



debian

Référence Debian

Osamu Aoki

Copyright © 2013-2021 Osamu Aoki

Ce guide de référence Debian (version 2.83) (2021-09-01 02:14:16 UTC) est destiné à procurer un large aperçu du système Debian en tant que guide de l'utilisateur d'un système installé. Il couvre de nombreux aspects de l'administration du système à l'aide d'exemples de commandes de l'interpréteur pour les non-développeurs.

INDEXATION DU DOCUMENT

	<i>TITRE :</i> Référence Debian		
<i>ACTION</i>	<i>NOM</i>	<i>DATE</i>	<i>SIGNATURE</i>
RÉDIGÉ PAR	Osamu Aoki	1 ^{er} septembre 2021	

SUIVI DU DOCUMENT

INDICE	DATE	MODIFICATIONS	NOM

Table des matières

1	Didacticiels GNU/Linux	1
1.1	Bases pour la console	1
1.1.1	L'invite de l'interpréteur de commandes	1
1.1.2	The shell prompt under GUI	2
1.1.3	Compte de l'administrateur (root)	2
1.1.4	Invite de l'interpréteur de commandes pour l'administrateur	3
1.1.5	Outils graphiques d'administration du système	3
1.1.6	Consoles virtuelles	3
1.1.7	Comment quitter l'invite de l'interpréteur de commandes	3
1.1.8	Comment arrêter le système	4
1.1.9	Récupérer une console propre	4
1.1.10	Suggestions de paquets supplémentaires pour le débutant	4
1.1.11	Compte pour un utilisateur supplémentaire	5
1.1.12	Configuration de sudo	5
1.1.13	À vous de jouer	6
1.2	Système de fichiers de type UNIX	6
1.2.1	Bases concernant les fichiers UNIX	6
1.2.2	Fonctionnement interne du système de fichiers	8
1.2.3	Permissions du système de fichiers	8
1.2.4	Contrôle des permissions pour les fichiers nouvellement créés : umask	10
1.2.5	Permissions pour les groupes d'utilisateurs (group)	11
1.2.6	Horodatage	12
1.2.7	Liens	13
1.2.8	Tubes nommés (FIFO)	14
1.2.9	Sockets	14
1.2.10	Fichiers de périphériques	14
1.2.11	Fichiers spéciaux de périphériques	15
1.2.12	procfs et sysfs	15
1.2.13	tmpfs	16
1.3	Midnight Commander (MC)	16

1.3.1	Personnalisation de MC	16
1.3.2	Démarrer MC	17
1.3.3	Gestionnaire de fichiers de MC	17
1.3.4	Astuces de la ligne de commandes dans MC	17
1.3.5	Éditeur interne de MC	18
1.3.6	Visualisateur interne de MC	18
1.3.7	Possibilités de démarrage automatique de MC	18
1.3.8	Système de fichiers FTP virtuel de MC	19
1.4	L'environnement élémentaire de travail de type UNIX	19
1.4.1	L'interpréteur de commandes de connexion	19
1.4.2	Personnaliser bash	19
1.4.3	Combinaisons particulières de touches	20
1.4.4	Mouse operations	20
1.4.5	Le visualisateur de fichiers	21
1.4.6	L'éditeur de texte	22
1.4.7	Définir un éditeur de texte par défaut	22
1.4.8	Using vim	22
1.4.9	Enregistrer les actions de l'interpréteur de commandes	23
1.4.10	Commandes UNIX de base	24
1.5	La commande simple de l'interpréteur de commandes	24
1.5.1	Exécution d'une commande et variables d'environnement	26
1.5.2	La variable « \$LANG »	26
1.5.3	La variable « \$PATH »	27
1.5.4	La variable « \$HOME »	27
1.5.5	Options de la ligne de commandes	28
1.5.6	Motifs génériques (« glob ») de l'interpréteur de commandes	28
1.5.7	Valeur de retour d'une commande	29
1.5.8	Séquences de commandes typiques et redirection de l'interpréteur de commandes	29
1.5.9	Alias de commande	31
1.6	Traitement des données textuelles à la UNIX	32
1.6.1	Outils de traitement de texte d'UNIX	32
1.6.2	Expressions rationnelles	33
1.6.3	Expressions de remplacement	34
1.6.4	Substitution globale avec des expressions rationnelles	34
1.6.5	Extraire des données d'un tableau contenu dans un fichier texte	35
1.6.6	Bouts de scripts pour les tubes	37

2	Gestion des paquets Debian	38
2.1	Prérequis pour la gestion des paquets Debian	38
2.1.1	Configuration de paquets	38
2.1.2	Précautions de base	39
2.1.3	La vie avec d'éternelles mises à jour	40
2.1.4	Bases concernant l'archive Debian	40
2.1.5	Debian est totalement libre	44
2.1.6	Dépendances des paquets	45
2.1.7	Flux des événements dans la gestion d'un paquet	45
2.1.8	Première réponse aux problèmes de gestion de paquets	46
2.2	Opérations de base de la gestion des paquets	47
2.2.1	apt comparé à apt-get / apt-cache comparé à aptitude	47
2.2.2	Opérations de base de gestion des paquets en ligne de commandes	48
2.2.3	Utilisation interactive d'aptitude	48
2.2.4	Raccourcis clavier d'aptitude	50
2.2.5	Vues des paquets sous aptitude	50
2.2.6	Options de la méthode de recherche avec aptitude	52
2.2.7	Les formules d'expressions rationnelles d'aptitude	52
2.2.8	Résolution des dépendances par aptitude	54
2.2.9	Journaux d'activité des paquets	54
2.3	Exemples d'opérations avec aptitude	54
2.3.1	Afficher les paquets dont les noms correspondent à une expression rationnelle	54
2.3.2	Parcours en correspondance avec une expression rationnelle	54
2.3.3	Purger pour de bon les paquets supprimés	54
2.3.4	Toilettage de l'état d'installation automatique/manuel	55
2.3.5	Mise à jour pour l'ensemble du système	55
2.4	Opérations avancées de gestion des paquets	56
2.4.1	Opérations avancées de gestion des paquets en ligne de commandes	56
2.4.2	Vérifier les fichiers de paquets installés	58
2.4.3	Protection contre les problèmes de paquets	58
2.4.4	Rechercher dans les métadonnées du paquet	59
2.5	Fonctionnement interne de la gestion des paquets Debian	59
2.5.1	Métadonnées de l'archive	59
2.5.2	Fichier « Release » de plus haut niveau et authenticité	59
2.5.3	Fichiers « Release » au niveau de l'archive	60
2.5.4	Récupérer les métadonnées d'un paquet	61
2.5.5	État des paquets pour APT	61
2.5.6	État des paquets pour aptitude	62
2.5.7	Copies locales des paquets téléchargés	62

2.5.8	Nom de fichier d'un paquet Debian	62
2.5.9	La commande <code>dpkg</code>	63
2.5.10	La commande <code>update-alternatives</code>	63
2.5.11	Commande <code>dpkg-statoverride</code>	64
2.5.12	Commande <code>dpkg-divert</code>	64
2.6	Récupérer un système cassé	64
2.6.1	Incompatibilité avec une ancienne configuration de l'utilisateur	65
2.6.2	Différents paquets ayant des fichiers communs	65
2.6.3	Corriger les scripts cassés des paquets	65
2.6.4	Récupération avec la commande <code>dpkg</code>	66
2.6.5	Récupérer les données de sélection des paquets	66
2.7	Astuces pour la gestion des paquets	67
2.7.1	Comment obtenir des paquets Debian	67
2.7.2	Paquets venant de sources mélangées	67
2.7.3	Ajuster la version candidate	68
2.7.4	Mises à jour et rétroportages	70
2.7.5	Blocage des paquets installés par « Recommends »	70
2.7.6	Suivre <code>testing</code> avec quelques paquets d' <code>unstable</code>	71
2.7.7	Suivre <code>unstable</code> avec quelques paquets d' <code>experimental</code>	72
2.7.8	Chargement et mise à niveau automatique de paquets	72
2.7.9	Diminuer la bande passante utilisée par APT	73
2.7.10	Retour d'urgence à une version précédente (downgrade)	73
2.7.11	Qui a envoyé le paquet ?	74
2.7.12	Paquet <code>equivs</code>	74
2.7.13	Porter un paquet vers le système stable	75
2.7.14	Serveur mandataire (proxy) pour APT	75
2.7.15	Autres lectures concernant la gestion des paquets	76
3	Initialisation du système	77
3.1	Aperçu du processus d'amorçage du système	77
3.1.1	Stage 1: the <code>UEFI</code>	77
3.1.2	Étage 2 : le chargeur initial	78
3.1.3	Étage 3 : le système mini-Debian	79
3.1.4	Étage 4 : le système Debian normal	80
3.2	<code>Systemd</code> init	80
3.2.1	Nom de machine (« <code>hostname</code> »)	81
3.2.2	Le système de fichiers	82
3.2.3	Initialisation de l'interface réseau	82
3.3	Messages du noyau	82

3.4	Messages du système	82
3.5	System management	83
3.6	Other system monitors	83
3.7	Customizing systemd	83
3.7.1	Socket activation	83
3.8	Le système udev	85
3.8.1	Initialisation des modules du noyau	86
4	Authentication and access controls	87
4.1	Authentification normale d'UNIX	87
4.2	Gestion des informations des comptes et des mots de passes	89
4.3	Mot de passe de qualité	89
4.4	Créer un mot de passe chiffré	89
4.5	PAM et NSS	90
4.5.1	Fichiers de configuration auxquels accèdent PAM et NSS	91
4.5.2	Le système de gestion centralisée moderne	91
4.5.3	« Pourquoi la commande su de GNU ne gère-t-elle pas le groupe wheel »	92
4.5.4	Règle de mots de passe plus stricte	92
4.6	Sécurité de l'authentification	93
4.6.1	Mot de passe sûr avec Internet	93
4.6.2	Le shell sûr (Secure Shell)	93
4.6.3	Mesures de sécurité supplémentaires pour Internet	93
4.6.4	sécuriser le mot de passe de l'administrateur	94
4.7	Autres contrôles d'accès	94
4.7.1	sudo	95
4.7.2	PolicyKit	95
4.7.3	Restreindre l'accès à certains services du serveur	95
4.7.4	Linux security features	96
5	Configuration du réseau	97
5.1	L'infrastructure de base du réseau	97
5.1.1	Résolution du nom d'hôte	97
5.1.2	Nom de l'interface réseau	99
5.1.3	Plage d'adresses réseau du réseau local (« LAN »)	99
5.1.4	La gestion du périphérique réseau	100
5.2	Configuration moderne de réseau pour ordinateur de bureau	100
5.2.1	Outils graphiques de configuration du réseau	100
5.3	The modern network configuration without GUI	101
5.4	Configuration réseau de bas niveau	101

5.4.1	Commandes Iproute2	102
5.4.2	Opérations sûres de bas niveau sur le réseau	102
5.5	Optimisation du réseau	103
5.5.1	Rechercher le MTU optimum	103
5.5.2	Optimisation de TCP sur le réseau Internet	104
5.6	Infrastructure de netfilter	104
6	Applications réseau	106
6.1	Navigateurs Web	106
6.1.1	Configuration du navigateur	107
6.2	Le système de courrier électronique	107
6.2.1	Bases du courrier électronique	108
6.2.2	Modern mail service limitation	108
6.2.3	Historic mail service expectation	109
6.2.4	Agent de transport de courrier électronique (« MTA »)	109
6.2.4.1	Configuration d'exim4	109
6.2.4.2	Configuration de postfix avec SASL	111
6.2.4.3	Configuration de l'adresse de courriel	112
6.2.4.4	Opération de base du MTA	113
6.3	Le serveur et les utilitaires d'accès à distance (SSH)	113
6.3.1	Bases de SSH	114
6.3.2	User name on the remote host	114
6.3.3	Se connecter sans mot de passe distant	115
6.3.4	Clients SSH exotiques	115
6.3.5	Configurer ssh-agent	115
6.3.6	Sending a mail from a remote host	116
6.3.7	Redirection de port pour un tunnel SMTP/POP3	116
6.3.8	Comment arrêter le système distant par SSH	116
6.3.9	Résoudre les problèmes avec SSH	116
6.4	Le serveur et les utilitaires d'impression	117
6.5	Autres serveurs d'applications réseau	117
6.6	Autres clients d'applications réseau	118
6.7	Le diagnostic des démons du système	118
7	GUI System	120
7.1	GUI desktop environment	120
7.2	GUI communication protocol	121
7.3	GUI infrastructure	122
7.4	GUI applications	122

7.5	Fonts	122
7.5.1	Polices de base	124
7.5.2	Font rasterization	125
7.6	Sandbox	125
7.7	Remote desktop	126
7.8	X server connection	126
7.8.1	X server local connection	126
7.8.2	X server remote connection	128
7.8.3	X server chroot connection	128
7.9	Presse-papier	128
8	I18N et L10N	130
8.1	Les paramètres linguistiques (« locale »)	130
8.1.1	Justification de l'utilisation d'UTF-8 dans les paramètres linguistiques	130
8.1.2	Reconfiguration des paramètres linguistiques	131
8.1.3	Coder les noms de fichiers	131
8.1.4	Messages et documentation traduits	132
8.1.5	Effet des paramètres linguistiques	132
8.2	L'entrée clavier	133
8.2.1	The keyboard input for Linux console and X Window	133
8.2.2	The keyboard input for Wayland	133
8.2.3	Prise en charge de la méthode d'entrée avec iBus	133
8.2.4	Un exemple pour le japonais	133
8.3	L'affichage de sortie	134
8.4	East Asian Ambiguous Character Width Characters	134
9	Astuces du système	135
9.1	The console tips	135
9.1.1	Enregistrer proprement l'activité de la console	135
9.1.2	Le programme screen	136
9.1.3	Navigating around directories	136
9.1.4	Readline wrapper	137
9.2	Personnaliser vim	137
9.2.1	Customizing vim with internal features	137
9.2.2	Customizing vim with external packages	138
9.3	Enregistrer et présenter des données	139
9.3.1	Le démon de journal	139
9.3.2	Analyseur de journaux	139
9.3.3	Affichage personnalisé des données de texte	139

9.3.4	Affichage personnalisé de la date et de l'heure	139
9.3.5	Écho colorisé de l'interpréteur de commandes	140
9.3.6	Commandes colorisées	140
9.3.7	Enregistrer l'activité de l'éditeur pour des répétitions complexes	141
9.3.8	Enregistrer l'image graphique d'une application X	141
9.3.9	Enregistrer les modifications dans des fichiers de configuration	141
9.4	Surveiller, contrôler et démarrer l'activité des programmes	142
9.4.1	Temps d'un processus	142
9.4.2	La priorité d'ordonnancement	143
9.4.3	La commande ps	143
9.4.4	La commande top	143
9.4.5	Afficher les fichiers ouverts par un processus	143
9.4.6	Tracer l'activité d'un programme	144
9.4.7	Identification des processus qui utilisent des fichiers ou des sockets	144
9.4.8	Répéter une commande avec un intervalle constant	144
9.4.9	Répéter une commande en bouclant entre des fichiers	144
9.4.10	Lancer un programme depuis l'interface graphique	145
9.4.11	Personnaliser le programme à lancer	146
9.4.12	Tuer un processus	147
9.4.13	Planifier des tâches qui s'exécutent une fois	147
9.4.14	Planifier des tâches qui s'exécutent régulièrement	148
9.4.15	touche Alt-SysRq	148
9.5	Astuces de maintenance du système	149
9.5.1	Qui se trouve sur le système ?	149
9.5.2	Prévenir tout le monde	149
9.5.3	Identification du matériel	149
9.5.4	Configuration matérielle	151
9.5.5	Heure système et matérielle	151
9.5.6	Configuration du terminal	151
9.5.7	L'infrastructure de gestion du son	152
9.5.8	Désactiver l'économiseur d'écran	152
9.5.9	Désactiver les bips	152
9.5.10	Utilisation de la mémoire	153
9.5.11	Vérification de la sécurité et de l'intégrité du système	153
9.6	Astuces relatives au stockage des données	154
9.6.1	Utilisation de l'espace disque	154
9.6.2	Configuration de la partition du disque	155
9.6.3	Accès à une partition en utilisant l'UUID	155
9.6.4	LVM2	156

9.6.5	Configuration de systèmes de fichiers	156
9.6.6	Création et vérification de l'intégrité d'un système de fichiers	157
9.6.7	Optimisation du système de fichiers à l'aide des options de montage	157
9.6.8	Optimisation du système de fichiers à l'aide du superbloc	158
9.6.9	Optimisation du disque dur	158
9.6.10	Optimisation du SSD	158
9.6.11	Utiliser SMART pour prédire les défaillances des disques durs	158
9.6.12	Indication du répertoire de stockage temporaire à l'aide de \$TMPDIR	159
9.6.13	Étendre l'espace de stockage utile à l'aide de LVM	159
9.6.14	Extension de l'espace de stockage en montant une autre partition	159
9.6.15	Extension de l'espace de stockage en remontant un autre répertoire	159
9.6.16	Expansion of usable storage space by overlay-mounting another directory	160
9.6.17	Extension de l'espace utilisable à l'aide de liens symboliques	160
9.7	Le fichier image du disque	160
9.7.1	Créer le fichier image du disque	160
9.7.2	Écrire directement sur le disque	161
9.7.3	Monter le fichier image du disque	161
9.7.4	Nettoyage d'un fichier image du disque	162
9.7.5	Réaliser le fichier image d'un disque vide	163
9.7.6	Créer un fichier image ISO9660	163
9.7.7	Écriture directe sur CD/DVD-R/RW	164
9.7.8	Monter le fichier image ISO9660	164
9.8	Les données binaires	164
9.8.1	Afficher et éditer des données binaires	165
9.8.2	Manipulation des fichiers sans monter le disque	165
9.8.3	Redondance des données	165
9.8.4	Récupération de fichiers de données et analyse par autopsie	165
9.8.5	Éclater un gros fichier en petits fichiers	167
9.8.6	Effacer le contenu d'un fichier	167
9.8.7	Fichiers fictifs	167
9.8.8	Effacer l'ensemble du disque dur	167
9.8.9	Effacer l'ensemble du disque dur	168
9.8.10	Récupérer des fichiers supprimés mais encore ouverts	168
9.8.11	Rechercher tous les liens physiques	169
9.8.12	Consommation d'espace disque invisible	169
9.9	Astuces de chiffrement des données	169
9.9.1	Chiffrement des disques amovibles à l'aide de dm-crypt/LUKS	170
9.9.2	Partition d'échange chiffrée avec dm-crypt	170
9.9.3	Monter des disques amovibles chiffrés à l'aide de dm-crypt/LUKS	171

9.10	Le noyau	171
9.10.1	Paramètres du noyau	171
9.10.2	En-têtes du noyau	171
9.10.3	Compiler le noyau et les modules associés	172
9.10.4	Compiler les sources du noyau : recommandations de l'équipe en charge du noyau Debian	173
9.10.5	Pilotes de matériel et microprogramme	173
9.11	Système virtualisé	174
9.11.1	Virtualization and emulation tools	174
9.11.2	Étapes de la virtualisation	174
9.11.3	Monter le fichier image du disque virtuel	176
9.11.4	Système protégé (chroot)	176
9.11.5	Systèmes de bureaux multiples	177
10	Gestion des données	179
10.1	Partager, copier et archiver	179
10.1.1	Outils d'archivage et de compression	181
10.1.2	Outils de copie et de synchronisation	181
10.1.3	Idiomes pour les archives	182
10.1.4	Idiomes pour la copie	182
10.1.5	Idiomes pour la sélection de fichiers	183
10.1.6	Support d'archive	184
10.1.7	Périphériques d'enregistrement amovibles	184
10.1.8	Choix de système de fichiers pour les données partagées	185
10.1.9	Partage de données au travers du réseau	187
10.2	Sauvegarde et restauration	187
10.2.1	Suites d'utilitaires de sauvegarde	188
10.2.2	Personal backup	190
10.3	Infrastructure de sécurité des données	190
10.3.1	Gestion de clés pour GnuPG	192
10.3.2	Utilisation de GnuPG sur des fichiers	192
10.3.3	Utiliser GnuPG avec Mutt	192
10.3.4	Utiliser GnuPG avec Vim	192
10.3.5	La somme de contrôle MD5	192
10.4	Outils pour fusionner le code source	194
10.4.1	Extraire des différences pour des fichiers sources	195
10.4.2	Fusionner les mises à jour des fichiers source	195
10.4.3	Interactive merge	195
10.5	Systèmes de contrôle de version	195
10.5.1	Configuration du client Git	195
10.5.2	Commandes de Git	197
10.5.3	Git pour l'enregistrement de l'historique de la configuration	198
10.5.4	Références de Git	198

11 Conversion de données	199
11.1 Outils de conversion de données textuelles	199
11.1.1 Convertir un fichier texte avec iconv	199
11.1.2 Vérifier que les fichiers sont codés en UTF-8 avec iconv	201
11.1.3 Convertir les noms de fichiers avec iconv	201
11.1.4 Convertir les fins de ligne (EOL)	201
11.1.5 Convertir les tabulations (TAB)	202
11.1.6 Éditeurs avec conversion automatique	202
11.1.7 Extraire du texte brut	203
11.1.8 Mettre en évidence et formater des données en texte brut	204
11.2 Données XML	204
11.2.1 Conseils de base pour XML	204
11.2.2 Traitement XML	205
11.2.3 Extraire des données XML	207
11.2.4 The XML data lint	207
11.3 Composition	207
11.3.1 Composition roff	208
11.3.2 TeX/LaTeX	208
11.3.3 Imprimer convenablement une page de manuel	209
11.3.4 Créer une page de manuel	209
11.4 Données imprimables	209
11.4.1 Ghostscript	209
11.4.2 Fusionner deux fichiers PS ou PDF	210
11.4.3 Utilitaires pour les données imprimables	210
11.4.4 Imprimer avec CUPS	210
11.5 La conversion de données de courrier électronique	210
11.5.1 Bases concernant les données de courrier électronique	212
11.6 Outils de données graphiques	212
11.7 Diverses conversions de données	214
12 Programmation	215
12.1 Les scripts de l'interpréteur de commande	215
12.1.1 Compatibilité de l'interpréteur de commandes avec POSIX	216
12.1.2 Paramètres de l'interpréteur de commandes	216
12.1.3 Opérateurs conditionnels de l'interpréteur	217
12.1.4 Boucles de l'interpréteur de commandes	218
12.1.5 Séquence de traitement de la ligne de commandes de l'interpréteur	218
12.1.6 Programmes utilitaires pour les scripts de l'interpréteur de commandes	219
12.2 Scripting in interpreted languages	220

12.2.1	Debugging interpreted language codes	220
12.2.2	GUI program with the shell script	221
12.2.3	Folie de courts scripts en Perl	221
12.3	Coding in compiled languages	222
12.3.1	C	222
12.3.2	Programme simple en C (gcc)	223
12.3.3	Flex -- un meilleur Lex	223
12.3.4	Bison -- un meilleur Yacc	224
12.4	Outils d'analyse du code statique	224
12.5	Déboguer	224
12.5.1	Exécution de base de gdb	224
12.5.2	Déboguer un paquet Debian	226
12.5.3	Obtenir une trace	227
12.5.4	Commandes avancées de gdb	227
12.5.5	Vérifier les dépendances avec les bibliothèques	227
12.5.6	Dynamic call tracing tools	228
12.5.7	Déboguer les erreurs de X	228
12.5.8	Outils de détection des fuites de mémoire	228
12.5.9	Désassembler un binaire	228
12.6	Build tools	229
12.6.1	Make	229
12.6.2	Autotools	230
12.6.2.1	Compiler et installer un programme	230
12.6.2.2	Désinstaller un programme	230
12.6.3	Meson	230
12.7	Web	231
12.8	La conversion du code source	231
12.9	Créer un paquet Debian	231
A	Annexe	232
A.1	Le labyrinthe de Debian	232
A.2	Historique du Copyright	232
A.3	Format du document	233

Liste des tableaux

1.1	Liste de paquets de programmes intéressants en mode texte	4
1.2	Liste de paquets de documentation	5
1.3	Utilisation des répertoires-clés	7
1.4	Liste des premiers caractères de la sortie de « <code>ls -l</code> » :	9
1.5	Mode numérique des permissions de fichiers dans les commandes <code>chmod(1)</code>	10
1.6	Exemples de valeurs de umask	10
1.7	Liste des groupes importants fournis par le système pour l'accès aux fichiers	11
1.8	Liste des groupes importants fournis par le système pour l'exécution de commandes particulières	11
1.9	Liste des types d'horodatage	12
1.10	Liste des fichiers spéciaux de périphériques	15
1.11	Touches de raccourcis de MC	17
1.12	Réaction à la touche Entrée dans MC	18
1.13	Liste d'interpréteurs de commandes (« shells »)	19
1.14	Liste des raccourcis clavier de bash	21
1.15	List of mouse operations and related key actions on Debian	21
1.16	List of basic Vim key strokes	23
1.17	Liste des commandes UNIX de base	25
1.18	Les trois parties des paramètres linguistiques	26
1.19	Liste des recommandations de paramètres linguistiques	26
1.20	Afficher les valeurs de la variable « <code>\$HOME</code> »	28
1.21	Motifs génériques d'expansion du nom de fichier de l'interpréteur de commandes	28
1.22	Codes de retour de la commande	29
1.23	Idiomes des commandes de l'interpréteur	30
1.24	Descripteurs de fichier prédéfinis	31
1.25	Métacaractères pour BRE et ERE	33
1.26	Expressions de remplacement	34
1.27	Liste de parties de scripts pour enchaîner (piping) les commandes	37
2.1	Liste des outils de gestion des paquets de Debian	39
2.2	Liste des sites d'archive de Debian	41

2.3	Liste des sections de l'archive de Debian	42
2.4	Relation entre version et nom de code	43
2.5	Liste de sites web clés pour résoudre les problèmes avec un paquet particulier	47
2.6	Opérations de base de gestion des paquets avec la ligne de commandes en utilisant apt(8) , aptitude(8) , apt-get(8) et apt-cache(8)	49
2.7	Options importantes de la commande aptitude(8)	49
2.8	Liste des raccourcis clavier d' aptitude	50
2.9	Liste des vues d' aptitude	51
2.10	Classement par catégories des vues de paquets standard	51
2.11	Liste des formules d'expressions rationnelles d' aptitude	53
2.12	Fichiers journaux de l'activité des paquets	54
2.13	Liste des opérations avancées de gestion des paquets	57
2.14	Contenu des métadonnées de l'archive Debian	59
2.15	Structure du nom des paquets Debian :	62
2.16	Caractères utilisables pour chacune des composantes des noms de paquets Debian	62
2.17	Fichiers particuliers créés par dpkg	63
2.18	Liste de valeurs remarquables des priorités d'épinglage (Pin-Priority) pour la technique d'épinglage apt (« apt-pinning ») :	70
2.19	Liste des outils de proxy spécifiques à l'archive Debian	75
3.1	Liste des chargeurs initiaux	78
3.2	The meaning of the menu entry of the above part of /boot/grub/grub.cfg	79
3.3	Liste d'utilitaires d'amorçage initial pour le système Debian :	80
3.4	Liste des niveaux d'erreur du noyau	82
3.5	List of typical journalctl command snippets	83
3.6	List of typical systemctl command snippets	84
3.7	List of other monitoring command snippets under systemd	85
4.1	3 fichiers de configuration importants pour pam_unix(8)	87
4.2	Contenu de la seconde entrée de « /etc/passwd »	88
4.3	Liste des commandes servant à gérer les informations des comptes	89
4.4	Liste d'outils permettant de générer des mots de passe	90
4.5	Liste des paquets importants des systèmes PAM et NSS	90
4.6	Liste des fichiers de configuration auxquels PAM et NSS accèdent	91
4.7	Liste des services et ports sûrs et non sûrs	93
4.8	Liste des outils fournissant des mesures de sécurité supplémentaires	94
5.1	Liste des outils de configuration du réseau	98
5.2	Liste des plages d'adresses de réseau	100
5.3	Table de conversion depuis les commandes obsolètes net-tools vers les nouvelles commandes iproute2	102
5.4	Liste des commandes de réseau de bas niveau	102

5.5	Liste des outils d'optimisation du réseau	103
5.6	Lignes directrices pour une valeur optimum de MTU	104
5.7	Liste d'outils de pare-feu	105
6.1	Liste de navigateurs web	106
6.2	Liste des paquets de greffons pour les navigateurs	107
6.3	Liste d'agents de courrier électronique de l'utilisateur (MUA)	108
6.4	List of basic mail transport agent related packages	110
6.5	Liste des pages de manuel importantes de postfix	111
6.6	Liste des fichiers de configuration liés aux adresses de courriel	112
6.7	Liste des opérations de base du MTA	113
6.8	Liste des serveurs et des utilitaires d'accès à distance	114
6.9	Liste des fichiers de configuration de SSH	114
6.10	Liste d'exemples de démarrage du client SSH	115
6.11	Liste des clients SSH libres pour d'autres plateformes	115
6.12	Liste des serveurs et utilitaires d'impression.	117
6.13	Liste d'autres serveurs d'applications réseau	118
6.14	Liste de clients d'applications réseau	119
6.15	Liste des RFC courantes	119
7.1	List of desktop environment	120
7.2	List of notable GUI infrastructure packages	122
7.3	List of notable GUI applications	123
7.4	List of notable TrueType and OpenType fonts	124
7.5	List of notable font environment and related packages	125
7.6	List of notable sandbox environment and related packages	126
7.7	List of notable remote access server	127
7.8	Liste des méthodes de connexion au serveur X	127
7.9	List of programs related to manipulating character clipboard	129
8.1	List of IBus and its engine packages	134
9.1	List of programs to support console activities	135
9.2	Liste des raccourcis clavier de screen	137
9.3	Information on the initialization of vim	138
9.4	Liste des analyseurs de journaux système	139
9.5	Display examples of time and date for the "ls -l" command with the time style value	140
9.6	Liste des outils de manipulation d'images	141
9.7	List of packages which can record configuration history	142
9.8	Liste des outils de surveillance et de contrôle de l'activité des programmes	142

9.9	Liste des valeurs de politesse pour la priorité d'ordonnancement	143
9.10	Liste des styles de la commande ps	143
9.11	Liste des signaux couramment utilisés avec la commande kill	147
9.12	List of notable SAK command keys	148
9.13	Listes des outils d'identification du matériel	150
9.14	Liste des outils de configuration du matériel	150
9.15	Liste des paquets son	152
9.16	Liste des commandes pour désactiver l'économiseur d'écran	153
9.17	Taille mémoire affichée	153
9.18	Liste d'outils pour la vérification de la sécurité et de l'intégrité du système	154
9.19	Listes de paquets de gestion de la partition du disque	155
9.20	Liste des paquets de gestion des systèmes de fichiers	156
9.21	Liste des paquets permettant de visualiser et d'éditer des données binaires	165
9.22	Liste des paquets pour manipuler les fichiers sans monter le disque	165
9.23	Liste d'outils pour ajouter des données de redondance aux fichiers	166
9.24	Liste de paquets pour la récupération de données et l'analyse par autopsie	166
9.25	Liste d'utilitaires de chiffrement des données	170
9.26	Liste des paquets-clés à installer pour la compilation du noyau sur un système Debian	172
9.27	Liste des outils de virtualisation	175
10.1	Liste des outils d'archivage et de compression	180
10.2	Liste des outils de copie et de synchronisation	181
10.3	Liste de choix de systèmes de fichiers pour des périphériques amovibles avec des scénarios typiques d'utilisation	186
10.4	Liste des services réseau à choisir avec le scénario typique d'utilisation	187
10.5	Liste de suites d'utilitaires de sauvegarde	189
10.6	Liste des outils d'une infrastructure de sécurité des données	190
10.7	Liste des commandes de GNU Privacy Guard pour la gestion des clés	191
10.8	Liste de la signification des codes de confiance	191
10.9	Liste des commandes de GNU Privacy Guard sur des fichiers	193
10.10	Liste d'outils destinés à fusionner du code source	194
10.11	Liste d'outils pour les systèmes de contrôle de version	196
10.12	Comparaison des commandes natives de VCS	196
10.13	Liste des paquets et des commandes relatifs à git	197
11.1	Liste des outils de conversion de texte	199
11.2	Liste de valeurs de codage et leur utilisation	200
11.3	Liste des styles d'EOL pour différentes plateformes	202
11.4	Liste des commande de conversion de TAB des paquets <code>bsdmainutils</code> et <code>coreutils</code>	202
11.5	Liste d'outils pour extraire des données en texte brut	203

11.6	Liste des outils pour mettre en évidence des données de texte brut	204
11.7	Liste des entités XML prédéfinies	205
11.8	Liste d'outils XML	206
11.9	Liste des outils DSSSL	206
11.10	Liste d'outils d'extraction de données XML	207
11.11	Liste d'outils d'impression élégante du XML	207
11.12	Liste des outils de typographie	208
11.13	Liste de paquets facilitant la création de pages de manuel	209
11.14	Liste des interpréteurs Ghostscript PostScript	210
11.15	Liste des utilitaires pour les données imprimables	211
11.16	Liste de paquets facilitant la conversion de données de courrier électronique	211
11.17	Liste d'outils pour les données graphiques	213
11.18	Liste d'outils divers de conversion de données	214
12.1	Liste de bashismes typiques	216
12.2	Liste des paramètres de l'interpréteur de commandes	216
12.3	Liste des expansions de paramètre de l'interpréteur	217
12.4	Liste des substitutions-clés de paramètres de l'interpréteur	217
12.5	Liste des opérateurs de comparaison dans les expressions conditionnelles	218
12.6	Liste des opérateurs de comparaison de chaîne de caractères dans les expressions conditionnelles	218
12.7	Liste des paquets comportant des petits programmes utilitaires pour les scripts de l'interpréteur de commandes	219
12.8	List of interpreter related packages	220
12.9	List of dialog programs	221
12.10	List of compiler related packages	222
12.11	Liste de générateurs d'analyseur LALR compatible avec Yacc	224
12.12	Liste des outils d'analyse du code statique :	225
12.13	List of debug packages	225
12.14	Liste des commandes avancées de gdb	227
12.15	Liste des outils de détection des fuites de mémoire	228
12.16	List of build tool packages	229
12.17	Liste des variables automatiques de make	229
12.18	Liste de l'expansion des variables de make	229
12.19	Liste des outils de conversion de code source	231

Résumé

Ce livre est libre ; vous pouvez le redistribuer et le modifier selon les termes de la Licence Publique Générale GNU (« GNU GPL ») avec n'importe quelle version compatible avec les Règles des Logiciels Libres selon Debian (DFSG).

Préface

Cette [Debian Reference \(version 2.83\)](#) (2021-09-01 02:14:16 UTC) est destinée à fournir un large aperçu de l'administration d'un système Debian, sous la forme d'un guide utilisateur de post-installation.

Le lecteur cible est quelqu'un qui désire apprendre les scripts de l'interpréteur de commandes mais qui ne souhaite pas lire tous les sources en C pour comprendre le fonctionnement du système [GNU /Linux](#).

Pour le guide d'installation, voir :

- [Debian GNU/Linux : guide d'installation pour le système stable](#)
- [Debian GNU/Linux : guide d'installation pour la version en cours de test](#)

Clause de non responsabilité

Toute garantie est rejetée. Toutes les marques déposées sont la propriété de leurs détenteurs respectifs.

Le système Debian lui-même est une cible mouvante. Cela rend difficile le maintien à jour et l'exactitude de sa documentation. Bien que la version instable (« unstable ») du système Debian ait été utilisée pour écrire ce document, certaines parties peuvent être dépassées au moment où vous lisez cela.

Veuillez prendre ce document comme une référence secondaire. Ce document ne remplace aucun des guides autorisés. L'auteur et les contributeurs ne pourront être tenus pour responsables des conséquences des erreurs, omissions ou ambiguïtés que comporte ce document.

Ce qu'est Debian

Le [Projet Debian](#) est une association de personnes qui ont fait cause commune afin de créer un système d'exploitation libre. Sa distribution est caractérisée par :

- un engagement dans la liberté du logiciel : [Le contrat social Debian et les Lignes directrices du logiciel libre selon Debian \(DFSG\)](#) ;
- Bénévolat non rémunéré distribué sur Internet
- Grand nombre de logiciels pré-compilés de haute qualité
- l'accent sur la stabilité et la sécurité avec un accès facile aux mises à jour de sécurité ;
- Accent mis sur une mise à niveau en douceur vers les dernières versions des logiciels par l'utilisation des archives `unstable` et `testing`
- la prise en charge d'un grand nombre d'architectures matérielles.

Les éléments des logiciels libres de Debian proviennent de [GNU](#), [Linux](#), [BSD](#), [X](#), [ISC](#), [Apache](#), [Ghostscript](#), [Common UNIX Printing System](#), [Samba](#), [GNOME](#), [KDE](#), [Mozilla](#), [LibreOffice.org](#), [Vim](#), [TeX](#), [LaTeX](#), [DocBook](#), [Perl](#), [Python](#), [Tcl](#), [Java](#), [Ruby](#), [PHP](#), [Berkeley DB](#), [MariaDB](#), [PostgreSQL](#), [Exim](#), [Postfix](#), [Mutt](#), [FreeBSD](#), [OpenBSD](#), [Plan 9](#) et de nombreux autres projets de logiciels libres indépendants. Debian intègre cette diversité de logiciels libres dans un seul système.

À propos de ce document

Règles

Les règles suivantes ont été suivies lors de la compilation de ce document.

- fournir un aperçu et passer les cas marginaux (**vue d'ensemble**) ;
- le garder court et simple (**KISS**) ;
- ne pas réinventer la roue (utiliser des liens pointant vers **les références existantes**) ;
- mettre l'accent sur les outils n'ayant pas d'interface graphique ou en mode console (utiliser **des exemples en ligne de commande**) ;
- Soyez objectif (Utilisez [popcon](#), etc.)

ASTUCE

J'ai essayé d'éclaircir les aspects hiérarchiques et les niveaux les plus bas du système.

Exigences de départ



AVERTISSEMENT

On attend de vous que vous fassiez des efforts pour rechercher des réponses par vous-même au-delà de cette documentation. Ce document ne donne que des points de départs efficaces.

Vous devez chercher vous-même une solution dans les sources primaires.

- [Le cahier de l'administrateur Debian](#)
- Le site de Debian <https://www.debian.org> pour des informations générales
- la documentation dans le répertoire « `/usr/share/doc/nom_paquet` » :
- les **pages de manuel (manpage)** de style UNIX : « `dpkg -L nom_paquet | grep '/man/man.*/'` » ;
- les **pages info** de style GNU : « `dpkg -L nom_paquet | grep '/info/'` » ;
- les signalements de bogues : http://bugs.debian.org/nom_paquet ;
- Le Wiki Debian en <https://wiki.debian.org/> pour les sujets spécifiques ou changeants
- les HOWTOs du projet de documentation Linux (TLPD) en <http://tldp.org/>
- « The Single UNIX Specification » depuis la page d'entrée « The UNIX System » de l'Open Group en <http://www.unix.org/> ;
- L'encyclopédie libre de Wikipedia à <http://www.wikipedia.org/>

Note

Pour accéder à une documentation détaillée, vous devrez installer les paquets de documentation qui correspondent au nom du paquet avec le suffixe « `-doc` ».

Conventions

Ce document fournit des informations en utilisant le style de présentation simplifié suivant, avec des exemples de commandes de l'interpréteur `bash(1)`.

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

Ces invites de l'interpréteur de commandes permettent de distinguer le compte utilisé et correspondent à la définition des variables d'environnement « `PS1='\$ '` » et « `PS2=' '` ». Ces valeurs ont été choisies pour ce document dans un but de lisibilité, elles ne sont pas représentatives d'un système réel.

All command examples are run under the English locale "LANG=en_US.UTF8". Please don't expect the placeholder strings such as *command-in-root-account* and *command-in-user-account* to be translated in command examples. This is an intentional choice to keep all translated examples to be up-to-date.

Note

Consultez la signification des variables d'environnement « `$PS1` » et « `$PS2` » dans `bash(1)`.

L'**action** demandée à l'administrateur du système est écrite sous forme d'une phrase impérative, par exemple « Pressez la touche Entrée après la saisie de chaque chaîne de commande dans l'interpréteur de commandes. »

La colonne de **description** ou similaire dans le tableau peut contenir une **locution nominale** selon la [convention de description courte du paquet](#) qui supprime les articles se trouvant en tête tels que « un » et « le » (« a », « the »). Elle peut contenir une phrase à l'infinitif comme **locution nominale** sans le « to » de tête (NdT : en français, une phrase impérative commençant par un verbe à l'infinitif), suivie de la description courte de la commande selon la convention des pages de manuel. Cela peut sembler bizarre à certaines personnes mais ce sont les choix voulus par l'auteur afin de garder cette documentation la plus simple possible. Ces **locutions nominales**, selon cette convention de description courte, n'ont pas de majuscule à la première lettre et ne se terminent pas par un point.

Note

Les noms propres, y compris les noms de commandes, gardent leur casse indépendamment de l'endroit où ils se trouvent.

Un **morceau de commande** cité dans le paragraphe d'un texte sera signalé par une police « typewriter » (machine à écrire) entre guillemets, comme par exemple « `aptitude safe-upgrade` ».

Les **données textuelles** d'un fichier de configuration citées dans un paragraphe seront signalées par une police de type machine à écrire entre guillemets, comme par exemple « `deb-src` ».

Une **commande** sera indiquée par son nom dans la police machine à écrire suivi, de manière facultative, par le numéro de section de la page de manuel entre parenthèses, comme par exemple `bash(1)`. Vous êtes encouragé à rechercher des informations complémentaires en entrant :

```
$ man 1 bash
```

Une **page de manuel** est indiquée par son nom dans la police machine à écrire suivie, entre parenthèses, du numéro de la section de la page de manuel, comme par exemple, `sources.list (5)`. Vous êtes encouragé à rechercher des informations complémentaires en entrant :

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

Une **page info** est indiquée par un fragment entre guillemets de la commande correspondante dans la police machine à écrire, comme par exemple, « `info make` ». Vous êtes encouragé à rechercher des informations complémentaires en entrant :

```
$ info make
```

Un **nom de fichier** est indiqué par une police machine à écrire entre guillemets, comme par exemple, « `/etc/passwd` ». En ce qui concerne les fichiers de configuration, vous êtes encouragé à rechercher des informations complémentaires en entrant :

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

Un **nom de répertoire** est indiqué par la police machine à écrire entre guillemets, comme par exemple, « `/etc/apt/` ». Vous êtes encouragé à explorer son contenu en tapant ce qui suit.

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

Un **nom de paquet** est indiqué par son nom dans la police machine à écrire, comme par exemple `vim`. Vous êtes encouragé à rechercher des informations complémentaires en entrant :

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

On peut indiquer l'emplacement d'une **documentation** par son nom de fichier dans la police machine à écrire entre guillemets, comme par exemple « `/usr/share/doc/sysv-rc/README.runlevels.gz` » et « `/usr/share/doc/base-passwd/users-and-groups.txt.gz` » ou par son **URL**, comme par exemple <https://www.debian.org>. Vous êtes encouragé à lire la documentation en entrant.

```
$ 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"
```

Une **variable d'environnement** est indiquée par son nom précédé d'un « `$` » dans la police machine à écrire entre guillemets, comme par exemple « `$TERM` ». Vous êtes encouragé à obtenir sa valeur actuelle en entrant :

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

Le concours de popularité (« popcon »)

Les données du [popcon](#) sont présentées comme une manière objective de mesurer la popularité de chaque paquet. Elles sont téléchargées depuis 2021-08-14 17:04:37 UTC et contiennent un total de 196036 soumissions de rapports pour 183707 paquets binaires et 26 architectures.

Note

Vous remarquerez que l'archive `amd64 unstable` ne contient actuellement que 63210 paquets. Les données de popularité contiennent des rapports venant de nombreuses installations anciennes.

Le numéro de « popcon », précédé par un « `V` » pour « votes », est calculé par « $1000 * (\text{soumissions popcon pour le paquet exécuté récemment sur le PC}) / (\text{nombre total des soumissions popcon})$ ».

Le numéro de « popcon », précédé par un « `I` » pour « installations », est calculé par « $1000 * (\text{soumissions popcon pour le paquet installé sur le PC}) / (\text{nombre total des soumissions popcon})$ ».

Note

Les données numériques de popcon ne doivent pas être considérées comme des mesures absolues de l'importance des paquets. Il y a de nombreux facteurs qui peuvent fausser les statistiques. Par exemple, certains systèmes participant au popcon ont monté des répertoires tels que « `/bin` » avec l'option « `noatime` » afin d'améliorer les performances du système et ont, de ce fait, désactivé le « vote » de tels systèmes.

Taille du paquet

Les données correspondant à la taille du paquet sont aussi présentées comme une mesure objective de chacun des paquets. Elles sont basées sur « `Installed-Size` » (« taille installée ») indiquée par la commande « `apt-cache show` » ou « `aptitude show` » (actuellement sur l'architecture `amd64` et pour la version `unstable`). La taille est indiquée en Kio ([kibiocet](#) = unité pour 1024 octets).

Note

Si un paquet a une taille de faible valeur numérique, cela peut vouloir dire que le paquet de la version `unstable` est un paquet « dummy » qui permet l'installation par dépendances d'autres paquets ayant un contenu significatif. Un paquet dummy permet une transition en douceur ou un éclatement du paquet.

Note

Une taille de paquet suivie de « (*) » indique que la version unstable du paquet est absente et que la taille du paquet venant de la version experimental a été utilisée en remplacement.

Signalements de bogues concernant ce document

Si vous découvrez des problèmes dans ce document, veuillez signaler les bogues du paquet `debian-reference` en utilisant `reportbug(1)`. Veuillez inclure des suggestions de correction en lançant « `diff -u` » sur la version en texte brut ou sur le source.

Rappels pour les nouveaux utilisateurs

Voici quelques rappels pour les nouveaux utilisateurs :

- Sauvegardez vos données
- Sécurisez votre mot de passe et vos clés de sécurité
- [KISS \(keep it simple stupid\)](#)
 - Ne modifiez pas trop votre système
- Lire vos fichiers journaux
 - La **PREMIÈRE** erreur est celle qui compte
- [LLSM \(Lisez Le Super Manuel\)](#)
- Faites une recherche sur Internet avant de poser des questions
- N'utilisez pas le compte root quand cela n'est pas nécessaire
- Ne modifiez pas le système de gestion des paquets
- Ne tapez rien que vous ne compreniez pas.
- Ne modifiez pas les permissions des fichiers (avant un examen complet de la sécurité)
- Ne pas quitter l'interpréteur root avant de **TESTER** les modifications.
- Ayez toujours un support d'amorçage alternatif (clé USB, CD, ...)

Quelques citations pour les nouveaux utilisateurs

Voici quelques citations intéressantes provenant de la liste de diffusion Debian qui pourraient aider les nouveaux utilisateurs à y voir plus clair.

- « This is UNIX. It gives you enough rope to hang yourself. » --- Miquel van Smoorenburg <miquels@cistron.nl> (C'est UNIX. Il vous donne assez de corde pour vous pendre vous-même).
- « UNIX IS user friendly... It's just selective about who its friends are. » --- Tollef Fog Heen <tollef@add.no> (UNIX est l'ami de l'utilisateur... Il choisit juste qui sont ses amis).

L'article Wikipédia « [Philosophie d'Unix](#) » contient une liste de citations intéressantes.

Chapitre 1

Didacticiels GNU/Linux

Je pense qu'apprendre un système d'exploitation est comme apprendre une nouvelle langue étrangère. Bien que les livres de didacticiels et de documentation soient utiles, vous devrez pratiquer vous-même. Pour vous aider à vous lancer en douceur, je vais développer quelques points fondamentaux.

La puissance de la conception de [Debian GNU/Linux](#) vient du système d'exploitation [UNIX](#), c'est-à-dire un système d'exploitation [multi-utilisateurs](#), [multi-tâches](#). Vous devrez apprendre à tirer parti de la puissance de ces fonctionnalités et des similitudes entre UNIX et GNU/Linux.

N'écartez pas des textes orientés UNIX en ne vous reposant que sur les textes GNU/Linux, cela vous priverait de beaucoup d'informations utiles.

Note

Si vous avez utilisé pendant un certain temps des systèmes [ressemblant à UNIX](#) avec des outils en ligne de commande, vous connaissez sans doute tout ce que j'explique ici. Vous pourrez utiliser ce document pour actualiser vos connaissances.

1.1 Bases pour la console

1.1.1 L'invite de l'interpréteur de commandes

Upon starting the system, you are presented with the character based login screen if you did not install any [GUI](#) environment such as [GNOME](#) or [KDE](#) desktop system. Suppose your hostname is `foo`, the login prompt looks as follows.

If you installed a [GUI](#) environment, then you can still get to the character based login prompt by Ctrl-Alt-F3, and you can return to the GUI environment via Ctrl-Alt-F2 (see Section [1.1.6](#) below for more).

```
foo login:
```

À l'invite d'identification, entrez votre nom d'utilisateur, par exemple `pingouin`, et pressez la touche Entrée, entrez ensuite votre mot de passe et pressez de nouveau la touche Entrée.

Note

Conformément à la tradition UNIX, l'identifiant de l'utilisateur et le mot de passe sur un système Debian sont sensibles à la casse. L'identifiant de l'utilisateur est habituellement choisi uniquement en minuscules. Le premier compte d'utilisateur est normalement créé lors de l'installation. Des comptes d'utilisateurs supplémentaires peuvent être créés avec la commande `adduser(8)` par l'administrateur (`root`).

Le système démarre avec le message de bienvenue qui se trouve dans « `/etc/motd` » (Message du jour : « Message Of The Day ») et présente une invite de commande comme :

```
Debian GNU/Linux 11 foo tty1

foo login: penguin
Password:
Linux foo 5.10.0-6-amd64 #1 SMP Debian 5.10.28-1 (2021-04-09) 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: Thu May 13 08:47:13 JST 2021 on tty1
foo:~$
```

Vous êtes maintenant sous [l'interpréteur de commandes](#) (« shell »). Le shell interprète vos commandes.

1.1.2 The shell prompt under GUI

If you installed a [GUI](#) environment during the installation, you are presented with the graphical login screen upon starting your system. You type your username and your password to login to the non-privileged user account. Use `tab` to navigate between username and password, or use the primary click of the mouse.

You can gain the shell prompt under GUI environment by starting a `x-terminal-emulator` program such as `gnome-terminal`(1), `rxvt`(1) or `xterm`(1). Under the GNOME Desktop environment, press `SUPER`-key (Windows-key) and typing in "terminal" to the search prompt does the trick.

Sous d'autres environnements de bureau (comme `fluxbox`), il peut ne pas y avoir de point d'accès évident au menu. Si cela se produit, essayez simplement de faire un clic-droit sur le fond d'écran de l'environnement de bureau en espérant voir apparaître un menu.

1.1.3 Compte de l'administrateur (root)

Le compte de l'administrateur (root) est encore appelé [superutilisateur](#) ou utilisateur privilégié. Depuis ce compte, vous pouvez effectuer les opérations d'administration du système suivantes :

- lire, écrire et effacer n'importe quel fichier du système quelles que soient ses permissions ;
- Défini l'appartenance et les permissions de n'importe quel fichier du système
- définir le mot de passe de n'importe quel utilisateur non-privilégié du système ;
- vous connecter à n'importe quel compte sans mot de passe.

La puissance illimitée du compte de l'administrateur fait que vous devez être attentif et responsable lorsque vous l'utilisez.



AVERTISSEMENT

Ne donnez jamais le mot de passe de l'administrateur à d'autres personnes.

Note

Les permissions d'un fichier (y compris les fichiers de périphériques tels que les CD-ROM ou autres, qui ne sont que des fichiers parmi d'autres pour le système Debian) peuvent le rendre inutilisable ou inaccessible à des utilisateurs autres que l'administrateur. Bien que l'utilisation du compte de l'administrateur soit un moyen rapide de test dans une telle situation, sa résolution devra être effectuée en définissant correctement les droits de fichiers et les membres des groupes d'utilisateurs (consultez Section [1.2.3](#)).

1.1.4 Invite de l'interpréteur de commandes pour l'administrateur

Voici quelques méthodes de base pour obtenir l'invite de l'interpréteur de commande de l'administrateur en utilisant le mot de passe de « root » .

- entrez `root` à l'invite de connexion en mode caractère ;
- entrez « `su -l` » depuis une invite quelconque de l'interpréteur de commandes ;
 - cela ne préserve pas l'environnement de l'utilisateur actuel ;
- entrez « `su` » depuis une invite quelconque de l'interpréteur de commandes ;
 - cela préserve une partie de l'environnement de l'utilisateur actuel.

1.1.5 Outils graphiques d'administration du système

When your desktop menu does not start GUI system administration tools automatically with the appropriate privilege, you can start them from the root shell prompt of the terminal emulator, such as `gnome-terminal(1)`, `rxvt(1)`, or `xterm(1)`. See Section 1.1.4 and Section 7.8.



AVERTISSEMENT

Never start the GUI display/session manager under the root account by typing in `root` to the prompt of the display manager such as `gdm3(1)`.

Ne jamais faire tourner de programme distant non sûrs avec une interface graphique sous X Window lorsque des informations critiques sont affichées parce que votre écran X peut être espionné.

1.1.6 Consoles virtuelles

Il y a, dans le système Debian par défaut, six consoles commutables en mode caractères de type [VT100](#) disponibles pour lancer directement l'interpréteur de commandes sur la machine Linux. À moins que vous ne disposiez d'un environnement graphique, vous pouvez basculer entre les consoles virtuelles en pressant la touche `Alt` de gauche et, simultanément, l'une des touches `F1` à `F6`. Chaque console en mode caractères permet de se connecter de manière indépendante à un compte et offre un environnement multi-utilisateurs. Cet environnement multi-utilisateurs est une fonctionnalité intéressante d'UNIX, très vite, on ne peut plus s'en passer.

If you are in the GUI environment, you gain access to the character console 3 by pressing `Ctrl-Alt-F3` key, i.e., the `left-Ctrl-key`, the `left-Alt-key`, and the `F3-key` are pressed together. You can get back to the GUI environment, normally running on the virtual console 2, by pressing `Alt-F2`.

You can alternatively change to another virtual console, e.g. to the console 3, from the commandline.

```
# chvt 3
```

1.1.7 Comment quitter l'invite de l'interpréteur de commandes

Pour arrêter l'activité de l'interpréteur de commandes, entrez `Ctrl-D`, c'est-à-dire la touche `Ctrl` de gauche et la touche `d` pressées simultanément, à l'invite de l'interpréteur. Si vous êtes sur une console en mode caractères, cela vous fera retourner alors à l'invite d'identification. Même si on se réfère à ces caractères de commande avec « control D » en majuscule, il n'est pas nécessaire de presser la touche majuscule. Le raccourci `^D`, est aussi utilisé pour `Ctrl-D`. Vous pouvez aussi entrer « exit ».

Si vous êtes sur un émulateur de `x-terminal(1)`, vous pouvez fermer la fenêtre de l'émulateur de `terminal-x` de la même manière.

1.1.8 Comment arrêter le système

Comme tout autre système d'exploitation moderne où les opérations sur les fichiers mettent en œuvre un [cache de données](#) en mémoire afin d'améliorer les performances, le système Debian demande que l'on exécute une procédure d'arrêt adaptée avant que le système ne puisse être mis hors tension. Cela permet de maintenir l'intégrité des fichiers, en forçant l'écriture sur disque de toutes les informations conservées en mémoire. Si un logiciel de contrôle de l'alimentation est disponible, la procédure d'arrêt coupera automatiquement l'alimentation du système. (Sinon, vous devrez presser le bouton d'alimentation pendant quelques secondes une fois la procédure d'arrêt terminée).

Dans le mode normal multi-utilisateurs, vous pouvez arrêter le système depuis la ligne de commandes :

```
# shutdown -h now
```

Dans le mode normal mono-utilisateur, vous pouvez arrêter le système depuis la ligne de commandes :

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

Consultez Section [6.3.8](#).

1.1.9 Récupérer une console propre

Lorsque l'écran est corrompu après que vous ayez fait des choses amusantes comme « `cat un-fichier-binaire` » entrez « `reset` » à l'invite de commande. Il est possible que vous ne voyiez pas ce que vous entrez. Vous pouvez aussi entrer « `clear` » pour nettoyer l'écran.

1.1.10 Suggestions de paquets supplémentaires pour le débutant

Bien que même une installation minimale du système Debian sans aucune tâche d'environnement de bureau fournisse les fonctionnalités UNIX de base, c'est une bonne idée pour les débutants d'installer pour commencer quelques paquets de ligne de commandes ou de terminal en mode caractères basé sur « curses » comme `mc` et `vim` avec `apt-get(8)` en utilisant les commandes suivantes :

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

Si vous avez déjà installé ces paquets, aucun nouveau paquet ne sera installé.

paquet	popcon	taille	description
mc	V:55, I:234	1492	gestionnaire de fichiers plein écran en mode texte
sudo	V:597, I:809	4589	programme donnant aux utilisateurs des privilèges d'administration limités
vim	V:102, I:404	3286	éditeur de texte UNIX Vi amélioré (Vi IMproved), éditeur de texte pour programmeurs (version standard)
vim-tiny	V:57, I:969	1574	éditeur de texte UNIX Vi amélioré (Vi IMproved), éditeur de texte pour programmeurs (version compacte)
emacs-nox	V:4, I:19	18364	GNU Emacs, éditeur de texte extensible basé sur Lisp
w3m	V:25, I:252	2367	navigateurs WWW en mode texte
gpm	V:11, I:16	540	couper-coller à la mode UNIX sur une console texte (démon)

Table 1.1 – Liste de paquets de programmes intéressants en mode texte

Ce peut être une bonne idée de lire quelques documentations.

Vous pouvez installer quelques-uns de ces paquets en passant les commandes suivantes :

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

paquet	popcon	taille	description
doc-debian	1:850	166	projet de documentation Debian, (FAQ Debian) et autres documents
debian-policy	1:29	4306	Charte Debian et documents associés
developers-reference	1:6	1916	Guides et informations pour les responsables Debian
debmake-doc	1:0	10989	Guide for Debian Maintainers
debian-history	1:1	4285	Histoire du projet Debian
debian-faq	1:846	817	FAQ Debian

Table 1.2 – Liste de paquets de documentation

1.1.11 Compte pour un utilisateur supplémentaire

Si vous ne souhaitez pas utiliser votre compte d'utilisateur principal pour les activités de formation qui suivent, vous pouvez créer un compte de formation, par exemple `poisson` en passant la commande qui suit :

```
# adduser fish
```

Répondez à toutes les questions.

Cela va créer un nouveau compte appelé `poisson`. Après vos exercices, vous pourrez supprimer ce compte d'utilisateur et son répertoire personnel par :

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

1.1.12 Configuration de sudo

Pour une station de travail typique avec un seul utilisateur tel que le système de bureau Debian sur un PC de bureau, il est habituel de mettre en œuvre une configuration simple de `sudo(8)` comme suit afin que l'utilisateur non privilégié, par exemple `pingouin`, puisse obtenir les privilèges d'administration avec simplement son mot de passe personnel et non avec le mot de passe de l'administrateur :

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

Il est aussi habituel, en remplacement, de faire comme suit afin que l'utilisateur non privilégié, par exemple `pingouin`, puisse obtenir les privilèges d'administration sans aucun mot de passe personnel.

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

Cette astuce ne doit être utilisée qu'avec une station de travail mono-utilisateur que vous administrez et dont vous êtes le seul utilisateur.



AVERTISSEMENT

Ne configurez pas de cette manière les utilisateurs normaux d'une station de travail multi-utilisateurs parce que cela serait très grave pour la sécurité du système.



Attention

Le mot de passe et le compte du `pingouin` de l'exemple précédent doivent bénéficier de la même protection que le mot de passe et le compte de l'administrateur.

Le privilège d'administration, dans ce contexte, appartient à quelqu'un autorisé à effectuer les tâches d'administration du système sur la station de travail. Ne jamais donner un tel privilège à un responsable du département administratif de votre entreprise ni à votre patron, à moins qu'ils n'y soient autorisés et en soient capables.

Note

Pour donner un accès privilégié à certains périphériques et certains fichiers, vous devriez envisager l'utilisation d'un **groupe** donnant un accès limité plutôt que d'utiliser le privilège de `root` par l'intermédiaire de `sudo(8)`. Avec une configuration plus approfondie et prudente, `sudo(8)` peut permettre à d'autres utilisateurs d'obtenir des privilèges limités d'administration sur un système partagé sans partager le mot de passe de l'administrateur. Cela peut améliorer la confiance sur les machines ayant plusieurs administrateurs de manière à ce que vous puissiez dire qui a fait quoi. D'un autre côté, vous ne devriez pas confier de tels privilèges à quelqu'un d'autre.

1.1.13 À vous de jouer

Vous êtes maintenant prêt à jouer avec le système Debian, sans risque aussi longtemps que vous utiliserez un compte d'utilisateur sans privilège.

