



debian

Ghidul de referință Debian

Osamu Aoki

Copyright © 2013-2024 Osamu Aoki

Acest Ghid de referință Debian (versiunea 2.138) (2026-02-24 09:02:45 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	Permițiuni ale sistemului de fișiere	8
1.2.4	Controlul permișiunilor pentru fișierele nou create: umask	11
1.2.5	Permițiuni 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	Combinatii 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	33
1.6.1	Instrumente pentru text în Unix	34
1.6.2	Expresii regulate	36
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	38
1.6.6	Fragmente de script pentru comenzi de direcționare	39

2	Gestionarea pachetelor Debian	41
2.1	Cerințe preliminare pentru gestionarea pachetelor Debian	41
2.1.1	Sistemul de gestionare a pachetelor Debian	41
2.1.2	Configurația pachetului	42
2.1.3	Precauții de bază	42
2.1.4	O viață cu îmbunătățiri veșnice	43
2.1.5	Noțiuni de bază despre arhiva Debian	44
2.1.6	Debian este software 100% liber	48
2.1.7	Dependențele pachetelor	49
2.1.8	Fluxul evenimentelor din gestionarea pachetelor	50
2.1.9	Primul răspuns la problemele legate de gestionarea pachetelor	51
2.1.10	Cum să selectați pachetele Debian	52
2.1.11	Cum să faceți față cerințelor contradictorii	52
2.2	Operații de bază de gestionare a pachetelor	53
2.2.1	apt vs. apt-get / apt-cache vs. aptitude	53
2.2.2	Operații de bază de gestionare a pachetelor din linia de comandă	54
2.2.3	Utilizarea interactivă a «aptitude»	56
2.2.4	Combinatii de taste pentru aptitude	56
2.2.5	Vizualizarea de pachete în aptitude	57
2.2.6	Opțiuni pentru metoda de căutare cu aptitude	58
2.2.7	Formula expresiei regulate pentru aptitudine	58
2.2.8	Rezolvarea dependențelor de către aptitude	60
2.2.9	Jurnale de activitate ale pachetelor	60
2.3	Exemple de operații cu aptitudine	60
2.3.1	Căutarea de pachete interesante	60
2.3.2	Listarea pachetelor care se potrivesc cu expresia regulată în numele pachetelor	61
2.3.3	Navigarea prin rezultatul potrivirilor expresiei regulate	61
2.3.4	Stergerea definitivă a pachetelor eliminate	61
2.3.5	Reorganizarea stării instalării automate/manuale	61
2.3.6	Actualizare la nivel de sistem	62
2.4	Operații avansate de gestionare a pachetelor	63
2.4.1	Operații avansate de gestionare a pachetelor din linia de comandă	63
2.4.2	Verificarea fișierelor pachetului instalat	65
2.4.3	Protejarea împotriva problemelor legate de pachete	65
2.4.4	Căutarea în metadatele pachetului	65
2.5	Detalii interne privind gestionarea pachetelor Debian	66
2.5.1	Metadate de arhivă	66
2.5.2	Fișierul „Release” de nivel superior și autenticitatea	66
2.5.3	Fișiere „Release” la nivel de arhivă	67

2.5.4	Preluarea metadatelor pentru pachet	68
2.5.5	Starea pachetului pentru APT	68
2.5.6	Starea pachetului pentru aptitude	69
2.5.7	Copiile locale ale pachetelor descărcate	69
2.5.8	Numele fișierelor pachetelor Debian	69
2.5.9	Comanda «dpkg»	70
2.5.10	Comanda «update-alternatives»	71
2.5.11	Comanda «dpkg-statoverride»	71
2.5.12	Comanda «dpkg-divert»	71
2.6	Recuperarea dintr-un sistem defect	72
2.6.1	Incompatibilitate cu configurația veche a utilizatorului	72
2.6.2	Erori de stocare în cache ale datelor pachetului	72
2.6.3	Recuperarea cu comanda dpkg	72
2.6.4	Instalare eșuată din cauza dependențelor lipsă	73
2.6.5	Pachete diferite cu fișiere suprapuse	73
2.6.6	Remediarea scriptului pachetului defect	73
2.6.7	Recuperarea datelor privind selecția pachetelor	74
2.7	Sfaturi pentru gestionarea pachetelor	74
2.7.1	Cine a încărcat pachetul?	74
2.7.2	Limitarea lățimii de bandă pentru descărcare pentru APT	75
2.7.3	Descărcarea și actualizarea automată a pachetelor	75
2.7.4	Actualizări și retro-adaptări (versiuni de software migrate din ramura principală de dezvoltare și adaptate pentru a funcționa cu această versiune)	75
2.7.5	Arhive de pachete externe	76
2.7.6	Pachete din surse mixte de arhive fără apt-pinning	76
2.7.7	Ajustarea versiunii candidate cu apt-pinning	77
2.7.8	Blocarea pachetelor instalate de „Recommends”	79
2.7.9	Urmărirea suitei <code>testing</code> cu unele pachete din <code>unstable</code>	79
2.7.10	Urmărirea suitei <code>unstable</code> cu unele pachete din <code>experimental</code>	80
2.7.11	Retrogradarea de urgență	81
2.7.12	Pachetul <code>equivs</code>	82
2.7.13	Adaptarea unui pachet la sistemul stabil	82
2.7.14	Server proxy pentru APT	83
2.7.15	Mai multe informații despre gestionarea pachetelor	83

3	Inițializarea sistemului	84
3.1	O prezentare generală a procesului de inițializare	84
3.1.1	Etapa 1: UEFI	85
3.1.2	Etapa 2: încărcătorul de pornire	85
3.1.3	Etapa 3: sistemul mini-Debian	86
3.1.4	Etapa 4: sistemul Debian normal	87
3.2	Rescue system	88
3.2.1	GRUB UEFI rescue system on USB	89
3.2.2	Linux live rescue system on USB	89
3.2.3	Linux live rescue system from GRUB	90
3.3	Systemd	90
3.3.1	Init systemd	90
3.3.2	Autentificarea cu systemd	91
3.4	Mesajele nucleului	92
3.5	Mesajele sistemului	92
3.6	Gestionarea sistemului	93
3.7	Alte monitoare de sistem	93
3.8	Configurația sistemului	93
3.8.1	Numele gazdei	93
3.8.2	Sistemul de fișiere	93
3.8.3	Inițializarea interfeței de rețea	95
3.8.4	Inițializarea sistemului cloud	95
3.8.5	Exemplu de personalizare pentru ajustarea serviciului sshd	95
3.9	Sistemul udev	96
3.10	Inițializarea modulelor de nucleu	96
4	Autentificare și controale de acces	98
4.1	Autentificare normală Unix	98
4.2	Gestionarea informațiilor privind contul și parola	100
4.3	Parolă bună	100
4.4	Crearea unei parole criptate	101
4.5	PAM și NSS	101
4.5.1	Fișierele de configurare accesate de PAM și NSS	102
4.5.2	Sistemul modern de gestionare centralizată	103
4.5.3	„De ce comanda «su» a GNU nu acceptă grupul wheel”	103
4.5.4	Reguli mai stricte privind parolele	104
4.6	Securitatea autentificării	104
4.6.1	Parolă sigură în Internet	104
4.6.2	Shell securizat	105

4.6.3	Măsurile suplimentare de securitate pentru Internet	105
4.6.4	Securizarea parolei root	105
4.7	Alte controale de acces	106
4.7.1	Liste de control al accesului (ACL)	106
4.7.2	sudo	107
4.7.3	PolicyKit	107
4.7.4	Restricționarea accesului la anumite servicii ale serverului	107
4.7.5	Caracteristici de securitate Linux	108
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 poșta 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	126
6.2.4.3	Configurarea adresei de poștă electronică	127
6.2.4.4	Operații MTA de bază	128
6.3	Serverul și instrumentele de acces la distanță (SSH)	128
6.3.1	Noțiuni de bază despre SSH	129
6.3.2	Numele de utilizator pe gazda la distanță	129
6.3.3	Connectarea fără parole la distanță	130
6.3.4	Gestionarea clienților SSH străini	130
6.3.5	Configurarea ssh-agent	130
6.3.6	Trimiterea unui mesaj de la o gazdă de la distanță	131
6.3.7	Redirecționarea porturilor pentru tunelarea SMTP/POP3	131
6.3.8	Cum să opriți sistemul de la distanță pe SSH	131
6.3.9	Soluționarea problemelor SSH	131
6.4	Serverul de imprimare și utilitățile	132
6.5	Alte servere de aplicații de rețea	132
6.6	Alți clienți de aplicații de rețea	133
6.7	Diagnosticul demonilor sistemului	133
7	Sistemul de interfață grafică	135
7.1	Mediul de birou cu interfață grafică	135
7.2	Protocolul de comunicare al interfeței grafice	136
7.3	Infrastructura de interfață grafică	137
7.4	Aplicații cu interfață grafică	138
7.5	Directoarele utilizatorilor	138
7.6	Tipuri de litere	138
7.6.1	Fonturi de bază	138
7.6.2	Conversia și redarea fonturilor	141
7.7	Cutia cu nisip (sandbox)	141
7.8	Mediu de birou la distanță	143
7.9	Conexiune la serverul X	143
7.9.1	Conexiune locală la serverul X	143
7.9.2	Conexiune de la distanță la serverul X	144
7.9.3	Conexiune chroot la serverul X	144
7.10	Clipboard	144

8	I18N și L10N	146
8.1	Configurația regională	146
8.1.1	Justificarea pentru utilizarea UTF-8 în configurația lingvistică	146
8.1.2	Reconfigurarea configurației regionale	147
8.1.3	Codificarea numelor de fișiere	148
8.1.4	Mesaje în limba maternă și documentație tradusă	148
8.1.5	Efectele configurației regionale	149
8.2	Introducere de la tastatură	149
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	150
8.2.4	Un exemplu pentru japoneză	150
8.3	Afișarea ieșirii	151
8.4	Caractere cu lățime de aspect ambiguu din Asia de Est	151
9	Sfaturi privind sistemul	152
9.1	Sfaturi pentru consolă	152
9.1.1	Înregistrarea activităților shell în mod curat	152
9.1.2	Programul «screen»	153
9.1.3	Navigarea între directoare	154
9.1.4	Readline wrapper (învăluitorul «readline»)	154
9.1.5	Scanarea arborelui codului sursă	154
9.2	Personalizarea vim	155
9.2.1	Personalizarea vim cu caracteristicile interne	155
9.2.2	Personalizarea vim cu pachete externe	157
9.3	Înregistrarea și prezentarea datelor	158
9.3.1	Demonul de jurnalizare	158
9.3.2	Analizator de jurnale	158
9.3.3	Afișarea personalizată a datelor textuale	159
9.3.4	Afișare personalizată a orei și datei	159
9.3.5	Ecou colorat al shell-ului	159
9.3.6	Comenzi colorate	160
9.3.7	Înregistrarea activităților editorului pentru repetări complexe	160
9.3.8	Înregistrarea imaginii grafice a unei aplicații X	161
9.3.9	Înregistrarea modificărilor din fișierele de configurare	161
9.4	Monitorizarea, controlul și inițierea activităților programului	163
9.4.1	Cronometrarea unui proces	163
9.4.2	Prioritatea de planificare	163
9.4.3	Comanda «ps»	163

9.4.4	Comanda «top»	164
9.4.5	Listarea fișierelor deschise de un proces	164
9.4.6	Urmărirea activităților programului	164
9.4.7	Identificarea proceselor care utilizează fișiere sau socluri	164
9.4.8	Repetarea unei comenzi la intervale constante	164
9.4.9	Repetarea unei comenzi care parcurge fișierele	165
9.4.10	Pornirea unui program din interfața grafică	165
9.4.11	Personalizarea programului care urmează să fie pornit	166
9.4.12	Omorârea unui proces	167
9.4.13	Programarea sarcinilor o singură dată	167
9.4.14	Programarea regulată a sarcinilor	167
9.4.15	Programarea sarcinilor la eveniment	169
9.4.16	Tasta Alt-SysRq	169
9.5	Sfaturi pentru întreținerea sistemului	170
9.5.1	Cine este în sistem?	170
9.5.2	Avertisment pentru toată lumea	170
9.5.3	Identificarea hardware-ului	171
9.5.4	Configurația hardware	171
9.5.5	Ora sistemului și ora hardware-ului	172
9.5.6	Configurația terminalului	172
9.5.7	Infrastructura de sunet	173
9.5.8	Dezactivarea protectorului de ecran	173
9.5.9	Dezactivarea bipurilor sonore	174
9.5.10	Utilizare memorie	174
9.5.11	Verificarea securității și integrității sistemului	175
9.6	Sfaturi pentru stocarea datelor	175
9.6.1	Utilizarea spațiului pe disc	176
9.6.2	Configurarea partițiilor de disc	176
9.6.3	Accesarea partiției folosind UUID	177
9.6.4	LVM2	177
9.6.5	Configurarea sistemului de fișiere	177
9.6.6	Crearea sistemului de fișiere și verificarea integrității	178
9.6.7	Optimizarea sistemului de fișiere prin opțiuni de montare	179
9.6.8	Optimizarea sistemului de fișiere prin super-bloc	179
9.6.9	Optimizarea discului dur	179
9.6.10	Optimizarea discului cu stare solidă	180
9.6.11	Utilizarea SMART pentru a prezice defectarea discului dur	180
9.6.12	Specificați directorul de stocare temporară prin \$TMPDIR	180
9.6.13	Extinderea spațiului de stocare utilizabil prin LVM	180

9.6.14	Extinderea spațiului de stocare utilizabil prin montarea unei alte partiții	181
9.6.15	Extinderea spațiului de stocare utilizabil prin montarea unui alt director	181
9.6.16	Extinderea spațiului de stocare utilizabil prin montarea suprapusă a unui alt director	181
9.6.17	Extinderea spațiului de stocare utilizabil folosind legături simbolice	181
9.7	Imaginea discului	182
9.7.1	Crearea fișierului imagine de disc	182
9.7.2	Scrierea direct pe disc	182
9.7.3	Montarea fișierului imagine disc	183
9.7.4	Curățarea unui fișier imagine de disc	184
9.7.5	Crearea fișierului imagine de disc gol	185
9.7.6	Crearea fișierului imagine ISO9660	185
9.7.7	Scrierea în mod direct pe CD/DVD-R/RW	186
9.7.8	Montarea fișierului imagine ISO9660	186
9.8	Datele binare	187
9.8.1	Vizualizarea și editarea datelor binare	187
9.8.2	Manipularea fișierelor fără montarea discului	187
9.8.3	Redundanța datelor	187
9.8.4	Recuperarea fișierelor de date și analiza tehnico-criminalistică	189
9.8.5	Împărțirea unui fișier mare în fișiere mici	189
9.8.6	Ștergerea conținutului fișierului	189
9.8.7	Fișiere fictive	189
9.8.8	Ștergerea întregului disc dur	190
9.8.9	Ștergerea zonei neutilizate a unui disc dur	190
9.8.10	Recuperarea fișierelor șterse, dar încă deschise	190
9.8.11	Căutarea tuturor legăturilor dure	191
9.8.12	Consum invizibil de spațiu pe disc	191
9.9	Sfaturi pentru criptarea datelor	192
9.9.1	Criptarea discurilor amovibile cu dm-crypt/LUKS	192
9.9.2	Montarea discului criptat cu dm-crypt/LUKS	193
9.10	Nucleul	193
9.10.1	Parametrii nucleului	193
9.10.2	Antetele nucleului	193
9.10.3	Compilarea nucleului și a modulelor aferente	194
9.10.4	Compilarea sursei nucleului: Recomandarea echipei Debian Kernel	194
9.10.5	Controlori hardware și firmware	195
9.11	Sistem virtualizat	196
9.11.1	Instrumente de virtualizare și emulare	196
9.11.2	Fluxul de lucru pentru virtualizare	198
9.11.3	Montarea fișierului imagine al discului virtual	198
9.11.4	Sistemul chroot	199
9.11.5	Mai multe sisteme de medii de birou	200

10 Gestionarea datelor	201
10.1 Sharing, copying, and archiving	201
10.1.1 Archive and compression tools	202
10.1.2 Copy and synchronization tools	202
10.1.3 Idioms for the archive	202
10.1.4 Idioms for the copy	204
10.1.5 Idioms for the selection of files	205
10.1.6 Archive media	206
10.1.7 Removable storage device	207
10.1.8 Filesystem choice for sharing data	208
10.1.9 Sharing data via network	209
10.2 Backup and recovery	210
10.2.1 Backup and recovery policy	210
10.2.2 Backup utility suites	211
10.2.3 Backup tips	211
10.2.3.1 GUI backup	213
10.2.3.2 Mount event triggered backup	213
10.2.3.3 Timer event triggered backup	214
10.3 Data security infrastructure	215
10.3.1 Key management for GnuPG	215
10.3.2 Using GnuPG on files	217
10.3.3 Using GnuPG with Mutt	217
10.3.4 Using GnuPG with Vim	217
10.3.5 Suma MD5	219
10.3.6 Password keyring	219
10.4 Source code merge tools	219
10.4.1 Extracting differences for source files	219
10.4.2 Merging updates for source files	221
10.4.3 Interactive merge	221
10.5 Git	221
10.5.1 Configuration of Git client	221
10.5.2 Basic Git commands	222
10.5.3 Git tips	223
10.5.4 Git references	223
10.5.5 Other version control systems	225

11 Data conversion	226
11.1 Text data conversion tools	226
11.1.1 Converting a text file with iconv	226
11.1.2 Checking file to be UTF-8 with iconv	228
11.1.3 Converting file names with iconv	228
11.1.4 EOL conversion	229
11.1.5 TAB conversion	229
11.1.6 Editors with auto-conversion	230
11.1.7 Plain text extraction	230
11.1.8 Highlighting and formatting plain text data	230
11.2 XML data	230
11.2.1 Basic hints for XML	232
11.2.2 XML processing	233
11.2.3 The XML data extraction	233
11.2.4 The XML data lint	233
11.3 Type setting	234
11.3.1 roff typesetting	234
11.3.2 TeX/LaTeX	235
11.3.3 Pretty print a manual page	235
11.3.4 Creating a manual page	235
11.4 Printable data	236
11.4.1 Ghostscript	236
11.4.2 Merge two PS or PDF files	236
11.4.3 Printable data utilities	237
11.4.4 Imprimarea cu CUPS	237
11.5 The mail data conversion	238
11.5.1 Mail data basics	238
11.6 Graphic data tools	239
11.6.1 Graphic data tools (metapackage)	239
11.6.2 Graphic data tools (GUI)	239
11.6.3 Graphic data tools (CLI)	239
11.7 Miscellaneous data conversion	242
12 Programare	243
12.1 The shell script	243
12.1.1 POSIX shell compatibility	244
12.1.2 Shell parameters	244
12.1.3 Condiționale shell	246
12.1.4 Bucle shell	246

12.1.5 Shell environment variables	247
12.1.6 The shell command-line processing sequence	247
12.1.7 Utility programs for shell script	248
12.2 Scripting in interpreted languages	249
12.2.1 Debugging interpreted language codes	249
12.2.2 GUI program with the shell script	250
12.2.3 Custom actions for GUI filer	250
12.2.4 Perl short script madness	251
12.3 Coding in compiled languages	251
12.3.1 C	251
12.3.2 Simple C program (gcc)	252
12.3.3 Flex — a better Lex	253
12.3.4 Bison — a better Yacc	253
12.4 Static code analysis tools	254
12.5 Depanare	255
12.5.1 Basic gdb execution	255
12.5.2 Debugging the Debian package	256
12.5.3 Obtaining backtrace	257
12.5.4 Advanced gdb commands	257
12.5.5 Check dependency on libraries	257
12.5.6 Dynamic call tracing tools	258
12.5.7 Debugging X Errors	258
12.5.8 Memory leak detection tools	258
12.5.9 Disassemble binary	258
12.6 Instrumentele de construcție	259
12.6.1 Make	259
12.6.2 Autotools	260
12.6.2.1 Compile and install a program	260
12.6.2.2 Uninstall program	260
12.6.3 Meson	260
12.7 Web	261
12.8 The source code translation	261
12.9 Crearea pachetului Debian	262
A Appendix	263
A.1 The Debian maze	263
A.2 Copyright history	263
A.3 Formatul documentului	264

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 comenzii «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	Metacactere 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	42
2.2	Lista siturilor de arhivă Debian	46

2.3	Lista secțiunilor de arhivă Debian	46
2.4	Relația dintre versiune și numele în cod	47
2.5	Lista siturilor web importante pentru rezolvarea problemelor legate de un pachet specific	52
2.6	Operații de bază de gestionare a pachetelor din linia de comandă folosind <code>apt(8)</code> , <code>aptitude(8)</code> și <code>apt-get(8)</code> / <code>apt-cache(8)</code>	55
2.7	Opțiuni de comandă demne de menționat pentru <code>aptitude(8)</code>	55
2.8	Lista combinațiilor de taste pentru « <code>aptitude</code> »	56
2.9	Lista vizualizărilor pentru <code>aptitude</code>	57
2.10	Clasificarea vizualizărilor standard ale pachetelor	58
2.11	Lista formulelor de expresii regulate pentru <code>aptitude</code>	59
2.12	Fișierele jurnal pentru activitățile pachetului	60
2.13	Lista operațiilor avansate de gestionare a pachetelor	64
2.14	Conținutul metadatelor arhivei Debian	66
2.15	Structura numelor pachetelor Debian	69
2.16	Caracterele utilizabile pentru fiecare componentă din numele pachetelor Debian	69
2.17	Fișierele importante create de <code>dpkg</code>	70
2.18	Lista valorilor notabile (celor mai importante) ale priorității Pin pentru tehnica apt-pinning	78
2.19	Lista instrumentelor proxy special pentru arhiva Debian	83
3.1	Lista încărcătorilor de pornire	85
3.2	Semnificația intrării din meniul din partea de sus a <code>/boot/grub/grub.cfg</code>	86
3.3	Lista instrumentelor de pornire pentru sistemul Debian	88
3.4	Lista nivelurilor de eroare ale nucleului	92
3.5	Lista fragmentelor tipice de comandă <code>journalctl</code>	92
3.6	Lista fragmentelor tipice de comenzi <code>systemctl</code>	94
3.7	Lista altor fragmente de comenzi de monitorizare sub <code>systemd</code>	95
4.1	3 fișiere de configurare importante pentru <code>pam_unix(8)</code>	98
4.2	Al doilea conținut al intrării „ <code>/etc/passwd</code> ”	99
4.3	Lista comenzilor pentru gestionarea informațiilor contului	100
4.4	Lista instrumentelor pentru generarea parolei	101
4.5	Lista pachetelor principale ale sistemelor PAM și NSS	102
4.6	Lista fișierelor de configurare accesate de PAM și NSS	102
4.7	Lista serviciilor și porturilor nesigure și sigure	104
4.8	Lista instrumentelor care oferă măsuri suplimentare de securitate	105
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 <code>net-tools</code> și noile comenzi <code>iproute2</code>	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	126
6.5	Lista fișierelor de configurare legate de adresele de poștă electronică	127
6.6	Lista operațiilor de bază ale MTA	128
6.7	Lista serverelor și instrumentelor de acces la distanță	129
6.8	Lista fișierelor de configurare SSH	129
6.9	Listă de exemple de pornire a clientului SSH	130
6.10	Lista clienților SSH liberi pentru alte platforme	130
6.11	Lista serverelor de imprimare și a utilităților	132
6.12	Lista altor servere de aplicații de rețea	133
6.13	Lista clienților de aplicații de rețea	134
6.14	Lista RFC-urilor populare	134
7.1	Lista mediilor grafice de birou	135
7.2	Lista pachetelor importante de infrastructură de interfață grafică	137
7.3	Lista aplicațiilor cu interfață grafică notabile	139
7.4	Lista fonturilor notabile TrueType și OpenType	140
7.5	Lista mediilor de fonturi notabile și a pachetelor asociate	141
7.6	Lista mediilor sandbox notabile și a pachetelor asociate	142
7.7	Lista serverelor notabile de acces la distanță	143
7.8	Lista metodelor de conectare la serverul X	143
7.9	Lista programelor legate de manipularea clipboardului de caractere	145
8.1	Lista IBus și a pachetelor sale de motoare	150
9.1	Lista programelor ce ajută în activitățile cu consola	152
9.2	Lista combinațiilor de taste pentru «screen»	154
9.3	Informații despre inițializarea vim	158
9.4	Lista analizatoarelor de jurnale de sistem	159
9.5	Exemplele de afișare a datei și orei pentru comanda „ls -l” cu valoarea stilului de afișare a datei și orei	160
9.6	Lista instrumentelor de manipulare a imaginilor grafice	161
9.7	Lista pachetelor care pot înregistra istoricul configurațiilor	161
9.8	Lista instrumentelor pentru monitorizarea și controlul activităților programului	162
9.9	Lista valorilor de curtoazie pentru prioritatea de programare	163

9.10	Lista stilurilor de comenzi ps	163
9.11	Lista semnalelor utilizate frecvent pentru comanda «kill»	168
9.12	Lista tastelor de comandă SAK importante	170
9.13	Lista instrumentelor de identificare a hardware-ului	171
9.14	Lista instrumentelor de configurare hardware	171
9.15	Lista pachetelor de sunet	173
9.16	Lista comenzilor pentru dezactivarea protectorului de ecran	174
9.17	Lista dimensiunilor memoriei raportate	174
9.18	Lista instrumentelor pentru verificarea securității și integrității sistemului	175
9.19	Lista pachetelor de gestionare a partițiilor de disc	176
9.20	Lista pachetelor de gestionare a sistemului de fișiere	178
9.21	Lista pachetelor care vizualizează și editează date binare	187
9.22	Lista pachetelor pentru manipularea fișierelor fără montarea discului	187
9.23	Lista instrumentelor pentru adăugarea redundanței datelor la fișiere	188
9.24	Lista pachetelor pentru recuperarea fișierelor de date și analiza tehnico-criminalistică	188
9.25	Lista instrumentelor de criptare a datelor	192
9.26	Lista pachetelor cheie care trebuie instalate pentru recompilarea nucleului pe sistemul Debian	194
9.27	Lista instrumentelor de virtualizare	197
10.1	List of archive and compression tools	203
10.2	List of copy and synchronization tools	204
10.3	List of filesystem choices for removable storage devices with typical usage scenarios	208
10.4	List of the network service to chose with the typical usage scenario	209
10.5	List of backup suite utilities	212
10.6	List of data security infrastructure tools	215
10.7	List of GNU Privacy Guard commands for the key management	216
10.8	List of the meaning of the trust code	216
10.9	List of GNU Privacy Guard commands on files	218
10.10	List of source code merge tools	220
10.11	List of git related packages and commands	221
10.12	Main Git commands	223
10.13	Git tips	224
10.14	List of other version control system tools	225
11.1	List of text data conversion tools	226
11.2	List of encoding values and their usage	227
11.3	List of EOL styles for different platforms	229
11.4	List of TAB conversion commands from <code>bsdmainutils</code> and <code>coreutils</code> packages	229
11.5	List of tools to extract plain text data	231

11.6 List of tools to highlight plain text data	231
11.7 List of predefined entities for XML	232
11.8 List of XML tools	233
11.9 List of DSSSL tools	233
11.10List of XML data extraction tools	234
11.11List of XML pretty print tools	234
11.12List of type setting tools	234
11.13List of packages to help creating the manpage	236
11.14List of Ghostscript PostScript interpreters	236
11.15List of printable data utilities	237
11.16List of packages to help mail data conversion	238
11.17List of graphics data tools (metapackage)	239
11.18List of graphics data tools (GUI)	240
11.19List of graphics data tools (CLI)	241
11.20List of miscellaneous data conversion tools	241
12.1 List of typical bashisms	244
12.2 List of shell parameters	245
12.3 List of shell parameter expansions	245
12.4 List of key shell parameter substitutions	245
12.5 List of file comparison operators in the conditional expression	246
12.6 List of string comparison operators in the conditional expression	247
12.7 List of packages containing small utility programs for shell scripts	248
12.8 List of interpreter related packages	249
12.9 List of dialog programs	250
12.10List of compiler related packages	252
12.11List of Yacc-compatible LALR parser generators	253
12.12List of tools for static code analysis	255
12.13List of debug packages	255
12.14List of advanced gdb commands	258
12.15List of memory leak detection tools	258
12.16List of build tool packages	259
12.17List of make automatic variables	259
12.18List of make variable expansions	259
12.19List of source code translation tools	261

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 \(versiunea 2.138\)](#) (2026-02-24 09:02:45 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 \(TLPD\)](#)

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-account* (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 „Apăsăți 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-groups.html” 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 2026-02-23 11:12:43 UTC și conțin totalul de 280503 de raportări referitoare la 215455 de pachete binare și 26 de arhitecturi.

Notă

Vă rugăm să rețineți că arhiva amd64 unstable conține în prezent numai pachete 73987. 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
 - Always have an alternative boot media ([USB flash drive](#), [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

Permisunile 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 acestora 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” la promptul de comandă. Este posibil să nu puteți vedea comanda introdusă pe măsură ce o tastați. De asemenea, puteți tasta „*c lear*” 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)	descriere	
mc	V:43, I:183	1590	Un gestionar de fișiere în modul text pe ecran complet
sudo	V:732, I:863	6773	Un program care permite utilizatorilor privilegii root limitate
vim	V:86, I:346	4089	Editorul de text Unix Vi IMproved, un editor de text pentru programatori (versiunea standard)
vim-tiny	V:57, I:978	1877	Editorul de text Unix Vi IMproved, un editor de text pentru programatori (versiunea compactă)
emacs-nox	V:3, I:13	46536	Proiectul GNU Emacs, editorul de text extensibil bazat pe Lisp
w3m	V:11, I:145	2853	Navigatoare web în modul text
gpm	V:9, I:9	526	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)	descriere	
doc-debian	I:882	187	Documentația proiectului Debian (Întrebări frecvente despre Debian) și alte documente
debian-policy	I:8	5061	Manualul de politici Debian și documentele conexe
developers-reference	V:0, I:2	2601	Ghiduri și informații pentru dezvoltatorii Debian
debmake-doc	I:0	11807	Ghid pentru responsabili cu întreținerea pachetelor Debian
debian-history	I:0	6251	Istoria proiectului Debian
debian-faq	I:880	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, alegeți 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 Permisuni ale sistemului de fișiere

Permisunile 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.

- Permisunea de **citare - (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 **screre - (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 propri-
etarului 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

Permișunile 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	-rwx-r--r--	-rwxr-xr-x	poate fi scris numai de către utilizator
0002	-rwx-rwx-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 Permițiuni 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 NetworkManager`»
 - 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).

The hardware devices are just another kind of file on the Debian system. If you have problems accessing devices such as [USB flash drive](#) and [CD-ROM](#) from a user account, you should make that user a member of the relevant group.

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

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

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)`.

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

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.

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 marcasele temporale ale fișierelor existente.

Pentru marcasele 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”.

