internet Mail Extensions (MIME) și selectarea setului de caractere (a se vedea Tabel 11.2).

Formatul standard de stocare a mesajelor de poștă electronică este mbox, conform standardului RFC2822 (versiunea actualizată a RFC822). A se vedea mbox(5) (furnizat de pachetul `mutt`).

În cazul limbilor europene, pentru mesaje se utilizează de obicei „`Content-Transfer-Encoding: quoted-printable`” cu setul de caractere ISO-8859-1, deoarece nu există prea multe caractere pe 8 biți. Dacă textul european este codificat în UTF-8, este probabil să se utilizeze „`Content-Transfer-Encoding: quoted-printable`”, deoarece este vorba în mare parte de date pe 7 biți.

În cazul limbii japoneze, se folosește de obicei „`Content-Type: text/plain; charset=ISO-2022-JP`” pentru mesaje, pentru a păstra textul în format pe 7 biți. Totuși, sistemele Microsoft mai vechi pot trimite date de poștă electronică în format Shift-JIS fără o declarare corespunzătoare. Dacă textul japonez este codificat în UTF-8, este probabil să se utilizeze Base64, deoarece conține multe date de 8 biți. Situația altor limbi asiatice este similară.

Notă
Dacă datele dvs. de poștă electronică non-Unix sunt accesibile printr-un program client non-Debian capabil să comunice cu un server IMAP4, este posibil să le puteți transfera rulând propriul server IMAP4.

Notă
Dacă utilizați alte formate de stocare a mesajelor, un prim pas util este să le convertiți în format mbox. Un program client versatil, precum mutt(1), vă poate fi de ajutor în acest sens.

Puteți împărți conținutul căsuței poștale în fiecare mesaj folosind procmail(1) și formail(1).

Fiecare mesaj de poștă electronică poate fi despachetat folosind comanda munpack(1) din pachetul `mpack` (sau alte instrumente specializate) pentru a obține conținutul codificat MIME.

11.6 Instrumente pentru date grafice

Deși programele cu interfață grafică, precum gimp(1), sunt foarte puternice, instrumentele de linie de comandă, precum imagemagick(1), sunt destul de utile pentru automatizarea manipulării imaginilor prin intermediul scripturilor.

Formatul de fișier imagine utilizat de facto de aparatele foto digitale este Exchangeable Image File Format (EXIF), care este formatul de fișier imagine JPEG cu etichete de metadate suplimentare. Acesta poate stoca informații precum data, ora și configurările aparatului foto.

Patentul Lempel-Ziv-Welch (LZW) pentru compresia fără pierderi a datelor a expirat. Instrumentele Graphics Interchange Format (GIF) care utilizează metoda de compresie LZW sunt acum disponibile fără restricții în sistemul Debian.

Indicație
Orice aparat foto digital sau scaner cu suport de stocare detașabil funcționează cu Linux prin intermediul cititoarelor de stocare USB, deoarece respectă Regula de proiectare pentru sistemul de fișiere al aparatului foto și utilizează sistemul de fișiere FAT. A se vedea Secțiune 10.1.7.

11.6.1 Instrumente pentru date grafice (metapachet)

Următoarele metapachete constituie un bun punct de plecare pentru căutarea instrumentelor de gestionare a datelor grafice folosind aptitude(8). „Prezentarea pachetelor pentru responsabilii de întreținere ai Debian PhotoTools” poate fi un alt punct de plecare.

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
`education-graphics`	I:0.34	26	`svg,jpeg,…`	metapachet pentru predarea graficii și a artelor plastice.
`open-font-design-toolkit`	I:0.04	9	`ttf,ps,…`	metapachet pentru proiectarea fonturilor deschise

Tabela 11.17: Lista instrumentelor pentru date grafice (metapachet)

Indicație
Căutați mai multe instrumente pentru imagini folosind expresia regulată „`~Gworks-with::image`” în aptitude(8) (a se vedea Secțiune 2.2.6).

11.6.2 Instrumente pentru date grafice (cu interfață grafică)

Următoarele pachete de instrumente cu interfață grafică pentru conversia, editarea și organizarea datelor grafice mi-au atras atenția.

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
gimp	V:34, I:211	33315	image(bitmap)	GNU Image Manipulation Program -- Program GNU de manipulare a imaginilor
xsane	V:9, I:128	1512	image(bitmap)	interfață X11 bazată pe GTK pentru SANE (Scanner Access Now Easy)
scribus	V:1, I:12	32942	ps/pdf/SVG/...	editorul DTP Scribus
libreoffice-draw	V:89, I:418	11418	image(vector)	suita de birou LibreOffice - desen
inkscape	V:12, I:76	113353	image(vector)	editor SVG (Scalable Vector Graphics)
dia	V:1, I:16	3846	image(vector)	editor de diagrame (Gtk)
xfig	V:0.5, I:8.5	7951	image(vector)	instrument pentru generarea interactivă de figuri în mediul X11
gocr	V:0.5, I:3.6	549	image→text	program liber de recunoaștere optică a caracterelor
eog	V:23, I:129	10535	image(Exif)	programul de vizualizare a imaginilor Eye of GNOME
gthumb	V:3, I:11	5162	image(Exif)	vizor de imagini și navigator (GNOME)
geeqie	V:3, I:11	3367	image(Exif)	vizor de imagini care utilizează GTK
shotwell	V:14, I:246	6334	image(Exif)	organizator de fotografii digitale (GNOME)
gwenview	V:41, I:114	6057	image(Exif)	vizor de imagini (KDE)
kamera	I:114	992	image(Exif)	suport pentru camere digitale în aplicațiile KDE
digikam	V:1.5, I:8.3	333	image(Exif)	aplicație de gestionare a fotografiilor digitale pentru KDE
darktable	V:4, I:11	36092	image(Exif)	masă de lumină și cameră obscură virtuale pentru fotografi
hugin	V:0.5, I:5.3	6524	image(Exif)	program de asamblare a fotografiilor panoramice
librecad	V:1, I:13	9164	DXF,...	editor de date CAD 2D
freecad	V:1, I:20	112	DXF,...	editor de date CAD 3D
blender	V:2, I:20	171893	blend,TIFF,VRML,...	editor de conținut 3D pentru animație etc.
mm3d	V:0.02, I:0.22	4123	ms3d,obj,dxf,...	editor de modele 3D bazat pe OpenGL
fontforge	V:0.6, I:5.3	4736	ttf,ps,...	editor de fonturi pentru fonturi PS, TrueType și OpenType
xgridfit	V:0.02, I:0.08	878	ttf	program pentru adaptarea la grilă și ajustarea contururilor fonturilor TrueType

Tabela 11.18: Lista instrumentelor pentru date grafice (cu interfață grafică)

11.6.3 Instrumente pentru date grafice (cu interfață de linie de comandă)

Următoarele pachete de instrumente cu interfață de linie de comandă pentru conversia, editarea și organizarea datelor grafice mi-au atras atenția.

11.7 Conversii diverse de date

Există multe alte programe pentru conversia datelor. Următoarele pachete mi-au atras atenția folosind expresia regulată „~Guse::converting” în aptitude(8) (a se vedea Secțiune 2.2.6).