Cela, parce que le système Debian, même après l'installation initiale, est configuré avec des permissions de fichiers adaptées qui évitent aux utilisateurs non privilégiés d'endommager le système. Bien entendu, il peut y avoir certaines failles qui peuvent être exploitées mais ceux qui s'inquiètent de ces problèmes ne devraient pas lire cette section mais plutôt le [Manuel de sécurisation](#) (« Securing Debian Manual »).

Nous allons apprendre le système Debian en tant que système [semblable à UNIX](#) (« UNIX-like ») avec :

- Section [1.2](#) (concept de base) ;
- Section [1.3](#) (méthode de survie) ;
- Section [1.4](#) (méthode de base) ;
- Section [1.5](#) (mécanisme de l'interpréteur de commandes) ;
- Section [1.6](#) (méthode de traitement des données textuelles).

1.2 Système de fichiers de type UNIX

Avec GNU/Linux et d'autres systèmes d'exploitation [semblables à UNIX](#), les [fichiers](#) sont organisés en [répertoires](#). Tous les fichiers et les répertoires sont disposés sous forme d'une grosse arborescence ancrée sur « / ». On l'appelle un arbre parce que si vous dessinez le système de fichiers, il ressemble à un arbre qui se trouverait disposé la tête en bas.

Ces fichiers et répertoires peuvent être répartis sur plusieurs périphériques. La commande `mount(8)` sert à attacher les systèmes de fichiers se trouvant sur certains périphériques à la grosse arborescence des fichiers. À l'opposé, la commande `umount(8)` les détachera de nouveau. Avec les noyaux Linux récents, `mount(8)` avec certaines options peut lier une partie d'une arborescence de fichiers à un autre emplacement ou peut monter un système de fichiers de manière partagée, privée, esclave ou « non-liable ». Vous trouverez les options de montage prises en compte par chaque système de fichiers dans « `/usr/share/doc/linux-doc-*/Documentation` ».

Les **répertoires** d'un système UNIX sont appelés **dossiers** sur d'autres systèmes. Vous remarquez aussi qu'il n'y a, sur aucun système UNIX, de concept de **lecteur** tel que « A: ». Il y a un système de fichiers qui comprend tout. C'est un gros avantage comparé à Windows.

1.2.1 Bases concernant les fichiers UNIX

Voici les bases des fichiers UNIX :

- les noms de fichiers sont **sensibles à la casse**. Ce qui veut dire que « `MONFICHIER` » et « `MonFichier` » sont des fichiers différents ;
 - on se réfère au **répertoire racine** (« root directory »), qui est la racine du système de fichiers, simplement par « / ». Ne pas le confondre avec le répertoire personnel de l'utilisateur `root` : « `/root` » ;
 - un nom de répertoire peut être constitué de n'importe quelle lettre ou symbole **sauf** « / ». Le répertoire racine est une exception, son nom est « / » (prononcé « slash » ou « le répertoire racine »), il ne peut pas être renommé ;
 - chaque fichier ou répertoire est désigné par un **nom de fichier entièrement qualifié**, **nom de fichier absolu** ou **chemin**, indiquant la séquence de répertoires que l'on doit traverser pour l'atteindre. Les trois expressions sont synonymes ;
-

- tous les **noms de fichiers entièrement qualifiés** commencent par le répertoire « / » et il y a un « / » entre chaque répertoire ou fichier dans le nom du fichier. Le premier « / » est le répertoire de plus haut niveau, et les autres « / » séparent les sous-répertoires successifs jusqu'à ce que l'on atteigne la dernière entrée qui est le nom fichier proprement dit. Les mots utilisés ici peuvent être source de confusion. Prenez comme exemple le **nom pleinement qualifié** suivant : « /usr/share/keytables/us. ». Cependant, les gens utiliseront souvent son nom de base « us.map.gz » seul comme nom de fichier ;
- le répertoire racine comporte de nombreuses branches, telles que « /etc/ » et « /usr/ ». Ces sous-répertoires se décomposent eux-mêmes en d'autres sous-répertoires comme « /etc/init.d/ » et « /usr/local/ ». L'ensemble de la chose, vu globalement, s'appelle l'**arborescence des répertoires**. Vous pouvez imaginer un nom de fichier absolu comme une route partant de la base de l'arbre (« / ») jusqu'à l'extrémité de certaines branches (le fichier). Vous entendrez aussi certains parler de l'arborescence des répertoires comme d'un arbre **généalogique** englobant tous les descendants directs d'un seul personnage appelé le répertoire racine (« / ») : les sous-répertoires ont alors des **parents** et un chemin montre l'ascendance complète d'un fichier. Il y a aussi des chemins relatifs qui commencent quelque part ailleurs qu'au niveau du répertoire racine. Il faut vous souvenir que le répertoire « ../ » indique le répertoire parent. Cette terminologie s'applique de la même manière aux autres structures ressemblant aux répertoires comme les structures de données hiérarchiques ;
- il n'y a pas de chemin de répertoire spécial correspondant à un périphérique physique tel que votre disque dur. C'est différent de [RT-11](#), [CP/M](#), [OpenVMS](#), [MS-DOS](#), [AmigaOS](#) et [Microsoft Windows](#), où le chemin comporte le nom du périphérique comme « C:\ ». (Il existe cependant des entrées de répertoire qui font référence aux périphériques physiques en tant qu'élément du système de fichiers normal. Consultez Section [1.2.2](#)).

Note

Bien que vous **puissiez** utiliser la plupart des lettres ou symboles dans un nom de fichier, c'est en pratique un mauvaise idée de le faire. Il est préférable d'éviter tous les caractères qui ont une signification particulière sur la ligne de commandes, comme les espaces, tabulations, sauts de ligne, et autres caractères spéciaux : { } () [] ' ` " \ / > < | ; ! # & ^ * % @ \$. Si vous voulez séparer des mots dans un nom, de bons choix sont le point, le tiret et le tiret souligné. Vous pouvez aussi mettre une majuscule en tête de chaque mot « CommeCeci ». Les utilisateurs Linux expérimentés ont tendance à aussi à éviter les espaces dans les noms de fichiers.

Note

Le mot « root » signifie soit « utilisateur root », soit « répertoire root ». Le contexte de son utilisation devrait permettre de les distinguer.

Note

Le mot **chemin** (« path ») n'est pas utilisé que pour les **noms de fichiers entièrement qualifiés** comme ci-dessus mais aussi pour le **chemin de recherche des commandes**. La signification voulue est habituellement claire selon le contexte.

Les meilleures façons de faire en ce qui concerne la hiérarchie des fichiers sont détaillées dans la « norme de hiérarchie du système de fichiers » (« Filesystem Hierarchy Standard ») (« /usr/share/doc/debian-policy/fhs/fhs-2.3.txt.gz » et [hier\(7\)](#)). Vous devriez, pour commencer, mémoriser les éléments suivants :

répertoire	utilisation du répertoire
/	répertoire racine
/etc/	fichiers de configuration valables pour l'ensemble du système
/var/log/	fichiers journaux du système
/home/	tous les répertoires personnels des utilisateurs non privilégiés

Table 1.3 – Utilisation des répertoires-clés

1.2.2 Fonctionnement interne du système de fichiers

En suivant la **tradition UNIX**, le système Debian GNU/Linux fournit un **système de fichiers** où les données physiques se trouvent sur des disques durs et d'autres périphériques de stockage et où les interaction avec les périphériques physiques tels que les écrans de console et les consoles distantes connectées en séries sont représentées de manière unifiée dans « `/dev/` ».

Chaque fichier, répertoire, tube nommé (une manière pour les programmes de partager des données) ou périphérique physique sur un système Debian GNU/Linux possède une structure de données appelée **inœud** (« inode ») qui décrit les attributs qui lui sont associés comme l'utilisateur qui le possède (propriétaire), le groupe auquel il appartient, l'heure de dernier accès, etc. L'idée de pouvoir presque tout représenter dans le système de fichiers était une innovation d'UNIX, et les noyaux modernes de Linux ont encore développé plus loin cette idée. À l'heure actuelle, même les informations concernant les processus qui tournent sur le système se trouvent sur le système de fichiers.

Cette représentation abstraite et unifiée des entités physiques et des processus internes est très puissante puisque cela nous permet d'utiliser la même commande pour le même type d'opération sur des périphériques complètement différents. Il est même possible de changer la manière dont fonctionne le noyau en écrivant des données dans des fichiers spéciaux liés aux processus en cours d'exécution.

ASTUCE

Si vous avez besoin de connaître la correspondance entre une arborescence de fichiers et un périphérique physique, lancez la commande `mount(8)` sans paramètre.

1.2.3 Permissions du système de fichiers

Les **permissions du système de fichiers** d'un système **basé sur UNIX** sont définies pour trois catégories d'utilisateurs :

- l'**utilisateur** qui possède le fichier (**u**) ;
- les autres utilisateurs du **groupe** à qui appartient le fichier (**g**) ;
- tous les **autres** utilisateurs (**o**) dont on parle aussi en tant que « monde entier » ou « tout le monde ».

Pour les fichiers, chaque permission correspondante permet les actions suivantes :

- la permission en **lecture** (**r**) permet à son propriétaire de voir le contenu du fichier ;
- la permission en **écriture** (**w**) permet à son propriétaire de modifier le fichier ;
- la permission d'**exécution** (**x**) permet à son propriétaire de lancer le fichier comme une commande.

Pour les répertoires, chaque permission correspondante permet les actions suivantes :

- la permission en **lecture** (**r**) permet à son propriétaire d'afficher le contenu du répertoire ;
- la permission en **écriture** (**w**) permet à son propriétaire d'ajouter ou supprimer des fichiers de ce répertoires ;
- la permission d'**exécution** (**x**) permet à son propriétaire d'accéder aux fichiers du répertoire.

Ici, la permission en **exécution** sur un répertoire ne signifie pas uniquement l'autorisation de lire des fichiers dans ce répertoire mais aussi l'autorisation de voir leurs attributs, tels que leur taille et l'heure de modification.

`ls(1)` est utilisé pour afficher les informations de permissions (et davantage) des fichiers et répertoires. Lorsque cette commande est passée avec l'option « `-l` », elle affiche les informations suivantes dans l'ordre donné :

- **type de fichier** (premier caractère) ;
 - **autorisation** d'accès au fichier (neuf caractères, constitués de trois caractères pour l'utilisateur, le groupe et « les autres », dans cet ordre) ;
 - **nombre de liens physiques** vers le fichier ;
 - nom de l'**utilisateur** propriétaire du fichier ;
 - nom du **groupe** à qui appartient le fichier ;
 - **taille** du fichier en caractères (octets) ;
 - **date et heure** du fichier (mtime) ;
 - **nom** du fichier.
-

caractère	signification
-	fichier normal
d	répertoire
l	lien symbolique
c	nœud de périphérique en mode caractère
b	nœud de périphérique en mode bloc
p	tube nommé
s	socket

Table 1.4 – Liste des premiers caractères de la sortie de « `ls -l` » :

`chown(1)` est utilisé depuis le compte de l’administrateur pour modifier le propriétaire d’un fichier. `chgrp(1)` est utilisé depuis le compte du propriétaire du fichier ou de l’administrateur pour changer le groupe du fichier. `chmod(1)` est utilisé depuis le compte du propriétaire du fichier ou de l’administrateur pour changer les droits d’accès à un fichier ou un répertoire. La syntaxe de base pour manipuler le fichier `toto` est la suivante :

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

Vous pouvez, par exemple, faire qu’une arborescence de répertoires soit la propriété de l’utilisateur `toto` et partagée par le groupe `titi` en faisant ce qui suit :

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

Il existe trois bits qui donnent des permissions particulières :

- le bit **set user ID** (**s** ou **S** situé à la place du **x**) de l’utilisateur ;
- le bit **set group ID** (**s** ou **S** situé à la place du **x**) du groupe ;
- le **bit collant** « sticky bit » (**t** ou **T** situé à la place du **x**) des « autres ».

Ici la sortie de « `ls -l` » avec ces bits est **en majuscules** si les bits d’exécution cachés par ces sorties ne sont **pas positionnés**.

Définir **set user ID** sur un fichier exécutable permet à un utilisateur d’exécuter les fichiers avec l’identifiant du propriétaire du fichier (par exemple **root**). De la même manière, définir **set group ID** sur un fichier exécutable permet d’exécuter le fichier avec l’identifiant de groupe du fichier (par exemple **root**). Parce que ces positionnements de bits peuvent créer des risques de sécurité, il ne faut les activer qu’avec des précautions extrêmes.

Définir **set group ID** sur un répertoire permet la création de fichiers [à la BSD](#) où tous les fichiers créés dans un répertoire appartiennent au **groupe** du répertoire.

Positionner le **sticky bit** d’un répertoire empêche un fichier de ce répertoire d’être supprimé par un utilisateur qui n’est pas le propriétaire du fichier. Pour sécuriser le contenu d’un fichier dans des répertoires pouvant être écrits par tout le monde tels que « `/tmp` » ou dans des répertoires pouvant être écrits par le groupe, il ne faut pas uniquement supprimer la permission du fichier en **écriture** mais aussi positionner le **sticky bit** sur le répertoire. Sinon, le fichier pourra être supprimé et un nouveau fichier créé avec le même nom par un utilisateur quelconque ayant accès en écriture au répertoire.

Voici quelques exemples intéressants de permissions de fichiers.

```
$ 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
```

Il existe aussi un mode numérique pour décrire les permissions des fichiers avec `chmod(1)`. Ce mode numérique utilise des nombres en base 8 (radix=8) codés sur 3 ou 4 chiffres.

chiffre	signification
1er chiffre optionnel	somme de set user ID (=4), set group ID (=2), et sticky bit (=1)
2ème chiffre	somme des permissions de lecture (=4), écriture (=2), et exécution (=1) pour l' utilisateur
3ème chiffre	identique pour groupe
4ème chiffre	identique pour autres

Table 1.5 – Mode numérique des permissions de fichiers dans les commandes `chmod(1)`

Cela peut sembler compliqué mais c'est en fait assez simple. Si vous regardez les quelques premières colonnes (2-10) de la sortie de la commande « `ls -l` » et que vous lisez en représentation binaire (base 2) les permissions des fichiers (le « - » représentant « 0 » et « `rwX` » représentant « 1 »), les trois derniers chiffres de la valeur numérique du mode devraient vous donner la représentation des permissions du fichier en octal (base 8).

Essayez, par exemple, ce qui suit :

```
$ 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
```

ASTUCE

Si vous avez besoin d'accéder aux informations affichées par « `ls -l` » depuis un script de l'interpréteur de commandes, vous devrez utiliser des commandes pertinentes telles que `test(1)`, `stat(1)` et `readlink(1)`. Vous pouvez aussi utiliser les commandes internes du shell, telles que « `[` » ou « `test` ».

1.2.4 Contrôle des permissions pour les fichiers nouvellement créés : `umask`

Les permissions appliquées à un fichier ou à un répertoire venant d'être créé sont restreintes par la commande interne du shell `umask`. Consultez `dash(1)`, `bash(1)` et `builtins(7)`.

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

umask	permissions des fichiers créés	permissions des répertoires créés	utilisation
0022	-rw-r--r--	-rwxr-xr-x	ne peut être écrit que par l'utilisateur
0002	-rw-rw-r--	-rwxrwxr-x	peut être écrit par le groupe

Table 1.6 – Exemples de valeurs de `umask`

Le système Debian utilise par défaut un schéma de groupe privé par utilisateur (UPG). Un UPG est créé chaque fois qu'un utilisateur est ajouté au système. Un UPG a le même nom que l'utilisateur pour lequel il a été créé et cet utilisateur est le seul membre de l'UPG. Un principe d'UPG rend sûr le positionnement de `umask` à `0002` car chaque utilisateur a son propre groupe privé. (Sur certaines variantes d'UNIX, il est assez courant de faire appartenir tous les utilisateurs au même groupe **users** et, dans ce cas, c'est une bonne idée de définir `umask` à `0022` pour des raisons de sécurité).

ASTUCE

Activez UPG en plaçant « `umask 002` » dans le fichier `~/ .bashrc`.

1.2.5 Permissions pour les groupes d'utilisateurs (group)

Afin que des permissions attribuées à un groupe soient appliquées à un utilisateur particulier, il faut que cet utilisateur soit déclaré membre du groupe à l'aide de « `sudo vigr` » pour `/etc/group` ou « `sudo vigr -s` » pour `/etc/gshadow`. La nouvelle configuration du groupe n'est effective qu'après une [re]connexion de l'utilisateur (ou l'exécution de « `exec newgrp` »).

Note

Vous pouvez aussi ajouter dynamiquement des utilisateurs aux groupes durant le processus d'authentification en ajoutant la ligne « `auth optional pam_group.so` » au fichier « `/etc/pam.d/common-auth` » et en définissant « `/etc/security/group.conf` » (consultez Chapitre 4).

Sous le système Debian, les périphériques ne sont qu'un autre type de fichier. Si vous avez des problèmes pour accéder à des périphériques tel qu'un CD-ROM ou une clé USB depuis le compte d'un utilisateur, il faudra rendre cet utilisateur membre du groupe concerné.

Certains groupes importants fournis par le système permettent à leurs membres l'accès à des fichiers et des périphériques particuliers sans avoir les privilèges de l'administrateur.

groupe	description des fichiers et périphériques accessibles
<code>dialout</code>	accès complet et direct aux ports série (« <code>/dev/ttyS[0-3]</code> »)
<code>dip</code>	accès limité aux ports série pour une connexion « Dialup IP (réseau commuté) vers des pairs de confiance
<code>cdrom</code>	lecteurs et graveurs de CD-ROM, DVD+/-RW.
<code>audio</code>	périphérique audio
<code>video</code>	périphérique vidéo
<code>scanner</code>	dispositifs de numérisation (scanners)
<code>adm</code>	journaux de surveillance du système
<code>staff</code>	quelques répertoires où effectuer du travail d'administration de début : « <code>/usr/local</code> », « <code>/home</code> ».

Table 1.7 – Liste des groupes importants fournis par le système pour l'accès aux fichiers

ASTUCE

Vous devez être membre du groupe `dialout` pour pouvoir reconfigurer un modem, numéroter vers n'importe où, etc. Mais si l'administrateur crée dans « `/etc/ppp/peers/` » des fichiers de configuration pour des pairs de confiance, vous ne devrez appartenir qu'au groupe `dip` pour créer une connexion **commutée** (« **Dialup IP** ») vers ces pairs de confiance avec commandes `pppd(8)`, `pon(1)` et `poff(1)`.

Certains groupes faisant partie du système, permettent à leurs membres d'exécuter des commandes particulières sans les privilèges de l'administrateur (`root`).

groupe	commandes accessibles
<code>sudo</code>	exécuter <code>sudo</code> sans son mot de passe.
<code>lpadmin</code>	exécuter des commandes pour ajouter, modifier et supprimer des imprimantes de la base de données des imprimantes

Table 1.8 – Liste des groupes importants fournis par le système pour l'exécution de commandes particulières

Pour une liste complète des groupes et des utilisateurs fournis par le système, veuillez consulter une version récente du document « Utilisateurs et groupes » (« `Users and Groups` » se trouvant dans `/usr/share/doc/base-passwd/users-and-groups.htm` qui est fourni par le paquet `base-passwd`.

Consultez `passwd(5)`, `group(5)`, `shadow(5)`, `newgrp(1)`, `vipw(8)`, `vigr(8)` et `pam_group(8)` pour les commandes de gestion des utilisateurs et des groupes du système.

1.2.6 Horodatage

Il existe trois types d'horodatage pour un fichier GNU/Linux.

type	signification (définition historique d'Unix)
mtime	date de modification du fichier (<code>ls -l</code>)
ctime	date de changement d'état du fichier (<code>ls -lc</code>)
atime	date de dernier accès au fichier (<code>ls -lu</code>)

Table 1.9 – Liste des types d'horodatage

Note

ctime n'est pas la date de création du fichier.

Note

La valeur d'**atime** sur GNU/Linux peut être en fait différente de celle donnée par la définition Unix historique.

- Écraser un fichier va modifier tous les attributs **mtime**, **ctime** et **atime** du fichier.
 - Modifier le propriétaire ou les droits d'un fichier va changer les attributs **ctime** et **atime** du fichier.
 - La lecture d'un fichier va modifier l'attribut **atime** du fichier sur le système Unix d'historique.
 - La lecture d'un fichier va modifier l'attribut **atime** du fichier que un système GNU/Linux si son système de fichier est monté avec "strictatime".
 - Lire un fichier pour la première fois ou après un jour modifie son attribut **atime** sur un système GNU/Linux avec un système de fichiers monté en « **relatime** » (par défaut depuis Linux 2.6.30).
 - Lire un fichier ne modifie pas son attribut **atime** sous GNU/Linux si le système de fichiers est monté avec « **noatime** ».
-

Note

The "noatime" and "relatime" mount options are introduced to improve the filesystem read performance under the normal use cases. Simple file read operation under the "strictatime" option accompanies the time-consuming write operation to update the **atime** attribute. But the **atime** attribute is rarely used except for the mbox(5) file. See mount(8).

Utilisez la commande `touch(1)` pour modifier l'horodatage des fichiers existants.

For timestamps, the `ls` command outputs localized strings under non-English locale ("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
```

ASTUCE

Consultez Section 9.3.4 pour personnaliser la sortie de « `ls -l` ».

1.2.7 Liens

Il existe deux méthodes pour associer le fichier « `toto` » avec un nom de fichier différent « `titi` » :

- [Lien physique](#)
 - Nom dupliqué d'un fichier existant
 - « `ln toto titi` »
- [Lien symbolique ou <symlink>](#)
 - Fichier spécial pointant vers un autre fichier par son nom
 - « `ln -s toto titi` »

Consultez l'exemple suivant pour des modifications du nombre de liens et les subtiles différences dans le résultat de la commande `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
```

The hardlink can be made within the same filesystem and shares the same inode number which the `-i` option with `ls(1)` reveals.

Le lien symbolique a les permissions d'accès nominales au fichier de « `rw-rw-rwx` » comme il apparaît dans l'exemple ci-dessus, alors que les permissions d'accès effectives sont celles du fichier vers lequel il pointe.



Attention

En règle générale — à moins d'avoir une très bonne raison pour cela — il faudrait s'abstenir de créer des liens physiques ou des liens symboliques compliqués. Cela peut provoquer des cauchemars lorsque la combinaison logique des liens symboliques crée une boucle dans le système de fichiers.

Note

Il est généralement préférable d'utiliser des liens symboliques plutôt que des liens physiques à moins que vous n'ayez une bonne raison d'utiliser un lien physique.

Le répertoire « `.` » est lié au répertoire dans lequel il apparaît, le nombre de liens de n'importe quel nouveau répertoire commence donc à 2. Le répertoire « `..` » est lié au répertoire parent, le nombre de liens du répertoire augmente donc lors de l'ajout de nouveaux sous-répertoires.

Si vous venez de passer à Linux depuis Windows, la bonne conception d'un système de fichiers UNIX comparé à l'équivalent le plus proche que sont les « raccourcis Windows » deviendra vite claire. Parce qu'il est implémenté dans le système de fichiers, les applications ne voient pas de différence entre un fichier lié et son original. Dans le cas de liens physiques, il n'y a vraiment aucune différence.

1.2.8 Tubes nommés (FIFO)

Un **tube nommé** est un fichier qui se comporte comme un tuyau. Vous mettez quelque chose dans le tuyau et il ressort à l'autre bout. C'est donc appelé une FIFO, ou « premier entré-premier sorti » (First-In-First-Out) : la première chose que vous mettez dans le tuyau est la première chose qui ressortira à l'autre bout.

Si vous écrivez vers un tube nommé, le processus qui écrit dans le tube ne se termine pas avant que l'information ne soit lue depuis le tube. Si vous effectuez une lecture depuis un tube nommé, le processus de lecture attendra jusqu'à ce qu'il n'y ait plus rien à lire avant de se terminer. La taille d'un tube est toujours nulle — il ne stocke pas de données, ce n'est qu'un lien entre deux processus, comparable à la fonction fournie par l'opérateur « | » dans l'interpréteur de commandes. Cependant, comme ce tube a un nom, il n'est pas nécessaire que les deux processus se trouvent sur la même ligne de commandes ni même qu'ils soient lancés par le même utilisateur. Les tubes sont une innovation déterminante d'UNIX.

Essayez, par exemple, ce qui suit :

```
$ 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 Sockets

Les sockets sont utilisées de manière intensive dans les communications par Internet, les bases de données et le système d'exploitation lui-même. Elles sont semblables aux tubes nommés (FIFO) et permettent aux processus d'échanger des informations même s'ils tournent sur des ordinateurs différents. Pour la socket, ces processus n'ont pas besoin de tourner en même temps ni de tourner en tant que fils du même processus père. C'est le point d'aboutissement du [processus d'intercommunication \(IPC\)](#). (« interprocess communication »). L'échange d'informations peut avoir lieu entre deux machines au travers du réseau. Les deux plus courantes sont la [socket Internet](#) (« Internet socket ») et la [socket du domaine UNIX](#) (« UNIX domain socket »).

ASTUCE

« netstat -an » fournit un aperçu très utile des sockets qui sont ouvertes sur un système donné.

1.2.10 Fichiers de périphériques

[Fichiers de périphériques](#) fait référence aux périphériques virtuels ou physiques de votre système, tels que vos disques durs, carte vidéo, écran ou clavier. Un exemple de périphérique virtuel est la console, représentée par « /dev/console »

Il y a 2 types de fichiers de périphériques :

- **périphérique en mode caractère** (« character device ») :
 - permettent l'accès à un caractère à la fois
 - 1 caractère = 1 octet
 - par exemple, les claviers, les ports série, ...
- **Périphériques en mode bloc** (« block device ») :
 - accèdent aux données par éléments plus importants appelés blocs
 - 1 bloc > 1 octet
 - par exemple, les disques durs, ...

Vous pouvez lire et écrire sur les fichiers de périphériques, cependant, le fichier peut fort bien contenir des données binaires qui peuvent être du charabia incompréhensible pour les êtres humains. Écrire des données directement dans ces fichiers est parfois utile pour déboguer des connexions matérielles. Vous pouvez, par exemple, vider un fichier texte vers le périphérique d'impression « `/dev/lp0` » ou envoyer des commandes de modem vers le port série approprié « `/dev/ttyS0` ». Mais, à moins que ce ne soit fait avec précautions, cela peut provoquer un désastre majeur. Soyez donc prudent.

Note

Pour l'accès normal à une imprimante, utilisez `lp(1)`.

On affiche le numéro de nœud du périphérique en utilisant la commande `ls(1)` comme suit :

```
$ 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` » a le numéro majeur de périphérique 8 et le numéro mineur de périphérique 0. Il est accessible en lecture et écriture aux utilisateurs qui appartiennent au groupe `disk`.
- « `/dev/sr0` » a le numéro majeur de périphérique 11 et le numéro mineur de périphérique 0. Il est accessible en lecture et écriture aux utilisateurs qui appartiennent au groupe `cdrom`.
- « `/dev/ttyS0` » a le numéro majeur de périphérique 4 et le numéro mineur de périphérique 64. Il est accessible en lecture et écriture aux utilisateurs qui appartiennent au groupe `dialout`.
- « `/dev/zero` » a le numéro majeur de périphérique 1 et le numéro mineur de périphérique 5. Il est accessible en lecture et écriture à tout le monde.

Sur les systèmes Linux modernes, le système de fichiers sous « `/dev/` » est automatiquement rempli par `udev(7)`.

1.2.11 Fichiers spéciaux de périphériques

Il y a aussi certains fichiers spéciaux de périphériques.

fichier de périphérique	action	description de la réponse
<code>/dev/null</code>	lire	retourne le « caractère fin de fichier (EOF) »
<code>/dev/null</code>	écrire	ne retourne rien (un puits de données sans fond)
<code>/dev/zero</code>	lire	retourne le « caractère <code>\0</code> (NULL) » (qui n'est pas identique au chiffre zéro ASCII)
<code>/dev/random</code>	lire	retourne des caractères aléatoires depuis un générateur de nombres aléatoires réel, en donnant une vraie entropie (lent)
<code>/dev/urandom</code>	lire	retourne des caractères aléatoires depuis un générateur de nombres pseudo-aléatoires, sécurisé par chiffrement
<code>/dev/full</code>	écrire	retourne une erreur disque plein (ENOSPC)

Table 1.10 – Liste des fichiers spéciaux de périphériques

Ils sont fréquemment utilisés en conjonction avec une redirection du shell (consultez Section 1.5.8).

1.2.12 `procfs` et `sysfs`

`procfs` et `sysfs` montés sur « `/proc` » et « `/sys` » sont des pseudo-systèmes de fichiers, ils présentent dans l'espace utilisateur des structures de données internes du noyau. En d'autres termes, ces entrées sont virtuelles, ce qui signifie qu'elles présentent une fenêtre sur le fonctionnement du système d'exploitation.

Le répertoire « `/proc` » contient (entre autres choses), pour chacun des processus tournant sur le système, un sous-répertoire dont le nom est l'identifiant du processus (PID). Les utilitaires du système qui accèdent aux informations des processus, tels que `ps(1)`, obtiennent leurs informations depuis cette structure de répertoires.

Les répertoires qui se trouvent sous « `/proc/sys/` » contiennent des interfaces permettant de modifier certains paramètres du système alors qu'il est en fonctionnement. (Vous pouvez faire la même chose au travers de la commande spécialisée `sysctl(8)` ou de son fichier de configuration « `/etc/sysctl.conf` »).

Certaines personnes paniquent lorsqu'elles remarquent un fichier particulier - « `/proc/kcore` » — qui est particulièrement énorme. C'est (plus ou moins) le contenu de la mémoire de votre ordinateur. Il est utilisé pour déboguer le noyau. C'est un fichier virtuel qui pointe vers la mémoire de l'ordinateur, ne vous inquiétez donc pas de sa taille.

Les répertoires sous « `/sys` » contiennent des structures de données exportées depuis le noyau, leurs attributs et les liens entre elles. Ils contiennent aussi des interfaces pour modifier certains paramètres du noyau pendant son fonctionnement.

Consultez « `proc.txt(.gz)` », « `sysfs.txt(.gz)` » et d'autres documents en rapport dans la documentation du noyau de Linux (« `/usr/share/doc/linux-doc-*/Documentation/filesystems/*` ») fournie par le paquet `linux-doc-*`.

1.2.13 tmpfs

Le `tmpfs` est un système de fichiers temporaire qui garde tous les fichiers en [mémoire virtuelle](#). Les données du `tmpfs` dans le [page cache](#) en mémoire peuvent être déplacées dans l'[espace d'échange](#) sur disque en cas de besoin.

Le répertoire « `/run` » est monté en `tmpfs` au tout début du processus de démarrage. Cela permet d'écrire dessus même quand le répertoire racine « `/` » est monté en lecture seule. C'est le nouvel emplacement pour le stockage de fichiers en état transitoire qui remplace plusieurs emplacements décrits dans la [norme de hiérarchie du système de fichiers](#) (« [Filesystem Hierarchy Standard](#) ») version 2.3 :

- « `/var/run` » → « `/run` »
- « `/var/lock` » → « `/run/lock` »
- « `/dev/shm` » → « `/run/shm` »

Consultez « `tmpfs.txt(.gz)` » dans la documentation du noyau de Linux (« `/usr/share/doc/linux-doc-*/Documentation` ») fournie par le paquet `linux-doc-*`.

1.3 Midnight Commander (MC)

[Midnight Commander \(MC\)](#) est un « couteau Suisse » GNU pour la console Linux et d'autres environnements de terminaux. Il permet au débutant d'acquérir une expérience de la console pilotée par des menus, ce qui est bien plus facile à apprendre que les commandes UNIX standard.

Il vous faudra peut-être installer le paquet Midnight Commander dont le nom est « `mc` » en effectuant ce qui suit :

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

Utilisez la commande `mc(1)` pour parcourir le système Debian. C'est la meilleure manière d'apprendre. Vous pouvez explorer certains emplacements intéressants en utilisant simplement les touches de curseur et la touche Entrée :

- « `/etc` » et ses sous-répertoires
- « `/var/log` » et ses sous-répertoires
- « `/usr/share/doc` » et ses sous-répertoires
- « `/sbin` » et « `/bin` »

1.3.1 Personnalisation de MC

Pour que MC modifie le répertoire de travail en quittant, et `cd` vers le répertoire, je vous suggère de modifier « `~/.bashrc` » afin d'inclure un script fourni par le paquet `mc` :

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

Vous trouverez une explication dans `mc(1)` (option « `-P` »). (Si vous ne comprenez pas exactement ce dont je parle ici, vous pourrez le faire plus tard).

1.3.2 Démarrer MC

MC peut être lancé par :

```
$ mc
```

MC prend en charge toutes les opérations sur les fichiers par l'intermédiaire de son menu, ce qui ne demande que peu d'effort de la part de l'utilisateur. Pressez simplement F1 pour obtenir l'écran d'aide. Vous pouvez jouer avec MC simplement en pressant les touches de curseur et les touches de fonctions.

Note

In some consoles such as `gnome-terminal(1)`, key strokes of function-keys may be stolen by the console program. You can disable these features in "Preferences" → "General" and "Shortcuts" menu for `gnome-terminal`.

Si vous rencontrez un problème de codage de caractères qui entraîne une corruption de l'affichage, ajouter « -a » à la ligne de commandes de MC peut aider à éviter les problèmes.

Si cela ne résout pas vos problèmes d'affichage avec MC, consultez Section [9.5.6](#).

1.3.3 Gestionnaire de fichiers de MC

Il y a par défaut, deux panneaux de répertoires affichant les listes de fichiers. Un autre mode utile est de définir la fenêtre de droite à « information » afin de voir les informations de privilèges d'accès aux répertoires. Vous trouverez ci-après quelques raccourcis clavier essentiels. Si le démon `gpm(8)` tourne, une souris est utilisable avec les consoles Linux en mode caractères. (Assurez-vous de presser la touche majuscules pour obtenir le comportement normal de couper-coller avec MC).

touche	affectation
F1	menu d'aide
F3	visualisateur interne de fichiers
F4	éditeur interne
F9	activer le menu déroulant
F10	quitter Midnight Commander
Tabulation	passer d'une fenêtre à l'autre
Ins ou Ctrl-T	marquer le fichier pour des opérations sur plusieurs fichiers telles que copier
Suppr	effacer le fichier (attention, configurez MC dans le mode d'effacement sécurisé)
Touches de curseur	autoexplicatif

Table 1.11 – Touches de raccourcis de MC

1.3.4 Astuces de la ligne de commandes dans MC

- la commande `cd` changera le répertoire affiché sur l'écran sélectionné ;
- **Ctrl-Entrée** ou **Alt-Entrée** copiera un nom de fichier sur la ligne de commandes. Utilisez cela avec les commandes `cp(1)` et `mv(1)` en association avec l'édition de la ligne de commandes ;
- **Alt-Tab** affichera les choix de l'interpréteur de commandes pour l'expansion du nom de fichier ;
- on peut indiquer le répertoire de départ pour les deux fenêtres en paramètre de MC. Par exemple « `mc /etc /root` » ;
- **Échap + touche n** → **Fn** (par exemple **Échap + 1** → **F1**, etc. ; **Échap + 0** → **F10**) ;
- Presser la touche **Échap** avant une touche a le même effet que presser simultanément **Alt** et la touche. Par exemple, entrez **Échap + c** pour **Alt -C**. **Échap** est appelée métatouche et parfois notée « **M-** ».

1.3.5 Éditeur interne de MC

L'éditeur interne possède une manière intéressante d'effectuer un copier-coller. Presser F3 marque le début de la sélection, F3 pressé une seconde fois marque la fin de la sélection et la met en surbrillance. Vous pouvez ensuite déplacer votre curseur. Si vous appuyez sur F6, la zone sélectionnée sera déplacée jusqu'à l'emplacement du curseur. Si vous pressez sur F5, la zone sélectionnée sera copiée et insérée à l'emplacement du curseur. F2 enregistrera le fichier. F10 vous permettra de quitter l'éditeur. La plupart des touches de déplacement du curseur fonctionnent de manière intuitive.

Cet éditeur peut être directement lancé avec un fichier en utilisant l'une des commandes suivantes :

```
$ mc -e filename_to_edit
```

```
$ mcedit filename_to_edit
```

Il ne s'agit pas d'un éditeur multi-fenêtres mais on peut faire usage de plusieurs consoles Linux pour obtenir le même effet. Pour copier d'une fenêtre sur l'autre, utilisez les touches **Alt-Fn** pour basculer d'une console virtuelle à l'autre et utilisez « File → Insert file » ou « File → Copy to file » pour déplacer une portion de fichier dans un autre fichier.

Cet éditeur interne peut être remplacé par n'importe quel autre éditeur externe de votre choix.

De nombreux programmes utilisent aussi les variables d'environnement « \$EDITOR » ou « \$VISUAL » afin de décider quel éditeur utiliser. Si, au départ, vous n'êtes à l'aise ni avec **vim**(1) ni avec **nano**(1), vous pouvez définir ces variable à « **mcedit** » en ajoutant les lignes suivantes au fichier « **~/.bashrc** » :

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

Je recommande de les définir à « **vim** » si possible.

Si vous n'êtes pas à l'aise avec **vim**(1), vous pouvez continuer à utiliser **mcedit**(1) pour la plupart des tâches de maintenance du système.

1.3.6 Visualisateur interne de MC

MC possède un visualisateur intelligent. C'est un très bon outil pour rechercher des mots dans des documents. Je l'utilise toujours pour lire les fichiers qui sont dans le répertoire « **/usr/share/doc** ». C'est la manière la plus rapide de naviguer dans les masses d'informations sur Linux. Ce visualisateur peut être chargé directement utilisant l'une des commandes suivantes :

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

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

1.3.7 Possibilités de démarrage automatique de MC

Pressez Entrée sur un fichier, et le programme approprié prendra en charge le contenu du fichier (consultez Section 9.4.11). Il s'agit là d'une fonctionnalité très pratique de MC.

type de fichier	réaction à la touche Entrée
fichier exécutable	exécuter la commande
fichier de page de manuel	envoyer (« pipe ») le contenu au logiciel de visualisation
fichier html	envoyer (« pipe ») le contenu au navigateur web
fichiers « *.tar.gz » et « *.deb »	parcourir le contenu comme si c'était un sous-répertoire

Table 1.12 – Réaction à la touche Entrée dans MC

Afin de permettre le fonctionnement de ces visualisateurs et de ces fonctionnalités de fichiers virtuels, les fichiers pouvant être visualisés ne doivent pas être définis comme étant exécutables. Modifiez leur état avec la commande **chmod**(1) ou par l'intermédiaire du menu fichiers de MC.

1.3.8 Système de fichiers FTP virtuel de MC

MC peut être utilisé pour accéder à des fichiers au travers d'Internet en utilisant FTP. Allez au menu en pressant F9, entrez ensuite « p » pour activer le système de fichiers virtuel FTP. Entrez une URL sous la forme « `nomutilisateur:motdepasse@nommachine` » qui va permettre de récupérer un répertoire distant qui apparaît alors comme s'il était local.

Essayez l'URL « `[deb.debian.org/debian]` » et parcourez l'archive Debian.

1.4 L'environnement élémentaire de travail de type UNIX

Bien que MC vous permette de faire à peu près n'importe quoi, il est très important que vous appreniez à utiliser les outils en ligne de commande appelés depuis l'invite de l'interpréteur de commandes, et que vous vous familiarisiez avec un environnement de travail de type UNIX.

1.4.1 L'interpréteur de commandes de connexion

Vous pouvez choisir votre interpréteur de commandes de connexion à l'aide de `chsh(1)`.

paquet	popcon	taille	Interpréteur POSIX	description
bash	V:796, I:999	6470	Oui	Bash : Shell GNU Bourne Again. (standard de fait)
bash-completion	V:31, I:922	1523	N/A	programmable completion for the bash shell
dash	V:913, I:993	221	Oui	Le Shell Almquist de Debian. Bon pour les scripts en shell
zsh	V:36, I:74	2467	Oui	Z shell : interpréteur standard avec de nombreuses améliorations
tcsh	V:8, I:28	1316	Non	Shell TENEX C : version améliorée de csh de Berkeley
mksh	V:7, I:12	1478	Oui	Une version de Korn shell
csh	V:2, I:8	343	Non	C Shell OpenBSD , une version de csh de Berkeley
sash	V:0, I:6	1090	Oui	Interpréteur de commandes autonome avec des commandes intégrées. (Ne convient pas en tant que « <code>/bin/sh</code> » standard)
ksh	V:2, I:15	3284	Oui	la vraie version de AT&T du shell Korn
rc	V:0, I:1	169	Non	implémentation du shell rc Plan 9 de AT&T
posh	V:0, I:0	190	Oui	Policy-compliant Ordinary SHell (dérivé de <code>pdcksh</code>)

Table 1.13 – Liste d'interpréteurs de commandes (« shells »)

ASTUCE

Les interpréteurs de commandes POSIX partagent une syntaxe commune, mais leur comportement peut diverger, même pour des choses aussi élémentaires que les variables de l'interpréteur ou les expansions de motifs. Veuillez consulter leur documentation pour une description détaillée.

Dans ce chapitre du didacticiel, l'interpréteur interactif sera toujours `bash`.

1.4.2 Personnaliser bash

Vous pouvez personnaliser le comportement de `bash(1)` à l'aide de « `~/ .bashrc` ».

Essayez, par exemple, ce qui suit :

```
# 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
```

ASTUCE

Vous pourrez trouver davantage d'informations concernant les astuces de personnalisation de bash, comme Section 9.3.6, dans Chapitre 9.

ASTUCE

Le paquet bash-completion permet la complétions programmable pour bash.

1.4.3 Combinaisons particulières de touches

Dans un environnement « [de type UNIX](#) », certaines séquences de touches ont une signification particulière. Vous remarquerez que sur une console Linux normale en mode caractères, seules les touches `Ctrl` et `Alt` situées à gauche fonctionnent de la manière voulue. Voici quelques séquences de touches dont il est intéressant de se souvenir :

ASTUCE

La fonctionnalité `Ctrl-S` du terminal peut être désactivée en utilisant `stty(1)`.

1.4.4 Mouse operations

[Mouse operations for text on Debian system mix 2 styles](#) with some twists:

- Traditional Unix style mouse operations:
 - use 3 buttons (click)
 - use PRIMARY
 - used by X applications such as `xterm` and text applications in Linux console
 - Modern GUI style mouse operations:
-

touche	description des raccourcis clavier
Ctrl-U	effacer la ligne avant le curseur
Ctrl-H	effacer le caractère précédant le curseur
Ctrl-D	terminer l'entrée (quitter l'interpréteur si vous en utilisez un)
Ctrl-C	terminer un programme en cours d'exécution
Ctrl-Z	arrêter temporairement un programme en le mettant en tâche de fond
Ctrl-S	arrêter le défilement de l'affichage à l'écran
Ctrl-Q	reprendre le défilement de l'affichage
Ctrl-Alt-Suppr	redémarrer ou arrêter le système, consultez <code>inittab(5)</code> .
Touche Alt de gauche (optionnellement, touche Windows)	touche « meta » pour les interfaces utilisateurs Emacs et similaires
Flèche haute	lancer la recherche dans l'historique des commandes sous <code>bash</code>
Ctrl-R	lancer la recherche incrémentale dans l'historique des commandes sous <code>bash</code>
Tabulation	compléter l'entrée du nom de fichier de la ligne de commandes sous <code>bash</code>
Ctrl-V Tab	entrer une <code>Tabulation</code> sans expansion de la ligne de commande sous <code>bash</code>

Table 1.14 – Liste des raccourcis clavier de `bash`

- use 2 buttons (drag + click)
- use PRIMARY and CLIPBOARD
- used in Modern GUI applications such as `gnome-terminal`

action	réponse
Clic-gauche et glisser de la souris	select range as PRIMARY selection
Clic-gauche	select the start of range for PRIMARY selection
Right-click (traditional)	select the end of range for PRIMARY selection
Right-click (modern)	context dependent menu (cut/copy/paste)
Middle-click or Shift-Ins	insert PRIMARY selection at the cursor
Ctrl-X	cut PRIMARY selection to CLIPBOARD
Ctrl-C (Shift-Ctrl-C in terminal)	copy PRIMARY selection to CLIPBOARD
Ctrl-V	paste CLIPBOARD at the cursor

Table 1.15 – List of mouse operations and related key actions on Debian

Here, the PRIMARY selection is the highlighted text range. Within the terminal program, `Shift-Ctrl-C` is used instead to avoid terminating a running program.

The center wheel on the modern wheel mouse is considered middle mouse button and can be used for middle-click. Clicking left and right mouse buttons together serves as the middle-click under the 2 button mouse system situation.

In order to use a mouse in Linux character consoles, you need to have `gpm(8)` running as daemon.

1.4.5 Le visualisateur de fichiers

The `less(1)` command is the enhanced pager (file content browser). It reads the file specified by its command argument or its standard input. Hit "h" if you need help while browsing with the `less` command. It can do much more than `more(1)` and can be supercharged by executing `"eval $(lesspipe)"` or `"eval $(lessfile)"` in the shell startup script. See more in `"/usr/share/doc/less/LESSOPEN"`. The `"-R"` option allows raw character output and enables ANSI color escape sequences. See `less(1)`.

1.4.6 L'éditeur de texte

Il faudrait que vous soyez compétant avec l'une des variantes des programmes [Vim](#) ou [Emacs](#) qui sont très populaires sur les systèmes semblables à UNIX.

Je pense que s'habituer aux commandes de Vim est une bonne chose, car l'éditeur Vi est toujours présent dans le monde Linux et UNIX. (En pratique, le `vi` d'origine ou le nouveau `nvi` sont des programmes que vous trouvez partout. Pour les débutants, j'ai plutôt choisi Vim parce qu'il propose de l'aide par l'intermédiaire de la touche F1 tout en restant assez semblable et plus puissant).

Si vous choisissez plutôt [Emacs](#) ou [XEmacs](#) comme éditeur, c'est aussi un bon choix évidemment, particulièrement pour la programmation. Emacs possède une pléthore d'autres fonctionnalités, y compris un lecteur de nouvelles, un éditeur de répertoires, un programme de courriel, etc. Lorsqu'il est utilisé pour programmer ou éditer des scripts en shell, il reconnaît de manière intelligente le format de ce sur quoi vous êtes en train de travailler et il essaie de vous aider. Certaines personnes affirment que le seul programme dont ils ont besoin sous Linux est Emacs. Dix minutes d'apprentissage d'Emacs maintenant vous économiseront des heures plus tard. Il est grandement recommandé d'avoir le manuel de GNU Emacs comme référence lors de son apprentissage.

Tous ces programmes sont habituellement accompagnés d'un programme d'apprentissage pour vous aider à les utiliser par la pratique. Lancez Vim en entrant « `vim` » et en pressant la touche F1. Vous devriez au moins en lire les 35 premières lignes. Suivez ensuite le cours en ligne en déplaçant le curseur sur « `|tutor|` » et en pressant `Ctrl-J`.

Note

Good editors, such as Vim and Emacs, can handle UTF-8 and other exotic encoding texts correctly. It is a good idea to use the GUI environment in the UTF-8 locale and to install required programs and fonts to it. Editors have options to set the file encoding independent of the GUI environment. Please refer to their documentation on multibyte text.

1.4.7 Définir un éditeur de texte par défaut

Debian est fourni avec de nombreux éditeurs différents. Nous recommandons d'installer le paquet `vim`, comme indiqué ci-dessus.

Debian offre un accès unifié à l'éditeur par défaut du système par l'intermédiaire de la commande « `/usr/bin/editor` ». Cela permet à d'autres programmes (par exemple `reportbug(1)`) de pourvoir l'appeler. Vous pouvez le modifier par la commande qui suit :

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

Pour les débutants, je recommande de choisir « `/usr/bin/vim.basic` » plutôt que « `/usr/bin/vim.tiny` » car il prend en charge la mise en évidence de la syntaxe.

ASTUCE

De nombreux programmes utilisent les variables d'environnement « `$EDITOR` » ou « `$VISUAL` » pour décider de l'éditeur à utiliser (consultez Section [1.3.5](#) et Section [9.4.11](#)). Pour des raisons de cohérence sur le système Debian, définissez-les à « `/usr/bin/editor` ». (Historiquement « `$EDITOR` » était défini à « `ed` » et « `$VISUAL` » était défini à « `vi` »).

1.4.8 Using vim

The recent `vim(1)` starts itself in the sane "nocompatible" option and enters into the NORMAL mode. ¹

Please use the "vimtutor" program to learn vim through an interactive tutorial course.

The `vim` program changes its behavior to typed key strokes based on **mode**. Typing in key strokes to the buffer is mostly done in INSERT-mode and REPLACE-mode. Moving cursor is mostly done in NORMAL-mode. Interactive selection is done in VISUAL-mode. Typing ":" in NORMAL-mode changes its **mode** to Ex-mode. Ex-mode accepts commands.

1. Even the older `vim` can starts in the sane "nocompatible" mode by starting it with the "-N" option.

mode	key strokes	action
NORMAL	:help only	display the help file
NORMAL	:e filename.ext	open new buffer to edit filename.ext
NORMAL	:w	overwrite current buffer to the original file
NORMAL	:w filename.ext	write current buffer to filename.ext
NORMAL	:q	quit vim
NORMAL	:q!	force to quit vim
NORMAL	:only	close all other split open windows
NORMAL	:set nocompatible?	check if vim is in the sane nocompatible mode
NORMAL	:set nocompatible	set vim to the sane nocompatible mode
NORMAL	i	enter the INSERT mode
NORMAL	R	enter the REPLACE mode
NORMAL	v	enter the VISUAL mode
NORMAL	V	enter the linewise VISUAL mode
NORMAL	Ctrl-V	enter the blockwise VISUAL mode
except TERMINAL - JOB	ESC-key	enter the NORMAL mode
NORMAL	:term	enter the TERMINAL - JOB mode
TERMINAL - NORMAL	i	enter the TERMINAL - JOB mode
TERMINAL - JOB	Ctrl-W N (or Ctrl-\ Ctrl-N)	enter the TERMINAL - NORMAL mode
TERMINAL - JOB	Ctrl-W :	enter the Ex-mode in TERMINAL - NORMAL mode

Table 1.16 – List of basic Vim key strokes

ASTUCE

The Vim comes with the **Netrw** package. Netrw supports reading files, writing files, browsing directories over a network, and local browsing! Try Netrw with "vim ." (a period as the argument) and read its manual at ":help netrw".

For the advanced configuration of vim, see Section [9.2](#).

1.4.9 Enregistrer les actions de l'interpréteur de commandes

La sortie d'une commande de l'interpréteur peut défiler, quitter votre écran et être définitivement perdue. C'est une bonne habitude d'enregistrer l'activité de l'interpréteur de commandes dans un fichier afin de la consulter plus tard. Ce type d'enregistrement est essentiel lorsque vous effectuez des tâches d'administration quelconques.

ASTUCE

The new Vim (version>=8.2) can be used to record the shell activities cleanly using TERMINAL - JOB-mode. See Section [1.4.8](#).

La méthode de base pour enregistrer l'activité de l'interpréteur de commandes est de la lancer sous script(1).

Essayez, par exemple, ce qui suit :

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

Lancez une commande quelconque sous script.

Pressez Ctrl-D pour quitter le script.

```
$ vim typescript
```

Consultez Section [9.1.1](#).

1.4.10 Commandes UNIX de base

Apprenons les commandes UNIX de base. J'utilise ici « UNIX » dans son sens générique. Tous les clones d'UNIX proposent habituellement des commandes équivalentes. Le système Debian ne fait pas exception. Ne vous inquiétez pas si certaines commandes ne fonctionnent pas comme vous le voudriez maintenant. Si un alias est utilisé dans le shell, la sortie correspondante sera différente. Ces exemples ne sont pas destinés à être exécutés dans cet ordre.

Essayez toutes les commandes qui suivent en utilisant un compte non privilégié :

Note

UNIX a pour tradition de cacher les fichiers dont le nom commence par un « . ». Ce sont traditionnellement des fichiers qui contiennent des informations de configuration et des préférences de l'utilisateur.

Pour la commande `cd`, consultez `builtin(7)`.

Le visualisateur (« pager ») par défaut d'un système Debian non personnalisé est `more(1)` qui ne permet pas le défilement vers l'arrière. En installant le paquet `less` à l'aide de la ligne de commandes « `apt-get install less` », `less(1)` deviendra le visualisateur par défaut et vous pourrez faire défiler le texte vers l'arrière à l'aide des touches de curseur.

« [» et «] » dans l'expression rationnelle de la commande « `ps aux | grep -e "[e]xim4*"` » ci-dessus permet d'éviter une correspondance de `grep` avec lui-même. Le « `4*` » de l'expression rationnelle signifie 0 ou plusieurs instances du caractère « `4` » et permet donc à `grep` de trouver la correspondance à la fois avec « `exim` » et « `exim4` ». Bien que « `*` » soit utilisé dans le motif générique (« glob ») des noms de fichiers de l'interpréteur de commandes et dans l'expression rationnelle, leurs significations sont différentes. Vous pourrez apprendre les expressions rationnelles dans `grep(1)`.

À titre d'exercice, parcourez les répertoires et jetez un coup d'œil au système en vous servant des commandes ci-dessus. Si vous avez des questions sur ces commandes de la console, veuillez consulter la page de manuel.

Essayez, par exemple, ce qui suit :

```
$ man man
$ man bash
$ man builtin
$ man grep
$ man ls
```

Il peut être un peu difficile de s'habituer au style des pages de manuel parce qu'elles sont plutôt succinctes, particulièrement les plus anciennes, celles qui sont vraiment traditionnelles. Mais une fois que vous y serez familiarisé, vous apprécierez leur concision.

Remarquez que beaucoup de commandes UNIX, y compris celles de GNU et BSD, affichent une information d'aide courte si vous les exécutez de l'une des façons suivantes (ou parfois sans paramètre) :

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

1.5 La commande simple de l'interpréteur de commandes

Vous avez maintenant une certaine sensation sur la manière d'utiliser un système Debian. Nous allons regarder plus profondément le mécanisme d'exécution des commandes sous le système Debian. J'ai ici, pour les débutants, simplifié la réalité. Consultez `bash(1)` pour l'explication exacte.

Une simple commande est une séquence de :

1. assignations de variables (optionnelles) ;
 2. nom de la commande ;
 3. paramètres (optionnels) ;
 4. redirections (optionnelles : `>` , `>>` , `<` , `<<` , etc.) ;
 5. opérateurs de contrôle (optionnels : `&&` , `||` , `<nouvelle ligne>` , `;` , `&` , `(,)`).
-

commande	description
<code>pwd</code>	afficher le nom du répertoire actuel ou de travail
<code>whoami</code>	afficher le nom de l'utilisateur actuel.
<code>id</code>	afficher l'identité de l'utilisateur actuel (nom, uid, gid, et groupes associés)
<code>file toto</code>	afficher le type de fichier du fichier « <i>toto</i> »
<code>type -p nom-de-commande</code>	afficher l'emplacement du fichier de la commande « <i>nom-de-commande</i> »
<code>which nom-de-commande</code>	, ,
<code>type nom-de-commande</code>	afficher des informations sur la commande « <i>nom-de-commande</i> »
<code>apropos mot-clé</code>	rechercher les commandes ayant un rapport avec « <i>mot-clé</i> »
<code>man -k mot-clé</code>	, ,
<code>whatis nom-de-commande</code>	afficher une ligne d'explication sur la commande « <i>nom-de-commande</i> »
<code>man -a nom-de-commande</code>	afficher une explication sur la commande « <i>nom-de-commande</i> » (style UNIX)
<code>info nom-de-commande</code>	afficher une explication assez longue de la commande « <i>nom-de-commande</i> » (style GNU)
<code>ls</code>	afficher le contenu du répertoire (tous les fichiers et répertoires non cachés)
<code>ls -a</code>	afficher le contenu du répertoire (tous les fichiers et répertoires)
<code>ls -A</code>	afficher le contenu du répertoire (presque tous les fichiers et répertoires, par exemple sauter « . » et « .. »)
<code>ls -la</code>	afficher tout le contenu du répertoire de façon détaillée
<code>ls -lai</code>	afficher tout le contenu du répertoire avec les numéros d'inœuds et les informations détaillées
<code>ls -d</code>	afficher tous les sous-répertoires du répertoire actuel
<code>tree</code>	afficher le contenu de l'arborescence des fichiers
<code>lsuf toto</code>	afficher l'état d'ouverture du fichier « <i>toto</i> »
<code>lsuf -p pid</code>	afficher les fichiers ouverts par le processus de numéro : « <i>pid</i> »
<code>mkdir toto</code>	créer le nouveau répertoire « <i>toto</i> » dans le répertoire en cours
<code>rmdir toto</code>	supprimer le répertoire « <i>toto</i> » du répertoire actuel
<code>cd toto</code>	allez au répertoire « <i>toto</i> » se trouvant dans le répertoire actuel ou dans le répertoire figurant dans la variable « <i>\$CDPATH</i> »
<code>cd /</code>	aller au répertoire racine
<code>cd</code>	aller au répertoire personnel de l'utilisateur actuel
<code>cd /toto</code>	aller au répertoire de chemin absolu « <i>/toto</i> »
<code>cd ..</code>	aller au répertoire parent
<code>cd ~toto</code>	aller au répertoire « <i>foo</i> » se trouvant dans le répertoire personnel de l'utilisateur
<code>cd -</code>	aller au répertoire précédent
<code></etc/motd visualisateur</code>	afficher le contenu de « <i>/etc/motd</i> » en utilisant le visualisateur (« pager ») par défaut
<code>touch fichier-poubelle</code>	créer un fichier vide « <i>fichier-poubelle</i> »
<code>cp toto titi</code>	copier le fichier « <i>toto</i> » existant dans le nouveau fichier « <i>titi</i> »
<code>rm fichier-poubelle</code>	supprimer le fichier « <i>fichier-poubelle</i> »
<code>mv toto titi</code>	renommer le fichier existant « <i>toto</i> » avec le nouveau nom « <i>titi</i> » (« <i>titi</i> » ne doit pas exister)
<code>mv toto titi</code>	déplacer le fichier existant « <i>toto</i> » vers le nouvel emplacement « <i>titi/toto</i> » (le répertoire « <i>titi</i> » doit exister)
<code>mv toto titi/tutu</code>	déplacer le fichier existant « <i>toto</i> » vers un nouvel emplacement avec le nouveau nom « <i>titi/tutu</i> » (le répertoire « <i>titi</i> » doit exister mais le répertoire « <i>titi/tutu</i> » ne doit pas exister)
<code>chmod 600 toto</code>	rendre le fichier « <i>toto</i> » non lisible et non modifiable par les autres personnes (non exécutable pour tous)
<code>chmod 644 toto</code>	rendre un fichier existant « <i>toto</i> » accessible en lecture mais non modifiable par les autres personnes (non exécutable pour tous)
<code>chmod 755 toto</code>	rendre un fichier existant « <i>toto</i> » accessible en lecture mais non modifiable par les autres utilisateurs (exécutable pour tous)
<code>find . -name motif</code>	rechercher les noms de fichier contenant le « <i>motif</i> » de l'interpréteur de commandes (plus lent).
<code>locate -d . motif</code>	rechercher les noms de fichiers contenant en utilisant un « <i>motif</i> » de l'interpréteur de commandes (plus rapide si on utilise une base de données régulièrement générée)

1.5.1 Exécution d'une commande et variables d'environnement

Les valeurs de certaines [variables d'environnement](#) modifient le comportement de certaines commandes UNIX.

Les valeurs par défaut des variables d'environnement sont définies initialement par le système PAM, certaines d'entre-elles peuvent donc être réinitialisées par certains programmes d'application :

- The PAM system such as `pam_env` may set environment variables by `/etc/pam.conf`, `/etc/environment` and `/etc/default/locale`.
- The display manager such as `gdm3` may reset environment variables for GUI session.
- The user specific program initialization may reset environment variables by `~/.profile`, `~/.bash_profile` and `~/.bashrc`.

1.5.2 La variable « \$LANG »

The default locale is defined in the `"$LANG"` environment variable and is configured as `"LANG=xx_YY.UTF-8"` by the installer or by the subsequent GUI configuration, e.g., "Settings" → "Region & Language" → "Language" / "Formats" for GNOME.

Note

I recommend you to configure the system environment just by the `"$LANG"` variable for now and to stay away from `"$LC_*` variables unless it is absolutely needed.