- [Legătură dură](#)
 - 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 Conducute 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. Conducutele 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

Soclrile 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

«netstat -an» 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 incompreensibile 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/scriș 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 redirectionarea 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 rulării; (puteți face același lucru prin comanda specializată `sysctl(8)` sau prin fișierul său de preîncărcare/configurare „/etc/sysctl.conf”).

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” conțin 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 executacd 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. Apăsăți 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 [9.5.6](#).

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-Fn` 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	directionează conținutul către software-ul de vizualizare
fișier html	directionează 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-1 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)	limbaj	shell POSIX	descriere
bash	V:866, I:999	7277	Da	Bash : GNU Bourne Again SHell (standard de facto)
bash-completion	V:35, I:953	1952	N/D	completare programabilă pentru shell-ul bash
dash	V:906, I:998	207	Da	Debian Almquist Shell , bun pentru scripturi shell
zsh	V:40, I:70	2509	Da	Z shell : shell-ul standard cu numeroase îmbunătățiri
tcsh	V:4, I:15	1366	Nu	TENEX C Shell : o versiune îmbunătățită a Berkeley csh
mksh	V:5, I:8	7713	Da	O versiune a Korn shell
csh	V:1, I:5	348	Nu	OpenBSD C Shell , o versiune a Berkeley csh
sash	V:0, I:5	1335	Da	Shell autonom cu comenzi încorporate (nu este destinat pentru standardul „/usr/bin/sh”)
ksh	I:8	65	Da	Versiunea reală, AT&T a Korn shell
rc	V:0, I:0	182	Nu	Implementarea AT&T Plan 9 rc shell
posh	V:0, I:0	187	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
<code>Ctrl-U</code>	șterge linia din fața cursorului
<code>Ctrl-H</code>	șterge un caracter înaintea cursorului
<code>Ctrl-D</code>	termină introducerea („ieșire din shell” dacă utilizați shell)
<code>Ctrl-C</code>	termină un program în execuție
<code>Ctrl-Z</code>	oprește temporar programul mutându-l în fundal
<code>Ctrl-S</code>	oprește afișarea pe ecran
<code>Ctrl-Q</code>	reactivează ieșirea pe ecran
<code>Ctrl-Alt-Del</code>	repornește/oprește sistemul, a se vedea <code>inittab(5)</code>
Tasta <code>Alt</code> stânga (opțional, Tasta <code>Windows</code>)	meta-tastă pentru Emacs și interfețe similare
Săgeată-sus	pornește căutarea istoricului comenzilor sub bash
<code>Ctrl-R</code>	pornește căutarea incrementală în istoricul comenzilor sub bash
<code>Tab</code>	completează introducerea numelui fișierului în linia de comandă sub bash
<code>Ctrl-V Tab</code>	introduce <code>Tab</code> 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)	menu 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 apăstate, 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.

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

comanda	descriere
<code>pwd</code>	afișează numele directorului curent/de lucru
<code>whoami</code>	afișează numele utilizatorului curent
<code>id</code>	afișează identitatea utilizatorului curent (nume, uid, gid și grupurile asociate)
<code>file foo</code>	afișează tipul de fișier pentru fișierul „ <i>foo</i> ”
<code>type -p nume-comandă</code>	afișează locația fișierului comenzii „ <i>nume-comandă</i> ”
<code>which nume-comandă</code>	, ,
<code>type nume-comandă</code>	afișează informații despre comanda „ <i>nume-comandă</i> ”
<code>apropos cuvânt-cheie</code>	găsește comenzi legate de „ <i>cuvânt-cheie</i> ”
<code>man -k cuvânt-cheie</code>	, ,
<code>whatis nume-comandă</code>	afișează o explicație pe o singură linie privind comanda „ <i>nume-comandă</i> ”
<code>man -a nume-comandă</code>	afișează explicația privind comanda „ <i>nume-comandă</i> ” (stil Unix)
<code>info nume-comandă</code>	afișează o explicație destul de lungă privind comanda „ <i>nume-comandă</i> ” (stil GNU)
<code>ls</code>	listează conținutul directorului (fișiere și directoare fără punct)
<code>ls -a</code>	listează conținutul directorului (toate fișierele și directoarele)
<code>ls -A</code>	listează conținutul directorului (aproape toate fișierele și directoarele, adică omite „.” și „.”)
<code>ls -la</code>	listează tot conținutul directorului cu informații detaliate
<code>ls -lai</code>	listează tot conținutul directorului cu numărul de nod-i și informații detaliate
<code>ls -d</code>	listează toate directoarele din directorul curent
<code>tree</code>	afișează conținutul arborelui de fișiere
<code>ls -l foo</code>	afișează starea deschisă a fișierului „ <i>foo</i> ”
<code>ls -l -p pid</code>	listează fișierele deschise de ID-ul procesului: „ <i>pid</i> ”
<code>mkdir foo</code>	crează un nou director „ <i>foo</i> ” în directorul curent
<code>rmdir foo</code>	elimină directorul „ <i>foo</i> ” din directorul curent
<code>cd foo</code>	schimbă directorul în directorul „ <i>foo</i> ” din directorul curent sau din directorul listat în variabila „ <i>\$CDPATH</i> ”
<code>cd /</code>	schimbă directorul la directorul rădăcină
<code>cd</code>	schimbă directorul la directorul personal al utilizatorului curent
<code>cd /foo</code>	schimbă directorul la directorul cu ruta absolută „ <i>/foo</i> ”
<code>cd ..</code>	schimbă directorul la directorul părinte
<code>cd ~foo</code>	schimbă directorul la directorul personal al utilizatorului „ <i>foo</i> ”
<code>cd -</code>	schimbă directorul la directorul anterior
<code></etc/motd pager</code>	afișează conținutul fișierului „ <i>/etc/motd</i> ” folosind paginatorul implicit
<code>touch fișier-test</code>	creați un fișier gol „ <i>fișier-test</i> ”
<code>cp foo bar</code>	copiază un fișier existent „ <i>foo</i> ” într-un fișier nou „ <i>bar</i> ”
<code>rm fișier-test</code>	elimină fișierul „ <i>fișier-test</i> ”
<code>mv foo bar</code>	redenumeste un fișier existent „ <i>foo</i> ” într-un nou nume „ <i>bar</i> ” („ <i>bar</i> ” nu trebuie să existe)
<code>mv foo bar</code>	mută un fișier existent „ <i>foo</i> ” într-o nouă locație „ <i>bar/foo</i> ” (directorul „ <i>bar</i> ” trebuie să existe)
<code>mv foo bar/baz</code>	mută un fișier existent „ <i>foo</i> ” într-o nouă locație cu un nou nume „ <i>bar/baz</i> ” (directorul „ <i>bar</i> ” trebuie să existe, dar directorul „ <i>bar/baz</i> ” nu trebuie să existe)
<code>chmod 600 foo</code>	face ca un fișier existent „ <i>foo</i> ” să fie necitibil și neinscriptibil de către alte persoane (neexecutabil pentru toți)
<code>chmod 644 foo</code>	face ca un fișier existent „ <i>foo</i> ” să fie citibil, dar neinscriptibil de către alte persoane (neexecutabil pentru toți)
<code>chmod 755 foo</code>	faceți ca un fișier existent „ <i>foo</i> ” să fie citibil, dar neinscriptibil de către alte persoane (executabil pentru toți)
<code>find . -name model</code>	găsește numele de fișiere ce coincid folosind shell-ul „ <i>model</i> ” (mai lent)
<code>locate -d . model</code>	găsește numele de fișiere ce coincid folosind shell-ul „ <i>model</i> ” (mai rapid folosind baza de date generată regulat)
<code>grep -e "model" *.html</code>	găsește un „ <i>model</i> ” în toate fișierele care se termină cu „ <i>.html</i> ” din directorul curent și le afișează pe toate
<code>top</code>	afișează informații despre proces pe ecran complet, tastează „q” pentru a ieși
	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`.
- Administratorul de afișare, cum ar fi `gdm3`, poate redefini variabilele de mediu pentru sesiunea de interfață grafică prin `~/ .profile`.
- 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/bin”. 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/utilizator-normal	programul rulează din shell-ul utilizatorului normal
/home/utilizator-normal	programul rulează din meniul mediului grafic de birou al utilizatorului normal
/home/utilizator-normal	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 (segment) nu începe cu „.”
.*	numele fișierului (segment) începe cu „.”
?	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

De exemplu, încercați următoarele

```
$ mkdir junk; cd junk; touch 1.txt 2.txt 3.c 4.h .5.txt ..6.txt
$ echo *.txt
1.txt 2.txt
$ echo *
1.txt 2.txt 3.c 4.h
$ echo *.hc]
3.c 4.h
$ echo .*
. . . .5.txt ..6.txt
$ echo .*[^.]*
.5.txt ..6.txt
$ echo [^1-3]*
4.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
```

```
$ 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.

expresia idiomatică a comenzii	descriere
comanda &	execuție în fundal a comenzii în subshell
comanda1 comanda2	redirecționează ieșirea standard a comenzii1 către intrarea standard a comenzii2 (execuție concomitentă)
comanda1 2>&1 comanda2	redirecționează atât ieșirea standard, cât și eroarea standard a comenzii1 către intrarea standard a comenzii2 (execuție concomitentă)
comanda1 ; comanda2	execută comanda1 și comanda2 secvențial
comanda1 && comanda2	execută comanda1; dacă are succes, execută comanda2 secvențial (returnează succes dacă atât comanda1 cât și comanda2 au succes)
comanda1 comanda2	execută comanda1; dacă nu are succes, execută comanda2 secvențial (returnează succes dacă comanda1 sau comanda2 au succes)
comanda > foo	redirecționează ieșirea standard a comenzii comanda către un fișier foo (suprascrie)
comanda 2> foo	redirecționează ieșirea de eroare standard a comenzii comanda către un fișier foo (suprascrie)
comanda >> foo	redirecționează ieșirea standard a comenzii comanda către un fișier foo (adăugare)
comanda 2>> foo	redirecționează ieșirea de eroare standard a comenzii comanda către un fișier foo (adăugare)
comanda > foo 2>&1	redirecționează atât ieșirea standard, cât și ieșirea de eroare standard a comenzii comanda către un fișier foo
comanda < foo	redirecționează intrarea standard a comenzii comanda către un fișier foo
comanda << delimitator	redirecționează intrarea standard a comenzii comanda către următoarele linii până când se întâlnește „delimitatorul” (aici, document)
comanda <<- delimitator	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

```
$ 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
stdîn	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.
 - `tc(1)` poate efectua orice operație de procesare a textului: consultați `re_syntax(3)`. Se utilizează adesea împreună cu `tk(3)`.
 - `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/h`”.

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.

BRE	ERE	descrierea expresiei regulate
\ . [] ^ \$ *	\ . [] ^ \$ *	metacaractere comune
\+ \? \(\) \{ \} \		BRE doar metacaractere eludate „\”
	+ ? () { }	ERE doar metacaractere fără caracterul de eludare „\”
c	c	potrivește cu un non-metacarakter „c”
\c	\c	potrivește un caracter literal „c” chiar dacă „c” este de fapt un metacarakter
.	.	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

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 **metacarakter** este doar un caracter cu o semnificație specială. Există două stiluri principale, **BRE** și **ERE**, în funcție de instrumentele de text descrise mai sus.

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 conduite (direcționări).

fragment de script (introduceți într-o singură linie)	efectul comenzii
<code>find /usr -print</code>	găsește toate fișierele din „/usr”
<code>seq 1 100</code>	imprimă numere de la 1 până la 100
<code> xargs -n 1 comanda</code>	rulează comanda în mod repetat cu fiecare element din conductă ca argument al său
<code> xargs -n 1 echo</code>	împarte elementele separate prin spații albe din conductă în linii
<code> xargs echo</code>	fuzionează toate liniile din conductă într-o singură linie
<code> grep -e model_expresie-regulată</code>	extrage linii din conductă care conțin <i>model_expresie-regulată</i>
<code> grep -v -e model_expresie-regulată</code>	extrage liniile din conductă care nu conțin <i>model_expresie-regulată</i>
<code> cut -d: -f3 -</code>	extrage al treilea câmp din conductă separat prin „:” (fișier passwd etc.)
<code> awk '{ print \$3 }'</code>	extrage al treilea câmp din șirul separat prin spații
<code> awk -F'\t' '{ print \$3 }'</code>	extrage al treilea câmp din șirul separat prin tabulator
<code> col -bx</code>	elimină „backspace” (retururile de cărucior) și transformă tabulatoarele în spații
<code> expand -</code>	transformă tabulatoarele în spații
<code> sort uniq</code>	sortează și elimină duplicatele
<code> tr 'A-Z' 'a-z'</code>	convertește majusculele în minuscule
<code> tr -d '\n'</code>	concatenează liniile într-o singură linie
<code> tr -d '\r'</code>	elimină CR (retururile de cărucior)
<code> sed 's/^/# /'</code>	adaugă „#” la începutul fiecărei linii
<code> sed 's/\.ext//g'</code>	elimină „.ext”
<code> sed -n -e 2p</code>	imprimă a doua linie
<code> head -n 2 -</code>	imprimă primele 2 linii
<code> tail -n 2 -</code>	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 73987 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)	dimensiune	descriere
dpkg	V:889, I:999	6385	sistem de gestionare a pachetelor de nivel scăzut pentru Debian (bazat pe fișiere)
apt	V:875, I:999	4670	interfață APT pentru gestionarea pachetelor cu CLI:(interfață de linie de comandă) apt / apt-get / apt-cache
aptitude	V:36, I:181	4624	interfață APT pentru gestionarea interactivă a pachetelor cu consolă pe ecran complet: aptitude (8)
tasksel	V:36, I:983	349	interfață APT pentru instalarea sarcinilor selectate: tasksel (8)
unattended-upgrades	V:124, I:184	317	pachet de îmbunătățiri pentru APT pentru a permite instalarea automată a actualizărilor de securitate
gnome-software	V:165, I:274	4476	Centru software pentru GNOME (interfață grafică pentru APT)
synaptic	V:37, I:304	7788	administrator grafic de pachete (interfață GTK APT)
apt-utils	V:399, I:997	1151	programe utilitare APT: apt-extracttemplates (1), apt-ftparchive (1) și apt-sortpkgs (1)
apt-listchanges	V:382, I:888	547	instrument de notificare a istoricului modificărilor pachetelor
apt-listbugs	V:5, I:7	514	listează erorile critice înainte de fiecare instalare APT
apt-file	V:15, I:58	89	instrument de căutare a pachetelor APT — interfață linie de comandă
apt-rdepends	V:0, I:4	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 scrip-tilor 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 sistemul să fie cu **pornire duală** instalând suita `stable` a sistemului Debian pe o altă partiție
- Păstrați CD-ul de instalare la îndemână pentru **pornirea de recuperare**
- 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** în stilul unei singure linii, este ca mai jos:

```
deb http://deb.debian.org/debian/ trixie main non-free-firmware contrib non-free
deb-src 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-src http://security.debian.org/debian-security trixie-security main non-free-firmware ↵
    contrib non-free
```

Alternativ, lista surselor echivalente în stil `deb822` este următoarea.

```
Types: deb deb-src
URIs: http://deb.debian.org/debian/
Suites: trixie
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
```


Punctele cheie ale **listei surselor** sunt următoarele.

- Format cu o singură linie
 - Fișierele sale de definiții se află în fișierul „/etc/apt/sources.list” și în fișierele „/etc/apt/sources.list.d/*”
 - Fiecare linie definește sursa de date pentru sistemul APT.
 - Linia „deb” definește pachetele binare.
 - Linia „deb-src” definește pachetele sursă.
 - Primul argument este adresa URL rădăcină a arhivei Debian.
 - Al doilea argument este numele distribuției, folosind fie numele suitei, fie numele în cod.
 - Al treilea argument și următoarele sunt lista numelor valide ale zonelor de arhivă din arhiva Debian.
- Format stil Deb822
 - 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 meta-datele 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.

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).

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/	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	Actualizări de securitate pentru versiunea stabilă (opțional)
http://deb.debian.org/debian/	stable-updates	trixie	Subset al suitei stable-proposed-updates care necesită actualizări urgente, cum ar fi datele privind fusul orar (opțional)
http://deb.debian.org/debian/	stable-backports	trixie	Colecție aleatorie de pachete recompilate, în mare parte din versiunea testing (opțional)
http://security.debian.org/debian-security/	stable-security	trixie	Actualizări de securitate pentru versiunea stabilă (important)
http://security.debian.org/debian-security/	testing-security	forky	Aceasta nu este susținută în mod activ și nici utilizată de echipa de securitate

Tabela 2.2: Lista siturilor de arhivă Debian

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	72513	conform cu DFSG și fără dependență de non-free
non-free-firmware	64	nu este conform cu DFSG, necesită firmware pentru o experiență rezonabilă de instalare a sistemului
contrib	377	conform cu DFSG, dar cu dependență de non-free
non-free	1033	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.

Calendarul	versiunea = <code>stable</code>	versiunea = <code>testing</code>	versiunea = <code>unstable</code>
după publicarea versiunii <code>trixie</code>	nume în cod = <code>trixie</code>	nume în cod = <code>forky</code>	nume în cod = <code>sid</code>
după publicarea versiunii <code>forky</code>	nume în cod = <code>forky</code>	nume în cod = <code>duke</code>	nume în cod = <code>sid</code>

Tabela 2.4: Relația dintre versiune și numele în cod

Istoria numelor în cod este descrisă în [Debian FAQ: 6.2.1 Ce alte nume în cod au fost folosite în trecut?](#)

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 metadata** („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/”. Dacă configurați shell-ul ca în Secțiune 1.4.2, tasteți 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 \(BTS\)](#).

Căutați cu [Google](#) cu cuvinte cheie precum „site:debian.org”, „site:wiki.debian.org”, „site:lists.debian.org” etc.

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

situl web	comanda
Pagina principală a sistemului de urmărire a erorilor Debian (BTS)	<code>sensible-browser "https://bugs.debian.org/"</code>
Raportul de eroare al unui pachet cunoscut	<code>sensible-browser "https://bugs.debian.org/nume-pachet"</code>
Raportul de eroare al erorii cunoscute cu numărul	<code>sensible-browser "https://bugs.debian.org/numărul-erorii"</code>

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

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-10 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 rexpresie-regulată`” poate afișa mai multe informații prin „`aptitude -v why expresie-regulată`”. 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/R`”.

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 unele 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 care pachetele învechite
apt show foo	aptitude show foo	apt-cache show foo	afișează informații detaliate despre pachetul „foo”
apt search <i>expresie-regulată</i>	aptitude search <i>expresie-regulată</i>	apt-cache search <i>expresie-regulată</i>	căută pachetele care corespund cu <i>expresia-regulată</i>
N/D	aptitude why <i>expresie-regulată</i>	N/D	explică motivul pentru care ar trebui instalate pachetele care corespund cu <i>expresia-regulată</i>
N/D	aptitude why-not <i>expresie-regulată</i>	N/D	explică motivul pentru care pachetele care corespund cu <i>expresia-regulată</i> 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 deinstalările automate

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

2.2.3 Utilizarea interactivă a «aptitude»

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”.

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    libsmclient          -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
Vizualizare pachet	vedeți Tabel 2.10 (implicit)
Recomandări de auditare	lista pachetelor recomandate de unele pachete instalate, dar care nu sunt încă instalate
Lista plană a pachetelor	lista pachetelor fără categorizare (pentru utilizare cu expresii regulate)
Navigator debtags	lista pachetelor clasificate în funcție de intrările lor debtags
Vizualizare pachet sursă	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.

Indicație

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

categoria	descrierea vizualizării
Pachete actualizabile	lista pachetele organizate ca secțiune → zonă → pachet
Pachete noi	, ,
Pachete instalate	, ,
Pachete neinstalate	, ,
Pachete învechite sau create local	, ,
Pachete virtuale	lista pachetelor cu aceeași funcție
Sarcini	lista pachetelor cu diferite funcții necesare în general pentru o sarcină

Tabela 2.10: Clasificarea vizualizărilor standard ale pachetelor

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.

descrierea regulii extinse de potrivire	formula expresiei regulate
potrivire cu numele pachetului	<code>~nexpresie-regulată_nume</code>
potrivire cu descrierea	<code>~dexpresie-regulată_descriere</code>
potrivire cu numele sarcinii	<code>~texpresie-regulată_sarcină</code>
potrivire cu debtag	<code>~Gexpresie-regulată_debtag</code>
potrivire cu responsabilul pachetului	<code>~mexpresie-regulată_responsabil</code>
potrivire cu secțiunea pachetului	<code>~sexpresie-regulată_secțiune</code>
potrivire cu versiunea pachetului	<code>~Vexpresie-regulată_versiune</code>
potrivire cu distribuția	<code>~A{trixie,forky,sid}</code>
potrivire cu originea	<code>~O{debian,...}</code>
coincidentă prioritate	<code>~p{extra,important,optional,required,standard}</code>
corespunde cu pachete esențiale	<code>~E</code>
corespunde cu pachete virtuale	<code>~V</code>
corespunde cu pachete noi	<code>~N</code>
potrivire cu acțiunea în așteptare	<code>~a{install,upgrade,downgrade,remove,purge,hold,keep}</code>
potrivire cu pachete instalate	<code>~i</code>
corespunde pachetelor instalate cu marca A (pachete instalate automat)	<code>~M</code>
corespunde pachetelor instalate fără marca A (pachete selectate de administrator)	<code>~i!~M</code>
potrivire cu pachete instalate și actualizabile	<code>~U</code>
potrivire cu pachete eliminate, dar nu șterse	<code>~C</code>
potrivire cu pachete eliminate, șterse sau care pot fi eliminate	<code>~g</code>
potrivire cu pachete care declară o dependență ruptă	<code>~b</code>
potrivire cu pachete care declară o dependență ruptă de <i>tipul</i>	<code>~Btipul</code>
corespunde cu <i>modelul</i> de pachete care declară o dependență de <i>tipul</i>	<code>~D[tipul:]modelul</code>
corespunde cu <i>modelul</i> de pachete care declară o dependență ruptă de <i>tipul</i>	<code>~DB[tipul:]modelul</code>
corespunde cu pachetele pentru care <i>modelul</i> de potrivire corespunde pachetelor care declară o dependență de <i>tipul</i>	<code>~R[tipul:]modelul</code>
corespunde cu pachetele pentru care <i>modelul</i> de potrivire corespunde pachetelor care declară o dependență ruptă de <i>tipul</i>	<code>~RB[tipul:]modlul</code>
corespunde cu pachetele de care depind alte pachete instalate	<code>~R~i</code>
corespunde cu pachetele de care nu depind alte pachete instalate	<code>!~R~i</code>
corespunde cu pachetele de care depind sau pe care le recomandă alte pachete instalate	<code>~R~i ~Rrecommends:~i</code>
potrivire cu <i>modelul</i> de pachet filtrat după versiune	<code>~S filtru-versiune model</code>
corespunde cu toate pachetele (adevărat)	<code>~T</code>
nu corespunde cu niciun pachet (fals)	<code>~F</code>

Tabela 2.11: Lista formulelor de expresii regulate pentru aptitude

Iată câteva scurtături.

- „~Ptermen” == „~Dprovides:termen”
- „~Ctermen” == „~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 tildă „~”. Consultați „Manualul utilizatorului” pentru noua sintaxă de **formă lungă**.

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 <code>aptitude</code>

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 „~prequired|~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 duală 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 operații 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:

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

- 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.

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_expresie *» 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.

fișier	locție	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- <i>arhitectura</i>	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 <code>apt_preferences(5)</code>
Packages	partea superioară a fiecărei combinații distribuție/secțiune/arhitectură binară	<code>debian/control</code> concatenat pentru pachetele binare
Sources	partea superioară a fiecărei combinații distribuție/secțiune/sursă	<code>debian/control</code> concatenat pentru pachetele sursă

Tabela 2.14: Conținutul metadatelor arhivei Debian

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

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.debian.org/debian/dists/unstable/main/binary-armel/Release>” 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”.

2.5.8 Numele fișierelor pachetelor Debian

Fișierele pachetelor Debian au structuri de nume specifice.

tipul pachetului	structura numelui
Pachetul binar (cunoscut și sub numele de <code>deb</code>)	<code>nume-pachet_versiunea-upstream-revizia-debian_arhitectura</code>
Pachetul binar pentru <code>debian-installer</code> (cunoscut și sub numele de <code>udeb</code>)	<code>nume-pachet_versiunea-upstream-revizia-debian_arhitectura</code>
Pachetul sursă (sursă upstream)	<code>nume-pachet_versiunea-upstream-revizia-debian.orig.tar.gz</code>
Pachetul sursă 1.0 (modificări Debian)	<code>nume-pachet_versiunea-upstream-revizia-debian.diff.gz</code>
Pachetul sursă 3.0 (<code>quilt</code>) (modificări Debian)	<code>nume-pachet_versiunea-upstream-revizia-debian.debian.tar.gz</code>
Pachetul sursă (descrierea)	<code>nume-pachet_versiunea-upstream-revizia-debian.dsc</code>

Tabela 2.15: Structura numelor pachetelor Debian

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
<code>nume-pachet</code>	<code>[a-z0-9] [-a-z0-9.]+</code>	cerut
<code>epoca:</code>	<code>[0-9]+:</code>	opțional
<code>versiunea-upstream</code>	<code>[-a-zA-Z0-9.+:]+</code>	cerut
<code>revizia-debian</code>	<code>[a-zA-Z0-9.+~]+</code>	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 l18n și l10n (Cap. 8).

fișier	descrierea conținutului
/var/lib/dpkg/info/ <i>nume-pachet.conf</i>	fișierul tip de configurare. (modificabile de utilizator)
/var/lib/dpkg/info/ <i>nume-pachet.list</i>	lista fișierelor și directoarelor instalate de pachet
/var/lib/dpkg/info/ <i>nume-pachet.md5sums</i>	lista valorilor sumelor de control MD5 pentru fișierele instalate de pachet
/var/lib/dpkg/info/ <i>nume-pachet.preinst</i>	scriptul pachetului care trebuie rulat înainte de instalarea pachetului
/var/lib/dpkg/info/ <i>nume-pachet.postinst</i>	scriptul pachetului care trebuie rulat după instalarea pachetului
/var/lib/dpkg/info/ <i>nume-pachet.prerm</i>	scriptul pachetului care trebuie rulat înainte de eliminarea pachetului
/var/lib/dpkg/info/ <i>nume-pachet.postrm</i>	scriptul pachetului care trebuie rulat după eliminarea pachetului
/var/lib/dpkg/info/ <i>nume-pachet.conffile</i>	scriptul pachetului pentru sistemul debconf
/var/lib/dpkg/alternatives/ <i>nume-pachet</i>	informațiile alternative utilizate de comanda update-alternatives
/var/lib/dpkg/available	informațiile privind disponibilitatea tuturor pachetelor
/var/lib/dpkg/diversions	informațiile de redirectionare 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»

Suprascriderile 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

Since dpkg is very low level package tool, it can function without network connection.

Let's assume foo package was broken and needs to be fixed.

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.

You should install all dependency packages by repeatedly using “`sudo dpkg -i ...`” or by using:

```
# 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 încercă 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 - vechi*:

```
$ 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 încheie 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-upgrades/”.

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.

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 încurajează 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.

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**.

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*”.

- 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.

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.

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.

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.list” 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 ($\ll 100$), 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.



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 oboseitoare 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/ forkyn 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-adaptare „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)	dimensiune	descriere
approx	V:0, I:0	8308	server proxy de stocare în cache pentru fișierele arhive Debian (program compilat OCaml)
apt-cacher	V:0, I:0	267	proxy cache pentru pachetele Debian și fișierele sursă (program Perl)
apt-cacher-ng	V:4, I:4	1968	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/index.html`” 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	compatibilitate	încărcător de pornire	descriere
grub-efi-amd64	I:436	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:539	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, I:0	7323	Compatibil	GRUB 2	Acestea sunt imagini de recuperare care pot fi pornite GRUB 2 (CD și dischetă) (versiunea PC/BIOS)
grml-rescueboot	V:0, I:1	36	N/D	GRUB 2 plug-in	grml-rescueboot adds live Linux ISO images to the grub2 boot menu
syslinux	V:2, I:31	325	Compatibil	Isolinux	Acesta înțelege sistemul de fișiere ISO9660. Acesta este utilizat de CD-ul de pornire.
syslinux	V:2, I:31	325	Compatibil	Syslinux	Acesta înțelege sistemul de fișiere MSDOS (FAT) . Acesta este utilizat de discheta de pornire.
loadlin	V:0, I:0	87	Compatibil	Loadlin	Noul sistem este pornit din sistemul FreeDOS/MSDOS.
mbr	V:0, I:3	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`

On Debian system, `"/boot/grub/grub.cfg"` is managed by the installed GRUB package (e.g. `grub-efi-amd64`) and direct user modification to this file is deprecated. You should customize configuration files in `"/etc/grub.d/"` and `"/etc/default/grub"`, instead. Then configure the GRUB configuration files and update NVRAM variables to automatically boot into Debian by:

```
# dpkg-reconfigure grub-efi-amd64
```

Indicație

You can show kernel boot log messages by removing "quiet" from the "GRUB_CMDLINE_LINUX_DEFAULT" value in `"/etc/default/grub"`.

Indicație

You can add GRUB splash background by placing its image file in `"/boot/grub/"`.

See "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

The normal Debian system is the 4th stage of the boot process which is started by the mini-Debian system. The system kernel for the mini-Debian system continues to run in this environment. The root filesystem is switched from the one on the memory to the one on the physical storage device.

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)	descriere
systemd	V:903, I:978 10639	demon bazat pe evenimente <code>init(8)</code> pentru execuție simultană (alternativă la <code>sysvinit</code>)
cloud-init	V:3, I:6 3231	sistem de inițializare pentru instanțele cloud de infrastructură
systemd-sysv	V:892, I:979 94	paginile de manual și legăturile necesare pentru ca <code>systemd</code> să înlocuiască <code>sysvinit</code>
init-system-helpers	V:906, I:985 133	instrumente auxiliare pentru comutarea între <code>sysvinit</code> și <code>systemd</code>
initscripts	V:17, I:69 203	scripturi pentru inițializarea și oprirea sistemului
sysvinit-core	V:3, I:3 369	instrumente System-V precum <code>init(8)</code>
sysv-rc	V:35, I:73 91	mecanism de schimbare a nivelului de rulare de tip System-V
sysvinit-utils	V:715, I:999 106	instrumente de tip System-V (<code>startpar(8)</code> , <code>bootlogd(8)</code> , ...)
lsb-base	V:250, I:363 12	Linux Standard Base 3.2 Funcționalitatea scriptului <code>init</code>
insserv	V:41, I:73 132	instrument pentru organizarea secvenței de pornire utilizând dependențele scriptului <code>LSB init.d</code>
kexec-tools	V:1, I:5 320	instrumentul <code>kexec</code> pentru reporniri <code>kexec(8)</code> (repornire la cald)
systemd-bootchart	V:0, I:0 131	analizator de performanță al procesului de pornire
mingetty	V:0, I:2 36	numai consolă <code>getty(8)</code>
mgetty	V:0, I:0 315	modem inteligent <code>getty(8)</code> înlocuitor

Tabela 3.3: Lista instrumentelor de pornire pentru sistemul Debian

3.2 Rescue system



Avertisment

Do not perform system administration tasks around the boot strap process without having a rescue system.