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
imagemagick	V:7, I:272	80	image(bitmap)	programe de manipulare a imaginilor
graphicsmagick	V:1.0, I:8.3	5945	image(bitmap)	programe de manipulare a imaginilor (o variantă a imagemagick)
netpbm	V:27, I:282	8435	image(bitmap)	instrumente de conversie grafică
libheif-examples	V:0.3, I:3.6	528	heif→jpeg(bitmap)	convertește High Efficiency Image File Format (HEIF) în formatele JPEG, PNG sau Y4M cu comanda heif-convert(1)
icoutils	V:3, I:33	221	png↔ico(bitmap)	convertește pictogramele și cursorii MS Windows în și din formatul PNG (favicon.ico)
pstoedit	V:2, I:37	1075	ps/pdf→image(vector)	convertor de fișiere PostScript și PDF în grafică vectorială editabilă (SVG)
libwmf-bin	V:5, I:80	149	Windows/image (vectorială)	instrumente de conversie a meta-fișierelor Windows (date grafice vectoriale)
fig2sxd	V:0.01, I:0.16	158	fig→sxd(vector)	convertește fișierele XFig în formatul OpenOffice.org Draw
unpaper	V:2, I:16	417	image→image	instrument de post-procesare a paginilor scanate pentru OCR
tesseract-ocr	V:8, I:34	2209	image→text	software OCR liber bazat pe motorul OCR comercial al HP
tesseract-ocr-eng	V:9, I:35	4032	image→text	motor de date OCR: fișiere de limbă tesseract-ocr pentru text în limba engleză
ocrad	V:0.3, I:2.2	608	image→text	program liber de recunoaștere optică a caracterelor
exif	V:3, I:49	335	image(Exif)	instrument de linie de comandă pentru afișarea informațiilor EXIF din fișierele JPEG
exiv2	V:1, I:18	436	image(Exif)	instrument de manipulare a metadatelor EXIF/IPTC
exiftran	V:1, I:11	81	image(Exif)	transformă imaginile JPEG realizate cu aparatul foto digital
exiftags	V:0.2, I:2.5	309	image(Exif)	instrument pentru citirea etichetelor Exif dintr-un fișier JPEG provenit de la un aparat foto digital
exifprobe	V:0.2, I:2.1	506	image(Exif)	citește metadatele din imaginile digitale
dcraw	V:0.7, I:7.2	428	image(Raw)→ppm	decodifică imaginile brute ale aparatului foto digital
findimagedupes	V:0.11, I:0.96	76	image→fingerprint	găsește imagini similare din punct de vedere vizual sau duplicate
ale	V:0.00, I:0.14	850	image→image	fuzionează imagini pentru a îmbunătăți calitatea sau pentru a crea mozaicuri
imageindex	V:0.1, I:1.1	143	image(Exif)→html	generează galerii HTML statice din imagini
outguess	V:0.1, I:1.0	230	jpeg,png	instrument Stenografic universal
jpegoptim	V:0.6, I:5.8	59	jpeg	optimizează fișierele JPEG
optipng	V:2, I:39	187	png	optimizează fișierele PNG, compresie fără pierderi
pngquant	V:1, I:11	62	png	optimizează fișierele PNG, compresie cu pierderi

Tabela 11.19: Lista instrumentelor pentru date grafice (cu interfață de linie de comandă)

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
alien	V:1, I:12	150	rpm/tgz → deb	convertor pentru transformarea pachetelor externe în pachete Debian
freepwing	V:0.00, I:0.01	447	EB → EPWING	convertor de la formatul „Electric Book” (popular în Japonia) la un singur format JIS X 4081 (un subset al formatului EPWING V1)
calibre	V:7, I:24	71216	any → EPUB	convertor de cărți electronice și gestionarea bibliotecii

Tabela 11.20: Lista instrumentelor diverse de conversie a datelor

De asemenea, puteți extrage date din fișiere în format RPM folosind următoarea comandă.

```
$ rpm2cpio file.src.rpm | cpio --extract
```

Capitolul 12

Programare

Vă ofer câteva sfaturi pentru cei care doresc să învețe programarea pe sistemul Debian la un nivel suficient pentru a putea urmări codul sursă al pachetelor. Iată câteva pachete importante și documentația aferentă pentru programare.

Referințele sunt disponibile introducând „`man nume`” după instalarea pachetelor `manpages` și `manpages-dev`. Referințele pentru instrumentele GNU sunt disponibile prin tastarea „`info nume_program`” după instalarea pachetelor de documentație relevante. Este posibil să fie necesar să includeți arhivele `contrib` și `non-free` în plus față de arhiva `main`, deoarece unele documentații GFDL nu sunt considerate conforme cu DFSG.

Vă recomandăm să utilizați instrumentele sistemului de control al versiunilor. Consultați Secțiune 10.5.



Avertisment
Nu folosiți „`test`” ca nume pentru un fișier executabil de testare. „`test`” este o comandă internă a shell-ului.



Atenție
Ar trebui să instalați programele software compilate direct din sursă în „`/usr/local`” sau „`/opt`” pentru a evita suprapunerea cu programele din sistem.

Indicație
Exemplele de cod pentru crearea melodiei „99 Bottles of Beer” ar trebui să vă ofere idei utile pentru aproape toate limbajele de programare.

12.1 Scriptul Shell

Scriptul shell este un fișier text cu dreptul de execuție activat și conține comenzi în următorul format.

```
#!/bin/sh
 ... command lines
```

Prima linie specifică interpretul shell care citește și execută conținutul acestui fișier.

Citirea scripturilor shell este **cea mai bună** modalitate de a înțelege cum funcționează un sistem de tip Unix. Aici vă ofer câteva indicații și sfaturi utile pentru programarea în shell. Consultați „Greșeli în shell” (https://www.greenend.org.uk/rjk/2001/04/shell.html) pentru a învăța din greșeli.

Spre deosebire de modul interactiv al shell-ului (a se vedea Secțiune 1.5 și Secțiune 1.6), scripturile shell utilizează frecvent parametri, condiții și bucle.

12.1.1 Compatibilitate cu shell-ul POSIX

Multe scripturi de sistem pot fi interpretate de oricare dintre shell-urile POSIX (vedeți Tabel 1.13).

- Shell-ul POSIX implicit, neinteractiv, „`/usr/bin/sh`”, este o legătură simbolică care indică spre `/usr/bin/dash` și este utilizat de multe programe din sistem.
- Shell-ul POSIX interactiv implicit este `/usr/bin/bash`.

Evitați să scrieți un script de shell care conține „**bashisme**” (particularități ale shell-ului bash) sau „**zshisme**>” (particularități ale shell-ului zsh), pentru a-l face compatibil cu toate shell-urile POSIX. Puteți verifica acest lucru folosind checkbashisms(1).

Bine: POSIX	**De evitat: „bashismul”**
`if [ "$foo" = "$bar" ] ; then …`	`if [ "$foo" == "$bar" ] ; then …`
`diff -u file.c.orig file.c`	`diff -u file.c{.orig,}`
`mkdir /foobar /foobaz`	`mkdir /foo{bar,baz}`
`funcname() { … }`	`function funcname() { … }`
format octal: ”`\377`”	format hexazecimal: ”`\xff`”

Tabela 12.1: Lista expresiilor tipice din Bash („bashisme”)

Comanda „`echo`” trebuie utilizată cu următoarele precauții, deoarece implementarea sa diferă între comenzile interne ale shell-ului și cele externe.

- Evitați utilizarea oricăror opțiuni ale comenzii «echo» cu excepția „`-n`”.
- Evitați utilizarea secvențelor de eludare în șir, deoarece gestionarea acestora variază.

Notă
Deși opțiunea „`-n`” **nu** este, de fapt, o sintaxă POSIX, ea este acceptată în general.

Indicație
Folosiți comanda „`printf`” în locul comenzii „`echo`” dacă trebuie să includeți secvențe de eludare în șirul de ieșire.

12.1.2 Parametri shell

În scripturile shell se utilizează frecvent parametri speciali ai shell-ului.

Expansiunile parametrilor de bază pe care trebuie să le rețineți sunt următoarele.

Aici, caracterul „`:`” din toți acești operatori este, de fapt, opțional.

- **cu** „`:`” operatorul = verifică dacă **există** și **nu este nul**
- **fără** „`:`” operatorul = verifică doar dacă **există**

prametru shell	**valoare**
`$0`	numele shell-ului sau al scriptului shell
`$1`	primul (1) argument al shell-ului
`$9`	al nouălea (9-lea) argument al shell-ului
`$#`	numărul parametrilor de poziție
`"$*"`	`"$1 $2 $3 $4 … "`
`"$@"`	`"$1" "$2" "$3" "$4" …`
`$?`	starea de ieșire a celei mai recente comenzi
`$$`	PID-ul acestui script de shell
`$!`	PID-ul celei mai recente sarcini de fundal pornite

Tabela 12.2: Lista parametrilor shell-ului

forma expresiei parametrului	**valoarea dacă `var` este definită**	**valoarea dacă `var` nu este definită**
`${var:-string}`	”`$var`”	”`string`”
`${var:+string}`	”`string`”	”`null`”
`${var:=string}`	”`$var`”	„`șir`” (și execută „`var=șir`”)
`${var:?string}`	”`$var`”	afișează „`șir`ul” la **stderr** (și iese cu eroare)

Tabela 12.3: Lista expansiunilor parametrilor shell-ului

forma de substituire a parametrilor	**rezultatul**
`${var%suffix}`	elimină cel mai mic model de sufix
`${var%%suffix}`	elimină cel mai mare model de sufix
`${var#prefix}`	elimină cel mai mic model de prefix
`${var##prefix}`	elimină cel mai mare model de prefix

Tabela 12.4: Lista substituțiilor parametrilor cheie ai shell-ului

12.1.3 Condiționale shell

Fiecare comandă returnează un **cod de ieșire** care poate fi utilizat în expresii condiționale.