La valeur complète des paramètres linguistiques indiqués par la variable « \$LANG est constituée de trois parties « `xx_YY.ZZZZ` ».

valeur des paramètres linguistiques	signification
xx	codes de langue ISO 639 (en minuscules) tel que « fr »
YY	codes de pays ISO 3166 (en majuscules), par exemple « FR »
ZZZZ	le jeu de caractères, toujours défini à « UTF-8 »

Table 1.18 – Les trois parties des paramètres linguistiques

recommandation de paramètres linguistiques	Langue (zone)
en_US.UTF-8	anglais (USA)
en_GB.UTF-8	anglais (Grande-Bretagne)
fr_FR.UTF-8	français (France)
de_DE.UTF-8	allemand (Allemagne)
it_IT.UTF-8	italien (Italie)
es_ES.UTF-8	espagnol (Espagne)
ca_ES.UTF-8	catalan (Espagne)
sv_SE.UTF-8	suédois (Suède)
pt_BR.UTF-8	portugais (Brésil)
ru_RU.UTF-8	russe (Russie)
zh_CN.UTF-8	chinois (RP de Chine)
zh_TW.UTF-8	chinois (Taiwan)
ja_JP.UTF-8	japonais (Japon)
ko_KR.UTF-8	coréen (République de Corée)
vi_VN.UTF-8	vietnamien (Vietnam)

Table 1.19 – Liste des recommandations de paramètres linguistiques

L'exécution typique d'une commande utilise une séquence de lignes telle que la suivante :

```
$ 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
```

Here, the program `date(1)` is executed with different locale values.

- Avec la première commande, « `$LANG` » est définie à la valeur des [paramètres linguistiques](#) par défaut du système « `fr_FR.UTF-8` ».
- Avec la deuxième commande, « `$LANG` » est définie à la valeur des [paramètres linguistiques](#) UTF-8 anglais des États-Unis « `en_US.UTF-8` ».

Habituellement, la plupart des exécutions de commandes ne sont pas précédées de la définition de variables d’environnement. Pour les exemples ci-dessus, vous pouvez aussi exécuter :

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

ASTUCE

When filing a bug report, running and checking the command under "en_US.UTF-8" locale is a good idea if you use non-English environment.

Pour des informations détaillées sur la configuration des paramètres linguistiques, consultez Section [8.1](#).

1.5.3 La variable « `$PATH` »

Lorsque vous entrez une commande dans l’interpréteur, il recherche la commande dans la liste des répertoires contenus dans la variable d’environnement « `$PATH` ». La valeur de la variable d’environnement « `$PATH` » est aussi appelée « chemin de recherche de l’interpréteur de commandes ».

Dans une installation Debian par défaut, la variable d’environnement « `$PATH` » des comptes d’utilisateurs peut ne pas inclure « `/sbin` » ni « `/usr/sbin` ». Par exemple, la commande `ifconfig` doit être lancée avec son chemin complet « `/sbin/ifconfig` » (La commande similaire `ip` est située dans « `/bin` ».)

Vous pouvez modifier la variable d’environnement « `$PATH` » de l’interpréteur de commandes Bash par l’intermédiaire des fichiers « `~/.bash_profile` » ou « `~/.bashrc` ».

1.5.4 La variable « `$HOME` »

De nombreuses commandes enregistrent la configuration spécifique à un utilisateur dans son répertoire personnel et modifient leur comportement en fonction de son contenu. Le répertoire personnel est identifié par la variable d’environnement « `$HOME` ».

ASTUCE

L’interpréteur de commandes étend « `~/` » pour former le répertoire personnel réel de l’utilisateur, par exemple, « `$HOME/` ». L’interpréteur de commandes étend « `~foo/` » sous la forme du répertoire personnel de `foo`, par exemple, « `/home/foo/` ».

valeur de « \$HOME »	situation d'exécution d'un programme
/	programme lancé par le processus init (démon)
/root	programme lancé depuis l'interpréteur de commandes normal de l'administrateur (« root »)
/home/utilisateur_normal	programme lancé depuis l'interpréteur de commandes d'un utilisateur normal
/home/utilisateur_normal	programme lancé depuis le menu du bureau de l'interface graphique de l'utilisateur
/home/utilisateur_normal	programme lancé en tant qu'administrateur avec « sudo programme »
/root	programme lancé en tant qu'administrateur avec « sudo -H programme »

Table 1.20 – Afficher les valeurs de la variable « \$HOME »

1.5.5 Options de la ligne de commandes

Certaines commandes prennent des paramètres. Les paramètres qui commencent par « - » ou « -- » sont appelés options et contrôlent le comportement de la commande.

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

Ici, le paramètre de la ligne de commandes « -R » modifie le comportement de la commande `date(1)` afin qu'elle donne en sortie une chaîne de date conforme à la [RFC2822](#).

1.5.6 Motifs génériques (« glob ») de l'interpréteur de commandes

Souvent, vous voudrez utiliser une commande sur un groupe de fichiers sans avoir à tous les entrer. C'est facilité par l'utilisation des motifs génériques d'expansion du nom de fichier de l'interpréteur de commandes (**glob**), (on les appelle parfois **jokers**).

motif générique de l'interpréteur	description de la règle de correspondance
*	nom de fichier (segment) ne commençant pas par « . »
.*	nom de fichier (segment) commençant par « . »
?	exactement un caractère
[...]	exactement un caractère, chaque caractère étant l'un de ceux entre crochets
[a-z]	exactement un caractère, chaque caractère étant compris entre « a » et « z »
[^...]	exactement un caractère devant être différent de tous les caractères entre crochets (sauf « ^ »)

Table 1.21 – Motifs génériques d'expansion du nom de fichier de l'interpréteur de commandes

Essayez, par exemple, ce qui suit :

```
$ 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
```

Consultez `glob(7)`.

Note

Contrairement à l'expansion normale du nom de fichier par l'interpréteur de commandes, le motif « `*` » de l'interpréteur testé par `find(1)` avec « `-name` » `test etc.`, correspond au « `.` » du nom de fichier. (Nouvelle fonctionnalité [POSIX](#)).

Note

On peut modifier le comportement d'expansion de fichiers selon des motifs génériques de BASH (« `glob` ») avec ses options « `shopt` » incluses telles que « `dotglob` », « `noglob` », « `nocaseglob` », « `nullglob` », « `extglob` », etc. Consultez `bash(1)`.

1.5.7 Valeur de retour d'une commande

Toutes les commandes retournent comme valeur de retour leur état de fin d'exécution (variable : « `$?` »).

état de sortie de la commande	valeur numérique de retour	valeur logique de retour
succès	zéro, 0	VRAI
erreur	non-nulle, -1	FAUX

Table 1.22 – Codes de retour de la commande

Essayez, par exemple, ce qui suit :

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

Note

Vous remarquerez que, dans le contexte logique de l'interpréteur de commandes, un **succès** est traité comme la valeur logique **VRAIE** qui possède la valeur 0 (zéro). C'est parfois un peu contre-intuitif et il fallait le rappeler ici.

1.5.8 Séquences de commandes typiques et redirection de l'interpréteur de commandes

Essayez de retenir les idiomes suivants de l'interpréteur de commandes entrés sur une seule ligne en tant qu'extrait d'une commande de l'interpréteur.

Le système Debian est un système multi-tâches. Les travaux s'exécutant en arrière-plan permettent aux utilisateurs de faire tourner plusieurs programmes depuis un seul interpréteur de commandes. La gestion des processus en arrière-plan fait appel aux commandes internes de l'interpréteur : `jobs`, `fg`, `bg` et `kill`. Veuillez lire les sections de `bash(1)` se trouvant sous « **SIGNAUX** » et « **CONTRÔLE DES TÂCHES** » ainsi que `builtin(1)`.

Essayez, par exemple, ce qui suit :

idiome de commande	description
commande &	exécuter la commande en arrière-plan dans le sous-shell
commande1 commande2	la sortie standard de commande1 est passée (« piped ») à l'entrée standard de commande2 . Les deux commandes peuvent tourner simultanément
commande1 2>&1 commande2	La sortie standard et la sortie d'erreur standard de commande1 sont toutes les deux passées à l'entrée standard de commande2 . Les deux commandes peuvent tourner simultanément
commande1 ; commande2	exécuter commande1 et commande2 séquentiellement
commande1 && commande2	exécuter commande1 , en cas de succès, exécuter commande2 séquentiellement (retourne un succès si à la fois commande1 et commande2 ont été réussies)
commande1 commande2	exécuter commande1 , en cas d'échec, exécuter commande2 séquentiellement (retourne un succès si commande1 ou commande2 a été réussie)
commande > toto	rediriger la sortie standard de commande vers le fichier toto (l'écraser)
commande 2> toto	rediriger la sortie d'erreur standard de la commande vers le fichier toto (et l'écraser)
commande >> toto	rediriger la sortie standard de la commande vers le fichier toto (ajouter à la fin du fichier)
commande 2>> toto	rediriger la sortie d'erreur standard de la commande vers le fichier toto (ajouter à la fin du fichier)
commande > toto 2>&1	rediriger à la fois la sortie standard et l'erreur standard de la commande vers le fichier « toto »
commande < toto	rediriger l'entrée standard de la commande vers le fichier toto
commande << délimiteur	rediriger l'entrée standard de la commande vers les lignes suivantes jusqu'à ce que le « délimiteur » soit rencontré (ce document)
command <<- délimiteur	rediriger l'entrée standard de commande vers les lignes qui suivent jusqu'à ce que le « délimiteur » soit rencontré, les caractères de tabulation de tête sont supprimés des lignes d'entrée)

Table 1.23 – Idiomes des commandes de l'interpréteur

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

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

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

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

Bien que ces 4 exemples de redirections d’interpréteur de commande affichent la même chose, le dernier exemple utilise la commande supplémentaire `cat` et gaspille des ressources sans raison.

L’interpréteur de commande vous permet d’ouvrir des fichiers en utilisant la commande interne `exec` avec un descripteur de fichier arbitraire.

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

Les descripteurs de fichiers 0-2 sont prédéfinis.

périphérique	description	descripteur de fichier
stdin	entrée standard	0
stdout	sortie standard	1
stderr	erreur standard	2

Table 1.24 – Descripteurs de fichier prédéfinis

1.5.9 Alias de commande

Vous pouvez définir des alias pour les commandes fréquemment utilisées.

Essayez, par exemple, ce qui suit :

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

Maintenant, « `la` » fonctionnera comme un raccourci pour « `ls -la` » qui donne la liste de tous les fichiers dans le format de liste long.

Vous pouvez afficher la liste de tous les alias existants par la commande `alias` (consultez `bash(1)` sous « COMMANDES INTERNES DU SHELL »).

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

Vous pouvez identifier le chemin exact ou identifier la commande par `type` (consultez `bash(1)` sous « COMMANDES INTERNES DU SHELL »).

Essayez, par exemple, ce qui suit :

```
$ 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
```

Ici, `ls` a été récemment recherché alors que « `file` » ne l’a pas été, donc « `ls` » est « hachée », c’est-à-dire que l’interpréteur de commandes possède un enregistrement interne permettant un accès rapide à l’emplacement de la commande « `ls` ».

ASTUCE

Consultez Section [9.3.6](#).

1.6 Traitement des données textuelles à la UNIX

Dans un environnement de travail à la UNIX, le traitement du texte est effectué en passant le texte par des tubes au travers d’une chaîne d’outils standards de traitement de texte. C’est une autre innovation cruciale d’UNIX.

1.6.1 Outils de traitement de texte d’UNIX

Il existe quelques outils standard de traitement de texte qui sont très souvent utilisés sur les systèmes « UNIX-like ».

- Aucune expression rationnelle n’est utilisée :
 - `cat(1)` concatène des fichiers et en affiche le contenu complet ;
 - `tac(1)` concatène des fichiers et les affiche en ordre inverse ;
 - `cut(1)` sélectionne des parties de lignes et les affiche ;
 - `head(1)` affiche le début d’un fichier ;
 - `tail(1)` affiche la fin d’un fichier ;
 - `sort(1)` trie des lignes de texte ;
 - `uniq(1)` supprime les lignes dupliquées d’un fichier trié ;
 - `tr(1)` traduit ou supprime des caractères ;
 - `diff(1)` compare des fichiers ligne par ligne.
- Basic regular expression (**BRE**) is used as default:
 - `ed(1)` est un éditeur par ligne primitif ;
 - `sed(1)` est un éditeur de flux ;
 - `grep(1)` analyse la correspondance d’un texte avec des motifs ;
 - `vim(1)` est un éditeur en mode écran ;
 - `emacs(1)` est un éditeur en mode écran. (un peu étendu **BRE**).
- Une expression rationnelle étendue (« Extended regular expression - ERE ») est utilisée :
 - `awk(1)` effectue un traitement simple du texte ;
 - `egrep(1)` fait correspondre du texte avec des motifs ;
 - `tc(1)` peut effectuer tous les traitements possibles du texte : voir `re_syntax(3)`. Souvent utilisé avec `tk(3)` ;
 - `perl(1)` peut effectuer tous les traitements imaginables sur du texte. Voir `perlre(1)` ;
 - `pcregrep(1)` du paquet `pcregrep` fait la correspondance de texte avec des motifs d’[expressions rationnelles compatibles avec Perl \(PCRE\)](#) (« Perl Compatible Regular Expressions ») ;
 - `python(1)` avec le module `re` peut faire tous les traitements imaginables sur du texte. Consultez « `/usr/share/doc/python` ».

Si vous n’êtes pas certain de ce que font exactement ces commandes veuillez utiliser la commande « `man` » pour vous en faire une idée par vous-même.

Note

Sort order and range expression are locale dependent. If you wish to obtain traditional behavior for a command, use **C** locale or **C.UTF-8** locale instead of normal **UTF-8** ones (see Section 8.1).

Note

Les expressions rationnelles de [Perl](#) (`perlre(1)`), [Expressions rationnelles compatible avec Perl \(PCRE\)](#) et les expressions rationnelles de [Python](#) proposées par le module `re` ont de nombreuses extensions courantes par rapport aux expressions rationnelles étendues **ERE**.

1.6.2 Expressions rationnelles

Les [expressions rationnelles](#) sont utilisées avec de nombreux outils de traitement du texte. Elles sont analogues aux motifs génériques « globs » du shell mais elles sont plus compliquées et plus puissantes.

L'expression rationnelle décrit le motif de correspondance, elle est constituée de caractères de texte et de **métacaractères**.

Un **métacaractère** est simplement un caractère ayant une signification particulière. Il en existe deux styles principaux, **BRE** et **ERE** suivant les outils de texte décrits ci-dessus.

BRE	ERE	description de l'expression rationnelle
<code>\ . [] ^ \$ *</code>	<code>\ . [] ^ \$ *</code>	métacaractères courants
<code>\+ \? \(\) \{ \} \ </code>		métacaractères uniquement BRE, déspecifiés par « \ »
	<code>+ ? () { } </code>	métacaractères uniquement ERE, non déspecifiés par « \ »
<code>c</code>	<code>c</code>	correspond au non métacaractère « <code>c</code> »
<code>\c</code>	<code>\c</code>	correspond au caractère littéral « <code>c</code> » même si « <code>c</code> » est un métacaractère en lui-même
<code>.</code>	<code>.</code>	correspond à n'importe quel caractère y compris le saut de ligne
<code>^</code>	<code>^</code>	position au début d'une chaîne de caractères
<code>\$</code>	<code>\$</code>	position à la fin d'une chaîne de caractères
<code>\<</code>	<code>\<</code>	position au début d'un mot
<code>\></code>	<code>\></code>	position à la fin d'un mot
<code>[abc...]</code>	<code>[abc...]</code>	correspond à n'importe quel caractère dans « <code>abc...</code> »
<code>[^abc...]</code>	<code>[^abc...]</code>	correspond à n'importe quel caractère sauf ceux se trouvant dans « <code>abc...</code> »
<code>r*</code>	<code>r*</code>	correspond à aucune ou plusieurs instances de l'expression rationnelle identifiée par « <code>r</code> »
<code>r\+</code>	<code>r+</code>	correspond à une ou plusieurs instances de l'expression rationnelle identifiée par « <code>r</code> »
<code>r\?</code>	<code>r?</code>	correspond à aucune ou une instance de l'expression rationnelle identifiée par « <code>r</code> »
<code>r1\ r2</code>	<code>r1 r2</code>	correspond à une instance de l'expression rationnelle identifiée par « <code>r1</code> » ou « <code>r2</code> ».
<code>\(r1\ r2\)</code>	<code>(r1 r2)</code>	correspond à une des expressions rationnelles identifiées par « <code>r1</code> » ou « <code>r2</code> » et la traite comme une expression rationnelle entre crochets

Table 1.25 – Métacaractères pour BRE et ERE

Une expression rationnelle d'**emacs** est **BRE de base** mais elle a été étendue afin de traiter « `+` » et « `?` » comme des **métacaractères** comme dans les **ERE**. Il n'est donc pas nécessaire de les échapper avec « \ » dans les expressions rationnelles d'**emacs**.

`grep(1)` peut être utilisé pour effectuer de la recherche de texte en utilisant une expression rationnelle.

Essayez, par exemple, ce qui suit :

```
$ 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
```

ASTUCE
Consultez Section [9.3.6](#).

1.6.3 Expressions de remplacement

Pour l’expression de remplacement, certains caractères ont une signification particulière.

expressions de remplacement	description du texte destiné à remplacer l’expression de remplacement
<code>&</code>	ce à quoi correspond l’expression rationnelle (utilisez <code>\&</code> avec <code>emacs</code>)
<code>\n</code>	ce à quoi la nième expression rationnelle entre crochets correspond (« <code>n</code> » étant un nombre)

Table 1.26 – Expressions de remplacement

Pour chaîne de caractères Perl de remplacement, “`$&`” est utilisé au lieu de “`&`” et “`$n`” est utilisé au lieu de “`\n`”.

Essayez, par exemple, ce qui suit :

```
$ echo zzz1abc2efg3hij4 | \
sed -e 's/\(1[a-z]*\)[0-9]*\(.*\)$/=&/'
zzz=1abc2efg3hij4=
$ echo zzz1abc2efg3hij4 | \
sed -e 's/\(1[a-z]*\)[0-9]*\(.*\)$/\2===\1/'
zzze fg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
perl -pe 's/(1[a-z]*)[0-9]*(.*)$/$2===$1/'
zzze fg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
perl -pe 's/(1[a-z]*)[0-9]*(.*)$/=&/'
zzz=1abc2efg3hij4=
```

Vous prêterez ici une attention particulière au style de l’expression rationnelle **entre crochets** et à la manière dont les chaînes qui correspondent sont utilisées dans le processus de remplacement du texte avec les différents outils.

Ces expressions rationnelles peuvent aussi être utilisées pour les déplacements du curseur et des actions de remplacement de texte dans certains éditeurs.

Un « back slash » (« `\` ») en fin de ligne sur la ligne de commandes du shell déspecifie le saut de ligne en un caractère d’espace et permet de poursuivre la ligne de commandes de l’interpréteur sur la ligne suivante.

Pour apprendre ces commandes, veuillez lire les pages de manuel correspondantes.

1.6.4 Substitution globale avec des expressions rationnelles

La commande `ed(1)` peut remplacer toutes les instances de « `REGEX_SOURCE` » par « `TEXTE_DESTINATION` » dans « `fichier` » :

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

La commande `sed(1)` peut remplacer toutes les instances de « `REGEX_SOURCE` » par « `TEXTE_DESTINATION` » dans « `fichier` » :

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

La commande `vim(1)` peut remplacer toutes les instances de « `REGEX_SOURCE` » avec « `TEXTE_DESTINATION` » dans « `file` » en utilisant les commandes « `ex(1)` » :

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

ASTUCE

L'indicateur « `c` » dans l'exemple qui précède permet une confirmation interactive de chaque substitution.

Plusieurs fichiers (« `fichier1` », « `fichier2` » et « `fichier3` ») peuvent être traités de manière similaire par une expression rationnelle avec `vim(1)` ou `perl(1)` :

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

ASTUCE

L'indicateur « `e` » de l'exemple précédent évite qu'une erreur « `No match` » (pas de correspondance) ne casse un mapping :

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

Dans l'exemple en `perl(1)` ci-dessus, le « `-i` » force l'édition directe de chaque fichier cible, et le « `-p` » induit une boucle implicite sur tous les fichiers.

ASTUCE

L'utilisation du paramètre « `-i.bak` » à la place de « `-i` » conserve le fichier d'origine en ajoutant « `.bak` » à son nom de fichier. Cela permet la récupération plus facile d'erreurs lors de substitutions complexes.

Note

`ed(1)` et `vim(1)` sont **BRE** ; `perl(1)` est **ERE**.

1.6.5 Extraire des données d'un tableau contenu dans un fichier texte

Considérons un fichier texte « `DPL` » dans lequel les noms de certains des responsables du projet Debian d'avant 2004 et leur date d'investiture sont présentés dans un format séparés par des espaces :

```
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
```

ASTUCE

Consultez « [Bref historique de Debian](#) » pour la dernière [Debian leadership history](#).

Awk est fréquemment utilisé pour extraire des données de ce type de fichiers.

Essayez, par exemple, ce qui suit :

```
$ 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
```

Des interpréteurs de commandes comme Bash peuvent aussi être utilisés pour analyser ce genre de fichiers.

Essayez, par exemple, ce qui suit :

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

Ici, la commande interne `read` utilise les caractères de « `$IFS` » (séparateurs de champs internes) pour scinder les lignes en mots.

Si vous changez « `$IFS` » en « `:` », vous pouvez analyser facilement le fichier « `/etc/passwd` » :

```
$ 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
```

(Si Awk est utilisé pour faire la même chose, utilisez « `FS=':'` » pour définir le séparateur de champs).

IFS est aussi utilisé par l'interpréteur de commandes pour scinder le résultat de l'expansion des paramètres, de la substitution de commande et de l'expansion arithmétique. Cela ne se produit pas pour les mots entre double ou simple apostrophes. La valeur de IFS par défaut est *space*, *tab* et *newline* combinés.

Faites attention en utilisant cette astuce IFS dans un shell. Des choses étranges peuvent survenir lorsque le shell interprète certaines parties du script comme son **entrée**.

```
$ IFS=":,"          # use ":" and "," as IFS
$ echo 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 Bouts de scripts pour les tubes

Les scripts suivants font des choses sympas avec les tubes.

fragment de script (à entrer sur une seule ligne)	effet de la commande
<code>find /usr -print</code>	rechercher tous les fichiers se trouvant sous « /usr »
<code>seq 1 100</code>	imprimer 1 à 100
<code> xargs -n 1 <i>commande</i></code>	lancer la commande de manière répétitive en utilisant chaque élément provenant du tube (pipe) comme paramètre
<code> xargs -n 1 echo</code>	scinder les éléments séparés par des espaces provenant du tube (pipe) en différentes lignes
<code> xargs echo</code>	concaténer les lignes provenant du tube en une seule ligne
<code> grep -e <i>motif_expression_rationnelle</i></code>	extraire du tube les lignes contenant <i>motif_expression_rationnelle</i>
<code> grep -v -e <i>motif_expression_rationnelle</i></code>	extraire du tube les lignes ne contenant pas <i>motif_expression_rationnelle</i>
<code> cut -d: -f3 -</code>	extraire du tube le troisième champ, séparé par « : » (fichier passwd, etc.)
<code> awk '{ print \$3 }'</code>	extraire du tube le troisième champ séparé par des caractères d'espacement
<code> awk -F'\t' '{ print \$3 }'</code>	extraire du tube le troisième champ séparé par une tabulation
<code> col -bx</code>	supprimer le retour arrière (backspace) et convertir les tabulations en espaces
<code> expand -</code>	convertir les tabulations en espaces
<code> sort uniq</code>	trier et supprimer les doublons
<code> tr 'A-Z' 'a-z'</code>	convertir de majuscules en minuscules
<code> tr -d '\n'</code>	concaténer les lignes en une seule ligne
<code> tr -d '\r'</code>	supprimer le retour à la ligne (CR)
<code> sed 's/^/# /'</code>	ajouter « # » au début de chaque ligne
<code> sed 's/\.ext//g'</code>	supprimer « .ext »
<code> sed -n -e 2p</code>	afficher la seconde ligne
<code> head -n 2 -</code>	afficher les deux premières lignes
<code> tail -n 2 -</code>	afficher les deux dernières lignes

Table 1.27 – Liste de parties de scripts pour enchaîner (piping) les commandes

Un script de l'interpréteur d'une seule ligne peut reboucler sur de nombreux fichiers en utilisant `find(1)` et `xargs(1)` afin d'effectuer des tâches assez complexes. Consultez Section 10.1.5 et Section 9.4.9.

Lorsque l'utilisation de l'interpréteur de commandes en mode interactif devient trop compliquée, pensez à écrire un script en shell (consultez Section 12.1).

Chapitre 2

Gestion des paquets Debian

Note

Ce chapitre a été écrit en supposant que le nom de code de la dernière version stable est `Bullseye`.

[Debian](#) est une association de volontaires qui construit des distributions **cohérentes** de paquets binaires de logiciels libres pré-compilés et les distribue depuis son archive.

L'[archive Debian](#) est proposée depuis de [nombreux sites-miroirs distants](#), on peut y accéder par les méthodes HTTP et FTP. Elle est aussi disponible sous forme de [CD-ROM/DVD](#).

The current Debian package management system which can utilize all these resources is [Advanced Packaging Tool \(APT\)](#).

Le système de gestion des paquets Debian, **lorsqu'il est proprement utilisé**, permet à l'utilisateur d'installer sur le système des **ensembles cohérents de paquets binaires** à partir de l'archive. Il y a actuellement 63210 paquets disponibles pour l'architecture amd64.

Le système de gestion des paquets de Debian possède un riche historique et de nombreux choix de programmes d'interface pour l'utilisateur final et de méthodes de fond pour l'accès aux archives. Actuellement, nous recommandons ce qui suit :

- `apt(8)` for all interactive command line operations, including package installation, removal and dist-upgrades.
- `apt-get(8)` pour appeler le système de gestion de paquet Debian depuis des scripts. C'est également une option de rechange lorsqu'`apt` n'est pas disponible (souvent le cas avec d'anciens systèmes Debian).
- `aptitude(8)` pour une interface interactive en mode texte permettant de gérer les paquets installés et de faire des recherches parmi les paquets disponibles.

2.1 Prérequis pour la gestion des paquets Debian

2.1.1 Configuration de paquets

Voici quelques points-clés de la configuration des paquets sur un système Debian :

- la configuration manuelle effectuée par l'administrateur du système est respectée. En d'autres termes, le système de configuration des paquets effectue, pour des raisons de commodité, une configuration non intrusive ;
 - chaque paquet possède son propre script de configuration avec une interface utilisateur standardisée appelée `debconf(7)` qui permet de faciliter le processus initial d'installation du paquet ;
 - les développeurs Debian font de leur mieux pour que vos mises à jour se fassent de manière impeccable avec les scripts de configuration du paquet ;
 - l'administrateur du système peut utiliser toutes les fonctionnalités des paquets de logiciels. Cependant, celles qui présentent un risque de sécurité sont désactivées lors de l'installation par défaut ;
 - si vous activez vous-même un service qui présente certains risques de sécurité, vous êtes responsable du confinement du risque ;
 - Des configurations ésotériques peuvent être activées manuellement par l'administrateur du système. Cela peut créer une interférence avec les programmes génériques d'assistance à la configuration du système.
-

paquet	popcon	taille	description
dpkg	V:932, I:999	6874	low level package management system for Debian (file based)
apt	V:887, I:999	4337	APT front-end to manage packages with CLI: apt / apt-get / apt-cache
aptitude	V:72, I:398	4256	APT front-end to interactively manage packages with full screen console: aptitude (8)
tasksel	V:33, I:975	395	APT front-end to install selected tasks: tasksel (8)
unattended-upgrades	V:326, I:447	326	paquet d'amélioration d'APT permettant une installation automatique des mises à niveau de sécurité
gnome-software	V:112, I:203	6559	Software Center for GNOME (GUI APT front-end)
synaptic	V:39, I:308	7873	graphical package manager (GTK APT front-end)
apt-utils	V:379, I:997	1164	Programmes utilitaires d'APT : apt-extracttemplates (1), apt-ftpparchive (1) et apt-sortpkgs (1)
apt-listchanges	V:356, I:849	423	outil de notification des modifications d'un paquet
apt-listbugs	V:8, I:12	465	affiche la liste des bogues critiques avant chaque installation par APT
apt-file	V:17, I:79	90	utilitaire de recherche de paquet d'APT — interface en ligne de commandes
apt-rdepends	V:0, I:6	40	afficher de manière récursive la liste des dépendances du paquet

Table 2.1 – Liste des outils de gestion des paquets de Debian

2.1.2 Précautions de base



AVERTISSEMENT

Ne pas installer de paquets provenant d'un mélange aléatoire de suites. Cela va probablement casser la cohérence des paquets et demande une connaissance en profondeur de la gestion du système, comme l'[ABI](#), d'un compilateur, la version d'une [bibliothèque](#), les fonctionnalités d'un interpréteur, etc.

L'administrateur [débutant](#) d'un système Debian devrait conserver la version **stable** de Debian en appliquant les mises à jour de sécurité. Je veux dire qu'il vaut mieux éviter, par précaution, certaines des actions valables suivantes, jusqu'à ce que vous ayez très bien compris le système Debian. Voici quelques rappels :

- ne pas inclure **testing**, ni **unstable** dans « `/etc/apt/sources.list` » ;
- ne pas mélanger des archives standard de Debian avec d'autres archives telles qu'Ubuntu dans « `/etc/apt/sources.list` » ;
- ne pas créer le fichier « `/etc/apt/preferences` » ;
- ne pas modifier le comportement par défaut des outils de gestion des paquets au travers des fichiers de configuration sans en connaître toutes les conséquences ;
- ne pas installer de paquets quelconques à l'aide de « `dpkg -i paquet_quelconque` » ;
- ne jamais installer de paquets quelconques à l'aide de « `dpkg --force-all -i paquet_quelconque` » ;
- ne pas effacer ni modifier les fichiers se trouvant dans « `/var/lib/dpkg/` » ;
- ne pas écraser les fichiers systèmes en installant des logiciels directement depuis les sources.
 - Au besoin, les installer dans « `/usr/local` » ou « `/opt` ».

Les effets non compatibles avec le système de gestion des paquets Debian engendrés par les actions ci-dessus peuvent rendre votre système inutilisable.

L'administrateur système Debian sérieux, qui s'occupe de serveurs dont la mission est critique, devra prendre des précautions supplémentaires :

- ne pas installer de paquets, y compris les mises à jour de sécurité provenant de Debian sans les avoir testés soigneusement, avec votre configuration particulière, dans des conditions sûres.
 - Vous êtes finalement, en tant qu'administrateur système, responsable de votre système.
 - La longue histoire de stabilité du système Debian n'est pas, en elle-même, une garantie.

2.1.3 La vie avec d'éternelles mises à jour

**Attention**

Pour votre **serveur de production**, la suite `stable` avec les mises à jour de sécurité est recommandée. On peut dire la même chose des PC de bureau sur lesquels vous ne pouvez dépenser que des efforts limités d'administration.

Despite my warnings above, I know many readers of this document may wish to run the newer `testing` or `unstable` suites.

La [Lumière](#) provenant de ce qui suit sauvera une personne de l'éternelle lutte [karmique](#) de l'[enfer](#) des mises à jour et lui permettra d'atteindre le [nirvana](#) de Debian.

This list is targeted for the **self-administered** Desktop environment.

- Use the `testing` suite since it is practically the rolling release automatically managed by the Debian archive QA infrastructure such as the [Debian continuous integration](#), the [source only upload practices](#), and the [library transition tracking](#). The packages in the `testing` suite are updated frequently enough to offer all the latest features.
- Set the codename corresponding to the `testing` suite (currently "bookworm") in the `"/etc/apt/sources.list"`.
- Manually update this codename in the `"/etc/apt/sources.list"` to the new one only after assessing situation by yourself for about a month after the major suite release. The Debian user and developer mailing list are good sources of information for this, too.

The use of the `unstable` suite isn't recommended. The `unstable` suite is **good for debugging packages** as a developer but tends to expose you to unnecessary risks for the normal Desktop usage. Even though the `unstable` suite of the Debian system looks very stable for most of the times, there have been some package problems and a few of them were not so trivial to resolve.

Here are some basic precautionary measure ideas to ensure quick and easy recovery from bugs in Debian packages.

- faites un système avec un **double démarrage** en installant la suite `stable` du système Debian sur une autre partition ;
- tenez à disposition le CD d'installation pour un **démarrage de secours** ;
- pensez à installer `apt-listbugs` afin de vérifier les informations du [Système Debian de suivi des bogues \(BTS\)](#) avant de faire une mise à jour ;
- apprenez suffisamment l'infrastructure du système de paquets pour contourner le problème ;
- Install a corresponding sandboxed upstream binary package in case of trouble (see Section 7.6)
- Créez un environnement isolé (« chroot ») ou similaire et faites-y tourner à l'avance la dernière version du système (consultez Section 9.11).

**Attention**

If you can not do any one of these precautionary actions, you are probably not ready for the `testing` and `unstable` suites.

2.1.4 Bases concernant l'archive Debian

Jetez un œil sur l'[archive Debian](#) avec le point de vue d'un utilisateur du système.

ASTUCE

La charte officielle de l'archive Debian est définie dans la [Charte Debian, chapitre 2 - l'archive Debian](#).

Pour un accès HTTP typique, l'archive est indiquée de la manière suivante dans le fichier `"/etc/apt/sources.list"`, par exemple pour le système `stable` = `bullseye` actuel :

```
deb http://deb.debian.org/debian/ bullseye main contrib non-free
deb-src http://deb.debian.org/debian/ bullseye main contrib non-free

deb http://security.debian.org/debian-security bullseye-security main contrib
deb-src http://security.debian.org/debian-security bullseye-security main contrib
```

Ici, j'ai tendance à utiliser le nom de code « **bullseye** » plutôt que le nom de la suite « **stable** » afin d'éviter des surprises lorsque la version **stable** suivante sera diffusée.

La signification de « `/etc/apt/sources.list` » est décrite dans `sources.list(5)`, ses points-clés sont les suivants :

- la ligne « **deb** » définit les paquets binaires ;
- la ligne « **deb-src** » définit les paquets sources ;
- le premier paramètre est l'URL-racine de l'archive Debian ;
- le deuxième paramètre est le nom de la distribution : soit le nom de la suite, soit son nom de code ;
- le troisième paramètre et les suivants sont la liste des noms de sections d'archives valables dans l'archive Debian.

Les lignes **deb-src** peuvent être omises sans risque (ou mises en commentaire en mettant un « **#** » au début de la ligne) si ce n'est qu'**aptitude** ne pourra pas avoir accès aux métadonnées liées aux sources. Cela accélérera les mises à jour des métadonnées de l'archive. L'URL peut-être « `http://` », « `ftp://` », « `file://` », etc.

ASTUCE

Si « **sid** » est utilisé dans l'exemple ci-dessus plutôt que « **bullseye** », il n'est pas obligatoire d'avoir la ligne « `deb: http://security.debian.org/ ...` » pour les mises à jour de sécurité dans le fichier « `/etc/apt/sources.list` » parce qu'il n'y a pas de mises à jour de sécurité pour « **sid** » (**unstable**).

Voici la liste des URL des sites d'archives de Debian et les noms de suite ou les noms de code utilisés dans le fichier de configuration :

URL de l'archive	nom de la suite (nom de code)	but
http://deb.debian.org/debian/	stable (bullseye)	édition stable (Bullseye)
http://deb.debian.org/debian/	testing (bookworm)	édition testing (Bookworm)
http://deb.debian.org/debian/	unstable (sid)	édition unstable (Sid)
http://deb.debian.org/debian/	experimental	pré-version expérimental (optionnelle, uniquement pour les développeurs)
http://deb.debian.org/debian/	stable-proposed-updates	mises à jour pour la prochaine version de stable (optionnel)
http://security.debian.org/debian-security/	bullseye-security	mises à jour de sécurité pour la version stable (important)
http://security.debian.org/debian-security/	testing-security	mises à jour de sécurité pour la version testing (important)
http://deb.debian.org/debian/	bullseye-updates	mises à jour compatibles pour le filtrage de spam, les clients de messagerie instantanée, etc. pour Bullseye
http://deb.debian.org/debian/	bullseye-backports	paquets plus récents rétroportés pour Bullseye (optionnel)

Table 2.2 – Liste des sites d'archive de Debian

**Attention**

Seule la version **stable** pure avec les mises à jour de sécurité présente la meilleure stabilité. Faire tourner une version principalement **stable** mélangée à quelques paquets venant des versions **testing** ou **unstable** est plus risqué que d'utiliser une version **unstable** pure parce que des versions de bibliothèques peuvent ne pas correspondre, etc. Si vous avez réellement besoin de la dernière version de certains programmes sous la version **stable**, utilisez alors les paquets venant de [bullseye-updates](#) et des services [backports.debian.org](#) (consultez Section 2.7.4). Ces services doivent être utilisés avec des précautions supplémentaires.

**Attention**

De base, vous ne devriez avoir qu'une seule des suites **stable**, **testing** ou **unstable** sur la ligne « **deb** ». Si vous avez une combinaison des suites **stable**, **testing** et **unstable** sur la ligne « **deb** », les programmes APT vont être ralentis bien que seule la dernière archive soit utilisée. Des mentions multiples ont un intérêt lorsqu'on utilise le fichier « **/etc/apt/preferences** » avec des objectifs clairs (consultez Section 2.7.3).

ASTUCE

Pour les systèmes Debian ayant les versions **stable** et **testing**, c'est une bonne idée d'inclure les lignes ayant « **http://security.debian.org/** » dans le fichier « **/etc/apt/sources.list** » afin d'activer les mises à jour de sécurité comme dans l'exemple ci-dessus.

Note

Les bogues de sécurité de l'archive **stable** sont corrigés par l'équipe de sécurité de Debian. Cette activité a été assez rigoureuse et fiable. Ceux de l'archive **testing** peuvent être corrigés par l'équipe de sécurité de Debian. Pour [diverses raisons](#), cette activité n'est pas aussi rigoureuse que pour **stable** et vous pouvez avoir à attendre la migration de paquets corrigés de **unstable**. Ceux de l'archive **unstable** sont corrigés par les responsables individuels. Les paquets de **unstable** maintenus de manière active sont habituellement maintenus dans un assez bon état par mise à niveau avec les dernières corrections de sécurité des développeurs amonts. Consultez [FAQ de sécurité de Debian](#) concernant la manière dont Debian gère les bogues de sécurité.

section	nombre de paquets	critères de composant du paquet
main	62089	conforme à DFSG sans dépendance vers non-free
contrib	351	conforme à DFSG mais avec des dépendances vers non-free
non-free	770	non conforme à DFSG

Table 2.3 – Liste des sections de l'archive de Debian

Ici, le nombre de paquets est celui de l'architecture **amd64**. La section **main** fournit le système Debian (consultez Section 2.1.5).

La meilleure manière d'étudier l'organisation de l'archive Debian est de pointer votre navigateur vers chacune des URL des archives en y ajoutant **dist**s ou **pool**.

On se réfère à la distribution de deux manières, la version ou le [nom de code](#). Le mot « distribution » est aussi utilisé comme synonyme de version dans de nombreuses documentations. La relation entre la version et le nom de code peut être résumée comme suit :

L'histoire des noms de code a été décrite dans la [FAQ Debian : 6.2.1 Quels noms de code ont déjà été utilisés ?](#)

Dans la terminologie la plus stricte de l'archive Debian, le mot « section » est spécifiquement utilisé pour la catégorisation des paquets par zone d'application. (Cependant l'expression « section principale » peut parfois être utilisée pour décrire la section de l'archive Debian qui fournit la zone « **main** »).

calendrier	version = stable	version = testing	version = unstable
après la diffusion de bullseye	nom de code = bullseye	nom de code = bookworm	nom de code = sid
après la diffusion de bookworm	nom de code = bookworm	nom de code = trixie	nom de code = sid

Table 2.4 – Relation entre version et nom de code

Chaque fois qu'un nouveau chargement est fait par un développeur Debian (DD) vers l'archive `unstable` (par l'intermédiaire du traitement d'[incoming](#)), le DD doit s'assurer que les paquets envoyés sont compatibles avec le dernier ensemble de paquets de l'archive `unstable`.

Si le DD casse intentionnellement cette compatibilité en raison de la mise à jour d'une bibliothèque importante, etc., il y a habituellement une annonce sur la [liste de diffusion debian-devel](#), etc.

Avant qu'un ensemble de paquets ne soit déplacé par le script de maintenance de l'archive Debian depuis l'archive `unstable` vers l'archive `testing`, le script de maintenance de l'archive ne se contente pas vérifier sa maturité (environ 10 jours) et l'état des rapports de bogues pour ces paquets mais essaie aussi de s'assurer qu'ils sont compatibles avec le dernier ensemble des paquets de l'archive `testing`. Ce processus rend l'archive `testing` très actuelle et utilisable.

Par le processus de gel progressif de l'archive dirigé par l'équipe de diffusion (« release team »), l'archive `testing` est mûrie afin de la rendre entièrement cohérente et sans bogue avec quelques interventions manuelles. Ensuite, la nouvelle version `stable` est créée en assignant le nom de code de l'ancienne archive `testing` à la nouvelle archive `stable` et en créant un nouveau nom de code pour la nouvelle archive `testing`. Le contenu initial de la nouvelle archive `testing` est exactement le même que celui de l'archive `stable` qui vient d'être diffusée.

Les archives `unstable` et `testing` peuvent toutes les deux souffrir temporairement de problèmes en raison de divers facteurs :

- chargement vers l'archive cassé (la plupart du temps, cela concerne `unstable`) ;
- délai pour accepter un nouveau paquet dans l'archive (la plupart du temps, cela concerne `unstable`) ;
- problème de temps de synchronisation de l'archive (à la fois pour `testing` et `unstable`) ;
- intervention manuelle sur l'archive comme la suppression d'un paquet (davantage pour `testing`), etc.

Si vous décidez donc d'utiliser ces archives, vous devriez être capable de corriger ou de contourner ces types de problèmes.

Attention



Pendant les quelques mois qui suivent la diffusion d'une nouvelle version `stable`, la plupart des utilisateurs de machines de bureau devraient utiliser l'archive `stable` avec ses mises à jour de sécurité même s'ils utilisent habituellement les archives `unstable` ou `testing`. Pendant cette période de transition, les archives `unstable` et `testing` ne sont pas bonnes pour la plupart des gens. Votre système sera difficile à conserver dans un bon état de fonctionnement avec l'archive `unstable` car elle souffre de pics d'importantes mises à jour de paquets fondamentaux. L'archive `testing` n'est pas utile non plus car elle a sensiblement le même contenu que l'archive `stable` sans la prise en compte de la sécurité ([Debian testing-security-announce 2008-12](#)). Après environ un mois, l'archive `unstable` peut être utilisée avec précautions.

ASTUCE

Lors du suivi de l'archive `testing`, un problème causé par la suppression d'un paquet est habituellement contournée en installant le paquet correspondant de l'archive `unstable` qui est envoyé pour la correction du bogue.

Consultez la [Charte Debian](#) pour la définition des archives.

- « [Sections](#) »
- « [Priorités](#) »
- « [Système de base](#) »
- « [Paquets essentiels](#) »

2.1.5 Debian est totalement libre

Debian est totalement libre pour les raisons suivantes :

- Debian n’installe que des logiciels libres par défaut pour respecter les libertés des utilisateurs ;
- Debian ne fournit que des logiciels libres dans `main` ;
- Debian recommande de n’utiliser que des logiciels libres de `main` ;
- Aucun paquet de `main` ne dépend ou ne recommande de paquets de `non-free` ni de `contrib`.

Certaines personnes se demandent si les deux faits suivants sont contradictoires ou non.

- « Debian demeurera totalement libre. » (Premier point du [contrat social Debian](#)).
- Debian héberge des paquets `non-free` (non libres) et `contrib` (contributions).

Ce n’est pas contradictoire pour les raisons suivantes.

- Le système Debian est totalement libre et ses paquets sont hébergés par les serveurs Debian dans la section `main` de l’archive.
- Des paquets hors du système Debian sont hébergés par les serveurs Debian dans les sections `non-free` et `contrib` de l’archive.

C’est précisément expliqué dans les quatrième et cinquième points du [contrat social Debian](#) :

- Nos priorités sont nos utilisateurs et les logiciels libres.
 - Les besoins de nos utilisateurs et de la communauté des logiciels libres nous guideront. Nous placerons leurs intérêts en tête de nos priorités. Nous répondrons aux besoins de nos utilisateurs dans de nombreux types d’environnements informatiques différents. Nous ne nous opposerons pas aux travaux non libres prévus pour fonctionner sur les systèmes Debian. Nous permettons, sans réclamer rétribution, que d’autres créent des distributions contenant conjointement des logiciels Debian et d’autres travaux. Pour servir ces objectifs, nous fournirons un système intégrant des composants de grande qualité sans restrictions légales incompatibles avec ces modes d’utilisation.
- Travaux non conformes à nos standards sur les logiciels libres.
 - Nous reconnaissons que certains de nos utilisateurs demandent à pouvoir utiliser des travaux qui ne sont pas conformes aux principes du logiciel libre selon Debian. Les paquets correspondant prennent place dans des sections nommées « `contrib` » (« contributions ») et « `non-free` » (« non libre »). Les paquets de ces sections ne font pas partie du système Debian, bien qu’ils aient été configurés afin d’être utilisés avec lui. Nous encourageons les fabricants de CD à lire les licences de ces paquets afin de déterminer s’ils peuvent les distribuer. Ainsi, bien que les travaux non libres ne fassent pas partie de Debian, nous prenons en compte leur utilisation et fournissons donc l’infrastructure nécessaire (à l’image de notre système de suivi des bogues et de nos listes de diffusion).

Les utilisateurs doivent être conscients des risques lors de l’utilisation de paquets des sections `non-free` et `contrib` de l’archive :

- l’absence de liberté pour de tels paquets de logiciel ;
- l’absence de suivi de la part de Debian pour de tels paquets de logiciel (Debian ne peut pas suivre correctement un logiciel sans avoir accès à son code source) ;
- la contamination de votre système Debian totalement libre.

Les principes du logiciel libre selon Debian sont les normes du logiciel libre pour Debian. Debian interprète « logiciel » de la façon la plus large possible, y compris la documentation, les microprogrammes, les logos et données artistiques des paquets. Cela rend les normes du logiciel libre de Debian très strictes.

Les paquets typiques de `non-free` et `contrib` contiennent des paquets librement distribuables des types suivants :

- Les paquets de documentation sous [licence de documentation libre GNU](#) avec des sections invariables comme celles de GCC et Make (la plupart sont dans la section `non-free/doc`).
- Les paquets de microprogramme contenant des données binaires sans source comme celles de Section 9.10.5 dans `non-free` (la plupart sont dans la section `non-free/kernel`).
- Les paquets de jeu ou de police avec des restrictions sur l’utilisation commerciale ou la modification de contenu.

Veuillez remarquer que le nombre de paquets de `non-free` et `contrib` est inférieur à 2 % de ceux dans `main`. Activer l’accès aux sections `non-free` et `contrib` ne cache pas la provenance des paquets. L’utilisation interactive d’`aptitude(8)` fournit une visibilité totale et un contrôle complet des paquets installés et de leur section pour garder le système aussi libre que voulu.

2.1.6 Dépendances des paquets

Le système Debian offre un ensemble cohérent de paquets binaires par l'intermédiaire de son mécanisme de déclaration de dépendances binaires versionnées dans les champs du fichier « control ». En voici une définition un peu simplifiée :

- « Depends »
 - Cela déclare une dépendance absolue du paquet et tous les paquets listés dans ce champ doivent être installés en même temps ou à l'avance.
- « Pre-Depends »
 - Comme pour « Depends » excepté que cela demande une installation complète et à l'avance des paquets cités.
- « Recommends »
 - Cela déclare une dépendance forte mais non absolue. La plupart des utilisateurs n'installeront pas le paquet si tous les paquets cités dans ce champ ne sont pas installés.
- « Suggests »
 - Cela déclare une dépendance lâche. De nombreux utilisateurs de ce paquet pourront tirer profit de l'installation des paquets cités dans ce champs mais auront cependant des fonctionnalités acceptables sans eux.
- « Enhances »
 - Cela déclare une dépendance lâche comme « Suggests » mais fonctionne à l'inverse.
- « Casse »
 - Cela déclare une incompatibilité de paquet avec habituellement une indication de version. La solution est en général de mettre à jour tous les paquets indiqués dans ce champ.
- « Conflicts »
 - Cela déclare une incompatibilité absolue. Tous les paquets cités dans ce champs doivent être supprimés pour installer ce paquet.
- « Replaces »
 - C'est déclaré lorsque les fichiers installés par ce paquet remplacent des fichiers des paquets cités.
- « Provides »
 - C'est déclaré lorsque ce paquet fournit tous les fichiers et les fonctionnalités des paquets cités.

Note

Remarquez que définir simultanément « Provides », « Conflicts » et « Replaces » pour un paquet virtuel est une configuration saine. Cela permet de s'assurer qu'un seul paquet réel fournissant ce paquet virtuel puisse être installé à un moment donné.

La définition officielle, y compris les dépendances de sources, se trouve dans la [Charte Debian ; Chapitre 7 - Déclaration des dépendances entre paquets](#).

2.1.7 Flux des événements dans la gestion d'un paquet

Voici un résumé du flux simplifié des événements de la gestion d'un paquet par APT.

- **Mettre à jour (« update »)** («apt update», « aptitude update » ou « apt-get update »):
 1. Rechercher les métadonnées d'une archive depuis l'archive distante
 2. Reconstruire et mettre à jour les métadonnées locales pour qu'elles puissent être utilisées par APT
 - **Mettre à niveau (« upgrade »)** («apt upgrade» et «apt full-upgrade», ou « aptitude safe-upgrade » et « aptitude full-upgrade » ou « apt-get upgrade » et « apt-get dist-upgrade »):
 1. Choisir la version candidate, qui est habituellement la dernière version disponible, pour tous les paquets installés (consultez Section [2.7.3](#) pour les exceptions)
 2. Effectuer la résolution des dépendances du paquet
-

3. Rechercher le paquet binaire sélectionné depuis l'archive distante si la version candidate est différente de la version installée
4. Dépaqueter les paquets binaires ayant été téléchargés
5. Lancer le script **preinst**
6. Installer les fichiers binaires
7. Lancer le script **postinst**

— **Installer** («`apt install ...`», «`aptitude install ...` » ou «`apt-get install ...` »):

1. Choisir les paquets indiqués sur la ligne de commandes
2. Effectuer la résolution des dépendances du paquet
3. Récupérer les paquets binaires sélectionnés depuis l'archive distante
4. Dépaqueter les paquets binaires ayant été téléchargés
5. Lancer le script **preinst**
6. Installer les fichiers binaires
7. Lancer le script **postinst**

— **Supprimer** («`apt remove ...`», «`aptitude remove ...` » ou «`apt-get remove ...` »):

1. Choisir les paquets indiqués sur la ligne de commandes
2. Effectuer la résolution des dépendances du paquet
3. Lancer le script **prerm**
4. Supprimer les fichiers installés **à l'exception** des fichiers de configuration
5. Lancer le script **postrm**

— **Purger** («`apt purge ...`», «`aptitude purge ...` » ou «`apt-get purge ...` »):

1. Choisir les paquets indiqués sur la ligne de commandes
2. Effectuer la résolution des dépendances du paquet
3. Lancer le script **prerm**
4. Supprimer les fichiers installés **y compris** leurs fichiers de configuration
5. Lancer le script **postrm**

J'ai ici intentionnellement sauté des détails techniques dans le souci d'avoir une vue d'ensemble.

2.1.8 Première réponse aux problèmes de gestion de paquets

Vous devriez lire l'excellente documentation officielle. Le premier document à lire est « `/usr/share/doc/nom_paquet/README.` » qui est spécifique à Debian. Les autres documents dans « `/usr/share/doc/nom_paquet/` » devraient aussi être consultés. Si vous avez configuré l'interpréteur de commande comme dans Section 1.4.2, entrez ce qui suit :

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

Vous aurez besoin d'installer le paquet de documentation correspondant au paquet dont le nom possède le suffixe « `-doc` » pour des informations détaillées.

Si vous rencontrez des problèmes avec un paquet particulier, faites d'abord une recherche sur le site du [système de suivi des bogues Debian \(BTS\)](#).

Rechercher sur [Google](#) avec des mots de recherche comprenant « `site:debian.org` », « `site:wiki.debian.org` », « `site:lists.debian.org` », etc.

Pour déposer un signalement de bogue, veuillez utiliser la commande `reportbug(1)`.

site web	commande
Page d'accueil du système de suivi des bogues Debian (BTS)	<code>sensible-browser « http://bugs.debian.org/ »</code>
Signalement de bogue d'un nom de paquet connu	<code>sensible-browser « http://bugs.debian.org/nom_paquet »</code>
Rapport de bogue concernant un numéro de bogue connu	<code>sensible-browser « http://bugs.debian.org/bug_number »</code>

Table 2.5 – Liste de sites web clés pour résoudre les problèmes avec un paquet particulier

2.2 Opérations de base de la gestion des paquets

Sur le système Debian, les opérations de gestion des paquets basées sur les dépôts peuvent être réalisées à l'aide de nombreux outils de gestion de paquets basés sur APT et disponibles dans le système Debian. Nous décrivons ici les outils de base de gestion des paquets : `apt`, `apt-get`/`apt-cache` et `aptitude`.

Pour les opérations de gestion des paquets qui concernent l'installation des paquets ou les mises à jour des métadonnées des paquets, vous aurez besoin des privilèges de l'administrateur.

2.2.1 `apt` comparé à `apt-get` / `apt-cache` comparé à `aptitude`

Bien qu'`aptitude` soit un très bon outil interactif et que l'auteur l'utilise, voici quelques avertissements que vous devriez connaître :

- La commande `aptitude` n'est pas recommandée pour une mise à niveau du système entre versions sur le système Debian **stable** après la sortie d'une nouvelle version.
 - L'utilisation de "`apt full-upgrade`" ou de "`apt-get dist-upgrade`" est recommandée pour cela. Voir [Bug #411280](#).
- La commande `aptitude` suggère parfois la suppression massive de paquets lors de la mise à niveau du système sur des systèmes Debian en **testing** ou **unstable**.
 - Cette situation a effrayé de nombreux administrateurs système. Pas de panique.
 - Il semblerait que cela soit principalement causé par un biais de version parmi des paquets dépendants de, ou recommandés par, un méta-paquet tel que `gnome-core`.
 - Cela peut être résolu en sélectionnant « Annuler les actions en attente » dans le menu de commande d'`aptitude`, en quittant `aptitude` et en utilisant la commande « `apt full-upgrade` ».

Les commandes `apt-get` et `apt-cache` sont les outils les plus **basiques** de gestion des paquets basés sur APT.

- `apt-get` et `apt-cache` n'offre qu'une interface utilisateur en ligne de commandes.
- `apt-get` est le mieux adapté pour les **mises à jour majeures du système** entre les versions, etc.
- `apt-get` offre un système de résolution des dépendances entre paquets **robuste**.
- `apt-get` nécessite moins de ressources matérielles. Il consomme moins de mémoire et fonctionne plus rapidement.
- `apt-cache` offre une recherche basée sur des expressions rationnelles **standard** sur les noms et les descriptions des paquets.
- `apt-get` et `apt-cache` peuvent gérer des versions multiples des paquets en utilisant `/etc/apt/preferences` mais est assez lourd.

The `apt` command is a high-level commandline interface for package management. It is basically a wrapper of `apt-get`, `apt-cache` and similar commands, originally intended as an end-user interface and enables some options better suited for interactive usage by default.

- `apt` provides a friendly progress bar when installing packages using `apt install`.
- `apt` will **remove** cached `.deb` packages by default after successful installation of downloaded packages.

ASTUCE

Users are recommended to use the new `apt(8)` command for **interactive** usage and use the `apt-get(8)` and `apt-cache(8)` commands in the shell script.

La commande `aptitude` est l'outil de gestion des paquets basé sur APT le plus **flexible**.

- `aptitude` offre une interface utilisateur interactive en plein écran en mode texte.
- `aptitude` offre aussi une interface utilisateur en ligne de commandes.
- `aptitude` est le mieux adapté pour la **gestion interactive journalière des paquets** comme, par exemple, la vérification des paquets installés et la recherche de paquets disponibles.
- `aptitude` nécessite plus de ressources matérielles. Il consomme plus de mémoire et fonctionne moins rapidement.
- `aptitude` offre une recherche **avancée** basée sur des expressions rationnelles pour la recherche sur toutes les métadonnées des paquets.
- `aptitude` peut gérer des versions multiples des paquets sans utiliser `/etc/apt/preferences` et est assez intuitif.

2.2.2 Opérations de base de gestion des paquets en ligne de commandes

Voici les opérations de base de gestion des paquets en ligne de commandes en utilisant `apt(8)`, `aptitude(8)`, `apt-get(8)` et `apt-cache(8)`.

`apt / apt-get` and `aptitude` can be mixed without major troubles.

« `aptitude why expression-rationnelle` » peut afficher plus d'informations par « `aptitude -v why expression` ». On peut obtenir des informations similaires par « `apt rdepends paquet` » ou « `apt-cache rdepends paquet` ».

Lorsque la commande `aptitude` est lancée en mode ligne de commande, et rencontre des problèmes tels que des conflits de paquets, vous pouvez passez en mode plein écran en pressant ensuite la touche « e » à l'invite de commande

Note

Bien que la commande `aptitude` soit disponible avec de riches fonctionnalités comme son solveur de paquets avancé, cette complexité a causé (et peut encore causer) certaines régressions comme le [bogue #411123](#), le [bogue #514930](#) et le [bogue #570377](#). En cas de doute, veuillez utiliser les commandes `apt`, `apt-get` et `apt-cache` plutôt que la commande `aptitude`.

Vous pouvez indiquer les options de commande juste après « `aptitude` ».

Consultez `aptitude(8)` et le « manuel de l'utilisateur d'`aptitude` » à « `/usr/share/doc/aptitude/README` » pour en apprendre davantage.

2.2.3 Utilisation interactive d'`aptitude`

Pour une gestion interactive des paquets, lancez `aptitude` en mode interactif depuis l'invite de l'interpréteur de commandes à la console comme suit :

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

Cela va mettre à jour la copie locale des informations de l'archive et afficher la liste des paquets en plein écran avec un menu. On trouvera la configuration d'`aptitude` dans « `~/.aptitude/config` ».

ASTUCE

Si vous désirez utiliser la configuration de l'administrateur (root) plutôt que celle de l'utilisateur, utilisez la commande « `sudo -H aptitude ...` » en remplacement de « `sudo aptitude ...` » dans l'expression précédente.

syntaxe d'apt	syntaxe d'aptitude	syntaxe d'apt-get et apt-cache	description
apt update	aptitude update	apt-get update	mettre à jour les métadonnées de l'archive du paquet
apt install toto	aptitude install toto	apt-get install toto	installer la version candidate du paquet « toto » ainsi que ses dépendances
apt upgrade	aptitude safe-upgrade	apt-get upgrade	installer les versions candidates des paquets installés sans supprimer aucun autre paquet
apt full-upgrade	aptitude full-upgrade	apt-get dist-upgrade	installer les versions candidates des paquets installés en supprimant certains paquets si nécessaire
apt remove toto	aptitude remove toto	apt-get remove toto	supprimer le paquet « toto » en laissant ses fichiers de configuration
apt autoremove	N/A	apt-get autoremove	supprimer les paquets installés automatiquement lorsqu'ils ne sont plus nécessaires
apt purge toto	aptitude purge toto	apt-get purge toto	purger le paquet « toto » ainsi que ses fichiers de configuration
apt clean	aptitude clean	apt-get clean	nettoyer complètement le dépôt local des fichiers de paquets récupérés
apt autoclean	aptitude autoclean	apt-get autoclean	nettoyer le dépôt local des fichiers des paquets périmés
apt show toto	aptitude show toto	apt-cache show toto	afficher des informations détaillées concernant le paquet « toto »
apt search <expression rationnelle>	aptitude search <expression rationnelle>	apt-cache search <expression rationnelle>	rechercher les paquets qui correspondent à l'« expression rationnelle »
N/A	aptitude why <expression rationnelle>	N/A	expliquer les raisons qui font que les paquets correspondant à l'« expression rationnelle » devront être installés
N/A	aptitude why-not <expression rationnelle>	N/A	expliquer les raisons pour lesquels les paquets qui correspondent à l'« expression rationnelle » ne peuvent pas être installés
N/A	aptitude search '~i!~M'	apt-mark showmanual	lister les paquets installés manuellement

Table 2.6 – Opérations de base de gestion des paquets avec la ligne de commandes en utilisant `apt(8)`, `aptitude(8)`, `apt-get(8)` et `apt-cache(8)`

option de la commande	description
-s	simuler le résultat de la commande
-d	télécharger seulement les paquets sans les installer ni les mettre à jour
-D	afficher une courte explication avant les installations ou les suppressions automatiques

Table 2.7 – Options importantes de la commande `aptitude(8)`

ASTUCE

Aptitude définit automatiquement les **actions en attente** lorsqu'il est lancé de manière interactive. Si elles ne vous conviennent pas, vous pouvez le réinitialiser depuis le menu : « Action » → « Annuler les opérations en attente ».

2.2.4 Raccourcis clavier d'aptitude

Les raccourcis clavier principaux pour parcourir l'état des paquets et pour définir les « actions prévues » sur ces paquets dans le mode plein écran sont les suivants :

touche	affectation
F10 ou Ctrl-t	menu
?	afficher l' aide pour les raccourcis clavier (liste plus complète)
F10 → Aide → Manuel de l'utilisateur	afficher le Manuel de l'utilisateur
u	mettre à jour les informations de l'archive des paquets
+	marquer le paquet pour mise à niveau ou installation
-	marquer le paquet pour suppression (conserver ses fichiers de configuration)
—	marquer le paquet pour être purgé (supprimer ses fichiers de configuration)
=	mettre le paquet dans l'état « conservé »
U	marquer tous les paquets susceptibles de mise à niveau (fonctionne comme full-upgrade)
g	lancer le téléchargement et l' installation des paquets sélectionnés
q	quitter l'écran actuel et enregistrer les modifications
x	quitter l'écran actuel en abandonnant les modifications
Entrée	afficher les informations concernant un paquet
C	afficher le journal des modifications (« changelog ») d'un paquet
l	modifier les limites pour les paquets affichés
/	rechercher la première correspondance
\	répéter la dernière recherche

Table 2.8 – Liste des raccourcis clavier d'aptitude

L'indication du nom de fichier sur la ligne de commandes et à l'invite du menu après avoir pressé « l » et « // » prend l'expression rationnelle d'aptitude telle que décrite ci-dessous. Une expression rationnelle d'aptitude peut correspondre explicitement à un nom de paquet en utilisant une chaîne de caractères commençant par « ~n » et suivie du nom de paquet.

ASTUCE

Vous devrez presser « U » pour obtenir la mise à niveau de tous les paquets installés vers la **version candidate** de l'interface visuelle. Sinon, seuls les paquets sélectionnés et certains paquets ayant des dépendances versionnées sur ces paquets seront mis à niveau vers la **version candidate**.

2.2.5 Vues des paquets sous aptitude

Dans le mode interactif en plein écran d'aptitude(8), les paquets de la liste des paquets sont affichés comme dans l'exemple suivant.

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

Cette ligne signifie, en partant de la gauche :

— Indicateur d'« état actuel » (la première lettre)

- Indicateur d'« action prévue » (la seconde lettre)
- Indicateur « automatique » (la troisième lettre)
- Nom du paquet
- Modification de l'utilisation du disque attribuée à l'« action prévue »
- Version actuelle du paquet
- Version candidate du paquet

ASTUCE

La liste complète des indicateurs est donnée en bas de l'écran d'**Aide** affiché en pressant « ? ».

La **version candidate** est choisie en fonction des préférences locales actuelles (consultez `apt_preferences(5)` et Section 2.7.3).

Plusieurs types de vues de paquets sont disponibles depuis le menu « Vues ».

vue	description de la vue
Vue des paquets	consultez Tableau 2.10 (défaut)
Recommandations d'audit	liste des paquets qui sont recommandés par certains paquets installés mais qui ne sont pas encore installés sur le système
Liste de paquet « à plat »	liste des paquets sans regroupement par catégories (pour l'utilisation avec des expressions rationnelles)
Navigateur de Debtags	liste des paquets classés selon leur entrée debtags
Source Package View	list packages grouped by source packages

Table 2.9 – Liste des vues d'aptitude

Note

Merci de nous aider à [améliorer le marquage des paquets avec debtags](#) !

La « Vue des paquets » standard classe les paquets un peu comme le fait `dselect` avec quelques fonctionnalités supplémentaires.

catégorie	description de la vue
Paquets susceptibles de mise à jour	liste des paquets organisée sous la forme <code>section → zone → paquet</code>
Nouveaux paquets	, ,
Paquets installés	, ,
Paquets non installés	, ,
Paquets obsolètes ou créés localement	, ,
Paquets virtuels	liste des paquets ayant la même fonction
Tâches	liste des paquets ayant les différentes fonctions généralement nécessaires à une tâche

Table 2.10 – Classement par catégories des vues de paquets standard

ASTUCE

La vue des tâches peut être utilisée pour choisir les paquets nécessaires à votre tâche.

2.2.6 Options de la méthode de recherche avec aptitude

Aptitude vous offre différentes options pour rechercher des paquets en utilisant sa formule d'expressions rationnelles.

- Ligne de commande du shell :
 - « `aptitude search 'expression_rationnelle_aptitude'` » afin d'afficher l'état d'installation, le nom du paquet et une courte description des paquets correspondants
 - « `aptitude show 'nom_paquet'` » pour afficher la description détaillée du paquet
- Mode interactif plein écran :
 - « `l` » pour limiter la vue des paquets à ceux qui correspondent
 - « `/` » pour rechercher un paquet correspondant
 - « `\` » pour rechercher en arrière un paquet correspondant
 - « `n` » pour rechercher le suivant
 - « `N` » pour rechercher le suivant (en arrière)

ASTUCE

La chaîne du *nom_paquet* est traitée comme la correspondance exacte de chaîne pour le nom de paquet à moins qu'il ne soit lancé explicitement avec « `~` » pour être la formule d'expression rationnelle.

2.2.7 Les formules d'expressions rationnelles d'aptitude

La formule des expressions rationnelles d'aptitude est étendue **ERE** de manière similaire à `mutt` (consultez Section 1.6.2) et la signification des extensions de règles de correspondance spécifiques à `aptitude` est la suivante :

- La partie expression rationnelle est la même **ERE** que celle utilisée dans les outils UNIX typiques en utilisant « `^` », « `.` », « `*` », « `$` » etc. comme dans `egrep(1)`, `awk(1)` et `perl(1)`.
- La dépendance *type* est comprise dans la liste (`depends`, `predepends`, `recommends`, `suggests`, `conflicts`, `replaces`, `provides`) et spécifie les relations du paquet avec d'autres paquets.
- Le *type* de relation par défaut est « `depends` ».

ASTUCE

Lorsqu'un <motif d'expression rationnelle> (« `regex_pattern` ») est une chaîne de caractères vide, placez « `~T` » directement après la commande.

Voici quelques raccourcis.

- « `~Pterm` » == « `~Dprovides:term` »
- « `~Cterm` » == « `~Dconflicts:term` »
- « `...~W term` » == « `(...|term)` »

Les utilisateurs familiers avec `mutt` comprendront rapidement car `mutt` a été la source d'inspiration pour la syntaxe des expressions. Consultez « `SEARCHING, LIMITING, AND EXPRESSIONS` » dans le manuel de l'utilisateur (« `/usr/share/doc/aptitude` »).

Note

Avec la version `lenny` d'`aptitude`(8), la nouvelle **forme longue** de la syntaxe comme « `?broken` » peut être utilisée pour la correspondance des expressions rationnelles en remplacement de l'ancien équivalent « `~b` » en **forme courte**. Le caractère d'espacement « » est maintenant considéré comme l'un des caractères de terminaison d'une expression rationnelle en plus du caractère tilde « `~` ». Consultez la syntaxe de la nouvelle **forme longue** dans le « Manuel de l'utilisateur ».

description des règles étendues de correspondance	formules d'expressions rationnelles
correspond au nom du paquet	<code>~nexpression_rationnelle_nom</code>
correspond à la description	<code>~dexpression_rationnelle_description</code>
correspond au nom de la tâche	<code>~texpression_rationnelle_tâche</code>
correspond à l'étiquette debtag	<code>~Gexpression_rationnelle_debtag</code>
correspond au responsable du paquet	<code>~mexpression_rationnelle_responsable</code>
correspond à la section du paquet	<code>~sexpression_rationnelle_section</code>
correspond à la version du paquet	<code>~Vexpression_rationnelle_version</code>
correspond à l'archive	<code>~A{bullseye,bookworm,sid}</code>
correspond à l'origine	<code>~O{debian,...}</code>
correspond à la priorité	<code>~p{extra,important,optional,required,standard}</code>
correspond aux paquets essentiels	<code>~E</code>
correspond aux paquets virtuels	<code>~V</code>
correspond aux nouveaux paquets	<code>~N</code>
correspond aux actions en attente	<code>~a{install,upgrade,downgrade,remove,purge,hold,keep}</code>
correspond aux paquets installés	<code>~i</code>
correspond aux paquets installés ayant la marque A (paquets installés automatiquement)	<code>~M</code>
correspond aux paquets installés n'ayant pas la marque A (paquets sélectionnés par l'administrateur)	<code>~i!~M</code>
correspond aux paquets installés et pouvant être mis à jour	<code>~U</code>
correspond aux paquets supprimés mais non purgés	<code>~C</code>
correspond aux paquets supprimés, purgés ou pouvant être supprimés	<code>~g</code>
correspond aux paquets ayant une dépendance cassée	<code>~b</code>
correspond aux paquets ayant une dépendance cassée de <i>type</i>	<code>~Btype</code>
correspond aux paquets filtrés par <i>motif</i> ayant une dépendance de <i>type</i>	<code>~D[type:]motif</code>
correspond aux paquets filtrés par <i>motif</i> ayant une dépendance cassée de <i>type</i>	<code>~DB[type:]motif</code>
correspond aux paquets vers lesquels le paquet filtré par <i>motif</i> déclare une dépendance de <i>type</i>	<code>~R[type:]motif</code>
correspond aux paquets vers lesquels le paquet filtré par <i>motif</i> a une dépendance cassée de <i>type</i>	<code>~RB[type:]motif</code>
correspond aux paquets desquels dépendent d'autres paquets installés	<code>~R~i</code>
correspond aux paquets desquels ne dépend aucun autre paquet	<code>!~R~i</code>
correspond aux paquets vers lesquels d'autres paquets installés dépendent ou qu'ils recommandent	<code>~R~i ~Rrecommends:~i</code>
correspond au paquet <i>motif</i> dont la version est filtrée	<code>~S filter motif</code>
correspond à tous les paquets (vrai)	<code>~T</code>
ne correspond à aucun paquet (faux)	<code>~F</code>

Table 2.11 – Liste des formules d'expressions rationnelles d'aptitude

2.2.8 Résolution des dépendances par aptitude

La sélection d'un paquet dans `aptitude` récupère non seulement les paquets définis dans son champ « `Depends :` » mais aussi ceux définis dans le champ « `Recommends :` » si la configuration a été faite dans ce sens dans le menu « `F10 → Options → Préférences → Gestion des dépendances` ». Ces paquets installés automatiquement seront supprimés automatiquement s'ils ne sont plus nécessaires sous `aptitude`.

Le drapeau contrôlant le comportement "auto install" de la commande `aptitude` peut aussi être manipulé en utilisant la commande `apt-mark(8)` du paquet `apt`.

2.2.9 Journaux d'activité des paquets

Vous pouvez vérifier l'activité de l'historique des paquets dans les fichiers journaux.

fichier	contenu
<code>/var/log/dpkg.log</code>	Enregistrement des actions au niveau de <code>dpkg</code> pour l'activité de tous les paquets
<code>/var/log/apt/term.log</code>	Journal de l'activité générique d'APT
<code>/var/log/aptitude</code>	Journal des actions de la commande <code>aptitude</code>

Table 2.12 – Fichiers journaux de l'activité des paquets

En réalité, il n'est pas aussi facile de comprendre la signification de ces journaux. Consultez Section 9.3.9 pour une façon de faire plus simple.

2.3 Exemples d'opérations avec aptitude

Voici quelques exemples d'opérations d'`aptitude(8)`.

2.3.1 Afficher les paquets dont les noms correspondent à une expression rationnelle

La commande suivante affiche les paquets dont les noms correspondent à une expression rationnelle.