Availability of a rescue system enables us to perform challenging tasks such as:

- Booting a system from a broken boot loader installation
- Fixing a broken boot loader installation
- Extracting data from a broken unbootable system
- Editing filesystems, disk partitions, and LVM volumes involving the root filesystem

Typically, a rescue system is provided as a ISO image file and written to a removable storage media such as:

- [USB flash drive](#) prepared as Secțiune [9.7.2](#)
- [CD / DVD](#) prepared as Secțiune [9.7.7](#)

For simplicity, USB flash drive cases are mentioned as examples below but CD or DVD may be used as well.

Indicație

You may need to change some UEFI NVRAM variables to boot arbitrary boot loaders on the removable storage media.

3.2.1 GRUB UEFI rescue system on USB

The GRUB UEFI rescue system with menu can be started by turning on system power with the "GRUB UEFI rescue system on USB" inserted.

This "GRUB UEFI rescue system on USB" is prepared by writing the [Super Grub2 Disk](#) ISO image to [USB flash drive](#) in advance.

In case of broken boot loader configuration by the installation of another operating system etc., you can fix this by:

- Start the GRUB UEFI rescue system to discover bootable installed systems automatically.
- Start the installed Debian system from the GRUB menu.
- At Linux root shell prompt:

```
# dpkg-reconfigure grub-efi-amd64
```

Notă

The bootable GRUB rescue ISO image can be generated by following "info grub-mkrescue", too. It offers a CLI GRUB shell prompt but doesn't offer automatic discovery of bootable installed systems.

3.2.2 Linux live rescue system on USB

The Linux live rescue system can be started by turning on system power with the "Linux live rescue system on USB" inserted.

This "Linux live rescue system on USB" can be prepared by writing one of Linux live ISO images based on Debian to a [USB flash drive](#) in advance. Here are some examples of such Linux live images.

- [Debian Live images](#)
- [Kali Linux Live](#)
- [Grml Live Linux](#)

Here are some use cases of this Linux live rescue system:

- Fix the broken boot loader configuration caused by the installation of another operating system etc.:
 - Start the Linux live rescue system.
 - Mount the partition containing the root filesystem of the unbootable installed Debian system to "/mnt".
 - At Linux root shell prompt:

```
# chroot /mnt dpkg-reconfigure grub-efi-amd64
```

- Fix the broken dpkg package:
 - Start the Linux live rescue system.
 - Mount the partition containing the root filesystem of the installed Debian system with the broken dpkg package to "/mnt".
 - At Linux root shell prompt:

```
# dpkg --root /mnt -i /mnt/var/cache/apt/archives/dpkg_old_version-amd64.deb
```

- Perform normally prohibited changes such as filesystem operations to the installed system (see Secțiune [9.6](#)).
-

**Avertisment**

Your GUI screen of the Linux live system may be locked after the inactivity.

Indicație

- It's a good idea to set [your password as soon as you start a Linux live system](#).
- You may set your password for example by "sudo passwd user" from the user's shell prompt on the Linux virtual consoles (see Secțiune [1.1.6](#)).
- Some Linux live systems may set their default "user/password": "Debian Live" = "user/live", "Kali Linux Live" = "kali/kali"

3.2.3 Linux live rescue system from GRUB

The Linux live rescue system can be started from the GRUB menu entry. The GRUB configuration for this is prepared by the following:

- Install the `grml-rescueboot` package
- Find a Linux live ISO image which has `/boot/grub/loopback.cfg` in its image.
 - ISO images mentioned in Secțiune [3.2.2](#) have `/boot/grub/loopback.cfg` in their image.
- Copy some of these Linux live ISO images to the `/boot/grml/` directory.
- Update GRUB menu by:

```
# dpkg-reconfigure grub-efi-amd64
```

Upon turning on system power, the GRUB displays menu with candidates for Linux live rescue systems.

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[0-6]`” 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
```

Operația	Fragmente de comandă
Afișarea jurnalului pentru serviciile de sistem și nucleu de la ultima pornire	«journalctl -b --system»
Afișarea jurnalului pentru serviciile utilizatorului curent de la ultima pornire	«journalctl -b --user»
Afișarea jurnalului de activitate al „\$unit” de la ultima pornire	«journalctl -b -u \$unit»
Afișarea jurnalului de activități al „\$unit” (în stilul „tail -f”) de la ultima pornire	«journalctl -b -u \$unit -f»

Tabela 3.5: Lista fragmentelor tipice de comandă `journalctl`

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.

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 (sufixe 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](#).

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.

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»
Verifică 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 variabilă=valoare»
Elimină definiția (valoarea) „variabilei” din mediul de lucru	«systemctl unset-environment variabilă»
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

3.8.3 Inițializarea interfeței de rețea

Interfețele de rețea sunt inițializate de obicei din „networking.service” pentru interfața `lo` și din „NetworkManager.service” 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
<code>/etc/passwd</code>	<code>-rw-r--r--</code>	<code>root</code>	<code>root</code>	informații despre contul utilizatorului (curățate)
<code>/etc/shadow</code>	<code>-rw-r-----</code>	<code>root</code>	<code>shadow</code>	informații securizate despre contul utilizatorului
<code>/etc/group</code>	<code>-rw-r--r--</code>	<code>root</code>	<code>root</code>	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:$1$VGZLVbS$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.

comanda	funcție
getent passwd <i>nume-utilizator</i>	răsfoiește informațiile contului „ <i>nume-utilizator</i> ”
getent shadow <i>nume-utilizator</i>	răsfoiește informațiile contului ascuns al „ <i>nume-utilizator</i> ”
getent group <i>nume-grup</i>	răsfoiește informațiile despre grupul „ <i>nume-grup</i> ”
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	comanda	funcție
whois	V:22, I:209	384	mkpasswd	interfață cu funcții excesive pentru biblioteca crypt(3)
openssl	V:834, I:995	2503	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)	descriere
libpam-modules	V:922, I:999 917	Module de autentificare conectabile (serviciul de bază)
libpam-ldapd	V:6, I:14 80	Modul de autentificare conectabil care permite interfețe LDAP
libpam-systemd	V:701, I:963 739	Modul de autentificare conectabil pentru înregistrarea sesiunilor utilizatorilor pentru logind
libpam-doc	I:6 1504	Module de autentificare conectabile (documentație în format html și text)
libc6	V:922, I:999 5370	Biblioteca GNU C: Biblioteci partajate care oferă de asemenea serviciul „Name Service Switch”
glibc-doc	I:5 3858	Biblioteca GNU C: Pagini de manual
glibc-doc-reference	I:3 14261	Biblioteca GNU C: Manual de referință în format info, pdf și html (non-free)
libnss-mdns	V:235, I:520 141	Modul NSS pentru rezolvarea numelor DNS multicast
libnss-ldapd	V:7, I:17 131	Modul NSS pentru utilizarea LDAP ca serviciu de denumire

Tabela 4.5: Lista pachetelor principale ale sistemelor PAM și 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
<code>/etc/pam.d/nume-program</code>	definește configurația PAM pentru programul „ <i>nume-program</i> ”; vedeți pam(7) și pam.d(5)
<code>/etc/nsswitch.conf</code>	definește configurația NSS cu intrarea pentru fiecare serviciu.; a se vedea nsswitch.conf(5)
<code>/etc/nologin</code>	limitează autentificarea utilizatorului prin modulul pam_nologin(8)
<code>/etc/securetty</code>	limitează tty pentru accesul root prin modulul pam_securetty(8)
<code>/etc/security/access.conf</code>	stabilește limita de acces prin modulul pam_access(8)
<code>/etc/security/group.conf</code>	definește restricția bazată pe grup prin modulul pam_group(8)
<code>/etc/security/pam_env.conf</code>	definește variabilele de mediu prin modulul pam_env(8)
<code>/etc/environment</code>	definește variabile de mediu suplimentare prin modulul pam_env(8) cu argumentul „ <code>readenv=1</code> ”
<code>/etc/default/locale</code>	definește configurația de limbă prin modulul pam_env(8) cu argumentul „ <code>readenv=1 envfile=/etc/default/locale</code> ” (Debian)
<code>/etc/security/limits.conf</code>	stabilește restricții de resurse (ulimit, core, ...) prin modulul pam_limits(8)
<code>/etc/security/time.conf</code>	stabilește restricția de timp prin modulul pam_time(8)
<code>/etc/systemd/logind.conf</code>	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-ldap` 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 (poștă electronică)	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.)

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 redirectionare 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 Măsurile 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	popularity	descriere
knockd	V:0, I:1	110	mic demon port-knock <code>knockd(1)</code> și client <code>knock(1)</code>
fail2ban	V:95, I:106	2191	interzice adresele IP care provoacă erori multiple de autentificare
libpam-shield	V:0, I:0	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.

- Prevent physical access to the system storage device ([HDD](#) / [SSD](#) / ...)
- Blocați `UEFI/BIOS` și împiedicați pornirea de pe suportul amovibil
- Definiți o parolă pentru sesiunea interactivă `GRUB`
- Blocați meniul `GRUB` împotriva editării

With physical access to the system storage device, resetting the password is relatively easy with following steps.

1. Move the system storage device to a PC with USB bootable `UEFI/BIOS`.
2. Boot system with a rescue media (see 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 9.5.11). 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 operații 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 [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 spațiului 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 rețelor. 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)	limita	tipul	descriere
network-manager	V:421, I:484	7805	config::NM	NetworkManager (daemon): gestionează rețeaua în mod automat
network-manager-gnome	V:53, I:197	18	config::NM	NetworkManager (interfață GNOME)
netplan.io	V:1, I:7	340	config::NM+networkd	Netplan (generator): interfață unificată și activă pentru NetworkManager și modulele systemd-networkd
ifupdown	V:619, I:973	201	config::ifupdown	instrument standardizat pentru activarea și dezactivarea rețelei (specific Debian)
isc-dhcp-client	V:169, I:707	2884	config::low-level	client DHCP
pppoeconf	V:0, I:4	174	config::helper	asistent de configurare pentru conexiunea PPPoE
wpa_supplicant	V:399, I:529	3901	config::helper	client ce oferă suport pentru WPA și WPA2 (IEEE 802.11i)
wpa_gui	V:0, I:1	784	config::helper	client cu interfață grafică Qt pentru wpa_supplicant
wireless-tools	V:192, I:264	292	config::helper	instrumente pentru manipularea extensiilor „fără fir” Linux (Linux Wireless Extensions)
iw	V:38, I:490	332	config::helper	instrument pentru configurarea dispozitivelor wireless Linux
iproute2	V:756, I:984	4122	config::iproute2	iproute2 , IPv6 și alte configurații avansate de rețea: ip(8), tc(8) etc.
iptables	V:353, I:629	2410	config::Netfilter	instrumente de administrare pentru filtrarea pachetelor și NAT (Netfilter)
nftables	V:212, I:850	191	config::Netfilter	instrumente de administrare pentru filtrarea pachetelor și NAT (Netfilter) (succesor al {ip,ip6,arp,eb}tables)
iputils-ping	V:195, I:997	188	test	testează accesibilitatea rețelei unei gazde aflată la distanță prin nume gazdă sau adresă IP (iproute2)
iputils-arping	V:1, I:19	53	test	testează accesibilitatea rețelei unei gazde de la distanță specificată de adresa ARP
iputils-tracert	V:2, I:21	50	test	trasează ruta de rețea către o gazdă la distanță
ethtool	V:93, I:250	1077	test	afișează sau modifică configurarea dispozitivului Ethernet
mtr-tiny	V:4, I:39	181	test::low-level	urmărește ruta de rețea către o gazdă la distanță (curses)
mtr	V:4, I:40	230	test::low-level	urmărește ruta de rețea către o gazdă la distanță (curses și GTK)
gnome-nettool	V:0, I:10	2480	test::low-level	instrumente pentru operații comune de informații de rețea (GNOME)
nmap	V:25, I:185	4607	test::low-level	cartograf de rețea / scanner de porturi (Nmap , consolă)
tcpdump	V:16, I:166	1343	test::low-level	analizator de trafic de rețea (Tcpdump , consolă)
wireshark	V:2, I:41	11267	test::low-level	analizator de trafic de rețea (Wireshark , GTK)
tshark	V:2, I:23	438	test::low-level	analizator de trafic de rețea (consolă)
tcptrace	V:0, I:1	407	test::low-level	produce o sinteză a conexiunilor din ieșirea tcpdump
ntopng	V:0, I:0	15604	test::low-level	afișează utilizarea rețelei în navigatorul web
dnsutils	I:174	23	test::low-level	clienți de rețea furnizați cu BIND : nslookup(8), nsupdate(8), dig(8)
dlint	V:0, I:2	51	test::low-level	verifică informațiile zonei DNS utilizând căutări în serverul de nume
dnstracer	V:0, I:1	59	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”.

Clasă	adrese de rețea	masca de rețea	masca de rețea /biți	numărul de subrețele
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.service” 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.Debian”.

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 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.

comanda	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”
tracert 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
netstat -a	găsește toate porturile deschise
netstat -l --inet	găsește porturile de ascultare
netstat -ln --tcp	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	popularity	descriere
iftop	V:6, I:88	93	afișează informații despre utilizarea lățimii de bandă pe o interfață de rețea
iperf	V:2, I:35	427	instrument de măsurare a lățimii de bandă a protocolului Internet
ifstat	V:0, I:5	52	monitorizarea statisticilor interfeței
bmon	V:1, I:20	141	monitor portabil de lățime de bandă și estimator de viteză
ethstatus	V:0, I:2	41	script care măsoară rapid rata de transfer a dispozitivelor de rețea
bing	V:0, I:0	80	tester empiric stocastic al lățimii de bandă
bwm-ng	V:1, I:10	95	monitor de lățime de bandă mic și simplu, bazat pe consolă
ethstats	V:0, I:0	21	monitor de statistici Ethernet bazat pe consolă
ipfm	V:0, I:0	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 `tracert`(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 „TCP MSS” 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](#)).

Programul principal al spațiului de utilizator al [netfilter](#) este `iptables`(8). Puteți configura manual [netfilter](#) în mod interactiv din shell, salva starea acestuia cu `iptables -save`(8) și s-o restaurați prin scriptul `init` cu `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/`”).

pachete	popcon	(populăritate)	descriere
nftables	V:212, I:850	191	instrumente de administrare pentru filtrarea pachetelor și NAT (Netfilter) (succesor al {ip,ip6,arp,eb}tables)
iptables	V:353, I:629	2410	instrumente de administrare pentru netfilter (iptables (8) pentru IPv4, ip6tables (8) pentru IPv6)
arptables	V:0, I:1	102	instrumente de administrare pentru netfilter (arptables (8) pentru ARP)
ebtables	V:14, I:24	276	instrumente de administrare pentru netfilter (ebtables (8) pentru punte Ethernet)
iptstate	V:0, I:1	122	monitorizează continuu starea netfilter (similar cu top (1))
ufw	V:73, I:99	859	Uncomplicated Firewall (UFW) -- (paravan de protecție fără complicații) este un program pentru gestionarea unui paravan de protecție netfilter
gufw	V:5, I:10	3663	interfață grafică cu utilizatorul pentru Uncomplicated Firewall (UFW)
firewalld	V:16, I:24	2482	firewalld este un program de paravan de protecție gestionat dinamic, cu suport pentru zone de rețea
firewall-config	V:0, I:3	1076	interfață grafică cu utilizatorul pentru firewalld
shorewall-init	V:0, I:0	88	inițializarea Shoreline Firewall (paravan de protecție al liniei de coastă)
shorewall	V:2, I:5	3090	Shoreline Firewall , generator de fișiere de configurare netfilter
shorewall-lite	V:0, I:0	71	Shoreline Firewall , generator de fișiere de configurare netfilter (versiune simplificată)
shorewall6	V:0, I:1	1334	Shoreline Firewall , generator de fișiere de configurare netfilter (versiune IPv6)
shorewall6-lite	V:0, I:0	71	Shoreline Firewall , generator de fișiere de configurare netfilter (versiune simplificată, IPv6)

Tabela 5.7: Lista instrumentelor de paravan de protecție

- [Linux Networking-concepts HOWTO](#) -- Rețetar pentru conceptele de rețea Linux
- [Linux 2.4 Packet Filtering HOWTO](#) -- Rețetar pentru filtrarea pachetelor Linux 2.4
- [Linux 2.4 NAT HOWTO](#) -- Rețetar pentru NAT în Linux 2.4

Indicație

Deși acestea au fost scrise pentru Linux **2.4**, atât comanda `iptables(8)` cât și funcția `netfilter` din nucleu se aplică pentru Linux **2.6** și seria de nuclee **3.x**.

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 situri 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:30, I:103	286985	X	Chromium , (navigator cu codul-sursă deschis de la Google)
firefox	V:15, I:21	277820	, ,	Firefox , (navigator cu codul-sursă deschis de la Mozilla, disponibil numai în Debian Unstable)
firefox-esr	V:214, I:442	266490	, ,	Firefox ESR , (versiune Firefox cu suport extins)
epiphany-browser	V:2, I:11	2258	, ,	GNOME , conform cu liniile directoare privind interfața umană Human Interface Guidelines: HIG , Epiphany
konqueror	V:27, I:116	7861	, ,	KDE , Konqueror
dillo	V:0, I:4	1585	, ,	Dillo , (navigator de dimensiuni reduse, bazat pe FLTK)
w3m	V:11, I:145	2853	text	w3m
lynx	V:29, I:457	1972	, ,	Lynx
elinks	V:2, I:16	1791	, ,	ELinks
links	V:2, I:21	2321	, ,	Links (text only)
links2	V:0, I:10	5466	grafică	Linkuri (grafică de consolă fără X)

Tabela 6.1: Lista navigatoarelor web

**Atenție**

Șirul falsificat al agentului utilizator poate provoca [efecte secundare negative cu Java](#).

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)	nr. de utilizatori	tipul
evolution	V:28, I:240	492	X GUI program (GNOME, groupware suite)
thunderbird	V:46, I:109	274692	program cu interfață grafică X (GTK, Mozilla Thunderbird)
kmail	V:43, I:107	25212	program cu interfață grafică X (KDE)
mutt	V:11, I:94	7118	program terminal de caractere probabil utilizat cu vim
mew	V:0, I:0	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 instalează în mod tradițional un program MTA pentru a oferi suport programelor care așteptau comanda `/usr/sbin/sendmail`. 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
```

pachet	popcon	(populație)	descriere
exim4-daemon-light	V:219, I:226	1649	agentul de transport de poștă Exim4 (MTA: implicit în Debian)
exim4-daemon-heavy	V:5, I:5	1814	agentul de transport de poștă Exim4 (MTA: alternativă flexibilă)
exim4-base	V:225, I:231	1646	documentația Exim4 (text) și fișiere comune
exim4-doc-html	I:1	3798	documentația Exim4 (html)
exim4-doc-info	I:0	648	documentația Exim4 (info)
postfix	V:106, I:112	4003	agentul de transport de poștă Postfix (MTA: alternativă securizată)
postfix-doc	I:4	4841	documentația Postfix (html+text)
sasl2-bin	V:4, I:10	368	implementarea API-ului SASL Cyrus (supliment postfix pentru SMTP AUTH)
cyrus-sasl2-doc	I:0	2142	Cyrus SASL - documentație
msmtp	V:7, I:13	811	MTA cu o dimensiune foarte mică
msmtp-mta	V:5, I:7	136	MTA cu o dimensiune foarte mică (extensie de compatibilitate cu sendmail pentru msmtp)
esmtplib	V:0, I:0	123	MTA cu o dimensiune foarte mică
esmtplib-run	V:0, I:0	27	MTA cu o dimensiune foarte mică (extensie de compatibilitate cu sendmail pentru esmtplib)
nullmailer	V:7, I:8	483	dezactivează MTA, fără poștă locală
ssmtp	V:4, I:6	133	dezactivează MTA, fără poștă locală
sendmail-bin	V:10, I:11	1959	MTA cu funcții complete (numai dacă sunteți deja familiarizat cu acesta)
courier-mta	V:0, I:0	2674	MTA cu funcții complete (interfață web etc.)
git-email	V:0, I:10	1187	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 „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 redirectionarea mesajelor către:”.

Definiți „Adresa IP sau numele gazdei smarthost-ului de ieșire:” la „smtp.hostname.dom:587”.

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/exim4/exim4.conf” (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.

comanda	funcție
<code>postfix(1)</code>	Controlul programului postfix
<code>postconf(1)</code>	Instrumentul de configurare postfix
<code>postconf(5)</code>	Parametrii de configurare postfix
<code>postmap(1)</code>	Administrarea tabelului de căutare postfix
<code>postalias(1)</code>	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 hash:/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
<code>sendmail</code>	<code>sendmail</code>	citește mesajele din intrarea standard și organizează livrarea (-bm)
<code>mailq</code>	<code>mailq</code>	listează coada de mesaje cu starea și ID-ul cozii (-bp)
<code>newaliases</code>	<code>newaliases</code>	inițializează baza de date alias (-I)
<code>exim4 -q</code>	<code>postqueue -f</code>	elimină mesajele în așteptare (-q)
<code>exim4 -qf</code>	<code>postsuper -r ALL deferred; postqueue -f</code>	elimină toate mesajele
<code>exim4 -qff</code>	<code>postsuper -r ALL; postqueue -f</code>	elimină chiar și mesajele înghețate
<code>exim4 -Mg queue_id</code>	<code>postsuper -h queue_id</code>	îngheață un mesaj după ID-ul cozii sale
<code>exim4 -Mrm queue_id</code>	<code>postsuper -d queue_id</code>	elimină un mesaj după ID-ul cozii sale
N/D	<code>postsuper -d ALL</code>	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-J).

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).

pachet	popcon(popularitate)	dimensiune	instrument	descriere
openssh-client	V:898, I:996	5133	ssh(1)	client de shell securizat
openssh-server	V:745, I:804	3502	sshd(8)	server de shell securizat
ssh-askpass	I:17	103	ssh-askpass(1)	solicită utilizatorului o frază de acces pentru ssh-add (X simplu)
ssh-askpass-gnome	V:0, I:3	215	ssh-askpass-gnome(1)	solicită utilizatorului o frază de acces pentru ssh-add (GNOME)
ssh-askpass-fullscreen	V:0, I:0	41	ssh-askpass-fullscreen(1)	solicită utilizatorului o frază de acces pentru ssh-add (GNOME) cu un plus de atracție vizuală
shellinabox	V:0, I:1	525	shellinaboxd(1)	server web pentru emulator terminal VT100 accesibil din navigator

Tabela 6.7: Lista serverelor și instrumentelor de acces la distanță

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](#)).

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_nume-tip-cheie	cheie secretă SSH-2 <i>key-type-name</i> precum ecdsa , ed25519 , ... a utilizatorului

Tabela 6.8: Lista fișierelor de configurare SSH

Următorul cod inițializează o conexiune [ssh\(1\)](#) de la un client.

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
```

comanda	descriere
ssh nume-utilizator@nume-gazdă.domeniu.ext	se conectează cu modul implicit
ssh -v nume-utilizator@nume-gazdă.domeniu.ext	se conectează cu modul implicit cu mesaje de depanare
ssh -o PreferredAuthentications=parolă nume-utilizator@nume-gazdă.domeniu.ext	forțează utilizarea parolei cu SSH versiunea 2
ssh -t nume-utilizator@nume-gazdă.domeniu.ext parolă	rulează programul passwd pentru a actualiza parola pe o gazdă la distanță

Tabela 6.9: Listă de exemple de pornire a clientului SSH

6.3.3 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 „`AllowTcpForwarding`” 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 \(PDF\)](#).

The CUPS uses [Internet Printing Protocol \(IPP\)](#). The IPP is the cross-platform de facto standard for remote printing with bi-directional communication capability.

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(popularitete)	dimensiune	tip	descriere
lpr	V:2, I:2	378	imprimantă (515)	BSD <code>lpr/lpd</code> (Demonul de imprimare în linie)
cups	V:118, I:460	1092	IPP (631)	Serverul CUPS pentru imprimare prin Internet
cups-client	V:137, I:474	433	, ,	Comenzi imprimare System V pentru CUPS: <code>lp(1)</code> , <code>lpstat(1)</code> , <code>lpoptions(1)</code> , <code>cancel(1)</code> , <code>lpmove(8)</code> , <code>lpinfo(8)</code> , <code>lpadmin(8)</code> , ...
cups-bsd	V:36, I:194	131	, ,	Comenzi BSD pentru imprimare pentru CUPS: <code>lpr(1)</code> , <code>lpq(1)</code> , <code>lprm(1)</code> , <code>lpc(8)</code>
printer-driver-gutenprint	V:13, I:61	1122	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.

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 `isc-dhcp-server`, așa cum este descris în [pagina DDNS din wiki-ul Debian](#).

pachet	popcon(popularitate)	linie de port	protocol	descriere
telnetd	V:0, I:1	51	TELNET	server TELNET
nfs-kernel-server	V:45, I:54	797	NFS	partajarea fișierelor Unix
samba	V:106, I:121	5011	SMB	partajarea fișierelor și imprimantelor Windows
netatalk	V:0, I:1	814	ATP	partajarea fișierelor și imprimantelor Apple/Mac (AppleTalk)
proftpd-basic	V:4, I:10	452	FTP	descărcare fișier general
apache2	V:186, I:227	586	HTTP	server web general
squid	V:9, I:10	9349	, ,	server proxy web general
bind9	V:35, I:39	888	DNS	adresa IP pentru alte gazde
isc-dhcp-server	V:15, I:25	6102	DHCP	adresa IP a clientului însuși

Tabela 6.12: Lista altor servere de aplicații de rețea

Indicație

Utilizarea unui server proxy precum [squid](#) este mult mai eficientă pentru economisirea lății 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:46, I:904	139	TCP/IP	cuțitul elvețian TCP/IP
netcat-openbsd	V:21, I:121	109	TCP/IP	cuțitul elvețian TCP/IP cu suport pentru IPv6, proxy-uri și socluri Unix
openssl	V:834, I:995	2503	SSL	binare Secure Socket Layer (SSL) și instrumente criptografice asociate
stunnel4	V:6, I:9	573	, ,	învăluitor SSL universal
telnet	V:12, I:239	51	TELNET	client TELNET
nfs-common	V:145, I:199	1137	NFS	partajarea fișierelor Unix
smbclient	V:27, I:209	2106	SMB	client de partajare a fișierelor și imprimantelor MS Windows
cifs-utils	V:31, I:118	351	, ,	comenzile de montare și demontare pentru fișiere MS Windows la distanță
wget	V:191, I:982	3784	HTTP și FTP	program de descărcare din rețea
curl	V:227, I:684	507	, ,	, ,
transmission-gtk	V:13, I:178	6245	BitTorrent	BitTorrent client (GTK)
transmission-qt	V:0, I:2	6203	, ,	BitTorrent client (Qt)
ktorrent	V:1, I:5	5167	, ,	BitTorrent client (Qt)
qbittorrent	V:9, I:22	14384	, ,	BitTorrent client (Qt)
bind9-host	V:121, I:940	140	DNS	host(1) din bind9, "Priority: standard"
dnsutils	I:174	23	, ,	dig(1) din bind, "Priority: standard"
isc-dhcp-client	V:169, I:707	2884	DHCP	obține adresa IP
ldap-utils	V:10, I:58	789	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	(popularity)	descriere
task-gnome-desktop	1:200	9	mediul grafic de birou GNOME
task-xfce-desktop	1:91	9	mediul grafic de birou Xfce
task-kde-desktop	1:96	6	mediul grafic de birou KDE Plasma
task-mate-desktop	1:35	9	mediul grafic de birou MATE
task-cinnamon-desktop	1:39	9	mediul grafic de birou Cinnamon
task-lxde-desktop	1:22	9	mediul grafic de birou LXDE
task-lxqt-desktop	1:17	9	mediul grafic de birou LXQt
task-gnome-flashback-desktop	1:11	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 modeb” 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:0, I:28	222	Administratorul de ferestre mutter al GNOME [auto]
xwayland	V:255, I:346	2541	Un server X care rulează pe wayland [auto]
gnome-remote-desktop	V:122, I:250	2215	Demon de mediu de birou la distanță pentru GNOME folosind PipeWire [auto]
gnome-tweaks	V:19, I:242	1145	Opțiuni avansate de configurare pentru GNOME
gnome-shell-extension-prefs	V:9, I:145	83	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:28, I:240	492	GNOME	gestionarea informațiilor personale (software de lucru în grup și poșta electronică)
thunderbird	V:46, I:109	274692	GTK	client de poșta electronică (Mozilla Thunderbird)
kontakt	V:1, I:11	2298	KDE	gestionarea informațiilor personale (software de lucru în grup și poșta electronică)
libreoffice-writer	V:120, I:441	33269	LO	procesor de text
abiword	V:1, I:5	3601	GNOME	procesor de text
calligrawords	V:0, I:3	6937	KDE	procesor de text
scribus	V:1, I:13	32052	KDE	editor de tehnoredactare computerizată pentru editarea fișierelor PDF
glabels	V:0, I:2	1283	GNOME	editor de etichete
libreoffice-calc	V:116, I:438	28287	LO	foaie de calcul
gnumeric	V:3, I:11	9958	GNOME	foaie de calcul
calligrasheets	V:0, I:2	13593	KDE	foaie de calcul
libreoffice-impress	V:97, I:436	2459	LO	prezentare
calligrastage	V:0, I:2	6017	KDE	prezentare
libreoffice-base	V:24, I:77	5004	LO	gestionare a bazei de date
kexi	V:0, I:0	7565	KDE	gestionare a bazei de date
libreoffice-draw	V:98, I:437	11003	LO	editor de grafică vectorială (desen)
inkscape	V:13, I:85	113183	GNOME	editor de grafică vectorială (desen)
karbon	V:0, I:3	3962	KDE	editor de grafică vectorială (desen)
dia	V:1, I:18	3812	GTK	editor de diagrame și organigrame
gimp	V:35, I:229	31748	GTK	editor grafic (pictură)
shotwell	V:15, I:258	6334	GTK	organizator de fotografii digitale
digikam	V:1, I:9	302	KDE	organizator de fotografii digitale
darktable	V:3, I:12	35895	GTK	masă de lumină și cameră obscură pentru fotografii
planner	V:0, I:4	1400	GNOME	gestionarea proiectelor
calligraplan	V:0, I:3	23545	KDE	gestionarea proiectelor
gncash	V:2, I:7	29395	GNOME	contabilitate personală
homebank	V:0, I:1	3194	GTK	contabilitate personală
lilypond	V:0, I:6	16924	-	compozitor de partituri muzicale
kmy money	V:0, I:2	18826	KDE	contabilitate personală
librecad	V:1, I:14	9100	aplicație Qt	sistem de proiectare asistată de calculator (CAD) (2D)
freecad	V:0, I:20	110	aplicație Qt	sistem de proiectare asistată de calculator (CAD) (3D)
kicad	V:3, I:15	163907	GTK	software pentru proiectarea schemelor electronice și a plăcilor cu circuite imprimate
xsane	V:10, I:135	1512	GTK	interfață scanner
libreoffice-math	V:90, I:439	1928	LO	editor de ecuații/formule matematice
calibre	V:8, I:26	65584	KDE	convertor de cărți electronice și gestionarea bibliotecii
fbreader	V:0, I:6	3783	GTK	cititor de cărți electronice
evince	V:80, I:302	952	GNOME	program de vizualizare a documentelor(pdf)
okular	V:44, I:135	4415	KDE	program de vizualizare a documentelor(pdf)
x11-apps	V:33, I:464	2461	aplicație X pură	xeyes(1), etc.
x11-utils	V:227, I:568	651	aplicație X pură	xev(1), xwininfo(1), etc.

Tabela 7.3: Lista aplicațiilor cu interfață grafică notabile

pachet	popcon(popularitate)	con(limita)	fonturi	serif	mono	informații despre font
fonts-cantarell	V:182, I:305	227	59	-	-	Cantarell (GNOME 3, afișare)
fonts-noto	I:157	31	61	63	40	Fonturi Noto (Google, multi-lingvistice cu CJK)
fonts-dejavu	I:405	35	58	68	40	DejaVu (GNOME 2, MCM: Verdana , extins Bitstream Vera)
fonts-liberation2	V:63, I:218	15	56	60	40	Fonturi Liberation pentru LibreOffice (Red Hat, MCMATC)
fonts-croscore	V:21, I:38	5274	56	60	40	Chrome OS: Arimo , Tinos și Cousine (Google, MCMATC)
fonts-crosextra-carlito	V:20, I:98	2696	57	-	-	Chrome OS: Carlito (Google, MCM: Calibri)
fonts-crosextra-caladea	V:11, I:93	347	-	55	-	Chrome OS: Caladea (Google, MCM: Cambria) (doar Latin)
fonts-freefont-ttf	V:82, I:207	14460	57	59	40	GNU FreeFont (URW Nimbus extins)
fonts-quicksand	V:209, I:465	392	56	-	-	Debian task-desktop, Quicksand (afișare, doar Latin)
fonts-hack	V:33, I:140	2507	-	-	40 P	Un tip de font conceput pentru codul sursă Hack (Facebook)
fonts-sil-gentiumplus	I:30	14345	-	54	-	Gentium SIL
fonts-sil-charis	I:29	6704	-	59	-	Charis SIL
fonts-urw-base35	V:194, I:541	15560	56	60	40	URW Nimbus (Nimbus Sans , Roman No. 9 L , Mono L , MCAHTC)
fonts-ubuntu	V:2, I:5	4339	58	-	33 P	Fonturi Ubuntu (afișare)
fonts-terminus	V:0, I:4	452	-	-	33	Fonturi retro faine pentru terminale
ttf-mscorefonts-installer	V:0, I:42	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 aptitudinea 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)	descriere
libfreetype6	V:580, I:997	1021 FreeType bibliotecă de conversie și redare a fonturilor
libfontconfig1	V:569, I:827	344 Fontconfig bibliotecă de configurare a fonturilor
fontconfig	V:462, I:706	415 <code>fc - *</code> : comenzi în linia de comandă pentru Fontconfig
font-manager	V:2, I:7	1118 Font Manager : comenzi din interfață grafică pentru Fontconfig
nautilus-font-manager	V:0, I:0	40 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 `font s.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	popularity	descriere
flatpak	V:101, I:107	8280	Flatpak cadru de implementare a aplicațiilor pentru aplicații de birou
gnome-software-plugin-flatpak	V:29, I:41	285	Suport Flatpak pentru software-ul GNOME
snapd	V:65, I:69	72012	Demon și instrumente care abilitază pachetele snap
gnome-software-plugin-snap	V:1, I:2	148	Suport Snap pentru software-ul GNOME
xdg-desktop-portal	V:364, I:449	2166	portal de integrare a mediului grafic de birou pentru Flatpak și Snap
xdg-desktop-portal-gtk	V:332, I:447	715	serviciul xdg-desktop-portal pentru gtk (GNOME)
xdg-desktop-portal-kde	V:79, I:112	2688	serviciul xdg-desktop-portal pentru Qt (KDE)
xdg-desktop-portal-wlr	V:1, I:6	160	serviciul xdg-desktop-portal pentru wlroots (Wayland)
firejail	V:1, I:4	1881	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	protocol	descriere
gnome-remote-desktop	V:122, I:250	2215	RDP	serverul GNOME Remote Desktop
xrdp	V:24, I:28	4502	RDP	xrdp, serverul Remote Desktop Protocol (RDP)
x11vnc	V:8, I:42	1863	RFB (VNC)	x11vnc, serverul Remote Framebuffer Protocol (VNC)
tigervnc-standalone-server	V:5, I:14	2967	RFB (VNC)	TigerVNC, serverul Remote Framebuffer Protocol (VNC)
gnome-connections	V:6, I:125	1599	RDP, RFB (VNC)	client GNOME remote desktop
vinagre	V:1, I:28	4249	RDP, RFB (VNC), SPICE, SSH	Vinagre: client GNOME remote desktop
remmina	V:15, I:64	971	RDP, RFB (VNC), SPICE, SSH, ...	Remmina: client de mediu grafic de birou la distanță GTK
krdc	V:1, I:16	4113	RDP, RFB (VNC)	KRDC: client de Birou la distanță KDE
virt-viewer	V:5, I:44	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	comanda	descriere
openssh-server	V:745, I:804	3502	sshd cu opțiunea X11-forwarding	server SSH (securizat)
openssh-client	V:898, I:996	5133	ssh -X	client SSH (securizat)
xauth	V:186, I:971	81	xauth	utilitarul de fișier de autoritate X
x11-xserver-utils	V:312, I:542	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 redirectionare 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:7, I:43	55	X	interfață de linie de comandă pentru selecții X (clipboard)
xclip	V:15, I:74	62	X	interfață de linie de comandă pentru selecții X (clipboard)
wl-clipboard	V:7, I:23	162	Wayland	wl-copy wl-paste: interfață de linie de comandă pentru clipboardul Wayland
gpm	V:9, I:9	526	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 regionale 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

When auto-mounting a hot-pluggable [USB flash drive](#) under modern desktop environment such as GNOME, you may provide such mount option by right clicking the icon on the desktop, click "Drive" tab, click to expand "Setting", and entering "utf8" to "Mount options:". The next time this USB flash drive is mounted, mount with UTF-8 is enabled.

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

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 pachetele `keyboard-configuration` și `console-setup`.