- Succes: 0 („True” - adevărat)
- Eroare: diferit de 0 („False” - fals)

Notă
În contextul condițional al shell-ului, „0” înseamnă „Adevărat”, în timp ce în contextul condițional al limbajului C, „0” înseamnă „Fals”.

Notă
„`[`” este echivalentul comenzii `test`, care evaluează argumentele sale până la „`]`” ca o expresie condițională.

Expresiile condiționale de bază pe care trebuie să le rețineți sunt următoarele.

- „*`comandă && dacă_succes_execută_și_această_comandă`* `|| true`”
- „*`comandă || dacă_nu_a_reușit_execută_și_această_comandă`* `|| true`”
- Un fragment de cod cu mai multe linii, ca în exemplul următor

```
if [ conditional_expression ]; then
 if_success_run_this_command
else
 if_not_success_run_this_command
fi
```

Aici a fost necesară adăugarea secvenței „`|| true`” pentru a se asigura că acest script de shell nu se închide accidental la această linie atunci când shell-ul este invocat cu opțiunea „`-e`”.

ecuație	condiție care returnează valoarea logică „adevărat”
`-e` *`fișier`*	*fișierul* există
`-d` *`fișier`*	*fișierul* există și este un director
`-f` *`fișier`*	*fișierul* există și este un fișier obișnuit
`-w` *`fișier`*	*fișierul* există și poate fi accesat pentru scriere
`-x` *`fișier`*	*fișierul* există și este executabil
`fișier1` `-nt` *`fișier2`*	*fișier1* este mai recent decât *fișier2* (data modificării)
`fișier1` `-ot` *`fișier2`*	*fișier1* este mai vechi decât *fișier2* (data modificării)
`fișier1` `-ef` *`fișier2`*	*fișier1* și *fișier2* se află pe același dispozitiv și au același număr de nod-i

Tabela 12.5: Lista operatorilor de comparare a fișierelor în expresia condițională

Operatorii **aritmetici** de comparare a numerelor întregi din expresia condițională sunt „`-eq`”, „`-ne`”, „ `-lt`„, ”`-le`„, ”`-gt`„ și ”`-ge`”.

ecuație	condiție care returnează valoarea logică „adevărat”
`-z` *șir*	lungimea *șirului* este zero
`-n` *șir*	lungimea *șirului* este diferită de zero
șir1 `=` *șir2*	*șirul1* și *șirul2* sunt egale
șir1 `!=` *șir2*	*șirul1* și *șirul2* nu sunt egale
șir1 `<` *șir2*	*șirul1* este sortat înaintea *șirului2* (în funcție de configurația regională)
șir1 `>` *șir2*	*șirul1* este sortat după *șirul2* (în funcție de configurația regională)

Tabela 12.6: Lista operatorilor de comparare a șirurilor în expresia condițională

12.1.4 Bucle shell

În shell-ul POSIX există mai multe expresii de tip buclă care pot fi utilizate.

- „`for x in foo1 foo2 … ; do comanda ; done`” efectuează o buclă prin atribuirea elementelor din lista „`foo1 foo2 …`” variabilei „x” și executarea „`comenzii`”.
- „`while condiție ; do comanda ; done`” repetă „`comanda`” atâta timp cât „`condiția`” este adevărată.
- „`until condiție ; do comanda ; done`” repetă „`comanda`” atâta timp cât „`condiția`” nu este adevărată.
- „`break`” permite ieșirea din buclă.
- „`continue`” permite reluarea următoarei iterații a buclei.

Indicație
Iterarea numerică specifică limbajului C poate fi realizată folosind seq(1) ca generator de tipul „`foo1 foo2 …`”.

Indicație
Consultați Secțiune 9.4.9.

12.1.5 Variabilele de mediu ale shell

Este posibil ca unele variabile de mediu uzuale pentru linia de comandă a shell-ului standard să nu fie disponibile în mediul de execuție al scriptului dumneavoastră.

- Pentru „$USER”, folosiți „`$(id -un)`”
- Pentru „$UID”, folosiți „`$(id -u)`”
- Pentru „$HOME”, folosiți „`$(getent passwd „$(id -u)”|cut -d „:” -f 6)`” (această comandă funcționează, de asemenea, pe Secțiune 4.5.2)

12.1.6 Secvența de procesare a liniei de comandă shell

Shell-ul procesează un script aproximativ după următoarea secvență.

- Shell-ul citește o linie.
- Shell-ul grupează o parte a liniei ca **un singur element (token)** dacă aceasta se află între `"…"` sau `'…'`.

- Shell-ul împarte restul unei linii în **elemente (token-uri)** după cum urmează.
  - Spații-albe: *spațiu tabulator linie-nouă*
  - Metacaractere::`< > | ; & ( )`
- Shell-ul verifică **cuvântul rezervat** pentru fiecare token, pentru a-și adapta comportamentul în cazul în care acesta nu se află între "…" sau '…'.
  - **cuvânt rezervat**: `if then elif else fi for in while unless do done case esac`
- Shell-ul extinde **alias** dacă acesta nu se află între "…" sau '…'.
- Shell-ul expandează **tilde** dacă nu se află între "…" sau '…'.
  - „~” → directorul personal al utilizatorului curent
  - „~*utilizator*” → directorul personal al *utilizatorului*
- Shell-ul înlocuiește **parametrul** cu valoarea acestuia, dacă nu se află între '…'.
  - **parametru**: „$PARAMETER” sau „${PARAMETER}”
- Shell-ul extinde **substituirea comenzii** dacă aceasta nu se află între '…'.
  - „`$( comandă )`” → ieșirea comenzii „`comandă`”
  - „`` ` comandă ` ``” → rezultatul comenzii „`comandă`”
- Shell-ul extinde **expresia de tip glob (cu caractere joker) pentru ruta** la numele de fișiere corespunzătoare, dacă acestea nu se află între "…" sau '…'.
  - `*` → orice caractere
  - `?` → un caracter
  - `[…]` → oricare dintre caracterele din ”…”
- Shell-ul caută **comanda** din următoarele și o execută.
  - definiția **funcției**
  - comanda **internă**
  - **fișierul executabil** din „$PATH”
- Shell-ul trece la linia următoare și repetă acest proces de la începutul secvenței.

Ghilimelele simple între ghilimele duble nu au niciun efect.

Executarea comenzii „`set -x`” în shell sau lansarea shell-ului cu opțiunea „`-x`” determină shell-ul să afișeze toate comenzile executate. Acest lucru este foarte util pentru depanare.

12.1.7 Programe utile pentru scripturile shell

Pentru a face programul dvs. de tip shell cât mai portabil posibil pe sistemele Debian, este recomandat să limitați utilizarea programelor utilitare la cele furnizate de pachetele **esențiale**.

- «`aptitude search ~E`» listează pachetele **esențiale**.
- «`dpkg -L` *`nume_pachet`* `| grep '/man/man.*/'`» afișează paginile de manual pentru comenzile oferite de pachetul *`nume_pachet`*.

Indicație
Deși `moreutils` s-ar putea să nu existe în afara distribuției Debian, acesta oferă câteva programe mici și interesante. Cel mai notabil dintre acestea este sponge(8), care se dovedește destul de util atunci când doriți să suprascrieți un fișier original.