```
$ 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
```

Il vous est assez facile de trouver le nom exact d'un paquet.

2.3.2 Parcours en correspondance avec une expression rationnelle

L'expression rationnelle « `~dipv6` » entrée dans la vue « Nouvelle liste des paquets » (« New Flat Package List » depuis l'invite « `l` » limite la vue aux paquets dont la description correspond à cette expression rationnelle et vous permet de parcourir les informations de manière interactive.

2.3.3 Purger pour de bon les paquets supprimés

Vous pouvez supprimer tous les fichiers de configuration subsistant des paquets supprimés.

Vérifiez le résultat de la commande suivante :

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

Si vous pensez que les paquets affichés doivent être purgés, exécutez la commande suivante :

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

Vous pouvez avoir envie de faire la même chose en mode interactif avec un contrôle plus fin.

Indiquez l'expression rationnelle « `~c` » dans la « Nouvelle liste des paquets » en utilisant l'invite « `l` ». Cela limite la vue des paquets à ceux qui correspondent à l'expression rationnelle, c'est-à-dire « supprimé mais non purgé ». On peut visualiser tous les paquets correspondant à cette expression rationnelle en pressant « `[` » depuis une section de haut niveau.

Pressez ensuite « `_` » depuis une fenêtre de haut niveau comme « Paquets non installés ». Seuls les paquets correspondants à l'expression rationnelle se trouvant dans cette section seront marqués comme devant être purgés par cette commande. Vous pouvez exclure certains paquets de cette opération en pressant de manière interactive la touche « `=` » en face de chacun d'eux.

Cette technique est assez pratique et fonctionne avec de nombreuses autres touches de commande.

2.3.4 Toilettage de l'état d'installation automatique/manuel

Voici comment je nettoie l'état d'installation automatique/manuel des paquets (après avoir utilisé un installateur de paquets autre qu'`aptitude`, etc.).

1. Démarrer `aptitude` en mode interactif en tant qu'administrateur.
2. Entrer « `u` », « `U` », « `f` » et « `g` » pour mettre à jour la liste des paquets et mettre à niveau les paquets.
3. Entrer « `l` » afin de définir la limite d'affichage des paquets avec « `~i(~R~i|~Rrecommends:~i)` » et entrez « `M` » sur « Paquets installés » automatiquement.
4. Entrer « `l` » afin de définir la limite d'affichage des paquets avec « `~prequired|~pimportant|~pstandard|~E` » et entrez « `m` » sur les « Paquets installés » manuellement.
5. Entrer « `l` » pour définir la limite d'affichage des paquets avec « `~i!~M` » et supprimez tous les paquets inutilisés en entrant « `-` » sur chacun d'eux après les avoir affichés en entrant « `[` » sur « Paquets installés ».
6. Entrer « `l` » pour définir la limite d'affichage des paquets avec « `~i` » puis entrez « `m` » sur les « Tâches » pour attribuer un marquage « manuellement installé » aux paquets.
7. Quitter `aptitude`.
8. Lancer « `apt-get -s autoremove|less` » en tant qu'administrateur pour vérifier les paquets non utilisés.
9. Redémarrer `aptitude` en mode interactif et marquer les paquets nécessaires comme « `m` ».
10. Redémarrer « `apt-get -s autoremove|less` » en tant qu'administrateur et vérifier à nouveau que « `REMOVED` » ne contient que les paquets voulus.
11. Lancer « `apt-get autoremove|less` » en tant qu'administrateur pour supprimer automatiquement les paquets inutilisés.

L'action « `m` » sur les « Tâches » est facultative pour éviter une situation de suppression en masse de paquets dans le futur.

2.3.5 Mise à jour pour l'ensemble du système

Note

Lors du changement vers une nouvelle version, etc., vous devriez envisager d'effectuer une installation propre d'un nouveau système même si Debian peut être mis à niveau comme décrit ci-dessous. Cela vous donne une chance de supprimer les résidus amassés et vous présente la meilleure combinaison des derniers paquets. Bien entendu, vous devrez effectuer une sauvegarde totale de votre système vers un endroit sûr (consultez [Section 10.2](#)) avant de faire cela. Je vous recommande de faire une configuration dual boot en utilisant des partitions différentes afin d'effectuer une transition en douceur.

Vous pouvez effectuer une mise à niveau de l'ensemble du système vers une nouvelle version en modifiant le contenu du fichier « `/etc/apt/sources.list` » pour qu'il pointe vers la nouvelle version et en lançant la commande « `apt update; apt dist-upgrade` ».

Pour effectuer la mise à jour depuis `stable` vers `testing` ou `unstable`, remplacez « `bullseye` » dans le fichier « `/etc/apt/sources.list` » d'exemple donné dans Section 2.1.4 par « `bookworm` » ou « `sid` ».

En réalité, vous pouvez rencontrer quelques complications en raison de problèmes de transition de paquets, le plus souvent pour des problèmes de dépendances de paquets. Plus la mise à jour est importante, plus vous avez de chances de rencontrer des problèmes importants. Lors de la transition de l'ancienne version `stable` vers la nouvelle version `stable` après sa diffusion, afin de minimiser les problèmes vous pouvez lire ses nouvelles [Notes de diffusion](#) et suivre la procédure exacte qui y est décrite.

Lorsque vous décidez de changer de la version `stable` vers la version `testing` avant sa diffusion formelle, il n'y a pas de [Notes de diffusion](#) pour vous aider. La différence entre `stable` et `testing` peut être devenue assez importante depuis la diffusion de la version `stable` précédente et rendre compliquée la situation de la mise à jour.

Vous devriez aller vers la mise à niveau complète avec précaution tout en récupérant les dernières informations depuis les listes de diffusion et en usant de bon sens.

1. Lire les « Notes de diffusion » précédentes.
2. Faire la sauvegarde de l'ensemble du système (particulièrement les données et les informations de configuration).
3. Avoir un support amorçable prêt au cas où le chargeur initial serait cassé.
4. Informer les utilisateurs du système bien à l'avance.
5. Enregistrer l'activité de mise à jour avec `script(1)`.
6. Appliquer « `unmarkauto` » aux paquets nécessaires, par exemple « `aptitude unmarkauto vim` », afin d'en éviter la suppression.
7. Minimiser les paquets installés pour réduire les chances de conflits de paquets, par exemple supprimer les paquets de la tâche « `bureau` ».
8. Supprimer le fichier « `/etc/apt/preferences` » (désactiver l'épinglage apt « `apt-pinning` »).
9. Essayer de mettre à jour par étapes : `oldstable` → `stable` → `testing` → `unstable`.
10. Mettre à jour le fichier « `/etc/apt/sources.list` » afin qu'il pointe uniquement vers la nouvelle archive et lancer « `aptitude update` ».
11. Installer d'abord, de manière facultative, le nouveau **core packages**, par exemple « `aptitude install perl` ».
12. Lancer la commande « `apt-get -s dist-upgrade` » pour contrôler quel en sera l'impact.
13. Et enfin lancer la commande « `apt-get dist-upgrade` ».

**Attention**

Il n'est pas sage de sauter une version majeure de Debian lors de la mise à niveau entre versions `stable`.

**Attention**

Dans les « Notes de diffusion » précédentes, GCC, Linux Kernel, initrd-tools, Glibc, Perl, APT tool chain, etc. ont demandé une attention particulière pour une mise à niveau de l'ensemble du système.

Pour une mise à jour quotidienne d'`unstable`, consultez Section 2.4.3.

2.4 Opérations avancées de gestion des paquets

2.4.1 Opérations avancées de gestion des paquets en ligne de commandes

Voici la liste des autres opérations de gestion des paquets pour lesquelles `aptitude` est de trop haut niveau ou n'a pas la fonctionnalité requise.

commande	action
<code>COLUMNS=120 dpkg -l motif_nom_paquet</code>	afficher l'état d'un paquet installé pour le signalement de bogue
<code>dpkg -L nom_paquet</code>	afficher le contenu d'un paquet installé.
<code>dpkg -L nom_paquet egrep '/usr/share/man/man.*/.+'</code>	afficher les pages de manuel d'un paquet installé.
<code>dpkg -S motif_nom_fichier</code>	afficher les paquets installés dont le nom correspond
<code>apt-file search motif_nom_fichier</code>	afficher les paquets de l'archive dont le nom correspond
<code>apt-file list motif_nom_paquet</code>	afficher le contenu d'un paquet correspondant de l'archive
<code>dpkg-reconfigure nom_paquet</code>	reconfigurer le paquet exact
<code>dpkg-reconfigure -p low package_name</code>	reconfigurer le paquet exact avec la question la plus détaillée
<code>configure-debian</code>	reconfigurer les paquets depuis le menu en plein écran
<code>dpkg --audit</code>	système de vérification des paquets partiellement installés
<code>dpkg --configure -a</code>	configurer tous les paquets partiellement installés
<code>apt-cache policy nom_paquet_binaire</code>	afficher la version disponible, la priorité et les informations concernant l'archive du paquet binaire
<code>apt-cache madison nom_paquet</code>	afficher la version disponible et les informations de l'archive concernant un paquet
<code>apt-cache showsrc nom_paquet_binaire</code>	afficher les informations concernant le paquet source d'un paquet binaire
<code>apt-get build-dep nom_paquet</code>	installer les paquets nécessaires à la construction d'un paquet
<code>aptitude build-dep nom_paquet</code>	installer les paquets nécessaires à la construction d'un paquet
<code>apt-get source nom_paquet</code>	télécharger une source (depuis l'archive standard)
<code>dget URL for dsc file</code>	télécharger un paquet source (depuis une autre archive)
<code>dpkg-source -x nom_paquet_version-version_debian.dsc</code>	construire une arborescence source depuis un ensemble de paquets source (« *.tar.gz » et « *.diff.gz »)
<code>debuild binary</code>	construire des paquets depuis une arborescence source locale
<code>make-kpkg kernel_image</code>	construire un paquet du noyau à partir de l'arborescence source du noyau
<code>make-kpkg --initrd kernel_image</code>	construire un paquet du noyau à partir de l'arborescence source du noyau avec initramfs activé
<code>dpkg -i nom_paquet_version-version_debian_arch.deb</code>	installer un paquet local sur le système
<code>apt install /chemin/vers/nom_paquet.dev</code>	install a local package to the system, meanwhile try to resolve dependency automatically
<code>debi nom_paquet_version-version_debian_arch.dsc</code>	installer des paquets locaux sur le système
<code>dpkg --get-selections '*'>selection.txt</code>	enregistrer l'information d'état de la sélection des paquets au niveau de dpkg
<code>dpkg --set-selections <selection.txt</code>	définir l'information d'état de sélection des paquets au niveau de dpkg
<code>echo nom_paquet hold dpkg --set-selections</code>	définir l'information d'état de sélection des paquets au niveau de dpkg à hold (gelé, équivalent à <code>aptitude hold nom_paquet</code>)

Table 2.13 – Liste des opérations avancées de gestion des paquets

Note

Pour les paquets disponibles en [multi-arch](#)(itectures), certaines commandes devront parfois être complétées du nom de l'architecture cible. Il faut, par exemple, utiliser « `dpkg -L libglb2.0-0:amd64` » pour obtenir une liste du contenu du paquet `libglb2.0-0` dans son architecture `amd64`.

**Attention**

Un outil de plus bas niveau tel que « `dpkg -i ...` » et « `debi ...` » devra être utilisé avec précautions par l'administrateur du système. Il ne s'assure pas automatiquement des dépendances exigées par le paquet. Les options « `--force-all` » et similaires de la ligne de commandes de `dpkg` (consultez `dpkg(1)`) ne sont prévues pour être utilisées que par des experts. Les utiliser sans comprendre entièrement leurs effets peut casser l'ensemble de votre système.

Veillez noter ce qui suit :

- Toutes les commandes de configuration et d'installation doivent être lancées avec le compte de l'administrateur.
- Au contraire d'`aptitude` qui utilise des expressions rationnelles (consultez Section [1.6.2](#)), les autres commandes de gestion des paquets utilisent des motifs semblables aux motifs génériques (« `glob` ») de l'interpréteur de commandes (consultez Section [1.5.6](#)).
- La commande `apt - file(1)` fournie par le paquet `apt - file` doit être précédée de l'exécution de la commande « `apt - file update` ».
- `configure-debian(8)` fourni par le paquet `configure-debian` exécute en fond `dpkg-reconfigure(8)`.
- `dpkg-reconfigure(8)` exécute les scripts du paquet en utilisant en fond `debconf(1)`.
- Les commandes « `apt-get build-dep` », « `apt-get source` » et « `apt-cache showsrc` » ont besoin d'une entrée « `deb-src` » dans « `/etc/apt/sources.list` ».
- `dget(1)`, `debuild(1)` et `debi(1)` ont besoin du paquet `devscripts`.
- Consultez la procédure de (re)paquetage en utilisant « `apt-get source` » dans Section [2.7.13](#).
- La commande `make-kpkg` exige le paquet `kernel-package` (consultez Section [9.10](#)).
- Consultez Section [12.9](#) pour la réalisation de paquets en général.

2.4.2 Vérifier les fichiers de paquets installés

L'installation de `debsums` permet, avec `debsums(1)`, la vérification des fichiers des paquets installés d'après les valeurs de `MD5sum` se trouvant dans le fichier « `/var/lib/dpkg/info/*.md5sums` » Consultez Section [10.3.5](#) pour le fonctionnement de `MD5sum`.

Note

Comme la base de données `MD5sum` peut être trafiquée par un intrus, `debsums(1)` est d'une utilité restreinte en tant qu'outil de sécurité. Il n'est bon que pour la vérification locale des modifications de l'administrateur ou des défauts en raison de problèmes de support.

2.4.3 Protection contre les problèmes de paquets

De nombreux utilisateurs préfèrent suivre la version **unstable** du système Debian parce qu'elle propose de nouvelles fonctionnalités et de nouveaux paquets. Cela rend le système davantage sujet aux bogues critiques de paquets.

L'installation du paquet `apt-listbugs` protège votre système contre les bogues critiques en recherchant automatiquement dans le BTS de Debian les bogues critiques lors de la mise à jour par l'intermédiaire du système APT.

L'installation du paquet `apt-listchanges` indique les nouveautés importantes se trouvant dans « `NEWS.Debian` » lors de la mise à jour du système avec APT.

2.4.4 Rechercher dans les métadonnées du paquet

Bien que visiter le site Debian à <https://packages.debian.org/> permette aujourd'hui de rechercher facilement les métadonnées des paquets, voyons les méthodes plus traditionnelles.

Les commandes `grep-dctrl(1)`, `grep-status(1)` et `grep-available(1)` peuvent être utilisées pour effectuer des recherches dans tous les fichiers dont le format général est celui d'un fichier de contrôle de Debian.

« `dpkg -S motif_nom_fichier` » peut être utilisé pour rechercher les noms de paquets installés par `dpkg` qui contiennent des fichiers dont le nom correspond au motif. Mais les fichiers créés par les scripts du responsable du paquet ne sont pas pris en compte.

Si vous devez faire des recherches plus élaborées sur les métadonnées de `dpkg`, il vous faudra lancer la commande « `grep -e motif_expression_rationnelle *` » dans le répertoire « `/var/lib/dpkg/info/` ». Cela vous permet de rechercher des mots mentionnés dans les scripts des paquets et les textes des requêtes d'installation.

Pour rechercher de manière récursive les dépendances de paquets, vous devrez utiliser `apt-rdepends(8)`.

2.5 Fonctionnement interne de la gestion des paquets Debian

Voyons comment le système Debian de gestion des paquets fonctionne de manière interne. Cela vous permettra de créer votre propre solution à certains problèmes de paquets.

2.5.1 Métadonnées de l'archive

Les fichiers de métadonnées de chaque distribution se trouvent sur chaque miroir Debian, dans « `dist/nom_de_code` », par exemple, « `http://deb.debian.org/debian/` ». On peut parcourir la structure de son archive à l'aide d'un navigateur web. Il existe 6 types de métadonnées clés :

fichier	emplacement	contenu
Release	sommet de la distribution	description de l'archive et informations d'intégrité
Release.gpg	sommet de la distribution	fichier signature du fichier « Release » signé avec la clé de l'archive
Contents-architecture	sommet de la distribution	liste de tous les fichiers pour tous les paquets dans l'archive pertinente
Release	sommet de chaque combinaison de distribution/section/architecture	description de l'archive utilisée pour la règle de <code>apt_preferences(5)</code>
Packages	sommet de chaque combinaison de distribution/section/architecture binaire	<code>debian/control</code> concaténés des paquets binaires
Sources	sommet de chaque combinaison distribution/section/source	<code>debian/control</code> concaténés des paquets sources

Table 2.14 – Contenu des métadonnées de l'archive Debian

Dans les archives récentes, ces métadonnées sont enregistrées sous forme compressée et différentielle afin de limiter le trafic réseau.

2.5.2 Fichier « Release » de plus haut niveau et authenticité

ASTUCE

Le fichier « Release » de plus haut niveau est utilisé pour signer l'archive au moyen du système **secure APT**.

Chaque version de l'archive Debian possède un fichier « `Re lease` » de plus haut niveau, par exemple, « `http://deb.debian.org` » comme ci-dessous :

```
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
  ...
```

Note

Vous pouvez trouver ici ma justification pour l'utilisation de « suite », « nom de code » dans Section 2.1.4. La « distribution » est utilisée pour désigner à la fois « suite » et « nom de code ». Tous les noms possibles de « sections » de l'archive sont indiqués dans l'archive sous « Composants ».

L'intégrité du fichier « `Re lease` » de plus haut niveau est vérifiée par une infrastructure cryptographique appelée [secure apt](#).

- Le fichier de signature chiffré « `Re lease . gpg` » est créé à partir du fichier « `Re lease` » réel de plus haut niveau et de la clé secrète de l'archive Debian.
- La clé publique de l'archive Debian peut être placée dans « `/etc/apt/trusted . gpg` » ;
 - automatiquement en installant le trousseau de clés avec la dernière version du paquet `base - files`, ou
 - manuellement, avec les outils `gpg` ou `apt - key` en utilisant [la dernière clé publique de l'archive postée sur ftp-master.debian.org](#).
- Le système **secure APT** vérifie de manière cryptographique, à l'aide du fichier « `Re lease . gpg` » et de la clé publique de l'archive Debian de « `/etc/apt/trusted . gpg` », l'intégrité du fichier « `Re lease` » de plus haut niveau téléchargé .

L'intégrité de tous les fichiers « `Packages` » et « `Sources` » est vérifiée en utilisant les valeurs des sommes MD5 se trouvant dans son fichier de plus haut niveau « `Re lease` ». L'intégrité de tous les fichiers de paquets est vérifiée en utilisant les valeurs des sommes MD5 se trouvant dans les fichiers « `Packages` » et « `Sources` ». Consultez `debsums(1)` et Section 2.4.2.

Comme la vérification de la signature cryptographique est un processus beaucoup plus consommateur de processeur que les calculs de sommes MD5, l'utilisation d'une somme MD5 pour chacun des paquets tout en utilisant une signature cryptographique pour le fichier « `Re lease` » de plus haut niveau allie [une bonne sécurité avec de bonnes performances](#) (consultez Section 10.3).

2.5.3 Fichiers « `Release` » au niveau de l'archive

ASTUCE

Les fichiers « `Re lease` » au niveau de l'archive sont utilisés pour la règle d'`apt_preferences(5)`.

Il y a des fichiers « `Re lease` » du niveau de l'archive pour tous les emplacements d'archives indiqués par une ligne « `deb` » dans « `/etc/apt/sources . list` » comme, par exemple, « `http://deb.debian.org/debian/dists/unstable/main/bi` » ou « `http://deb.debian.org/debian/dists/sid/main/binary-amd64/Release` » comme suit :

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

**Attention**

Pour l'entrée « Archive: », les noms de version (« stable », « testing », « unstable », ...) sont utilisés dans [l'archive Debian](#) alors que les noms de code (« trusty », « xenial », « artful », ...) sont utilisés dans [l'archive Ubuntu](#).

Pour certaines archives, comme `experimental` et `bullseye-backports`, qui contiennent des paquets qui ne devraient pas être installés automatiquement, il y a une ligne supplémentaire, par exemple « `http://deb.debian.org/debian/dists/experimental` » comme suit :

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

Remarquez que les archives normales, sans « `NotAutomatic: yes` », la valeur par défaut de « `Pin-Priority` » est de 500, alors que pour les archives spéciales avec « `NotAutomatic: yes` » la valeur par défaut de « `Pin-Priority` » est de 1 (consultez `apt_preferences(5)` et Section 2.7.3).

2.5.4 Récupérer les métadonnées d'un paquet

Lorsqu'on utilise les outils APT, comme `aptitude`, `apt-get`, `synaptic`, `apt-file`, `auto-apt`, etc., il faut mettre à jour les copies locales des métadonnées contenant les informations de l'archive Debian. Ces copies locales ont les noms de fichier suivants, correspondant aux noms de la `distribution`, de la `section` et de l'`architecture` indiquées dans « `/etc/apt/sources.list` » (consultez Section 2.1.4).

- « `/var/lib/apt/lists/deb.debian.org_debian_dists_distribution_Release` »
- « `/var/lib/apt/lists/deb.debian.org_debian_dists_distribution_Release.gpg` »
- « `/var/lib/apt/lists/deb.debian.org_debian_dists_distribution_area_binary-architecture_Packages` »
- « `/var/lib/apt/lists/deb.debian.org_debian_dists_distribution_area_source_Sources` »
- « `/var/cache/apt/apt-file/deb.debian.org_debian_dists_distribution_Contents-architecture` » (pour `apt-file`)

Les quatre premiers types de fichiers sont partagés par toutes les commandes APT pertinentes et sont mis à jour depuis la ligne de commandes par « `apt-get update` » ou « `aptitude update` ». Les métadonnées « `Packages` » sont mises à jour s'il y a une ligne « `deb` » dans le fichier « `/etc/apt/sources.list` ». Les métadonnées « `Sources` » sont mises à jour s'il y a une ligne « `deb-src` » dans le fichier « `/etc/apt/sources.list` ».

Les métadonnées « `Packages` » et « `Sources` » contiennent une entrée « `Filename:` » pointant vers l'emplacement du paquet binaire et du paquet source. Actuellement, ces paquets sont situés dans l'arborescence du répertoire « `pool/` » afin d'améliorer le passage d'une version à l'autre.

On peut effectuer des recherches interactivement dans les copies locales des métadonnées « `Packages` » à l'aide d'`aptitude`. La commande de recherche spécialisée `grep-dctrl(1)` peut effectuer des recherches dans les copies locales des métadonnées « `Packages` » et « `Sources` ».

La copie locale des métadonnées « `Contents-architecture` » peut être mise à jour par « `apt-file update` », son emplacement est différent des quatre autres. Consultez `apt-file(1)`. (`auto-apt` utilise par défaut un emplacement différent pour la copie locale de « `Contents-architecture.gz` »).

2.5.5 État des paquets pour APT

En plus des métadonnées récupérées par téléchargement, l'outil APT des versions ultérieures à Lenny enregistre l'état de l'installation généré localement dans « `/var/lib/apt/extended_states` » qui est utilisé par tous les outils APT afin de suivre tous les paquets installés automatiquement.

2.5.6 État des paquets pour aptitude

En plus des métadonnées récupérées par téléchargement, la commande `aptitude` enregistre l'état de l'installation généré localement dans « `/var/lib/aptitude/pkgstates` » qu'il est le seul à utiliser.

2.5.7 Copies locales des paquets téléchargés

Tous les paquets ayant été téléchargés au moyen du mécanisme APT sont enregistrés dans le répertoire « `/var/cache/apt/archive` » jusqu'à ce qu'ils en soient supprimés.

Cette politique de nettoyage des fichiers de cache pour `aptitude` peut être spécifiée sous « Options » → « Préférences » et peut être déclenchée manuellement par le menu « Nettoyer le cache des paquets » ou « Enlever les fichiers périmés » sous « Actions ».

2.5.8 Nom de fichier d'un paquet Debian

Les fichiers de paquets Debian ont une structure de nom particulière.

type de paquet	structure du nom
Paquet binaire (encore appelé deb)	<code>package-name_upstream-version-debian.revision_architecture</code>
Le paquet binaire de l'installateur de Debian (connu sous le nom udeb)	<code>package-name_upstream-version-debian.revision_architecture.deb</code>
paquet source (source amont)	<code>package-name_upstream-version-debian.revision.orig.tar.gz</code>
Paquet source 1.0 (modifications Debian)	<code>package-name_upstream-version-debian.revision.diff.gz</code>
Paquet source 3.0 (quilt) (modifications Debian)	<code>package-name_upstream-version-debian.revision.debian.tar.gz</code>
paquet source (description)	<code>package-name_upstream-version-debian.revision.dsc</code>

Table 2.15 – Structure du nom des paquets Debian :

ASTUCE

Seuls les formats de paquets source de base sont décrits ici. Consultez `dpkg-source(1)` pour davantage d'informations.

nom de la composante	usable characters (ERE regex)	existence
<code>nom_paquet</code>	<code>[a-z0-9][-a-z0-9.]+</code>	nécessaire
<code>epoch:</code>	<code>[0-9]+:</code>	optionnel
<code>version_amont</code>	<code>[-a-zA-Z0-9.+:]+</code>	nécessaire
<code>version_debian</code>	<code>[a-zA-Z0-9.+~]+</code>	optionnel

Table 2.16 – Caractères utilisables pour chacune des composantes des noms de paquets Debian

Note

Vous pouvez vérifier l'ordre des versions d'un paquet à l'aide de `dpkg(1)`, par exemple, « `dpkg --compare-versions 7.0 gt 7.~pre1 ; echo $?` ».

Note

L'installateur Debian (d-i) utilise udeb comme extension du nom de fichier de ses paquets binaires plutôt que le deb normal. Un paquet udeb est un paquet deb allégé dont certaines parties non essentielles du contenu, comme la documentation, sont supprimées afin d'économiser de la place en relâchant les exigences de la charte des paquets. Les paquet deb et udeb partagent la même structure de paquet. Le « u » signifie micro.

2.5.9 La commande dpkg

`dpkg(1)` est l'outil de plus bas niveau pour la gestion des paquets de Debian. C'est un outil très puissant et il faut l'utiliser avec précaution.

Lors de l'installation d'un paquet appelé « *nom_paquet* », `dpkg` le traite selon l'ordre suivant :

1. dépaquetage du fichier deb (équivalent à « `ar -x` ») ;
2. exécution de « *nom_paquet.preinst* » en utilisant `debconf(1)` ;
3. installation du contenu du paquet sur le système (équivalent à « `tar -x` ») ;
4. exécution de « *nom_paquet.postinst* » en utilisant `debconf(1)`.

Le système `debconf` fournit une interaction standardisée avec l'utilisateur avec la prise en charge de I18N and L10N (Chapitre 8).

fichier	description du contenu
<code>/var/lib/dpkg/info/nom_paquet.conf</code>	fichiers de configuration. (modifiables par l'utilisateur)
<code>/var/lib/dpkg/info/nom_paquet.list</code>	liste des fichiers et répertoires installés par le paquet
<code>/var/lib/dpkg/info/nom_paquet.md5sums</code>	liste des valeurs de hachage MD5 pour les fichiers installés par le paquet
<code>/var/lib/dpkg/info/nom_paquet.preinst</code>	script du paquet à exécuter avant l'installation du paquet
<code>/var/lib/dpkg/info/nom_paquet.postinst</code>	script du paquet à exécuter après l'installation du paquet
<code>/var/lib/dpkg/info/nom_paquet.prerm</code>	script du paquet à exécuter avant la suppression du paquet
<code>/var/lib/dpkg/info/nom_paquet.postrm</code>	script du paquet à exécuter après la suppression du paquet
<code>/var/lib/dpkg/info/nom_paquet.conffile</code>	script du paquet pour le système <code>debconf</code>
<code>/var/lib/dpkg/alternatives/nom_paquet</code>	information d'alternative utilisée par la commande <code>update-alternatives</code>
<code>/var/lib/dpkg/available</code>	information de disponibilité de tous les paquets
<code>/var/lib/dpkg/diversions</code>	information sur les détournements utilisés par <code>dpkg(1)</code> et établis avec <code>dpkg-divert(8)</code>
<code>/var/lib/dpkg/statoverride</code>	information de remplacement de statut utilisé par <code>dpkg(1)</code> et établi avec <code>dpkg-statoverride(8)</code>
<code>/var/lib/dpkg/status</code>	informations d'état pour tous les paquets
<code>/var/lib/dpkg/status-old</code>	fichier de sauvegarde de première génération du fichier « <code>/var/lib/dpkg/status</code> »
<code>/var/backups/dpkg.status*</code>	fichier de sauvegarde de seconde génération du fichier « <code>/var/lib/dpkg/status</code> »

Table 2.17 – Fichiers particuliers créés par `dpkg`

Le fichier « `status` » est aussi utilisé par des outils comme `dpkg(1)`, « `dselect update` » et « `apt-get -u dselect-upgrade` ».

La commande de recherche spécialisée `grep-dctrl(1)` peut rechercher des copies locales des métadonnées « `status` » et « `available` ».

ASTUCE

Dans l'environnement de l'installateur `debian`, la commande `udpkg` est utilisée pour ouvrir les paquets `udeb`. La commande `udpkg` est une version allégée de la commande `dpkg`.

2.5.10 La commande update-alternatives

Le système Debian possède un mécanisme pour installer paisiblement des paquets qui présentent un certain recouvrement en utilisant `update-alternatives(1)`. Par exemple, vous pouvez faire que la commande `vi` choisisse de lancer `vim` alors que les paquets `vim` et `nvi` sont tous deux installés.

```
$ 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
```

Le système d'alternatives de Debian utilise des liens symboliques dans « /etc/alternatives/ » pour enregistrer ses sélections. Le processus de sélection utilise le fichier correspondant de « /var/lib/dpkg/alternatives/ ».

2.5.11 Commande dpkg-statoverride

Stat overrides, fournie par la commande `dpkg-statoverride(8)` est un moyen d'indiquer à `dpkg(1)` d'utiliser un propriétaire ou un mode différent pour un **fichier** lorsqu'un paquet est installé. Si « `--update` » est indiqué et que le fichier existe, il est immédiatement configuré avec le nouveau propriétaire et le nouveau mode.



Attention

Une modification directe par l'administrateur du propriétaire ou du mode d'un **fichier** dont le propriétaire est le paquet en utilisant les commandes `chmod` ou `chown` sera réinitialisée lors d'une nouvelle mise à niveau du paquet.

Note

J'utilise ici le mot **fichier**, mais en réalité, ce peut être n'importe quel objet d'un système de fichiers que gère `dpkg`, y compris les répertoires, les périphériques, etc.

2.5.12 Commande dpkg-divert

Les fichiers **diversions** fournis par la commande `dpkg-divert(8)` sont un moyen de forcer `dpkg(1)` à ne pas installer un fichier à son emplacement par défaut, mais à un emplacement **détourné** (« `diverted` »). L'utilisation de `dpkg-divert` est destinée à la maintenance de paquets par des scripts. Son utilisation occasionnelle par l'administrateur du système est obsolète.

2.6 Récupérer un système cassé

En utilisant le système `unstable`, l'administrateur peut avoir à restaurer le système à partir d'une situation où la gestion des paquets est défectueuse.



Attention

Certaines des méthodes décrites ici sont des actions très risquées. Vous avez été prévenu !

2.6.1 Incompatibilité avec une ancienne configuration de l'utilisateur

Si un programme avec une interface graphique présente une instabilité après une mise à niveau amont importante, vous devriez songer à des interférences avec les anciens fichiers de configurations locaux qu'il avait créés. S'il est stable avec un compte d'utilisateur fraîchement créé, cette hypothèse est confirmée. (C'est un bogue de réalisation du paquet et c'est le plus souvent évité par le responsable du paquet).

Pour retrouver la stabilité, vous devrez déplacer les fichiers de configuration locaux et redémarrer le programme ayant une interface graphique. Il vous faudra peut-être lire le contenu de l'ancien fichier de configuration pour retrouver plus tard vos informations de configuration. (Ne les effacez pas trop rapidement).

2.6.2 Différents paquets ayant des fichiers communs

Les systèmes de gestion d'archive au niveau du paquet, tels qu'`aptitude(8)` ou `apt-get(1)` ne tenteront même pas, grâce aux dépendances des paquets, d'installer des paquets ayant des fichiers qui se superposent (consultez Section 2.1.6).

Des erreurs du responsable du paquet ou le déploiement de sources d'archives mélangées et incohérentes (consultez Section 2.7.2) par l'administrateur du système peuvent créer une situation où les dépendances des paquets sont décrites de manière incorrecte. Lorsque, dans une telle situation, vous installez un paquet qui écrase des fichiers en utilisant `aptitude(8)` ou `apt-get(1)`, `dpkg(1)` qui dépaquette le paquet va retourner une erreur au programme appelant sans écraser les fichiers existants.



Attention

L'utilisation de programmes tierce partie introduit un risque significatif par l'intermédiaire des scripts du responsable qui sont lancés avec les privilèges de l'administrateur et peuvent effectuer n'importe quoi sur votre système. La commande `dpkg(1)` ne protège que contre l'écrasement des fichiers lors du dépaquetage.

Vous pouvez contourner un tel problème d'installation cassée en supprimant d'abord l'ancien paquet, *ancien_paquet*, qui pose des problèmes.

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

2.6.3 Corriger les scripts cassés des paquets

Lorsqu'une commande dans le script du paquet retourne une erreur pour une raison quelconque et que le script retourne une erreur, le système de gestion des paquets arrête son action et se termine en laissant des paquets partiellement installés. Lorsqu'un paquet comporte des bogues dans les scripts de suppression, le paquet peut devenir impossible à supprimer et assez déplaisant.

Pour les problèmes avec le script de paquet de « *nom_paquet* », il vous faudra regarder dans les scripts du paquet suivants :

- « `/var/lib/dpkg/info/nom_paquet.preinst` »
- « `/var/lib/dpkg/info/nom_paquet.postinst` »
- « `/var/lib/dpkg/info/nom_paquet.prerm` »
- « `/var/lib/dpkg/info/nom_paquet.postrm` »

Éditez le script du paquet posant problème avec le compte de l'administrateur en utilisant les techniques suivantes :

- désactiver la ligne posant problème avec un « `#` » en tête de ligne ;
- forcer un retour avec succès en ajoutant à la fin de la ligne qui pose problème « `|| true` ».

Configurez tous les paquets partiellement installés à l'aide de la commande suivante :

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

2.6.4 Récupération avec la commande dpkg

Comme `dpkg` est un outil de gestion des paquets de très bas niveau, il peut fonctionner dans des situations très difficiles telles qu'un système qu'on ne peut pas démarrer et qui n'a pas de connexion réseau. Supposons que le paquet `toto` soit cassé et doive être remplacé.

You may still find cached copies of older bug free version of `foo` package in the package cache directory: `"/var/cache/apt/archives/` (If not, you can download it from archive of <https://snapshot.debian.org/> or copy it from package cache of a functioning machine.)

Si vous pouvez démarrer le système, vous pouvez l'installer avec la commande suivante :

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

ASTUCE

Si la casse du système est minime, vous pouvez faire un retour en arrière (downgrade) de tout le système comme dans Section 2.7.10 en utilisant le système de plus haut niveau APT.

Si votre système ne peut pas être démarré depuis le disque dur, vous devrez rechercher d'autres manières de le démarrer :

1. Démarrez le système en utilisant le CD de l'installateur Debian en mode secours (« rescue mode »).
2. Montez le système qui ne veut pas démarrer sur le disque dur sur « `/target` ».
3. Installez une version plus ancienne du paquet `toto` en faisant ce qui suit :

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

Cet exemple fonctionne même si la commande `dpkg` qui se trouve sur le disque dur est cassée.

ASTUCE

Tous les systèmes GNU/Linux démarrés depuis un autre système sur le disque dur, depuis un CD autonome GNU/Linux, depuis une clé USB amorçable, depuis le réseau peuvent être utilisés de manière similaire pour restaurer un système cassé.

Si la tentative d'installation d'un paquet de cette manière échoue en raison de la violation de certaines dépendances et que vous voulez vraiment le faire, vous pouvez, en dernier ressort, outrepasser les dépendances en utilisant les options « `--ignore-depends` », « `--force-depends` » de `dpkg` et d'autres options. Si vous le faites, vous aurez un sérieux effort à faire pour restaurer les dépendances correctes par la suite. Consultez `dpkg(8)` pour davantage d'informations.

Note

Si votre système est sérieusement cassé, vous devriez faire une sauvegarde complète du système dans un endroit sûr (consultez Section 10.2) et effectuer une installation propre. Cela demande moins de temps et donne, en fin de compte, de meilleurs résultats.

2.6.5 Récupérer les données de sélection des paquets

Si le fichier « `/var/lib/dpkg/status` » est corrompu pour une raison quelconque, le système Debian perd les données de paquets sélectionnés et est sérieusement endommagé. Regardez l'ancienne version du fichier « `/var/lib/dpkg/status` » dans « `/var/lib/dpkg/status-old` » ou « `/var/backups/dpkg.status.*` ».

Conserver « `/var/backups/` » sur une partition séparée peut être une bonne idée car ce répertoire contient de nombreuses données importantes du système.

Pour les casses sévères, je recommande de faire une réinstallation propre après avoir fait une sauvegarde du système. Même si vous avez perdu tout ce qui se trouve dans « `/var/` », vous pouvez encore récupérer certaines informations depuis les répertoires qui se trouvent dans « `/usr/share/doc/` » afin de vous guider dans votre nouvelle installation.

Réinstaller un système (de bureau) minimum.

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

Montez l'ancien système sur « /chemin/vers/ancien/système/ ».

```
# 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
```

Le système vous présentera alors les noms de paquets à installer. (Il peut y avoir des noms qui ne soient pas des noms de paquets, comme, par exemple, « `texmf` »).

2.7 Astuces pour la gestion des paquets

2.7.1 Comment obtenir des paquets Debian

Vous pouvez rechercher les paquets qui satisfont à vos besoins avec `aptitude` à partir de la description du paquet ou depuis la liste se trouvant dans « Tasks ».

Si vous trouvez deux paquets similaires et que vous vous demandez lequel installer sans faire des efforts d'« essais et erreurs », vous pouvez user de **bon sens**. Je considère que les points suivants constituent de bonnes indications pour les paquets à privilégier :

- Essential : yes > no
- Section : main > contrib > non-free
- Priorité : required > important > standard > optional > extra
- Tâches : paquets affichés dans les tâches tels que « Environnement de bureau »
- Packages selected by the dependency package (e.g., `gcc-10` by `gcc`)
- Popcon : les votes et le nombre d'installations les plus élevés
- Journaux des modifications (« Changelog ») : mises à jour régulières par le responsable
- BTS : pas de bogue RC (pas de bogue critique, grave ou sérieux)
- BTS : réactivité du responsable aux signalements de bogues
- BTS : le plus grand nombre de bogues réglés récemment
- BTS : le plus faible nombre de bogues restants qui ne soient pas dans la liste des vœux

Debian étant un projet basé sur le volontariat avec un modèle développement distribué, son archive contient de nombreux paquets avec des cibles différentes et de qualité variable. Vous devrez choisir vous-même ce que vous voulez en faire.

2.7.2 Paquets venant de sources mélangées



Attention

Installer des paquets provenant de sources d'archives mélangées n'est pas pris en charge par la distribution officielle de Debian sauf pour la prise en charge officielle d'une combinaison particulière d'archives telle que `stable` avec [security updates](#) et [bullseye-updates](#).

Voici un exemple des opérations pour inclure des paquets d'une version amont spécifique plus récente se trouvant dans `unstable` pour une seule occasion tout en suivant `testing` :

1. modifier le fichier « `/etc/apt/sources.list` » de manière temporaire avec la seule entrée « `unstable` » ;
2. lancer « `aptitude update` » ;

3. lancer « `aptitude install nom_paquet` » ;
4. rétablir le fichier « `/etc/apt/sources.list` » pour `testing`.
5. lancer « `aptitude update` » ;

Vous ne créez pas le fichier « `/etc/apt/preferences` » et vous n’avez pas besoin de vous préoccuper de l’épinglage apt (« apt-pinning ») avec cette approche manuelle. Mais c’est très lourd.

**Attention**

En utilisant une source d’archive mixte, vous devez vous assurer par vous-même de la compatibilité des paquets car Debian ne la garantit pas. S’il existe des incompatibilités de paquets, vous pouvez casser votre système. Vous devrez être capable d’apprécier ces exigences techniques. L’utilisation de sources mixtes d’archives aléatoires est une opération entièrement facultative et son utilisation est quelque chose que je ne vous encourage pas à faire.

Les règles générales pour l’installation de paquets de différentes archives sont les suivantes :

- Les paquets non binaires (« `Architecture: all` ») sont plus **sûrs** à installer :
 - paquets de documentation : pas d’exigence particulière
 - paquet de programmes d’interpréteur : un interpréteur compatible doit être disponible
- Les paquets binaires (qui ne sont pas « `Architecture: all` ») sont confrontés à de nombreux barrages et ne sont **pas sûrs** à installer
 - compatibilité des versions de bibliothèques (y compris « `libc` »)
 - compatibilité des versions des programmes utilitaires en rapport
 - compatibilité avec l’[ABI](#) du noyau
 - compatibilité avec l’[ABI](#) C++
 - ...

Note

De manière à rendre un paquet **plus sûr** à installer, certains programmes commerciaux binaires et non libres peuvent être fournis liés avec des bibliothèques complètement statiques. Vous devrez quand même vérifier leurs problèmes de compatibilité avec l’[ABI](#), etc.

Note

Sauf pour éviter à court terme un paquet cassé, installer des paquets binaires depuis des archives officiellement non supportées est en général une mauvaise idée. C’est vrai même si vous utilisez l’épinglage apt (« apt-pinning ») (consultez Section [2.7.3](#)). Vous devriez envisager chroot ou des techniques similaires (consultez Section [9.11](#)) afin de faire tourner des programmes en provenance d’archives différentes.

2.7.3 Ajuster la version candidate

**AVERTISSEMENT**

L’utilisation de l’épinglage apt par un débutant est une source certaine de problèmes majeurs. Vous devriez éviter l’utilisation de l’épinglage apt sauf si vous en avez absolument besoin.

Sans le fichier « `/etc/apt/preferences` », le système APT choisit, en utilisant la chaîne de version, la dernière version disponible comme **version candidate**. C’est l’état normal et l’utilisation la plus recommandée du système APT. Toutes les combinaisons d’archives officiellement prises en charge n’exigent pas le fichier « `/etc/apt/preferences` » car certaines archives qui ne peuvent pas être utilisées comme source des mises à jour automatiques sont marquées **NotAutomatic** et gérées proprement.

ASTUCE

La règle de comparaison de la chaîne de version peut être vérifiée avec, par exemple « `dpkg --compare-versions ver1.1 gt ver1.1~1; echo $?` » (consultez `dpkg(1)`).

Lorsque vous installez régulièrement des paquets depuis un mélange de sources d'archives (consultez Section 2.7.2), vous pouvez automatiser ces opérations compliquées en créant le fichier « `/etc/apt/preferences` » avec les entrées correctes et en ajustant la règle de sélection des paquets pour la **version candidate** comme décrit dans `apt_preferences(5)`. C'est appelé **épinglage apt** (« apt-pinning »).

**Attention**

Lorsque vous utilisez l'épinglage apt, vous devez vérifier vous-même la compatibilité des paquets car le système Debian ne la garantit pas. L'épinglage apt est une opération entièrement facultative et je ne vous encourage pas à l'utiliser.

**Attention**

Les fichiers Release au niveau de l'archive (consultez Section 2.5.3) sont utilisés pour la règle de `apt_preferences(5)`. L'épinglage apt ne fonctionne donc qu'avec le nom de « suite » pour les [archives normales de Debian](#) et les [archives de sécurité de Debian](#) c'est différent des archives [Ubuntu](#)). Par exemple, dans le fichier « `/etc/apt/preferences` », il est possible de mettre « `Pin: release a=unstable` » mais pas « `Pin: release a=sid` »

**Attention**

Lorsque vous utilisez une archive ne venant pas de Debian en tant que partie d'épinglage apt, vous devez vérifier ce pour quoi elles sont prévues et aussi vérifier leur crédibilité. Par exemple, Ubuntu et Debian ne sont pas prévues pour être mélangées.

Note

Même si vous ne créez pas le fichier « `/etc/apt/preferences` », vous pouvez effectuer des opérations assez complexes sur le système sans épinglage apt (consultez Section 2.6.4 et Section 2.7.2).

Voici une explication simplifiée de la technique d'**épinglage apt** (« apt-pinning ») :

Le système APT choisit la **mise à niveau** du paquet de plus haute priorité d'épinglage (« Pin-Priority ») dans la liste des sources de paquets disponible dans le fichier « `/etc/apt/sources.list` » comme paquet de « **version candidate**. Si la priorité d'épinglage du paquet est supérieure à 1000, cette restriction de version pour la **mise à niveau** est levée afin de permettre le retour vers une version précédente (consultez Section 2.7.10).

La valeur de priorité d'épinglage de chaque paquet est définie par l'entrée « Pin-Priority » dans le fichier « `/etc/apt/preferences` » ou utilise sa valeur par défaut.

Il y a plusieurs méthodes pour définir l'archive **version cible** :

- fichier de configuration « `/etc/apt/apt.conf` » avec la ligne « `APT::Default-Release "stable";` »
- option de la ligne de commandes, par exemple, « `apt-get install -t testing un_paquet` »

L'archive **non automatique mais mises à niveau automatiques** est définie par le serveur d'archive dont le fichier Release au niveau de l'archive (consultez Section 2.5.3) contient à la fois « `NotAutomatic: yes` » et « `ButAutomaticUpgrades: yes` ». L'archive **non automatique** est définie par le serveur d'archive dont le fichier Release au niveau de l'archive contient « `NotAutomatic: yes` ».

La **situation d'épinglage apt** de *paquet* provenant de sources d'archive multiple est affichée par « `apt-cache policy paquet` ».

priorité d'épinglage	conséquence de l'épinglage apt sur le paquet
1001	installer le paquet même s'il s'agit d'un retour en arrière
990	utilisé par défaut pour l'archive version cible
500	utilisé par défaut pour l'archive normale
100	utilisé par défaut pour l'archive non automatique mais mises à niveau automatiques
100	utilisé pour le paquet installé
1	utilisé par défaut pour l'archive non automatique
-1	ne jamais installer le paquet même s'il est recommandé

Table 2.18 – Liste de valeurs remarquables des priorités d'épinglage (Pin-Priority) pour la technique d'**épinglage apt** (« apt-pinning ») :

- Une ligne commençant par « **Package pin:** » affiche la version d'**épinglage** du paquet si l'association n'est définie que pour *paquet* par exemple, « **Package pin:** 0.190 ».
- Il n'existe pas de ligne avec « **Package pin:** » s'il n'y a pas d'association définie uniquement avec *paquet*.
- La valeur de Pin-Priority associée uniquement avec *paquet* est affichée sur la partie droite de toutes les chaînes de version, par exemple, « 0.181 700 ».
- « 0 » est affiché à droite de toutes les chaînes de version s'il n'y a pas d'association définie avec uniquement *paquet*, par exemple, « 0.181 0 ».
- Les valeurs de Pin-Priority des archives (définies par « **Package:** * » dans le fichier « /etc/apt/preferences ») sont affichées sur la gauche de tous les chemins vers les archives, par exemple, « 100 http://deb.debian.org/debian/bullseye-backports/main Packages ».

2.7.4 Mises à jour et rétroportages

Les archives du [bullseye-updates](http://deb.debian.org/debian/bullseye) et backports.debian.org fournissent des paquets mis à niveau pour stable (bullseye).

Afin d'utiliser ces archives, placez la liste de toutes les archives nécessaires dans le fichier « /etc/apt/sources.list » de la manière suivante :

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

Il n'est pas nécessaire de définir explicitement la valeur de Pin-Priority dans le fichier « /etc/apt/preferences ». Quand de nouveaux paquets sont disponibles, la configuration par défaut fournit les mises à niveau les plus saines (consultez Section 2.5.3).

- Tous les paquets installés les plus anciens sont mis à niveau vers les plus récents à partir de **bullseye-updates**.
- Seuls les paquets les plus anciens installés à partir de **bullseye-backports** sont mis à niveau vers les plus récents à partir de **bullseye-backports**.

Chaque fois que vous désirerez installer un paquet nommé « *nom_paquet* » avec ses dépendances depuis l'archive **bullseye-backports** vous-même, vous utiliserez la commande suivante en changeant de version cible avec l'option « -t » :

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

2.7.5 Blocage des paquets installés par « Recommends »



AVERTISSEMENT

L'utilisation de l'épinglage apt par un débutant est une source certaine de problèmes majeurs. Vous devriez éviter l'utilisation de l'épinglage apt sauf si vous en avez absolument besoin.

Si vous ne voulez pas tirer des paquets particuliers automatiquement avec « Recommends », vous devez créer le fichier « `/etc/apt/preferences` » et y placer la liste explicite de tous ces paquets au début du fichier comme suit :

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

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

2.7.6 Suivre testing avec quelques paquets d'instable



AVERTISSEMENT

L'utilisation de l'épingle apt par un débutant est une source certaine de problèmes majeurs. Vous devriez éviter l'utilisation de l'épingle apt sauf si vous en avez absolument besoin.

Voici un exemple de technique d'**épingle apt** permettant d'inclure de manière régulière une version amont plus récente de paquets spécifiques se trouvant dans `unstable` tout en suivant `testing`. Listez toutes les archives nécessaires dans le fichier « `/etc/apt/sources.list` » de la manière suivante :

```
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/ testing-security main contrib
```

Configurez le fichier « `/etc/apt/preferences` » comme suit :

```
Package: *
Pin: release a=unstable
Pin-Priority: 100
```

Lorsque vous désirez installer un paquet appelé « *nom_paquet* » avec ses dépendances depuis l'archive `unstable` avec cette configuration, vous utilisez la commande suivante qui modifie la version cible avec l'option « `-t` » (la priorité d'épingle de `unstable` devient 990).