```
# dpkg-reconfigure keyboard-configuration
# dpkg-reconfigure console-setup
```

Pentru consola Linux și sistemul X Window, aceasta actualizează parametrii de configurare din „/etc/default/keyboard” și „/etc/default/console-setup”. De asemenea, aceasta configurează fontul consolei Linux. Multe caractere non-ASCII, inclusiv caracterele accentuate utilizate de multe limbi europene, pot fi disponibile cu [tasta moartă](#), [tasta AltGr](#) și [tasta de compunere](#).

8.2.2 Introducerea de la tastatură pentru Wayland

Pentru GNOME în sistemul grafic de birou Wayland, Secțiune 8.2.1 nu poate oferi suport pentru limbile europene non-engleze. [IBus](#) a fost creat pentru a oferi suport nu numai pentru limbile asiatice, ci și pentru limbile europene. Dependența pachetului mediului grafic de birou GNOME recomandă „ibus” prin „gnome-shell”. Codul „ibus” a fost actualizat pentru a integra funcționalitățile `setxkbmap` și opțiunea XKB. Trebuie să configurați `ibus` din „Configurări GNOME” sau „Ajustări GNOME” pentru introducerea multilingvă de la tastatură.

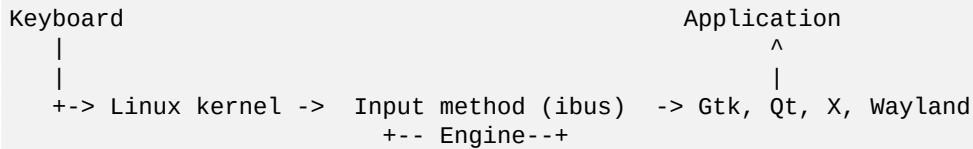
Notă

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](#).

8.2.3 Suportul pentru metoda de introducere cu IBus

Deoarece mediul grafic de birou GNOME recomandă „ibus” prin „gnome-shell”, „ibus” este alegerea potrivită pentru metoda de introducere.

Introducerea multilingvă în aplicație este procesată astfel:



Lista pachetelor IBus și a motorului său este următoarea.

pachet	popcon	popularity	configurația lingvistică acceptată
ibus	V:217, I:264	1874	infrastructura metodei de introducere a datelor utilizând dbus
ibus-mozc	V:2, I:3	980	japoneză
ibus-anthy	V:0, I:1	8867	, ,
ibus-skk	V:0, I:0	243	, ,
ibus-kkc	V:0, I:0	211	, ,
ibus-libpinyin	V:1, I:4	2769	chineză (pentru zh_CN)
ibus-chewing	V:0, I:0	288	, , (pentru zh_TW)
ibus-libzhuyin	I:0	41009	, , (pentru zh_TW)
ibus-rime	V:0, I:0	78	, , (pentru zh_CN/zh_TW)
ibus-cangjie	V:0, I:0	235	, , (pentru zh_HK)
ibus-hangul	V:0, I:2	264	coreeană
ibus-libthai	I:0	84	thailandeză
ibus-table-thai	I:0	59	thailandeză
ibus-unkey	V:0, I:0	286	vietnameză
ibus-keyman	V:0, I:0	191	multilingv: motorul Keyman pentru peste 2000 de limbi
ibus-table	V:0, I:1	2271	motor de tabele pentru IBus
ibus-m17n	V:0, I:1	448	multilingv: indic, arabă și altele
plasma-widgets-addons	V:68, I:113	5132	elemente grafice de control suplimentare, pentru Plasma 5 care conțin indicatorul tastaturii

Tabela 8.1: Lista IBus și a pachetelor sale de motoare

Notă

Pentru limba chineză, „fcitx5” poate fi o alternativă la infrastructura metodei de introducere. Pentru pasionații de Emacs, „uim” poate fi o alternativă. În ambele cazuri, poate fi necesară o configurare manuală suplimentară cu `im-config`. Unele [metode de introducere](#) clasice vechi, cum ar fi „kinput2”, pot exista încă în depozitul Debian, dar nu sunt recomandate pentru mediul modern.

8.2.4 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 pentru GNOME în Wayland:

1. Instalați pachetul de instrumente de introducere a textului în limba japoneză `ibus-mozc` (sau `ibus-anthy`) împreună cu pachetele recomandate, cum ar fi `im-config`.
2. Selectați „Configurări” → „Tastatură” → „Surse de introducere” → faceți clic pe „+” în „Surse de introducere” → „Japoneză” → „Japoneză mozc (sau anthy)” și faceți clic pe „Adaugă” dacă nu a fost activată.

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-încercuit).

- 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).

8.4 Caractere cu lățime de aspect ambiguu din Asia de Est

Under the East Asian locale, the box drawing, Greek, and Cyrillic characters may be displayed wider than your desired width to cause the unaligned terminal output (see [Unicode Standard Annex #11, 4.2 Ambiguous Characters](#)).

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)	dimensiune	descriere
mc	V:43, I:183	1590	consultați Secțiune 1.3
bsdutils	V:439, I:999	335	<code>script(1)</code> comandă pentru înregistrarea sesiunii terminalului
screen	V:54, I:199	1006	multiplexor de terminal cu emulare de terminal VT100/ANSI
tmux	V:79, I:153	1292	alternativă multiplexor de terminal (utilizați «Control-B» în loc de acesta)
fzf	V:8, I:31	4651	căutare text aproximativ
fzy	V:0, I:0	59	căutare text aproximativ
rlwrap	V:1, I:12	328	învăluitor de linie de comandă pentru funcționalitatea readline
ledit	V:0, I:8	375	învăluitor de linie de comandă pentru funcționalitatea readline
rlfe	V:0, I:0	45	învăluitor de linie de comandă pentru funcționalitatea readline
ripgrep	V:9, I:30	5342	căutare rapidă recursivă de șiruri în arborele codului sursă cu filtrare automată

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
<code>^A ?</code>	afișează un ecran de ajutor (afișează combinațiile de taste)
<code>^A c</code>	crează o fereastră nouă și comută la ea
<code>^A n</code>	trece la fereastră următoare
<code>^A p</code>	trece la fereastră anterioară
<code>^A 0</code>	trece la fereastră numărul 0
<code>^A 1</code>	trece la fereastră numărul 1
<code>^A w</code>	afișează o listă de ferestre
<code>^A a</code>	trimite «Ctrl-A» către fereastră curentă ca intrare de la tastatură
<code>^A h</code>	scrie o copie fizică a ferestrei curente într-un fișier
<code>^A H</code>	începe/încheie înregistrarea ferestrei curente în fișier
<code>^A ^X</code>	blochează terminalul (protejat cu parolă)
<code>^A d</code>	detașează sesiunea «screen» de terminal
<code>^A DD</code>	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("$") |
        \   exe "normal! g'"'"' |
        \ endif
augroup END
"""" Force to use underline for spell check results
augroup SpellUnderline
    autocmd!
    autocmd ColorScheme * highlight SpellBad term=Underline gui=Undercurl
    autocmd ColorScheme * highlight SpellCap term=Underline gui=Undercurl
    autocmd ColorScheme * highlight SpellLocal term=Underline gui=Undercurl
    autocmd ColorScheme * highlight SpellRare term=Underline gui=Undercurl
augroup END
"""" highlight trailing spaces except when typing as red (set after colorscheme)
highlight TailingWhitespaces ctermbg=red guibg=red
"""" \s\+      1 or more whitespace character: <Space> and <Tab>
"""" \%#\@<!  Matches with zero width if the cursor position does NOT match.
match TailingWhitespaces /\s\+\%#\@<!$/
```

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
- [VimAwesome](#) -- 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 [depozițul 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 modalități² de a gestiona și încărca aceste pachete externe în vim. Verificarea informațiilor originale este cea mai bună soluție.

combinația de taste	informații
<code>:help package</code>	explicație privind mecanismul pachetului vim
<code>:help runtimepath</code>	explicație privind mecanismul <code>runtimepath</code>
<code>:version</code>	stări interne, inclusiv candidați pentru fișierul <code>vimrc</code>
<code>:echo \$VIM</code>	variabila de mediu „\$VIM” utilizată pentru localizarea fișierului <code>vimrc</code>
<code>:set runtimepath?</code>	lista directoarelor care vor fi căutate pentru toate fișierele de suport pentru rulare
<code>:echo \$VIMRUNTIME</code>	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.

²[vim-pathogen](#) a fost popular.

pachet	popcon(popularitate)	descriere	
logwatch	V:9, I:10	2435	analizator de jurnale cu ieșire formatată plăcut, scris în Perl
fail2ban	V:95, I:106	2191	interzice adresele IP care provoacă erori multiple de autentificare
analog	V:3, I:88	3739	analizator de jurnale de server web
awstats	V:5, I:8	6935	analizator de jurnale de server web, performant și cu multe funcții
sarg	V:0, I:0	863	generator de rapoarte de analizare squid
pflogsumm	V:1, I:3	170	program de sinteză a intrărilor din jurnalul Postfix
fwlogwatch	V:0, I:0	487	analizator de jurnale de paravan de protecție
squidview	V:0, I:0	189	monitorizează și analizează fișierele squid access.log
swatch	V:0, I:0	99	vizor de fișiere jurnal cu potrivire de expresii regulate, evidențiere și cârlige
crm114	V:0, I:0	1371	Filtru de spam și filtru cu expresii regulate controlabile („Controllable Regex Mutilator”: CRM114)
icmpinfo	V:0, I:0	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'
```

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

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**

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	(popul. dintr-un sistem)	descriere
gnome-screenshot	V:13, I:111	1115	Wayland
flameshot	V:7, I:17	3532	Wayland
gimp	V:35, I:229	31748	Wayland + X
x11-apps	V:33, I:464	2461	X
imagemagick	I:290	77	X
scrot	V:4, I:53	141	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	(popul. dintr-un sistem)	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	utilitate de restaurare a sistemului folosind rsync sau instantanee BTRFS
snapper	V:6, I:8	2410	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](#).

pachet	popcon(popularitate)	descriere
coreutils	V:892, I:999 18457	nice(1): rulează un program cu prioritate de planificare modificată
bsdutils	V:439, I:999 335	renice(1): modifică prioritatea de planificare a unui proces în execuție
procps	V:818, I:998 2404	Utilități ale sistemului de fișiere „/proc”: ps(1), top(1), kill(1), watch(1), ...
psmisc	V:410, I:742 950	Utilități ale sistemului de fișiere „/proc”: killall(1), fuser(1), peekfd(1), pstree(1)
time	V:5, I:85 129	time(1): rulează un program pentru a raporta utilizarea resurselor sistemului în funcție de timp
sysstat	V:123, I:162 1904	sar(1), iostat(1), mpstat(1), ...: instrumente de performanță a sistemului pentru Linux
isag	V:0, I:3 109	Grafic interactiv al activității sistemului pentru sysstat
lsof	V:443, I:949 492	lsof(8): listează fișierele deschise de un proces în execuție folosind opțiunea „-p”
strace	V:10, I:103 3253	strace(1): urmărește apelurile de sistem și semnalele
ltrace	V:0, I:11 420	ltrace(1): trasează apelurile către bibliotecă
xtrace	V:0, I:0 353	xtrace(1): trasează comunicarea între clientul X11 și server
powertop	V:33, I:232 696	power top(1): informații despre consumul de energie al sistemului
cron	V:901, I:996 250	rulează procesele conform unei programări în fundal din cron(8) daemon
anacron	V:418, I:493 112	programator de comenzi de tip cron pentru sisteme care nu funcționează 24 de ore pe zi
at	V:74, I:102 158	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

9.4 Monitorizarea, controlul și inițierea activităților programului

Activitățile programului pot fi monitorizate și controlate folosind instrumente specializate.

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

Procese-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.

The desktop environment sets \$XDG_DATA_HOME and \$XDG_DATA_DIR environment variables. For example, under the 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:
 - GNOME desktop: "Settings" → "Apps" → "Default Apps"
 - KDE desktop: "Application Launcher" → "System Settings" → "Default Applications"
 - 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)`.

valoare semnal	nume semnal	acțiune	notă
0	---	nu se trimite niciun semnal (vedeți <code>kill(2)</code>)	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»

```
# 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
```

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.

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` sincronizează și udemontează toate sistemele de fișiere montate.

Nucleul Linux Debian amd64 actual (2021) are `/proc/sys/kernel/sysrq=438=0b110110110`:

tasta ce urmează după Alt-SysRq	descrierea acțiunii
k	kill (omoară) toate procesele din consola virtuală curentă (SAK)
s	sincronizează toate sistemele de fișiere montate pentru a evita coruperea datelor
u	remontează toate sistemele de fișiere montate în mod citire-numai (umount)
r	restabilește tastatura din modul raw (brut) după blocarea X

Tabela 9.12: Lista tastelor de comandă SAK importante

- 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 direc-toare din „/sys/bus/pci” (consultați Secțiune [1.2.12](#)).

pachet	popcon(popularitate)	dimensiune	descriere
pciutils	V:254, I:992	280	utilități PCI Linux: <code>lspci(8)</code>
usbutils	V:80, I:887	322	utilități USB Linux: <code>lsusb(8)</code>
nvme-cli	V:22, I:30	2222	utilități NVMe pentru Linux: <code>nvme(1)</code>
pcmciautils	V:4, I:7	92	utilități PCMCIA pentru Linux: <code>pccardctl(8)</code>
scsitools	V:0, I:2	261	colecție de instrumente pentru gestionarea hardware-ului SCSI: <code>lsscsi(8)</code>
procinfo	V:0, I:6	149	informații despre sistem obținute din „/proc”: <code>lsdev(8)</code>
lshw	V:13, I:90	971	informații despre configurația hardware: <code>lshw(1)</code>
discover	V:27, I:685	81	sistem de identificare hardware: <code>discover(8)</code>

Tabela 9.13: Lista instrumentelor de identificare a hardware-ului

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.

pachet	popcon(popularitate)	dimensiune	descriere
console-setup	V:77, I:973	420	utilități pentru fonturi și tabele de taste pentru consola Linux
x11-xserver-utils	V:312, I:542	559	utilități server X: <code>xset(1)</code> , <code>xmodmap(1)</code>
acpid	V:58, I:90	158	demon pentru gestionarea evenimentelor transmise de Interfața avansată de configurare și alimentare (ACPI)
acpi	V:7, I:85	49	instrument pentru afișarea informațiilor despre dispozitivele ACPI
sleepd	V:0, I:0	84	demon pentru a pune laptopul în stare de repaus în timpul inactivității
hdparm	V:117, I:222	246	optimizarea accesului la discurile dure (a se vedea Secțiune 9.6.9)
smartmontools	V:229, I:264	2455	control and monitor storage systems using S.M.A.R.T.
setserial	V:3, I:5	104	colecție de instrumente pentru gestionarea porturilor seriale
memtest86+	V:0, I:19	12473	colecție de instrumente pentru gestionarea hardware-ului de memorie
scsitools	V:0, I:2	261	colecție de instrumente pentru gestionarea hardware-ului SCSI
setcd	V:0, I:0	33	optimizarea accesului la unitatea de disc compact
big-cursor	I:0	26	cursoare de mouse mai mari pentru X

Tabela 9.14: Lista instrumentelor de configurare hardware

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`.

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 Configurația terminalului

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”.

9.5.7 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	popularity	descriere
alsa-utils	V:341, I:475	2702	Instrumente pentru configurarea și utilizarea ALSA
oss-compat	V:0, I:10	18	compatibilitatea OSS sub ALSA previne erorile „/dev/dsp not found”
pipewire	V:316, I:373	142	audio and video processing engine multimedia server - metapachet
pipewire-bin	V:324, I:373	2094	audio and video processing engine multimedia server - server audio și programe CLI
pipewire-alsa	V:168, I:238	197	audio and video processing engine multimedia server - server audio pentru a înlocui ALSA
pipewire-pulse	V:280, I:340	64	audio and video processing engine multimedia server - server audio pentru a înlocui PulseAudio
pulseaudio	V:166, I:197	6606	server PulseAudio
libpulse0	V:439, I:587	977	bibliotecă client pentru PulseAudio
jackd	V:2, I:15	8	server (latență redusă) JACK Audio Connection Kit. (JACK)
libjack0	V:2, I:9	330	bibliotecă (latență redusă) pentru JACK Audio Connection Kit. (JACK)
libgststreamer1.0-0	V:465, I:602	5280	GStreamer : motorul de sunet GNOME
libphonon4qt5-4	V:28, I:64	572	Phonon : motorul de sunet KDE

Tabela 9.15: Lista pachetelor de sunet

9.5.8 Dezactivarea protectorului de ecran

Pentru a dezactiva protectorul de ecran, utilizați următoarele comenzi.

mediu	comanda
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 (interfața grafică de configurare a protectorului de ecran)	xscreensaver-command -prefs

Tabela 9.16: Lista comenzilor pentru dezactivarea protectorului de ecran

9.5.9 Dezactivarea bipurilor sonore

Se poate întotdeauna deconecta difuzorul PC-ului pentru a dezactiva sunetele bip. Eliminarea modului nucleului pcspkr face acest lucru pentru dvs.

Următorul cod împiedică programul read line(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
```

9.5.10 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

9.5.11 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](#).

pachet	popcon(popularitate)	nr. de instalații	descriere
logcheck	V:5, I:6	120	demon pentru a trimite prin poșta electronică anomaliile din fișierele jurnal ale sistemului către administrator
debsums	V:4, I:30	107	ustensilă pentru verificarea fișierelor pachetelor instalate în raport cu sumele de control MD5
chkrootkit	V:9, I:14	966	detector de rootkit
clamav	V:8, I:39	33154	ustensilă antivirus pentru Unix - interfață de linie de comandă
tiger	V:1, I:1	7800	raportează vulnerabilitățile de securitate ale sistemului
tripwire	V:1, I:1	5050	verificator de integritate a fișierelor și directoarelor
john	V:1, I:7	469	instrument activ de spargere a parolelor
aide	V:1, I:2	331	mediu avansat de detectare a intruziunilor - binar static
integrit	V:0, I:0	2939	program de verificare a integrității fișierelor
crack	V:0, I:0	153	program de ghicire a parolelor

Tabela 9.18: Lista instrumentelor pentru verificarea securității și integrității sistemului

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.

9.6 Sfaturi pentru stocarea datelor

Booting your system as the Linux live system (see Secțiune 3.2.2 and Secțiune 3.2.3) makes it easy for you to reconfigure data storage on the installed system.

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(1x)` 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	populabilitate	descriere
util-linux	V:897, I:999	4384	utilitare de sistem diverse, inclusiv <code>fdisk(8)</code> și <code>cfdisk(8)</code>
parted	V:445, I:579	126	programul GNU Parted pentru redimensionarea partițiilor de disc
gparted	V:13, I:92	2313	editor de partiții GNOME bazat pe <code>libparted</code>
gdisk	V:20, I:309	940	editor de partiții pentru discul hibrid GPT/MBR
kpartx	V:17, I:27	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)
- bibliotecă 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 `unde1` pentru a recupera fișierele șterse).
-

pachet	popcon(popularitate)	descriere
e2fsprogs	V:796, I:997 1549	ustensile pentru sistemele de fișiere ext2/ext3/ext4
btrfs-progs	V:48, I:76 5204	ustensile pentru sistemul de fișiere Btrfs
reiserfsprogs	V:9, I:21 473	ustensile pentru sistemul de fișiere Reiserfs
zfsutils-linux	V:31, I:32 1891	ustensile pentru sistemul de fișiere OpenZFS
dosfstools	V:247, I:572 310	ustensile pentru sistemul de fișiere FAT (Microsoft: MS-DOS, Windows)
exfatprogs	V:36, I:467 352	ustensile pentru sistemul de fișiere exFAT menținut de Samsung.
exfat-fuse	V:2, I:50 73	citește/scrie sistemul de fișiere exFAT controlor (Microsoft) pentru FUSE.
xfsprogs	V:36, I:86 4386	ustensile pentru sistemul de fișiere XFS (SGI: IRIX)
ntfs-3g	V:193, I:527 1500	citește/scrie sistemul de fișiere NTFS (Microsoft: Windows NT, ...) controlor pentru FUSE.
jfsutils	V:0, I:7 1104	ustensile pentru sistemul de fișiere JFS . (IBM: AIX, OS/2)
xorriso	V:14, I:63 347	utilities for the ISO-9660 filesystem and CD/DVD writing from libburnia
wodim	V:8, I:96 898	command line CD/DVD writing tool from cdrkit
genisoimage	V:18, I:167 1567	command line ISO-9660 filesystem tool from cdrkit
reiser4progs	V:0, I:1 1367	ustensile pentru sistemul de fișiere Reiser4
hfsprogs	V:0, I:3 394	ustensile pentru sistemul de fișiere HFS și HFS Plus (Apple: Mac OS)
zerofree	V:5, I:119 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

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 suportul asigurat 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**.

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-dfce3146d2f7	/	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 converția sistemului de fișiere de la `ext2` la `ext3` pe „`/dev/sda1`”; (efecuați această operațiune pe sistemul de fișiere nemontat).
- Executarea comenzii „`sudo tune2fs -O extents,uninit_bg,dir_index /dev/sda1 && fsck -pf /dev/sda1`” 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.
 - decommentați „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

The disk image file, "disk.img", of an unmounted device, e.g., the second SCSI or serial ATA drive "/dev/sdb", can be made by one of the following.

```
# 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

The disk image file, "disk.img" can be written to an unmounted device, e.g., the second SCSI drive "/dev/sdb" with matching size, by one of the following.

```
# 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

Both `genisoimage(1)` provided by [cdrkit](#) and `xorrisofs(1)` provided by [Libburnia](#) share the same command syntax except for the command name.

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 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 is only a large CD to `wodim(1)` provided by [cdrkit](#) and `xorrecord(1)` provided by [Libburnia](#). These commands share the same command syntax except for the command name.

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	popularity	descriere
coreutils	V:892, I:999	18457	pachetul de bază care are <code>od(1)</code> pentru a descărca fișiere (HEX, ASCII, OCTAL, ...)
bsdmainutils	V:4, I:152	17	pachet de ustensiler care are <code>hd(1)</code> pentru a descărca fișiere (HEX, ASCII, OCTAL, ...)
hexedit	V:1, I:8	70	editor și vizor binar (HEX, ASCII)
bless	V:0, I:1	924	editor hexazecimal cu funcții avansate (GNOME)
okteta	V:1, I:12	1590	editor hexazecimal cu funcții avansate (KDE4)
ncurses-hexedit	V:0, I:1	130	editor și vizor binar (HEX, ASCII, EBCDIC)
beav	V:0, I:0	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	popularity	descriere
mtools	V:7, I:54	400	instrumente pentru fișiere MSDOS fără a le monta
hfsutils	V:0, I:3	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)	descriere	
par2	V:10, I:119	298	set de volume de arhivă Parity (Parity Archive Volume Set), pentru verificarea și repararea fișierelor
dvdisaster	V:0, I:1	1422	protecție împotriva pierderii datelor/zgârieturilor/îmbătrânirii pentru suporturile CD/DVD
dvbackup	V:0, I:0	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)	descriere	
testdisk	V:2, I:26	1495	ustensile pentru scanarea partițiilor și recuperarea discurilor
magicrescue	V:0, I:2	257	ustensilă pentru recuperarea fișierelor prin căutarea de octeți magici
scalpel	V:0, I:2	89	economic, cu performanțe ridicate în prelucrarea fișierelor pentru recuperarea acestora
myrescue	V:0, I:2	83	recuperează date de pe discuri dure deteriorate
extundelete	V:0, I:7	152	instrument pentru recuperarea fișierelor șterse din sistemul de fișiere ext3/4
ext4magic	V:0, I:3	235	instrument pentru recuperarea fișierelor șterse din sistemul de fișiere ext3/4
ext3grep	V:0, I:2	299	instrument pentru recuperarea fișierelor șterse din sistemul de fișiere ext3
scrounge-ntfs	V:0, I:1	49	program de recuperare a datelor pentru sisteme de fișiere NTFS
gzrt	V:0, I:0	33	set de instrumente de recuperare gzip
sleuthkit	V:2, I:23	1729	instrumente pentru analiza tehnico-criminalistică. (Sleuthkit -- Kit de detectiv)
autopsy	V:0, I:1	1026	interfață grafică pentru SleuthKit
foremost	V:0, I:4	102	aplicație tehnico-criminalistică pentru recuperarea datelor
guymager	V:0, I:0	1047	instrument de procesare a imaginilor de fișiere tehnico-criminalistică bazat pe Qt
dcfldd	V:0, I:2	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 `unde1` 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

There are several ways to completely erase data from an entire hard disk like device, e.g., [USB flash drive](#) at `"/dev/sda"`.

**Atenție**

Check your [USB flash drive](#) location with `mount(8)` first before executing commands here. The device pointed by `"/dev/sda"` may be SCSI hard disk or serial-ATA hard disk where your entire system resides.