Consultați Secțiune 1.6 pentru exemple.

pachet	popcon(popularitate) și dimensiune	dimensiune	descriere
dash	V:929, I:999	207	un shell mic și rapid, compatibil cu POSIX, pentru sh
coreutils	V:912, I:1000	17994	instrumente de bază GNU
grep	V:782, I:1000	1297	GNU grep, egrep și fgrep
sed	V:769, I:1000	987	GNU sed
mawk	V:495, I:998	299	mic și rapid awk
debianutils	V:935, I:996	228	diverse instrumente specifice Debian
bsdutils	V:421, I:999	338	instrumente de bază din 4.4BSD-Lite
bsdextrautils	V:761, I:870	382	instrumente extra din 4.4BSD-Lite
moreutils	V:18, I:39	232	instrumente Unix suplimentare

Tabela 12.7: Lista pachetelor care conțin mici programe utilitare pentru scripturi shell

12.2 Programarea în limbaje interpretate

pachet	popcon(popularitate) și dimensiune	dimensiune	documentație
dash	V:929, I:999	207	sh: un shell mic și rapid, compatibil cu POSIX, pentru sh
bash	V:886, I:999	7308	sh: „info bash” furnizat de bash-doc
mawk	V:495, I:998	299	AWK: o versiune compactă și rapidă a awk
gawk	V:247, I:294	3289	AWK: „info gawk” furnizat de gawk-doc
perl	V:663, I:990	840	Perl: perl(1) și paginile HTML furnizate de perl-doc și perl-doc-html
libterm-readline-gnu-perl	V:2, I:27	439	Extensie Perl pentru biblioteca GNU ReadLine/History: perlsh(1)
libreply-perl	V:0.01, I:0.07	171	REPL pentru Perl: reply(1)
libdevel-repl-perl	V:0.02, I:0.50	237	REPL pentru Perl: re.pl(1)
python3	V:720, I:975	81	Python: python3(1) și paginile HTML furnizate de python3-doc
tcl	V:24, I:173	20	Tcl: tcl(3) și paginile de manual detaliate furnizate de tcl-doc
tk	V:18, I:167	20	Tk: tk(3) și paginile de manual detaliate furnizate de tk-doc
ruby	V:72, I:154	32	Ruby: ruby(1), erb(1), irb(1), rdoc(1), ri(1)

Tabela 12.8: Lista pachetelor legate de interpret

Când doriți să automatizați o sarcină în Debian, ar trebui să o programați mai întâi într-un limbaj interpretat. Ghidul pentru alegerea limbajului interpretat este:

- Folosiți dash dacă sarcina este una simplă, care combină programe CLI cu un program shell.
- Folosiți python3 dacă sarcina nu este una simplă și o scrieți de la zero.
- Folosiți perl, tcl, ruby, ... dacă există un cod existent în Debian care utilizează unul dintre aceste limbaje și care trebuie modificat pentru a îndeplini sarcina.

Dacă codul rezultat este prea lent, puteți rescrie doar porțiunea esențială pentru viteza de execuție într-un limbaj compilat și o puteți apela din limbajul interpretat.

12.2.1 Depanarea codurilor în limbaje interpretate

Majoritatea interpreților oferă funcționalități de bază pentru verificarea sintaxei și urmărirea codului.

- «**dash -n** *script.sh*» - Verificarea sintaxei unui script Shell
- «**dash -x** *script.sh*» - Trasarea execuției unui script Shell
- «**python -m py_compile** *script.py*» - Verificarea sintaxei unui script Python
- «**python -mtrace --trace** *script.py*» - Trasarea execuției unui script Python
- “**perl -I ../libpath -c** *script.pl*” - Verificarea sintaxei unui script Perl
- “**perl -d:Trace** *script.pl*” - Trasarea execuției unui script Perl

Pentru a testa codul pentru `dash`, încercați Secțiune 9.1.4, care oferă un mediu interactiv similar cu `bash`.

Pentru a testa codul în `Perl`, încercați mediul REPL pentru Perl, care oferă un mediu de tip Python – adică un mediu REPL (=READ + EVAL + PRINT + LOOP) pentru Perl.

12.2.2 Program cu interfață grafică (GUI) și script de shell

Scriptul Shell poate fi îmbunătățit pentru a crea un program cu interfață grafică atractivă. Secretul constă în utilizarea unuia dintre așa-numitele programe de tip dialog, în locul interacțiunii monotone bazate pe comenzile `echo` și `read`.

pachet	popcon(popularitate) și dimensiune		descriere
`x11-utils`	V:228, I:555	651	xmessage(1): afișează un mesaj sau o solicitare într-o fereastră (X)
`whiptail`	V:307, I:997	61	afișează casete de dialog ușor de utilizat din scripturi shell (newt)
`dialog`	V:8, I:77	520	afișează casete de dialog ușor de utilizat din scripturi shell (ncurses)
`zenity`	V:68, I:341	193	afișează casete de dialog grafice din scripturi shell (GTK)
`ssft`	V:0.01, I:0.16	80	Instrument de interfață pentru scripturi Shell (înveliș pentru zenity, kdialog și dialog cu gettext)
`gettext`	V:49, I:210	20549	”`/usr/bin/gettext.sh`”: traduce mesajul

Tabela 12.9: Lista programelor de dialog

Iată un exemplu de program cu interfață grafică care demonstrează cât de ușor este să realizezi acest lucru folosind doar un script de shell.

Acest script utilizează `zenity` pentru a selecta un fișier (implicit `/etc/motd`) și a-l afișa.

Lansatorul GUI pentru acest script poate fi creat urmând instrucțiunile de la Secțiune 9.4.10.

```
#!/bin/sh -e
# Copyright (C) 2021 Osamu Aoki <osamu@debian.org>, Public Domain
# vim:set sw=2 sts=2 et:
DATA_FILE=$(zenity --file-selection --filename="/etc/motd" --title="Select a file to check ↩
    ") || \
  ( echo "E: File selection error" >&2 ; exit 1 )
# Check size of archive
if ( file -ib "$DATA_FILE" | grep -qe '^text/' ) ; then
  zenity --info --title="Check file: $DATA_FILE" --width 640  --height 400 \
    --text="$(head -n 20 "$DATA_FILE")"
else
  zenity --info --title="Check file: $DATA_FILE" --width 640  --height 400 \
    --text="The data is MIME=$(file -ib "$DATA_FILE")"
fi
```

Acest tip de abordare a programelor cu interfață grafică (GUI) folosind scripturi shell este utilă doar în cazurile simple. Dacă intenționați să scrieți un program mai complex, vă recomandăm să îl scrieți pe o platformă mai performantă.

12.2.3 Acțiuni personalizate pentru gestionarul de fișiere cu interfață grafică

Programele de gestionare a fișierelor cu interfață grafică pot fi extinse pentru a efectua anumite operații uzuale asupra fișierelor selectate, folosind pachete de extensii suplimentare. De asemenea, pot fi configurate să execute operații personalizate foarte specifice prin adăugarea propriilor scripturi.

- Pentru GNOME, consultați NautilusScriptsHowto.
- Pentru KDE, consultați Creating Dolphin Service Menus (Crearea meniurilor de servicii Dolphin).
- Pentru Xfce, consultați Thunar - Custom Actions (Thunar - Acțiuni personalizate) și https://help.ubuntu.com/community/-ThunarCustomActions.
- Pentru LXDE, consultați Custom Actions (Acțiuni personalizate).

12.2.4 Nebunia scripturilor scurte în Perl

Pentru a procesa datele, `sh` trebuie să lanseze subprocese care execută `cut`, `grep`, `sed` etc., iar acest lucru este lent. Pe de altă parte, `perl` dispune de capacități interne de procesare a datelor și este rapid. De aceea, multe scripturi de întreținere a sistemului pe Debian folosesc `perl`.

Să analizăm următorul fragment de script AWK de o singură linie și echivalentele sale în Perl.

```
awk '($2=="1957") { print $3 }' |
```

Aceasta este echivalentă cu oricare dintre următoarele linii.

```
perl -ne '@f=split; if ($f[1] eq "1957") { print "$f[2]\n"}' |
```

```
perl -ne 'if ((@f=split)[1] eq "1957") { print "$f[2]\n"}' |
```

```
perl -ne '@f=split; print $f[2] if ( $f[1]==1957 )' |
```

```
perl -lane 'print $F[2] if $F[1] eq "1957"' |
```

```
perl -lane 'print$F[2]if$F[1]eq+1957' |
```

Ultima este o ghicitoare. Aceasta a folosit următoarele caracteristici ale limbajului Perl.

- Spațiul în alb este opțional.
- Există o conversie automată de la număr la șir de caractere.
- Trucuri pentru executarea Perl prin opțiuni de linie de comandă: perlrun(1)
- Variabile speciale Perl: perlvar(1)

Această flexibilitate este punctul forte al limbajului Perl. În același timp, aceasta ne permite să creăm coduri criptice și încâlcite. Așadar, fiți atenți.