```
$ sudo apt-get install -t unstable package-name
```

Avec cette configuration, l'exécution habituelle de « `apt-get upgrade` » et « `apt-get dist-upgrade` » (ou « `aptitude safe-upgrade` » et « `aptitude full-upgrade` » met à niveau les paquets qui avaient été installés depuis l'archive `testing` en utilisant l'archive `testing` actuelle et les paquets qui avaient été installés depuis l'archive `unstable` en utilisant l'archive `unstable` actuelle.



Attention

Faites bien attention à ne pas supprimer l'entrée « `testing` » du fichier « `/etc/apt/sources.list` ». Sans l'entrée « `testing` », le système APT mettra à niveau les paquets en utilisant la nouvelle archive `unstable`.

ASTUCE

J'édite habituellement le fichier « `/etc/apt/sources.list` » en commentant l'entrée correspondant à l'archive « `unstable` » juste après avoir effectué les opérations ci-dessus. Cela évite un processus de mise à jour lent en raison du nombre trop important d'entrées dans le fichier « `/etc/apt/sources.list` » bien que cela ne permette pas de mettre à niveau les paquets qui avaient été installés depuis l'archive `unstable` en utilisant l'archive `unstable` actuelle.

ASTUCE

Si on utilise « Pin-Priority: 1 » à la place de « Pin-Priority: 100 » dans le fichier « /etc/apt/preferences », les paquets déjà installés ayant une valeur de Pin-Priority de 100 ne seront pas mis à niveau depuis l'archive `unstable` même si l'entrée « `testing` » du fichier « /etc/apt/sources.list » est supprimée.

Si vous désirez suivre automatiquement un paquet particulier dans `unstable` sans une installation initiale « `-t unstable` », vous devrez créer le fichier « /etc/apt/preferences » et y placer la liste explicite de tous ces paquets au début du fichier de la manière suivante :

```
Package: package-1
Pin: release a=unstable
Pin-Priority: 700
```

```
Package: package-2
Pin: release a=unstable
Pin-Priority: 700
```

Cela définit la valeur de Pin-Priority pour chacun de ces paquets spécifiques. Par exemple, pour suivre la dernière version `unstable` de cette « Référence Debian » en français, vous devrez ajouter les entrées suivantes dans le fichier « /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
```

ASTUCE

Cette technique d'épinglage apt est aussi valable si vous suivez l'archive `stable`. Jusqu'à présent et selon mon expérience, les paquets de documentation ont toujours été sûrs à installer depuis l'archive `unstable`.

2.7.7 Suivre `unstable` avec quelques paquets d'`experimental`

Voici un autre exemple de technique d'**épinglage apt** destinée à inclure une version amont plus récente de paquets spécifiques se trouvant dans `experimental` tout en suivant `unstable`. Vous donnez la liste de toutes les archives nécessaires dans fichier « /etc/apt/sources.list » de la manière suivante :

```
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
```

La valeur de Pin-Priority par défaut pour l'archive `experimental` est toujours de 1 (<<100) car c'est une archive **non automatique** (consultez Section 2.5.3). Il n'y a pas besoin de définir explicitement de valeur de Pin-Priority dans le fichier « /etc/apt/preferences » simplement pour utiliser l'archive `experimental` à moins que vous ne désiriez suivre des paquets particuliers dans cette archive de manière automatique pour la mise à niveau suivante.

2.7.8 Chargement et mise à niveau automatique de paquets

**AVERTISSEMENT**

L'utilisation de l'épinglage apt par un débutant est une source certaine de problèmes majeurs. Vous devriez éviter l'utilisation de l'épinglage apt sauf si vous en avez absolument besoin.

Le paquet `apt` est diffusé avec son propre script d'événements planifiés (cron) « `/etc/cron.daily/apt` » afin de gérer le téléchargement automatique de paquets. Ce script peut être amélioré afin d'effectuer la mise à niveau automatique des paquets en installant le paquet `unattended-upgrades`. Cela peut être personnalisé à l'aide de paramètres se trouvant dans « `/etc/apt/apt.conf.d/02backup` » et « `/etc/apt/apt.conf.d/50unattended-upgrades` » comme décrit dans « `/usr/share/doc/unattended-upgrades/README` ».

Le paquet `unattended-upgrades` est principalement destiné à des mises à jour de sécurité des systèmes `stable`. Si le risque de casser un système `stable` existant par la mise à niveau automatique est plus faible que celui d'avoir un système cassé par un intrus utilisant une de ses failles de sécurité qui a été fermée par une mise à jour de sécurité, vous devriez envisager d'utiliser cette mise à niveau automatique avec les paramètres de configuration suivants :

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

Si vous faites tourner un système `unstable`, vous ne devriez pas utiliser les mises à niveau automatiques car cela cassera probablement votre système un jour ou l'autre. Même dans ce cas, avec `unstable`, vous pourrez télécharger des paquets à l'avance afin de gagner du temps pour la mise à niveau interactive avec les paramètres de configuration suivants :

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

2.7.9 Diminuer la bande passante utilisée par APT

Si vous désirez limiter la bande passante utilisée par APT à, par exemple, 800Kib/sec (=100kio/sec), vous devrez configurer APT avec son paramètre de configuration comme suit :

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

2.7.10 Retour d'urgence à une version précédente (downgrade)



AVERTISSEMENT

L'utilisation de l'épinglage `apt` par un débutant est une source certaine de problèmes majeurs. Vous devriez éviter l'utilisation de l'épinglage `apt` sauf si vous en avez absolument besoin.



Attention

Le retour vers une version antérieure n'est pas officiellement géré par Debian dans sa conception. Ce ne devrait être fait qu'en tant que partie d'un processus de récupération d'urgence. Malgré cette situation, on sait que cela fonctionne bien pour de nombreux incidents. Avec les systèmes critiques vous devrez effectuer une sauvegarde des données importantes du système après l'opération de récupération et réinstaller le nouveau système depuis le départ.

Vous pouvez être assez chanceux pour revenir en arrière depuis une archive plus récente vers une archive plus ancienne afin de récupérer une mise à jour du système en manipulant la **version candidate** (consultez Section 2.7.3). Cette méthode est un remplacement de paresseux des nombreuses et fastidieuses commandes « `dpkg -i paquet-cassé_ancienne-version.deb` » (consultez Section 2.6.4).

Recherchez les lignes du fichier « `/etc/apt/sources.list` » permettant de suivre `unstable` ayant la forme suivante :

```
deb http://deb.debian.org/debian/ sid main contrib non-free
```

Remplacez-la avec la suivante pour suivre `testing` :

```
deb http://deb.debian.org/debian/ bookworm main contrib non-free
```

Configurez le fichier « `/etc/apt/preferences` » comme suit :

```
Package: *  
Pin: release a=testing  
Pin-Priority: 1010
```

Lancez « `apt-get update; apt-get dist-upgrade` » pour forcer l'installation à une version antérieure des paquets du système.

Supprimez ce fichier spécial « `/etc/apt/preferences` » après ce retour en arrière d'urgence.

ASTUCE

C'est une bonne idée de supprimer (sans purger) autant de paquets que possible afin de limiter les problèmes de dépendances. Vous devrez peut-être supprimer et installer manuellement un certain nombre de paquets afin de remettre le système dans un état antérieur. Le noyau de Linux, le gestionnaire d'amorçage, udev, PAM, APT, et les paquets relatifs au réseau ainsi que leurs fichiers de configuration demandent une attention particulière.

2.7.11 Qui a envoyé le paquet ?

Bien que le nom du responsable figure dans « `/var/lib/dpkg/available` » et « `/usr/share/doc/nom_paquet/changes` », procure quelques informations sur « qui se trouve derrière l'activité de construction des paquets », celui qui a réellement envoyé le paquet est un peu obscur. `who-uploads(1)` dans le paquet `devscripts` identifie celui qui a réellement envoyé les paquets sources Debian.

2.7.12 Paquet `equivs`

Si vous devez compiler un programme à partir de ses sources pour remplacer un paquet Debian, le mieux est d'en faire un paquet local réellement « debianisé » (`*.deb`) et d'utiliser une archive privée.

Si vous choisissez de compiler un programme depuis ses sources et de l'installer plutôt sous « `/usr/local` », vous pouvez avoir besoin d'utiliser `equivs` en dernier ressort pour satisfaire les dépendances des paquets manquants.

```
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 Porter un paquet vers le système stable

Pour des mises à niveau partielles du système stable, il est souhaitable de reconstruire un paquet dans son environnement en utilisant le paquet source. Cela évite des mises à niveau massives de paquets en raison de leurs dépendances.

Ajoutez les entrées suivantes au fichier « `/etc/apt/sources.list` » d'un système stable :

```
deb-src http://deb.debian.org/debian unstable main contrib non-free
```

Installez les paquets nécessaires à la compilation et téléchargez les sources comme suit :

```
# 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*
```

Mettez à jour certains paquets de la chaîne d'outils tels que `dpkg`, et `debhelper` à partir des paquets rétroportés s'ils sont requis pour le rétroportage.

Exécutez ce qui suit :

```
$ dch -i
```

Incrémentez la version du paquet, en ajoutant, par exemple « `+bp1` » dans « `debian/changelog` »

Construisez les paquets et installez-les sur le système en faisant ce qui suit :

```
$ debuild
$ cd ..
# debi foo*.changes
```

2.7.14 Serveur mandataire (proxy) pour APT

Comme effectuer le miroir complet d'une sous-section d'une archive Debian gaspille de l'espace disque et de la bande passante du réseau, il est souhaitable, lorsque vous administrez de nombreux systèmes sur le LAN, de mettre en œuvre un serveur mandataire (« proxy ») local pour APT. APT peut être configuré pour utiliser un serveur mandataire web (http) générique comme `squid` (consultez Section 6.5) tel que décrit dans `apt.conf(5)` et dans « `/usr/share/doc/apt/examples/configure-index.gz` ». La variable d'environnement « `$http_proxy` » peut être utilisée pour outrepasser le serveur mandataire défini dans le fichier « `/etc/apt/apt.conf` ».

Il y a des outils de proxy spécifiques pour l'archive Debian. Vous devriez consulter le BTS avant de les installer.

paquet	popcon	taille	description
approx	V:0, I:0	6317	serveur proxy avec cache pour les fichiers de l'archive Debian (programme OCaml compilé)
apt-cacher	V:0, I:0	289	proxy avec cache pour les paquets et les fichiers source Debian (programme Perl)
apt-cacher-ng	V:5, I:5	1562	proxy avec cache pour la distribution de paquets de logiciel (programme C++ compilé)

Table 2.19 – Liste des outils de proxy spécifiques à l'archive Debian



Attention

Lors que Debian réorganise la structure de son archive, ces outils de proxy spécialisés ont tendance à exiger que le code soit réécrit par le responsable du paquet et peuvent ne plus fonctionner pendant un certain temps. D'un autre côté, les serveurs mandataires web (http) génériques sont plus robustes et s'accommodent plus facilement de tels changements.

2.7.15 Autres lectures concernant la gestion des paquets

Vous pouvez en apprendre davantage sur la gestion des paquets dans les documentations suivantes :

- Documentations primaires sur la gestion des paquets :
 - `aptitude(8)`, `dpkg(1)`, `tasksel(8)`, `apt(8)`, `apt-get(8)`, `apt-config(8)`, `apt-key(8)`, `sources.list(5)`, `apt.conf(5)`, and `apt_preferences(5)`;
 - « `/usr/share/doc/apt-doc/guide.html/index.html` » et « `/usr/share/doc/apt-doc/offline.html` » du paquet `apt-doc` ;
 - « `/usr/share/doc/aptitude/html/en/index.html` » du paquet `aptitude-doc-fr`.
- Documentations officielles et détaillées sur l'archive Debian :
 - « [Charte Debian, chapitre 2 - L'archive Debian](#) »,
 - « [Manuel de référence du développeur Debian, chapitre 4 - Ressources pour les développeurs Debian 4.6 L'archive Debian](#) » et
 - « [FAQ de Debian GNU/Linux, chapitre 6 - Les archives FTP Debian](#) ».
- Didacticiel pour la construction d'un paquet Debian pour les utilisateurs Debian :
 - « [<Guide des responsables Debian>](#) ».

Chapitre 3

Initialisation du système

En tant qu'administrateur du système, il est sage que vous sachiez en gros comment le système Debian est démarré et configuré. Bien que les détails exacts figurent dans les fichiers sources des paquets installés et dans leurs documentations, c'est un peu pénible pour la plupart d'entre-nous.

Here is a rough overview of the key points of the Debian system initialization. Since the Debian system is a moving target, you should refer to the latest documentation.

- [Debian Linux Kernel Handbook](#) is the primary source of information on the Debian kernel.
- `bootup(7)` describes the system bootup process based on `systemd`. (Recent Debian)
- `boot(7)` describes the system bootup process based on UNIX System V Release 4. (Older Debian)

3.1 Aperçu du processus d'amorçage du système

Le système informatique subit plusieurs phases de [processus d'amorçage](#) (« boot strap process ») depuis l'événement de mise sous tension jusqu'à ce qu'il offre à l'utilisateur un système d'exploitation (OS) pleinement fonctionnel.

Pour des raisons de simplicité, je limiterai la discussion à une plateforme PC typique avec l'installation par défaut.

Le processus d'amorçage typique est comme une fusée à quatre étages. Chaque étage de la fusée passe le contrôle du système à l'étage suivant.

- Section [3.1.1](#)
- Section [3.1.2](#)
- Section [3.1.3](#)
- Section [3.1.4](#)

Bien entendu, elles peuvent être configurées de manière différente. Par exemple, si vous avez compilé votre propre noyau, vous pouvez sauter l'étape avec le système mini-Debian. Ne supposez donc pas que c'est le cas sur votre système avant de l'avoir vérifié vous-même.

3.1.1 Stage 1: the UEFI

The [Unified Extensible Firmware Interface \(UEFI\)](#) defines a boot manager as part of the UEFI specification. When a computer is powered on, the boot manager is the 1st stage of the boot process which checks the boot configuration and based on its settings, then executes the specified OS boot loader or operating system kernel (usually boot loader). The boot configuration is defined by variables stored in NVRAM, including variables that indicate the file system paths to OS loaders or OS kernels. An [EFI system partition \(ESP\)](#) is a data storage device partition that is used in computers adhering to the UEFI specification. Accessed by the UEFI firmware when a computer is powered up, it stores UEFI applications and the files these applications need to run, including operating system boot loaders. (On the legacy PC system, [BIOS](#) stored in the [MBR](#) may be used instead.)

3.1.2 Étage 2 : le chargeur initial

The **boot loader** is the 2nd stage of the boot process which is started by the UEFI. It loads the system kernel image and the **initrd** image to the memory and hands control over to them. This initrd image is the root filesystem image and its support depends on the bootloader used.

The Debian system normally uses the Linux kernel as the default system kernel. The initrd image for the current 5.x Linux kernel is technically the **initramfs** (initial RAM filesystem) image.

There are many boot loaders and configuration options available.

paquet	popcon	taille	initrd	chargeur initial	description
grub-efi-amd64	I:214	158	Pris en charge	GRUB UEFI	This is smart enough to understand disk partitions and filesystems such as vfat, ext4, ... (UEFI)
grub-pc	V:26, I:757	533	Pris en charge	GRUB 2	This is smart enough to understand disk partitions and filesystems such as vfat, ext4, ... (BIOS)
grub-rescue-pc	V:0, I:1	6380	Pris en charge	GRUB 2	images de secours amorçables de GRUB 2 (CD et disquettes) (versions PC/BIOS)
lilo	V:0, I:2	697	Pris en charge	Lilo	réside sur les emplacements des secteurs de données du disque dur (Ancien).
syslinux	V:3, I:46	344	Pris en charge	Isolinux	il comprend le système de fichiers ISO9660. C'est utilisé pour le CD d'amorçage.
syslinux	V:3, I:46	344	Pris en charge	Syslinux	il comprend le système de fichiers MSDOS (FAT) . Il est utilisé par la disquette d'amorçage.
loadlin	V:0, I:1	90	Pris en charge	Loadlin	un nouveau système est démarré depuis le système FreeDOS/MSDOS.
mbr	V:0, I:6	50	Non pris en charge	MBR par Neil Turton	c'est un logiciel libre qui se substitue au MBR de MSDOS. Il ne comprend que les partitions sur disques.

Table 3.1 – Liste des chargeurs initiaux



AVERTISSEMENT

Do not play with boot loaders without having bootable rescue media (USB memory stick, CD or floppy) created from images in the `grub-rescue-pc` package. It makes you boot your system even without functioning bootloader on the hard disk.

For GRUB 2, the menu configuration file is located at `"/boot/grub/grub.cfg"` and its key part of menu entry looks like:

```
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
}
```

For this part of `/boot/grub/grub.cfg`, this menu entry means the following.

setting	valeur
GRUB2 modules loaded	gzio, part_gpt, ext2
root file system partition used	partition identified by UUID=fe3e1db5-6454-46d6-a14c-071208ebe4b1
kernel image path in the root file system	/boot/vmlinuz-5.10.0-6-amd64
kernel boot parameter used	"root=UUID=fe3e1db5-6454-46d6-a14c-071208ebe4b1 ro quiet"
initrd image path in the root file system	/boot/initrd.img-5.10.0-6-amd64

Table 3.2 – The meaning of the menu entry of the above part of `/boot/grub/grub.cfg`

ASTUCE

You can customize GRUB splash image by setting `GRUB_BACKGROUND` variable in `/etc/default/grub` pointing to the image file or placing the image file itself in `/boot/grub/`.

Consultez « `info grub` » et `grub-install(8)`.

3.1.3 Étage 3 : le système mini-Debian

Le système mini-Debian est la troisième étape du processus d'amorçage lancée par le chargeur d'amorçage. Elle lance le noyau du système avec son système de fichiers racine en mémoire. C'est une étape préparatoire facultative du processus de démarrage.

Note

Le terme « système mini-Debian » est utilisé par l'auteur pour décrire dans ce document cette 3ème étape du processus de démarrage. On désigne souvent ce système par système `initrd`. Un système semblable en mémoire est utilisé par [l'installateur Debian](#).

The `"/init"` program is executed as the first program in this root filesystem on the memory. It is a program which initializes the kernel in user space and hands control over to the next stage. This mini-Debian system offers flexibility to the boot process such as adding kernel modules before the main boot process or mounting the root filesystem as an encrypted one.

- The `"/init"` program is a shell script program if `initramfs` was created by `initramfs-tools`.
 - Vous pouvez interrompre cette partie du processus d'amorçage afin d'obtenir l'invite de l'interpréteur de l'administrateur en indiquant « `break=init` » etc. comme paramètre de démarrage du noyau. Consultez le script « `/init` » pour d'autres conditions d'interruption. Cet environnement d'interpréteur de commandes est suffisamment sophistiqué pour effectuer une bonne inspection du matériel de votre machine.
 - Les commandes disponibles avec ce système mini-Debian sont des commandes réduites et sont principalement fournies par un outil GNU appelé `busybox(1)`.
- The `"/init"` program is a binary `systemd` program if `initramfs` was created by `dracut`.
 - Commands available in this mini-Debian system are stripped down `systemd(1)` environment.



Attention

Vous devrez utiliser l'option « `-n` » de la commande `mount` lorsque vous êtes sur le système de fichiers en lecture seule.

3.1.4 Étape 4 : le système Debian normal

Le système Debian normal est le quatrième étage du processus d'amorçage, il est lancé par le système mini-Debian. Le noyau du système mini-Debian continue de tourner dans cet environnement. Le système de fichiers racine passe de celui en mémoire à celui, réel, lu sur le disque dur.

Le programme `init` est le premier à être exécuté, assorti du `PID=1`, afin qu'il accomplisse son rôle de processus principal du démarrage, qui consiste à commander l'exécution de plusieurs programmes. Le chemin par défaut du programme `init` est « `/sbin/init` » mais il peut être modifié en passant un paramètre de démarrage au noyau, comme suit : « `init=/chemin/vers/programme_init` »
`"/sbin/init" is symlinked to "/lib/systemd/systemd" after Debian 8 Jessie (released in 2015).`

ASTUCE

Il est possible de vérifier le niveau d'exécution courant de la commande `init` du système avec la commande « `ps --pid 1 -f` ».

paquet	popcon	taille	description
systemd	V:826, I:928	16015	démon <code>init(8)</code> basé sur des événements pour la concurrence (remplacement de <code>sysvinit</code>)
systemd-sysv	V:818, I:925	139	the manual pages and links needed for <code>systemd</code> to replace <code>sysvinit</code>
init-system-helpers	V:687, I:940	131	helper tools for switching between <code>sysvinit</code> and <code>systemd</code>
initscripts	V:77, I:283	176	scripts pour initialiser et arrêter le système
sysvinit-core	V:7, I:9	279	Utilitaires <code>init(8)</code> de type System-V
sysv-rc	V:154, I:296	81	Mécanisme de changement de niveau de fonctionnement de type System-V
sysvinit-utils	V:459, I:999	79	Utilitaires de type System-V (<code>startpar(8)</code> , <code>bootlogd(8)</code> , ...)
lsb-base	V:890, I:999	49	Linux Standard Base 3.2 fonctionnalité de script d'initialisation
insserv	V:204, I:292	150	outil pour organiser la séquence de démarrage en utilisant les dépendances du script LSB de <code>init.d</code>
uswsusp	V:3, I:7	714	outil permettant d'utiliser le programme de l'espace utilisateur de mise en veille fourni par Linux
kexec-tools	V:1, I:9	288	outil <code>kexec</code> pour le redémarrage par <code>kexec(8)</code> (redémarrage à chaud)
systemd-bootchart	V:0, I:1	128	analyseur des performances du processus de démarrage
bootchart2	V:0, I:0	94	analyseur des performances du processus de démarrage
pybootchartgui	V:0, I:0	177	analyseur des performances du processus de démarrage (virtualisation)
mingetty	V:0, I:3	38	<code>getty(8)</code> en mode console uniquement
mgetty	V:0, I:1	315	modem intelligent (« smart modem ») remplaçant de <code>getty(8)</code>

Table 3.3 – Liste d'utilitaires d'amorçage initial pour le système Debian :

ASTUCE

Vous trouverez des conseils actualisés pour accélérer le processus de démarrage sur [Debian wiki:BootProcessSpeedup](#).

3.2 Systemd init

This section describes how system is started by the `systemd(1)` program with `PID=1` (i.e., `init` process).

The `systemd` init process spawns processes in parallel based on the unit configuration files (see `systemd.unit(5)`) which are written in declarative style instead of SysV-like procedural style.

The spawned processes are placed in individual [Linux control groups](#) named after the unit which they belong to in the private `systemd` hierarchy (see [cgroups](#) and Section 4.7.4).

The unit configuration files are loaded from a set of paths (see `systemd-system.conf(5)`) as follows:

- `"/lib/systemd/system"`: OS default configuration files
- `"/etc/systemd/system"`: system administrator configuration files which override the OS default configuration files
- `"/run/systemd/system"`: run-time generated configuration files which override the installed configuration files

Their inter-dependencies are specified by the directives `"Wants="`, `"Requires="`, `"Before="`, `"After="`, ... (see "MAPPING OF UNIT PROPERTIES TO THEIR INVERSES" in `systemd.unit(5)`). The resource controls are also defined (see `systemd.resource-control(5)`).

The suffix of the unit configuration file encodes their types as:

- `*.service` describes the process controlled and supervised by `systemd`. See `systemd.service(5)`.
- `*.device` describes the device exposed in the `sysfs(5)` as `udev(7)` device tree. See `systemd.device(5)`.
- `*.mount` describes the file system mount point controlled and supervised by `systemd`. See `systemd.mount(5)`.
- `*.automount` describes the file system auto mount point controlled and supervised by `systemd`. See `systemd.automount(5)`.
- `*.swap` describes the swap device or file controlled and supervised by `systemd`. See `systemd.swap(5)`.
- `*.path` describes the path monitored by `systemd` for path-based activation. See `systemd.path(5)`.
- `*.socket` describes the socket controlled and supervised by `systemd` for socket-based activation. See `systemd.socket(5)`.
- `*.timer` describes the timer controlled and supervised by `systemd` for timer-based activation. See `systemd.timer(5)`.
- `*.slice` manages resources with the `cgroups(7)`. See `systemd.slice(5)`.
- `*.scope` is created programmatically using the bus interfaces of `systemd` to manages a set of system processes. See `systemd.scope(5)`.
- `*.target` groups other unit configuration files to create the synchronization point during start-up. See `systemd.target(5)`.

Upon system start up (i.e., init), the `systemd` process tries to start the `"/lib/systemd/system/default.target` (normally symlinked to `"graphical.target"`). First, some special target units (see `systemd.special(7)`) such as `"local-fs.target"`, `"swap.target"` and `"cryptsetup.target"` are pulled in to mount the filesystems. Then, other target units are also pulled in by the target unit dependencies. For details, read `bootup(7)`.

`systemd` offers backward compatibility features. SysV-style boot scripts in `"/etc/init.d/rc[0123456S].d/[KS]name"` are still parsed and `telinit(8)` is translated into `systemd` unit activation requests.



Attention

Emulated runlevel 2 to 4 are all symlinked to the same `"multi-user.target"`.

3.2.1 Nom de machine (« hostname »)

The kernel maintains the system **hostname**. The system unit started by `systemd-hostnamed.service` sets the system hostname at boot time to the name stored in `"/etc/hostname"`. This file should contain **only** the system hostname, not a fully qualified domain name.

Pour afficher le nom de la machine utilisée, lancez la commande `hostname` (1) sans paramètre.

3.2.2 Le système de fichiers

The mount options of normal disk and network filesystems are set in `/etc/fstab`. See `fstab(5)` and Section 9.6.7.

The configuration of the encrypted filesystem is set in `/etc/crypttab`. See `crypttab(5)`

The configuration of software RAID with `mdadm(8)` is set in `/etc/mdadm/mdadm.conf`. See `mdadm.conf(5)`.



AVERTISSEMENT

Une fois tous les systèmes de fichiers montés, les fichiers temporaires se trouvant dans « `/tmp` », « `/var/lock` » et « `/var/run` » sont effacés lors de chaque démarrage du système.

3.2.3 Initialisation de l'interface réseau

Network interfaces are typically initialized in `networking.service` for the `lo` interface and `NetworkManager.service` for other interfaces on modern Debian desktop system under `systemd`.

Voir Chapitre 5 pour savoir comment les configurer.

3.3 Messages du noyau

The kernel error message displayed to the console can be configured by setting its threshold level.

```
# dmesg -n3
```

valeur du niveau d'erreur	nom du niveau d'erreur	signification
0	KERN_EMERG	le système est inutilisable
1	KERN_ALERT	une action doit être entreprise immédiatement
2	KERN_CRIT	conditions critiques
3	KERN_ERR	conditions d'erreur
4	KERN_WARNING	conditions d'avertissement
5	KERN_NOTICE	condition normale mais significative
6	KERN_INFO	information
7	KERN_DEBUG	messages du niveau de débogage

Table 3.4 – Liste des niveaux d'erreur du noyau

3.4 Messages du système

Under `systemd`, both kernel and system messages are logged by the journal service `systemd-journald.service` (a.k.a `journald`) either into a persistent binary data below `/var/log/journal` or into a volatile binary data below `/run/log/journal`. These binary log data are accessed by the `journalctl(1)` command. For example, you can display log from the last boot as:

```
$ journalctl -b
```

Under `systemd`, the system logging utility `rsyslogd(8)` may be uninstalled. If it is installed, it changes its behavior to read the volatile binary log data (instead of pre-`systemd` default `/dev/log`) and to create traditional permanent ASCII system log data. This can be customized by `/etc/default/rsyslog` and `/etc/rsyslog.conf` for both the log file and on-screen display. See `rsyslogd(8)` and `rsyslog.conf(5)`. See also Section 9.3.2.

Operation	Command snippets
View log for system services and kernel from the last boot	<code>"journalctl -b --system"</code>
View log for services of the current user from the last boot	<code>"journalctl -b --user"</code>
View job log of "\$unit" from the last boot	<code>"journalctl -b -u \$unit"</code>
View job log of "\$unit" ("tail -f" style) from the last boot	<code>"journalctl -b -u \$unit -f"</code>

Table 3.5 – List of typical `journalctl` command snippets

3.5 System management

The `systemd` offers not only init system but also generic system management operations with the `systemctl(1)` command.

Here, "\$unit" in the above examples may be a single unit name (suffix such as `.service` and `.target` are optional) or, in many cases, multiple unit specifications (shell-style globs `"**"`, `"?"`, `"["`] using `fnmatch(3)` which will be matched against the primary names of all units currently in memory).

System state changing commands in the above examples are typically preceded by the `"sudo"` to attain the required administrative privilege.

The output of the `"systemctl status $unit| $PID| $device"` uses color of the dot ("**●**") to summarize the unit state at a glance.

- White "**●**" indicates an "inactive" or "deactivating" state.
- Red "**●**" indicates a "failed" or "error" state.
- Green "**●**" indicates an "active", "reloading" or "activating" state.

3.6 Other system monitors

Here are a list of other monitoring command snippets under `systemd`. Please read the pertinent manpages including `cgroups(7)`.

3.7 Customizing systemd

3.7.1 Socket activation

With default installation, many network services (see Chapitre 6) are started as daemon processes after `network.target` at boot time by `systemd`. The `"sshd"` is no exception. Let's change this to on-demand start of `"sshd"` as a customization example.

First, disable system installed service unit.

```
$ sudo systemctl stop sshd.service
$ sudo systemctl mask sshd.service
```

The on-demand socket activation system of the classic Unix services was through the `inetd` (or `xinetd`) superserver. Under `systemd`, the equivalent can be enabled by adding `*.socket` and `*.service` unit configuration files.

`sshd.socket` for specifying a socket to listen on

```
[Unit]
Description=SSH Socket for Per-Connection Servers

[Socket]
```

Operation	Command snippets
List all target unit configuration	<code>"systemctl list-units --type=target"</code>
List all service unit configuration	<code>"systemctl list-units --type=service"</code>
List all unit configuration types	<code>"systemctl list-units --type=help"</code>
List all socket units in memory	<code>"systemctl list-sockets"</code>
List all timer units in memory	<code>"systemctl list-timers"</code>
Start "\$unit"	<code>"systemctl start \$unit"</code>
Stop "\$unit"	<code>"systemctl stop \$unit"</code>
Reload service-specific configuration	<code>"systemctl reload \$unit"</code>
Stop and start all "\$unit"	<code>"systemctl restart \$unit"</code>
Start "\$unit" and stop all others	<code>"systemctl isolate \$unit"</code>
Switch to "graphical" (GUI system)	<code>"systemctl isolate graphical"</code>
Switch to "multi-user" (CLI system)	<code>"systemctl isolate multi-user"</code>
Switch to "rescue" (single user CLI system)	<code>"systemctl isolate rescue"</code>
Send kill signal to "\$unit"	<code>"systemctl kill \$unit"</code>
Check if "\$unit" service is active	<code>"systemctl is-active \$unit"</code>
Check if "\$unit" service is failed	<code>"systemctl is-failed \$unit"</code>
Check status of "\$unit \$PID device"	<code>"systemctl status \$unit \$PID \$device"</code>
Show properties of "\$unit \$job"	<code>"systemctl show \$unit \$job"</code>
Reset failed "\$unit"	<code>"systemctl reset-failed \$unit"</code>
List dependency of all unit services	<code>"systemctl list-dependencies --all"</code>
List unit files installed on the system	<code>"systemctl list-unit-files"</code>
Enable "\$unit" (add symlink)	<code>"systemctl enable \$unit"</code>
Disable "\$unit" (remove symlink)	<code>"systemctl disable \$unit"</code>
Unmask "\$unit" (remove symlink to "/dev/null")	<code>"systemctl unmask \$unit"</code>
Mask "\$unit" (add symlink to "/dev/null")	<code>"systemctl mask \$unit"</code>
Get default-target setting	<code>"systemctl get-default"</code>
Set default-target to "graphical" (GUI system)	<code>"systemctl set-default graphical"</code>
Set default-target to "multi-user" (CLI system)	<code>"systemctl set-default multi-user"</code>
Show job environment	<code>"systemctl show-environment"</code>
Set job environment "variable" to "value"	<code>"systemctl set-environment variable=value"</code>
Unset job environment "variable"	<code>"systemctl unset-environment variable"</code>
Reload all unit files and daemons	<code>"systemctl daemon-reload"</code>
Shut down the system	<code>"systemctl poweroff"</code>
Shut down and reboot the system	<code>"systemctl reboot"</code>
Suspend the system	<code>"systemctl suspend"</code>
Hibernate the system	<code>"systemctl hibernate"</code>

Table 3.6 – List of typical `systemctl` command snippets

Operation	Command snippets
Show time spent for each initialization steps	"systemd-analyze time"
List of all units by the time to initialize	"systemd-analyze blame"
Load and detect errors in "\$unit" file	"systemd-analyze verify \$unit"
Show terse runtime status information of the user of the caller's session	"loginctl user-status"
Show terse runtime status information of the caller's session	"loginctl session-status"
Track boot process by the cgroups	"systemd-cgls"
Track boot process by the cgroups	"ps xawf -eo pid,user,cgroup,args"
Track boot process by the cgroups	Read sysfs under "/sys/fs/cgroup/systemd/"

Table 3.7 – List of other monitoring command snippets under systemd

```
ListenStream=22
Accept=yes
```

```
[Install]
WantedBy=sockets.target
```

sshd@.service as the matching service file of sshd.socket

```
[Unit]
Description=SSH Per-Connection Server
```

```
[Service]
ExecStart=-/usr/sbin/sshd -i
StandardInput=socket
```

Then reload.

```
$ sudo systemctl daemon-reload
```

3.8 Le système udev

The [udev system](#) provides mechanism for the automatic hardware discovery and initialization (see [udev\(7\)](#)) since Linux kernel 2.6. Upon discovery of each device by the kernel, the udev system starts a user process which uses information from the [sysfs](#) filesystem (see Section [1.2.12](#)), loads required kernel modules supporting it using the [modprobe\(8\)](#) program (see Section [3.8.1](#)), and creates corresponding device nodes.

ASTUCE

Si « /lib/modules/kernel-version/modules.dep » n'a pas été proprement créé par [depmod\(8\)](#) pour quelque raison, les modules peuvent ne pas être chargés par le système udev comme on le souhaiterait. Lancez « [depmod -a](#) » pour corriger ce problème.

Les nœuds de périphériques n'ont pas besoin d'être statiques pour les règles de montage se trouvant dans « /etc/fstab ». Vous pouvez utiliser [UUID](#) à la place de leur nom de périphérique tel que « /dev/sda » pour monter les périphériques. Consultez Section [9.6.3](#).

Comme le système udev est une cible quelque peu mouvante, je laisse les détails pour d'autres documentations et je ne donnerai ici qu'un minimum d'informations.

3.8.1 Initialisation des modules du noyau

Le programme `modprobe(8)` nous permet de configurer, depuis un processus utilisateur, un noyau Linux en cours d'exécution en ajoutant ou en supprimant des modules du noyau. Le système `udev` (consultez Section 3.8) en automatise l'appel afin d'aider à l'initialisation du module du noyau.

Il existe des modules non liés au matériel et des modules qui pilotent des éléments matériels particuliers comme les suivants qui demandent à être préchargés en les déclarant dans le fichier « `/etc/modules` » (consultez `modules(5)`).

- les modules `TUN/TAP` fournissent un périphérique de réseau virtuel point-à-point (TUN) et un périphérique de réseau virtuel Ethernet (TAP) ;
- les modules `netfilter` fournissent les fonctions de pare-feu `netfilter` (`iptables(8)`, Section 5.6) ;
- le module du pilote du `temporisateur de chien de garde`.

Les fichiers de configuration du programme `modprobe(8)` se trouvent dans le répertoire « `/etc/modprobes.d/` » comme c'est expliqué dans `modprobe.conf(5)`. (Si vous souhaitez que certains modules du noyau ne soient pas chargés automatiquement, vous pouvez les mettre en liste noire dans le fichier « `/etc/modprobes.d/blacklist` »).

Le fichier « `/lib/modules/version/modules.dep` » généré par le programme `depmod(8)` décrit les dépendances des modules utilisés par le programme `modprobe(8)`.

Note

Si vous rencontrez des problèmes de chargement de modules lors du chargement des modules au démarrage ou avec `modprobe(8)`, « `depmod -a` » peut résoudre ces problèmes en reconstruisant « `modules.dep` ».

Le programme `modinfo(8)` affiche des informations concernant les modules du noyau.

Le programme `lsmod(8)` formate de manière agréable le contenu de « `/proc/modules` », affichant quels sont les modules du noyau actuellement chargés.

ASTUCE

Vous pouvez identifier le matériel exact installé sur votre système. Consultez Section 9.5.3.

Vous pouvez configurer le matériel au moment du démarrage pour activer les fonctionnalités désirées de ce matériel. Consultez Section 9.5.4.

Vous pouvez probablement ajouter la prise en charge d'un périphérique particulier en recompilant le noyau. Consultez Section 9.10.

Chapitre 4

Authentication and access controls

Lorsqu'une personne (ou un programme) demande l'accès au système, l'authentification confirme que l'identité est autorisée.



AVERTISSEMENT

Des erreurs de configuration de PAM peuvent vous mettre à la porte de votre propre système. Vous devez avoir un CD de secours prêt ou une partition de démarrage de remplacement. Pour restaurer, démarrez le système depuis l'un de ces moyens de secours et corrigez les choses depuis là.

4.1 Authentification normale d'UNIX

L'authentification normale d'UNIX est fournie par le module `pam_unix(8)` avec [PAM \(Pluggable Authentication Modules : « Modules attachables d'authentification »\)](#). Il y a trois fichiers de configuration importants, dont les entrées sont séparées par des « : », ce sont :

fichier	autorisation	utilisateur	groupe	description
/etc/passwd	-rw-r--r--	root	root	informations des comptes utilisateurs (assainie)
/etc/shadow	-rw-r-----	root	shadow	informations sécurisées des comptes utilisateurs
/etc/group	-rw-r--r--	root	root	informations des groupes

Table 4.1 – 3 fichiers de configuration importants pour `pam_unix(8)`

« /etc/passwd » contient ce qui suit :

```
...
user1:x:1000:1000:User1 Name,,,:/home/user1:/bin/bash
user2:x:1001:1001:User2 Name,,,:/home/user2:/bin/bash
...
```

Comme il est expliqué dans `passwd(5)`, les entrées de ce fichier, séparées par des « : », ont la signification suivante :

- nom de l'utilisateur pour la connexion ;
- entrée de spécification du mot de passe ;
- identifiant numérique de l'utilisateur ;
- identifiant numérique du groupe ;
- nom de l'utilisateur ou champ de commentaire ;

- répertoire personnel de l'utilisateur ;
- interpréteur de commandes, optionnel, de l'utilisateur.

La seconde entrée de « `/etc/passwd` » était autrefois utilisée comme entrée de mot de passe chiffré. Depuis l'introduction de « `/etc/shadow` », cette entrée est utilisée comme entrée de spécification du mot de passe.

contenu	signification
(vide)	compte sans mot de passe
x	le mot de passe chiffré se trouve dans « <code>/etc/shadow</code> »
*	ce compte ne peut pas se connecter
!	ce compte ne peut pas se connecter

Table 4.2 – Contenu de la seconde entrée de « `/etc/passwd` »

« `/etc/shadow` » contient ceci :

```
...
user1:$1$Xop0FYH9$IfxyQwBe9b8tiyIkt2P4F/:13262:0:99999:7:::
user2:$1$vXGZLVbS$ElyErNf/agUDsm1DehJMS/:13261:0:99999:7:::
...
```

Comme c'est expliqué dans `shadow(5)`, les différentes entrées de ce fichier, séparées par des « `:` » ont les significations suivantes :

- nom de l'utilisateur pour la connexion ;
- mot de passe chiffré (le « `1` » du début indique l'utilisation d'un chiffrement MD5. Le signe « `*` » indique que le compte ne peut pas se connecter) ;
- Date du dernier changement de mot de passe, exprimé en nombre de jours passés depuis le premier janvier 1970.
- Nombre de jours avant qu'un utilisateur ne soit autorisé à changer à nouveau son mot de passe.
- Nombre de jours avant que l'utilisateur ne soit tenu de changer son mot de passe.
- nombre de jours avant qu'un mot de passe n'arrive à expiration et durant lesquels l'utilisateur doit être averti ;
- Nombre de jours durant lesquels un mot de passe devrait encore être accepté, passé sa date d'expiration.
- Date d'expiration du compte, exprimée en nombre de jours passés depuis le premier janvier 1970.
- ...

« `/etc/group` » contient ce qui suit :

```
group1:x:20:user1,user2
```

Comme il est expliqué dans `group(5)`, les entrées de ce fichier, séparées par des « `:` », ont la signification suivante :

- nom du groupe ;
- mot de passe chiffré (non utilisé en pratique) ;
- identifiant numérique du groupe ;
- liste des noms d'utilisateurs séparés par des « `,` ».

Note

« `/etc/gshadow` » fournit les mêmes fonctions que « `/etc/shadow` » pour « `/etc/group` » mais n'est pas réellement utilisé.

Note

Le groupe d'appartenance réel d'un utilisateur peut être ajouté dynamiquement si la ligne « `auth optional pam_group.so` » est ajoutée à « `/etc/pam.d/common-auth` » et défini dans « `/etc/security/group.conf` ». Consultez `pam_group(8)`.

Note

Le paquet `base-passwd` contient une liste faisant autorité d'utilisateurs et de groupes :
« `/usr/share/doc/base-passwd/users-and-groups.html` ».

4.2 Gestion des informations des comptes et des mots de passes

Voici quelques commandes importantes pour gérer les informations des comptes :

commande	fonction
<code>getent passwd nom_utilisateur</code>	consulter les informations du compte « <i>nom_utilisateur</i> »
<code>getent shadow nom_utilisateur</code>	consulter les informations cachées du compte « <i>nom_utilisateur</i> »
<code>getent group nom_groupe</code>	consulter les informations du groupe « <i><nom_ groupe></i> »
<code>passwd</code>	gérer le mot de passe de ce compte
<code>passwd -e</code>	définir un mot de passe à usage unique pour l'activation du compte
<code>chage</code>	gérer les informations de durée de validité du mot de passe

Table 4.3 – Liste des commandes servant à gérer les informations des comptes

Vous pouvez avoir besoin des droits de l'administrateur pour certaines fonctions. Consultez `crypt(3)` pour le chiffrement des mots de passe et des données.

Note

On the system set up with PAM and NSS as the Debian [salsa](#) machine, the content of local `"/etc/passwd"`, `"/etc/group"` and `"/etc/shadow"` may not be actively used by the system. Above commands are valid even under such environment.

4.3 Mot de passe de qualité

Lors de la création d'un mot de passe à l'installation de votre système ou avec la commande `passwd(1)`, il vous faudra choisir un [bon mot de passe](#) composé d'au moins 6 à 8 caractères, comprenant au moins un des caractères appartenant à l'ensemble suivant conformément à `passwd(1)` :

- caractères alphabétiques en minuscules ;
- chiffres de 0 à 9 ;
- marques de ponctuation.

**AVERTISSEMENT**

Ne choisissez pas des mots qui se devinent aisément pour le mot de passe. Les noms de compte, les numéros de sécurité sociale, les numéros de téléphone, les adresses, les dates anniversaire, les noms des membres de votre famille ou de vos animaux domestiques, les mots du dictionnaire, les suites simples de caractères telles que « 12345 » ou « qwerty »... constituent tous de mauvais choix pour le mot de passe.

4.4 Créer un mot de passe chiffré

Il existe des outils autonomes permettant de [créer des mots de passe chiffrés à partir d'une « semence »](#).

paquet	popcon	taille	commande	fonction
whois	V:32, I:371	364	mkpasswd	frontal de la bibliothèque <code>crypt(3)</code> avec fonctionnalités surabondantes
openssl	V:814, I:994	1465	<code>openssl passwd</code>	calculer le hachage du mot de passe (OpenSSL). <code>passwd(1ssl)</code>

Table 4.4 – Liste d’outils permettant de générer des mots de passe

4.5 PAM et NSS

De nombreux systèmes modernes [semblable à UNIX](#), comme le système Debian, fournissent les mécanismes [PAM \(Pluggable Authentication Modules\)](#) et [NSS \(Name Service Switch\)](#) pour la configuration du système par l’administrateur local. Leur rôle peut être résumé de la manière suivante :

- PAM offre un mécanisme d’authentification souple qui est utilisé par les logiciels applicatifs lorsqu’ils ont besoins d’échanger des mots de passe.
- NSS fournit un mécanisme souple de service de noms qui est fréquemment utilisé par la [bibliothèque standard C](#) pour obtenir le nom de groupe de programmes comme `ls(1)` et `id(1)`.

Ces systèmes PAM et NSS doivent être configurés de manière cohérente.

Les paquets importants des systèmes PAM et NSS sont les suivants :

paquet	popcon	taille	description
libpam-modules	V:813, I:999	1048	PAM (« Pluggable Authentication Modules »), Modules attachables d’authentification (service de base)
libpam-ldap	I:12	249	Modules attachables d’authentification permettant l’utilisation d’interfaces LDAP
libpam-cracklib	I:15	117	Modules attachables d’authentification permettant la prise en compte de cracklib
libpam-systemd	V:484, I:869	574	Pluggable Authentication Module to register user sessions for <code>logind</code>
libpam-doc	I:1	1046	Modules attachables d’authentification (documentation en html et texte)
libc6	V:946, I:999	12772	bibliothèque GNU C Library : bibliothèques partagées qui fournissent aussi le service « Name Service Switch »
glibc-doc	I:11	3161	bibliothèque GNU C : pages de manuel
glibc-doc-reference	I:4	12740	bibliothèque GNU C : manuel de référence dans les formats info, pdf et html (non libre)
libnss-mdns	I:508	150	module NSS pour la résolution des noms DNS Multicast
libnss-ldap	I:11	265	module NSS pour l’utilisation de LDAP comme service de nommage
libnss-ldapd	I:14	153	module NSS pour l’utilisation de LDAP en tant que service de nommage (nouveau fork de <code>libnss-ldap</code>)

Table 4.5 – Liste des paquets importants des systèmes PAM et NSS

- « The Linux-PAM System Administrators’ Guide » de `libpam-doc` est essentiel à l’apprentissage de la configuration de PAM.
- La section « System Databases and Name Service Switch » de `glibc-doc-reference` est essentielle pour l’apprentissage de la configuration de NSS.

Note

Vous en trouverez une liste plus complète et actuelle avec la commande « `aptitude search 'libpam-|libnss-'` ». L’acronyme NSS peut aussi signifier « Network Security Service » qui est différent de « Name Service Switch ».

Note

PAM est la manière la plus élémentaire d'initialiser des variables d'environnement pour tous les programmes avec des valeurs par défaut valables pour l'ensemble du système.

Under [systemd](#), `libpam-systemd` package is installed to manage user logins by registering user sessions in the `systemd` control group hierarchy for [logind](#). See `systemd-logind(8)`, `logind.conf(5)`, and `pam-systemd(8)`.

4.5.1 Fichiers de configuration auxquels accèdent PAM et NSS

Voici quelques fichiers de configuration importants auxquels PAM et NSS peuvent accéder :

fichier de configuration	fonction
<code>/etc/pam.d/nom_programme</code>	définir la configuration de PAM pour le « <i>nom_programme</i> », consultez <code>pam(7)</code> et <code>pam.d(5)</code>
<code>/etc/nsswitch.conf</code>	définir la configuration de NSS avec une entrée pour chaque service. Consultez <code>nsswitch.conf(5)</code>
<code>/etc/nologin</code>	limiter la connexion de l'utilisateur à l'aide du module <code>pam_nologin(8)</code>
<code>/etc/securetty</code>	limiter l'accès de l'administrateur à certains tty à l'aide du module <code>pam_securetty(8)</code>
<code>/etc/security/access.conf</code>	limiter les accès à l'aide du module <code>pam_access(8)</code>
<code>/etc/security/group.conf</code>	définir les limitations en fonction du groupe à l'aide du module <code>pam_group(8)</code>
<code>/etc/security/pam_env.conf</code>	définir des variables d'environnement à l'aide du module <code>pam_env(8)</code>
<code>/etc/environment</code>	définir des variables d'environnement supplémentaires à l'aide du module <code>pam_env(8)</code> avec le paramètre « <code>readenv=1</code> »
<code>/etc/default/locale</code>	définir les paramètres linguistiques (« locale ») à l'aide du module <code>pam_env(8)</code> avec le paramètre « <code>readenv=1</code> <code>envfile=/etc/default/locale</code> » (Debian)
<code>/etc/security/limits.conf</code>	définir les limitations de ressources (<code>ulimit</code> , <code>core</code> , ...) à l'aide du module <code>pam_limits(8)</code>
<code>/etc/security/time.conf</code>	définir les restrictions de temps à l'aide du module <code>pam_time(8)</code>
<code>/etc/systemd/logind.conf</code>	set <code>systemd</code> login manager configuration (see <code>logind.conf(5)</code> and <code>systemd-logind.service(8)</code>)

Table 4.6 – Liste des fichiers de configuration auxquels PAM et NSS accèdent

Les limitations dans la sélection des mots de passe est implémentée par les modules PAM `pam_unix(8)` et `pam_cracklib(8)`. Ils peuvent être configurés à l'aide de leurs paramètres.

ASTUCE

Les noms de fichiers des modules PAM ont le suffixe « `.so` ».

4.5.2 Le système de gestion centralisée moderne

La gestion centralisée du système peut être mise en œuvre en utilisant le serveur centralisé [LDAP Protocole léger d'accès aux répertoires](#) (« [Lightweight Directory Access Protocol](#) ») pour administrer de nombreux systèmes semblables à UNIX ou autres sur le réseau. Le [logiciel OpenLDAP](#) est l'implémentation à sources ouvertes du protocole LDAP

Sur un système Debian, le serveur LDAP fournit les informations de compte en utilisant PAM et de NSS avec les paquets `libpam-ldap` et `libnss-ldap`. Un certain nombre d'actions sont nécessaires pour l'activer (je n'ai pas utilisé cette configuration et ce qui suit est une information secondaire. Veuillez la lire dans ce contexte) :

- définissez un serveur LDAP centralisé en faisant tourner un programme tel que le démon LDAP, `slapd(8)` ;
- modifiez les fichiers de configuration de PAM dans le répertoire « `/etc/pam.d/` » pour utiliser « `pam_ldap.so` » plutôt que le module par défaut « `pam_unix.so` » ;
 - Debian utilise « `/etc/pam_ldap.conf` » comme fichier de configuration de `libpam-ldap` et « `/etc/pam_ldap.s` » pour enregistrer le mot de passe de root.
- modifiez la configuration de NSS dans le fichier « `/etc/nsswitch.conf` » pour utiliser « `ldap` » plutôt que ce qui s’y trouve par défaut (« `compat` » ou « `file` ») ;
 - Debian utilise « `/etc/libnss-ldap.conf` » comme fichier de configuration de `libnss-ldap`.
- vous devez configurer `libpam-ldap` de manière à ce qu’il utilise une connexion [SSL \(ou TLS\)](#) pour la sécurité du mot de passe ;
- vous pouvez configurer `libnss-ldap` de manière à ce qu’il utilise une connexion [SSL \(ou TLS\)](#) afin d’assurer l’intégrité des données au prix d’une surcharge du réseau LDAP ;
- Afin de réduire le trafic réseau de LDAP, vous devrez faire tourner `nscd(8)` localement pour mettre en cache les résultats de recherche de LDAP .

Consultez les documentations dans `pam_ldap.conf(5)` et « `/usr/share/doc/libpam-doc/html/` » qui sont fournies par le paquet `libpam-doc` et « `info libc 'Name Service Switch'` » fournie par le paquet `glibc-doc`.

De manière similaire, vous pouvez mettre en œuvre des systèmes centralisés de remplacement avec d’autres méthodes.

- Intégration d’utilisateur et de groupe au système Windows.
 - Accès aux services de [Domaine \(Microsoft\)](#) avec les paquets `winbind` et `libpam_winbind`.
 - Consultez `winbindd(8)` et [Intégration de réseaux MS Windows avec Samba](#).
- Intégration d’utilisateur et de groupe à l’ancien système de type UNIX.
 - Accès à [NIS \(appelé initialement YP\)](#) ou [NIS+](#) avec le paquet `nis`.
 - Consultez [Le Linux NIS\(YP\)/NYS/NIS+ HOWTO](#).

4.5.3 « Pourquoi la commande `su` de GNU ne gère-t-elle pas le groupe `wheel` »

C’est la célèbre phrase de Richard M. Stallman en bas de l’ancienne page `info su`. Ne pas s’inquiéter : la commande `su` actuelle de <https://www.debian.org> utilise PAM, on peut donc restreindre l’accès de `su` au groupe `root` en activant la ligne de « `/etc/pam.d/su` » comportant « `pam_wheel.so` ».

4.5.4 Règle de mots de passe plus stricte

Installing the `libpam-cracklib` package enables you to force stricter password rule.

On a typical GNOME system which automatically installs `libpam-gnome-keyring`, « `/etc/pam.d/common-password` » looks like:

```
# 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 Sécurité de l'authentification

Note

Les informations données ici pourraient ne répondre **que partiellement** à vos besoins en matière de sécurité mais elles devraient néanmoins constituer **un bon point de départ**.

4.6.1 Mot de passe sûr avec Internet

La couche de transport de nombreux services populaires communique les messages, y compris les mots de passe d'authentification, en texte clair. C'est une très mauvaise idée de transmettre un mot de passe en texte clair dans la jungle d'Internet où il peut être intercepté. Vous pouvez faire tourner des services sur une couche de transport sécurisée « [Sécurité de la couche de transport](#) » (« [Transport Layer Security](#) ») (TLS) ou son prédécesseur « Secure Sockets Layer » (SSL) pour sécuriser par chiffrement la communication dans son ensemble, y compris le mot de passe.

nom de service non sûr	port	nom de service sûr	port
www (http)	80	https	443
smtp (courrier électronique)	25	ssmtp (smtps)	465
données ftp	20	données ftps	989
ftp	21	ftps	990
telnet	23	telnets	992
imap2	143	imaps	993
pop3	110	pop3s	995
ldap	389	ldaps	636

Table 4.7 – Liste des services et ports sûrs et non sûrs

Le chiffrement coûte du temps processeur. Comme solution de remplacement pour économiser du temps processeur, vous pouvez continuer à effectuer les communications en texte clair tout en ne sécurisant que le mot de passe avec un protocole d'authentification sécurisé comme « Authenticated Post Office Protocol » (APOP) pour POP et « Challenge-Response Authentication Mechanism MD5 » (CRAM-MD5) pour SMTP et IMAP. (Depuis peu, pour envoyer des messages de courrier électronique au travers d'Internet à votre serveur de courrier depuis votre client de courrier, il est devenu habituel d'utiliser le port 587 en remplacement du port SMTP 25 habituel pour soumettre le courrier afin d'éviter le blocage du port 25 par le fournisseur d'accès au réseau tout en vous authentifiant avec CRAM-MD5).

4.6.2 Le shell sûr (Secure Shell)

Le programme [SSH, Shell sûr](#) (« Secure Shell ») permet une communication chiffrée sûre entre deux machines qui ne sont pas « de confiance » au travers d'un réseau non sûr avec une authentification sûre. Il est constitué du client [OpenSSH](#), de `ssh(1)`, et du démon [OpenSSH](#), `sshd(8)`. SSH peut être utilisé pour « tunneler » de manière sécurisée un protocole de communications non sûr tel que POP et X au travers d'Internet à l'aide de la fonctionnalité de transfert de port.

Le client essaie de s'authentifier en utilisant l'authentification basée sur l'hôte, une clé publique d'authentification, une authentification par question-réponse ou une authentification par mot de passe. L'utilisation d'une authentification par clé publique permet la connexion à distance sans mot de passe. Consultez Section [6.3](#).

4.6.3 Mesures de sécurité supplémentaires pour Internet

Même si vous tournez avec des services sécurisés comme des serveurs [Secure Shell \(SSH\)](#) et [Point-to-point tunneling protocol \(PPTP\)](#), il reste des possibilités d'effraction en utilisant depuis Internet une attaque de force brute pour trouver le mot de passe, etc. L'utilisation d'une politique de pare-feu (consultez Section [5.6](#)) conjointement avec les outils sûrs qui suivent peut améliorer la sécurité :

paquet	popcon	taille	description
knockd	V:0, I:3	102	démon de serrure électronique IP knockd(1) et client konck(1)
fail2ban	V:116, I:129	2092	bannir les IP qui provoquent des erreurs d'authentification multiples
libpam-shield	V:0, I:0	115	verrouiller les attaquants distants cherchant à deviner le mot de passe

Table 4.8 – Liste des outils fournissant des mesures de sécurité supplémentaires

4.6.4 sécuriser le mot de passe de l'administrateur

Afin d'éviter que des personnes accèdent à votre machine avec les privilèges de l'administrateur, vous devez prendre les mesures suivante :

- rendre impossible l'accès physique au disque dur ;
- Lock UEFI/BIOS and prevent booting from the removable media
- définir un mot de passe pour la session interactive de GRUB ;
- verrouiller l'édition du menu de GRUB.

Avec un accès physique au disque, réinitialiser le mot de passe est relativement facile en suivant les étapes suivantes :

1. Move the hard disk to a PC with CD bootable UEFI/BIOS.
2. Démarrer le système avec un support de secours (disque d'amorçage de Debian, CD Knoppix, CD GRUB...)
3. Monter la partition racine avec les droits en lecture et écriture.
4. Éditer « `/etc/passwd` » de la partition racine et vider la seconde entrée du compte `root`.

Si vous avez l'accès en édition au menu GRUB (consultez Section 3.1.2), avec `grub-rescue-pc`, il est encore plus simple d'effectuer les étapes suivantes au moment du démarrage :

1. Démarrer le système avec les paramètres du noyau modifiés en quelque chose qui ressemble à « `root=/dev/hda6 rw init=/bin/sh` ».
2. Éditer « `/etc/passwd` » et vider la seconde entrée du compte `root`.
3. Redémarrer le système.

L'interpréteur de commandes de l'administrateur est maintenant accessible sans mot de passe.

Note

Une fois que quelqu'un a accès à l'interpréteur de commandes de l'administrateur, il peut accéder à l'ensemble du système et en réinitialiser tous les mots de passe. De plus, il peut compromettre le mot de passe de n'importe quel utilisateur en utilisant des outils de cassage de mots de passe par force brute tels que les paquets `john` et `crack` (consultez Section 9.5.11). Ces mots de passes cassés peuvent permettre de compromettre d'autres systèmes.

La seule solution logicielle raisonnable pour éviter tout ça est d'utiliser une partition racine (ou une partition « `/etc` ») chiffrée par logiciel en utilisant `dm-crypt` et `initramfs` (consultez Section 9.9). Vous aurez alors toujours besoin d'un mot de passe pour vous connecter au système.

4.7 Autres contrôles d'accès

There are access controls to the system other than the password based authentication and file permissions.

Note

Consultez Section 9.4.15 pour restreindre la fonctionnalité de [touche d'appel sécurisée \(SAK\)](#) (« [secure attention key](#) ») du noyau.

4.7.1 sudo

`sudo(8)` est un programme conçu pour permettre à un administrateur système de donner des privilèges d'administration limités aux utilisateurs et d'enregistrer dans un journal les actions de l'administrateur (« root ». `sudo` ne demande que le mot de passe d'un utilisateur normal. Installez le paquet `sudo` et activez-le en définissant les options dans « `/etc/sudoers` ». Consultez l'exemple de configuration dans « `/usr/share/doc/sudo/examples/sudoers` » et Section 1.1.12.

Mon utilisation de `sudo` sur un système avec un seul utilisateur (consultez Section 1.1.12) est destinée à me protéger moi-même contre ma propre stupidité. Personnellement, je considère que l'utilisation de `sudo` est une meilleure alternative que l'utilisation permanente du système depuis le compte de l'administrateur. Par exemple, les modifications suivantes du propriétaire de « `un_fichier` » par « `mon_nom` » :

```
$ sudo chown my_name some_file
```

Bien sûr, si vous connaissez le mot de passe de root (comme beaucoup d'utilisateurs de Debian qui ont installé eux-mêmes leur système), n'importe quelle commande peut être lancée en tant qu'administrateur depuis un compte utilisateur par « `SU -C` ».

4.7.2 PolicyKit

[PolicyKit](#) est un composant du système d'exploitation permettant de contrôler les droits globaux sur les systèmes de type UNIX.

Les applications graphiques les plus récentes ne sont pas conçues pour fonctionner comme des processus privilégiés. Elles échangent avec les processus privilégiés par l'intermédiaire de PolicyKit pour réaliser les opérations d'administration.