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

Unused area on an hard disk (or [USB flash drive](#)), e.g. `"/dev/sdb1"` may still contain erased data themselves since they are only unlinked from the filesystem. These can be cleaned by overwriting them.

```
# 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**

This is usually good enough for your [USB flash drive](#). But this is not perfect. Most parts of erased filenames and their attributes may be hidden and remain in the filesystem.

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 `lsuf` instalat) următoarele.

```
$ ls -li bar
2228329 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:02 bar
$ lsuf |grep bar|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/bar
$ rm bar
$ lsuf |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 comun „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.

```
# lsuf -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)	nr. de pachete	descriere
cryptsetup	V:30, I:80	465	instrumente pentru dispozitivul de blocuri criptat (dm-crypt / LUKS)
cryptmount	V:2, I:2	231	instrumente pentru dispozitivul de blocuri criptat (dm-crypt / LUKS) cu accent pe montarea/demontarea de către utilizatorii normali
fscrypt	V:0, I:1	6471	instrumente pentru criptarea sistemelor de fișiere Linux (fscrypt)
libpam-fscrypt	V:0, I:0	5589	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

You can encrypt contents of removable mass devices, e.g. [USB flash drive](#) on `/dev/sdx`, using [dm-crypt/LUKS](#). You simply format it as the following.

```
# 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	popularity	descriere
build-essential	I:504	17	pachete esențiale pentru construirea pachetelor Debian: make, gcc, ...
bzip2	V:168, I:972	114	instrumente de comprimare și decompimare pentru fișiere bz2
libncurses5-dev	I:40	6	biblioteci pentru dezvoltatori și documente pentru ncurses
git	V:382, I:597	50172	git: sistem distribuit de control al reviziei utilizat de nucleul Linux
fakeroot	V:31, I:507	225	furnizează mediul fakeroot pentru construirea pachetului ca non-root
initramfs-tools	V:479, I:988	52	instrument pentru a construi un initramfs (specific Debian)
dkms	V:74, I:146	235	dynamic kernel module support (DKMS) (generic)
module-assistant	V:0, I:14	391	instrument de ajutor pentru crearea pachetului de module (specific Debian)
devscripts	V:5, I:33	2770	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)` 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/v6.x/linux-version.tar.xz
$ tar --xz -xvf linux-version.tar.xz
$ cd linux-version
$ cp /boot/config-version .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-3.*
$ 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

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.

pachet	popcon(popularitate)	dimensiune	descriere
coreutils	V:892, I:999	18457	Instrumente GNU de bază care conțin chroot(8)
util-linux	V:897, I:999	4384	utilități de sistem diverse care conțin unshare(1)
systemd-container	V:72, I:76	2276	instrumente systemd container/nspawn care conțin systemd-nspawn(1)
schroot	V:5, I:7	2627	instrument specializat pentru executarea pachetelor binare Debian în chroot
sbuild	V:1, I:4	157	instrument pentru construirea de pachete binare Debian din surse Debian
debootstrap	V:4, I:46	330	crează și pornește un sistem Debian de bază (scris în sh)
mmdebstrap	V:5, I:10	574	crează și pornește un sistem Debian (scris în Perl)
cdebootstrap	V:0, I:1	114	crează și pornește un sistem Debian (scris în C)
cloud-image-utils	V:0, I:14	66	utilități de gestionare a imaginilor în cloud
cloud-guest-utils	V:4, I:18	71	utilități pentru invitați în cloud
virt-manager	V:12, I:49	2310	Gestionar mașină virtuală : aplicație de mediu grafic de birou pentru gestionarea mașinilor virtuale
libvirt-clients	V:49, I:72	1155	programe pentru biblioteca libvirt
incus	V:0, I:2	21	Incus : container de sistem și gestionar de mașini virtuale
podman	V:25, I:29	81828	podman : motor pentru a rula containere bazate pe OCI în Pods
podman-docker	V:2, I:2	275	motor pentru a rula containere bazate pe OCI în Pods - învâluitor pentru docker
docker.io	V:44, I:47	95958	docker : mediu de execuție de containere Linux
games-emulator	I:0	21	games-emulator : emulatoare Debian pentru jocuri
bochs	V:0, I:0	8180	Bochs : emulator PC IA-32
qemu-system	I:22	80	QEMU : binare de emulare completă a sistemului
qemu-user	V:5, I:9	464225	QEMU : binare de emulare în modul utilizator
qemu-utils	V:14, I:109	12157	QEMU : utilități
qemu-system-x86	V:53, I:93	67511	KVM : virtualizare completă pe hardware x86 cu hardware-assisted virtualization
virtualbox	V:3, I:4	151525	VirtualBox : soluție de virtualizare x86 pe i386 și amd64
gnome-boxes	V:1, I:6	6847	Boxes : aplicație GNOME simplă pentru accesarea sistemelor virtuale
xen-tools	V:0, I:1	719	instrumente pentru gestionarea serverului virtual debian XEN
wine	V:14, I:57	204	Wine : implementarea API Windows (suita standard)
dosbox	V:1, I:13	2697	DOSBox : emulator x86 cu grafică Tandy/Herc/CGA/EGA/VGA/SVGA, sunet și DOS
lxc	V:9, I:12	1626	Linux containers user space tools
python3-venv	V:8, I:136	6	venv pentru crearea de medii virtuale python (biblioteca de sistem)
python3-virtualenv	V:8, I:41	417	virtualenv pentru crearea de medii python virtuale izolate
pipx	V:7, I:43	3613	pipx pentru instalarea aplicațiilor python în medii izolate

Tabela 9.27: Lista instrumentelor de virtualizare

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](#), `vfat` (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 [sbuid](#) pentru a construi pachete Debian din sursă utilizează mediul `chroot` gestionat de pachetul [schroot](#). Acesta vine cu scriptul ajutor `sbuid-createchroot(1)`. Să învățăm cum funcționează executându-l după cum urmează.

```
$ sudo mkdir -p /srv/chroot
$ sudo sbuid-createchroot -v --include=eatmydata,ccache unstable /srv/chroot/unstable- ↵
amd64-sbuid 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-sbuid
```

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 `sbuid-createchroot` conform celor de mai sus. De exemplu, „`/sys`”, „`/etc/passwd`”, „`/etc/group`”, „`/var/run/utmp`”, „`/var/log/wtmp`” etc. pot necesita să fie montate cu opțiunea „`--bind`” sau copiate.

Indicație

Pachetul `sbuid` 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 [sbuid la Debian wiki](#) și [sbuid 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

Tools and tips for managing binary and text data on the Debian system are described.

10.1 Sharing, copying, and archiving

**Avertisment**

The uncoordinated write access to actively accessed devices and files from multiple processes must not be done to avoid the [race condition](#). [File locking](#) mechanisms using `flock(1)` may be used to avoid it.

The security of the data and its controlled sharing have several aspects.

- Crearea arhivei de date
- Accesul la stocare de la distanță
- Duplicarea
- The tracking of the modification history
- The facilitation of data sharing
- The prevention of unauthorized file access
- The detection of unauthorized file modification

These can be realized by using some combination of tools.

- Archive and compression tools
 - Copy and synchronization tools
 - Network filesystems
 - Removable storage media
 - The secure shell
 - The authentication system
 - Version control system tools
 - Hash and cryptographic encryption tools
-

10.1.1 Archive and compression tools

Here is a summary of archive and compression tools available on the Debian system.

**Avertisment**

Do not set the "\$TAPE" variable unless you know what to expect. It changes tar(1) behavior.

- The gzipped tar(1) archive uses the file extension ".tgz" or ".tar.gz".
- The xz-compressed tar(1) archive uses the file extension ".txz" or ".tar.xz".
- Popular compression method in FOSS tools such as tar(1) has been moving as follows: gzip → bzip2 → xz
- cp(1), scp(1) and tar(1) may have some limitation for special files. cpio(1) is most versatile.
- cpio(1) is designed to be used with find(1) and other commands and suitable for creating backup scripts since the file selection part of the script can be tested independently.
- Internal structure of Libreoffice data files are ".jar" file which can be opened also by unzip.
- The de-facto cross platform archive tool is zip. Use it as "zip -rX" to attain the maximum compatibility. Use also the "-s" option, if the maximum file size matters.

10.1.2 Copy and synchronization tools

Here is a summary of simple copy and backup tools available on the Debian system.

Copying files with rsync(8) offers richer features than others.

- delta-transfer algorithm that sends only the differences between the source files and the existing files in the destination
- quick check algorithm (by default) that looks for files that have changed in size or in last-modified time
- "--exclude" and "--exclude-from" options similar to tar(1)
- "a trailing slash on the source directory" syntax that avoids creating an additional directory level at the destination.

Indicație

Version control system (VCS) tools in Tabel [10.14](#) can function as the multi-way copy and synchronization tools.

10.1.3 Idioms for the archive

Here are several ways to archive and unarchive the entire content of the directory "./source" using different tools.

GNU tar(1):

```
$ tar -cvJf archive.tar.xz ./source
$ tar -xvJf archive.tar.xz
```

Alternatively, by the following.

```
$ 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
```

pachet	popcon(popularitate)	unitate	extensie	comanda	comentariu
tar	V:889, I:999	3085	.tar	tar(1)	arhivatorul standard (standardul de facto)
cpio	V:404, I:998	1201	.cpio	cpio(1)	Arhivator în stil Unix System V, se utilizează cu find(1)
binutils	V:184, I:636	1119	.ar	ar(1)	arhivator pentru crearea de biblioteci statice
fastjar	V:1, I:10	183	.jar	fastjar(1)	arhivator pentru Java (precum zip)
pax	V:5, I:10	167	.pax	pax(1)	nou arhivator standard POSIX, compromis între tar și cpio
gzip	V:885, I:999	256	.gz	gzip(1), zcat(1), ...	GNU LZ77 compression utility (de facto standard)
bzip2	V:168, I:972	114	.bz2	bzip2(1), bzip2cat(1), ...	Burrows-Wheeler block-sorting compression utility with higher compression ratio than gzip(1) (slower than gzip with similar syntax)
lzma	V:0, I:11	349	.lzma	lzma(1)	LZMA compression utility with higher compression ratio than gzip(1) (deprecated)
xz-utils	V:309, I:980	1475	.xz	xz(1), xzdec(1), ...	XZ compression utility with higher compression ratio than bzip2(1) (slower than gzip but faster than bzip2; replacement for LZMA compression utility)
zstd	V:265, I:777	2313	.zstd	zstd(1), zstdcat(1), ...	Zstandard fast lossless compression utility
p7zip	V:8, I:237	8	.7z	7zr(1), p7zip(1)	7-Zip file archiver with high compression ratio (LZMA compression)
p7zip-full	V:27, I:259	12	.7z	7z(1), 7za(1)	7-Zip file archiver with high compression ratio (LZMA compression and others)
lzop	V:12, I:137	164	.lzo	lzop(1)	LZO compression utility with higher compression and decompression speed than gzip(1) (lower compression ratio than gzip with similar syntax)
zip	V:50, I:367	627	.zip	zip(1)	InfoZIP : DOS archive and compression tool
unzip	V:116, I:759	387	.zip	unzip(1)	InfoZIP : DOS unarchive and decompression tool

Tabela 10.1: List of archive and compression tools

pachet	popcon(popularity)	limite de dimensiune	instrument	funcție
coreutils	V:892, I:999	18457	GNU cp	locally copy files and directories ("-a" for recursive)
openssh-client	V:898, I:996	5133	scp	remotely copy files and directories (client, "-r" for recursive)
openssh-server	V:745, I:804	3502	sshd	remotely copy files and directories (remote server)
rsync	V:200, I:539	814		1-way remote synchronization and backup
unison	V:3, I:13	14		2-way remote synchronization and backup

Tabela 10.2: List of copy and synchronization tools

10.1.4 Idioms for the copy

Here are several ways to copy the entire content of the directory `"/source"` using different tools.

- Local copy: `"/source"` directory → `"/dest"` directory
- Remote copy: `"/source"` directory at local host → `"/dest"` directory at `"user@host.dom"` host

rsync(8):

```
# cd ./source; rsync -aHAXSv . /dest
# cd ./source; rsync -aHAXSv . user@host.dom:/dest
```

You can alternatively use "a trailing slash on the source directory" syntax.

```
# rsync -aHAXSv ./source/ /dest
# rsync -aHAXSv ./source/ user@host.dom:/dest
```

Alternatively, by the following.

```
# 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 xvpf - )
# (cd ./source && tar cf - . ) | ssh user@host.dom '(cd /dest && tar xvpf - )'
```

cpio(1):

```
# cd ./source; find . -print0 | cpio -pvdm --null --sparse /dest
```

You can substitute `"/"` with `"foo"` for all examples containing `"/"` to copy files from `"/source/foo"` directory to `"/dest/foo"` directory.

You can substitute `"/"` with the absolute path `"/path/to/source/foo"` for all examples containing `"/"` to drop `"cd ./source;"`. These copy files to different locations depending on tools used as follows.

- `"/dest/foo"`: rsync(8), GNU cp(1), and scp(1)
- `"/dest/path/to/source/foo"`: GNU tar(1), and cpio(1)

Indicație

rsync(8) and GNU cp(1) have option `"-u"` to skip files that are newer on the receiver.

10.1.5 Idioms for the selection of files

`find(1)` is used to select files for archive and copy commands (see Secțiune 10.1.3 and Secțiune 10.1.4) or for `xargs(1)` (see Secțiune 9.4.9). This can be enhanced by using its command arguments.

Basic syntax of `find(1)` can be summarized as the following.

- Its conditional arguments are evaluated from left to right.
- This evaluation stops once its outcome is determined.
- "Logical **OR**" (specified by `"-o"` between conditionals) has lower precedence than "logical **AND**" (specified by `"-a"` or nothing between conditionals).
- "Logical **NOT**" (specified by `"!"` before a conditional) has higher precedence than "logical **AND**".
- `"-prune"` always returns logical **TRUE** and, if it is a directory, searching of file is stopped beyond this point.
- `"-name"` matches the base of the filename with shell glob (see Secțiune 1.5.6) but it also matches its initial `"."` with metacharacters such as `"*"` and `"?"`. (New [POSIX](#) feature)
- `"-regex"` matches the full path with emacs style **BRE** (see Secțiune 1.6.2) as default.
- `"-size"` matches the file based on the file size (value preceded with `"+"` for larger, preceded with `"-"` for smaller)
- `"-newer"` matches the file newer than the one specified in its argument.
- `"-print0"` always returns logical **TRUE** and print the full filename ([null terminated](#)) on the standard output.

`find(1)` is often used with an idiomatic style as the following.

```
# 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
```

This means to do following actions.

1. Search all files starting from `"/path/to"`
2. Globally limit its search within its starting filesystem and uses **ERE** (see Secțiune 1.6.2) instead
3. Exclude files matching regex of `".*\.cpio"` or `".*~"` from search by stop processing
4. Exclude directories matching regex of `".*\/\.git"` from search by stop processing
5. Exclude files larger than 99 Megabytes (units of 1048576 bytes) from search by stop processing
6. Print filenames which satisfy above search conditions and are newer than `"/path/to/timestamp"`

Please note the idiomatic use of `"-prune -o"` to exclude files in the above example.

Notă

For non-Debian [Unix-like](#) system, some options may not be supported by `find(1)`. In such a case, please consider to adjust matching methods and replace `"-print0"` with `"-print"`. You may need to adjust related commands too.

10.1.6 Archive media

When choosing [computer data storage media](#) for important data archive, you should be careful about their limitations. For small personal data backup, I use CD-R and DVD-R by the brand name company and store in a cool, shaded, dry, clean environment. (Tape archive media seem to be popular for professional use.)

Notă

[A fire-resistant safe](#) are meant for paper documents. Most of the computer data storage media have less temperature tolerance than paper. I usually rely on multiple secure encrypted copies stored in multiple secure locations.

Optimistic storage life of archive media seen on the net (mostly from vendor info).

- 100+ years : Acid free paper with ink
- 100 years : Optical storage (CD/DVD, CD/DVD-R)
- 30 years : Magnetic storage (tape, floppy)
- 20 years : Phase change optical storage (CD-RW)

These do not count on the mechanical failures due to handling etc.

Optimistic write cycle of archive media seen on the net (mostly from vendor info).

- 250,000+ cycles : Harddisk drive
- 10,000+ cycles : Flash memory
- 1,000 cycles : CD/DVD-RW
- 1 cycles : CD/DVD-R, paper

**Atenție**

Figures of storage life and write cycle here should not be used for decisions on any critical data storage. Please consult the specific product information provided by the manufacture.

Indicație

Since CD/DVD-R and paper have only 1 write cycle, they inherently prevent accidental data loss by overwriting. This is advantage!

Indicație

If you need fast and frequent backup of large amount of data, a hard disk on a remote host linked by a fast network connection, may be the only realistic option.

Indicație

If you use re-writable media for your backups, use of filesystem such as [btrfs](#) or [zfs](#) which supports read-only snapshots may be a good idea.

10.1.7 Removable storage device

Removable storage devices may be any one of the following.

- [USB flash drive](#)
- [Hard disk drive](#)
- [Optical disc drive](#)
- Digital camera
- Digital music player

They may be connected via any one of the following.

- [USB](#)
- [IEEE 1394 / FireWire](#)
- [PC Card](#)

Modern desktop environments such as GNOME and KDE can mount these removable devices automatically without a matching `/etc/fstab` entry.

- `udisks2` package provides a daemon and associated utilities to mount and unmount these devices.
- [D-bus](#) creates events to initiate automatic processes.
- [PolicyKit](#) provides required privileges.

Indicație

Automounted devices may have the `uhelper=` mount option which is used by `umount(8)`.

Indicație

Automounting under modern desktop environment happens only when those removable media devices are not listed in `/etc/fstab`.

Mount point under modern desktop environment is chosen as `/media/username/disk_label` which can be customized by the following.

- `mlabel(1)` for FAT filesystem
- `genisoimage(1)` with `-V` option for ISO9660 filesystem
- `tune2fs(1)` with `-L` option for ext2/ext3/ext4 filesystem

Indicație

The choice of encoding may need to be provided as mount option (see Secțiune [8.1.3](#)).

Indicație

The use of the GUI menu to unmount a filesystem may remove its dynamically generated device node such as `/dev/sdc`. If you wish to keep its device node, unmount it with the `umount(8)` command from the shell prompt.

10.1.8 Filesystem choice for sharing data

When sharing data with other system via removable storage device, you should format it with common [filesystem](#) supported by both systems. Here is a list of filesystem choices.

Notă

"Hard disk like device" means storage devices such as [HDD](#) / [SSD](#) / [USB flash drive](#) / Their device names have several variants; `/dev/sda`, `/dev/nvme0`, `/dev/mmcblk0`,

numele sistemului de fișiere	scenariul de utilizare tipic
FAT12	cross platform sharing of data on the floppy disk (<32MiB)
FAT16	cross platform sharing of data on the small hard disk like device (<2GiB)
FAT32	cross platform sharing of data on the large hard disk like device (<8TiB, supported by newer than MS Windows95 OSR2)
exFAT	cross platform sharing of data on the large hard disk like device (<512TiB, supported by WindowsXP, Mac OS X Snow Leopard 10.6.5, and Linux kernel since 5.4 release)
NTFS	cross platform sharing of data on the large hard disk like device (supported natively on MS Windows NT and later version, and supported by NTFS-3G via FUSE on Linux)
ISO9660	cross platform sharing of static data on CD-R and DVD+/-R
UDF	incremental data writing on CD-R and DVD+/-R (new)
MINIX	space efficient unix file data storage on the floppy disk
ext2	sharing of data on the hard disk like device with older Linux systems
ext3	sharing of data on the hard disk like device with older Linux systems
ext4	sharing of data on the hard disk like device with current Linux systems
btrfs	sharing of data on the hard disk like device with current Linux systems with read-only snapshots

Tabela 10.3: List of filesystem choices for removable storage devices with typical usage scenarios

Indicație

See Secțiune [9.9.1](#) for cross platform sharing of data using device level encryption.

The FAT filesystem is supported by almost all modern operating systems and is quite useful for the data exchange purpose via removable hard disk like media.

When formatting removable hard disk like devices for cross platform sharing of data with the FAT filesystem, the following should be safe choices.

- Partitioning them with `fdisk(8)`, `cfdisk(8)` or `parted(8)` (see Secțiune [9.6.2](#)) into a single primary partition and to mark it as the following.
 - Type "6" for FAT16 for media smaller than 2GB.
 - Type "c" for FAT32 (LBA) for larger media.
- Formatting the primary partition with `mkfs.vfat(8)` with the following.
 - Just its device name, e.g. `"/dev/sda1"` for FAT16
 - The explicit option and its device name, e.g. `"-F 32 /dev/sda1"` for FAT32

When using the FAT or ISO9660 filesystems for sharing data, the following should be the safe considerations.

- Archiving files into an archive file first using `tar(1)`, or `cpio(1)` to retain the long filename, the symbolic link, the original Unix file permission and the owner information.
- Splitting the archive file into less than 2 GiB chunks with the `split(1)` command to protect it from the file size limitation.
- Encrypting the archive file to secure its contents from the unauthorized access.

Notă

For FAT filesystems by its design, the maximum file size is $(2^{32} - 1)$ bytes = (4GiB - 1 byte). For some applications on the older 32 bit OS, the maximum file size was even smaller $(2^{31} - 1)$ bytes = (2GiB - 1 byte). Debian does not suffer the latter problem.

Notă

Microsoft itself does not recommend to use FAT for drives or partitions of over 200 MB. Microsoft highlights its short comings such as inefficient disk space usage in their "[Overview of FAT, HPFS, and NTFS File Systems](#)". Of course, we should normally use the ext4 filesystem for Linux.

Indicație

For more on filesystems and accessing filesystems, please read "[Filesystems HOWTO](#)".

10.1.9 Sharing data via network

When sharing data with other system via network, you should use common service. Here are some hints.

network service	description of typical usage scenario
SMB/CIFS network mounted filesystem with Samba	sharing files via "Microsoft Windows Network", see <code>smb.conf(5)</code> and The Official Samba 3.x.x HOWTO and Reference Guide or the <code>samba-doc</code> package
NFS network mounted filesystem with the Linux kernel	sharing files via "Unix/Linux Network", see <code>exports(5)</code> and Linux NFS-HOWTO
serviciul HTTP	sharing file between the web server/client
HTTPS service	sharing file between the web server/client with encrypted Secure Sockets Layer (SSL) or Transport Layer Security (TLS)
FTP service	sharing file between the FTP server/client

Tabela 10.4: List of the network service to chose with the typical usage scenario

Although these filesystems mounted over network and file transfer methods over network are quite convenient for sharing data, these may be insecure. Their network connection must be secured by the following.

- Encrypt it with [SSL/TLS](#)
- Tunnel it via [SSH](#)
- Tunnel it via [VPN](#)
- Limit it behind the secure firewall

See also Secțiune [6.5](#) and Secțiune [6.6](#).

10.2 Backup and recovery

We all know that computers fail sometime or human errors cause system and data damages. Backup and recovery operations are the essential part of successful system administration. All possible failure modes hit you some day.

Indicație

Keep your backup system simple and backup your system often. Having backup data is more important than how technically good your backup method is.

10.2.1 Backup and recovery policy

There are 3 key factors which determine actual backup and recovery policy.

1. Knowing what to backup and recover.

- Data files directly created by you: data in `"~/`
- Data files created by applications used by you: data in `" /var /"` (except `" /var /cache /"`, `" /var /run /"`, and `" /var /tmp /"`)
- System configuration files: data in `" /etc /"`
- Local programs: data in `" /usr /local /"` or `" /opt /"`
- System installation information: a memo in plain text on key steps (partition, ...)
- Proven set of data: confirmed by experimental recovery operations in advance
 - Cron job as a user process: files in `" /var /spool /cron /crontabs "` directory and restart `cron(8)`. See [Secțiune 9.4.14](#) for `cron(8)` and `crontab(1)`.
 - Systemd timer jobs as user processes: files in `" ~/ .config /systemd /user "` directory. See `systemd.timer(5)` and `systemd.service(5)`.
 - Autostart jobs as user processes: files in `" ~/ .config /autostart "` directory. See [Desktop Application Autostart Specification](#).

2. Knowing how to backup and recover.

- Secure storage of data: protection from overwrite and system failure
- Frequent backup: scheduled backup
- Redundant backup: data mirroring
- Fool proof process: easy single command backup

3. Assessing risks and costs involved.

- Risk of data when lost
 - Data should be at least on different disk partitions preferably on different disks and machines to withstand the filesystem corruption. Important data are best stored on a read-only filesystem. [1](#)
- Risk of data when breached
 - Sensitive identity data such as `" /etc /ssh /ssh_host _*_key "`, `" ~/ .gnupg /*"` , `" ~/ .ssh /*"` , `" ~/ .local /share "`, `" /etc /passwd "`, `" /etc /shadow "`, `" popularity-contest.conf "`, `" /etc /ppp /pap-secrets "`, and `" /etc /x11 /xorg.conf "` should be backed up as encrypted. [2](#) (See [Secțiune 9.9.](#))
 - Never hard code system login password nor decryption passphrase in any script even on any trusted system. (See [Secțiune 10.3.6.](#))

¹A write-once media such as CD/DVD-R can prevent overwrite accidents. (See [Secțiune 9.8](#) for how to write to the storage media from the shell commandline. GNOME desktop GUI environment gives you easy access via menu: "Places → CD/DVD Creator".)

²Some of these data can not be regenerated by entering the same input string to the system.

- Failure mode and their possibility
 - Hardware (especially HDD) will break
 - Filesystem may be corrupted and data in it may be lost
 - Remote storage system can't be trusted for security breaches
 - Weak password protection can be easily compromised
 - File permission system may be compromised
- Required resources for backup: human, hardware, software, ...
 - Automatic scheduled backup with cron job or systemd timer job

Indicație

You can recover debconf configuration data with `"debconf-set-selections debconf-selections"` and dpkg selection data with `"dpkg --set-selection <dpkg-selections.list"`.

Notă

Do not back up the pseudo-filesystem contents found on `/proc`, `/sys`, `/tmp`, and `/run` (see Secțiune 1.2.12 and Secțiune 1.2.13). Unless you know exactly what you are doing, they are huge useless data.

Notă

You may wish to stop some application daemons such as MTA (see Secțiune 6.2.4) while backing up data.

10.2.2 Backup utility suites

Here is a select list of notable backup utility suites available on the Debian system.

Backup tools have their specialized focuses.

- [Mondo Rescue](#) is a backup system to facilitate restoration of complete system quickly from backup CD/DVD etc. without going through normal system installation processes.
- [Bacula](#), [Amanda](#), and [BackupPC](#) are full featured backup suite utilities which are focused on regular backups over network.
- [Duplicity](#), and [Borg](#) are simpler backup utilities for typical workstations.

10.2.3 Backup tips

For a personal workstation, full featured backup suite utilities designed for the server environment may not serve well. At the same time, existing backup utilities for workstations may have some shortcomings.

Here are some tips to make backup easier with minimal user efforts. These techniques may be used with any backup utilities.

For demonstration purpose, let's assume the primary user and group name to be `penguin` and create a backup and snapshot script example `"/usr/local/bin/bkss.sh"` as:

```
#!/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
```

pachet	popcon	(populație)	descriere
bacula-common	V:6, I:7	2501	Bacula : network backup, recovery and verification - common support files
bacula-client	V:0, I:2	199	Bacula : network backup, recovery and verification - client meta-package
bacula-console	V:0, I:2	112	Bacula : network backup, recovery and verification - text console
bacula-server	I:0	199	Bacula : network backup, recovery and verification - server meta-package
amanda-common	V:0, I:2	9851	Amanda : Advanced Maryland Automatic Network Disk Archiver (Libs)
amanda-client	V:0, I:2	1099	Amanda : Advanced Maryland Automatic Network Disk Archiver (Client)
amanda-server	V:0, I:0	1093	Amanda : Advanced Maryland Automatic Network Disk Archiver (Server)
backuppc	V:1, I:1	3088	BackupPC is a high-performance, enterprise-grade system for backing up PCs (disk based)
duplicity	V:6, I:50	2649	(remote) incremental backup
deja-dup	V:30, I:45	5031	GUI frontend for duplicity
borgbackup	V:12, I:28	3478	(remote) deduplicating backup
borgmatic	V:3, I:4	946	borgbackup helper
rdiff-backup	V:2, I:7	1207	(remote) incremental backup
restic	V:5, I:10	24708	(remote) incremental backup
backupninja	V:2, I:2	360	lightweight, extensible meta-backup system
slbackup	V:0, I:0	147	(remote) incremental backup
backup-manager	V:0, I:0	573	command-line backup tool
backup2l	V:0, I:0	110	low-maintenance backup/restore tool for mountable media (disk based)

Tabela 10.5: List of backup suite utilities


```

    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!"

```

Here, only the basic tool `rsync(1)` is used to facilitate system backup and the storage space is efficiently used by [Btrfs](#).

Indicație

FYI: This author uses his own similar shell script "[bss: Btrfs Subvolume Snapshot Utility](#)" for his workstation.

10.2.3.1 GUI backup

Here is an example to setup the single GUI click backup.

- Prepare a USB storage device to be used for backup.
 - Format a USB storage device with one partition in btrfs with its label name as "BKUP". This can be encrypted (see [Secțiune 9.9.1](#)).
 - Plug this in to your system. The desktop system should automatically mount it as `/media/penguin/BKUP`.
 - Execute `sudo chown penguin:penguin /media/penguin/BKUP` to make it writable by the user.
- Create `~/ .local/share/applications/BKUP.desktop` following techniques written in [Secțiune 9.4.10](#) as:

```

[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

```

For each GUI click, your data is backed up from `~/Documents` to a USB storage device and a read-only snapshot is created.

10.2.3.2 Mount event triggered backup

Here is an example to setup for the automatic backup triggered by the mount event.

- Prepare a USB storage device to be used for backup as in [Secțiune 10.2.3.1](#).
- Create a systemd service unit file `~/ .config/systemd/user/back-BKUP.service` as:

```

[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
```

- Enable this systemd unit configuration with the following:

```
$ systemctl --user enable bkup-BKUP.service
```

For each mount event, your data is backed up from “~/Documents” to a USB storage device and a read-only snapshot is created.

Here, names of systemd mount units that systemd currently has in memory can be asked to the service manager of the calling user with “systemctl --user list-units --type=mount”.

10.2.3.3 Timer event triggered backup

Here is an example to setup for the automatic backup triggered by the timer event.

- Prepare a USB storage device to be used for backup as in Secțiune [10.2.3.1](#).
- Create a systemd timer unit file “~/ .config/systemd/user/snap-Documents.timer” as:

```
[Unit]
Description=Run btrfs subvolume snapshot on timer
Documentation=man:btrfs(1)

[Timer]
OnStartupSec=30
OnUnitInactiveSec=900

[Install]
WantedBy=timers.target
```

- Create a systemd service unit file “~/ .config/systemd/user/snap-Documents.service” as:

```
[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
```

- Enable this systemd unit configuration with the following:

```
$ systemctl --user enable snap-Documents.timer
```

For each timer event, your data is backed up from “~/Documents” to a USB storage device and a read-only snapshot is created.