12.3 Programarea în limbaje compilate

Aici, Secțiune 12.3.3 și Secțiune 12.3.4 sunt incluse pentru a arăta cum se pot scrie programe de tip compilator în limbajul C, prin compilarea unei descrieri de nivel superior în limbajul C.

pachet	popcon(popularitate)	dimensiune	descriere
gcc	V:156, I:561	36	compilatorul GNU C
libc6-dev	V:258, I:580	12863	biblioteca GNU C: biblioteci de dezvoltare și fișiere de antet
g++	V:61, I:529	13	compilatorul GNU C++
libstdc++-16-dev	V:3, I:19	24398	biblioteca standard GNU C++ v3 (fișiere de dezvoltare)
cpp	V:337, I:720	18	preprocesorul GNU C
gettext	V:49, I:210	20549	instrumente de internaționalizare GNU
glade	V:0.2, I:2.6	1613	constructor de interfețe grafice de utilizator GTK
valac	V:0.2, I:3.0	533	limbaj similar cu C# pentru sistemul GObject
flex	V:6, I:67	1247	generator de analizare lexicală rapidă compatibil cu LEX-
bison	V:7, I:71	3122	generator de analizare compatibil cu YACC
susv2	I:0.03	16	preia „Specificațiile UNIX unice v2”
susv3	I:0.05	16	preia „Specificațiile UNIX unice v3”
susv4	I:0.02	16	preia „Specificațiile UNIX unice v4”
golang	I:21	11	compilator pentru limbajul de programare Go
rustc	V:5, I:20	13195	limbajul de programare de sisteme Rust
gfortran	V:4, I:49	15	compilatorul GNU Fortran 95
fpc	I:2.2	101	Free Pascal

Tabela 12.10: Lista pachetelor legate de compilator

12.3.1 C

Puteți configura mediul necesar pentru a compila programe scrise în limbajul de programare C urmând pașii de mai jos.

```
# apt-get install glibc-doc manpages-dev libc6-dev gcc build-essential
```

Pachetul `libc6-dev`, adică biblioteca GNU C, oferă biblioteca standard C, care este o colecție de fișiere antet și rutine de bibliotecă utilizate de limbajul de programare C.

A se vedea următoarele referințe pentru limbajul C.

- „`info libc`” (referința funcțiilor bibliotecii C)
- gcc(1) și «`info gcc`»
- each_C_library_function_name(3)
- Kernighan & Ritchie, „The C Programming Language” - Limbajul de programare C, ediția a II-a (Prentice Hall)

12.3.2 Program simplu în limbajul C (gcc)

Un exemplu simplu, „`example.c`”, poate fi compilat împreună cu biblioteca „`libm`” într-un fișier executabil „`run_example`” după cum urmează.

```
$ cat > example.c << EOF
#include <stdio.h>
#include <math.h>
#include <string.h>

int main(int argc, char **argv, char **envp){
        double x;
        char y[11];
        x=sqrt(argc+7.5);
        strncpy(y, argv[0], 10); /* prevent buffer overflow */
```

```
        y[10] = '\0'; /* fill to make sure string ends with '\0' */
        printf("%5i, %5.3f, %10s, %10s\n", argc, x, y, argv[1]);
        return 0;
}
EOF
$ gcc -Wall -g -o run_example example.c -lm
$ ./run_example
        1, 2.915, ./run_exam,     (null)
$ ./run_example 1234567890qwerty
        2, 3.082, ./run_exam, 1234567890qwerty
```

Aici, „`-lm`” este necesar pentru a lega biblioteca „`/usr/lib/libm.so`” din pachetul `libc6` pentru sqrt(3). Biblioteca propriu-zisă se află în „`/lib/`” cu numele de fișier „`libm.so.6`”, care este o legătură simbolică către „`libm-2.7.so`”.

Uită-te la ultimul parametru din textul afișat. Are mai mult de 10 caractere, deși s-au specificat „`%10s`”.

Utilizarea funcțiilor de operare cu memoria prin intermediul indicatoare fără verificări de limite, precum sprintf(3) și strcpy(3), este învechită pentru a preveni exploatările de tip „buffer overflow” (debordare a memoriei tampon) care profită de efectele de depășire menționate mai sus. În schimb, utilizați snprintf(3) și strncpy(3).

12.3.3 Flex — o versiune îmbunătățită a Lex

Flex este un generator de analizare lexicală rapidă, compatibil cu Lex.

Tutorialul pentru flex(1) poate fi găsit în «`info flex`».

Multe exemple simple pot fi găsite în „`/usr/share/doc/flex/examples/`”. 1

12.3.4 Bison — o versiune îmbunătățită a Yacc

În Debian, există mai multe pachete care oferă un generator de analizare LR cu citire anticipată compatibil cu Yacc sau un generator de analizare LALR.

pachet	popcon(popularitate) și dimensiune		descriere
`bison`	V:7, I:71	3122	generator de analizare LALR GNU
`byacc`	V:0.1, I:2.9	263	generator de analizare LALR Berkeley
`btyacc`	V:0.01, I:0.04	247	generator de analizare cu capacitate de urmărire bazat pe `byacc`

Tabela 12.11: Lista generatoarelor de analizare LALR compatibile cu Yacc

Tutorialul pentru bison(1) poate fi găsit în «`info bison`».

Trebuie să furnizați propriile funcții „`main()`” și „`yyerror()`”. Funcția „`main()`” apelează funcția „`yyparse()`”, care la rândul ei apelează funcția „`yylex()`”, create de obicei cu ajutorul Flex.

Iată un exemplu de creare a unui program simplu de calculator pentru terminal.

Să creăm `example.y`:

```
/* calculator source for bison */
%{
#include <stdio.h>
extern int yylex(void);
extern int yyerror(char *);
%}
```

1Este posibil să fie necesare câteva ajustări pentru ca acestea să funcționeze în sistemul actual.

```
/* declare tokens */
%token NUMBER
%token OP_ADD OP_SUB OP_MUL OP_RGT OP_LFT OP_EQU

%%
calc:
 | calc exp OP_EQU    { printf("Y: RESULT = %d\n", $2); }
 ;

exp: factor
 | exp OP_ADD factor  { $$ = $1 + $3; }
 | exp OP_SUB factor  { $$ = $1 - $3; }
 ;

factor: term
 | factor OP_MUL term { $$ = $1 * $3; }
 ;

term: NUMBER
 | OP_LFT exp OP_RGT  { $$ = $2; }
  ;
%%

int main(int argc, char **argv)
{
  yyparse();
}

int yyerror(char *s)
{
  fprintf(stderr, "error: '%s'\n", s);
}
```

Să creăm `example.l`:

```
/* calculator source for flex */
%{
#include "example.tab.h"
%}

%%
[0-9]+ { printf("L: NUMBER = %s\n", yytext); yylval = atoi(yytext); return NUMBER; }
"+"    { printf("L: OP_ADD\n"); return OP_ADD; }
"-"    { printf("L: OP_SUB\n"); return OP_SUB; }
"*"    { printf("L: OP_MUL\n"); return OP_MUL; }
"("    { printf("L: OP_LFT\n"); return OP_LFT; }
")"    { printf("L: OP_RGT\n"); return OP_RGT; }
"="    { printf("L: OP_EQU\n"); return OP_EQU; }
"exit" { printf("L: exit\n");   return YYEOF; } /* YYEOF = 0 */
.      { /* ignore all other */ }
%%
```

Apoi, pentru a încerca acest lucru, executați următoarea comandă din linia de comandă:

```
$ bison -d example.y
$ flex example.l
$ gcc -lfl example.tab.c lex.yy.c -o example
$ ./example
1 + 2 * ( 3 + 1 ) =
L: NUMBER = 1
```

```
L: OP_ADD
L: NUMBER = 2
L: OP_MUL
L: OP_LFT
L: NUMBER = 3
L: OP_ADD
L: NUMBER = 1
L: OP_RGT
L: OP_EQU
Y: RESULT = 9

exit
L: exit
```

12.4 Instrumente de analiză statică a codului

Instrumentele de tip Lint pot facilita efectuarea automată a analizei statice a codului.

Instrumentele de tip Indent pot facilita revizuirea manuală a codului prin reformatarea consecventă a codurilor sursă.

Instrumentele precum Ctags pot ajuta la revizuirea manuală a codului prin generarea unui fișier index (sau de etichete) cu numele identificate în codurile sursă.

Indicație
Configurarea editorului preferat (`emacs` sau `vim`) pentru a utiliza module de motoare de verificare sintactică (lint) asincrone vă ajută la scrierea codului. Aceste module devin din ce în ce mai performante, profitând de Language Server Protocol. Având în vedere că evoluează rapid, utilizarea codului lor din sursa originală, în locul pachetului Debian, poate fi o opțiune bună.

12.5 Depanare

Depanarea este o parte importantă a activităților de programare. Dacă știi să depanezi programele, devii un utilizator Debian competent, capabil să întocmească rapoarte de erori relevante.

12.5.1 Executarea de bază a gdb

Principalul depanator din Debian este gdb(1), care vă permite să examinați un program în timpul execuției.