PolicyKit limite de telles opérations aux comptes d'utilisateurs appartenant au groupe `sudo` sur le système Debian.

Consultez `polkit(8)`.

4.7.3 Restreindre l'accès à certains services du serveur

Pour la sécurité du système, il est préférable de désactiver autant de programmes de serveurs que possible. Cela devient critique pour les services par l'intermédiaire du réseau. Avoir des services réseau inutilisés, qu'ils soient activés directement en tant que [démon](#) ou par l'intermédiaire du programme [super-serveur](#), est considéré comme un risque de sécurité.

De nombreux programmes, tels que `sshd(8)`, utilisent un contrôle d'accès basé sur PAM. Il y a de nombreuses manières de restreindre l'accès à certains serveurs de services :

- fichiers de configuration : « `/etc/default/nom_programme` » ;
- Systemd service unit configuration for [daemon](#)
- [PAM \(Modules d'authentification attachables\)](#) (« [Pluggable Authentication Modules](#) ») ;
- « `/etc/inetd.conf` » pour le [super-serveur](#) ;
- « `/etc/hosts.deny` » et « `/etc/hosts.allow` » pour l'[enrobeur TCP](#), `tcpd(8)` ;
- « `/etc/rpc.conf` » pour [Sun RPC](#) ;
- « `/etc/at.allow` » et « `/etc/at.deny` » pour `atd(8)` ;
- « `/etc/cron.allow` » et « `/etc/cron.deny` » pour `crontab(1)` ;
- un [pare-feu réseau](#) de l'infrastructure [netfilter](#).

See Section 3.5, Section 4.5.1, and Section 5.6.

ASTUCE

Les services [Sun RPC](#) doivent être actif pour [NFS](#) et les autres programmes basés sur RPC.

ASTUCE

Si vous avez des problèmes pour les accès à distance sur un système Debian récent, commentez la ligne de configuration posant problème, comme « `ALL: PARANOID` » de « `/etc/hosts.deny` » si elle existe. (Mais vous devrez faire attention au risque de sécurité induit par ce type d'action).

4.7.4 Linux security features

Linux kernel has evolved and supports security features not found in traditional UNIX implementations.

Linux supports [extended attributes](#) which extend the traditional UNIX attributes (see `xattr(7)`).

Linux divides the privileges traditionally associated with superuser into distinct units, known as [capabilities\(7\)](#), which can be independently enabled and disabled. Capabilities are a per-thread attribute since kernel version 2.2.

The [Linux Security Module \(LSM\) framework](#) provides a [mechanism for various security checks](#) to be hooked by new kernel extensions. For example:

- [AppArmor](#)
- [Security-Enhanced Linux \(SELinux\)](#)
- [Smack \(Simplified Mandatory Access Control Kernel\)](#)
- [Tomoyo Linux](#)

Since these extensions may tighten privilege model tighter than the ordinary Unix-like security model policies, even the root power may be restricted. You are advised to read the [Linux Security Module \(LSM\) framework document at kernel.org](#).

Linux [namespaces](#) wrap a global system resource in an abstraction that makes it appear to the processes within the namespace that they have their own isolated instance of the global resource. Changes to the global resource are visible to other processes that are members of the namespace, but are invisible to other processes. Since kernel version 5.6, there are 8 kinds of namespaces (see [namespaces\(7\)](#), [unshare\(1\)](#), [nsenter\(1\)](#)).

As of Debian 11 Bullseye (2021), Debian uses unified cgroup hierarchy (a.k.a. [cgroups-v2](#)).

Usage examples of [namespaces](#) with [cgroups](#) to isolate their processes and to allow resource control are:

- [Systemd](#). See Section [3.2](#).
- [Sandbox environment](#). See Section [7.6](#).
- [Linux containers](#) such as [Docker](#), [LXC](#). See Section [9.11](#).

These functionalities can't be realized by Section [4.1](#). These advanced topics are mostly out-of-scope for this introductory document.

Chapitre 5

Configuration du réseau

ASTUCE

Pour un guide actualisé de la gestion réseau sous Debian, lire le [Guide de l'administrateur Debian - configurer le réseau](#) (« The Debian Administrator's Handbook —Configuring the Network »).

ASTUCE

Sous [systemd](#), [networkd](#) peut être utilisé pour gérer les réseaux. Consultez `systemd-networkd(8)`.

5.1 L'infrastructure de base du réseau

Passons en revue l'infrastructure de base du réseau sur un système Debian moderne.

5.1.1 Résolution du nom d'hôte

La résolution du nom d'hôte est actuellement prise en charge aussi par le mécanisme [NSS \(Name Service Switch\)](#). Le flux de cette résolution est le suivant :

1. Le fichier « `/etc/nsswitch.conf` » avec une entrée comme « `hosts: files dns` » donne l'ordre de la résolution du nom d'hôte (cela remplace l'ancienne fonctionnalité de l'entrée « `order` » dans « `/etc/host.conf` »).
2. La méthode `files` est d'abord appelée. Si le nom d'hôte est trouvé dans le fichier « `/etc/hosts` », elle retourne toutes les adresses valables qui y correspondent et quitte. (Le fichier « `/etc/host.conf` » contient « `multi on` »).
3. La méthode `dns` est appelée. Si le nom d'hôte est trouvé par une requête au [Système de noms de domaine Internet \(DNS\)](#) (« Internet Domain Name System ») identifié par le fichier « `/etc/resolv.conf` », elle retourne toutes les adresses valables correspondantes et quitte.

Par exemple, « `/etc/hosts` » ressemble à ce qui suit :

```
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
```

Chaque ligne commence par une [adresse IP](#) et est suivie du [nom d'hôte](#) associé.

paquets	popcon	taille	type	description
network-manager	V:360, I:426	15210	config::NM	NetworkManager (démon) : gère automatiquement le réseau
network-manager-gnome	V:124, I:357	5351	config::NM	NetworkManager (frontal de GNOME)
ifupdown	V:612, I:989	217	config::ifupdown	outil standard pour activer ou désactiver le réseau (spécifique à Debian)
isc-dhcp-client	V:222, I:980	686	config::low-level	client DHCP
pppoeconf	V:0, I:8	192	config::helper	assistant de configuration d'une connexion PPPoE
wpa_supplicant	V:335, I:491	3439	, ,	client prenant en charge WPA et WPA2 (IEEE 802.11i)
wpa_gui	V:0, I:2	780	, ,	client graphique (Qt) pour wpa_supplicant
wireless-tools	V:180, I:239	297	, ,	outils pour manipuler les « Extensions Linux sans fil » (Linux Wireless Extensions)
iw	V:320, I:467	293	, ,	tool for configuring Linux wireless devices
iproute2	V:695, I:937	2867	config::iproute2	iproute2 , IPv6 et autres configurations avancées du réseau : ip(8) , tc(8) , etc
iptables	V:314, I:990	2521	config::Netfilter	outils d'administration pour le filtrage des paquets et NAT (Netfilter)
iputils-ping	V:224, I:997	113	test	tester l'accessibilité d'une machine distante par nom de machine ou adresse IP (iproute2)
iputils-arping	V:7, I:98	55	test	tester l'accessibilité réseau d'une machine distante spécifiée par une adresse ARP
iputils-tracepath	V:4, I:53	72	test	tracer le chemin du réseau vers une machine distante
ethtool	V:103, I:266	597	test	afficher ou modifier les paramètres d'un périphérique Ethernet
mtr-tiny	V:6, I:55	161	test::low-level	tracer le chemin réseau vers une machine distante (curses)
mtr	V:4, I:46	214	, ,	tracer le chemin réseau vers une machine distante (curses et GTK)
gnome-nettool	V:1, I:38	2105	, ,	outils pour des opérations d'informations habituelles sur le réseau (GNOME)
nmap	V:27, I:246	4509	, ,	cartographie réseau / balayage de ports (Nmap , console)
zenmap	V:1, I:8	2939	, ,	cartographie réseau / balayage de ports (GTK)
tcpdump	V:19, I:208	1329	, ,	analyseur de trafic réseau (Tcpdump , console)
wireshark	I:54	65	, ,	analyseur de trafic réseau (Wireshark , GTK)
tshark	V:3, I:33	408	, ,	analyseur de trafic réseau (console)
tcptrace	V:0, I:3	401	, ,	produit un résumé des connexions à partir d'une sortie de tcpdump
snort	V:0, I:1	2206	, ,	système souple de détection d'intrusion par le réseau (Snort)
ntopng	V:1, I:2	969	, ,	afficher l'utilisation du réseau dans le navigateur web
dnsutils	V:59, I:490	261	, ,	clients réseau fournis par BIND : nslookup(8) , nsupdate(8) , dig(8)
dlint	V:0, I:6	53	, ,	vérifier les zones d'information DNS en utilisant des requêtes du serveur de noms
dnstracer	V:0, I:1	61	, ,	tracer une chaîne de serveurs DNS jusqu'à la source

Table 5.1 – Liste des outils de configuration du réseau

L'adresse IP 127.0.1.1 en deuxième ligne de cet exemple pourrait ne pas être présente sur d'autres systèmes de type UNIX. L'[installateur Debian](#) ajoute cette entrée pour les systèmes sans adresse IP permanente en tant que contournement pour certains programmes (par exemple GNOME) comme expliqué dans le [bogue n° 719621](#).

Le *nom_hote* correspond au nom d'hôte défini dans « /etc/hostname ».

Pour un système avec une adresse IP permanente, cette adresse IP devrait être utilisée à la place de 127.0.1.1.

Pour un système avec une adresse IP permanente et un [nom de domaine complètement qualifié \(FQDN\)](#) fourni par le [système de noms de domaine \(DNS\)](#), les *nom_hote* et *nom_domaine* canoniques devraient être utilisés ici, plutôt que le simple *nom_hote*.

« /etc/resolv.conf » est un fichier statique si le paquet `resolvconf` n'est pas installé. S'il est installé, c'est un lien symbolique. Dans tous les cas, il contient des informations qui initialisent les routines du résolveur. Si le DNS est trouvé à l'IP=« 192.168.11.1 », il contient ce qui suit :

```
nameserver 192.168.11.1
```

Le paquet `resolvconf` fait de ce « /etc/resolv.conf » un lien symbolique et gère son contenu automatiquement par le script hook.

Pour une station de travail « compatible PC » sur un réseau local ad hoc typique, le nom d'hôte peut être résolu à l'aide du Multicast DNS (mDNS, [Zeroconf](#)) en plus des méthodes traditionnelles par `fichiers` et `dns`.

- Sur les systèmes Debian, [Avahi](#) fournit un cadre pour le « Multicast DNS Service Discovery ».
- Il est l'équivalent de [Apple Bonjour/Apple Rendezvous](#).
- Le greffon de la bibliothèque `libnss-mdns` fournit une résolution de nom d'hôte à l'aide de mDNS pour la fonction « GNU Name Service Switch (NSS) » de « GNU C Library (glibc) ».
- Le fichier « /etc/nsswitch.conf » devrait contenir une section telle que « `hosts: files mdns4_minimal [NOTFOUND=return] dns mdns4` ».
- La résolution est appliquée aux noms d'hôtes se terminant avec le [pseudo-top-level domain](#) (TLD) « `.local` ».
- L'adresse IPV4 de lien-local multicast mDNS « 224.0.0.251 » ou son équivalente pour l'IPV6 « `FF02::FB` » est utilisée pour faire une requête DNS sur les noms se terminant en « `.local` ».

La résolution du nom de machine par le protocole obsolète [NETBios over TCP/IP](#) utilisé par les systèmes Windows plus anciens peut être fournie en installant le paquet `winbind`. Le fichier « /etc/nsswitch.conf » devra avoir une entrée semblable à « `hosts: files mdns4_minimal [NOTFOUND=return] dns mdns4 wins` » afin d'activer cette fonctionnalité (les systèmes Windows récents utilisent normalement la méthode `dns` pour la résolution de nom d'hôte).

Note

L'[expansion de nom de domaine de premier niveau générique « generic Top-Level Domains \(gTLD\) »](#) dans le [système de noms de domaine](#) est en cours de réalisation. Soyez attentifs aux [conflits de noms](#) lors du choix d'un nom de domaine utilisé uniquement au sein d'un réseau local « LAN ».

5.1.2 Nom de l'interface réseau

The `systemd` uses "[Predictable Network Interface Names](#)" such as "`enp0s25`".

5.1.3 Plage d'adresses réseau du réseau local (« LAN »)

Un rappel des plages d'adresses IPv4 32 bits de chacune des classes réservées à l'utilisation sur un [réseau local \(LAN\)](#) par la [rfc1918](#). Ces adresses garantissent qu'aucun conflit ne sera créé avec aucune des adresses présentes sur Internet proprement dit.

Note

IP address written with colon are [IPv6 address](#), e.g., "`:::1`" for `localhost`.

Classe	adresses de réseau	masque de réseau	masque de réseau /bits	of subnets
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

Table 5.2 – Liste des plages d’adresses de réseau

Note

Si une de ces adresses est assignée à une machine, cette machine ne doit alors pas accéder directement à Internet mais passer par une passerelle qui agit en tant que serveur mandataire (« proxy ») pour les services individuels ou sinon effectuer une [traduction d’adresse réseau \(NAT\)](#) (« Network Address Translation »). Un routeur à large bande effectue en général la NAT pour l’environnement du LAN de l’utilisateur grand public.

5.1.4 La gestion du périphérique réseau

La plupart des périphériques matériels sont pris en charge par le système Debian, il y a quelques périphériques de réseau qui exigent, pour les gérer, des microprogrammes non libres d’après les [principes du logiciel libre selon Debian](#). Veuillez consulter Section [9.10.5](#).

5.2 Configuration moderne de réseau pour ordinateur de bureau

Network interfaces are typically initialized in "networking.service" for the lo interface and "NetworkManager.service" for other interfaces on modern Debian desktop system under systemd.

Debian can manage the network connection via management [daemon](#) software such as [NetworkManager \(NM\)](#) (network-manager and associated packages).

- Ils sont fournis avec leur propre interface utilisateur graphique ([GUI](#)) et en ligne de commandes.
- Ils ont leur propre [démon](#) en tant que système dorsal.
- Ils permettent une connexion facile de votre système à Internet.
- Ils permettent une gestion facile de la configuration du réseau filaire ou sans fil.
- Ils nous permettent de configurer le réseau indépendamment de l’ancien paquet « ifupdown »

Note

Ne pas utiliser ces outils de configuration automatique du réseau sur un serveur. Ils ont été prévus principalement pour les utilisateurs de système de bureau tournant sur des ordinateurs portables.

Ces outils modernes de configuration du réseau doivent être configurés correctement afin d’éviter des conflits avec l’ancien paquet ifupdown et son fichier de configuration « /etc/network/interfaces ».

5.2.1 Outils graphiques de configuration du réseau

Official documentations for NM on Debian are provided in "/usr/share/doc/network-manager/README.Debian".

Essentiellement, la configuration réseau pour un ordinateur de bureau est faite de la manière suivante :

1. Rendez l’utilisateur du bureau, par exemple `toto`, membre du groupe « `netdev` » à l’aide de la commande suivante (vous pouvez aussi le faire automatiquement à l’aide de [D-bus](#) sous les environnements de bureau modernes comme GNOME et KDE) :

```
$ sudo adduser foo netdev
```

2. Gardez la configuration de « `/etc/network/interfaces` » aussi simple que possible comme ce qui suit :

```
auto lo
iface lo inet loopback
```

3. Restart NM by the following.

```
$ sudo systemctl restart network-manager
```

4. Configurez votre réseau à l'aide d'une interface graphique.

Note

Only interfaces which are **not** listed in `/etc/network/interfaces` are managed by NM to avoid conflict with `ifupdown`.

ASTUCE

If you wish to extend network configuration capabilities of NM, please seek appropriate plug-in modules and supplemental packages such as `network-manager-openconnect`, `network-manager-openvpn-gnome`, `network-manager-pptp-gnome`, `mobile-broadband-provider-info`, `gnome-bluetooth`, etc.

5.3 The modern network configuration without GUI

Sous [systemd](#), le réseau peut aussi être configuré dans `/etc/systemd/network/`. Voir : `systemd-resolved(8)`, `resolved.conf(5)` et `systemd-networkd(8)`.

This allows the modern network configuration without GUI.

A DHCP client configuration can be set up by creating `/etc/systemd/network/dhcp.network`. E.g.:

```
[Match]
Name=en*

[Network]
DHCP=yes
```

A static network configuration can be set up by creating `/etc/systemd/network/static.network`. E.g.:

```
[Match]
Name=en*

[Network]
Address=192.168.0.15/24
Gateway=192.168.0.1
```

5.4 Configuration réseau de bas niveau

For the low level network configuration on Linux, use the [iproute2](#) programs (`ip(8)`, ...).

net-tools obsolètes	nouveau iproute2, etc.	manipulation
ifconfig(8)	ip addr	adresse de protocole (IP ou IPv6) d'un périphérique
route(8)	ip route	entrée de la table de routage
arp(8)	ip neigh	entrée de cache ARP ou NDISC
ipmaddr	ip maddr	adresse multicast
iptunnel	ip tunnel	tunnel sur IP
nameif(8)	ifrename(8)	nommer les interfaces réseau en se basant sur l'adresse MAC
mi-tool(8)	ethtool(8)	paramétrage du périphérique Ethernet

Table 5.3 – Table de conversion depuis les commandes obsolètes `net-tools` vers les nouvelles commandes `iproute2`

5.4.1 Commandes Iproute2

Les commandes `iproute2` offrent des possibilités complètes de configuration de bas niveau du réseau. Voici une table de conversion des commandes obsolètes `net-tools` obsolètes vers les nouvelles commandes `iproute2`, etc.

Consultez `ip(8)` et [Howto de la suite utilitaire IPROUTE2](#).

5.4.2 Opérations sûres de bas niveau sur le réseau

Vous pouvez utiliser de manière sûre les commandes de réseau de bas niveau de la manière suivante car elles ne modifient pas la configuration du réseau :

commande	description
ip addr show	afficher l'état et l'adresse du lien des interfaces actives
route -n	afficher toutes les tables de routage sous forme d'adresses numériques
ip route show	afficher toutes les tables de routage sous forme d'adresses numériques
arp	afficher le contenu actuel des tables de cache d' ARP
ip neigh	afficher le contenu actuel des tables de cache d' ARP
plog	afficher le journal du démon ppp
ping yahoo.com	vérifier la connexion internet vers « yahoo.com »
whois yahoo.com	vérifier qui a enregistré « yahoo.com » dans la base de données des domaines
traceroute yahoo.com	tracer la connexion Internet vers « yahoo.com »
tracpath yahoo.com	tracer la connexion Internet vers « yahoo.com »
mtr yahoo.com	tracer la connexion Internet vers « yahoo.com » (de manière répétitive)
dig [@dns-serveur.com] example.com [{a mx any}]	vérifier les enregistrements DNS de « example.com » par « dns-serveur.com » pour un enregistrement « a », « mx » ou « any »
iptables -L -n	vérifier le filtre de paquets
netstat -a	rechercher tous les ports ouverts
netstat -l --inet	rechercher les ports à l'écoute
netstat -ln --tcp	rechercher les ports TCP à l'écoute (numérique)
dlint example.com	vérifier les informations de zone DNS de « example.com »

Table 5.4 – Liste des commandes de réseau de bas niveau

ASTUCE

Certains de ces outils de configuration du réseau se trouvent dans « `/sbin/` ». Il vous faudra peut-être utiliser le chemin complet vers la commande comme « `/sbin/ifconfig` » ou ajouter « `/sbin` » à la liste « `$PATH` » dans votre fichier « `~/.bashrc` »

5.5 Optimisation du réseau

L'optimisation générique du réseau est en dehors des buts de cette documentation. Je ne parle que des sujets pertinents pour une connexion de l'utilisateur grand public.

paquets	popcon	taille	description
iftop	V:8, I:118	97	afficher l'utilisation de la bande passante d'une interface réseau
iperf	V:4, I:55	305	outil de mesure de la bande passante du protocole Internet
ifstat	V:0, I:8	60	InterFace STATistics Monitoring (surveillance des statistiques de l'interface)
bmon	V:1, I:18	146	surveillance portable de la bande passante et estimation du débit
ethstatus	V:0, I:4	40	script qui mesure rapidement le débit d'une interface réseau
bing	V:0, I:1	80	testeur de bande passante empirique et stochastique
bwm-ng	V:2, I:18	90	moniteur de bande passante simple en mode console
ethstats	V:0, I:0	23	moniteur de statistiques Ethernet en mode console
ipfm	V:0, I:0	78	outil d'analyse de bande passante

Table 5.5 – Liste des outils d'optimisation du réseau.

5.5.1 Rechercher le MTU optimum

La valeur du [Maximum Transmission Unit \(MTU\)](#) (Unité de transmission maximum) peut être déterminée expérimentalement par `ping(8)` avec l'option « `-M do` » qui envoie des paquets ICMP dont la taille commence par 1500 (avec un décalage de 28 octets pour l'en-tête IP+ICMP) et recherche la taille la plus grande sans fragmentation d'IP.

Essayez, par exemple, ce qui suit :

```
$ ping -c 1 -s $((1500-28)) -M do www.debian.org
PING www.debian.org (194.109.137.218) 1472(1500) bytes of data.
From 192.168.11.2 icmp_seq=1 Frag needed and DF set (mtu = 1454)

--- www.debian.org ping statistics ---
0 packets transmitted, 0 received, +1 errors
```

Try MTU=1454 instead of MTU=1500

You see `ping(8)` succeed with MTU=1454.

If MTU is not 1500, you may want to configure MTU settings in NM.

Ce processus est la [découverte du chemin MTU \(PMTU\)](#) ([RFC1191](#)) et la commande `tracepath(8)` peut l'automatiser.

ASTUCE

L'exemple ci-dessus avec une valeur de PMTU de 1454 correspond à mon fournisseur FTTP précédent qui utilisait [Asynchronous Transfer Mode](#) (ATM) comme réseau fédérateur (« backbone ») et servait ses clients avec [PPPoE](#). La valeur réelle de PMTU dépend de votre environnement, par exemple, 1500 avec mon nouveau fournisseur FTTP.

En plus de ces lignes directrices, vous devriez savoir ce qui suit :

- Toute utilisation d'une méthode de tunneling ([VPN](#), etc.) peut réduire le MTU optimal en raison de la surcharge qu'elle engendre.
 - La valeur de MTU ne doit pas excéder la valeur expérimentale déterminée de PMTU .
 - La valeur de MTU la plus élevée est généralement meilleure lors que les autres limitations sont remplies.
-

environnement de réseau	MTU	justification
Lien commuté (IP : PPP)	576	standard
Lien Ethernet (IP : DHCP ou fixe)	1500	standard et par défaut
lien Ethernet (IP : PPPoE)	1492 (=1500-8)	2 octets pour l'en-tête PPP et 6 octets pour l'en-tête PPPoE
lien Ethernet (épine dorsale de réseau du FAI : ATM, IP : DHCP ou fixe)	1462 (=48*31-18-8)	spéculations de l'auteur : 18 octets pour l'en-tête Ethernet, 8 octets pour le « trailer » SAR
lien Ethernet (réseau fédérateur du FAI : ATM, IP : PPPoE)	1454 (=48*31-8-18-8)	author's speculation: 18 bytes for Ethernet header, 8 bytes for SAR trailer, 2 bytes for PPP header and 6 bytes for PPPoE header

Table 5.6 – Lignes directrices pour une valeur optimum de MTU

La [taille maximum de segment](#) (MSS : « maximum segment size ») est utilisée comme mesure de remplacement de la taille des paquets. La relation entre MSS et MTU est la suivante :

- MSS = MTU - 40 pour IPv4
- MSS = MTU - 60 pour IPv6

Note

Les optimisations basées sur `iptables(8)` (consultez Section 5.6) peuvent limiter la taille des paquets au MSS, ce qui est utile pour le routeur. Consultez "TCPMSS" dans `iptables(8)`.

5.5.2 Optimisation de TCP sur le réseau Internet

Le débit de TCP peut être maximisé en ajustant les paramètres de taille de tampon TCP comme cela est décrit dans « [Guide de réglage de TCP](#) » (« TCP Tuning Guide ») et « [réglage de TCP](#) » (« TCP tuning ») pour les réseaux WAN modernes de haut débit et de faible latence. À ce jour, les paramètres par défaut de Debian fonctionnent bien même lorsque mon réseau local est connecté par l'intermédiaire d'un service rapide sur fibre optique (FTTP) à 1Gb/s.

5.6 Infrastructure de netfilter

[Netfilter](#) fournit l'infrastructure pour un [pare-feu dynamique](#) (« stateful firewall ») et la [traduction d'adresses réseau \(NAT\)](#) (« network address translation ») avec des modules du [noyau de Linux](#) (consultez Section 3.8.1).

L'outil [netfilter](#) principal de l'espace utilisateur est `iptables(8)`. Vous pouvez configurer vous-même [netfilter](#) de manière interactive depuis l'interpréteur de commandes, enregistrer son état avec `iptables-save(8)` et le restaurer par l'intermédiaire d'un script d'init avec `iptables-restore(8)` lors du redémarrage du système.

Des scripts d'assistant tels que [shorewall](#) facilitent ce processus.

Consultez les documentations se trouvant sur <http://www.netfilter.org/documentation/> (ou dans « /usr/share/doc/iptables/htm

- [Linux Networking-concepts HOWTO](#) (HOWTO des concepts réseau de Linux)
- [Linux 2.4 Packet Filtering HOWTO](#) (HOWTO du filtrage des paquets de Linux 2.4)
- [Linux 2.4 NAT HOWTO](#) (HOWTO du NAT de Linux 2.4)

ASTUCE

Bien qu'elles aient été écrites pour Linux 2.4, la commande `iptables(8)` et la fonction `netfilter` du noyau s'appliquent toutes deux aux séries 2.6 et 3.x du noyau Linux.

paquets	popcon	taille	description
iptables	V:314, I:990	2521	outils d'administration pour netfilter (iptables(8) pour IPv4, ip6tables(8) for IPv6)
arptables	V:0, I:2	96	outils d'administration pour netfilter (arptables(8) pour ARP)
ebtables	V:16, I:37	265	outils d'administration pour netfilter (ebtables(8) pour le pontage Ethernet)
iptstate	V:0, I:3	116	surveillance continue de l'état de netfilter (semblable à top(1))
shorewall-init	V:0, I:0	68	initialisation de Shoreline Firewall
shorewall	V:4, I:12	2458	Shoreline Firewall , générateur de fichier de configuration pour netfilter
shorewall-lite	V:0, I:0	65	Shoreline Firewall , générateur de fichier de configuration pour netfilter (version légère)
shorewall6	V:1, I:2	779	Shoreline Firewall , générateur de fichier de configuration pour netfilter (version IPv6)
shorewall6-lite	V:0, I:0	64	Shoreline Firewall , générateur de fichier de configuration pour netfilter (version légère, IPv6)

Table 5.7 – Liste d'outils de pare-feu

Chapitre 6

Applications réseau

Après avoir établi une connexion réseau (consultez Chapitre 5), vous pouvez faire tourner diverses applications réseau.

ASTUCE

Pour un guide spécifique de Debian moderne sur les infrastructures réseaux, lisez [Le Livre de l'Administrateur Debian - Infrastructure réseau](#).

ASTUCE

If you enabled "2-Step Verification" with some ISP, you need to obtain an application password to access POP and SMTP services from your program. You may need to approve your host IP in advance.

6.1 Navigateurs Web

Il y a de nombreux paquets de [navigateurs web](#) permettant d'accéder à des contenus distants avec le [protocole de transfert hyper-texte](#) (« Hypertext Transfer Protocol (HTTP) »).

paquet	popcon	taille	type	description du navigateur web
chromium	V:54, I:134	192549	X	Chromium (navigateur libre de Google)
firefox	V:13, I:20	209827	, ,	Firefox (navigateur libre de Mozilla, uniquement disponible dans Debian Unstable)
firefox-esr	V:199, I:421	200011	, ,	Firefox ESR (Firefox Extended Support Release = Firefox Support à Long Terme)
epiphany-browser	V:4, I:23	3730	, ,	GNOME , Epiphany respectant HIG
konqueror	V:17, I:95	20763	, ,	KDE , Konqueror
dillo	V:1, I:7	1536	, ,	Dillo (navigateur léger, basé sur FLTK)
w3m	V:25, I:252	2367	texte	w3m
lynx	V:11, I:97	1948	, ,	Lynx
elinks	V:5, I:27	1767	, ,	ELinks
links	V:5, I:38	2250	, ,	Links (texte uniquement)
links2	V:1, I:14	5418	graphique	Links (graphique en mode console sans X)

Table 6.1 – Liste de navigateurs web

6.1.1 Configuration du navigateur

Vous pouvez utiliser les chaînes d'URL spéciales suivantes pour confirmer le paramétrage de certains navigateurs :

- « `about:` »
- « `about:config` »
- « `about:plugins` »

Debian propose pour les navigateurs, dans la section main, de nombreux paquets de greffons libres qui peuvent gérer non seulement [Java \(plateforme logicielle\)](#) et [Flash](#) mais aussi les fichiers [MPEG](#), [MPEG2](#), [MPEG4](#), [DivX](#), [Windows Media Video \(.wmv\)](#), [QuickTime \(.mov\)](#), [MP3 \(.mp3\)](#), [Ogg/Vorbis](#), les DVD, VCD, etc. Debian propose aussi, dans les sections contrib ou non-free, des paquets de programmes facilitant l'installation de greffons non libres pour les navigateurs.

paquet	popcon	taille	section	description
pepperflashplugin-nonfree	V:0, I:15	26	contrib	Pepper Flash Player - browser plugin

Table 6.2 – Liste des paquets de greffons pour les navigateurs

ASTUCE

Bien que l'utilisation des paquets Debian ci-dessus soit plus facile, il est toujours possible d'installer vous-même des greffons de navigateur en installant les fichiers « `*.so` » dans les répertoires de greffons, (par exemple, « `/usr/lib/iceweasel/plugins/` ») et en redémarrant le navigateur.

Certains sites web refusent les connexions en se basant sur la chaîne « `user-agent` » de votre navigateur. Vous pouvez contourner cette situation en [usurpant la chaîne user-agent](#). Vous pouvez le faire, par exemple, en ajoutant la ligne suivante dans les fichiers de configuration de l'utilisateur tels que « `~/.gnome2/epiphany/mozilla/epiphany/user.js` » ou « `~/.mozilla/firefox/` » :

```
user_pref("general.useragent.override", "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0) ↵";
```

Vous pouvez aussi ajouter et réinitialiser cette variable en entrant l'URL « `about:config` » et en faisant un clic-droit pour afficher son contenu.



Attention

Usurper la chaîne `user-agent` peut provoquer de [mauvais effets de bord avec Java](#).

6.2 Le système de courrier électronique

This section focuses on typical mobile workstations on consumer grade Internet connections.



Attention

Si vous êtes sur le point de configurer le serveur de courrier pour échanger directement du courrier avec Internet, vous feriez mieux de lire ce document élémentaire.

6.2.1 Bases du courrier électronique

Un [courrier électronique](#) est composé de trois parties : l'enveloppe, l'en-tête et le corps du message.

- Les renseignements « To » et « From » de l'enveloppe sont utilisés par le [SMTP](#) pour délivrer le courrier électronique (« From » dans l'enveloppe indique l'[adresse de rebond](#), « From_ », etc.).
 - Les renseignements « To » et « From » de l'en-tête sont affichés par le [client de messagerie](#) (même s'ils sont généralement identiques à ceux de l'enveloppe, ce n'est pas toujours le cas).
 - The email message format covering header and body data is extended by [Multipurpose Internet Mail Extensions \(MIME\)](#) from the plain ASCII text to other character encodings, as well as attachments of audio, video, images, and application programs.
- Full featured GUI based [email clients](#) offer all the following functions using the GUI based intuitive configuration.
- It creates and interprets the message header and body data using [Multipurpose Internet Mail Extensions \(MIME\)](#) to deal the content data type and encoding.
 - It authenticates itself to the ISP's SMTP and IMAP servers using the legacy [basic access authentication](#) or modern [OAuth 2.0](#). (For [OAuth 2.0](#), set it via Desktop environment settings. E.g., "Settings" -> "Online Accounts".)
 - It sends the message to the ISP's smarthost SMTP server listening to the message submission port (587).
 - It receives the stored message on the ISP's server from the TLS/IMAP4 port (993).
 - It can filter mails by their attributes.
 - It may offer additional functionalities: Contacts, Calendar, Tasks, Memos.

paquet	popcon	taille	type
evolution	V:29, I:219	475	programme X avec une interface graphique (GNOME 3, suite « groupware »)
thunderbird	V:57, I:134	168690	X GUI program (GTK, Mozilla Thunderbird)
kmail	V:33, I:85	18011	programme X avec une interface graphique (KDE)
mutt	V:33, I:278	7071	programme de terminal en mode caractère, probablement utilisé avec vim
mew	V:0, I:0	2334	programme de terminal en mode caractères sous (x) emacs

Table 6.3 – Liste d'agents de courrier électronique de l'utilisateur (MUA)

6.2.2 Modern mail service limitation

Modern mail service are under some limitations in order to minimize exposure to the spam (unwanted and unsolicited email) problems.

- It is not realistic to run SMTP server on the consumer grade network to send mail directly to the remote host reliably.
- A mail may be rejected by any host en route to the destination quietly unless it appears as authentic as possible.
- It is not realistic to expect a single smarthost to send mails of unrelated source mail addresses to the remote host reliably.

This is because:

- The SMTP port (25) connections from hosts serviced by the consumer grade network to the Internet are blocked.
- The SMTP port (25) connections to hosts serviced by the consumer grade network from the Internet are blocked.
- The outgoing messages from hosts serviced by the consumer grade network to the Internet can only be sent via the message submission port (587).
- [Anti-spam techniques](#) such as [DomainKeys Identified Mail \(DKIM\)](#), [Sender_Policy_Framework \(SPF\)](#), and [Domain-based Message Authentication, Reporting and Conformance \(DMARC\)](#) are widely used for the [email filtering](#).
- Le service de [DomainKeys Identified Mail](#) peut être fourni pour vos messages envoyés par l'intermédiaire du smarthost.
- The smarthost may rewrite the source mail address in the message header to your mail account on the smarthost to prevent email address spoofing.

6.2.3 Historic mail service expectation

Some programs on Debian expect to access the `/usr/sbin/sendmail` command to send emails as their default or customized setting since the mail service on a UNIX system functioned historically as:

- An email is created as a text file.
- The email is handed to the `/usr/sbin/sendmail` command.
- For the destination address on the same host, the `/usr/sbin/sendmail` command makes local delivery of the email by appending it to the `/var/mail/$username` file.
 - Commands expecting this feature: `apt-listchanges`, `cron`, `at`, ...
- For the destination address on the remote host, the `/usr/sbin/sendmail` command makes remote transfer of the email to the destination host found by the DNS MX record using SMTP.
 - Commands expecting this feature: `popcon`, `reportbug`, `bts`, ...

6.2.4 Agent de transport de courrier électronique (« MTA »)

In principle, mobile workstations should function without the `/usr/sbin/sendmail` command provided by the [mail transfer agent \(MTA\)](#) program.

The Debian system usually installs MTA to cope with Section 6.2.2 and Section 6.2.3 even if mobile workstations installed full featured GUI based [email clients](#).

For mobile workstations, the typical choice of MTA is either `exim4-daemon-light` or `postfix` with its installation option such as "Mail sent by smarthost; received via SMTP or fetchmail" selected. These are light weight MTAs that respect `/etc/aliases`.

ASTUCE

Configuring `exim4` to send the Internet mail via multiple corresponding smarthosts for multiple source email addresses is non-trivial. If you need such capability for some programs, set them up to use `msmtp` which is easy to set up for multiple source email addresses. Then leave main MTA only for a single email address.

6.2.4.1 Configuration d'exim4

Pour le courrier d'Internet par l'intermédiaire d'un smarthost, vous (re)configurerez les paquets `exim4-*` comme suit :

```
$ sudo systemctl stop exim4
$ sudo dpkg-reconfigure exim4-config
```

Choisir « envoi via relais (« smarthost ») - réception SMTP ou fetchmail » : pour « Configuration du serveur de courriel ».

Définir « Nom de courriel du système » à sa valeur par défaut qui est le nom pleinement qualifié (FQDN, consultez Section 5.1.1).

Définir « Liste d'adresses IP où Exim sera en attente de connexions SMTP entrantes » à sa valeur par défaut qui est « 127.0.0.1 ; ::1 ».

Supprimer le contenu de « Autres destinations dont le courriel doit être accepté ».

Supprimer le contenu de « Machines à relayer ».

Définir « Nom réseau ou adresse IP du système « smarthost » » à « `smtp.hostname.dom:587` ».

Select "No" for "Hide local mail name in outgoing mail?". (Use `/etc/email-addresses` as in Section 6.2.4.3, instead.)

Donner à « Faut-il optimiser les requêtes DNS (connexion à la demande) ? » l'une des réponses suivantes :

- « Non » si le système est connecté à Internet au démarrage.
 - « Oui » si le système n'est **pas** connecté à Internet au démarrage.
-

paquet	popcon	taille	description
exim4-daemon-light	V:320, I:346	1517	Agent de transport de courrier électronique Exim4 (MTA : par défaut dans Debian)
exim4-daemon-heavy	V:7, I:8	1668	Exim4 mail transport agent (MTA: flexible alternative)
exim4-base	V:328, I:355	1720	Documentation d'Exim4 (texte) et fichiers communs
exim4-doc-html	I:1	3667	Documentation d'Exim4 (html)
exim4-doc-info	I:1	625	Documentation d'Exim4 (info)
postfix	V:147, I:162	4183	Postfix mail transport agent (MTA: secure alternative)
postfix-doc	I:9	4444	Documentation de Postfix (html+texte)
saslm2-bin	V:5, I:19	428	Implémentation de l'API Cyrus SASL (complément à Postfix pour SMTP AUTH)
cyru0s-saslm2-doc	I:1	575	Cyrus SASL - documentation
msmt0p	V:6, I:12	547	Light weight MTA
msmt0p-mta	V:4, I:5	86	Light weight MTA (sendmail compatibility extension to msmt0p)
esmt0p	V:0, I:0	128	Light weight MTA
esmt0p-run	V:0, I:0	32	Light weight MTA (sendmail compatibility extension to esmt0p)
nullmailer	V:8, I:10	479	Strip down MTA, no local mail
ssmt0p	V:6, I:10	2	Strip down MTA, no local mail
sendmail-bin	V:14, I:15	1859	Full featured MTA (only if you are already familiar)
courier-mta	V:0, I:0	2442	Full featured MTA (web interface etc.)

Table 6.4 – List of basic mail transport agent related packages

Définir « Méthode de distribution du courrier local : » à « Format « mbox » dans /var/mail ».

Select "Yes" for "Split configuration into small files?:".

Créer les entrées de mots de passe pour le smarthost en éditant « /etc/exim4/passwd.client ».

```
$ sudo vim /etc/exim4/passwd.client
...
$ cat /etc/exim4/passwd.client
^smtp.*\.hostname\.dom:username@hostname.dom:password
```

Configure exim4(8) with "QUEUERUNNER='queueonly'", "QUEUERUNNER='nodaemon'", etc. in "/etc/default/exim4" to minimize system resource usages. (optional)

Lancer exim4 par la commande suivante :

```
$ sudo systemctl start exim4
```

Le nom de machine dans « /etc/exim4/passwd.client » ne doit pas être un alias. Vérifiez le nom de machine réel comme suit :

```
$ host smtp.hostname.dom
smtp.hostname.dom is an alias for smtp99.hostname.dom.
smtp99.hostname.dom has address 123.234.123.89
```

J'utilise une expression rationnelle dans « /etc/exim4/passwd.client » pour contourner le problème d'alias. SMTP AUTH fonctionne probablement même dans le cas où le FAI déplace la machine pointée par l'alias.

Vous pouvez mettre à jour vous-même la configuration d'exim4 de la façon suivante :

- Mettre à jour les fichiers de configuration d'exim4 dans « /etc/exim4/ ».
 - Créer « /etc/exim4/exim4.conf.localmacros » pour configurer les macros et éditer « /etc/exim4/exim4.conf » (configuration en un seul fichier).
 - Créer de nouveaux fichiers ou éditer des fichiers existants dans les sous-répertoires de « /etc/exim4/exim4.conf.d » (configuration séparée en plusieurs fichiers).

— Run `systemctl reload exim4`.



Attention

Le lancement d'exim4 est long si on a choisi « Non » (valeur par défaut) à la demande « Faut-il optimiser les requêtes DNS (connexion à la demande) ? » lors de la configuration debconf et que le système n'est **pas** connecté à Internet lors du démarrage.

Veuillez lire le guide officiel se trouvant à « `/usr/share/doc/exim4-base/README.Debian.gz` » et `update-exim4.conf`.



AVERTISSEMENT

For all practical consideration, use **SMTP** with **STARTTLS** on port 587 or **SMTPS** SSL (SMTPS) on port 465, instead of plain SMTP on port 25.

6.2.4.2 Configuration de postfix avec SASL

Pour utiliser le courrier électronique d'Internet par l'intermédiaire d'un smarthost, vous devrez d'abord lire la [documentation postfix](#) et les pages de manuel importantes.

commande	fonction
<code>postfix(1)</code>	Programme de contrôle de postfix
<code>postconf(1)</code>	Utilitaire de configuration de postfix
<code>postconf(5)</code>	Paramètres de configuration de postfix
<code>postmap(1)</code>	Maintenance des tables de consultation de postfix
<code>postalias(1)</code>	Maintenance de la base de données des alias de postfix

Table 6.5 – Liste des pages de manuel importantes de postfix

Vous (re)configurez les paquets `postfix` et `sasl2-bin` comme suit :

```
$ sudo systemctl stop postfix
$ sudo dpkg-reconfigure postfix
```

Choisir « Internet avec smarthost ».

Définissez « machine de relais SMTP (blanc pour aucun): » à « `[smtp.hostname.dom]:587` » et configurez-le de la manière suivante :

```
$ 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
```

Créez les entrées de mots de passe pour le smarthost

```
$ cat /etc/postfix/sasl_passwd
[smtp.hostname.dom]:587      username:password
$ sudo postmap hush:/etc/postfix/sasl_passwd
```

Lancez `postfix` comme suit :

```
$ sudo systemctl start postfix
```

Ici, l'utilisation de « `[]` » et « `]` » dans le dialogue de `dpkg-reconfigure` et « `/etc/postfix/sasl_passwd` » permet de s'assurer de ne pas vérifier l'enregistrement MX mais d'utiliser directement le nom exact de la machine indiquée. Consultez « Enabling SASL authentication in the Postfix SMTP client » dans « `/usr/share/doc/postfix/html/SASL_README.html` ».

6.2.4.3 Configuration de l'adresse de courriel

Il existe plusieurs [fichiers de configuration de l'adresse de courriel](#) pour l'acheminement du courriel, sa diffusion et les agents d'utilisateur.

fichier	fonction	application
/etc/mailname	nom de machine par défaut pour le courrier (sortant)	Spécifique à Debian, mailname(5)
/etc/email-addresses	usurpation du nom de machine pour le courriel sortant	Spécifique à <code>exim(8)</code> , <code>exim4-config_files(5)</code>
/etc/postfix/generic	usurpation du nom de machine pour le courriel sortant	Spécifique à <code>postfix(1)</code> <code>specific</code> , activé après l'exécution de la commande <code>postmap(1)</code> .
/etc/aliases	alias du nom de compte pour le courrier entrant	général, activé après l'exécution de la commande <code>newaliases(1)</code> .

Table 6.6 – Liste des fichiers de configuration liés aux adresses de courriel

Le **nom de courriel** (« mailname » dans le fichier « /etc/mailname » est habituellement un nom de domaine entièrement qualifié (FQDN) qui est résolu vers l'une des adresses IP de la machine. Pour les stations de travail mobiles qui n'ont pas de nom de machine pouvant être résolu par une adresse IP, définissez ce **mailname** à la valeur donnée par « `hostname -f` ». (C'est un choix sûr et qui fonctionne à la fois avec `exim4-*` et `postfix`.)

ASTUCE

Le contenu de « /etc/mailname » est utilisé par de nombreux programmes autres que les MTA pour définir leur comportement par défaut. Pour `mutt`, définissez les variables « `hostname` » et « `from` » dans le fichier `~/muttrc` pour passer outre la valeur de **mailname**. Pour les programmes du paquet `devscripts`, comme `bts(1)` et `dch(1)`, exportez les variables d'environnement « `$DEBFULLNAME` » et « `$DEBEMAIL` » afin de passer outre cette définition.

ASTUCE

Le paquet `popularity-contest` envoie normalement un courriel depuis le compte de l'administrateur avec un nom de domaine pleinement qualifié (FQDN). Vous devez définir `MAILFROM` dans `/etc/popularity-contest.conf` comme c'est décrit dans le fichier `/usr/share/popularity-contest/default.conf`. Sinon, votre courriel sera rejeté par le serveur SMTP sur « `smarthost` ». Bien que ce soit fastidieux, cette approche est plus sûre que la réécriture par le MTA de l'adresse source pour tous les courriels en provenance de l'administrateur (« `root` ») et devrait être utilisé pour les autres démons et les scripts des tâches planifiées (« `cron` »).

Lors de la définition de **mailname** avec la valeur donnée par « `hostname -f` », l'usurpation de l'adresse source du courrier par le MTA peut être réalisée par l'intermédiaire :

- du fichier « /etc/email-addresses » pour `exim4(8)` comme expliqué dans `exim4-config_files(5)`
- du fichier « /etc/postfix/generic » pour `postfix(1)` comme expliqué dans `generic(5)`

Pour `postfix`, les étapes suivantes sont nécessaires :

```
# postmap hash:/etc/postfix/generic
# postconf -e 'smtp_generic_maps = hash:/etc/postfix/generic'
# postfix reload
```

Vous pouvez tester la configuration de l'adresse de courriel de la manière suivante :

- `exim(8)` avec les options `-brw`, `-bf`, `-bF`, `-bV`, ...
- `postmap(1)` avec l'option `-q`

ASTUCE

Il existe, avec Exim, un certain nombre de programmes utilitaires tels qu'`exiqgrep(8)` et `exipick(8)`. Consultez « `dpkg -L exim4-base | grep man8/` » pour les commandes disponibles.

6.2.4.4 Opération de base du MTA

Il y a quelques opérations de base du MTA. Certaines peuvent être effectuées à l'aide de l'interface de compatibilité avec `sendmail(1)`.

commande exim	commande postfix	description
<code>sendmail</code>	<code>sendmail</code>	lire les courriels depuis l'entrée standard et les classer pour la diffusion (<code>-bm</code>)
<code>mailq</code>	<code>mailq</code>	afficher la file d'attente des courriels avec leur état et leur identifiant de file d'attente (« queue ID ») (<code>-bp</code>)
<code>newaliases</code>	<code>newaliases</code>	initialiser la base de données des alias (<code>-I</code>)
<code>exim4 -q</code>	<code>postqueue -f</code>	supprimer les courriels en attente (<code>-q</code>)
<code>exim4 -qf</code>	<code>postsuper -r ALL deferred; postqueue -f</code>	supprimer tous les courriels
<code>exim4 -qff</code>	<code>postsuper -r ALL; postqueue -f</code>	supprimer tous les courriels gelés
<code>exim4 -Mg queue_id</code>	<code>postsuper -h queue_id</code>	geler un message d'après son identifiant de file d'attente
<code>exim4 -Mrm queue_id</code>	<code>postsuper -d queue_id</code>	supprimer un message d'après son identifiant de file d'attente
N/A	<code>postsuper -d ALL</code>	supprimer tous les messages.

Table 6.7 – Liste des opérations de base du MTA

ASTUCE

Ce peut être une bonne idée de supprimer tous les messages à l'aide d'un script placé dans « `/etc/ppp/ip-up.d/*` ».

6.3 Le serveur et les utilitaires d'accès à distance (SSH)

SSH, le « [Secure SHell](#) », est la manière **sûre** de se connecter au travers d'Internet. Une version libre de SSH, appelée [OpenSSH](#), est disponible sous Debian sous forme des paquets `openssh-client` et `openssh-server`.

Pour l'utilisateur, `ssh(1)` fonctionne comme un `telnet(1)` intelligent et plus sûr. Contrairement à la commande `telnet`, la commande `ssh` ne s'arrête pas avec le caractère d'échappement de `telnet` (valeur initiale par défaut Ctrl-J).

Although `shellinabox` is not a SSH program, it is listed here as an interesting alternative for the remote terminal access.

See also Section 7.8 for connecting to remote X client programs.

**Attention**

Consultez Section 4.6.3 si votre serveur SSH est accessible depuis Internet.

paquet	popcon	taille	outil	description
openssh-client	V:809, I:997	4298	ssh(1)	client de l'interpréteur de commandes sécurisé
openssh-server	V:701, I:844	1567	sshd(8)	serveur de l'interpréteur de commandes sécurisé
ssh-askpass	V:2, I:32	106	ssh-askpass(1)	demande à l'utilisateur une phrase de passe pour ssh-add (X natif)
ssh-askpass-gnome	V:0, I:5	294	ssh-askpass-gnome(1)	saisir une phrase de passe pour ssh-add (GNOME)
ssh-askpass-fullscreen	V:0, I:0	42	ssh-askpass-fullscreen(1)	asks user for a pass phrase for ssh-add (GNOME) with extra eye candy
shellinabox	V:0, I:2	507	shellinaboxd(1)	web server for browser accessible VT100 terminal emulator

Table 6.8 – Liste des serveurs et des utilitaires d'accès à distance

ASTUCE

Utilisez le programme `screen(1)` pour qu'un processus de l'interpréteur de commandes distant survive à une interruption de la connexion (consultez Section [9.1.2](#)).

6.3.1 Bases de SSH

The OpenSSH SSH daemon supports SSH protocol 2 only.

Please read `"/usr/share/doc/openssh-client/README.Debian.gz"`, `ssh(1)`, `sshd(8)`, `ssh-agent(1)`, and `ssh-keygen(1)` and `ssh-add(1)` and `ssh-agent(1)`.

**AVERTISSEMENT**

Il ne faut pas que `« /etc/ssh/sshd_not_to_be_run »` soit présent si l'on souhaite faire tourner le serveur OpenSSH.

Don't enable rhost based authentication (HostbasedAuthentication in `/etc/ssh/sshd_config`).

fichier de configuration	description du fichier de configuration
<code>/etc/ssh/ssh_config</code>	valeurs par défauts des paramètres du client SSH, consultez <code>ssh_config(5)</code> .
<code>/etc/ssh/sshd_config</code>	valeurs par défauts des paramètres du serveur SSH, consultez <code>sshd_config(5)</code> .
<code>~/.ssh/authorized_keys</code>	clés publiques SSH par défaut utilisées pour se connecter à ce compte sur ce serveur SSH
<code>~/.ssh/id_rsa</code>	clé secrète SSH-2 RSA de l'utilisateur
<code>~/.ssh/id_key-type-name</code>	secret SSH-2 <i>key-type-name</i> key such as <code>ecdsa</code> , <code>ed25519</code> , ... of the user

Table 6.9 – Liste des fichiers de configuration de SSH

Ce qui suit permettra de démarrer une connexion `ssh(1)` depuis un client :

6.3.2 User name on the remote host

If you use the same user name on the local and the remote host, you can eliminate typing `"username@"`.

Even if you use different user name on the local and the remote host, you can eliminate it using `~/.ssh/config`. For [Debian Salsa service](#) with account name `"foo-guest"`, you set `~/.ssh/config` to contain the following.

commande	description
ssh nomutilisateur@nommachine..domaine.ext	connexion avec le mode par défaut
ssh -v nomutilisateur@nommachine..domaine.ext	connexion avec le mode par défaut et les messages de débogage
ssh -o PreferredAuthentications=password username@hostname.domain.ext	forcer l'utilisation d'un mot de passe avec SSH version 2
ssh -t username@hostname.domain.ext passwd	run passwd program to update password on a remote host

Table 6.10 – Liste d'exemples de démarrage du client SSH

```
Host salsa.debian.org people.debian.org
User foo-guest
```

6.3.3 Se connecter sans mot de passe distant

One can avoid having to remember passwords for remote systems by using "PubkeyAuthentication" (SSH-2 protocol). On the remote system, set the respective entries, "PubkeyAuthentication yes", in "/etc/ssh/sshd_config". Générez ensuite localement les clés d'identification et installez la clé publique sur le système distant en faisant ce qui suit :

```
$ ssh-keygen -t rsa
$ cat .ssh/id_rsa.pub | ssh user1@remote "cat - >>.ssh/authorized_keys"
```

You can add options to the entries in "~/.ssh/authorized_keys" to limit hosts and to run specific commands. See sshd(8) "AUTHORIZED_KEYS FILE FORMAT".

6.3.4 Clients SSH exotiques

Il existe quelques clients [SSH](#) libres disponibles pour d'autres plateformes.

environnement	programme SSH libre
Windows	puTTY (http://www.chiark.greenend.org.uk/~sgtatham/putty/) (GPL)
Windows (cygwin)	SSH sous cygwin (http://www.cygwin.com/) (GPL)
Macintosh Classic	macSSH (http://www.macssh.com/) (GPL)
Mac OS X	OpenSSH ; utilise ssh dans l'application Terminal (GPL)

Table 6.11 – Liste des clients SSH libres pour d'autres plateformes

6.3.5 Configurer ssh-agent

Il est plus sûr de protéger les clés secrètes de votre authentification SSH avec une phrase de passe. Si la phrase de passe n'a pas été définie, utilisez « ssh-keygen -p » pour le faire.

Placez votre clé publique SSH (par exemple « ~/.ssh/id_rsa.pub ») dans « ~/.ssh/authorized_keys » sur la machine distante en utilisant une connexion basée sur un mot de passe comme décrit ci-dessus.

```
$ 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)
```

Il n'y a plus besoin de mot de passe distant, à partir de maintenant, pour la commande suivante :

```
$ scp foo username@remote.host:foo
```

Pressez `^D` pour quitter la session de l'agent ssh.

Pour le serveur X, le script de démarrage normal de Debian exécute `ssh-agent` comme processus-père. Vous n'aurez donc à exécuter `ssh-add` qu'une seule fois. Pour davantage d'informations, veuillez lire `ssh-agent(1)` et `ssh-add(1)`.

6.3.6 Sending a mail from a remote host

If you have an SSH shell account on a server with proper DNS settings, you can send a mail generated on your workstation as an email genuinely sent from the remote server.

```
$ ssh username@example.org /usr/sbin/sendmail -bm -ti -f "username@example.org" < mail_data ↵
.txt
```

6.3.7 Redirection de port pour un tunnel SMTP/POP3

Pour mettre en place un tube pour se connecter au port 25 du serveur-distant depuis le port 4025 de `localhost`, et au port 110 du serveur-distant depuis le port 4110 de `localhost` au travers de `ssh`, exécutez ce qui suit sur la machine locale :

```
# ssh -q -L 4025:remote-server:25 4110:remote-server:110 username@remote-server
```

C'est une manière sécurisée d'effectuer une connexion à des serveurs SMTP / POP3 par Internet. Définissez l'entrée « `AllowTcpForwarding` » à « `yes` » dans « `/etc/ssh/sshd_config` » sur la machine distante.

6.3.8 Comment arrêter le système distant par SSH

Vous devez protéger le processus qui effectue « `shutdown -h now` » (consultez Section 1.1.8) de l'arrêt de SSH en utilisant la commande `at(1)` (consultez Section 9.4.13) comme suit :

```
# echo "shutdown -h now" | at now
```

Lancer « `shutdown -h now` » dans une `sessionScreen(1)` (consultez Section 9.1.2) est une autre manière d'effectuer la même chose.

6.3.9 Résoudre les problèmes avec SSH

Si vous rencontrez des problèmes, vérifiez les permissions des fichiers de configuration et lancez `ssh` avec l'option « `-v` ».

Utilisez l'option « `-p` » si vous êtes administrateur et que vous rencontrez des problèmes avec un pare-feu. Cela évite l'utilisation des ports 1 —1023 du serveur.

Si les connexions `ssh` vers un site distant s'arrêtent subitement de fonctionner, cela peut être suite à des bidouilles de l'administrateur, le plus probablement un changement de « `host_key` » pendant une maintenance du système. Après s'être assuré que c'est bien le cas et que personne n'essaie de se faire passer pour la machine distante par une habile bidouille, on peut se reconnecter en supprimant sur la machine locale l'entrée « `host_key` » de « `~/.ssh/known_hosts` ».

6.4 Le serveur et les utilitaires d'impression

In the old Unix-like system, the BSD [Line printer daemon \(lpr\)](#) was the standard and the standard print out format of the classic free software was [PostScript \(PS\)](#). Some filter system was used along with [Ghostscript](#) to enable printing to the non-PostScript printer. See Section [11.4.1](#).

In the modern Debian system, the [Common UNIX Printing System](#) (CUPS) is the de facto standard and the standard print out format of the modern free software is [Portable Document Format \(PDF\)](#).

The CUPS uses [Internet Printing Protocol](#) (IPP). The IPP is now supported by other OSs such as Windows XP and Mac OS X and has become new cross-platform de facto standard for remote printing with bi-directional communication capability.

Grâce à la fonctionnalité d'autoconversion dépendante du format du fichier du système CUPS, passer simplement les données à la commande `lpr` devrait créer la sortie imprimable souhaitée. (Dans CUPS, `lpr` peut être activé en installant la paquet `cups-bsd`).

Le système Debian possède certains paquets notables de serveurs et d'utilitaires d'impression :

paquet	popcon	taille	port	description
lpr	V:3, I:4	362	imprimante (515)	BSD lpr/lpd (démon d'impression)
lprng	V:0, I:1	3064	, ,	, , (Amélioré)
cups	V:129, I:376	1134	IPP (631)	Serveur Internet d'impression CUPS
cups-client	V:139, I:434	493	, ,	commandes d'impression System V pour 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:35, I:361	122	, ,	commandes d'impression BSD pour CUPS : <code>lpr(1)</code> , <code>lpq(1)</code> , <code>lprm(1)</code> , <code>lpc(8)</code>
printer-driver-gutenprint	V:55, I:343	941	Non applicable	pilotes d'impression pour CUPS

Table 6.12 – Liste des serveurs et utilitaires d'impression.

ASTUCE

Vous pouvez configurer le système CUPS en pointant votre navigateur web sur « <http://localhost:631/> ».

6.5 Autres serveurs d'applications réseau

Voici d'autres serveurs d'applications réseau :

Le protocole « Common Internet File System Protocol » (CIFS) est le même protocole que [Server Message Block \(SMB\)](#), il est largement utilisé par Microsoft Windows.

ASTUCE

Consultez Section [4.5.2](#) pour l'intégration de systèmes de type serveur.

ASTUCE

La résolution de nom d'hôte est normalement fournie par le serveur [DNS](#). Pour l'affectation dynamique d'adresse IP hôte par [DHCP](#), le [DNS dynamique](#) peut être configuré pour la résolution de nom d'hôte en utilisant `bind9` et `isc-dhcp-server` comme décrit sur la [page DDNS du wiki Debian](#).

paquet	popcon	taille	protocole	description
telnetd	V:1, I:2	111	TELNET	Serveur TELNET
telnetd-ssl	V:0, I:0	169	, ,	, , (prise en charge de SSL)
nfs-kernel-server	V:42, I:78	342	NFS	Partage de fichiers UNIX
samba	V:105, I:155	16630	SMB	Partage de fichiers et d'imprimantes Windows
netatalk	V:2, I:3	2077	ATP	Partage de fichiers et d'imprimantes Apple/Mac (AppleTalk)
proftpd-basic	V:23, I:31	488	FTP	Téléchargement généraliste de fichiers
apache2	V:244, I:310	614	HTTP	Serveur Web généraliste
squid	V:12, I:14	8381	, ,	Serveur mandataire (proxy) web généraliste
squid3	V:3, I:9	240	, ,	, ,
bind9	V:52, I:63	1070	DNS	adresses IP des autres machines
isc-dhcp-server	V:18, I:52	1471	DHCP	adresse IP du client lui-même

Table 6.13 – Liste d'autres serveurs d'applications réseau

ASTUCE

L'utilisation d'un serveur mandataire tel que [squid](#) est bien plus efficace pour économiser de la bande passante que l'utilisation d'un serveur miroir local comportant tout le contenu de l'archive Debian.

6.6 Autres clients d'applications réseau

Voici d'autres clients d'applications réseau :

6.7 Le diagnostic des démons du système

Le programme `telnet` permet la connexion manuelle aux démons du système et leur diagnostic.