Here, names of systemd timer user units that systemd currently has in memory can be asked to the service manager of the calling user with “systemctl --user list-units --type=timer”.

For the modern desktop system, this systemd approach can offer more fine grained control than the traditional Unix ones using at(1), cron(8), or anacron(8).

10.3 Data security infrastructure

The data security infrastructure is provided by the combination of data encryption tool, message digest tool, and signature tool.

pachet	populare (popularity)	dimensiune (size)	comanda (command)	descriere (description)
gnupg	V:367, I:872	468	gpg(1)	GNU Privacy Guard - OpenPGP encryption and signing tool
gpgv	V:258, I:955	559	gpgv(1)	GNU Privacy Guard - OpenPGP signature verification tool
sq	V:0, I:17	22408	sq(1)	Sequoia-PGP - OpenPGP encryption and signing tool
sqv	V:324, I:406	1813	sqv(1)	Sequoia-PGP - OpenPGP signature verification tool
paperkey	V:1, I:13	58	paperkey(1)	extract just the secret information out of OpenPGP secret keys
cryptsetup	V:30, I:80	465	cryptsetup(8) ...	utilities for dm-crypt block device encryption supporting LUKS
coreutils	V:892, I:999	18457	md5sum(1)	compute and check MD5 message digest
coreutils	V:892, I:999	18457	sha1sum(1)	compute and check SHA1 message digest
openssl	V:834, I:995	2503	openssl(1ssl)	compute message digest with "openssl dgst" (OpenSSL)
libsecret-tools	V:0, I:9	49	secret-tool(1)	store and retrieve passwords (CLI)
seahorse	V:81, I:273	7971	seahorse(1)	key management tool (GNOME)

Tabela 10.6: List of data security infrastructure tools

See Secțiune 9.9 on [dm-crypt](#) and [fscrypt](#) which implement automatic data encryption infrastructure via Linux kernel modules.

10.3.1 Key management for GnuPG

Here are [GNU Privacy Guard](#) commands for the basic key management.

Here is the meaning of the trust code.

The following generates my key "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
```

comanda	descriere
gpg --gen-key	generează o cheie nouă
gpg --gen-revoke ID-ul_utilizatorului_meu	generate revoke key for my_user_ID
gpg --edit-key user_ID	edit key interactively, "help" for help
gpg -o fișier --export	export all public keys to file
gpg -o fișier --export-secret-keys	export all private key to file
gpg --import fișier	import all keys from file
gpg --send-keys ID-utilizator	send key of user_ID to keyserver
gpg --recv-keys ID-utilizator	recv. key of user_ID from keyserver
gpg --list-keys ID-utilizator	list keys of user_ID
gpg --list-sigs ID-utilizator	list sig. of user_ID
gpg --check-sigs ID-utilizator	check sig. of user_ID
gpg --fingerprint ID-utilizator	check fingerprint of user_ID
gpg --refresh-keys	update local keyring

Tabela 10.7: List of GNU Privacy Guard commands for the key management

cod	description of trust
-	no owner trust assigned / not yet calculated
e	trust calculation failed
q	not enough information for calculation
n	never trust this key
m	marginally trusted
f	fully trusted
u	ultimately trusted

Tabela 10.8: List of the meaning of the trust code

```
uid          Osamu Aoki <osamu@debian.org>
sub   cv25519 2026-02-14 [E] [expires: 2029-02-13]
```

The following uploads my key "9563FC29932C409F1A77F9C1AB8A1522D8234C6A" to the popular keyserver "hkp://key

```
$ gpg --keyserver hkp://keys.gnupg.net --send-keys 9563FC29932C409F1A77F9C1AB8A1522D8234C6A
```

A good default keyserver set up in "~/.gnupg/gpg.conf" (or old location "~/.gnupg/options") contains the following.

```
keyserver hkp://keys.gnupg.net
```

The following obtains unknown keys from the keyserver.

```
$ gpg --list-sigs --with-colons | grep '^sig.*\[User ID not found\]' | \
    cut -d ':' -f 5 | sort | uniq | xargs gpg --recv-keys
```

There was a bug in [OpenPGP Public Key Server](#) (pre version 0.9.6) which corrupted key with more than 2 sub-keys. The newer gnupg (>1.2.1-2) package can handle these corrupted subkeys. See [gpg\(1\)](#) under "-repair-pks-subkey-bu" option.

Use of SHA-1 for hash has been deprecated. If your old RSA based openPGP key uses SHA-1 for hash, fix it using [FixKeyWithSha1](#).

Indicație

The [sq\(1\)](#) and [sqv\(1\)](#) commands are an alternative set of openPGP commands. See [sq user documentation](#) and [A Practical Introduction to using sq, Sequoia PGP's CLI](#).

10.3.2 Using GnuPG on files

Here are examples for using [GNU Privacy Guard](#) commands on files.

10.3.3 Using GnuPG with Mutt

Add the following to "~/.muttrc" to keep a slow GnuPG from automatically starting, while allowing it to be used by typing "S" at the index menu.

```
macro index S ":toggle pgp_verify_sig\n"
set pgp_verify_sig=no
```

10.3.4 Using GnuPG with Vim

The gnupg plugin let you run GnuPG transparently for files with extension ".pgp", ".asc", and ".pgp".[3](#)

```
$ sudo aptitude install vim-scripts
$ echo "packadd! gnupg" >> ~/.vim/vimrc
```

³If you use "~/.vimrc" instead of "~/.vim/vimrc", please substitute accordingly.

comanda	descriere
<code>gpg -a -s fișier</code>	sign file into ASCII armored file.asc
<code>gpg --armor --sign fișier</code>	, ,
<code>gpg --clearsign fișier</code>	clear-sign message
<code>gpg --clearsign file mail foo@example.org</code>	mail a clear-signed message to foo@example.org
<code>gpg --clearsign --not-dash-escaped patchfile</code>	clear-sign patchfile
<code>gpg --verify fișier</code>	verify clear-signed file
<code>gpg -o file.sig -b file</code>	create detached signature
<code>gpg -o file.sig --detach-sign file</code>	, ,
<code>gpg --verify file.sig file</code>	verify file with file.sig
<code>gpg -o crypt_file.gpg -r name -e file</code>	public-key encryption intended for name from file to binary crypt_file.gpg
<code>gpg -o crypt_file.gpg --recipient name --encrypt file</code>	, ,
<code>gpg -o crypt_file.asc -a -r name -e file</code>	public-key encryption intended for name from file to ASCII armored crypt_file.asc
<code>gpg -o crypt_file.gpg -c file</code>	symmetric encryption from file to crypt_file.gpg
<code>gpg -o crypt_file.gpg --symmetric file</code>	, ,
<code>gpg -o crypt_file.asc -a -c file</code>	symmetric encryption intended for name from file to ASCII armored crypt_file.asc
<code>gpg -o file -d crypt_file.gpg -r name</code>	decriptare
<code>gpg -o file --decrypt crypt_file.gpg</code>	, ,

Tabela 10.9: List of GNU Privacy Guard commands on files

10.3.5 Suma MD5

md5sum(1) provides utility to make a digest file using the method in [rfc1321](#) and verifying each file with it.

```
$ md5sum foo bar >baz.md5
$ cat baz.md5
d3b07384d113edec49eaa6238ad5ff00  foo
c157a79031e1c40f85931829bc5fc552  bar
$ md5sum -c baz.md5
foo: OK
bar: OK
```

Notă

The computation for the [MD5](#) sum is less CPU intensive than the one for the cryptographic signature by [GNU Privacy Guard \(GnuPG\)](#). Usually, only the top level digest file is cryptographically signed to ensure data integrity.

10.3.6 Password keyring

On GNOME system, the GUI tool `seahorse(1)` manages passwords and stores them securely in the keyring `~/.local/share/secret-tool(1)` can store password to the keyring from the command line.

Let's store passphrase used for LUKS/dm-crypt encrypted disk image

```
$ secret-tool store --label='LUKS passphrase for disk.img' LUKS my_disk.img
Password: *****
```

This stored password can be retrieved and fed to other programs, e.g., `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

Whenever you need to provide password in a script, use `secret-tool` and avoid directly hardcoding the passphrase in it.

10.4 Source code merge tools

There are many merge tools for the source code. Following commands caught my eyes.

10.4.1 Extracting differences for source files

The following procedures extract differences between two source files and create unified diff files "file.patch0" or "file.patch1" depending on the file location.

```
$ diff -u file.old file.new > file.patch0
$ diff -u old/file new/file > file.patch1
```

pachet	popcon(popularity)	dimensiune	comanda	descriere
patch	V:97, I:712	242	<code>patch(1)</code>	apply a diff file to an original
vim	V:86, I:346	4089	<code>vimdiff(1)</code>	compare 2 files side by side in vim
imediff	V:0, I:0	348	<code>imediff(1)</code>	interactive full screen 2/3-way merge tool
meld	V:5, I:25	3546	<code>meld(1)</code>	compare and merge files (GTK)
wiggle	V:0, I:0	175	<code>wiggle(1)</code>	apply rejected patches
diffutils	V:875, I:998	1768	<code>diff(1)</code>	compare files line by line
diffutils	V:875, I:998	1768	<code>diff3(1)</code>	compare and merges three files line by line
quilt	V:1, I:18	880	<code>quilt(1)</code>	manage series of patches
wdiff	V:6, I:42	651	<code>wdiff(1)</code>	display word differences between text files
diffstat	V:10, I:104	79	<code>diffstat(1)</code>	produce a histogram of changes by the diff
patchutils	V:13, I:103	242	<code>combinediff(1)</code>	create a cumulative patch from two incremental patches
patchutils	V:13, I:103	242	<code>dehtmldiff(1)</code>	extract a diff from an HTML page
patchutils	V:13, I:103	242	<code>filterdiff(1)</code>	extract or excludes diffs from a diff file
patchutils	V:13, I:103	242	<code>fixcvsdiff(1)</code>	fix diff files created by CVS that <code>patch(1)</code> mis-interprets
patchutils	V:13, I:103	242	<code>flipdiff(1)</code>	exchange the order of two patches
patchutils	V:13, I:103	242	<code>grepdiff(1)</code>	show which files are modified by a patch matching a regex
patchutils	V:13, I:103	242	<code>interdiff(1)</code>	show differences between two unified diff files
patchutils	V:13, I:103	242	<code>lsdiff(1)</code>	show which files are modified by a patch
patchutils	V:13, I:103	242	<code>recountdiff(1)</code>	recompute counts and offsets in unified context diffs
patchutils	V:13, I:103	242	<code>rediff(1)</code>	fix offsets and counts of a hand-edited diff
patchutils	V:13, I:103	242	<code>splitdiff(1)</code>	separate out incremental patches
patchutils	V:13, I:103	242	<code>unwrapdiff(1)</code>	demangle patches that have been word-wrapped
dirdiff	V:0, I:1	167	<code>dirdiff(1)</code>	display differences and merge changes between directory trees
docdiff	V:0, I:0	554	<code>docdiff(1)</code>	compare two files word by word / char by char
makepatch	V:0, I:0	99	<code>makepatch(1)</code>	generate extended patch files
makepatch	V:0, I:0	99	<code>applypatch(1)</code>	apply extended patch files

Tabela 10.10: List of source code merge tools

10.4.2 Merging updates for source files

The diff file (alternatively called patch file) is used to send a program update. The receiving party applies this update to another file by the following.

```
$ patch -p0 file < file.patch0
$ patch -p1 file < file.patch1
```

10.4.3 Interactive merge

If you have two versions of a source code, you can perform 2-way merge interactively using `imediff(1)` by the following.

```
$ imediff -o file.merged file.old file.new
```

If you have three versions of a source code, you can perform 3-way merge interactively using `imediff(1)` by the following.

```
$ imediff -o file.merged file.yours file.base file.theirs
```

10.5 Git

Git is the tool of choice these days for the [version control system \(VCS\)](#) since Git can do everything for both local and remote source code management.

Debian provides free Git services via [Debian Salsa service](#). Its documentation can be found at <https://wiki.debian.org/Salsa>.

Here are some Git related packages.

pachet	popcon	popularity	comanda	descriere
git	V:382, I:597	50172	git(7)	Git, the fast, scalable, distributed revision control system
gitk	V:4, I:29	2003	gitk(1)	GUI Git repository browser with history
git-gui	V:1, I:18	2508	git-gui(1)	GUI for Git (No history)
git-email	V:0, I:10	1187	git-send-email(1)	collection of patches as email from the Git
git-buildpackage	V:1, I:7	2030	git-buildpackage(1)	the Debian packaging with the Git
dgit	V:0, I:1	724	dgit(1)	git interoperability with the Debian archive
imediff	V:0, I:0	348	git-ime(1)	interactive git commit split helper tool
stgit	V:0, I:0	604	stg(1)	quilt on top of git (Python)
git-doc	I:11	14238	N/D	official documentation for Git
gitmagic	I:0	721	N/D	"Git Magic", easier to understand guide for Git

Tabela 10.11: List of git related packages and commands

10.5.1 Configuration of Git client

You may wish to set several global configuration in "`~/.gitconfig`" such as your name and email address used by Git by the following.

```
$ git config --global user.name "Name Surname"
$ git config --global user.email yourname@example.com
```

You may also customize the Git default behavior by the following.

```
$ git config --global init.defaultBranch main
$ git config --global pull.rebase true
$ git config --global push.default current
```

If you are too used to CVS or Subversion commands, you may wish to set several command aliases by the following.

```
$ git config --global alias.ci "commit -a"
$ git config --global alias.co checkout
```

You can check your global configuration by the following.

```
$ git config --global --list
```

10.5.2 Basic Git commands

Git operation involves several data.

- The working tree which holds user facing files and to which you make changes.
 - The changes to be recorded must be explicitly selected and staged to the index. This is `git add` and `git rm` commands.
- The index which holds staged files.
 - Staged files will be committed to the local repository upon the subsequent request. This is `git commit` command.
- The local repository which holds committed files.
 - Git records the linked history of the committed data and organizes them as branches in the repository.
 - The local repository can send data to the remote repository by `git push` command.
 - The local repository can receive data from the remote repository by `git fetch` and `git pull` commands.
 - * The `git pull` command performs `git merge` or `git rebase` command after `git fetch` command.
 - * Here, `git merge` combines two separate branches of history at the end to a point. (This is default of `git pull` without customization and may be good for upstream people who publish branch to many people.)
 - * Here, `git rebase` creates one single branch of sequential history of the remote branch one followed by the local branch one. (This is `pull.rebase true` customization case and may be good for rest of us.)
- The remote repository which holds committed files.
 - The communication to the remote repository uses secure communication protocols such as SSH or HTTPS.

The working tree is files outside of the `.git/` directory. Files inside of the `.git/` directory hold the index, the local repository data, and some git configuration text files.

Here is an overview of main Git commands.

Comanda git	funcție
git init	create the (local) repository
git clone URL	clone the remote repository to a local repository with the working tree
git pull origin main	update the local main branch by the remote repository origin
git add .	add file(s) in the working tree to the index for pre-existing files in index only
git add -A .	add file(s) in the working tree to the index for all files including removals
git rm nume-fișier	remove file(s) from the working tree and the index
git commit	commit staged changes in the index to the local repository
git commit -a	add all changes in the working tree to the index and commit them to the local repository (add + commit)
git push -u origin branch_name	update the remote repository origin by the local branch_name branch (initial invocation)
git push origin branch_name	update the remote repository origin by the local branch_name branch (subsequent invocation)
git diff treeish1 treeish2	show difference between <i>treeish1</i> commit and <i>treeish2</i> commit
gitk	GUI display of VCS repository branch history tree

Tabela 10.12: Main Git commands

10.5.3 Git tips

Here are some Git tips.



Avertisment

Do not use the tag string with spaces in it even if some tools such as `gitk(1)` allow you to use it. It may choke some other git commands.



Atenție

If a local branch which has been pushed to remote repository is rebased or squashed, pushing this branch has risks and requires `--force` option. This is usually not an acceptable for main branch but may be acceptable for a topic branch before merging to main branch.



Atenție

Invoking a git subcommand directly as `"git-xyz"` from the command line has been deprecated since early 2006.

Indicație

If there is a executable file `git-foo` in the path specified by `$PATH`, entering `"git foo"` without hyphen to the command line invokes this `git-foo`. This is a feature of the git command.

10.5.4 Git references

Consultați următoarele.

Git command line	funcție
<code>gitk --all</code>	see complete Git history and operate on them such as resetting HEAD to another commit, cheery-picking patches, creating tags and branches ...
<code>git stash</code>	get the clean working tree without loosing data
<code>git remote -v</code>	check settings for remote
<code>git branch -vv</code>	check settings for branch
<code>git status</code>	show working tree status
<code>git config -l</code>	list git settings
<code>git reset --hard HEAD; git clean -x -d -f</code>	revert all working tree changes and clean them up completely
<code>git rm --cached nume-fișier</code>	revert staged index changed by <code>git add filename</code>
<code>git reflog</code>	get reference log (useful for recovering commits from the removed branch)
<code>git branch new_branch_name HEAD@{6}</code>	create a new branch from reflog information
<code>git remote add new_remote URL</code>	add a new_remote remote repository pointed by URL
<code>git remote rename origin upstream</code>	rename the remote repository name from origin to upstream
<code>git branch -u upstream/branch_name</code>	set the remote tracking to the remote repository upstream and its branch name branch_name.
<code>git remote set-url origin https://foo/bar.git</code>	change URL of origin
<code>git remote set-url --push upstream DISABLED</code>	disable push to upstream (Edit .git/config to re-enable)
<code>git remote update upstream</code>	fetch updates of all remote branches in the upstream repository
<code>git fetch upstream foo:upstream-foo</code>	create a local (possibly orphan) upstream-foo branch as a copy of foo branch in the upstream repository
<code>git checkout -b topic_branch ; git push -u topic_branch origin</code>	make a new topic_branch and push it to origin
<code>git branch -m oldname newname</code>	rename local branch name
<code>git push -d origin branch_to_be_removed</code>	remove remote branch (new method)
<code>git push origin :branch_to_be_removed</code>	remove remote branch (old method)
<code>git checkout --orphan unconnected</code>	create a new unconnected branch
<code>git rebase -i origin/main</code>	reorder/drop/squish commits from origin/main to clean branch history
<code>git reset HEAD^; git commit --amend</code>	squash last 2 commits into one
<code>git checkout topic_branch ; git merge --squash topic_branch</code>	squash entire topic_branch into a commit
<code>git fetch --unshallow --update-head-ok origin '+refs/heads/*:refs/heads/*'</code>	convert a shallow clone to the full clone of all branches
<code>git ime</code>	split the last commit into a series of file-by-file smaller commits etc. (imdiff package required)
<code>git repack -a -d; git prune</code>	repack the local repository into single pack (this may limit chance of lost data recovery from erased branch etc.)

Tabela 10.13: Git tips

- [pagina de manual: git\(1\)](/usr/share/doc/git-doc/git.html) (/usr/share/doc/git-doc/git.html)
- [Manualul utilizatorului git](/usr/share/doc/git-doc/user-manual.html) (/usr/share/doc/git-doc/user-manual.html)
- [A tutorial introduction to git](/usr/share/doc/git-doc/gittutorial.html) (/usr/share/doc/git-doc/gittutorial.html)
- [A tutorial introduction to git: part two](/usr/share/doc/git-doc/gittutorial-2.html) (/usr/share/doc/git-doc/gittutorial-2.html)
- [Everyday GIT With 20 Commands Or So](/usr/share/doc/git-doc/giteveryday.html) (/usr/share/doc/git-doc/giteveryday.html)
- [Git Magic](/usr/share/doc/gitmagic/html/index.html) (/usr/share/doc/gitmagic/html/index.html)

10.5.5 Other version control systems

The [version control systems \(VCS\)](#) is sometimes known as the revision control system (RCS), or the software configuration management (SCM).

Here is a summary of the notable other non-Git VCS on the Debian system.

pachet	popcon	popularity	instrument	VCS type	comentariu
mercurial	V:3, I:26	2579	Mercurial	distributed	DVCS in Python and some C
darcs	V:0, I:3	38784	Darcs	distributed	DVCS with smart algebra of patches (slow)
bazaar	I:4	28	GNU Bazaar	distributed	DVCS influenced by t l a written in Python (historic)
tla	V:0, I:0	1022	GNU arch	distributed	DVCS mainly by Tom Lord (historic)
subversion	V:10, I:59	4849	Subversion	remote	"CVS done right", newer standard remote VCS (historic)
cvs	V:3, I:26	4835	CVS	remote	previous standard remote VCS (historic)
tkcvs	V:0, I:0	34	CVS, ...	remote	GUI display of VCS (CVS, Subversion, RCS) repository tree
rcs	V:1, I:9	578	RCS	local	" Unix SCCS done right" (historic)
cssc	V:0, I:0	2044	CSSC	local	clone of the Unix SCCS (historic)

Tabela 10.14: List of other version control system tools

Capitolul 11

Data conversion

Tools and tips for converting data formats on the Debian system are described.

Standard based tools are in very good shape but support for proprietary data formats are limited.

11.1 Text data conversion tools

Following packages for the text data conversion caught my eyes.

pachet	popcon(popularity)	limita (limit)	keyword	descriere
libc6	V:922, I:999	5370	charset	text encoding converter between locales by <code>iconv(1)</code> (fundamental)
recode	V:2, I:13	528	charset+eol	text encoding converter between locales (versatile, more aliases and features)
konwert	V:1, I:43	137	charset	text encoding converter between locales (fancy)
nkf	V:0, I:8	359	charset	character set translator for Japanese
tcs	V:0, I:0	518	charset	character set translator
unaccent	V:0, I:0	34	charset	replace accented letters by their unaccented equivalent
tofromdos	V:0, I:12	50	eol	text format converter between DOS and Unix: <code>fromdos(1)</code> and <code>todos(1)</code>
macutils	V:0, I:0	319	eol	text format converter between Macintosh and Unix: <code>frommac(1)</code> and <code>tomac(1)</code>

Tabela 11.1: List of text data conversion tools

11.1.1 Converting a text file with `iconv`

Indicație

`iconv(1)` is provided as a part of the `libc6` package and it is always available on practically all Unix-like systems to convert the encoding of characters.

You can convert encodings of a text file with `iconv(1)` by the following.

```
$ iconv -f encoding1 -t encoding2 input.txt >output.txt
```

encoding value	utilizare
ASCII	American Standard Code for Information Interchange , 7 bit code w/o accented characters
UTF-8	current multilingual standard for all modern OSs
ISO-8859-1	old standard for western European languages, ASCII + accented characters
ISO-8859-2	old standard for eastern European languages, ASCII + accented characters
ISO-8859-15	old standard for western European languages, ISO-8859-1 with euro sign
CP850	code page 850, Microsoft DOS characters with graphics for western European languages, ISO-8859-1 variant
CP932	code page 932, Microsoft Windows style Shift-JIS variant for Japanese
CP936	code page 936, Microsoft Windows style GB2312 , GBK or GB18030 variant for Simplified Chinese
CP949	code page 949, Microsoft Windows style EUC-KR or Unified Hangul Code variant for Korean
CP950	code page 950, Microsoft Windows style Big5 variant for Traditional Chinese
CP1251	code page 1251, Microsoft Windows style encoding for the Cyrillic alphabet
CP1252	code page 1252, Microsoft Windows style ISO-8859-15 variant for western European languages
KOI8-R	old Russian UNIX standard for the Cyrillic alphabet
ISO-2022-JP	standard encoding for Japanese email which uses only 7 bit codes
eucJP	old Japanese UNIX standard 8 bit code and completely different from Shift-JIS
Shift-JIS	JIS X 0208 Appendix 1 standard for Japanese (see CP932)

Tabela 11.2: List of encoding values and their usage

Encoding values are case insensitive and ignore "-" and "_" for matching. Supported encodings can be checked by the "iconv -l" command.

Notă

Some encodings are only supported for the data conversion and are not used as locale values (Secțiune 8.1).

For character sets which fit in single byte such as [ASCII](#) and [ISO-8859](#) character sets, the [character encoding](#) means almost the same thing as the character set.

For character sets with many characters such as [JIS X 0213](#) for Japanese or [Universal Character Set \(UCS, Unicode, ISO-10646-1\)](#) for practically all languages, there are many encoding schemes to fit them into the sequence of the byte data.

- [EUC](#) and [ISO/IEC 2022 \(also known as JIS X 0202\)](#) for Japanese
- [UTF-8](#), [UTF-16/UCS-2](#) and [UTF-32/UCS-4](#) for Unicode

For these, there are clear differentiations between the character set and the character encoding.

The [code page](#) is used as the synonym to the character encoding tables for some vendor specific ones.

Notă

Please note most encoding systems share the same code with ASCII for the 7 bit characters. But there are some exceptions. If you are converting old Japanese C programs and URLs data from the casually-called shift-JIS encoding format to UTF-8 format, use "CP932" as the encoding name instead of "shift - JIS" to get the expected results: 0x5C → "\" and 0x7E → "~". Otherwise, these are converted to wrong characters.

Indicație

recode(1) may be used too and offers more than the combined functionality of iconv(1), fromdos(1), todos(1), frommac(1), and tomac(1). For more, see "info recode".

11.1.2 Checking file to be UTF-8 with iconv

You can check if a text file is encoded in UTF-8 with iconv(1) by the following.

```
$ iconv -f utf8 -t utf8 input.txt >/dev/null || echo "non-UTF-8 found"
```

Indicație

Use "- -verbose" option in the above example to find the first non-UTF-8 character.

11.1.3 Converting file names with iconv

Here is an example script to convert encoding of file names from ones created under older OS to modern UTF-8 ones in a single directory.

```
#!/bin/sh
ENCDN=iso-8859-1
for x in *;
do
  mv "$x" "$(echo "$x" | iconv -f $ENCDN -t utf-8)"
done
```


The "\$ENCDN" variable specifies the original encoding used for file names under older OS as in Tabel 11.2.

For more complicated case, please mount a filesystem (e.g. a partition on a disk drive) containing such file names with proper encoding as the mount(8) option (see Secțiune 8.1.3) and copy its entire contents to another filesystem mounted as UTF-8 with "cp -a" command.

11.1.4 EOL conversion

The text file format, specifically the end-of-line (EOL) code, is dependent on the platform.

platforma	EOL code	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: List of EOL styles for different platforms

The EOL format conversion programs, fromdos(1), todos(1), frommac(1), and tomac(1), are quite handy. recode(1) is also useful.

Notă

Some data on the Debian system, such as the wiki page data for the python-moinmoin package, use MSDOS style CR-LF as the EOL code. So the above rule is just a general rule.

Notă

Most editors (eg. vim, emacs, gedit, ...) can handle files in MSDOS style EOL transparently.

Indicație

The use of "sed -e '/\r\$/!s/\$/\r/'" instead of todos(1) is better when you want to unify the EOL style to the MSDOS style from the mixed MSDOS and Unix style. (e.g., after merging 2 MSDOS style files with diff3(1).) This is because todos adds CR to all lines.

11.1.5 TAB conversion

There are few popular specialized programs to convert the tab codes.

funcție	bsdmainutils	coreutils
expand tab to spaces	"col -x"	expand
unexpand tab from spaces	"col -h"	unexpand

Tabela 11.4: List of TAB conversion commands from bsdmainutils and coreutils packages

indent(1) from the indent package completely reformats whitespaces in the C program.

Editor programs such as vim and emacs can be used for TAB conversion, too. For example with vim, you can expand TAB with ":set expandtab" and ":%retab" command sequence. You can revert this with ":set noexpandtab" and ":%retab!" command sequence.

11.1.6 Editors with auto-conversion

Intelligent modern editors such as the `vim` program are quite smart and copes well with any encoding systems and any file formats. You should use these editors under the UTF-8 locale in the UTF-8 capable console for the best compatibility.

An old western European Unix text file, "`u-file.txt`", stored in the `latin1` (`iso-8859-1`) encoding can be edited simply with `vim` by the following.

```
$ vim u-file.txt
```

This is possible since the auto detection mechanism of the file encoding in `vim` assumes the UTF-8 encoding first and, if it fails, assumes it to be `latin1`.

An old Polish Unix text file, "`pu-file.txt`", stored in the `latin2` (`iso-8859-2`) encoding can be edited with `vim` by the following.

```
$ vim '+e ++enc=latin2 pu-file.txt'
```

An old Japanese unix text file, "`ju-file.txt`", stored in the `eucJP` encoding can be edited with `vim` by the following.

```
$ vim '+e ++enc=eucJP ju-file.txt'
```

An old Japanese MS-Windows text file, "`jw-file.txt`", stored in the so called `shift-JIS` encoding (more precisely: `CP932`) can be edited with `vim` by the following.

```
$ vim '+e ++enc=CP932 ++ff=dos jw-file.txt'
```

When a file is opened with "`++enc`" and "`++ff`" options, "`:w`" in the Vim command line stores it in the original format and overwrite the original file. You can also specify the saving format and the file name in the Vim command line, e.g., "`:w ++enc=utf8 new.txt`".

Please refer to the `mbyte.txt` "multi-byte text support" in `vim` on-line help and Tabel [11.2](#) for locale values used with "`++enc`".

The `emacs` family of programs can perform the equivalent functions.

11.1.7 Plain text extraction

The following reads a web page into a text file. This is very useful when copying configurations off the Web or applying basic Unix text tools such as `grep(1)` on the web page.

```
$ w3m -dump https://www.remote-site.com/help-info.html >textfile
```

Similarly, you can extract plain text data from other formats using the following.

11.1.8 Highlighting and formatting plain text data

You can highlight and format plain text data by the following.

11.2 XML data

The [Extensible Markup Language \(XML\)](#) is a markup language for documents containing structured information.

See introductory information at [XML.COM](#).

pachet	popcon(popularitate)	dimensiune	keyword	funcție
w3m	V:11, I:145	2853	html → text	HTML to text converter with the "w3m -dump" command
html2text	V:3, I:71	298	html → text	advanced HTML to text converter (ISO 8859-1)
lynx	V:29, I:457	1972	html → text	HTML to text converter with the "lynx -dump" command
elinks	V:2, I:16	1791	html → text	HTML to text converter with the "elinks -dump" command
links	V:2, I:21	2321	html → text	HTML to text converter with the "links -dump" command
links2	V:0, I:10	5466	html → text	HTML to text converter with the "links2 -dump" command
catdoc	V:17, I:176	682	MSWord → text	Convert MSWord files to plain text or TeX
antiword	V:0, I:6	587	MSWord → text	convert MSWord files to plain text or ps
unhtml	V:0, I:0	40	html → text	remove the markup tags from an HTML file
odt2txt	V:1, I:25	60	odt → text	converter from OpenDocument Text to text

Tabela 11.5: List of tools to extract plain text data

pachet	popcon(popularitate)	dimensiune	keyword	descriere
vim-runtime	V:16, I:365	38132	highlight	Vim MACRO to convert source code to HTML with " :source \$VIMRUNTIME/syntax/html.vim"
cxref	V:0, I:0	1191	c → html	converter for the C program to latex and HTML (C language)
src2tex	V:0, I:0	1799	highlight	convert many source codes to TeX (C language)
source-highlight	V:0, I:3	2131	highlight	convert many source codes to HTML, XHTML, LaTeX, Texinfo, ANSI color escape sequences and DocBook files with highlight (C++)
highlight	V:0, I:3	1411	highlight	convert many source codes to HTML, XHTML, RTF, LaTeX, TeX or XSL-FO files with highlight (C++)
grc	V:0, I:5	208	text → culoare	generic colouriser for everything (Python)
pandoc	V:10, I:47	201782	text → orice	general markup converter (Haskell)
python3-docutils	V:13, I:52	2009	text → orice	ReStructured Text document formatter to XML (Python)
markdown	V:0, I:6	56	text → html	Markdown text document formatter to (X)HTML (Perl)
asciidoc	V:0, I:5	101	text → orice	AsciiDoc text document formatter to XML/HTML (Ruby)
python3-sphinx	V:7, I:27	2996	text → orice	ReStructured Text based document publication system (Python)
hugo	V:0, I:5	61651	text → html	Markdown based static site publication system (Go)

Tabela 11.6: List of tools to highlight plain text data

- ["What is XML?"](#)
- ["What Is XSLT?"](#)
- ["What Is XSL-FO?"](#)
- ["What Is XLink?"](#)

11.2.1 Basic hints for XML

XML text looks somewhat like [HTML](#). It enables us to manage multiple formats of output for a document. One easy XML system is the docbook-xsl package, which is used here.

Each XML file starts with standard XML declaration as the following.