Să instalăm `gdb` și programele asociate urmând pașii de mai jos.

```
# apt-get install gdb gdb-doc build-essential devscripts
```

Un bun tutorial despre `gdb` este disponibil:

- «`info gdb`»
- „Depanarea cu GDB” în `/usr/share/doc/gdb-doc/html/gdb/index.html`
- “tutorial în Internet”

Iată un exemplu simplu de utilizare a gdb(1) pe un „`program`” compilat cu opțiunea „`-g`” pentru a genera informații de depanare.

pachet	popcon(popularitate) și	dimensiune	descriere
vim-ale	I:0.74	2833	motor lint asincron pentru Vim 8 și NeoVim
vim-syntastic	I:1.9	1379	trucuri pentru verificarea sintaxei în Vim
elpa-flycheck	V:0.1, I:1.6	1130	verificare sintactică din mers pentru Emacs
elpa-relint	V:0.00, I:0.04	150	instrument de detectare a erorilor în expresiile regulate Emacs Lisp
cppcheck-gui	V:0.11, I:0.97	7973	instrument pentru analiza statică a codului C/C++ (interfață grafică)
shellcheck	V:4, I:17	22860	instrument de analizare sintatică a scripturilor shell
pyflakes3	V:2, I:14	20	verificator pasiv al programelor Python 3
pylint	V:4, I:19	2104	verificator static al codului Python
perl	V:663, I:990	840	interpret cu verificator intern de cod static: `B::Lint`(3perl)
rubocop	V:0.1, I:1.0	4502	analizator de cod static Ruby
clang-tidy	V:2, I:11	20	instrument de analizare sintactică a codului C++ bazat pe clang
splint	V:0.05, I:0.79	2325	instrument pentru verificarea statică a programelor C în vederea depistării erorilor
flawfinder	V:0.05, I:0.43	187	instrument pentru examinarea codului sursă C/C++ și identificarea vulnerabilităților de securitate
black	V:3, I:15	5868	instrument de formatare a codului Python fără compromisuri
perltidy	V:0.4, I:2.9	3086	instrument de indentare și reformatare a scripturilor Perl
indent	V:0.3, I:4.7	438	program de formatare a codului sursă în limbajul C
astyle	V:0.2, I:2.5	770	instrument de indentare a codului sursă pentru limbajele C, C++, Objective-C, C# și Java
bcpp	V:0.02, I:0.22	114	instrument de formatare elegantă pentru C(++)
xmlindent	V:0.07, I:0.73	52	instrument de reformatare a fluxului XML
global	V:0.2, I:1.5	1923	instrumente de căutare și navigare în codul sursă
exuberant-ctags	V:1, I:12	341	creează indexuri ale fișierelor de etichete pentru definițiile codului sursă
universal-ctags	V:1, I:12	4238	creează indexuri ale fișierelor de etichete pentru definițiile codului sursă

Tabela 12.12: Lista instrumentelor pentru analiza statică a codului

pachet	popcon(popularitate) și	dimensiune	documentație
gdb	V:92, I:162	13472	«`info gdb`» furnizat de `gdb-doc`
ddd	V:0.3, I:4.9	5094	«`info ddd`» furnizat de `ddd-doc`

Tabela 12.13: Lista pachetelor de depanare

```
$ gdb program
(gdb) b 1               # set break point at line 1
(gdb) run args          # run program with args
(gdb) next              # next line
...
(gdb) step              # step forward
...
(gdb) p parm            # print parm
...
(gdb) p parm=12         # set value to 12
...
(gdb) quit
```

Indicație
Multe comenzi gdb(1) pot fi abreviate. Extinderea cu tasta «Tab» funcționează la fel ca în shell.

12.5.2 Depanarea pachetului Debian

Deoarece toate fișierele binare instalate trebuie să fie curățate de simboluri în sistemul Debian în mod implicit, majoritatea simbolurilor de depanare sunt eliminate din pachetul standard. Pentru a depana pachetele Debian cu gdb(1), trebuie instalate pachetele `*-dbgsym` (de exemplu, `coreutils-dbgsym` în cazul `coreutils`). Pachetele sursă generează automat pachete `*-dbgsym` împreună cu pachetele binare normale, iar aceste pachete de depanare sunt plasate separat în arhiva debian-debug. Vă rugăm să consultați articolele de pe Debian Wiki pentru mai multe informații.

Dacă un pachet care urmează să fie depanat nu include pachetul `*-dbgsym`, trebuie să îl instalați după ce îl recompilați, urmând pașii de mai jos.

```
$ mkdir /path/new ; cd /path/new
$ sudo apt-get update
$ sudo apt-get dist-upgrade
$ sudo apt-get install fakeroot devscripts build-essential
$ apt-get source package_name
$ cd package_name*
$ sudo apt-get build-dep ./
```

Remediați erorile, dacă este necesar.

Actualizați versiunea pachetului la una care nu intră în conflict cu versiunile oficiale Debian, de exemplu una la care se adaugă „`+debug1`” atunci când recompilați o versiune existentă a pachetului, sau una la care se adaugă „`~pre1`” atunci când compilați o versiune a pachetului care nu a fost încă lansată, urmând pașii de mai jos.

```
$ dch -i
```

Compilați și instalați pachetele cu simboluri de depanare urmând pașii de mai jos.

```
$ export DEB_BUILD_OPTIONS="nostrip noopt"
$ debuild
$ cd ..
$ sudo debi package_name*.changes
```

Trebuie să verificați scripturile de compilare ale pachetului și să vă asigurați că utilizați „`CFLAGS=-g -Wall`” pentru compilarea fișierelor binare.

12.5.3 Obținerea urmăririi traseului execuției

Când programul se „prăbușește” (crash), este recomandat să trimiteți un raport de eroare care să conțină informațiile traseului de execuție copiate și lipite.

Urmărirea traseului de execuție poate fi obținută prin gdb(1) folosind una dintre următoarele metode:

- Abordarea „Crash-in-GDB”:
  - Rulați programul din GDB.
  - Blocați programul (faceți-l să se prăbușească).
  - Tastați „`bt`” în promptul GDB.
- Abordarea „crash-first”:
  - Actualizați fișierul „**/etc/security/limits.conf**” pentru a include următoarele:

```
* soft core unlimited
```

  - Tastați «`ulimit -c unlimited`» în promptul shell-ului.
  - Rulați programul din acest prompt shell.
  - Provocați prăbușirea programului pentru a genera un fișier core dump.
  - Încărcați fișierul core dump în GDB folosind comanda «`gdb gdb ./program_binary core`».
  - Tastați „`bt`” în promptul GDB.

În cazul unei bucle infinite sau al blocării tastaturii, puteți forța închiderea programului apăsând `Ctrl-\` sau `Ctrl-C` sau executând comanda „`kill -ABRT` *`PID`*”. (Consultați Secțiune 9.4.12)

Indicație
Adesea, se întâmplă să vezi o urmă de execuție în care una sau mai multe dintre primele linii sunt de tipul „`malloc()`” sau „`g_malloc()`”. Când se întâmplă acest lucru, este foarte probabil ca traseul de execuție să nu fie foarte util. Cea mai simplă modalitate de a găsi informații utile este să configurați variabila de mediu „`$MALLOC_CHECK_`” la valoarea 2 (malloc(3)). Puteți face acest lucru în timp ce rulați `gdb`, urmând pașii de mai jos.

```
$ MALLOC_CHECK_=2 gdb hello
```

12.5.4 Comenzi avansate gdb

comandă	descrierea obiectivelor comenzii
`(gdb) thread apply all bt`	obține o urmărire a traseului de execuție pentru toate firele de execuție ale unui program cu mai multe fire de execuție
`(gdb) bt full`	obține parametrii aflați în stiva de apeluri de funcții
`(gdb) thread apply all bt full`	obține o urmărire a traseului de execuției și parametrii ca urmare a combinării opțiunilor anterioare
`(gdb) thread apply all bt full 10`	obțineți o urmărire a traseului de execuție și parametrii pentru primele 10 apeluri, pentru a elimina rezultatele irelevante
`(gdb) set logging on`	înregistrează jurnalul ieșirii `gdb` într-un fișier (implicit este „`gdb.txt`”)

Tabela 12.14: Lista comenzilor avansate gdb

12.5.5 Verificați dependențele de biblioteci

Folosiți ldd(1) pentru a afla de ce biblioteci depinde un program, urmând pașii de mai jos.

```
$ ldd /usr/bin/ls
        librt.so.1 => /lib/librt.so.1 (0x4001e000)
        libc.so.6 => /lib/libc.so.6 (0x40030000)
        libpthread.so.0 => /lib/libpthread.so.0 (0x40153000)
        /lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

Pentru ca ls(1) să funcționeze într-un mediu „chroot”, bibliotecile menționate mai sus trebuie să fie disponibile în mediul „chroot”.

Consultați Secțiune 9.4.6.

12.5.6 Instrumente de urmărire dinamică a apelurilor

În Debian sunt disponibile mai multe instrumente de urmărire dinamică a apelurilor. Consultați Secțiune 9.4.

12.5.7 Depanarea erorilor X

Dacă un program GNOME `preview1` a primit o eroare X, ar trebui să apară următorul mesaj.