Pour tester le service [POP3](#) brut, essayez ce qui suit :

```
$ telnet mail.ispname.net pop3
```

Pour tester le service [POP3](#), ayant [TLS](#)/SSL activé, de certains fournisseurs d'accès Internet (FAI), vous devrez avoir un client `telnet` ayant TLS/SSL activé en utilisant l'un des paquets `telnet-ssl` ou `openssl`.

```
$ telnet -z ssl pop.gmail.com 995
```

```
$ openssl s_client -connect pop.gmail.com:995
```

Les [RFC](#) suivantes proposent les connaissances nécessaires pour chaque démon :

L'utilisation des ports est décrite dans « `/etc/services` ».

paquet	popcon	taille	protocole	description
netcat	I:43	16	TCP/IP	couteau de l'armée Suisse pour TCP/IP
openssl	V:814, I:994	1465	SSL	binaire Secure Socket Layer (SSL) et outils de chiffrement associés
stunnel4	V:6, I:16	520	, ,	enrobeur SSL universel
telnet	V:61, I:901	163	TELNET	Client TELNET
telnet-ssl	V:0, I:3	209	, ,	, , (prise en charge de SSL)
nfs-common	V:182, I:323	768	NFS	Partage de fichiers UNIX
smbclient	V:16, I:174	2017	SMB	Client de partage de fichiers et imprimantes MS Windows
cifs-utils	V:29, I:123	307	, ,	commande de montage et de démontage de fichiers MS Windows distants
ftp	V:16, I:251	137	FTP	Client FTP
lftp	V:6, I:40	2255	, ,	, ,
ncftp	V:3, I:21	1339	, ,	client FTP plein écran
wget	V:255, I:987	3477	HTTP et FTP	téléchargement web
curl	V:160, I:575	427	, ,	, ,
axel	V:0, I:4	216	, ,	accélérateur de téléchargement
aria2	V:2, I:19	1854	, ,	accélérateur de téléchargement avec prise en charge de BitTorrent et Metalink
bind9-host	V:176, I:946	370	DNS	host(1) de bind9, « Priority: standard »
dnsutils	V:59, I:490	261	, ,	dig(1) de bind, « Priority: standard »
isc-dhcp-client	V:222, I:980	686	DHCP	obtenir une adresse IP
ldap-utils	V:12, I:74	719	LDAP	obtenir des données d'un serveur LDAP

Table 6.14 – Liste de clients d'applications réseau

RFC	description
rfc1939 et rfc2449	service POP3
rfc3501	service IMAP4
rfc2821 (rfc821)	service SMTP
rfc2822 (rfc822)	Format de fichier de courrier électronique
rfc2045	Extensions multifonctions du courrier Internet « Multipurpose Internet Mail Extensions (MIME) »
rfc819	service DNS
rfc2616	service HTTP
rfc2396	définition d'une URI

Table 6.15 – Liste des RFC courantes

Chapitre 7

GUI System

7.1 GUI desktop environment

There are several choices for the full featured [GUI](#) desktop environment on the Debian system.

task package	popcon	taille	description
task-gnome-desktop	1:166	9	GNOME desktop environment
task-xfce-desktop	1:102	9	Xfce desktop environment
task-kde-desktop	1:65	6	KDE Plasma desktop environment
task-mate-desktop	1:47	9	MATE desktop environment
task-cinnamon-desktop	1:36	9	Cinnamon desktop environment
task-lxde-desktop	1:33	9	LXDE desktop environment
task-lxqt-desktop	1:13	9	LXQt desktop environment
task-gnome-flashback-desktop	1:0	6	GNOME Flashback desktop environment

Table 7.1 – List of desktop environment

ASTUCE

Dependency packages selected by a task metapackage may be out of sync with the latest package transition state under the Debian unstable/testing environment. For `task-gnome-desktop`, you may need to adjust package selections as follows:

- Start `aptitude(8)` as `sudo aptitude -u`.
 - Move cursor to "Tasks" and press "Enter".
 - Move cursor to "End-user" press "Enter".
 - Move cursor to "GNOME" press "Enter".
 - Move cursor to `task-gnome-desktop` and press "Enter".
 - Move cursor to "Depends" and press "m" (manually selected).
 - Move cursor to "Recommends" and press "m" (manually selected).
 - Move cursor to `task-gnome-desktop` and press "-". (drop)
 - Adjust selected packages while dropping problematic ones causing package conflicts.
 - Press "g" to start install.
-

This chapter will focus mostly on the default desktop environment of Debian: `task-gnome-desktop` offering [GNOME](#) on [wayland](#).

7.2 GUI communication protocol

GUI communication protocol used on the GNOME desktop can be:

- [Wayland \(display server protocol\)](#) (native)
- [X Window System core protocol](#) (via `xwayland`)

Please check freedesktop.org site for how [Wayland architecture](#) is different from [X Window architecture](#).

From user's perspective, differences can be colloquially summarized as:

- Wayland is a same-host GUI communication protocol: new, simpler, faster, no `setuid` root binary
- X Window is a network-capable GUI communication protocol: traditional, complex, slower, `setuid` root binary

For applications using Wayland protocol, the access to their display contents from a remote host is supported by the [VNC](#) or [RDP](#). See [Section 7.7](#)

Modern X servers have [the MIT Shared Memory Extension](#) and communicate with their local X clients using the local shared memory. This bypasses the network transparent [Xlib](#) interprocess communication channel and gains performance. This situation was the [background](#) of creating Wayland as a local-only GUI communication protocol.

Using the `xeyes` program started from the GNOME terminal, you can check GUI communication protocol used by each GUI application.

```
$ xeyes
```

- If the mouse cursor is on an application such as "GNOME terminal" which uses Wayland display server protocol, eyes don't move with the mouse cursor.
- If the mouse cursor is on an application such as "xterm" which uses X Window System core protocol, eyes move with the mouse cursor exposing not-so-isolated nature of X Window architecture.

As of April 2021, many popular GUI applications such as GNOME and [LibreOffice \(LO\)](#) applications have been migrated to the Wayland display server protocol. I see `xterm`, `gitk`, `chromium`, `firefox`, `gimp`, `dia`, and KDE applications still use X Window System core protocol.

Note

For both the xwayland on Wayland or the native X Window System, the old X server configuration file `/etc/X11/xorg.conf` shouldn't exist on the system. The graphics and input devices are now configured by the kernel with [DRM](#), [KMS](#), and [udev](#). The native X server has been rewritten to use them. See "[modedb default video mode support](#)" in the Linux kernel documentation.

7.3 GUI infrastructure

Here are notable GUI infrastructure packages for the GNOME on Wayland environment.

paquet	popcon	taille du pa- quet	description
mutter	V:7, I:204	208	GNOME's mutter window manager [auto]
xwayland	V:137, I:227	4572	An X server running on top of wayland [auto]
gnome-remote-desktop	V:14, I:23	207	Remote desktop daemon for GNOME using PipeWire [auto]
gnome-tweaks	V:14, I:169	1264	Advanced configuration settings for GNOME

Table 7.2 – List of notable GUI infrastructure packages

Here, "**[auto]**" means that these packages are automatically installed when `task-gnome-desktop` is installed.

ASTUCE

`gnome-tweaks` is the indispensable configuration utility. For example:

- You can force "Over-Amplification" of sound volume from "General".
- You can force "Caps" to become "Esc" from "Keyboard & Mouse" -> "Keyboard" -> "Additional Layout Option".

7.4 GUI applications

Many useful GUI applications are available on Debian now. Installing software packages such as `scribus` (KDE) on GNOME desktop environment are quite acceptable since corresponding functionality is not available under GNOME desktop environment. But installing too many packages with duplicated functionalities may clutter your system.

Here is a list of GUI applications which caught my eyes.

7.5 Fonts

Many useful scalable fonts are available for users on Debian. User's concern is how to avoid redundancy and how to configure parts of installed fonts to be disabled. Otherwise, useless font choices may clutter your GUI application menus.

Debian system uses [FreeType](#) 2.0 library to rasterise many scalable font formats for screen and print:

- [Type 1 \(PostScript\) fonts](#) which use cubic [Bézier curves](#) (almost obsolete format)
- [TrueType fonts](#) which use quadratic [Bézier curves](#) (good choice format)
- [OpenType fonts](#) which use cubic [Bézier curves](#) (best choice format)

paquet	popcon	taille du pa- quet	type	description
evolution	V:29, I:219	475	GNOME	Gestion d'informations personnelles (logiciel de travail collaboratif et courriel)
thunderbird	V:57, I:134	168690	GTK	Email client (Mozilla Thunderbird)
kontakt	V:1, I:14	2152	KDE	Gestion d'informations personnelles (logiciel de travail collaboratif et courriel)
libreoffice-writer	V:167, I:423	39325	LO	traitement de texte
abiword	V:1, I:11	5133	GNOME	traitement de texte
calligrawords	V:0, I:7	5717	KDE	traitement de texte
scribus	V:1, I:22	30370	KDE	desktop publishing editor to edit PDF files
glabels	V:0, I:4	1326	GNOME	éditeur d'étiquettes
libreoffice-calc	V:166, I:418	32969	LO	feuille de calcul
gnumeric	V:5, I:20	9933	GNOME	feuille de calcul
calligrasheets	V:0, I:5	10890	KDE	feuille de calcul
libreoffice-impress	V:149, I:415	9930	LO	présentation
calligrastage	V:0, I:5	5102	KDE	présentation
libreoffice-base	V:116, I:289	7465	LO	gestion de base de données
kexi	V:0, I:2	7576	KDE	gestion de base de données
libreoffice-draw	V:150, I:416	14596	LO	éditeur de graphiques vectoriels (draw)
inkscape	V:42, I:188	84892	GNOME	éditeur de graphiques vectoriels (draw)
karbon	V:0, I:6	3473	KDE	éditeur de graphiques vectoriels (draw)
dia	V:4, I:30	3727	GTK	éditeur d'organigrammes et de diagrammes
gimp	V:58, I:315	19706	GTK	éditeur de graphiques en champs de bits (« bitmap ») (paint)
shotwell	V:16, I:217	6451	GTK	gestionnaire de photos numériques
digikam	V:2, I:12	2646	KDE	gestionnaire de photos numériques
darktable	V:6, I:17	22902	GTK	lighttable and darkroom for photographers
planner	V:0, I:4	1146	GNOME	gestion de projets
calligraplan	V:0, I:2	18517	KDE	gestion de projets
gnucash	V:3, I:11	32304	GNOME	gestion financière personnelle
homebank	V:0, I:2	1114	GTK	gestion financière personnelle
lilypond	V:0, I:8	7337	-	music typesetter
kmy money	V:0, I:2	12036	KDE	gestion financière personnelle
librecad	V:3, I:18	7893	Qt-app	computer-aided design (CAD) system (2D)
freecad	I:19	53	Qt-app	computer-aided design (CAD) system (3D)
kicad	V:2, I:15	98707	GTK	electronic schematic and PCB design software
xsane	V:14, I:165	2346	GTK	interface pour dispositifs de numérisation (scanner)
libreoffice-math	V:144, I:419	1963	LO	éditeur d'équations et de formules mathématiques
calibre	V:8, I:35	55048	KDE	Convertisseur de livre numérique et gestion de bibliothèque.
fbreader	V:1, I:14	3074	GTK	lectrice de e-book
evince	V:105, I:313	956	GNOME	afficheur de documents (pdf)
okular	V:43, I:115	14646	KDE	afficheur de documents (pdf)
x11-apps	V:27, I:454	2437	pure X-app	xeyes (1), etc.
x11-utils	V:171, I:581	712	pure X-app	xev (1), xwininfo (1)etc.

Table 7.3 – List of notable GUI applications

7.5.1 Polices de base

The following table is compiled in the hope to help users to chose appropriate scalable fonts with clear understanding of the metric compatibility and the glyph coverage. Most fonts cover all Latin fonts, Greek, and Cyril character glyphs. The final choice of activated fonts can also be affected by your aesthetics. These fonts can be used for the screen display or for the paper printing.

paquet	popcon	taille	sans	serif	mono	note on font
fonts-cantarell	V:37, I:273	836	59	-	-	Cantarell (GNOME 3, display)
fonts-noto	I:133	35	61	63	40	Noto fonts (Google, multi-lingual with CJK)
fonts-dejavu	I:462	39	58	68	40	DejaVu (GNOME 2, MCM: Verdana , extended Bitstream Vera)
fonts-liberation2	V:48, I:333	4261	56	60	40	Liberation fonts for LibreOffice (Red Hat, MCMATC)
fonts-croscore	V:10, I:47	5278	56	60	40	Chrome OS: Arimo, Tinos and Cousine (Google, MCMATC)
fonts-crosextra-carlito	I:276	2732	57	-	-	Chrome OS: Carlito (Google, MCM: Calibri)
fonts-crosextra-caladea	I:275	258	-	55	-	Chrome OS: Caladea (Google, MCM: Cambria) (Latin only)
fonts-freefont-ttf	V:55, I:257	6656	57	59	40	GNU FreeFont (extended URW Nimbus)
fonts-quicksand	I:358	392	56	-	-	Debian task-desktop, Quicksand (display, Latin only)
fonts-hack	V:9, I:89	2508	-	-	40 P	A typeface designed for source code Hack (Facebook)
fonts-sil-gentiumplus	I:39	5756	-	54	-	Gentium SIL
fonts-sil-charis	V:2, I:12	8900	-	59	-	Charis SIL
fonts-urw-base35	V:46, I:94	11095	56	60	40	URW Nimbus (Sans , Roman No. 9 L , Mono L , MCAHTC)
fonts-ubuntu	V:1, I:5	4103	58	-	33 P	Ubuntu fonts (display)
fonts-terminus	V:0, I:1	450	-	-	33	Cool retro terminal fonts
ttf-mscorefonts-installer	V:1, I:61	92	56?	60	40	Downloader of Microsoft non-free fonts (see below)

Table 7.4 – List of notable [TrueType](#) and [OpenType](#) fonts

Here:

- "MCM" stands for "metric compatible with fonts provided by Microsoft"
- "MCMATC" stands for "metric compatible with fonts provided by Microsoft: [Arial](#), [Times New Roman](#), [Courier New](#)"
- "MCAHTC" stands for "metric compatible with fonts provided by [Adobe](#): Helvetica, Times, Courier"
- Numbers in font type columns stands for the rough relative "M" width for the same point size font.
- "P" in mono font type columns stands for its usability for programming having clearly distinguishable "0"/"O" and "1"/"l"/"I".
- The `ttf-mscorefonts-installer` package downloads Microsoft's "[Core fonts for the Web](#)" and installs [Arial](#), [Times New Roman](#), [Courier New](#), [Verdana](#), These installed font data are non-free data.

Many free Latin fonts have their lineage traced to [URW Nimbus](#) family or [Bitstream Vera](#).

ASTUCE

If your locale needs fonts not covered well by the above fonts, please use aptitude to check under task packages listed under "Tasks" -> "Localization". The font packages listed as "Depends:" or "Recommends:" in the localization task packages are the primary candidates.

7.5.2 Font rasterization

Debian uses [FreeType](#) to rasterize fonts. Its font choice infrastructure is provided by the [Fontconfig](#) font configuration library.

paquet	popcon	taille	description
libfreetype6	V:432, I:995	896	FreeType font rasterization library
libfontconfig1	V:432, I:860	526	Fontconfig font configuration library
fontconfig	V:353, I:757	583	<code>fc - *</code> : CLI commands for Fontconfig
font-manager	V:1, I:9	1015	Font Manager : GUI command for Fontconfig
nautilus-font-manager	V:0, I:0	37	Nautilus extension for Font Manager

Table 7.5 – List of notable font environment and related packages

ASTUCE

Some font packages such as `fonts-noto*` install too many fonts. You may also want to keep some font packages installed but disabled under the normal use situation. The multiple [glyphs](#) are expected for some [Unicode](#) code points due to [Han unification](#) and unwanted glyphs may be chosen by the unconfigured [Fontconfig](#) library. One of the most annoying case is "U+3001 IDEOGRAPHIC COMMA" and "U+3002 IDEOGRAPHIC FULL STOP" among CJK countries. You can avoid this problematic situation easily by configuring font availability using [Font Manager](#) GUI ([font-manager](#)).

You can list font configuration state from the command line, too.

- `"fc-match(1)"` for fontconfig font default
- `"fc-list(1)"` for available fontconfig fonts

You can configure font configuration state from the text editor but this is non-trivial. See `fonts.conf(5)`.

7.6 Sandbox

Many mostly GUI applications on Linux are available in binary formats from non-Debian sources.

- [AppImage -- Linux apps that run anywhere](#)
- [FLATHUB -- Apps for Linux, right here](#)
- [snapcraft -- The app store for Linux](#)



AVERTISSEMENT

Binaries from these sites may include proprietary non-free software packages.

There is some *raison d'être* for these binary format distributions for Free Software aficionados using Debian since these can accommodate clean set of libraries used for each application by the respective upstream developer independent of the ones provided by Debian.

The inherent risk of running external binaries can be reduced by using the [sandbox environment](#) which leverages modern Linux security features (see Section [4.7.4](#)).

- For binaries from AppImage and some upstream sites, run them in [firejail](#) with [manual configuration](#).
- For binaries from FLATHUB, run them in [Flatpak](#) . (No manual configuration required.)
- For binaries from snapcraft, run them in [Snap](#) . (No manual configuration required. Compatible with daemon programs.)

The `xdg-desktop-portal` package provides a standardized API to common desktop features. See [xdg-desktop-portal \(flatpak\)](#) and [xdg-desktop-portal \(snap\)](#)

paquet	popcon	taille	description
flatpak	V:31, I:35	6562	Flatpak application deployment framework for desktop apps
gnome-software-plugin-flatpak	V:4, I:14	335	Flatpak support for GNOME Software
snapd	V:53, I:58	55035	Daemon and tooling that enable snap packages
gnome-software-plugin-snap	V:0, I:2	237	Snap support for GNOME Software
xdg-desktop-portal	V:125, I:209	1425	desktop integration portal for Flatpak and Snap
xdg-desktop-portal-gtk	V:71, I:205	921	xdg-desktop-portal backend for gtk (GNOME)
xdg-desktop-portal-kde	V:1, I:4	842	xdg-desktop-portal backend for Qt (KDE)
xdg-desktop-portal-wlr	V:0, I:0	83	xdg-desktop-portal backend for wlroots (Wayland)
firejail	V:1, I:5	1370	a SUID security sandbox program firejail for use with AppImage

Table 7.6 – List of notable sandbox environment and related packages

This sandbox environment technology is very much like apps on smart phone OS where apps are executed under controlled resource accesses.

Some large GUI applications such as web browsers on Debian also use sandbox environment technology internally to make them more secure.

7.7 Remote desktop

7.8 X server connection

There are several ways to connect from an application on a remote host to the X server including `xwayland` on the local host.

7.8.1 X server local connection

Access to the local X server by the local applications which use X core protocol can be locally connected through a local UNIX domain socket. This can be authorized by the authority file holding [access cookie](#). The authority file location is identified by the `"$XAUTHORITY"` environment variable and X display is identified by the `"$DISPLAY"` environment variable. Since these are normally set automatically, no special action is needed, e.g. `"gitk"` as the following.

```
username $ gitk
```

paquet	popcon	taille	protocols	description
gnome-remote-desktop	V:14, I:23	207	RDP, RFB (VNC)	GNOME Remote Desktop server
vinagre	V:9, I:222	4249	RDP, RFB (VNC), SPICE, SSH	Vinagre: GNOME remote desktop client
remmina	V:12, I:47	550	RDP, RFB (VNC), SPICE, SSH, ...	Remmina: GTK remote desktop client
krdc	V:2, I:21	3538	RDP, RFB (VNC)	KRDC: KDE remote desktop client
guacd	V:0, I:0	83	RDP, RFB (VNC), SSH / HTML5	Apache Guacamole: clientless remote desktop gateway (HTML5)
virt-viewer	V:4, I:51	1545	RFB (VNC), SPICE	Virtual Machine Manager's GUI display client of guest OS

Table 7.7 – List of notable remote access server

paquet	popcon	taille	commande	description
openssh-server	V:701, I:844	1567	sshd with option X11-forwarding	SSH server (secure)
openssh-client	V:809, I:997	4298	ssh -X	SSH client (secure)
xauth	V:155, I:951	86	xauth	X authority file utility
x11-xserver-utils	V:272, I:516	511	xhost	server access control for X

Table 7.8 – Liste des méthodes de connexion au serveur X

Note

For xwayland, XAUTHORITY holds value like `"/run/user/1000/.mutter-Xwaylandauth.YVSU30"`.

7.8.2 X server remote connection

Access to the local X server display from the remote applications which use X core protocol is supported by using the X11 forwarding feature.

- Open an `gnome-terminal` on the local host.
- Run `ssh(1)` with `-X` option to establish a connection with the remote site as the following.

```
localname @ localhost $ ssh -q -X loginname@remotehost.domain
Password:
```

- Run an X application command, e.g. `"gitk"`, on the remote site as the following.

```
loginname @ remotehost $ gitk
```

Cette méthode permet l’affichage du client X distant comme s’il était connecté par une socket UNIX locale.

See Section [6.3](#) for SSH/SSHD.

**AVERTISSEMENT**

A remote [TCP/IP](#) to the X server is disabled by default on the Debian system for security reasons. Don’t enable them by simply setting `"xhost +"` nor by enabling [XDMCP connection](#), if you can avoid it.

7.8.3 X server chroot connection

Access to the X server by the applications which use X core protocol and run on the same host but in an environment such as chroot where the authority file is not accessible, can be authorized securely with `xhost` by using the [User-based access](#), e.g. `"gitk"` as the following.

```
username $ xhost + si:localuser:root ; sudo chroot /path/to
# cd /src
# gitk
# exit
username $ xhost -
```

7.9 Presse-papier

For clipping text to clipboard, see Section [1.4.4](#).

For clipping graphics to clipboard, see Section [11.6](#).

Some CLI commands can manipulate character clipboard (PRIMARY and CLIPBOARD), too.

paquet	popcon	taille du pa- quet	cible	description
xsel	V:9, I:43	59	X	command line interface to X selections (clipboard)
xclip	V:10, I:51	64	X	command line interface to X selections (clipboard)
wl-clipboard	V:0, I:1	129	Wayland	wl-copy wl-paste : command line interface to Wayland clipboard
gpm	V:11, I:16	540	Linux console	a daemon that captures mouse events on Linux console

Table 7.9 – List of programs related to manipulating character clipboard

Chapitre 8

I18N et L10N

Le [multilinguisme \(M17N\)](#) ou la [gestion de la langue natale \(« Native Language Support »\)](#) d'un logiciel applicatif est réalisé en deux étapes :

- L'internationalisation (I18N) : donne la possibilité à un logiciel de gérer plusieurs paramètres linguistiques ;
- La localisation (L10N) : permet au logiciel de prendre en charge des paramètres linguistiques particuliers.

ASTUCE

There are 17, 18, or 10 letters between "m" and "n", "i" and "n", or "l" and "n" in multilingualization, internationalization, and localization which correspond to M17N, I18N, and L10N. See [Introduction to i18n](#) for details.

8.1 Les paramètres linguistiques (« locale »)

The behavior of programs supporting internationalization are configured by the environment variable `"$LANG"` to support localization. Actual support of locale dependent features by the `libc` library requires to install `locales` or `locales-all` packages. The `locales` package requires to be initialized properly.

If neither `locales` or `locales-all` package are installed, support of locale features are lost and system uses US English messages and handles data as `ASCII`. This behavior is the same way as `"$LANG"` is set by `"LANG="`, `"LANG=C"`, or `"LANG=POSIX"`.

The modern software such as GNOME and KDE are multilingualized. They are internationalized by making them handle [UTF-8](#) data and localized by providing their translated messages through the `gettext(1)` infrastructure. Translated messages may be provided as separate localization packages.

The current Debian desktop GUI system normally sets the locale under GUI environment as `"LANG=xx_YY.UTF-8"`. Here, `"xx"` is [ISO 639 language codes](#) and `"YY"` is [ISO 3166 country codes](#). These values are set by the desktop configuration GUI dialogue and change the program behavior. See [Section 1.5.2](#)

8.1.1 Justification de l'utilisation d'UTF-8 dans les paramètres linguistiques

The simplest representation of the text data is `ASCII` which is sufficient for English and uses less than 127 characters (representable with 7 bits).

Même le texte anglais en texte brut peut contenir des caractères non-ASCII, par exemple les guillemets apostrophes culbutés gauches et droits ne sont pas disponibles en ASCII.

```
b'' "b''double quoted textb''" b'' is not "double quoted ASCII"
b'' 'b''single quoted textb'''b'' is not 'single quoted ASCII'
```

In order to support more characters, many character sets and encoding systems have been used to support many languages (see [Tableau 11.2](#)).

[Unicode](#) character set can represent practically all characters known to human with 21 bit code point range (i.e., 0 to 10FFFF in hexadecimal notation).

Text encoding system [UTF-8](#) fits Unicode code points into a sensible 8 bit data stream mostly compatible with the ASCII data processing system. This makes **UTF-8** the modern preferred choice. **UTF** stands for Unicode Transformation Format. When [ASCII](#) plain text data is converted to [UTF-8](#) one, it has exactly the same content and size as the original ASCII one. So you loose nothing by deploying UTF-8 locale.

Under [UTF-8](#) locale with the compatible application program, you can display and edit any foreign language text data as long as required fonts and input methods are installed and enabled. For example under "`LANG=fr_FR.UTF-8`" locale, `gedit(1)` (text editor for the GNOME Desktop) can display and edit Chinese character text data while presenting menus in French.

ASTUCE

Both the new standard "`en_US.UTF-8`" locale and the old standard "`C`" / "`POSIX`" locale use the standard US English message, they have subtle differences in sorting order etc. If you want to handle not only ASCII characters but also handle all UTF-8 encoded characters gracefully while maintaining the old "`C`" local behavior, use the non-standard "`C.UTF-8`" locale on Debian.

Note

Certains programmes utilisent davantage de mémoire lors de l'utilisation de l18N. Cela parce qu'ils sont codés avec l'utilisation interne d'[UTF-32\(UCS4\)](#) pour la prise en compte d'Unicode afin d'optimiser la vitesse, ils utilisent 4 octets pour chaque caractère ASCII indépendamment de la « locale » sélectionnée. De nouveau, il n'y a rien à perdre en mettant en œuvre des paramètres linguistiques UTF-8.

8.1.2 Reconfiguration des paramètres linguistiques

In order for the system to access a particular locale, the locale data must be compiled from the locale database.

The `locales` package does **not** come with pre-compiled locale data. You need to configure it as:

```
# dpkg-reconfigure locales
```

This process involves 2 steps.

1. Select all required locale data to be compiled into the binary form. (Please make sure to include at least one UTF-8 locale)
2. Set the system wide default locale value by creating "`/etc/default/locale`" for use by PAM (see [Section 4.5](#)).

The system wide default locale value set in "`/etc/default/locale`" may be overridden by the GUI configuration for GUI applications.

Note

Actual traditional encoding system can be identified by "`/usr/share/i18n/SUPPORTED`". Thus, the "`LANG=en_US`" is "`LANG=en_US.ISO-8859-1`".

The `locales-all` package comes with all locale data pre-compiled but doesn't creating "`/etc/default/locale`".

8.1.3 Coder les noms de fichiers

Pour les échanges de données entre plateformes (consultez [Section 10.1.7](#)), il vous faudra peut-être monter certains systèmes de fichiers ayant un codage particulier. Par exemple, la commande `mount(8)` pour un [système de fichiers vfat](#) suppose que l'on utilise [CP437](#) si on l'utilise sans option. Vous devrez fournir les options explicites à `mount` pour utiliser des noms de fichiers codés en [UTF-8](#) ou en [CP932](#).

Note

Lors du montage automatique d'un clé USB pouvant être branchée à chaud sur un environnement de bureau moderne comme GNOME, vous pouvez indiquer ces options de montage avec un clic-droit sur l'icône du bureau. Cliquez l'onglet « Drive », cliquez « Paramètres » pour l'étendre et entrez « utf8 » dans « Options de montage ». La prochaine fois que vous monterez cette clé mémoire, le montage avec UTF-8 sera activé.

Note

Si vous êtes en train de mettre à jour le système ou de déplacer des disques depuis un ancien système qui n'était pas UTF-8, les noms de fichiers avec des caractères non ASCII peuvent être codés avec des codages historiques et obsolètes tels que [ISO-8859-1](#) ou [eucJP](#). Veuillez consulter l'aide des outils de conversion de texte pour les convertir en [UTF-8](#). Consultez Section [11.1](#).

[Samba](#) utilise Unicode pour les clients les plus récents (Windows NT, 200x, XP) mais utilise par défaut [CP850](#) pour des clients plus anciens (DOS et Windows 9x/Me). Cette valeur par défaut pour les anciens clients peut être modifiée en utilisant « dos charset » dans le fichier « `/etc/samba/smb.conf` », par exemple, avec [CP932](#) pour le japonais.

8.1.4 Messages et documentation traduits

Il existe des traductions de nombreux messages et documents affichés par le système Debian, comme les messages d'erreur, la sortie standard des programmes, les menus et les pages de manuel. La [GNU gettext\(1\) command tool chain](#) est utilisé comme outil de base pour la plupart des activités de traduction.

Dans « Tâches » → « Localisation » [aptitude\(8\)](#) fournit une liste exhaustive de paquets binaires utiles qui ajoutent les traductions de messages aux applications et fournissent de la documentation traduite.

Vous pouvez, par exemple, obtenir les messages traduits pour une page de manuel en installant le paquet `manpages-LANG`. Pour lire la page de manuel de `nom_programme` en italien depuis « `/usr/share/man/it/` », lancez le programme de la manière suivante :

```
LANG=it_IT.UTF-8 man programname
```

GNU gettext can accommodate priority list of translation languages with `$LANGUAGE` environment variable. For example:

```
$ export LANGUAGE="pt:pt_BR:es:it:fr"
```

For more, see `info gettext` and read the section "The LANGUAGE variable".

8.1.5 Effet des paramètres linguistiques

L'ordre de tri des caractères avec `sort(1)` est affecté par le choix de la langue des paramètres linguistiques. Les paramètres linguistiques espagnol et anglais effectuent le tri de manière différente.

Le format de date de `ls(1)` est affecté par les paramètres linguistiques. Les formats de date de « `LANG=C ls -l` » et « `LANG=en_US.U` » sont différents (consultez Section [9.3.4](#)).

La ponctuation des nombres est différente selon les paramètres linguistiques. Par exemple, avec les paramètres linguistiques anglais, Mille un est affiché sous la forme « `1, 000.1` » alors qu'avec les paramètres linguistiques allemands ce sera « `1. 000, 1` ». Vous pouvez voir cette différence dans un programme de feuille de calculs.

Each detail feature of "`$LANG`" environment variable may be overridden by setting "`$LC_*`" variables. These environment variables can be overridden again by setting "`$LC_ALL`" variable. See `locale(7)` manpage for the details. Unless you have strong reason to create complicated configuration, please stay away from them and use only "`$LANG`" variable set to one of the UTF-8 locales.

8.2 L'entrée clavier

8.2.1 The keyboard input for Linux console and X Window

Le système Debian peut être configuré pour travailler avec de nombreuses configurations de clavier internationales en utilisant les paquets `keyboard-configuration` et `console-setup`.

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

For the Linux console and the X Window system, this updates configuration parameters in `/etc/default/keyboard` and `/etc/default/console-setup`. This also configures the Linux console font. Many non-ASCII characters including accented characters used by many European languages can be made available with [dead key](#), [AltGr key](#), and [compose key](#).

8.2.2 The keyboard input for Wayland

For GNOME on Wayland desktop system, Section [8.2.1](#) can't support non-English European languages. [IBus](#) was made to support not only Asian languages but also European languages. The package dependency of GNOME Desktop Environment recommends "ibus" via "gnome-shell". The code of "ibus" has been updated to integrate `setxkbmap` and XKB option functionalities. You need to configure `ibus` from "GNOME Settings" or "GNOME Tweaks" for the multilingualized keyboard input.

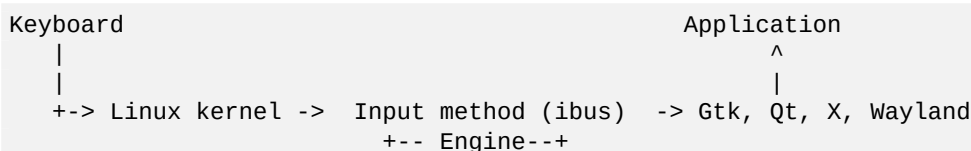
Note

If `ibus` is active, your classic X keyboard configuration by the `setxkbmap` may be overridden by `ibus` even under classic X-based desktop environment. You can disable installed `ibus` using `im-config` to set input method to "None". For more, see [Debian Wiki on keyboard](#).

8.2.3 Prise en charge de la méthode d'entrée avec iBus

Since GNOME Desktop Environment recommends "ibus" via "gnome-shell", "ibus" is the good choice for input method.

Les entrées multilingues dans une application sont traitées comme suit :



The list of IBus and its engine packages are the following.

Note

For Chinese, "fcitx" may be an alternative input method framework. For Emacs aficionados, "uim" may be an alternative. Either cases, you may need to do extra manual configuration with `im-config`. Some old classic [input methods](#) such as "kinput2" may still exist in Debian repository but are not recommended for the modern environment.

8.2.4 Un exemple pour le japonais

I find the Japanese input method started under English environment ("en_US.UTF-8") very useful. Here is how I did this with IBus for GNOME on Wayland:

1. Install the Japanese input tool package `ibus-mozc` (or `ibus-anthy`) with its recommended packages such as `im-config`.
-

paquet	popcon	taille	paramètres linguistiques pris en charge
ibus	V:82, I:99	1584	infrastructure de méthode d'entrée utilisant dbus
ibus-mozc	V:1, I:2	1006	Japonais
ibus-anthy	V:1, I:1	8790	, ,
ibus-kkc	V:0, I:0	214	, ,
ibus-skk	V:0, I:0	244	, ,
ibus-pinyin	V:0, I:1	1434	Chinois (pour zh_CN)
ibus-chewing	V:0, I:0	415	, , (pour zh_TW)
ibus-hangul	V:0, I:1	288	Coréen
ibus-table	V:0, I:1	1809	moteur de tables pour iBus
ibus-table-thai	I:0	47	Thaï
ibus-unikey	V:0, I:0	318	Vietnamien
ibus-m17n	V:0, I:1	187	Multilingue : Indien, Arabe et autres

Table 8.1 – List of IBus and its engine packages

2. Select "Settings" → "Keyboard" → "Input Sources" → click "+" in "Input Sources" → "Japanese" → "Japanese mozc (or anthy)" and click "Add" if it hasn't been activated.
3. You may chose as many input sources.
4. Reconnectez-vous au compte utilisateur.
5. Setup each input source by right clicking the GUI toolbar icon.
6. Choisissez parmi les sources d'entrée installées avec SUPER+ESPACE. (en général, SUPER désigne la touche Windows)

ASTUCE

If you wish to have access to alphabet only keyboard environment with the physical Japanese keyboard on which shift-2 has " (double quotation mark) engraved, you select "Japanese" in the above procedure. You can enter Japanese using "Japanese mozc (or anthy)" with physical "US" keyboard on which shift-2 has @ (at mark) engraved.

- The GUI menu entry for `im-config(8)` is "Input method".
- Alternatively, execute "`im-config`" from user's shell.
- `im-config(8)` se comporte différemment selon que la commande est exécutée depuis le compte de l'administrateur ou non.
- `im-config(8)` active la meilleure méthode de saisie sur le système par défaut sans intervention de l'utilisateur.

8.3 L'affichage de sortie

Linux console can only display limited characters. (You need to use special terminal program such as `jfbterm(1)` to display non-European languages on the non-GUI console.)

GUI environment (Chapitre 7) can display any characters in the UTF-8 as long as required fonts are installed and enabled. (The encoding of the original font data is taken care and transparent to the user.)

8.4 East Asian Ambiguous Character Width Characters

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](#)).

You can work around this problem:

- `gnome-terminal`: Preferences → Profiles → *Profile name* → Compatibility → Ambiguous-wide characters → Narrow
- `ncurses` : paramètre l'environnement `export NCURSES_NO_UTF8_ACS=0`.

Chapitre 9

Astuces du système

Je décris ici les astuces de base pour configurer et gérer les systèmes, la plupart depuis la console.

9.1 The console tips

There are some utility programs to help your console activities.

paquet	popcon	taille	description
mc	V:55, I:234	1492	See Section 1.3
bsdutils	V:708, I:999	394	<code>script</code> command to make typescript of terminal session
screen	V:114, I:283	1014	multiplexeur de terminal avec une émulation de terminal VT100/ANSI
tmux	V:35, I:143	830	terminal multiplexer alternative (Use "Control-B" instead)
fzf	V:2, I:7	2637	fuzzy text finder
fzy	V:0, I:0	54	fuzzy text finder
rlwrap	V:1, I:20	297	readline feature command line wrapper
ledit	V:0, I:14	306	readline feature command line wrapper
rlfe	V:0, I:0	49	readline feature command line wrapper

Table 9.1 – List of programs to support console activities

9.1.1 Enregistrer proprement l'activité de la console

La simple utilisation de `script(1)` (consultez Section [1.4.9](#)) pour enregistrer l'activité de l'interpréteur de commandes produit un fichier avec des caractères de contrôle. Cela peut être évité en utilisant `col(1)` comme suit :

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

faites quelque chose...et pressez `Ctrl-D` pour quitter `script`.

```
$ col -bx < typescript > cleanedfile
$ vim cleanedfile
```

There are alternative methods to record the shell activities:

— Use `tee` (usable during the boot process in the `initramfs`):

```
$ sh -i 2>&1 | tee typescript
```

- Use `gnome-terminal` with the `extend_line_buffer` for scrollbar.
- Use `screen` with `"^A H"` (see Section 9.1.2) to perform recording of console.
- Use `vim` with `":terminal"` to enter the terminal mode. Use `"Ctrl-W N"` to exit from terminal mode to normal mode. Use `":w typescript"` to write the buffer to a file.
- Use `emacs` with `"M-x shell"`, `"M-x eshell"`, or `"M-x term"` to enter recording console. Use `"C-x C-w"` to write the buffer to a file.

9.1.2 Le programme screen

`screen(1)` ne permet pas uniquement de faire tourner plusieurs processus dans une fenêtre de terminal, mais aussi à un **processus de l'interpréteur de commandes distant de survivre à d'éventuelles interruptions de la connexion**. Voici un scénario typique de `screen(1)` :

1. vous-vous connectez à une machine distante ;
2. vous démarrez `screen` sur une seule console ;
3. vous exécutez plusieurs programmes dans les fenêtres `screen` créées avec `^A c` (« Ctrl-A » suivi de « c ») ;
4. vous passez d'une des fenêtres multiples de `screen` à l'autre avec `^A n` (« Ctrl-A » suivi de « n ») ;
5. vous avez alors besoin de quitter votre terminal, mais vous ne voulez pas perdre votre travail en cours perdant la connexion.
6. vous pouvez **détacher** la session `screen` par différentes méthodes :
 - débrancher brutalement votre connexion réseau ;
 - entrer `^A d` (« Ctrl-A » suivi de « d ») et en quittant manuellement la connexion distante ;
 - entrer `^A DD` (« Ctrl-A » suivi de « DD ») pour détacher `screen` et vous déconnecter.
7. Vous vous reconnectez à la même machine distante (même depuis un autre terminal) ;
8. Vous lancez `screen` avec « `screen -r` » ;
9. `screen` **réattache** magiquement toutes les fenêtres `screen` précédentes avec tous les programmes qui y tournent.

ASTUCE

Avec `screen`, vous pouvez économiser des frais de connexion pour les connexions limitées, telles que les connexion commutées ou par paquets, parce que vous laissez un processus actif alors que vous êtes déconnecté. Vous pouvez le ré-attacher plus tard, lorsque vous vous reconnectez.

Dans une session `screen`, toutes les entrées clavier sont envoyées vers votre fenêtre actuelle sauf les séquences de touche de commande. Toutes les séquences de touche de commande `screen` sont entrées par `^A` (« Ctrl-A ») suivi d'une seule touche [plus les paramètres]. Voici celles dont il est important de se souvenir.

Consultez `screen(1)` pour davantage d'informations.

See `tmux(1)` for functionalities of the alternative command.

9.1.3 Navigating around directories

In Section 1.4.2, 2 tips to allow quick navigation around directories are described: `$CDPATH` and `mc`.

If you use fuzzy text filter program, you can do without typing the exact path. For `fzf`, include following in `~/.bashrc`.

```
FZF_KEYBINDINGS_PATH=/usr/share/doc/fzf/examples/key-bindings.bash
if [ -f $FZF_KEYBINDINGS_PATH ]; then
    . $FZF_KEYBINDINGS_PATH
fi
FZF_COMPLETION_PATH=/usr/share/doc/fzf/examples/completion.bash
if [ -f $FZF_COMPLETION_PATH ]; then
    . $FZF_COMPLETION_PATH
fi
```

affectation	signification
<code>^A ?</code>	afficher l'aide de screen (afficher les raccourcis clavier)
<code>^A c</code>	créer une nouvelle fenêtre et basculer vers celle-ci
<code>^A n</code>	aller à la fenêtre suivante
<code>^A p</code>	aller à la fenêtre précédente
<code>^A 0</code>	aller à la fenêtre 0
<code>^A 1</code>	aller à la fenêtre 1
<code>^A w</code>	afficher la liste des fenêtres
<code>^A a</code>	envoyer un Ctrl-A à la fenêtre actuelle en tant qu'entrée clavier
<code>^A h</code>	écrire dans un fichier une copie de la fenêtre actuelle
<code>^A H</code>	commencer et finir l'enregistrement de la fenêtre en cours vers un fichier
<code>^A ^X</code>	verrouiller le terminal (protégé par un mot de passe)
<code>^A d</code>	détacher la session screen du terminal
<code>^A DD</code>	détacher la session screen et se déconnecter

Table 9.2 – Liste des raccourcis clavier de screen

For example:

- You can jump to a very deep subdirectory with minimal efforts. You first type `cd **` and press **Tab**. Then you will be prompted with candidate paths. Typing in partial path strings, e.g., `s/d/b foo`, will narrow down candidate paths. You select the path to be used by `cd` with cursor and return keys.
- You can select a command from the command history more efficiently with minimal efforts. You press **Ctrl-R** at the command prompt. Then you will be prompted with candidate commands. Typing in partial command strings, e.g., `vim d`, will narrow down candidates. You select the one to be used with cursor and return keys.

9.1.4 Readline wrapper

Some commands such as `/usr/bin/dash` which lacks command line history editing capability can add such functionality transparently by running under `rlwrap` or its equivalents.

```
$ rlwrap dash -i
```

This provides convenient platform to test subtle points for `dash` with friendly `bash`-like environment.

9.2 Personnaliser vim

After you learn basics of `vim(1)` through Section 1.4.8, please read Bram Moolenaar's "[Seven habits of effective text editing \(2000\)](#)" to understand how `vim` should be used.



Attention

Don't try to change the default key bindings without very good reasons.

9.2.1 Customizing vim with internal features

The behavior of `vim` can be changed significantly by enabling its internal features through the Ex-mode commands such as `set ...` to set vim options.

These Ex-mode commands can be included in user's `vimrc` file, traditional `~/.vimrc` or git-friendly `~/.vim/vimrc`. Here is a very simple example¹:

1. More elaborate customization examples: "[Vim Galore](#)", "[sensible.vim](#)", "[#vim Recommendations](#)" ...

```

colorscheme murphy          " from /usr/share/vim/vim??/colors/*.vim
filetype plugin indent on   " filetype aware behavior
syntax enable               " Syntax highlight
"set spelllang=en_us        " Spell check language as en_us
"set spell                  " Enable spell check
set autoindent              " Copy indent from current line
set smartindent             " More than autoindent (Drop/Pop after {/})
set nosmarttab              " <Tab>-key always inserts blanks
set backspace=indent,eol,start " Back space through everything
set laststatus=2            " Always show status line
set statusline=%<%f%m%r%h%w%=%y[U+%04B]%2l/%2L=%P,%2c%V

```

9.2.2 Customizing vim with external packages

Simple customization to enable `securemodelines` and classical IDE can be enabled by installing [vim-scripts](#) package and appending the following to user's `vimrc` file.

```

packadd! securemodelines
packadd! winmanager
let mapleader = ' '
" IDE-like UI to navigate files and buffers with <space>wm, <space>wf, and <space>wb .
nnoremap <leader>wm      :WMToggle<CR>
nnoremap <leader>wf      :FirstExplorerWindow<CR>
nnoremap <leader>wb      :BottomExplorerWindow<CR>

```

The new native Vim package system works nicely with "git" and "git submodule". One such example configuration can be found at [my git repository: dot-vim](#). This does essentially:

- By using "git" and "git submodule", latest external packages, such as "name", are placed into `~/.vim/pack/*/opt/name` and similar.
- By adding `:packadd! name` line to user's `vimrc` file, these packages are placed on `runtimepath`.
- Vim loads these packages on `runtimepath` during its initialization.
- At the end of its initialization, tags for the installed documents are updated with `"helptags ALL"`.

For more, please start vim with `"vim --startuptime vimstart.log"` to check actual execution sequence and time spent for each step.

Interesting external plugin packages can be found:

- [Vim - the ubiquitous text editor](#) -- The official upstream site of Vim and vim scripts
- [VimAwesome](#) -- The listing of Vim plugins
- [vim-scripts](#) -- Debian package: a collection of vim scripts

It is quite confusing to see too many ways² to manage and load these external packages to vim. Checking the original information is the best cure.

key strokes	information
<code>:help package</code>	explanation on the vim package mechanism
<code>:help runtimepath</code>	explanation on the <code>runtimepath</code> mechanism
<code>:version</code>	internal states including candidates for the vimrc file
<code>:echo \$VIM</code>	the environment variable "\$VIM" used to locate the vimrc file
<code>:set runtimepath?</code>	list of directories which will be searched for all runtime support files
<code>:echo \$VIMRUNTIME</code>	the environment variable "\$VIMRUNTIME" used to locate various system provided runtime support files

Table 9.3 – Information on the initialization of vim

2. [vim-pathogen](#) was popular.

9.3 Enregistrer et présenter des données

9.3.1 Le démon de journal

De nombreux programmes enregistrent leur activité dans le répertoire « `/var/log/` ».

Consultez Section 3.4 et Section 3.3.

9.3.2 Analyseur de journaux

Voici les principaux analyseurs de journaux (« `~Gsecurity::log-analyzer` » dans `aptitude(8)`).

paquet	popcon	taille	description
logwatch	V:16, I:18	2265	analyseur de journal avec une sortie sympathique en Perl
fail2ban	V:116, I:129	2092	bannir les IP qui provoquent des erreurs d'authentification multiples
analog	V:3, I:105	3534	analyseur des journaux du serveur web
awstats	V:9, I:14	6910	analyseur des journaux du serveur web puissant ayant de nombreuses fonctionnalités
sarg	V:2, I:3	843	générateur de rapport d'analyse de squid
pflogsumm	V:2, I:4	111	résumer les entrées de journal de postfix
syslog-summary	V:0, I:2	30	résumer le contenu du fichier journal syslog
fwlogwatch	V:0, I:0	478	analyseur des journaux du pare-feu
squidview	V:0, I:1	189	surveiller et analyser les fichiers access.log de squid
swatch	V:0, I:0	101	visualisateur de fichier journal avec correspondance d'expressions rationnelles, mise en évidence et « hooks »
crm114	V:0, I:0	1119	analyseur et filtre de données diverses, dont les pourriels (CRM114)
icmpinfo	V:0, I:0	44	interpréter les messages ICMP

Table 9.4 – Liste des analyseurs de journaux système

Note

CRM114 fournit une architecture de vocabulaire pour écrire des filtres **fuzzy** avec la [bibliothèque d'expressions rationnelles TRE](#). Une de ses utilisations courantes est le filtrage des pourriels mais il peut aussi être utilisé pour l'analyse de journaux.

9.3.3 Affichage personnalisé des données de texte

Bien que des outils de visualisation de texte (« pager » tels que `more(1)` et `less(1)` (consultez Section 1.4.5) et des outils personnalisés de mise en évidence et de formatage (consultez Section 11.1.8) peuvent afficher des données textuelles de manière agréable, les éditeurs généralistes (consultez Section 1.4.6) sont plus souples et paramétrables.

ASTUCE

Pour `vim(1)` et ses alias de visualisation de texte `view(1)`, « `:set hls` » active la recherche avec mise en évidence.

9.3.4 Affichage personnalisé de la date et de l'heure

The default display format of time and date by the "`ls -l`" command depends on the **locale** (see Section 1.2.6 for value). The "`$LANG`" variable is referred first and it can be overridden by the "`$LC_TIME`" or "`$LC_ALL`" exported environment variables.

The actual default display format for each locale depends on the version of the standard C library (the `libc6` package) used. I.e., different releases of Debian had different defaults. For iso-formats, see [ISO 8601](#).

Si vous désirez vraiment personnaliser ce format d’affichage de la date et de l’heure plus loin que ne le fait **locale**, vous pouvez définir la **valeur de style d’heure** avec le paramètre « `--time-style` » ou par la valeur de « `$TIME_STYLE` » (consultez `ls(1)`, `date(1)`, « `info coreutils 'ls invocation'` »).

valeur de style pour l’heure	locale	affichage de la date et l’heure
iso	n’importe laquelle	01-19 00:15
long-iso	n’importe laquelle	2009-01-19 00:15
full-iso	n’importe laquelle	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	n’importe laquelle	19.01.09 00:15
+%d.%b.%y %H:%M	C ou en_US.UTF-8	19. Jan.09 00:15
+%d.%b.%y %H:%M	es_ES.UTF-8	19. ene.09 00:15

Table 9.5 – Display examples of time and date for the “`ls -l`” command with the **time style value**

ASTUCE

You can eliminate typing long option on commandline using command alias (see Section [1.5.9](#)):

```
alias ls='ls --time-style=+%d.%m.%y %H:%M'
```

9.3.5 Écho colorisé de l’interpréteur de commandes

L’écho de l’interpréteur de commandes sur la plupart des terminaux peut être colorisé en utilisant le [code ANSI d’échappement](#) (consultez « `/usr/share/doc/xterm/ctlseqs.txt.gz` »).

Essayez, par exemple, ce qui suit :

```
$ RED=$(printf "\x1b[31m")
$ NORMAL=$(printf "\x1b[0m")
$ REVERSE=$(printf "\x1b[7m")
$ echo "${RED}RED-TEXT${NORMAL} ${REVERSE}REVERSE-TEXT${NORMAL}"
```

9.3.6 Commandes colorisées

Les commandes colorisées sont pratiques pour examiner la sortie d’une commande dans l’environnement interactif. J’inclus ce qui suit dans mon fichier « `~/ .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
```

L'utilisation d'alias réserve les effets de couleurs à l'utilisation interactives des commandes. Il a l'avantage sur l'exportation de la variable d'environnement « `export GREP_OPTIONS='--color=auto'` » car la couleur peut être affichée avec des programmes de visualisation (« pager » tels que `less(1)`). Si vous souhaitez supprimer la couleur lors du tubage (« piping » à destination d'autres programmes, utilisez plutôt « `--color=auto` » dans l'exemple ci-dessus pour « `~/ .bashrc` ».

ASTUCE

Vous pouvez désactiver ces alias dans l'environnement interactif en appelant l'interpréteur de commandes par « `TERM=dumb bash` ».

9.3.7 Enregistrer l'activité de l'éditeur pour des répétitions complexes

Vous pouvez enregistrer l'activité de l'éditeur pour des répétitions complexes.

Pour [Vim](#), de la manière suivante :

- « `qa` » : démarre l'enregistrement des caractères entrés dans le registre appelé « `a` ».
- ...activité de l'éditeur
- « `q` » : termine l'enregistrement des caractères entrés.
- « `@a` » : exécute le contenu du registre « `a` ».

Pour [Emacs](#), de la manière suivante :

- « `:C-x (` » : commencer la définition d'une macro clavier.
- ...activité de l'éditeur
- « `C-x)` » : terminer la définition d'une macro clavier.
- « `C-x e` » : exécuter une macro clavier.

9.3.8 Enregistrer l'image graphique d'une application X

Il existe plusieurs manières d'enregistrer l'image graphique d'une application X, y compris un affichage `xterm`.

paquet	popcon	taille	screen
gnome-screenshot	V:27, I:264	1173	Wayland
flameshot	V:5, I:11	1510	Wayland
gimp	V:58, I:315	19706	Wayland + X
x11-apps	V:27, I:454	2437	X
imagemagick	I:373	220	X
scrot	V:7, I:76	123	X

Table 9.6 – Liste des outils de manipulation d'images

9.3.9 Enregistrer les modifications dans des fichiers de configuration

There are specialized tools to record changes in configuration files with help of DVCS and to make system snapshots on [Btrfs](#).

You may also think about local script Section [10.2.2](#) approach.

paquet	popcon	taille	description
etckeeper	V:28, I:32	176	enregistrer les fichiers de configuration et leurs métadonnées avec Git (par défaut), Mercurial ou Bazaar (nouveau)
timeshift	V:2, I:4	3158	system restore utility using rsync or BTRFS snapshots
snapper	V:2, I:3	2119	Linux filesystem snapshot management tool

Table 9.7 – List of packages which can record configuration history

9.4 Surveiller, contrôler et démarrer l'activité des programmes

L'activité des programmes peut être surveillée et contrôlée à l'aide d'outils spécialisés.

paquet	popcon	taille	description
coreutils	V:901, I:999	17478	nice(1) : lancer un programme avec une priorité d'ordonnancement modifiée
bsdutils	V:708, I:999	394	renice(1) : modifier la priorité d'ordonnancement d'un programme en cours d'exécution
procps	V:742, I:999	1648	utilitaires du système de fichiers « /proc » : ps(1) , top(1) , kill(1) , watch(1) , ...
psmisc	V:433, I:831	793	utilitaires du système de fichiers « /proc » : killall(1) , fuser(1) , peekfd(1) , pstree(1)
time	V:13, I:249	129	time(1) : lancer un programme qui indique l'utilisation des ressources du système en fonction du temps
sysstat	V:168, I:190	1923	sar(1) , iostat(1) , mpstat(1) , ... : outils de mesure des performances du système pour Linux
isag	V:0, I:4	117	Générateur interactif de graphes de l'activité système pour sysstat
lsof	V:393, I:944	451	lsof(8) : afficher la liste des fichiers ouverts par un processus en cours d'utilisation en utilisant l'option « -p »
strace	V:16, I:151	2367	strace(1) : tracer les appels système et les signaux
ltrace	V:1, I:22	363	ltrace(1) : tracer les appels de bibliothèque
xtrace	V:0, I:0	353	xtrace(1) : tracer la communication entre un client X11 et le serveur
powertop	V:8, I:204	662	powertop(1) : information concernant la puissance électrique utilisée
cron	V:800, I:997	263	faire tourner des processus en arrière plan selon un calendrier depuis le démon cron(8)
anacron	V:399, I:463	104	ordonnanceur de type cron pour les systèmes qui ne tournent pas 24 heures sur 24
at	V:154, I:278	161	at(1) ou batch(1) : lancer un travail à une heure déterminée ou en dessous d'un niveau de charge donné

Table 9.8 – Liste des outils de surveillance et de contrôle de l'activité des programmes

ASTUCE

Le paquet `procps` fournit des fonctions très basiques de surveillance, de contrôle et de lancement des activités du programme. Vous devriez toutes les apprendre.

9.4.1 Temps d'un processus

Afficher la durée du processus indiqué dans la commande.

```
# 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 La priorité d'ordonnancement

Une valeur de politesse (« nice » est utilisée pour contrôler la priorité d'ordonnancement du processus.

valeur de politesse	priorité d'ordonnancement
19	la plus basse priorité d'un processus (poli)
0	très haute priorité de processus pour un utilisateur
-20	très haute priorité d'un processus pour root (non poli)

Table 9.9 – Liste des valeurs de politesse pour la priorité d'ordonnancement

```
# nice -19 top # very nice
# nice --20 wodim -v -eject speed=2 dev=0,0 disk.img # very fast
```

Parfois, une valeur extrême de politesse (« nice ») fait plus de mal que de bien au système. Utilisez cette commande avec précaution.

9.4.3 La commande ps

La commande `ps(1)` sous Debian comporte à la fois les fonctionnalités de BSD et de SystemV, elle aide à identifier l'activité des processus de manière statique.

style	commande typique	fonction
BSD	<code>ps aux</code>	afficher %CPU %MEM
System V	<code>ps -efH</code>	afficher le PPID

Table 9.10 – Liste des styles de la commande ps

Les processus fils zombies (« defunct ») peuvent être tués par l'identifiant du processus parent identifié dans le champ « PPID » La commande `ps tree(1)` affiche une arborescence des processus.

9.4.4 La commande top

`top(1)` sous Debian a de riches fonctionnalités et aide à identifier de manière dynamique quels sont les processus qui ont une activité curieuse.

It is an interactive full screen program. You can get its usage help press by pressing the "h"-key and terminate it by pressing the "q"-key.

9.4.5 Afficher les fichier ouverts par un processus

Vous pouvez afficher la liste des fichiers ouverts par un processus avec un identifiant de processus (PID), par exemple de 1 en faisant ce qui suit :

```
$ sudo lsof -p 1
```

PID=1 est habituellement le programme `init`.

9.4.6 Tracer l'activité d'un programme

Vous pouvez tracer l'activité d'un programme avec `strace(1)`, `ltrace(1)` ou `xtrace(1)` pour les appels système, les appels bibliothèque ou la communication entre le serveur et le client X11.

Vous pouvez tracer les appels système de la commande `ls` de la manière suivante :

```
$ sudo strace ls
```

ASTUCE

Use **strace-graph** script found in `/usr/share/doc/strace/examples/` to make a nice tree view

9.4.7 Identification des processus qui utilisent des fichiers ou des sockets

Vous pouvez aussi identifier les processus qui utilisent des fichiers par `fuser(1)`, par exemple pour « `/var/log/mail.log` » en faisant ce qui suit :

```
$ sudo fuser -v /var/log/mail.log
                USER      PID ACCESS COMMAND
/var/log/mail.log: root      2946 F.... rsyslogd
```

Vous voyez que le fichier « `/var/log/mail.log` » est ouvert en écriture par la commande `rsyslogd(8)`.

Vous pouvez aussi identifier les processus qui utilisent des sockets par `fuser(1)`, par exemple, pour « `smtp/tcp` » par ce qui suit :

```
$ sudo fuser -v smtp/tcp
                USER      PID ACCESS COMMAND
smtp/tcp:       Debian-exim 3379 F.... exim4
```

Vous savez maintenant que `exim4(8)` tourne sur votre système pour gérer les connexions [TCP](#) du port [SMTP](#) (25).

9.4.8 Répéter une commande avec un intervalle constant

`watch(1)` exécute un programme de manière répétitive avec un intervalle constant tout en affichant sa sortie en plein écran.

```
$ watch w
```

Cela permet l'affichage, mis à jour toutes les deux secondes, de qui est connecté au système.

9.4.9 Répéter une commande en bouclant entre des fichiers

Il existe plusieurs manières de répéter une boucle de commande entre des fichiers correspondant à une condition, par exemple, correspondant au motif « glob » « `*.ext` ».

— Méthode de la boucle « `for` » de l'interpréteur de commandes (consultez [Section 12.1.4](#)) :

```
for x in *.ext; do if [ -f "$x" ]; then command "$x" ; fi; done
```

— combinaison de `find(1)` et de `xargs(1)` :

```
find . -type f -maxdepth 1 -name '*.ext' -print0 | xargs -0 -n 1 command
```

— `find(1)` avec l'option « `-exec` » avec une commande :

```
find . -type f -maxdepth 1 -name '*.ext' -exec command '{}' \;
```

— `find(1)` avec l'option « `-exec` » avec un court script de l'interpréteur :

```
find . -type f -maxdepth 1 -name '*.ext' -exec sh -c "command '{}'" && echo 'successful' \;
```

Les exemple ci-dessus ont été écrits afin d'assurer une prise en compte correcte de noms de fichiers étranges tels que ceux qui comportent des espaces. Consultez Section 10.1.5 pour une utilisation avancée de `find(1)`.

9.4.10 Lancer un programme depuis l'interface graphique

Pour l'[interface en ligne de commande \(CLI\)](#), le premier programme trouvé dans les répertoires spécifiés dans la variable d'environnement `$PATH`, et dont le nom correspond, est exécuté. Voir Section 1.5.3

For the [graphical user interface \(GUI\)](#) compliant to the [freedesktop.org](#) standards, the `*.desktop` files in the `/usr/share/applications` directory provide necessary attributes for the GUI menu display of each program. Each package which is compliant to [Free-desktop.org](#)'s xdg menu system installs its menu data provided by `"*.desktop"` under `"/usr/share/applications/"`. Modern desktop environments which are compliant to [Freedesktop.org](#) standard use these data to generate their menu using the `xdg-utils` package. See `"/usr/share/doc/xdg-utils/README"`.