```
<?xml version="1.0" encoding="UTF-8"?>
```

The basic syntax for one XML element is marked up as the following.

```
<name attribute="value">content</name>
```

XML element with empty content is marked up in the following short form.

```
<name attribute="value" />
```

The "attribute="value"" in the above examples are optional.

The comment section in XML is marked up as the following.

```
<!-- comment -->
```

Other than adding markups, XML requires minor conversion to the content using predefined entities for following characters.

predefined entity	character to be converted into
"	" : quote
'	' : apostrophe
<	< : less-than
>	> : greater-than
&	& : ampersand

Tabela 11.7: List of predefined entities for XML



Atenție

"<" or "&" can not be used in attributes or elements.

Notă

When SGML style user defined entities, e.g. "&some-tag;", are used, the first definition wins over others. The entity definition is expressed in "<!ENTITY some-tag "entity value">".

Notă

As long as the XML markup are done consistently with certain set of the tag name (either some data as content or attribute value), conversion to another XML is trivial task using [Extensible Stylesheet Language Transformations \(XSLT\)](#).

11.2.2 XML processing

There are many tools available to process XML files such as [the Extensible Stylesheet Language \(XSL\)](#).

Basically, once you create well formed XML file, you can convert it to any format using [Extensible Stylesheet Language Transformations \(XSLT\)](#).

The [Extensible Stylesheet Language for Formatting Objects \(XSL-FO\)](#) is supposed to be solution for formatting. The fop package is new to the Debian main archive due to its dependence to the [Java programming language](#). So the LaTeX code is usually generated from XML using XSLT and the LaTeX system is used to create printable file such as DVI, PostScript, and PDF.

pachet	popcon(popularity)	limitate	keyword	descriere
docbook-xml	I:424	2126	xml	XML document type definition (DTD) for DocBook
docbook-xsl	V:16, I:151	14823	xml/xslt	XSL stylesheets for processing DocBook XML to various output formats with XSLT
xsltproc	V:16, I:75	83	xslt	XSLT command line processor (XML → XML, HTML, plain text, etc.)
xmlto	V:0, I:9	124	xml/xslt	XML-to-any converter with XSLT
fop	V:0, I:8	281	xml/xsl-fo	convert Docbook XML files to PDF
dblatex	V:1, I:6	4636	xml/xslt	convert Docbook files to DVI, PostScript, PDF documents with XSLT
dbtoepub	V:0, I:0	37	xml/xslt	DocBook XML to .epub converter

Tabela 11.8: List of XML tools

Since XML is subset of [Standard Generalized Markup Language \(SGML\)](#), it can be processed by the extensive tools available for SGML, such as [Document Style Semantics and Specification Language \(DSSSL\)](#).

pachet	popcon(popularity)	limitate	keyword	descriere
openjade	V:1, I:22	1066	dsssl	ISO/IEC 10179:1996 standard DSSSL processor (latest)
docbook-dsssl	V:0, I:8	2594	xml/dsssl	DSSSL stylesheets for processing DocBook XML to various output formats with DSSSL
docbook-utils	V:0, I:6	287	xml/dsssl	utilities for DocBook files including conversion to other formats (HTML, RTF, PS, man, PDF) with docbook2* commands with DSSSL

Tabela 11.9: List of DSSSL tools

Indicație

[GNOME](#)'s `ye1p` is sometimes handy to read [DocBook](#) XML files directly since it renders decently on X.

11.2.3 The XML data extraction

You can extract HTML or XML data from other formats using followings.

11.2.4 The XML data lint

For non-XML HTML files, you can convert them to XHTML which is an instance of well formed XML. XHTML can be processed by XML tools.

Syntax of XML files and goodness of URLs found in them may be checked.

Once proper XML is generated, you can use XSLT technology to extract data based on the mark-up context etc.

pachet	popcon(popularitate)	dimensiune	keyword	descriere
man2html	V:0, I:1	142	manpage → html	converter from manpage to HTML (CGI support)
doclifter	V:0, I:0	473	troff → xml	converter from troff to DocBook XML
texi2html	V:0, I:3	1847	texi → html	converter from Texinfo to HTML
info2www	V:1, I:1	74	info → html	converter from GNU info to HTML (CGI support)
wv	V:0, I:2	733	MSWord → orice	document converter from Microsoft Word to HTML, LaTeX, etc.
unrtf	V:0, I:3	159	rtf → html	document converter from RTF to HTML, etc
wp2x	I:0	200	WordPerfect → orice	WordPerfect 5.0 and 5.1 files to TeX, LaTeX, troff, GML and HTML

Tabela 11.10: List of XML data extraction tools

pachet	popcon(popularitate)	dimensiune	funcție	descriere
libxml2-utils	V:64, I:216	205	xml ↔ html ↔ xhtml	command line XML tool with xmllint(1) (syntax check, reformat, lint, ...)
tidy	V:0, I:7	79	xml ↔ html ↔ xhtml	HTML syntax checker and reformatter
weblint-perl	V:0, I:0	32	lint	syntax and minimal style checker for HTML
linklint	V:0, I:0	343	link check	fast link checker and web site maintenance tool

Tabela 11.11: List of XML pretty print tools

11.3 Type setting

The Unix [troff](#) program originally developed by AT&T can be used for simple typesetting. It is usually used to create manpages.

[TeX](#) created by Donald Knuth is a very powerful type setting tool and is the de facto standard. [LaTeX](#) originally written by Leslie Lamport enables a high-level access to the power of TeX.

pachet	popcon(popularitate)	dimensiune	keyword	descriere
texlive	I:29	57	(La)TeX	TeX system for typesetting, previewing and printing
groff	V:1, I:25	20577	troff	GNU troff text-formatting system

Tabela 11.12: List of type setting tools

11.3.1 roff typesetting

Traditionally, [roff](#) is the main Unix text processing system. See [roff\(7\)](#), [groff\(7\)](#), [groff\(1\)](#), [grotty\(1\)](#), [troff\(1\)](#), [groff_md\(7\)](#), [groff_man\(7\)](#), [groff_ms\(7\)](#), [groff_me\(7\)](#), [groff_mm\(7\)](#), and "info groff".

You can read or print a good tutorial and reference on "-me" [macro](#) in "/usr/share/doc/groff/" by installing the groff package.

Indicație

"groff -Tascii -me -" produces plain text output with [ANSI escape code](#). If you wish to get manpage like output with many "^H" and "_", use "GROFF_NO_SGR=1 groff -Tascii -me -" instead.

Indicație

To remove "^H" and "_" from a text file generated by groff, filter it by "col -b -x".

11.3.2 TeX/LaTeX

The [TeX Live](#) software distribution offers a complete TeX system. The `texlive` metapackage provides a decent selection of the [TeX Live](#) packages which should suffice for the most common tasks.

There are many references available for [TeX](#) and [LaTeX](#).

- [The teTeX HOWTO: The Linux-teTeX Local Guide](#)
- `tex(1)`
- `latex(1)`
- `texdoc(1)`
- `texdoctk(1)`
- "The TeXbook", by Donald E. Knuth, (Addison-Wesley)
- "LaTeX - A Document Preparation System", by Leslie Lamport, (Addison-Wesley)
- "The LaTeX Companion", by Goossens, Mittelbach, Samarin, (Addison-Wesley)

This is the most powerful typesetting environment. Many [SGML](#) processors use this as their back end text processor. [Lyx](#) provided by the `lyx` package and [GNU TeXmacs](#) provided by the `texmacs` package offer nice [WYSIWYG](#) editing environment for [LaTeX](#) while many use [Emacs](#) and [Vim](#) as the choice for the source editor.

There are many online resources available.

- The TEX Live Guide - TEX Live 2007 ("`/usr/share/doc/texlive-doc-base/english/texlive-en/live.html`") (`texlive-doc-base` package)
- [A Simple Guide to Latex/Lyx](#)
- [Word Processing Using LaTeX](#)

When documents become bigger, sometimes TeX may cause errors. You must increase pool size in "`/etc/texmf/texmf.d`" (or more appropriately edit "`/etc/texmf/texmf.d/95NonPath`" and run `update-texmf(8)`) to fix this.

Notă

The TeX source of "The TeXbook" is available at www.ctan.org/tex-archive/site/texbook.tex. This file contains most of the required macros. I heard that you can process this document with `tex(1)` after commenting lines 7 to 10 and adding "`\input manmac \proofmodefalse`". It's strongly recommended to buy this book (and all other books from Donald E. Knuth) instead of using the online version but the source is a great example of TeX input!

11.3.3 Pretty print a manual page

You can print a manual page in PostScript nicely by one of the following commands.

```
$ man -Tps some_manpage | lpr
```

11.3.4 Creating a manual page

Although writing a manual page (`manpage`) in the plain [troff](#) format is possible, there are few helper packages to create it.

pachet	popcon(popularity)	dimensiune	keyword	descriere
docbook-to-man	V:0, I:6	189	SGML → manpage	converter from DocBook SGML into roff man macros
help2man	V:0, I:6	542	text → manpage	automatic manpage generator from --help
info2man	V:0, I:0	134	info → manpage	converter from GNU info to POD or man pages
txt2man	V:0, I:0	112	text → manpage	convert flat ASCII text to man page format

Tabela 11.13: List of packages to help creating the manpage

11.4 Printable data

Printable data is expressed in the [PostScript](#) format on the Debian system. [Common Unix Printing System \(CUPS\)](#) uses Ghostscript as its rasterizer backend program for non-PostScript printers.

Printable data may also be expressed in the [PDF](#) format on the recent Debian system.

PDF files can displayed and its form entries may be filled using GUI viewer tools such as [Evince](#) and [Okular](#) (see Secțiune 7.4); and modern browsers such as [Chromium](#).

PDF files can be edited using some graphics tools such as [LibreOffice](#), [Scribus](#), and [Inkscape](#) (see Secțiune 11.6).

Indicație

You can read a PDF file with [GIMP](#) and convert it into [PNG](#) format using higher than 300 dpi resolution. This may be used as a background image for [LibreOffice](#) to produce a desirable altered printout with minimum efforts.

11.4.1 Ghostscript

The core of printable data manipulation is the [Ghostscript PostScript \(PS\)](#) interpreter which generates raster image.

pachet	popcon(popularity)	dimensiune	descriere
ghostscript	V:156, I:579	183	The GPL Ghostscript PostScript/PDF interpreter
ghostscript-x	I:17	88	GPL Ghostscript PostScript/PDF interpreter - X display support
libpoppler147	V:109, I:278	4891	PDF rendering library forked from the xpdf PDF viewer
libpoppler-glib8t64	V:63, I:273	550	PDF rendering library (GLib-based shared library)
poppler-data	V:171, I:600	13086	CMaps for PDF rendering library (for CJK support: Adobe-*)

Tabela 11.14: List of Ghostscript PostScript interpreters

Indicație

"gs -h" can display the configuration of Ghostscript.

11.4.2 Merge two PS or PDF files

You can merge two [PostScript \(PS\)](#) or [Portable Document Format \(PDF\)](#) files using gs(1) of 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

The [PDF](#), which is a widely used cross-platform printable data format, is essentially the compressed [PS](#) format with few additional features and extensions.

Indicație

For command line, `psmerge(1)` and other commands from the `psutils` package are useful for manipulating PostScript documents. `pdftk(1)` from the `pdftk` package is useful for manipulating PDF documents, too.

11.4.3 Printable data utilities

The following packages for the printable data utilities caught my eyes.

pachet	popcon(popularity)	dimensiune	keyword	descriere
poppler-utils	V:135, I:489	760	pdf → ps, text, ...	PDF utilities: <code>pdftops</code> , <code>pdfinfo</code> , <code>pdfimages</code> , <code>pdftotext</code> , <code>pdffonts</code>
psutils	V:3, I:53	34	ps → ps	PostScript document conversion tools
poster	V:0, I:1	58	ps → ps	create large posters out of PostScript pages
enscript	V:0, I:11	2138	text → ps, html, rtf	convert ASCII text to PostScript, HTML, RTF or Pretty-Print
a2ps	V:0, I:7	4083	text → ps	'Anything to PostScript' converter and pretty-printer
pdftk	I:25	28	pdf → pdf	PDF document conversion tool: <code>pdftk</code>
html2ps	V:0, I:1	256	html → ps	converter from HTML to PostScript
gnuhtml2latex	V:0, I:0	26	html → latex	converter from html to latex
latex2rtf	V:0, I:2	495	latex → rtf	convert documents from LaTeX to RTF which can be read by MS Word
ps2eps	V:1, I:34	95	ps → eps	converter from PostScript to EPS (Encapsulated PostScript)
e2ps	V:0, I:0	104	text → ps	Text to PostScript converter with Japanese encoding support
impose+	V:0, I:1	118	ps → ps	PostScript utilities
trueprint	V:0, I:0	148	text → ps	pretty print many source codes (C, C++, Java, Pascal, Perl, Pike, Sh, and Verilog) to PostScript. (C language)
pdf2svg	V:0, I:3	33	pdf → svg	converter from PDF to Scalable vector graphics format
pdftoipe	V:0, I:0	74	pdf → ipe	converter from PDF to IPE's XML format

Tabela 11.15: List of printable data utilities

11.4.4 Imprimarea cu CUPS

Both `lp(1)` and `lpr(1)` commands offered by [Common Unix Printing System \(CUPS\)](#) provides options for customized printing the printable data.

You can print 3 copies of a file collated using one of the following commands.

```
$ lp -n 3 -o Collate=True filename
```

```
$ lpr -#3 -o Collate=True filename
```

You can further customize printer operation by using printer option such as "-o number-up=2", "-o page-set=even", "-o page-set=odd", "-o scaling=200", "-o natural-scaling=200", etc., documented at [Command-Line Printing and Options](#).

11.5 The mail data conversion

The following packages for the mail data conversion caught my eyes.

pachet	popcon(popularity)	limitare	keyword	descriere
sharutils	V:2, I:30	1436	mail	shar(1), unshar(1), uuencode(1), uudecode(1)
mpack	V:0, I:8	109	MIME	encoding and decoding of MIME messages: mpack(1) and munpack(1)
tnef	V:0, I:4	103	ms-tnef	unpacking MIME attachments of type "application/ms-tnef" which is a Microsoft only format
uudeview	V:0, I:1	105	mail	encoder and decoder for the following formats: uuencode , xxencode , BASE64 , quoted printable , and BinHex

Tabela 11.16: List of packages to help mail data conversion

Indicație

The [Internet Message Access Protocol](#) version 4 (IMAP4) server may be used to move mails out from proprietary mail systems if the mail client software can be configured to use IMAP4 server too.

11.5.1 Mail data basics

Mail ([SMTP](#)) data should be limited to series of 7 bit data. So binary data and 8 bit text data are encoded into 7 bit format with the [Multipurpose Internet Mail Extensions \(MIME\)](#) and the selection of the charset (see Tabel 11.2).

The standard mail storage format is mbox formatted according to [RFC2822 \(updated RFC822\)](#). See mbox(5) (provided by the `mutt` package).

For European languages, "Content-Transfer-Encoding: quoted-printable" with the ISO-8859-1 charset is usually used for mail since there are not much 8 bit characters. If European text is encoded in UTF-8, "Content-Transfer-Encoding: quoted-printable" is likely to be used since it is mostly 7 bit data.

For Japanese, traditionally "Content-Type: text/plain; charset=ISO-2022-JP" is usually used for mail to keep text in 7 bits. But older Microsoft systems may send mail data in Shift-JIS without proper declaration. If Japanese text is encoded in UTF-8, [Base64](#) is likely to be used since it contains many 8 bit data. The situation of other Asian languages is similar.

Notă

If your non-Unix mail data is accessible by a non-Debian client software which can talk to the IMAP4 server, you may be able to move them out by running your own IMAP4 server.

Notă

If you use other mail storage formats, moving them to mbox format is the good first step. The versatile client program such as `mutt(1)` may be handy for this.

You can split mailbox contents to each message using `procmail(1)` and `formail(1)`.

Each mail message can be unpacked using `munpack(1)` from the `mpack` package (or other specialized tools) to obtain the MIME encoded contents.

11.6 Graphic data tools

Although GUI programs such as `gimp(1)` are very powerful, command line tools such as `imagemagick(1)` are quite useful for automating image manipulation via scripts.

The de facto image file format of the digital camera is the [Exchangeable Image File Format](#) (EXIF) which is the [JPEG](#) image file format with additional metadata tags. It can hold information such as date, time, and camera settings.

The [Lempel-Ziv-Welch \(LZW\) lossless data compression](#) patent has been expired. [Graphics Interchange Format \(GIF\)](#) utilities which use the LZW compression method are now freely available on the Debian system.

Indicație

Any digital camera or scanner with removable recording media works with Linux through [USB storage](#) readers since it follows the [Design rule for Camera Filesystem](#) and uses [FAT](#) filesystem. See Secțiune [10.1.7](#).

11.6.1 Graphic data tools (metapackage)

The following metapackages are good starting points for searching graphics data tools using `aptitude(8)`. "[Packages overview for Debian PhotoTools Maintainers](#)" can be another starting point.

pachet	popcon(popularity)	limite(limit)	keyword	descriere
education-graphics	1.0	31	svg, jpeg, ...	metapackage for teaching graphics and pictural art.
open-font-design-toolkit	1.0	9	ttf, ps, ...	metapackage for open font design

Tabela 11.17: List of graphics data tools (metapackage)

Indicație

Search more image tools using regex `"~Gworks-with::image"` in `aptitude(8)` (see Secțiune [2.2.6](#)).

11.6.2 Graphic data tools (GUI)

The following packages for the GUI graphics data conversion, editing, and organization tools caught my eyes.

11.6.3 Graphic data tools (CLI)

The following packages for the CLI graphics data conversion, editing, and organization tools caught my eyes.

pachet	popcon(popularity)	limita	keyword	descriere
gimp	V:35, I:229	31748	image(bitmap)	GNU Image Manipulation Program
xsane	V:10, I:135	1512	image(bitmap)	GTK-based X11 frontend for SANE (Scanner Access Now Easy)
scribus	V:1, I:13	32052	ps/pdf/SVG/...	Scribus DTP editor
libreoffice-draw	V:98, I:437	11003	image(vector)	LibreOffice office suite - drawing
inkscape	V:13, I:85	113183	image(vector)	SVG (Scalable Vector Graphics) editor
dia	V:1, I:18	3812	image(vector)	diagram editor (Gtk)
xfig	V:0, I:9	7951	image(vector)	Facility for Interactive Generation of figures under X11
gocr	V:0, I:4	549	image → text	free OCR software
eog	V:32, I:166	10310	image(Exif)	Eye of GNOME graphics viewer program
gthumb	V:3, I:12	5162	image(Exif)	image viewer and browser (GNOME)
geeqie	V:3, I:11	2982	image(Exif)	image viewer using GTK
shotwell	V:15, I:258	6334	image(Exif)	digital photo organizer (GNOME)
gwenview	V:41, I:119	6001	image(Exif)	image viewer (KDE)
kamera	I:118	982	image(Exif)	digital camera support for KDE applications
digikam	V:1, I:9	302	image(Exif)	digital photo management application for KDE
darktable	V:3, I:12	35895	image(Exif)	virtual lighttable and darkroom for photographers
hugin	V:0, I:6	6489	image(Exif)	panorama photo stitcher
librecad	V:1, I:14	9100	DXF, ...	2D CAD data editor
freecad	V:0, I:20	110	DXF, ...	3D CAD data editor
blender	V:2, I:23	92911	blend, TIFF, VRML, ...	3D content editor for animation etc
mm3d	V:0, I:0	4123	ms3d, obj, dxf, ...	OpenGL based 3D model editor
fontforge	V:0, I:5	4058	ttf, ps, ...	font editor for PS, TrueType and OpenType fonts
xgridfit	V:0, I:0	878	ttf	program for gridfitting and hinting TrueType fonts

Tabela 11.18: List of graphics data tools (GUI)

pachet	popcon(popularity)	limite	keyword	descriere
imagemagick	I:290	77	image(bitmap)	image manipulation programs
graphicsmagick	V:1, I:9	5816	image(bitmap)	image manipulation programs (fork of imagemagick)
netpbm	V:28, I:300	8435	image(bitmap)	graphics conversion tools
libheif-examples	V:0, I:3	439	heif → jpeg(bitmap)	convert High Efficiency Image File Format (HEIF) JPEG, PNG, or Y4M formats with heif-conver t(1) command
icoutils	V:4, I:35	221	png ↔ ico(bitmap)	convert MS Windows icons and cursors to and from PNG formats (favicon.ico)
pstoedit	V:1, I:40	1076	ps/pdf → image(vector)	PostScript and PDF files to editable vector graphics converter (SVG)
libwmf-bin	V:5, I:90	151	Windows/image(vector)	Windows metafile (vector graphics data) conversion tools
fig2sxd	V:0, I:0	151	fig → sxd(vector)	convert XFig files to OpenOffice.org Draw format
unpaper	V:2, I:16	417	image → image	post-processing tool for scanned pages for OCR
tesseract-ocr	V:7, I:32	2243	image → text	free OCR software based on the HP's commercial OCR engine
tesseract-ocr-eng	V:7, I:33	4032	image → text	OCR engine data: tesseract-ocr language files for English text
ocrad	V:0, I:2	604	image → text	free OCR software
exif	V:3, I:53	335	image(Exif)	command-line utility to show EXIF information in JPEG files
exiv2	V:1, I:21	432	image(Exif)	EXIF/IPTC metadata manipulation tool
exiftran	V:0, I:12	81	image(Exif)	transform digital camera jpeg images
exiftags	V:0, I:3	309	image(Exif)	utility to read Exif tags from a digital camera JPEG file
exifprobe	V:0, I:2	502	image(Exif)	read metadata from digital pictures
dcraw	V:1, I:8	428	image(Raw)	decode raw digital camera images
findimagedupes	V:0, I:1	76	image → imagine(digi-tală(fingerprint)	find visually similar or duplicate images
ale	V:0, I:0	818	image → image	merge images to increase fidelity or create mosaics
imageindex	V:0, I:1	143	image(Exif)	generate static HTML galleries from images
outguess	V:0, I:1	230	jpeg,png	universal Steganographic tool
jpegoptim	V:0, I:6	59	jpeg	optimize JPEG files
optipng	V:2, I:42	187	png	optimize PNG files, lossless compression
pngquant	V:1, I:10	62	png	optimize PNG files, lossy compression

Tabela 11.19: List of graphics data tools (CLI)

pachet	popcon(popularity)	limite	keyword	descriere
alien	V:1, I:13	150	rpm/tgz → deb	converter for the foreign package into the Debian package
freepwing	V:0, I:0	447	EB → EPWING	converter from "Electric Book" (popular in Japan) to a single JIS X 4081 format (a subset of the EPWING V1)
calibre	V:8, I:26	65584	orice → EPUB	convertor de cărți electronice și gestionarea bibliotecii

Tabela 11.20: List of miscellaneous data conversion tools

11.7 Miscellaneous data conversion

There are many other programs for converting data. Following packages caught my eyes using regex "~Guse : : converting in aptitude(8) (see Secțiune [2.2.6](#)).

You can also extract data from RPM format with the following.

```
$ rpm2cpio file.src.rpm | cpio --extract
```

Capitolul 12

Programare

I provide some pointers for people to learn programming on the Debian system enough to trace the packaged source code. Here are notable packages and corresponding documentation packages for programming.

Online references are available by typing "man name" after installing manpages and manpages-dev packages. Online references for the GNU tools are available by typing "info program_name" after installing the pertinent documentation packages. You may need to include the contrib and non-free archives in addition to the main archive since some GFDL documentations are not considered to be DFSG compliant.

Please consider to use version control system tools. See Secțiune [10.5](#).



Avertisment

Do not use "test" as the name of an executable test file. "test" is a shell builtin.



Atenție

You should install software programs directly compiled from source into "/usr/local" or "/opt" to avoid collision with system programs.

Indicație

[Code examples of creating "Song 99 Bottles of Beer"](#) should give you good ideas of practically all the programming languages.

12.1 The shell script

The [shell script](#) is a text file with the execution bit set and contains the commands in the following format.

```
#!/bin/sh
... command lines
```

The first line specifies the shell interpreter which read and execute this file contents.

Reading shell scripts is the **best** way to understand how a Unix-like system works. Here, I give some pointers and reminders for shell programming. See "Shell Mistakes" (<https://www.greenend.org.uk/rjk/2001/04/shell.html>) to learn from mistakes.

Unlike shell interactive mode (see Secțiune [1.5](#) and Secțiune [1.6](#)), shell scripts frequently use parameters, conditionals, and loops.

12.1.1 POSIX shell compatibility

Many system scripts may be interpreted by any one of [POSIX](#) shells (see Tabel [1.13](#)).

- The default non-interactive POSIX shell `/usr/bin/sh` is a symlink pointing to `/usr/bin/dash` and used by many system programs.
- The default interactive POSIX shell is `/usr/bin/bash`.

Avoid writing a shell script with **bashisms** or **zshisms** to make it portable among all POSIX shells. You can check it using `checkbashisms(1)`.

Good: POSIX	Avoid: bashism
<code>if ["\$foo" = "\$bar"] ; then ...</code>	<code>if ["\$foo" == "\$bar"] ; then ...</code>
<code>diff -u file.c.orig file.c</code>	<code>diff -u file.c{.orig,}</code>
<code>mkdir /foobar /foobaz</code>	<code>mkdir /foo{bar,baz}</code>
<code>funcname() { ... }</code>	<code>function funcname() { ... }</code>
octal format: <code>"\377"</code>	hexadecimal format: <code>"\xff"</code>

Tabela 12.1: List of typical bashisms

The `"echo"` command must be used with following cares since its implementation differs among shell builtin and external commands.

- Avoid using any command options except `"-n"`.
- Avoid using escape sequences in the string since their handling varies.

Notă

Although `"-n"` option is **not** really POSIX syntax, it is generally accepted.

Indicație

Use the `"printf"` command instead of the `"echo"` command if you need to embed escape sequences in the output string.

12.1.2 Shell parameters

Special shell parameters are frequently used in the shell script.

Basic **parameter expansions** to remember are as follows.

Here, the colon `:"` in all of these operators is actually optional.

- **with** `":"` = operator test for **exist** and **not null**
 - **without** `":"` = operator test for **exist** only
-

parametru shell	valoare
\$0	name of the shell or shell script
\$1	first (1st) shell argument
\$9	ninth (9th) shell argument
\$#	number of positional parameters
"\$*"	"\$1 \$2 \$3 \$4 ... "
"\$@"	"\$1" "\$2" "\$3" "\$4" ...
\$?	exit status of the most recent command
\$\$	PID of this shell script
\$!	PID of most recently started background job

Tabela 12.2: List of shell parameters

parameter expression form	value if var is set	value if var is not set
\${var:-string}	"\$var"	"string"
\${var:+string}	"string"	„null"
\${var:=string}	"\$var"	"string" (and run "var=string")
\${var:?string}	"\$var"	echo "string" to stderr (and exit with error)

Tabela 12.3: List of shell parameter expansions

parameter substitution form	rezultatul
\${var%suffix}	remove smallest suffix pattern
\${var%%suffix}	remove largest suffix pattern
\${var#prefix}	remove smallest prefix pattern
\${var##prefix}	remove largest prefix pattern

Tabela 12.4: List of key shell parameter substitutions

12.1.3 Condiționale shell

Each command returns an **exit status** which can be used for conditional expressions.

- Success: 0 ("True")
- Error: non 0 ("False")

Notă

"0" in the shell conditional context means "True", while "0" in the C conditional context means "False".

Notă

"[" is the equivalent of the `test` command, which evaluates its arguments up to "]" as a conditional expression.

Basic **conditional idioms** to remember are the following.

- `"command && if_success_run_this_command_too || true"`
- `"command || if_not_success_run_this_command_too || true"`
- A multi-line script snippet as the following