```
The program 'preview1' received an X Window System error.
```

În acest caz, puteți încerca să rulați programul cu opțiunea „`--sync`” și să opriți execuția la funcția „`gdk_x_error`” pentru a obține o urmărire a traseului execuției.

12.5.8 Instrumente de detectare a „scurgerilor” de memorie

În Debian sunt disponibile mai multe instrumente de detectare a „scurgerilor” de memorie.

pachet	popcon(popularitate) și descriere		
`libc6-dev`	V:258, I:580	12863	mtrace(1): funcționalitate de depanare a malloc în glibc
`valgrind`	V:4, I:32	89318	instrument de depanare și optimizare a memoriei
`electric-fence`	V:0.2, I:2.2	69	instrument de depanare malloc(3)
`libdmalloc5`	V:0.01, I:0.57	373	bibliotecă de depanare a alocării memoriei
`duma`	V:0.00, I:0.05	297	bibliotecă pentru detectarea depășirilor și a subutilizărilor de memorie tampon în programele C și C++
`leaktracer`	V:0.01, I:0.27	56	instrument de depistare a scurgerilor de memorie pentru programe C++

Tabela 12.15: Lista instrumentelor de detectare a „scurgerilor” de memorie

12.5.9 Dezasamblarea unui fișier binar

Puteți dezasambla codul binar cu objdump(1) urmând pașii de mai jos.

```
$  objdump -m i386 -b binary -D /usr/lib/grub/x86_64-pc/stage1
```

Notă
gdb(1) poate fi utilizat pentru a dezasambla codul în mod interactiv.

12.6 Instrumentele de construcție

pachet	popcon(popularitate) și dimensiune		documentație
make	V:145, I:562	1762	«info make» furnizat de make-doc
autoconf	V:27, I:193	2244	«info autoconf» furnizat de autoconf-doc
automake	V:26, I:192	1958	«info automake» furnizat de automake1.10-doc
libtool	V:23, I:176	1276	«info libtool» furnizat de libtool-doc
cmake	V:22, I:123	59735	cmake(1) sistem de compilare multiplatformă, cu codul-sursă deschis
ninja-build	V:8, I:54	456	ninja(1) un sistem de compilare compact, care se apropie cel mai mult de spiritul Make
meson	V:6, I:29	4402	meson(1) sistem de compilare de înaltă productivitate bazat pe ninja
xutils-dev	V:0.6, I:6.7	1495	imake(1), xmkmf(1), etc.

Tabela 12.16: Lista pachetelor de instrumente de construire

12.6.1 Make

Make este un instrument destinat gestionării grupurilor de programe. La executarea comenzii make(1), make citește fișierul de reguli, „Makefile”, și actualizează o țintă dacă aceasta depinde de fișiere prealabile care au fost modificate de la ultima modificare a țintei respective sau dacă ținta nu există. Executarea acestor actualizări poate avea loc simultan.

Sintaxa fișierului de reguli este următoarea.

```
target: [ prerequisites ... ]
 [TAB]  command1
 [TAB]  -command2 # ignore errors
 [TAB]  @command3 # suppress echoing
```

Aici „[TAB]” reprezintă un cod TAB (codul de tabulare). Fiecare linie este interpretată de shell după ce se efectuează substituirea variabilelor. Folosiți „\” la sfârșitul unei linii pentru a continua scriptul. Folosiți „$$” pentru a introduce „$” ca valori de mediu pentru un script de shell.

Regulile implicite pentru țintă și condițiile prealabile pot fi formulate, de exemplu, după cum urmează.

```
%.o: %.c header.h
```

În acest caz, ținta conține caracterul „%” (exact unul). „%” poate corespunde oricărui subșir non-vid din numele de fișiere țintă efective. De asemenea, condițiile preliminare utilizează „%” pentru a ilustra modul în care numele lor se raportează la numele țintei efective.

variabila automată	valoare
$@	ținta
$<	prima cerință prealabilă
$?	toate cerințele prealabile mai recente
$^	toate cerințele prealabile
$*	„%” corespunde cu rădăcina din modelul țintă

Tabela 12.17: Lista variabilelor automate «make»

Rulați «make -p -f/dev/null» pentru a vedea regulile interne automate.

expansiunea variabilei	descriere
`foo1 := bar`	expansiune unică
`foo2 = bar`	expansiune recursivă
`foo3 += bar`	adăugare

Tabela 12.18: Lista expansiunilor variabilelor «make»

12.6.2 Autotools

Autotools este o suită de instrumente de programare concepută pentru a facilita portabilitatea pachetelor de cod sursă pe numeroase sisteme de tip Unix.

- Autoconf este un instrument care generează un script shell „`configure`” din „`configure.ac`”.
  - «`configure`» este utilizat ulterior pentru a genera fișierul «`Makefile`» pornind de la șablonul „`Makefile.in`”.
- Automake este un instrument care generează fișierul „`Makefile.in`” din fișierul „`Makefile.am`”.
- Libtool este un script shell care rezolvă problema portabilității software-ului la compilarea bibliotecilor partajate din codul sursă.

12.6.2.1 Compilarea și instalarea unui program



Avertisment
Nu suprascrieți fișierele de sistem cu programele compilate atunci când le instalați.

Debian nu modifică fișierele din „`/usr/local/`” sau „`/opt`”. Așadar, dacă compilați un program din sursă, instalați-l în „`/usr/local/`”, pentru a nu interfera cu Debian.

```
$ cd src
$ ./configure --prefix=/usr/local
$ make # this compiles program
$ sudo make install # this installs the files in the system
```

12.6.2.2 Dezinstalarea unui program

Dacă dispuneți de sursa originală și dacă aceasta utilizează autoconf(1)/automake(1) și dacă vă amintiți cum ați configurat-o, executați următoarele comenzi pentru a dezinstala programul.

```
$ ./configure all-of-the-options-you-gave-it
$ sudo make uninstall
```

Alternativ, dacă sunteți absolut sigur că procesul de instalare plasează fișierele numai în directorul „`/usr/local/`” și că acolo nu se află nimic important, puteți șterge tot conținutul acestuia urmând pașii de mai jos.