```
if [ conditional_expression ]; then
    if_success_run_this_command
else
    if_not_success_run_this_command
fi
```

Here trailing `"|| true"` was needed to ensure this shell script does not exit at this line accidentally when shell is invoked with `"-e"` flag.

equation	condition to return logical true
<code>-e fișier</code>	<i>file</i> exists
<code>-d fișier</code>	<i>file</i> exists and is a directory
<code>-f fișier</code>	<i>file</i> exists and is a regular file
<code>-w fișier</code>	<i>file</i> exists and is writable
<code>-x fișier</code>	<i>file</i> exists and is executable
<code>fișier1 -nt fișier2</code>	<i>file1</i> is newer than <i>file2</i> (modification)
<code>fișier1 -ot fișier2</code>	<i>file1</i> is older than <i>file2</i> (modification)
<code>fișier1 -ef fișier2</code>	<i>file1</i> and <i>file2</i> are on the same device and the same inode number

Tabela 12.5: List of file comparison operators in the conditional expression

Arithmetic integer comparison operators in the conditional expression are `"-eq"`, `"-ne"`, `"-lt"`, `"-le"`, `"-gt"`, and `"-ge"`.

12.1.4 Bucle shell

There are several loop idioms to use in POSIX shell.

- `"for x in foo1 foo2 ... ; do command ; done"` loops by assigning items from the list `"foo1 foo2 ..."` to variable `"x"` and executing `"command"`.
-

equation	condition to return logical true
<code>-z <i>str</i></code>	the length of <i>str</i> is zero
<code>-n <i>str</i></code>	the length of <i>str</i> is non-zero
<code><i>str1</i> = <i>str2</i></code>	<i>str1</i> and <i>str2</i> are equal
<code><i>str1</i> != <i>str2</i></code>	<i>str1</i> and <i>str2</i> are not equal
<code><i>str1</i> < <i>str2</i></code>	<i>str1</i> sorts before <i>str2</i> (locale dependent)
<code><i>str1</i> > <i>str2</i></code>	<i>str1</i> sorts after <i>str2</i> (locale dependent)

Tabela 12.6: List of string comparison operators in the conditional expression

- "while condition ; do command ; done" repeats "command" while "condition" is true.
- "until condition ; do command ; done" repeats "command" while "condition" is not true.
- "break" enables to exit from the loop.
- "continue" enables to resume the next iteration of the loop.

Indicație

The C-language like numeric iteration can be realized by using `seq(1)` as the "foo1 foo2 ..." generator.

Indicație

See Secțiune [9.4.9](#).

12.1.5 Shell environment variables

Some popular environment variables for the normal shell command prompt may not be available under the execution environment of your script.

- For "\$USER", use "\$(id -un)"
- For "\$UID", use "\$(id -u)"
- For "\$HOME", use "\$(getent passwd "\$(id -u)" | cut -d ':' -f 6)" (this works also on Secțiune [4.5.2](#))

12.1.6 The shell command-line processing sequence

The shell processes a script roughly as the following sequence.

- The shell reads a line.
- The shell groups a part of the line as **one token** if it is within "\"" or '... '.
- The shell splits other part of a line into **tokens** by the following.
 - Spații-albe: *spațiu tabulator linie-nouă*
 - Metacaractere: < > | ; & ()
- The shell checks the **reserved word** for each token to adjust its behavior if not within "\"" or '... '.
 - **reserved word**: if then elif else fi for in while unless do done case esac
- The shell expands **alias** if not within "\"" or '... '.

- The shell expands **tilde** if not within `"..."` or `'...'`.
 - `"~"` → current user's home directory
 - `"~user"` → *user*'s home directory
- The shell expands **parameter** to its value if not within `'...'`.
 - **parameter**: `"$PARAMETER"` or `"${PARAMETER}"`
- The shell expands **command substitution** if not within `'...'`.
 - `"$(command)"` → the output of `"command"`
 - `"` command `"` → the output of `"command"`
- The shell expands **pathname glob** to matching file names if not within `"..."` or `'...'`.
 - `*` → orice caractere
 - `?` → un caracter
 - `[...]` → any one of the characters in `"..."`
- The shell looks up **command** from the following and execute it.
 - **function** definition
 - **builtin** command
 - **executable file** in `"$PATH"`
- The shell goes to the next line and repeats this process again from the top of this sequence.

Single quotes within double quotes have no effect.

Executing `"set -x"` in the shell or invoking the shell with `"-x"` option make the shell to print all of commands executed. This is quite handy for debugging.

12.1.7 Utility programs for shell script

In order to make your shell program as portable as possible across Debian systems, it is a good idea to limit utility programs to ones provided by **essential** packages.

- `"aptitude search ~E"` lists **essential** packages.
- `"dpkg -L package_name |grep '/man/man.*/'"` lists manpages for commands offered by *package_name* package.

pachet	popcon(popularitete)	dimensiune	descriere
dash	V:906, I:998	207	small and fast POSIX-compliant shell for sh
coreutils	V:892, I:999	18457	GNU core utilities
grep	V:765, I:999	1297	GNU grep, egrep și fgrep
sed	V:804, I:999	987	GNU sed
mawk	V:466, I:997	296	small and fast awk
debianutils	V:915, I:996	225	miscellaneous utilities specific to Debian
bsdutils	V:439, I:999	335	basic utilities from 4.4BSD-Lite
bsdextrautils	V:728, I:849	361	extra utilities from 4.4BSD-Lite
moreutils	V:16, I:37	231	additional Unix utilities

Tabela 12.7: List of packages containing small utility programs for shell scripts

Indicație

Although `moreutils` may not exist outside of Debian, it offers interesting small programs. Most notable one is `sponge(8)` which is quite useful when you wish to overwrite original file.

Consultați Secțiune 1.6 pentru exemple.

12.2 Scripting in interpreted languages

pachet	popcon	popularity	documentație
dash	V:906, I:998	207	sh : small and fast POSIX-compliant shell for <code>sh</code>
bash	V:866, I:999	7277	sh : "info bash" provided by <code>bash-doc</code>
mawk	V:466, I:997	296	AWK : small and fast <code>awk</code>
gawk	V:251, I:310	3289	AWK : "info gawk" provided by <code>gawk-doc</code>
perl	V:672, I:991	841	Perl : <code>perl(1)</code> and html pages provided by <code>perl-doc</code> and <code>perl-doc-html</code>
libterm-readline-gnu-perl	V:2, I:28	439	Perl extension for the GNU ReadLine/History Library: <code>perlsh(1)</code>
libreply-perl	V:0, I:0	171	REPL for Perl: <code>reply(1)</code>
libdevel-repl-perl	V:0, I:0	237	REPL for Perl: <code>re.pl(1)</code>
python3	V:713, I:970	82	Python : <code>python3(1)</code> and html pages provided by <code>python3-doc</code>
tcl	V:24, I:184	20	Tcl : <code>tcl(3)</code> and detail manual pages provided by <code>tcl-doc</code>
tk	V:18, I:178	20	Tk : <code>tk(3)</code> and detail manual pages provided by <code>tk-doc</code>
ruby	V:69, I:166	32	Ruby : <code>ruby(1)</code> , <code>erb(1)</code> , <code>irb(1)</code> , <code>rdoc(1)</code> , <code>ri(1)</code>

Tabela 12.8: List of interpreter related packages

When you wish to automate a task on Debian, you should script it with an interpreted language first. The guide line for the choice of the interpreted language is:

- Use `dash`, if the task is a simple one which combines CLI programs with a shell program.
- Use `python3`, if the task isn't a simple one and you are writing it from scratch.
- Use `perl`, `tcl`, `ruby`, ... if there is an existing code using one of these languages on Debian which needs to be touched up to do the task.

If the resulting code is too slow, you can rewrite only the critical portion for the execution speed in a compiled language and call it from the interpreted language.

12.2.1 Debugging interpreted language codes

Most interpreters offer basic syntax check and code tracing functionalities.

- "**dash -n** *script.sh*" - Syntax check of a Shell script
- "**dash -x** *script.sh*" - Trace a Shell script
- "**python -m py_compile** *script.py*" - Syntax check of a Python script
- "**python -mtrace --trace** *script.py*" - Trace a Python script

- “**perl -l ../libpath -c script.pl**” - Syntax check of a Perl script
- “**perl -d:Trace script.pl**” - Trace a Perl script

For testing code for dash, try Secțiune 9.1.4 which accommodates bash-like interactive environment.

For testing code for perl, try REPL environment for Perl which accommodates Python-like REPL (=READ + EVAL + PRINT + LOOP) environment for Perl.

12.2.2 GUI program with the shell script

The shell script can be improved to create an attractive GUI program. The trick is to use one of so-called dialog programs instead of dull interaction using echo and read commands.

pachet	popcon(popularitete)	descriere
x11-utils	V:227, I:568 651	xmessage(1): display a message or query in a window (X)
whiptail	V:298, I:996 61	displays user-friendly dialog boxes from shell scripts (newt)
dialog	V:9, I:82 520	displays user-friendly dialog boxes from shell scripts (ncurses)
zenity	V:67, I:357 194	display graphical dialog boxes from shell scripts (GTK)
ssft	V:0, I:0 75	Shell Scripts Frontend Tool (wrapper for zenity, kdialog, and dialog with gettext)
gettext	V:53, I:229 7165	"/usr/bin/gettext.sh": translate message

Tabela 12.9: List of dialog programs

Here is an example of GUI program to demonstrate how easy it is just with a shell script.

This script uses zenity to select a file (default /etc/motd) and display it.

GUI launcher for this script can be created following 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
```

This kind of approach to GUI program with the shell script is useful only for simple choice cases. If you are to write any program with complexities, please consider writing it on more capable platform.

12.2.3 Custom actions for GUI filer

GUI filer programs can be extended to perform some popular actions on selected files using additional extension packages. They can also made to perform very specific custom actions by adding your specific scripts.

- For GNOME, see [NautilusScriptsHowto](#).
- For KDE, see [Creating Dolphin Service Menus](#).
- For Xfce, see [Thunar - Custom Actions](#) and <https://help.ubuntu.com/community/ThunarCustomActions>.
- For LXDE, see [Custom Actions](#).

12.2.4 Perl short script madness

In order to process data, `sh` needs to spawn sub-process running `cut`, `grep`, `sed`, etc., and is slow. On the other hand, `perl` has internal capabilities to process data, and is fast. So many system maintenance scripts on Debian use `perl`.

Let's think following one-liner AWK script snippet and its equivalents in Perl.

```
awk '($2=="1957") { print $3 }' |
```

This is equivalent to any one of the following lines.

```
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' |
```

The last one is a riddle. It took advantage of following Perl features.

- The whitespace is optional.
- The automatic conversion exists from number to the string.
- Perl execution tricks via command line options: `perlrun(1)`
- Perl special variables: `perlvar(1)`

This flexibility is the strength of Perl. At the same time, this allows us to create cryptic and tangled codes. So be careful.

12.3 Coding in compiled languages

Here, Secțiune 12.3.3 and Secțiune 12.3.4 are included to indicate how compiler-like program can be written in C language by compiling higher level description into C language.

12.3.1 C

You can set up proper environment to compile programs written in the [C programming language](#) by the following.

```
# apt-get install glibc-doc manpages-dev libc6-dev gcc build-essential
```

The `libc6-dev` package, i.e., GNU C Library, provides [C standard library](#) which is collection of header files and library routines used by the C programming language.

See references for C as the following.

- „`info libc`” (referința funcțiilor bibliotecii C)
- `gcc(1)` și «`info gcc`»
- `each_C_library_function_name(3)`
- Kernighan & Ritchie, "The C Programming Language", 2nd edition (Prentice Hall)

pachet	popcon	(popularity)	descriere
gcc	V:151, I:561	36	GNU C compiler
libc6-dev	V:268, I:579	12694	GNU C Library: Development Libraries and Header Files
g++	V:56, I:524	13	GNU C++ compiler
libstdc++-14-dev	V:31, I:227	24527	GNU Standard C++ Library v3 (development files)
cpp	V:335, I:725	18	GNU C preprocessor
gettext	V:53, I:229	7165	GNU Internationalization utilities
glade	V:0, I:3	1613	GTK User Interface Builder
valac	V:0, I:3	532	C# like language for the GObject system
flex	V:6, I:69	1247	LEX-compatible fast lexical analyzer generator
bison	V:6, I:73	3122	YACC-compatible parser generator
susv2	I:0	16	fetch "The Single UNIX Specifications v2"
susv3	I:0	16	fetch "The Single UNIX Specifications v3"
susv4	I:0	16	fetch "The Single UNIX Specifications v4"
golang	I:21	12	Go programming language compiler
rustc	V:5, I:18	13748	Rust systems programming language
gfortran	V:5, I:52	15	GNU Fortran 95 compiler
fpc	I:2	104	Free Pascal

Tabela 12.10: List of compiler related packages

12.3.2 Simple C program (gcc)

A simple example "example.c" can be compiled with a library "libm" into an executable "run_example" by the following.

```
$ 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
```

Here, "-lm" is needed to link library "/usr/lib/libm.so" from the libc6 package for sqrt(3). The actual library is in "/lib/" with filename "libm.so.6", which is a symlink to "libm-2.7.so".

Look at the last parameter in the output text. There are more than 10 characters even though "%10s" is specified.

The use of pointer memory operation functions without boundary checks, such as sprintf(3) and strcpy(3), is deprecated to prevent buffer overflow exploits that leverage the above overrun effects. Instead, use snprintf(3) and strncpy(3).

12.3.3 Flex — a better Lex

Flex is a [Lex](#)-compatible fast [lexical analyzer](#) generator.

Tutorial for `flex(1)` can be found in `"info flex"`.

Many simple examples can be found under `"/usr/share/doc/flex/examples/"`. [1](#)

12.3.4 Bison — a better Yacc

Several packages provide a [Yacc](#)-compatible lookahead [LR parser](#) or [LALR parser](#) generator in Debian.

pachet	popcon	popularity	descriere
bison	V:6, I:73	3122	GNU LALR parser generator
byacc	V:0, I:3	263	Berkeley LALR parser generator
btyacc	V:0, I:0	251	backtracking parser generator based on byacc

Tabela 12.11: List of Yacc-compatible LALR parser generators

Tutorial for `bison(1)` can be found in `"info bison"`.

You need to provide your own `"main()"` and `"yyerror()"`. `"main()"` calls `"yyparse()"` which calls `"yylex()"`, usually created with Flex.

Here is an example to create a simple terminal calculator program.

Let's create `example.y`:

```
/* calculator source for bison */
%{
#include <stdio.h>
extern int yylex(void);
extern int yyerror(char *);
%}

/* 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; }
;
%%
```

¹Some [tweaks](#) may be required to get them work under the current system.

```
int main(int argc, char **argv)
{
    yyparse();
}

int yyerror(char *s)
{
    fprintf(stderr, "error: '%s'\n", s);
}
```

Let's create, `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 */ }
%%
```

Then execute as follows from the shell prompt to try this:

```
$ 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 Static code analysis tools

[Lint](#) like tools can help automatic [static code analysis](#).

[Indent](#) like tools can help human code reviews by reformatting source codes consistently.

[Ctags](#) like tools can help human code reviews by generating an index (or tag) file of names found in source codes.

Indicație

Configuring your favorite editor (emacs or vim) to use asynchronous lint engine plugins helps your code writing. These plugins are getting very powerful by taking advantage of [Language Server Protocol](#). Since they are moving fast, using their upstream code instead of Debian package may be a good option.

pachet	popcon(popularitate)	driftage	descriere
vim-ale	I:0	2833	Asynchronous Lint Engine for Vim 8 and NeoVim
vim-syntastic	I:2	1379	Syntax checking hacks for vim
elpa-flycheck	V:0, I:1	815	modern on-the-fly syntax checking for Emacs
elpa-relint	I:0	150	Emacs Lisp regexp mistake finder
cppcheck-gui	V:0, I:1	7682	tool for static C/C++ code analysis (GUI)
shellcheck	V:2, I:15	22859	lint tool for shell scripts
pyflakes3	V:2, I:15	20	passive checker of Python 3 programs
pylint	V:4, I:20	2089	Python code static checker
perl	V:672, I:991	841	interpreter with internal static code checker: B::Lint(3perl)
rubocop	V:0, I:0	3247	Ruby static code analyzer
clang-tidy	V:1, I:11	22	clang-based C++ linter tool
splint	V:0, I:1	2328	tool for statically checking C programs for bugs
flawfinder	V:0, I:0	205	tool to examine C/C++ source code and looks for security weaknesses
black	V:4, I:16	10138	uncompromising Python code formatter
perltidy	V:0, I:3	3086	Perl script indenter and reformatter
indent	V:0, I:5	438	C language source code formatting program
astyle	V:0, I:2	769	Source code indenter for C, C++, Objective-C, C#, and Java
bcpp	V:0, I:0	114	C(++) beautifier
xmlindent	V:0, I:0	52	XML stream reformatter
global	V:0, I:1	1923	Source code search and browse tools
exuberant-ctags	V:1, I:14	341	build tag file indexes of source code definitions
universal-ctags	V:1, I:12	4238	build tag file indexes of source code definitions

Tabela 12.12: List of tools for static code analysis

12.5 Depanare

Debug is important part of programming activities. Knowing how to debug programs makes you a good Debian user who can produce meaningful bug reports.

pachet	popcon(popularitate)	driftage	documentație
gdb	V:79, I:156	12478	„info gdb” furnizat de gdb-doc
ddd	V:0, I:5	4210	„info ddd” furnizat de ddd-doc

Tabela 12.13: List of debug packages

12.5.1 Basic gdb execution

Primary [debugger](#) on Debian is gdb(1) which enables you to inspect a program while it executes.

Let's install gdb and related programs by the following.

```
# apt-get install gdb gdb-doc build-essential devscripts
```

Good tutorial of gdb can be found:

- “info gdb”
- “Debugging with GDB” in `/usr/share/doc/gdb-doc/html/gdb/index.html`
- [“tutorial on the web”](#)

Here is a simple example of using `gdb(1)` on a “program” compiled with the “-g” option to produce debugging information.

```
$ 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

Many `gdb(1)` commands can be abbreviated. Tab expansion works as in the shell.

12.5.2 Debugging the Debian package

Since all installed binaries should be stripped on the Debian system by default, most debugging symbols are removed in the normal package. In order to debug Debian packages with `gdb(1)`, *-dbgsym packages need to be installed (e.g. `coreutils-dbgsym` in the case of `coreutils`). The source packages generate *-dbgsym packages automatically along with normal binary packages and those debug packages are placed separately in [debian-debug](#) archive. Please refer to [articles on Debian Wiki](#) for more information.

If a package to be debugged does not provide its *-dbgsym package, you need to install it after rebuilding it by the following.

```
$ 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 ./
```

Fix bugs if needed.

Bump package version to one which does not collide with official Debian versions, e.g. one appended with “+debug1” when recompiling existing package version, or one appended with “~pre1” when compiling unreleased package version by the following.

```
$ dch -i
```

Compile and install packages with debug symbols by the following.

```
$ export DEB_BUILD_OPTIONS="nostrip noopt"
$ debuild
$ cd ..
$ sudo debi package_name*.changes
```

You need to check build scripts of the package and ensure to use "CFLAGS=-g -Wall" for compiling binaries.

12.5.3 Obtaining backtrace

When you encounter program crash, reporting bug report with cut-and-pasted backtrace information is a good idea. The backtrace can be obtained by `gdb(1)` using one of the following approaches:

- Crash-in-GDB approach:
 - Rulați programul din GDB.
 - Crash the program.
 - Tastați „bt” în promptul GDB.
- Crash-first approach:
 - Update the **“/etc/security/limits.conf”** file to include the following:

```
* soft core unlimited
```

- Type `“ulimit -c unlimited”` to the shell prompt.
- Run the program from this shell prompt.
- Crash the program to produce a [core dump](#) file.
- Load the [core dump](#) file to GDB as `“gdb gdb ./program_binary core”`.
- Tastați „bt” în promptul GDB.

For infinite loop or frozen keyboard situation, you can force to crash the program by pressing `Ctrl-\` or `Ctrl-C` or executing `“kill -ABRT PID”`. (See Secțiune [9.4.12](#))

Indicație

Often, you see a backtrace where one or more of the top lines are in `“malloc()”` or `“g_malloc()”`. When this happens, chances are your backtrace isn't very useful. The easiest way to find some useful information is to set the environment variable `“$MALLOCCHECK_”` to a value of 2 (`malloc(3)`). You can do this while running `gdb` by doing the following.

```
$ MALLOCCHECK_=2 gdb hello
```

12.5.4 Advanced gdb commands

12.5.5 Check dependency on libraries

Use `ldd(1)` to find out a program's dependency on libraries by the followings.

```
$ 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)
```

For `ls(1)` to work in a ``chroot`ed` environment, the above libraries must be available in your ``chroot`ed` environment. See Secțiune [9.4.6](#).

comanda	description for command objectives
(gdb) thread apply all bt	get a backtrace for all threads for multi-threaded program
(gdb) bt full	get parameters came on the stack of function calls
(gdb) thread apply all bt full	get a backtrace and parameters as the combination of the preceding options
(gdb) thread apply all bt full 10	get a backtrace and parameters for top 10 calls to cut off irrelevant output
(gdb) set logging on	write log of gdb output to a file (the default is "gdb.txt")

Tabela 12.14: List of advanced gdb commands

12.5.6 Dynamic call tracing tools

There are several dynamic call tracing tools available in Debian. See Secțiune 9.4.

12.5.7 Debugging X Errors

If a GNOME program `preview1` has received an X error, you should see a message as follows.

The program 'preview1' received an X Window System error.

If this is the case, you can try running the program with `"- - sync"`, and break on the `"gtk_x_error"` function in order to obtain a backtrace.

12.5.8 Memory leak detection tools

There are several memory leak detection tools available in Debian.

pachet	popcon	popularity	descriere
libc6-dev	V:268, I:579	12694	mt race(1): malloc debugging functionality in glibc
valgrind	V:6, I:34	87847	memory debugger and profiler
electric-fence	V:0, I:2	69	ma lloc(3) debugger
libdmalloc5	V:0, I:0	380	debug memory allocation library
duma	V:0, I:0	297	library to detect buffer overruns and under-runs in C and C++ programs
leaktracer	I:0	56	memory-leak tracer for C++ programs

Tabela 12.15: List of memory leak detection tools

12.5.9 Disassemble binary

You can disassemble binary code with `objdump(1)` by the following.

```
$ objdump -m i386 -b binary -D /usr/lib/grub/x86_64-pc/stage1
```

Notă

`gdb(1)` may be used to disassemble code interactively.

12.6 Instrumentele de construcție

pachet	popcon	populartate	documentație
make	V:145, I:562	1762	"info make" provided by make-doc
autoconf	V:29, I:205	2197	"info autoconf" provided by autoconf-doc
automake	V:28, I:204	1933	"info automake" provided by automake1.10-doc
libtool	V:23, I:187	1245	"info libtool" provided by libtool-doc
cmake	V:18, I:118	44267	cmake(1) cross-platform, open-source make system
ninja-build	V:7, I:51	456	ninja(1) small build system closest in spirit to Make
meson	V:6, I:28	4186	meson(1) high productivity build system on top of ninja
xutils-dev	V:0, I:7	1495	imake(1), xmkmf(1), etc.

Tabela 12.16: List of build tool packages

12.6.1 Make

Make is a utility to maintain groups of programs. Upon execution of `make(1)`, make read the rule file, "Makefile", and updates a target if it depends on prerequisite files that have been modified since the target was last modified, or if the target does not exist. The execution of these updates may occur concurrently.

The rule file syntax is the following.

```
target: [ prerequisites ... ]
[TAB] command1
[TAB] -command2 # ignore errors
[TAB] @command3 # suppress echoing
```

Here "[TAB]" is a TAB code. Each line is interpreted by the shell after make variable substitution. Use "\" at the end of a line to continue the script. Use "\$\$" to enter "\$" for environment values for a shell script.

Implicit rules for the target and prerequisites can be written, for example, by the following.

```
%.o: %.c header.h
```

Here, the target contains the character "%" (exactly one of them). The "%" can match any nonempty substring in the actual target filenames. The prerequisites likewise use "%" to show how their names relate to the actual target name.

automatic variable	valoare
\$@	ținta
\$<	first prerequisite
\$?	all newer prerequisites
\$^	all prerequisites
\$*	"%" matched stem in the target pattern

Tabela 12.17: List of make automatic variables

expandarea variabilei	descriere
foo1 := bar	one-time expansion
foo2 = bar	recursive expansion
foo3 += bar	adăugare

Tabela 12.18: List of make variable expansions

Run "`make -p -f/dev/null`" to see automatic internal rules.

12.6.2 Autotools

Autotools is a suite of programming tools designed to assist in making source code packages portable to many **Unix-like** systems.

- **Autoconf** is a tool to produce a shell script "configure" from "configure.ac".
 - "configure" is used later to produce "Makefile" from "Makefile.in" template.
- **Automake** is a tool to produce "Makefile.in" from "Makefile.am".
- **Libtool** is a shell script to address the software portability problem when compiling shared libraries from source code.

12.6.2.1 Compile and install a program



Avertisment

Do not overwrite system files with your compiled programs when installing them.

Debian does not touch files in "/usr/local/" or "/opt". So if you compile a program from source, install it into "/usr/local/" so it does not interfere with 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 Uninstall program

If you have the original source and if it uses `autoconf(1)`/`automake(1)` and if you can remember how you configured it, execute as follows to uninstall the program.

```
$ ./configure all-of-the-options-you-gave-it
$ sudo make uninstall
```

Alternatively, if you are absolutely sure that the install process puts files only under "/usr/local/" and there is nothing important there, you can erase all its contents by the following.

```
# find /usr/local -type f -print0 | xargs -0 rm -f
```

If you are not sure where files are installed, you should consider using `checkinstall(8)` from the `checkinstall` package, which provides a clean path for the uninstall. It now supports to create a Debian package with "-D" option.

12.6.3 Meson

The software build system has been evolving:

- **Autotools** on the top of **Make** has been the de facto standard for the portable build infrastructure since 1990s. This is extremely slow.
 - **CMake** initially released in 2000 improved speed significantly but was originally built on the top of inherently slow **Make**. (Now **Ninja** can be its backend.)
-

- [Ninja](#) initially released in 2012 is meant to replace Make for the further improved build speed and is designed to have its input files generated by a higher-level build system.
- [Meson](#) initially released in 2013 is the new popular and fast higher-level build system which uses [Ninja](#) as its backend.

See documents found at "[The Meson Build system](#)" and "[The Ninja build system](#)".

12.7 Web

Basic interactive dynamic web pages can be made as follows.

- Queries are presented to the browser user using [HTML](#) forms.
- Filling and clicking on the form entries sends one of the following [URL](#) string with encoded parameters from the browser to the web server.
 - "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" in URL is replaced with a character with hexadecimal nn value.
- The environment variable is set as: "QUERY_STRING="VAR1=VAL1 VAR2=VAL2 VAR3=VAL3"".
- [CGI](#) program (any one of "program.*") on the web server executes itself with the environment variable "\$QUERY_STRING".
- stdout of CGI program is sent to the web browser and is presented as an interactive dynamic web page.

For security reasons it is better not to hand craft new hacks for parsing CGI parameters. There are established modules for them in Perl and Python. [PHP](#) comes with these functionalities. When client data storage is needed, [HTTP cookies](#) are used. When client side data processing is needed, [Javascript](#) is frequently used.

For more, see the [Common Gateway Interface](#), [The Apache Software Foundation](#), and [JavaScript](#).

Searching "CGI tutorial" on Google by typing encoded URL <https://www.google.com/search?hl=en&ie=UTF-8&q=CGI+tutorial> directly to the browser address is a good way to see the CGI script in action on the Google server.

12.8 The source code translation

There are programs to convert source codes.

pachet	popcon(popularity contest)	limita	keyword	descriere
perl	V:672, I:991	841	AWK → PERL	convertește codurile sursă din AWK în PERL: a2p(1)
f2c	V:0, I:1	443	FORTTRAN → C	convertește codurile sursă din FORTRAN 77 în C/C++: f2c(1)
intel2gas	V:0, I:0	178	intel → gas	convector de la NASM (format Intel) la GNU Assembler (GAS)

Tabela 12.19: List of source code translation tools

12.9 Crearea pachetului Debian

If you want to make a Debian package, read followings.

- Cap. 2 to understand the basic package system
- Secțiune 2.7.13 to understand basic porting process
- Secțiune 9.11.4 to understand basic chroot techniques
- `debuild(1)`, și `sbuid(1)`
- Secțiune 12.5.2 for recompiling for debugging
- [Guide for Debian Maintainers](#) (the `debmake-doc` package)
- [Debian Developer's Reference](#) (the `developers-reference` package)
- [Debian Policy Manual](#) (the `debian-policy` package)

There are packages such as `debmake`, `dh-make`, `dh-make-perl`, etc., which help packaging.

Anexa A

Appendix

Here are backgrounds of this document.

A.1 The Debian maze

The Linux system is a very powerful computing platform for a networked computer. However, learning how to use all its capabilities is not easy. Setting up the LPR printer queue with a non-PostScript printer was a good example of stumble points. (There are no issues anymore since newer installations use the new CUPS system.)

There is a complete, detailed map called the "SOURCE CODE". This is very accurate but very hard to understand. There are also references called HOWTO and mini-HOWTO. They are easier to understand but tend to give too much detail and lose the big picture. I sometimes have a problem finding the right section in a long HOWTO when I need a few commands to invoke.

I hope this "Debian Reference (version 2.138)" (2026-02-24 09:02:45 UTC) provides a good starting direction for people in the Debian maze.

A.2 Copyright history

The Debian Reference was initiated by me, Osamu Aoki <osamu at debian dot org>, as a personal system administration memo. Many contents came from the knowledge I gained from [the debian-user mailing list](#) and other Debian resources.

Following a suggestion from Josip Rodin, who was very active with the [Debian Documentation Project \(DDP\)](#), "Debian Reference (version 1, 2001-2007)" was created as a part of DDP documents.

After 6 years, I realized that the original "Debian Reference (version 1)" was outdated and started to rewrite many contents. New "Debian Reference (version 2)" is released in 2008.

I have updated "Debian Reference (version 2)" to address new topics (Systemd, Wayland, IMAP, PipeWire, Linux kernel 5.10) and removed outdated topics (SysV init, CVS, Subversion, SSH protocol 1, Linux kernels before 2.5). References to Jessie 8 (2015-2020) release situation or older are mostly removed.

This "Debian Reference (version 2.138)" (2026-02-24 09:02:45 UTC) covers mostly Trixie (=stable) and Forky (=testing) Debian releases.

The tutorial contents can trace its origin and its inspiration in followings.

- "[Linux User's Guide](#)" by Larry Greenfield (December 1996)
 - obsoleted by "Debian Tutorial"

- "Debian Tutorial" by Havoc Pennington. (11 December, 1998)
 - partially written by Oliver Elphick, Ole Tetlie, James Treacy, Craig Sawyer, and Ivan E. Moore II
 - obsoleted by "Debian GNU/Linux: Guide to Installation and Usage"
- "[Debian GNU/Linux: Guide to Installation and Usage](#)" by John Goerzen and Ossama Othman (1999)
 - obsoleted by "Debian Reference (version 1)"

The package and archive description can trace some of their origin and their inspiration in following.

- "[Debian FAQ](#)" (March 2002 version, when this was maintained by Josip Rodin)

The other contents can trace some of their origin and their inspiration in following.

- "Debian Reference (version 1)" by Osamu Aoki (2001–2007)
 - obsoleted by the newer "Debian Reference (version 2)" in 2008.

The previous "Debian Reference (version 1)" was created with many contributors.

- the major contents contribution on network configuration topics by Thomas Hood
- significant contents contribution on X and VCS related topics by Brian Nelson
- the help on the build scripts and many content corrections by Jens Seidel
- extensive proofreading by David Sewell
- many contributions by the translators, contributors, and bug reporters

Many manual pages and info pages on the Debian system as well as upstream web pages and [Wikipedia](#) documents were used as the primary references to write this document. To the extent Osamu Aoki considered within the [fair use](#), many parts of them, especially command definitions, were used as phrase pieces after careful editorial efforts to fit them into the style and the objective of this document.

The gdb debugger description was expanded using [Debian wiki contents on backtrace](#) with consent by Ari Pollak, Loïc Minier, and Dafydd Harries.

Contents of the current "Debian Reference (version 2.138)" (2026-02-24 09:02:45 UTC) are mostly my own work except as mentioned above. These has been updated by the contributors too.

The author, Osamu Aoki, thanks all those who helped make this document possible.

A.3 Formatul documentului

The source of the English original document is currently written in [DocBook](#) XML files. This Docbook XML source are converted to HTML, plain text, PostScript, and PDF. (Some formats may be skipped for distribution.)