```
# find /usr/local -type f -print0 | xargs -0 rm -f
```

Dacă nu știți exact unde sunt instalate fișierele, vă recomandăm să folosiți checkinstall(8) din pachetul `checkinstall`, care asigură o dezinstalare completă. Acum acceptă crearea unui pachet Debian cu opțiunea „`-D`”.

12.6.3 Meson

Sistemul de construire a software-ului a evoluat:

- Autotools, care a precedat Make, a reprezentat standardul de facto pentru infrastructura de compilare portabilă încă din anii 1990. Acesta este extrem de lent.
- CMake, lansat inițial în 2000, a îmbunătățit semnificativ viteza, dar a fost construit inițial pe baza instrumentului Make, care este lent prin natura sa. În prezent, Ninja poate fi folosit ca motor al acestuia.
- Ninja, lansat inițial în 2012, are rolul de a înlocui Make pentru a îmbunătăți și mai mult viteza de compilare și este conceput astfel încât fișierele sale de intrare să fie generate de un sistem de compilare de nivel superior.
- Meson, lansat inițial în 2013, este un sistem de compilare de nivel superior, popular și rapid, care utilizează Ninja ca motor.

Consultați documentele disponibile la „Sistemul de compilare Meson” și „Sistemul de compilare Ninja”.

12.7 Web

Paginile web dinamice și interactive simple pot fi create după cum urmează.

- Întrebările sunt prezentate utilizatorului navigatorului folosind formulare HTML.
- Completarea câmpurilor formularului și apăsarea butonului de trimitere trimite unul dintre următoarele șiruri URL cu parametri codificați din navigator către serverul web.
  - ”`https://www.foo.dom/cgi-bin/program.pl?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3`”
  - ”`https://www.foo.dom/cgi-bin/program.py?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3`”
  - ”`https://www.foo.dom/program.php?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3`”
- „%nn” din adresa URL este înlocuit cu un caracter cu valoarea hexazecimală `nn`.
- Variabila de mediu este configurată astfel: ”`QUERY_STRING="VAR1=VAL1 VAR2=VAL2 VAR3=VAL3"`”.
- Programul CGI (oricare dintre programele de tip „`program.*`”) de pe serverul web se execută cu variabila de mediu „`$QUERY_STRING`”.
- `stdout` ieșirea standard a programului CGI este transmisă către navigatorul web și este afișată sub forma unei pagini web dinamice interactive.

Din motive de securitate, este recomandat să nu creați manual noi metode de analizare a parametrilor CGI. Există module consacrate pentru acest scop în Perl și Python. PHP include deja aceste funcționalități. Când este necesară stocarea datelor pe partea clientului, se utilizează cookie-urile HTTP. Când este necesară procesarea datelor pe partea clientului, se utilizează frecvent Javascript.

Pentru mai multe informații, consultați Common Gateway Interface, The Apache Software Foundation și JavaScript.

Căutarea expresiei „CGI tutorial” în Google, introducând adresa URL codificată https://www.google.com/search?hl=en&ie=UTF-8&q=CGI+tutorial direct în bara de adrese a browserului, este o modalitate bună de a vedea scriptul CGI în acțiune pe serverul Google.

12.8 Conversia codului sursă

Există programe pentru conversia codurilor sursă.

pachet	popcon(popularitate)	dimensiune	cuvânt-cheie	descriere
perl	V:663, I:990	840	AWK → PERL	convertește codurile sursă din AWK în PERL: a2p(1)
f2c	V:0.1, I:1.9	443	FORTRAN → C	convertește codurile sursă din FORTRAN 77 în C/C++: f2c(1)
intel2gas	V:0.00, I:0.16	178	NASM → GAS	convertor de la NASM (format Intel) la GNU Assembler (GAS)

Tabela 12.19: Lista instrumentelor de conversie a codului sursă

12.9 Crearea pachetului Debian

Dacă doriți să creați un pachet Debian, citiți informațiile de mai jos.

- Cap. 2 pentru a înțelege principiile de bază ale sistemului de pachete
- Secțiune 2.7.13 pentru a înțelege principiile de bază ale procesului de portabilitate
- Secțiune 9.11.4 pentru a înțelege tehnicile de bază ale „chroot”
- debuild(1), și sbuild(1)
- Secțiune 12.5.2 pentru recompilarea în scopul depanării
- Guide for Debian Maintainers - Ghid pentru administratorii Debian (pachetul `debmake-doc`)
- Debian Developer’s Reference - Ghidul de referință pentru dezvoltatorii Debian (pachetul `developers-reference`)
- Debian Policy Manual - Manualul de politici Debian (pachetul `debian-policy`)

Există pachete precum `debmake`, `dh-make`, `dh-make-perl` etc., care facilitează procesul de creare a pachetelor.

Anexa A

Anexa

Iată contextul acestui document.

A.1 Labirintul Debian

Sistemul Linux este o platformă informatică extrem de puternică pentru un calculator conectat la rețea. Cu toate acestea, învățarea modului de utilizare a tuturor funcționalităților sale nu este ușoară. Configurarea cozii de imprimare LPR cu o imprimantă care nu acceptă PostScript a fost un bun exemplu de dificultate; (acum nu mai există probleme, deoarece instalările mai recente utilizează noul sistem CUPS).

Există o hartă completă și detaliată numită „SOURCE CODE”. Aceasta este foarte precisă, dar foarte greu de înțeles. Există, de asemenea, ghiduri numite HOWTO și mini-HOWTO. Acestea sunt mai ușor de înțeles, dar au tendința de a oferi prea multe detalii și de a pierde din vedere imaginea de ansamblu. Uneori am dificultăți în a găsi secțiunea potrivită într-un ghid HOWTO lung atunci când am nevoie doar de câteva comenzi pe care să le execut.

Sper că această „Referință Debian (versiunea 2.152)” (2026-10-10 14:56:25 UTC) va oferi un bun punct de plecare pentru cei care se află în labirintul Debian.

A.2 Istoricul drepturilor de autor

Referința Debian a fost inițiată de mine, Osamu Aoki <osamu at debian dot org>, ca un ghid personal de administrare a sistemului. O mare parte din conținut provine din cunoștințele pe care le-am dobândit din lista de discuții debian-user și din alte resurse Debian.

În urma unei sugestii a lui Josip Rodin, care a fost foarte activ în cadrul Proiectului de documentare Debian (DDP), „Referința Debian (versiunea 1, 2001-2007)” a fost creată ca parte a documentației DDP.

După 6 ani, mi-am dat seama că versiunea inițială a „Referinței Debian (versiunea 1)” era depășită și am început să rescriu o mare parte din conținut. Noua „Referință Debian (versiunea 2)” a fost lansată în 2008.

Am actualizat „Referința Debian (versiunea 2)” pentru a include subiecte noi (Systemd, Wayland, IMAP, PipeWire, nucleul Linux 5.10) și am eliminat subiectele învechite (SysV init, CVS, Subversion, protocolul SSH 1, nucleele Linux anterioare versiunii 2.5). Referințele la versiuni „Jessie 8 (2015-2020)” sau mai vechi au fost eliminate în mare parte.

Această „Referință Debian (versiunea 2.152)” (2026-10-10 14:56:25 UTC) acoperă în principal versiunile Debian Trixie (`=stable`) și Forky (`=testing`).

Conținutul tutorialului își are originea și sursa de inspirație în următoarele elemente.

- ”Linux User’s Guide” de Larry Greenfield (decembrie 1996)

  - înlocuit de „Debian Tutorial”

- „Debian Tutorial” de Havoc Pennington. (11 decembrie 1998)
  - scris parțial de Oliver Elphick, Ole Tetlie, James Treacy, Craig Sawyer și Ivan E. Moore II
  - înlocuit de „Debian GNU/Linux: Ghidul de instalare și utilizare”
- ”Debian GNU/Linux: Guide to Installation and Usage” de John Goerzen și Ossama Othman (1999)
  - înlocuit de „Referința Debian (versiunea 1)”

Descrierea pachetului și a arhivei oferă câteva indicii despre originea și sursa de inspirație a acestora, după cum urmează.

- ”Debian FAQ” (Versiunea din martie 2002, când acest sit era administrat de Josip Rodin)

Celelalte conținuturi își pot găsi o parte din origini și surse de inspirație în cele ce urmează.

- „Referința Debian (versiunea 1)“ de Osamu Aoki (2001–2007)
  - înlocuită de noua „Referință Debian (versiunea 2)” în 2008.

Versiunea anterioară a „Referinței Debian (versiunea 1)” a fost realizată cu ajutorul multor colaboratori.

- contribuția principală la conținutul referitor la configurarea rețelelor, realizată de Thomas Hood
- o contribuție semnificativă la conținutul referitor la subiecte legate de X și VCS, realizată de Brian Nelson
- ajutorul acordat în ceea ce privește scripturile de compilare și numeroasele corecturi de conținut realizate de Jens Seidel
- corectură amănunțită realizată de David Sewell
- numeroase contribuții din partea traducătorilor, colaboratorilor și celor care au semnalat erori

Pentru redactarea acestui document s-au folosit ca referințe principale numeroase pagini de manual și pagini de informații din sistemul Debian, precum și pagini web ale dezvoltatorilor inițiali și documente de pe Wikipedia. În măsura în care Osamu Aoki a considerat că se încadrează în utilizarea corectă, multe părți din acestea, în special definițiile comenzilor, au fost folosite ca fragmente de fraze după eforturi editoriale atente pentru a le adapta stilului și obiectivului acestui document.

Descrierea depanatorului gdb a fost completată cu ajutorul conținutului wiki Debian referitor la urmărirea traseului de execuție, cu acordul lui Ari Pollak, Loïc Minier și Dafydd Harries.

Conținutul actualei „Referințe Debian (versiunea 2.152)” (2026-10-10 14:56:25 UTC) este în mare parte opera mea, cu excepția cazurilor menționate mai sus. Acesta a fost actualizat și de către colaboratori.

Autorul, Osamu Aoki, mulțumește tuturor celor care au contribuit la realizarea acestui document.

A.3 Formatul documentului

Sursa documentului original în limba engleză este redactată în prezent în fișiere XML DocBook. Aceste fișiere sursă XML DocBook sunt convertite în HTML, text simplu, PostScript și PDF; (anumite formate pot fi omise la distribuire).