Par exemple, le fichier `chromium.desktop` définit les attributs pour le "Navigateur Web Chromium" tel que "Name" pour le nom du programme, "Exec" pour le chemin et les paramètres d'exécution du programme, "Icon" pour l'icône utilisée, etc. (voir la [Spécification d'Entrée de fichier Desktop](#)) comme suit :

```
[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
```

Ceci est une description extrêmement simplifiée. Les fichiers `*.desktop` sont analysés comme suit:

L'environnement de bureau définit les variables d'environnement `$XDG_DATA_HOME` et `$XDG_DATA_DIR`. Par exemple, sous GNOME 3 :

- « `$XDG_DATA_HOME` » n'est pas paramétré. (La valeur par défaut de « `$HOME/.local/share` » est utilisée.)
- « `$XDG_DATA_DIRS` » est paramétré à « `/usr/share/gnome:/usr/local/share:/usr/share/` ».

Les répertoires de base (voir [Spécification du répertoire de base XDG](#) et les répertoires des « applications » sont donc les suivants :

- `$HOME/.local/share/` → `$HOME/.local/share/applications`
- `/usr/share/gnome/` → `/usr/share/gnome/applications/`
- `/usr/local/share/` → `/usr/local/share/applications`
- `/usr/share/` → `/usr/share/applications`

Les fichiers `*.desktop` sont parcourus dans ces répertoires `applications` dans cet ordre.

ASTUCE

Une entrée de menu « graphique » personnalisée peut être créée en ajoutant un fichier « *.desktop » dans le dossier « \$HOME/.local/share/applications/ ».

ASTUCE

De la même façon, si un fichier « *.desktop » est créé dans le répertoire « autostart » dans ces répertoires de base, le programme spécifié dans le fichier « *.desktop » est exécuté automatiquement lorsque l'environnement de bureau est démarré. Consultez la [Spécification de démarrage automatique d'applications](#) ;

ASTUCE

De la même façon, si un fichier « *.desktop » est créé dans le répertoire « \$HOME/Desktop » et que l'environnement de bureau est configuré afin de prendre en charge l'exécution depuis une icône de bureau, le programme qui y est spécifié est exécuté lorsqu'on clique sur l'icône. Veuillez noter que le nom du répertoire « \$HOME/Desktop » est dépendant de la localisation. Consultez `xdg-user-dirs-update(1)`.

9.4.11 Personnaliser le programme à lancer

Certains programmes lancent automatiquement d'autres programmes. Voici des points-clés pour la personnalisation de ce processus :

- Menu de configuration des applications :
 - bureau GNOME : « Paramètres » → « Système » → « Détails » → « Applications par défaut »
 - bureau KDE : « K » → « Centre de contrôle » → « Composants KDE » → « Sélection de composant »
 - navigateur Iceweasel : « Éditer » → « Préférences » → « Applications »
 - `mc(1)` : « /etc/mc/mc.ext »
- des variables d'environnement telles que « \$BROWSER », « \$EDITOR », « \$VISUAL » et « \$PAGER » (consultez `environ(7)`)
- le système `update-alternatives(1)` pour des programmes tels que « editor », « view », « x-www-browser », « gnome-www-browser » et « www-browser » (consultez Section 1.4.7)
- le contenu des fichiers « ~/.mailcap » et « /etc/mailcap » qui associe un type [MIME](#) avec un programme (consultez `mailcap(5)`)
- le contenu des fichiers « ~/.mime.types » et « /etc/mime.types » qui associe l'extension du nom de fichier avec un type [MIME](#) (consultez `run-mailcap(1)`)

ASTUCE

`update-mime(8)` met à jour le fichier « /etc/mailcap » en utilisant le fichier « /etc/mailcap.order » (consultez `mailcap.order(5)`).

ASTUCE

Le paquet `debianutils` fournit `sensible-browser(1)`, `sensible-editor(1)`, et `sensible-pager(1)` qui prennent des décisions raisonnables concernant l'éditeur, le visualisateur, le navigateur à appeler respectivement. Je vous recommande de lire ces scripts de l'interpréteur de commandes.

ASTUCE

De façon à faire tourner une application de console telle que `mutt` sous X en tant qu'application préférée, vous devriez créer une application X comme suit et définir « /usr/local/bin/mutt-term » comme étant votre application préférée à lancer comme il est décrit :

```
# cat /usr/local/bin/mutt-term <<EOF
#!/bin/sh
gnome-terminal -e "mutt \$@"
EOF
chmod 755 /usr/local/bin/mutt-term
```

9.4.12 Tuer un processus

Utilisez `kill(1)` pour tuer (ou envoyer un signal à) un processus avec son identifiant de processus.

Utilisez `killall(1)` ou `pkill(1)` pour faire la même chose avec le nom de commande du processus et d'autres attributs.

valeur du signal	nom du signal	action	note
0	---	no signal is sent (see <code>kill(2)</code>)	check if process is running
1	<code>SIGHUP</code>	terminate the process	disconnected terminal (signal hang up)
2	<code>SIGINT</code>	terminate the process	interrupt from keyboard (CTRL - C)
3	<code>SIGQUIT</code>	terminate the process and dump core	quit from keyboard (CTRL - \)
9	<code>SIGKILL</code>	terminate the process	unblockable kill signal
15	<code>SIGTERM</code>	terminate the process	blockable termination signal

Table 9.11 – Liste des signaux couramment utilisés avec la commande `kill`

9.4.13 Planifier des tâches qui s'exécutent une fois

Exécutez de la manière suivante la commande `at(1)` pour planifier un travail qui s'exécute une fois

```
$ echo 'command -args' | at 3:40 monday
```

9.4.14 Planifier des tâches qui s'exécutent régulièrement

Utilisez `cron(8)` pour planifier des tâches qui s'exécutent régulièrement. Consultez `crontab(1)` et `crontab(5)`.

Vous pouvez planifier le lancement des processus en tant qu'utilisateur normal, par exemple `toto` en créant un fichier `crontab(5)` file comme « `/var/spool/cron/crontabs/toto` » avec la commande « `crontab -e` ».

Voici un exemple de fichier `crontab(5)`.

```
# use /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
```

ASTUCE

Sur un système qui ne tourne pas en permanence, installez le paquet `anacron` afin de planifier les tâches périodiques à des intervalles particulier dès que le temps de fonctionnement « `uptime` » de la machine le permet. Consultez `anacron(8)` et `anacrontab(5)`.

ASTUCE

Vous pouvez lancer périodiquement les scripts de maintenance planifiée du système, depuis le compte de l'administrateur en les plaçant dans « `/etc/cron.hourly/` », « `/etc/cron.daily/` », « `/etc/cron.weekly/` » ou « `/etc/cron.monthly/` ». L'échéancier d'exécution de ces scripts peut être personnalisé dans « `/etc/crontab` » et « `/etc/anacrontab` ».

[Systemd](#) has low level capability to schedule programs to run without `CRON` daemon. For example, `/lib/systemd/system/apt-daily.timer` and `/lib/systemd/system/apt-daily.service` set up daily apt download activities. See `systemd.timer(5)`.

9.4.15 touche Alt-SysRq

Pressing `Alt-SysRq` (`PrtScr`) followed by one keys does the magic of rescuing control of the system.

touche suivant Alt-Sys	description de l'action
k	kill all processes on the current virtual console (SAK)
s	synchroniser tous les systèmes de fichiers montés afin d'éviter la corruption des données
u	remonter en lecture seule tous les systèmes de fichiers montés (umount)
r	restaurer le clavier depuis le mode raw (brut) après un plantage de X

Table 9.12 – List of notable SAK command keys

See more on [Linux kernel user's and administrator's guide](#) » [Linux Magic System Request Key Hacks](#)

ASTUCE

Depuis un terminal SSH, etc., vous pouvez utiliser la fonctionnalité Alt-SysRq en écrivant vers « `/proc/sysrq-trigger` ». Par exemple, « `echo s > /proc/sysrq-trigger; echo u > /proc/sysrq-trigger` » depuis l'invite de l'interpréteur de commandes de l'administrateur `syncs` (synchronise) et `umount` (démonte) tous les systèmes de fichiers montés.

The current (2021) Debian amd64 Linux kernel has `/proc/sys/kernel/sysrq=438=0b110110110`:

- `2 = 0x2` - enable control of console logging level (ON)
- `4 = 0x4` - enable control of keyboard (SAK, unraw) (ON)
- `8 = 0x8` - enable debugging dumps of processes etc. (OFF)
- `16 = 0x10` - enable sync command (ON)
- `32 = 0x20` - enable remount read-only (ON)
- `64 = 0x40` - enable signaling of processes (term, kill, oom-kill) (OFF)
- `128 = 0x80` - allow reboot/poweroff (ON)
- `256 = 0x100` - allow nicing of all RT tasks (ON)

9.5 Astuces de maintenance du système

9.5.1 Qui se trouve sur le système ?

Vous pouvez rechercher qui se trouve sur le système par les commandes suivantes :

- `who(1)` shows who is logged on.
- `w(1)` shows who is logged on and what they are doing.
- `last(1)` shows listing of last logged in user.
- `lastb(1)` affiche une liste des utilisateurs s'étant mal connectés.

ASTUCE

« `/var/run/utmp` » et « `/var/log/wtmp` » conservent ces informations de l'utilisateur. Consultez `login(1)` et `utmp(5)`.

9.5.2 Prévenir tout le monde

Vous pouvez envoyer un message à toutes les personnes connectées au système avec `wall(1)` en faisant ce qui suit :

```
$ echo "We are shutting down in 1 hour" | wall
```

9.5.3 Identification du matériel

Pour les périphériques similaires à [PCI](#) ([AGP](#), [PCI-Express](#), [CardBus](#), [ExpressCard](#), etc.), `lspci(8)` (probablement avec l'option « `-nn` ») est un bon point de départ pour l'identification du matériel.

Vous pouvez aussi identifier le matériel en lisant le contenu de « `/proc/bus/pci/devices` » ou en parcourant l'arborescence de répertoires se trouvant sous « `/sys/bus/pci` » (consultez [Section 1.2.12](#)).

paquet	popcon	taille	description
pciutils	V:200, I:991	196	utilitaires PCI de Linux : lspci(8)
usbutils	V:73, I:858	325	utilitaires USB de Linux : lsusb(8)
nvme-cli	V:2, I:10	779	NVMe utilities for Linux: nvme(1)
pcmciautils	V:11, I:17	97	utilitaires PCMCIA pour Linux : pccardctl(8)
scsitools	V:0, I:3	375	collection d'outils pour la gestion des périphériques SCSI : lsscsi(8)
procinfo	V:0, I:12	135	informations sur le système obtenues dans « /proc » : lsdev(8)
lshw	V:11, I:96	850	informations concernant la configuration matérielle : lshw(1)
discover	V:37, I:948	90	système d'identification du matériel : discover(8)

Table 9.13 – Listes des outils d'identification du matériel

paquet	popcon	taille	description
console-setup	V:133, I:960	419	Police de la console Linux et utilitaires de table de caractères
x11-xserver-utils	V:272, I:516	511	utilitaires pour le serveur X : xset(1) , xmodmap(1)
acpid	V:135, I:281	176	démon servant à gérer les événements délivrés par l'Interface avancée de configuration et de gestion de l'énergie ACPI (« Advanced Configuration and Power Interface »)
acpi	V:15, I:266	45	utilitaire d'affichage des informations des périphériques ACPI
sleepd	V:0, I:0	86	démon permettant de mettre un ordinateur portable en veille lorsqu'il est inactif
hdparm	V:446, I:717	256	optimisation de l'accès aux disques durs (consultez Section 9.6.9)
smartmontools	V:150, I:208	2117	contrôle et surveillance des systèmes de stockage en utilisant S.M.A.R.T.
setserial	V:5, I:9	117	collection d'outils pour gérer les ports série
memtest86+	V:1, I:28	2391	collection d'outils pour gérer la mémoire physique
scsitools	V:0, I:3	375	collection d'outils pour gérer le matériel SCSI
setcd	V:0, I:1	35	optimisation de l'accès au lecteur de CD
big-cursor	I:1	27	curseurs de souris plus grands pour X

Table 9.14 – Liste des outils de configuration du matériel

9.5.4 Configuration matérielle

Bien que l'essentiel de la configuration du matériel puisse être gérée au moyen des outils graphiques qui accompagnent les environnements de bureau graphiques modernes comme GNOME ou KDE, c'est une bonne idée de connaître certaines méthodes de base permettant de le configurer.

Ici, [ACPI](#) est une infrastructure de gestion de l'alimentation électrique du système plus récente qu'[APM](#).

ASTUCE

L'ajustement de la fréquence d'horloge des processeurs modernes est gérée par des modules du noyau tels que `acpi_cpufreq`.

9.5.5 Heure système et matérielle

Ce qui suit permet de définir l'heure du système et du matériel à MM/DD hh:mm, CCYY :

```
# date MMDDhhmmCCYY
# hwclock --utc --systohc
# hwclock --show
```

Sur un système Debian, l'heure est normalement affichée en heure locale mais l'heure système et matérielle utilisent habituellement l'heure [TUC\(GMT\)](#).

If the hardware time is set to UTC, change the setting to "UTC=yes" in the `/etc/default/rcS`.

La commande suivante relance la configuration du fuseau horaire utilisé par le système Debian.

```
# dpkg-reconfigure tzdata
```

Si vous désirez ajuster l'heure de votre système par l'intermédiaire du réseau, vous pouvez envisager l'utilisation du service [NTP](#) avec un paquet tel que `ntp`, `ntpd` ou `chrony`.

ASTUCE

Sous [systemd](#), utilisez plutôt `systemd-timesyncd` pour la synchronisation avec l'heure du réseau. Voir `systemd-timesyncd(8)`.

Consultez ce qui suit.

- [Comment gérer précisément la date et l'heure](#)
- [Projet de services publics NTP](#)
- Le paquet `ntp-doc`

ASTUCE

`ntptrace(8)` du paquet `ntp` peut suivre la trace d'une chaîne de serveurs NTP jusqu'à la source primaire.

9.5.6 Configuration du terminal

Il existe plusieurs composants pour configurer la console en mode caractères et les fonctionnalités du système `ncurses(3)`.

- Le fichier `/etc/terminfo/*/*` (`terminfo(5)`)
- La variable d'environnement `$TERM` (`term(7)`)
- `setterm(1)`, `stty(1)`, `tic(1)` et `toe(1)`

Si l'entrée `terminfo` pour `xterm` ne fonctionne pas avec un `xterm` non Debian, changez le type de terminal dans `$TERM` de `xterm` pour une version limitée en fonctionnalités comme `xterm-r6` lorsque vous vous connectez à distance à un système Debian. Consultez `/usr/share/doc/libncurses5/FAQ` pour davantage d'informations. `dumb` est le plus petit dénominateur commun pour `$TERM`.

9.5.7 L'infrastructure de gestion du son

Les pilotes de périphériques des cartes sons pour les versions actuelles de Linux sont fournies par [Advanced Linux Sound Architecture \(ALSA\)](#). ALSA fournit un mode d'émulation du système précédent [Open Sound System \(OSS\)](#) pour des raisons de compatibilité.

Application softwares may be configured not only to access sound devices directly but also to access them via some standardized sound server system. Currently, PulseAudio, JACK, and PipeWire are used as sound server system. See [Debian wiki page on Sound](#) for the latest situation.

Il y a habituellement un moteur de son commun pour chacun des environnements de bureau les plus courants. Chaque moteur de son utilisé par l'application peut choisir de se connecter à un serveur de son différent.

ASTUCE

Utilisez « `cat /dev/urandom > /dev/audio` » ou `speaker-test(1)` pour tester les hauts-parleurs (^C pour arrêter).

ASTUCE

Si vous n'arrivez pas à obtenir de sons, il est possible que votre haut-parleur soit connecté à une sortie muette (« muted »). Les systèmes de son modernes ont de nombreuses sorties. `alsamixer(1)` du paquet `alsa-utils` est pratique pour configurer les paramètres de volume et de coupure son

paquet	popcon	taille	description
alsa-utils	V:337, I:456	2283	utilitaires de configuration et d'utilisation d'ALSA
oss-compat	V:1, I:27	20	Compatibilité OSS sous ALSA évitant les erreurs « <code>/dev/dsp not found</code> »
pipewire	V:59, I:78	30	audio and video processing engine multimedia server - metapackage (unify JACK and PulseAudio)
pipewire-bin	V:67, I:76	1848	audio and video processing engine multimedia server - programs (unify JACK and PulseAudio)
pipewire-audio-client-libraries	V:0, I:1	380	audio and video processing engine multimedia server - audio client libraries to replace ALSA, JACK and PulseAudio (experimental)
pulseaudio	V:357, I:453	6774	PulseAudio server (standard)
libpulse0	V:292, I:586	971	PulseAudio client library (standard)
jackd	V:3, I:26	9	serveur du kit de connexion audio JACK. (JACK) (faible latence)
libjack0	V:1, I:12	338	bibliothèque du kit de connexion JACK. (JACK) (faible latence)
libgstreamer1.0-0	V:374, I:567	5283	GStreamer : moteur de son de GNOME
libphonon4	I:101	680	Phonon : moteur de son de KDE

Table 9.15 – Liste des paquets son

9.5.8 Désactiver l'économiseur d'écran

Pour désactiver l'écran de veille, utilisez les commandes suivantes :

9.5.9 Désactiver les bips

On peut toujours débrancher le haut-parleur du PC pour désactiver les bips. La suppression du module du noyau `pcspkr` le fait pour vous.

Ce qui suit évite que le programme `readline(3)` utilisé par `bash(1)` ne bipe lors de l'apparition d'un caractère d'alerte (ASCII=7).

environnement	commande
Console Linux	setterm -powersave off
Système X Window (couper l'économiseur d'écran)	xset s off
X Window (désactive dpms)	xset -dpms
X Window (interface de configuration graphique d'économiseur d'écran)	xscreensaver-command -prefs

Table 9.16 – Liste des commandes pour désactiver l'économiseur d'écran

```
$ echo "set bell-style none">> ~/.inputrc
```

9.5.10 Utilisation de la mémoire

Vous avez à votre disposition deux moyens d'obtenir l'état d'utilisation de la mémoire.

- Le message de démarrage du noyau dans « `/var/log/dmesg` » donne exactement la taille de la mémoire disponible.
- `free(1)` et `top(1)` affichent des informations concernant les ressources mémoire sur le système actif.

Voici un exemple.

```
# 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			

Vous pourriez vous demander « `dmesg` me rapporte 990 Mo de mémoire libre et « `free -k` » me dit que 320 Mo sont libres. Il manque plus de 600 Mo... ».

Ne vous inquiétez pas de la taille importante de « `used` » et de la petite taille de « `free` » sur la ligne « `Mem:` », mais lisez ce qui se trouve sous celle-ci (675404 et 321780 dans l'exemple ci-dessous) et détendez-vous.

Pour mon MacBook avec 1Go=1048576k DRAM (la mémoire vidéo en prend une partie), je peux voir ce qui suit :

affiché	taille
Taille totale dans <code>dmesg</code>	1016784k = 1Go - 31792k
Libre dans <code>dmesg</code>	990528k
Total sous l'interpréteur de commandes	997184k
Libre sous l'interpréteur de commandes	20256k (mais réellement 321780k)

Table 9.17 – Taille mémoire affichée

9.5.11 Vérification de la sécurité et de l'intégrité du système

Une mauvaise maintenance du système peut rendre votre système vulnérable à une attaque externe.

Pour la vérification de la sécurité et de l'intégrité du système, vous pouvez démarrer avec ce qui suit :

- Le paquet `debsums`, consultez `debsums(1)` et Section [2.5.2](#).
- Le paquet `chkrootkit`, consultez `chkrootkit(1)`.
- La familles de paquets `clamav`, consultez `clamscan(1)` et `freshclam(1)`.

paquet	popcon	taille	description
logcheck	V:8, I:10	102	démon pour poster à l'administrateur les anomalies des fichiers journaux du système
debsums	V:5, I:44	108	utilitaire pour vérifier les fichiers des paquets installés d'après leur somme de contrôle MD5
chkrootkit	V:5, I:24	971	détecteur de rootkit .
clamav	V:12, I:57	775	utilitaire anti-virus pour UNIX - interface en ligne de commandes.
tiger	V:2, I:3	7822	signale les vulnérabilités du système pour la sécurité
tripwire	V:2, I:3	11618	vérificateur d'intégrité des fichiers et répertoires
john	V:2, I:12	460	outils de casse des mots de passe actifs
aide	V:1, I:2	2049	Environnement avancé de détection d'intrusion (« Advanced Intrusion Detection Environment ») — bibliothèque statique
integrit	V:0, I:0	329	programme de vérification de l'intégrité des fichiers
crack	V:0, I:1	152	programme pour deviner les mots de passe

Table 9.18 – Liste d'outils pour la vérification de la sécurité et de l'intégrité du système

— [FAQ de sécurité Debian](#).

— [Manuel de sécurisation de Debian](#).

Voici un script simple pour rechercher des fichiers typiques ayant des permissions incorrectes d'écriture pour tout le monde.

```
# find / -perm 777 -a \! -type s -a \! -type l -a \! \! -type d -a -perm 1777 \)
```



Attention

Comme le paquet `debsums` utilise des sommes de contrôle [MD5](#) enregistrées de manière statique, on ne peut pas lui faire entièrement confiance comme outil d'audit de la sécurité envers des attaques malveillantes.

9.6 Astuces relatives au stockage des données

Démarrer votre système avec un [CD autonome](#) de Linux ou avec un [CD de l'installateur debian](#) en mode secours (rescue) vous facilite la reconfiguration de l'enregistrement des données sur votre disque de démarrage.

9.6.1 Utilisation de l'espace disque

L'utilisation de l'espace disque peut être estimée à l'aide de programmes fournis par les paquets `mount`, `coreutils` et `xdu` :

- `mount(8)` indique tous les systèmes de fichiers (disques) montés.
- `df(1)` indique l'espace disque occupé par les systèmes de fichiers.
- `du(1)` indique l'espace disque occupé par une arborescence de répertoires.

ASTUCE

La sortie de `du(1)` peut être renvoyée vers `xdu(1x)` pour réaliser une présentation graphique et interactive avec « `du -k . |xdu` », « `sudo du -k -x / |xdu` », etc.

9.6.2 Configuration de la partition du disque

Bien que `fdisk(8)` ait été considéré comme un standard pour la configuration de la [partition du disque dur](#), il mérite une certaine attention. « Données de partition du disque », « table de partition » et « Étiquette de disque » sont tous des synonymes.

Older PCs use the classic [Master Boot Record \(MBR\)](#) scheme to hold [disk partitioning](#) data in the first sector, i.e., [LBA](#) sector 0 (512 bytes).

Recent PCs with [Unified Extensible Firmware Interface \(UEFI\)](#), including Intel-based Macs, use [GUID Partition Table \(GPT\)](#) scheme to hold [disk partitioning](#) data not in the first sector.

Alors que `fdisk(8)` a été l'outil standard de partitionnement de disque, `parted(8)` le remplace maintenant.

paquet	popcon	taille	description
util-linux	V:899, I:999	4599	divers utilitaires systèmes dont <code>fdisk(8)</code> et <code>cfdisk(8)</code>
parted	V:367, I:548	307	programme GNU de redimensionnement des partitions <code>Parted</code>
gparted	V:16, I:127	2063	éditeur de partitions de GNOME basé sur <code>libparted</code>
gdisk	V:298, I:501	880	partition editor for the GPT/MBR hybrid disk
kpartx	V:19, I:30	87	programme pour créer des mappages de périphériques pour les partitions

Table 9.19 – Listes de paquets de gestion de la partition du disque



Attention

Bien que `parted(8)` prétend pouvoir créer et redimensionner aussi les systèmes de fichiers, il est plus sûr de toucher à ces choses-là en utilisant des outils spécialisés et bien maintenus tels que `mkfs(8)` (`mkfs.msdos(8)`, `mkfs.ext2(8)`, `mkfs.ext3(8)`, `mkfs.ext4(8)`, ...) et `resize2fs(8)`.

Note

De manière à passer de [GPT](#) à [MBR](#), il vous faut d'abord effacer les premiers blocs du disque directement (consultez [Section 9.8.6](#)) et utiliser « `parted /dev/sdx mklabel gpt` » ou « `parted /dev/sdx mklabel msdos` » afin de le mettre en place. Vous remarquerez que « `msdos` » est utilisé ici pour [MBR](#).

9.6.3 Accès à une partition en utilisant l'UUID

Although reconfiguration of your partition or activation order of removable storage media may yield different names for partitions, you can access them consistently. This is also helpful if you have multiple disks and your BIOS/UEFI doesn't give them consistent device names.

- `mount(8)` avec l'option « `-U` » peut monter un périphérique en mode bloc en utilisant l'[UUID](#) plutôt que son nom de fichier de périphérique comme « `/dev/sda3` ».
- « `/etc/fstab` » (consultez `fstab(5)`) peut utiliser l'[UUID](#).
- Les chargeurs initiaux ([Section 3.1.2](#)) peuvent aussi utiliser [UUID](#).

ASTUCE

Vous pouvez tester l'[UUID](#) d'un périphérique spécial en mode bloc avec `blkid(8)`.
You can also probe it and other information with "`lsblk -f`".

9.6.4 LVM2

LVM2 est un [gestionnaire de volume logique](#) pour le noyau Linux. Avec LVM2, les partitions peuvent être créées sur des volumes logiques plutôt que sur des disques durs physiques.

LVM requiert ce qui suit :

- le support « device-mapper » dans le noyau Linux (présent par défaut sur les noyaux Debian)
- La bibliothèque de support du « device-mapper » en espace utilisateur (du paquet `libdevmapper*`)
- le paquet des outils LVM2 en espace utilisateur (`lvm2`)

Démarrez l'apprentissage de LVM2 par la lecture des pages de manuel suivantes :

- `lvm(8)`: les bases du mécanisme de LVM2 (liste de toutes les commandes LVM2)
- `lvm.conf(5)`: le fichier de configuration pour LVM2
- `lvs(8)`: information sur les rapports relatifs aux volumes logiques
- `vgs(8)`: information sur les rapports relatifs aux groupes de volumes
- `pvs(8)`: information sur les rapports relatifs aux volumes physiques.

9.6.5 Configuration de systèmes de fichiers

Pour le système de fichiers [ext4](#), le paquet `e2fsprogs` fournit les éléments suivants :

- `mkfs.ext4(8)` pour créer un nouveau système de fichiers [ext4](#)
- `fsck.ext4(8)` pour vérifier et réparer un système de fichiers [ext4](#) existant
- `tune2fs(8)` pour configurer le superbloc d'un système de fichiers [ext4](#)
- `debugfs(8)` pour un débogage interactif du système de fichiers [ext4](#). (Il possède la commande `unde l` permettant de récupérer des fichiers effacés.)

Les commandes `mkfs(8)` et `fsck(8)` font partie du paquet `e2fsprogs` en tant que frontal à de nombreux programmes dépendant du système de fichiers (`mkfs.fstype` et `fsck.fstype`). Pour le système de fichiers [ext4](#), il y a `mkfs.ext4(8)` et `fsck.ext4(8)` (ils sont liés par un lien symboliquement à `mke2fs(8)` et `e2fsck(8)`).

Des commandes semblables sont disponibles pour chaque système de fichiers pris en charge par Linux.

paquet	popcon	taille	description
e2fsprogs	V:616, I:999	1511	utilitaires pour les systèmes de fichiers ext2/ext3/ext4
btrfs-progs	V:44, I:68	3959	utilitaire pour le système de fichiers Btrfs
reiserfsprogs	V:15, I:29	1132	utilitaire pour le système de fichiers Reiserfs
zfsutils-linux	V:19, I:23	1497	utilities for the OpenZFS filesystem
dosfstools	V:132, I:508	315	utilitaire pour le système de fichiers FAT (Microsoft :MS-DOS, Windows)
exfatprogs	V:0, I:1	161	utilities for the exFAT filesystem maintained by Samsung.
exfat-fuse	V:25, I:429	71	read/write exFAT filesystem (Microsoft) driver for FUSE.
exfat-utils	V:19, I:430	231	utilities for the exFAT filesystem maintained by the exfat-fuse author.
xfsprogs	V:22, I:101	3281	utilitaire pour le système de fichiers XFS (SGI : IRIX)
ntfs-3g	V:227, I:494	1483	read/write NTFS filesystem (Microsoft: Windows NT, ...) driver for FUSE.
jfsutils	V:1, I:12	1577	utilitaire pour le système de fichiers JFS (IBM : AIX, OS/2)
reiser4progs	V:0, I:4	1373	utilitaire pour le système de fichiers Reiser4
hfsprogs	V:0, I:7	389	utilitaire pour les systèmes de fichiers HFS et HFS Plus (Apple : Mac OS)
zerofree	V:3, I:100	25	programme pour mettre à zéro les blocs libres des systèmes de fichiers ext2 , ext3 et ext4

Table 9.20 – Liste des paquets de gestion des systèmes de fichiers

ASTUCE

Le système de fichiers [ext4](#) est le système de fichiers par défaut pour les systèmes Linux. Son utilisation est fortement recommandée, sauf cas spécifiques.

[Btrfs](#) status can be found at [Debian wiki on btrfs](#) and [kernel.org wiki on btrfs](#). It is expected to be the next default filesystem after the ext4 filesystem.

Certains outils permettent l'accès au système de fichiers sans prise en charge par le noyau Linux (consultez Section [9.8.2](#)).

9.6.6 Création et vérification de l'intégrité d'un système de fichiers

La commande `mkfs(8)` permet de créer un système de fichiers sur un système Linux. La commande `fsck(8)` permet de vérifier l'intégrité du système de fichiers et de le réparer sur un système Linux.

Maintenant, par défaut, Debian n'utilise pas `fsck` après la création d'un système de fichier.

**Attention**

En général, il n'est pas sûr de faire tourner `fsck` sur un **système de fichiers monté**.

ASTUCE

Vous pouvez exécuter la commande `fsck(8)` sans risque sur tous les systèmes de fichiers incluant le système de fichiers racine au redémarrage avec le paramètre `"enable_periodic_fsck"` dans `"/etc/mke2fs.conf"` et le compteur de montage maximum mis à 0 en utilisant `"tune2fs -c0 /dev/partition_name"`. Voir `mke2fs.conf(5)` et `tune2fs(8)`.

Vous trouverez les résultats de la commande `fsck(8)` lancée depuis le script de démarrage dans `« /var/log/fsck/ »`.

9.6.7 Optimisation du système de fichiers à l'aide des options de montage

La configuration statique de base du système de fichiers est donnée par `« /etc/fstab »`. Par exemple,

«file system»	«mount point»	«type»	«options»	«dump»	«pass»
<code>proc</code>	<code>/proc</code>	<code>proc</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
<code>UUID=709cbe4c-80c1-56db-8ab1-dbce3146d2f7</code>	<code>/</code>	<code>ext4</code>	<code>errors=remount-ro</code>	<code>0</code>	<code>1</code>
<code>UUID=817bae6b-45d2-5aca-4d2a-1267ab46ac23</code>	<code>none</code>	<code>swap</code>	<code>sw</code>	<code>0</code>	<code>0</code>
<code>/dev/scd0</code>	<code>/media/cdrom0</code>	<code>udf,iso9660</code>	<code>user,noauto</code>	<code>0</code>	<code>0</code>

ASTUCE

Un [UUID](#) (consultez Section [9.6.3](#)) peut être utilisé pour identifier un périphérique bloc au lieu des noms de périphériques blocs normaux comme `« /dev/sda1 »`, ou `« /dev/sda2 »`,...

Since Linux 2.6.30, the kernel defaults to the behavior provided by `"relatime"` option.

See `fstab(5)` and `mount(8)`.

9.6.8 Optimisation du système de fichiers à l'aide du superbloc

Les caractéristiques du système de fichiers peuvent être optimisées par l'intermédiaire de son superbloc en utilisant la commande `tune2fs(8)`.

- L'exécution de « `sudo tune2fs -l /dev/hda1` » affiche le contenu du superbloc du système de fichiers situé sur « `/dev/hda1` ».
- L'exécution de « `sudo tune2fs -c 50 /dev/hda1` » modifie la fréquence des vérifications du système de fichiers (exécution de `fsck` lors du démarrage) à 50 démarrages sur « `/dev/hda1` ».
- L'exécution de la commande « `sudo tune2fs -j /dev/hda1` » ajoute la possibilité de journalisation au système de fichiers, c'est-à-dire la conversion de système de fichiers de [ext2](#) vers [ext3](#) sur « `/dev/hda1` » (effectuez cela sur un système de fichiers non monté).
- L'exécution de « `sudo tune2fs -O extents,uninit_bg,dir_index /dev/hda1 && fsck -pf /dev/hda1` » le convertit de [ext3](#) vers [ext4](#) sur « `/dev/hda1` ». (À effectuer sur un système de fichiers non monté).

ASTUCE

En dépit de son nom, `tune2fs(8)` ne fonctionne pas uniquement sur le système de fichiers [ext2](#) mais aussi sur les systèmes de fichiers [ext3](#) et [ext4](#).

9.6.9 Optimisation du disque dur



AVERTISSEMENT

Veuillez vérifier votre matériel et lire la page de manuel `hdparm(8)` avant de jouer avec la configuration de vos disques durs parce que ce peut être assez dangereux pour l'intégrité des données.

Vous pouvez tester la vitesse de lecture d'un disque dur, par exemple « `/dev/hda` » par « `hdparm -tT /dev/hda` ». Vous pouvez accélérer certains disques (E)IDE avec « `hdparm -q -c3 -d1 -u1 -m16 /dev/hda` » en activant la prise en charge des E/S 32 bits (« (E)IDE 32-bit I/O support »), en positionnant l'indicateur d'utilisation de dma « `using_dma flag` », en positionnant l'indicateur de démasquage des interruptions (« `interrupt-unmask flag` ») et en positionnant les E/S multiples sur 15 secteurs (« `multiple 16 sector I/O` ») (dangereux !).

Vous pouvez tester la fonctionnalité de cache d'un disque dur, par exemple « `/dev/sda` » par « `hdparm -W /dev/sda` ». Vous pouvez désactiver le cache en écriture avec « `hdparm -W 0 /dev/sda` ».

Vous pouvez réussir à lire un CD-ROM vraiment mal pressé sur un lecteur de CD-ROM moderne rapide en le ralentissant avec « `setcd -x 2` ».

9.6.10 Optimisation du SSD

[Solid state drive \(SSD\)](#) is auto detected now.

Reduce unnecessary disk accesses to prevent disk wear out by mounting "tmpfs" on volatile data path in `/etc/fstab`.

9.6.11 Utiliser SMART pour prédire les défaillances des disques durs

Vous pouvez surveiller et enregistrer les disques durs conformes à [SMART](#) à l'aide du démon `smartd(8)`.

1. Activez la fonctionnalité [SMART](#) dans le BIOS.
 2. Installez le paquet `smartmontools`.
 3. Identifiez vos disques durs en affichant la liste avec `df(1)`.
 - Supposons que le disque dur à surveiller soit « `/dev/hda` ».
-

4. Contrôlez la sortie de « `smartctl -a /dev/hda` » pour voir si la fonctionnalité [SMART](#) est effectivement activée.
 - Si elle ne l'est pas, activez-la avec « `smartctl -s on -a /dev/hda` ».
5. Autorisez le fonctionnement du démon `smartd(8)` par l'action suivante :
 - décommentez « `start_smartd=yes` » dans le fichier « `/etc/default/smartmontools` » ;
 - restart the `smartd(8)` daemon by "sudo `systemctl restart smartmontools`".

ASTUCE

Le démon `smartd(8)` peut être personnalisé par l'intermédiaire du fichier `/etc/smartd.conf` y compris pour la manière d'être informé des avertissements.

9.6.12 Indication du répertoire de stockage temporaire à l'aide de `$TMPDIR`

Les applications créent normalement des fichiers temporaires dans le répertoire de stockage temporaire « `/tmp` ». Si « `/tmp` » ne fournit pas assez d'espace, vous pouvez indiquer un autre répertoire de stockage temporaire à l'aide de la variable `$TMPDIR` pour les programmes qui l'intègrent.

9.6.13 Étendre l'espace de stockage utile à l'aide de LVM

Les partitions créées sur le [gestionnaire de volumes logiques \(LVM\)](#) (« Logical Volume Manager ») (fonctionnalité de Linux) au moment de l'installation peuvent être facilement redimensionnées en y concaténant des extensions (« extents ») ou en tronquant les extensions sur plusieurs périphériques de stockage sans reconfiguration majeure du système.

9.6.14 Extension de l'espace de stockage en montant une autre partition

Si vous avez une partition vide (par exemple « `/dev/sdx` »), vous pouvez la formater avec `mkfs.ext4(1)` et la monter (« `mount(8)` ») sur un répertoire où vous avez besoin de davantage d'espace (vous devrez copier les données d'origine).

```
$ 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
```

ASTUCE

Vous pouvez aussi monter un fichier image de disque vide (consultez [Section 9.7.5](#)) en tant que périphérique de rebouclage (consultez [Section 9.7.3](#)). L'utilisation réelle du disque croît avec les données réellement enregistrées.

9.6.15 Extension de l'espace de stockage en remontant un autre répertoire

Si vous avez un répertoire vide (par exemple, « `/chemin/vers/répertoire-temporaire` ») sur une autre partition avec de l'espace disponible, vous pouvez le remonter avec l'option « `--bind` » vers un répertoire (par exemple « `répertoire-travail` ») où vous avez besoin de place.

```
$ sudo mount --bind /path/to/emp-dir work-dir
```

9.6.16 Expansion of usable storage space by overlay-mounting another directory

If you have usable space in another partition (e.g., `/path/to/empty` and `/path/to/work`), you can create a directory in it and stack that on to an old directory (e.g., `/path/to/old`) where you need space using the [OverlayFS](#) for Linux kernel 3.18 or newer (Debian Stretch 9.0 or newer).

```
$ sudo mount -t overlay overlay \
  -olowerdir=/path/to/old-dir,upperdir=/path/to/empty,workdir=/path/to/work
```

Here, `/path/to/empty` and `/path/to/work` should be on the RW-enabled partition to write on `/path/to/old`.

9.6.17 Extension de l'espace utilisable à l'aide de liens symboliques



Attention

This is a deprecated method. Some software may not function well with "symlink to a directory". Instead, use the "mounting" approaches described in the above.

Si vous avez un répertoire vide (par exemple, « `/chemin/vers/répertoire-temporaire` ») sur une autre partition avec de l'espace disponible, vous pouvez créer un lien symbolique vers ce répertoire avec `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
```



AVERTISSEMENT

N'utilisez pas de « liens symboliques vers un répertoire » pour un répertoire géré par le système comme par exemple « `/opt` ». De tels liens symboliques peuvent être écrasés lors de la mise à niveau du système.

9.7 Le fichier image du disque

Nous discutons ici des manipulations sur l'image disque.

9.7.1 Créer le fichier image du disque

Le fichier image du disque, « `disk.img` » d'un périphérique non monté, par exemple, le second périphérique SCSI ou Serial ATA « `/dev/sdb` » peut être créé en utilisant `cp(1)` ou `dd(1)` comme suit :

```
# cp /dev/sdb disk.img
# dd if=/dev/sdb of=disk.img
```

L'image disque du [master boot record \(MBR\)](#) (secteur principal d'amorçage) (consultez Section [9.6.2](#)) qui se trouve sur le premier secteur du disque primaire IDE peut être faite en utilisant `dd(1)` comme suit :

```
# dd if=/dev/hda of=mbr.img bs=512 count=1
# dd if=/dev/hda of=mbr-nopart.img bs=446 count=1
# dd if=/dev/hda of=mbr-part.img skip=446 bs=1 count=66
```

- « `mbr.img` » : MBR avec la table des partitions
- « `mbr-nopart.img` » : MBR sans la table des partitions
- « `mbr-part.img` » : table de partition du MBR seul

Si vous avez un périphérique SCSI ou Serial ATA comme disque d’amorçage, remplacez « `/dev/hda` » avec « `/dev/sda` ».

Si vous réalisez une image d’une partition du disque d’origine, remplacez « `/dev/hda` » par « `/dev/hda1` », etc.

9.7.2 Écrire directement sur le disque

Le fichier image du disque « `disk.img` » peut être écrit vers un disque non monté, par exemple le second disque SCSI « `/dev/sdb` » avec la taille correspondante par ce qui suit :

```
# dd if=disk.img of=/dev/sdb
```

De la même manière, le fichier image de la partition du disque, « `partition.img` » peut être écrit sur une partition non montée, par exemple, la première partition du second disque SCSI « `/dev/sdb1` » avec la taille correspondante comme suit :

```
# dd if=partition.img of=/dev/sdb1
```

9.7.3 Monter le fichier image du disque

L’image disque « `partition.img` », qui contient une partition image unique, peut être monté et démonté en utilisant le [périphérique de rebouclage \(loop device\)](#) de la manière suivante :

```
# losetup -v -f partition.img
Loop device is /dev/loop0
# mkdir -p /mnt/loop0
# mount -t auto /dev/loop0 /mnt/loop0
...hack...hack...hack
# umount /dev/loop0
# losetup -d /dev/loop0
```

Cela peut être simplifié de la manière suivante :

```
# mkdir -p /mnt/loop0
# mount -t auto -o loop partition.img /mnt/loop0
...hack...hack...hack
# umount partition.img
```

Chaque partition de l’image disque « `disk.img` » contenant plusieurs partitions peut être montée en utilisant le [périphérique de rebouclage \(loop device\)](#). Étant donné que le périphérique de rebouclage ne gère pas, par défaut, les partitions, il faut le réinitialiser de la manière suivante :

```
# modinfo -p loop # verify kernel capability
max_part:Maximum number of partitions per loop device
max_loop:Maximum number of loop devices
# losetup -a # verify nothing using the loop device
# rmmod loop
# modprobe loop max_part=16
```

Maintenant, le périphérique de rebouclage peut gérer jusqu’à 16 partitions.

```
# losetup -v -f disk.img
Loop device is /dev/loop0
# fdisk -l /dev/loop0
```

```
Disk /dev/loop0: 5368 MB, 5368709120 bytes
```

```
255 heads, 63 sectors/track, 652 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0x452b6464
```

Device	Boot	Start	End	Blocks	Id	System
/dev/loop0p1		1	600	4819468+	83	Linux
/dev/loop0p2		601	652	417690	83	Linux

```
# mkdir -p /mnt/loop0p1
# mount -t ext4 /dev/loop0p1 /mnt/loop0p1
# mkdir -p /mnt/loop0p2
# mount -t ext4 /dev/loop0p2 /mnt/loop0p2
...hack...hack...hack
# umount /dev/loop0p1
# umount /dev/loop0p2
# losetup -d /dev/loop0
```

En remplacement, des effets similaires peuvent être obtenus en utilisant les périphériques [device mapper](#) créés par `kpartx(8)` du paquet `kpartx` de la manière suivante :

```
# kpartx -a -v disk.img
...
# mkdir -p /mnt/loop0p2
# mount -t ext4 /dev/mapper/loop0p2 /mnt/loop0p2
...
...hack...hack...hack
# umount /dev/mapper/loop0p2
...
# kpartx -d /mnt/loop0
```

Note

Vous pouvez monter une partition unique d'une telle image de disque avec le [périphérique de rebouclage](#) en utilisant un décalage pour sauter le [MBR](#) ou autre. Mais c'est susceptible d'induire des erreurs.

9.7.4 Nettoyage d'un fichier image du disque

Un fichier image disque « `disk.img` » peut être nettoyé de tous les fichiers supprimés pour donner une image propre « `new.img` » de la manière suivante :

```
# 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
```

Si « `disk.img` » est un système de fichiers `ext2`, `ext3` ou `ext4`, vous pouvez aussi utiliser `zerofree(8)` du paquet `zerofree` de la manière suivante :

```
# losetup -f -v disk.img
Loop device is /dev/loop3
# zerofree /dev/loop3
# cp --sparse=always disk.img new.img
```

9.7.5 Réaliser le fichier image d'un disque vide

Le fichier image du disque vide « `disk.img` », qui pourra s'étendre jusqu'à 5Gio peut être fait en utilisant `dd(1)` comme suit :

```
$ dd bs=1 count=0 if=/dev/zero of=disk.img seek=5G
```

Vous pouvez créer un système de fichiers `ext4` sur cette image disque « `disk.img` » en utilisant le [périphérique de rebouclage \(loop device\)](#) de la manière suivante :

```
# losetup -f -v disk.img
Loop device is /dev/loop1
# mkfs.ext4 /dev/loop1
...hack...hack...hack
# losetup -d /dev/loop1
$ du --apparent-size -h disk.img
5.0G disk.img
$ du -h disk.img
83M disk.img
```

Pour « `disk.img` », sa taille de fichier est de 5.0 Gio et son utilisation disque est uniquement de 83Mio. Cette discordance est possible car `ext4` sait maintenir un [fichier creux \(sparse\)](#).

ASTUCE

L'utilisation réelle sur le disque du [fichier creux](#) croît au fur et à mesure qu'on y écrit des données .

En utilisant des opérations similaires sur les périphériques créés par [loop device](#) ou les périphériques [device mapper](#) comme dans Section 9.7.3, vous pouvez partitionner cette image disque « `disk.img` » en utilisant `parted(8)` ou `fdisk(8)`, et y créer un système de fichiers en utilisant `mkfs.ext4(8)`, `mkswap(8)`, etc.

9.7.6 Créer un fichier image ISO9660

On peut faire le fichier image [ISO9660](#), « `cd.iso` », depuis l'arborescence de répertoire source située à « `répertoire_source` », en utilisant `genisoimage(1)` fourni par [cdrkit](#) de la manière suivante :

```
# genisoimage -r -J -T -V volume_id -o cd.iso source_directory
```

De la même manière, on peut créer le fichier image ISO9660 amorçable « `cdboot.iso` » depuis une arborescence comme celle de `debian-installer` située en « `source_directory` », de la manière suivante :

```
# 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
```

Ici, le [chargeur d'amorçage Isolinux](#) (consultez Section 3.1.2) est utilisé pour l'amorçage.

Vous pouvez calculer la valeur de la somme `md5` (`md5sum`) et construire des image ISO9660 directement depuis un lecteur de CD-ROM de la manière suivante :

```
$ 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
```

**AVERTISSEMENT**

Vous devrez prendre garde d'éviter le bogue de lecture anticipée du système de fichiers ISO9660 de Linux comme ci-dessus afin d'obtenir les résultats corrects.

9.7.7 Écriture directe sur CD/DVD-R/RW

ASTUCE

Un DVD n'est qu'un gros CD pour `wodim(1)` qui est fourni par [cdrkit](#).

Vous pouvez rechercher un périphérique utilisable comme suit :

```
# wodim --devices
```

Le CD-R vierge est alors inséré dans le graveur de CD et le fichier image ISO9660 « `cd.iso` » est écrit vers le périphérique, par exemple; « `/dev/hda` » en utilisant `wodim(1)` de la manière suivante :

```
# wodim -v -eject dev=/dev/hda cd.iso
```

Si un CD-RW est utilisé à la place d'un CD-R, faites alors ce qui suit :

```
# wodim -v -eject blank=fast dev=/dev/hda cd.iso
```

ASTUCE

Si votre système de bureau monte automatiquement le CD, démontez-le par la commande « `sudo umount /dev/hda` » depuis un terminal avant d'utiliser `wodim(1)`.

9.7.8 Monter le fichier image ISO9660

Si « `cd.iso` » contient une image ISO9660, ce qui suit permet alors de le monter manuellement sur « `/cdrom` » :

```
# mount -t iso9660 -o ro,loop cd.iso /cdrom
```

ASTUCE

Les systèmes de bureau modernes peuvent monter automatiquement les supports amovibles tels que les CD formatés en ISO9660 (consultez Section [10.1.7](#)).

9.8 Les données binaires

Nous allons ici discuter de la manipulation directe des données binaires sur le support d'enregistrement.

paquet	popcon	taille	description
coreutils	V:901, I:999	17478	paquet de base utilisant <code>od(1)</code> pour vider des fichiers (HEX, ASCII, OCTAL, ...)
bsdmainutils	V:53, I:953	27	paquets utilitaires qui utilisent <code>hd(1)</code> pour vider les fichiers (HEX, ASCII, OCTAL, ...)
hexedit	V:1, I:12	73	éditeurs et visualisateurs binaires (HEX, ASCII)
bless	V:0, I:4	1028	éditeur hexadécimal complet (GNOME)
okteta	V:1, I:14	1504	éditeur hexadécimal complet (KDE4)
ncurses-hexedit	V:0, I:2	132	éditeur et visualisateur binaire (HEX, ASCII, EBCDIC)
beav	V:0, I:0	133	éditeur et visualisateur binaire (HEX, ASCII, EBCDIC, OCTAL, ...)

Table 9.21 – Liste des paquets permettant de visualiser et d’éditer des données binaires

9.8.1 Afficher et éditer des données binaires

La méthode la plus basique pour visualiser des données binaires est d’utiliser la commande « `od -t x1` ».

ASTUCE

HEX est utilisé comme l’acronyme du format [hexadécimal](#) en [base 16](#). OCTAL désigne le format [octal](#) en [base 8](#). ASCII est employé pour [Code américain standard pour l’échange d’informations](#) (« American Standard Code for Information Interchange ») c’est-à-dire le code pour texte normal en anglais. EBCDIC signifie [Code d’échange étendu décimal codé binaire](#) (« Extended Binary Coded Decimal Interchange Code »), il est utilisé avec par les systèmes d’exploitation des [mainframe IBM](#).

9.8.2 Manipulation des fichiers sans monter le disque

Il existe des outils permettant de lire et d’écrire des fichiers sans avoir à monter le disque.

paquet	popcon	taille	description
mtools	V:9, I:82	399	utilitaires pour les fichiers MSDOS sans les monter
hfsutils	V:0, I:6	184	utilitaires pour les fichiers HFS et HFS+ sans les monter

Table 9.22 – Liste des paquets pour manipuler les fichiers sans monter le disque

9.8.3 Redondance des données

Les systèmes s’appuyant sur le [RAID](#) logiciel offert par le noyau Linux permettent une redondance des données au niveau du système de fichiers du noyau afin d’obtenir un haut niveau de fiabilité du système de stockage.

Il existe aussi des outils pour ajouter des données de redondance aux fichiers au niveau du programme applicatif permettant d’obtenir de hauts niveaux de fiabilité de stockage.

9.8.4 Récupération de fichiers de données et analyse par autopsie

Il y a des outils pour la récupération des données et l’analyse par autopsie.

ASTUCE

Vous pouvez annuler l’effacement de fichiers sur un système de fichiers `ext2` en utilisant les commandes `list_deleted_inodes` et `unde1` de `debugfs(8)` dans le paquet `e2fsprogs`.

paquet	popcon	taille	description
par2	V:4, I:26	271	Parity Archive Volume Set, pour vérifier et réparer des fichiers
dvdisaster	V:0, I:2	1741	protection des supports CD et DVD contre les pertes de données, les rayures et le vieillissement
dvbackup	V:0, I:0	413	outil de sauvegarde utilisant des caméscopes MiniDV (fournissant rsbep(1))
vdmfec	V:0, I:0	97	récupération de blocs en utilisant une correction d'erreur vers l'avant (« Forward Error Correction »)

Table 9.23 – Liste d'outils pour ajouter des données de redondance aux fichiers

paquet	popcon	taille	description
testdisk	V:3, I:37	1426	utilitaires pour l'examen de partitions et la récupération de disque
magicrescue	V:0, I:3	259	utilitaire pour la récupération de fichiers et de recherche des octets magiques
scalpel	V:0, I:4	87	récupérateur de fichiers sobre de haute performance
myrescue	V:0, I:3	83	récupérer des données depuis des disques endommagés
extundelete	V:0, I:11	147	utilitaire pour récupérer des fichiers effacés d'un système de fichiers ext3/4
ext4magic	V:0, I:5	233	utilitaire pour récupérer des fichiers effacés d'un système de fichiers ext3/4
ext3grep	V:0, I:3	293	outil pour aider à la récupération de fichiers effacés sur un système de fichiers ext3
scrounge-ntfs	V:0, I:3	50	programme de récupération de données pour les systèmes de fichiers NTFS
gzrt	V:0, I:0	33	boîte à outils de récupération gzip
sleuthkit	V:2, I:25	1511	outil pour autopsie (« forensics analysis »). (Sleuthkit)
autopsy	V:0, I:1	1027	interface graphique à SleuthKit
foremost	V:0, I:7	100	application d'autopsie pour la récupération de données
guymager	V:0, I:1	1035	outil de création d'image d'autopsie basée sur Qt
dcfldd	V:0, I:5	106	version améliorée de dd pour les autopsies et la sécurité

Table 9.24 – Liste de paquets pour la récupération de données et l'analyse par autopsie

9.8.5 Éclater un gros fichier en petits fichiers

Lorsque les données ont un volume trop important pour pouvoir être sauvegardée dans un seul fichier, vous pouvez en sauvegarder le contenu après l'avoir éclaté en morceaux de, par exemple, 2000Mio et réassembler ces morceaux par la suite sous la forme du fichier d'origine.

```
$ split -b 2000m large_file
$ cat x* >large_file
```

**Attention**

Assurez-vous ne pas pas avoir de nom de fichier commençant par « x » afin d'éviter des plantages de nom.

9.8.6 Effacer le contenu d'un fichier

Pour effacer le contenu d'un fichier comme, par exemple, un fichier journal, n'utilisez pas la commande `rm(1)` pour supprimer le fichier et recréer ensuite un fichier vide parce qu'on peut encore accéder au fichier dans l'intervalle entre les commandes. Voici la manière sûre d'effacer le contenu d'un fichier :

```
$ :>file_to_be_cleared
```

9.8.7 Fichiers fictifs

Les commandes suivantes créent des fichiers factices ou vides.

```
$ 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
```

Vous obtiendrez les fichiers suivants :

- « 5kb.file » avec 5K de zéros ;
- « 7mb.file » avec 7Mo de données aléatoires ;
- « zero.file » devrait être un fichier de 0 octet. S'il existait, son `mtime` est mis à jour alors que son contenu et sa taille sont conservés ;
- « alwayszero.file » fait toujours 0 octet. S'il existait son `mtime` est mis à jour et son contenu vidé.

9.8.8 Effacer l'ensemble du disque dur

Il existe plusieurs manières d'effacer complètement les données d'un périphérique semblable à un disque dur, par exemple, une clé USB se trouvant en « `/dev/sda` ».

**Attention**

Vérifiez d'abord l'emplacement de votre clé USB avec `mount(8)` avant d'exécuter ces commandes. Le périphérique pointé par « `/dev/sda` » peut être le disque dur SCSI ou un disque dur SATA sur lequel se trouve l'ensemble de votre système.

Effacer tout le contenu du disque en réinitialisant toutes les données à 0 avec la commande suivante :

```
# dd if=/dev/zero of=/dev/sda
```

Tout effacer en écrasant les données existantes par des données aléatoires par la commande suivante :

```
# dd if=/dev/urandom of=/dev/sda
```

Effacer de manière très efficace toutes les données en les écrasant avec des données aléatoires par la commande suivante :

```
# shred -v -n 1 /dev/sda
```

Comme `dd(1)` est disponible depuis l'interpréteur de commandes de nombreux CD amorçables de Linux tels que le CD de l'installateur Debian, vous pouvez effacer complètement votre système installé en lançant la commande d'effacement du disque dur du système, par exemple, « `/dev/hda` », « `/dev/sda` », etc. depuis un tel support,

9.8.9 Effacer l'ensemble du disque dur

Une zone inutilisée du disque dur (ou d'une clé mémoire USB), par ex. « `/dev/sdb1` » peut encore contenir les données effacées elles-mêmes puisqu'elles ne sont que déliées du système de fichiers. Elles peuvent être nettoyées en les surchargeant.

```
# 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
```



AVERTISSEMENT

C'est en général suffisamment bon pour votre clé mémoire USB. Mais ce n'est pas parfait. La plupart des noms des fichiers effacés et leurs attributs peuvent être cachés et rester dans le système de fichiers.

9.8.10 Récupérer des fichiers supprimés mais encore ouverts

Même si vous avez accidentellement supprimé un fichier, tant que ce fichier est en cours d'utilisation par une application quelconque, (en mode lecture ou écriture); il est possible de récupérer un tel fichier.

Essayez, par exemple, ce qui suit :

```
$ 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
```

Exécutez sur un autre terminal (lorsque vous avez le paquet `lsOf` installé) comme suit

```
$ ls -li bar
2228329 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:02 bar
$ lsof |grep bar|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/bar
$ rm bar
$ lsof |grep bar|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/bar (deleted)
$ cat /proc/4775/fd/4 >bar
$ ls -li bar
2228302 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:05 bar
$ cat bar
foo
```

9.8.11 Rechercher tous les liens physiques

Les fichiers ayant des liens physiques peuvent être identifiés par « `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
```

« `tutu` » et « `toto` » ont tous les deux un nombre de liens égal à « 2 » (>1), ce qui indique qu'ils ont des liens physiques. Leur numéro d'[inode](#) commun est « 2738404 ». Cela signifie qu'ils représentent le même fichier lié par des liens physiques. Si vous n'arrivez pas à trouver de fichiers liés par des liens physiques, vous pouvez les rechercher par[inode](#), par exemple « 2738404 », de la manière suivante :

```
# find /path/to/mount/point -xdev -inum 2738404
```

9.8.12 Consommation d'espace disque invisible

Tous les fichiers supprimés mais ouverts prennent de l'espace disque même s'ils ne sont pas visibles par la commande `du(1)` normale. On peut en afficher la liste avec leur taille par la commande suivante :

```
# lsof -s -X / |grep deleted
```

9.9 Astuces de chiffrement des données

Avec un accès physique à votre PC, n'importe qui peut facilement obtenir les privilèges de l'administrateur et accéder à tous les fichiers de votre PC (consultez Section [4.6.4](#)). Cela signifie qu'un système avec un mot de passe de connexion ne permet pas de sécuriser vos données personnelles ou sensibles en cas de vol de votre PC. Vous devez déployer des technologies de chiffrements des données pour assurer cette protection. Bien que [GNU privacy guard](#) (consultez Section [10.3](#)) puisse chiffrer des fichiers, il demande quelques efforts de la part de l'utilisateur.

[Dm-crypt](#) facilitates automatic data encryption via native Linux kernel modules with minimal user efforts using [device-mapper](#).



Attention

Le chiffrement des données a un coût en matière de temps processeur, etc. Veuillez comparer ses avantages à son coût.

paquet	popcon	taille	description
cryptsetup	V:41, I:77	407	utilitaires pour chiffrer un périphérique en mode bloc (dm-crypt / LUKS)
cryptmount	V:4, I:5	228	utilitaires pour chiffrer un périphérique en mode bloc (dm-crypt / LUKS), l'accent étant mis sur le montage et le démontage par un utilisateur normal
fscrypt	V:0, I:0	4688	utilities for Linux filesystem encryption (fscrypt)
libpam-fscrypt	V:0, I:0	4904	PAM module for Linux filesystem encryption (fscrypt)

Table 9.25 – Liste d'utilitaires de chiffrement des données

Note

Le système Debian dans son ensemble peut être installé sur un disque chiffré par l'[installateur debian](#) (lenny ou plus récent) en utilisant [dm-crypt/LUKS](#) et [initramfs](#).

ASTUCE

Consultez Section [10.3](#) pour un utilitaire de chiffrement de l'espace utilisateur : [GNU Privacy Guard](#).

9.9.1 Chiffrement des disques amovibles à l'aide de dm-crypt/LUKS

Vous pouvez chiffrer le contenu des périphériques de masse amovible, par exemple, une clé USB sur « `/dev/sdx` » en utilisant [dm-crypt/LUKS](#). Il suffit de le formater de la manière suivante :

```
# badblocks -c 1024 -s -w -t random -v /dev/sdx
# fdisk /dev/sdx
... "n" "p" "1" "return" "return" "w"
# cryptsetup luksFormat /dev/sdx1
...
# cryptsetup open --type luks /dev/sdx1 sdx1
...
# ls -l /dev/mapper/
total 0
crw-rw---- 1 root root 10, 60 2008-10-04 18:44 control
brw-rw---- 1 root disk 254, 0 2008-10-04 23:55 sdx1
# mkfs.vfat /dev/mapper/sdx1
...
# cryptsetup luksClose sdx1
```

Il peut alors être monté simplement comme un périphérique normal sur « `/media/étiquette_disque` » à l'exception de la demande de mot de passe (consultez Section [10.1.7](#)) sous un environnement de bureau moderne tel que GNOME en utilisant `gnome-mount(1)`. La différence est que toutes les données qui y seront écrites seront chiffrées. Vous pouvez aussi formater le support dans un système de fichiers différent, par exemple ext4 avec « `mkfs.ext4 /dev/sdx1` ».

Note

Si vous êtes vraiment paranoïaque en ce qui concerne la sécurité des données, vous pourrez réécrire par dessus plusieurs fois (à l'aide de la commande « `badblocks` » dans l'exemple ci-dessus). Cette opération prend cependant beaucoup de temps.

9.9.2 Partition d'échange chiffrée avec dm-crypt

Supposons que votre « `/etc/fstab` » d'origine comporte ce qui suit :

```
/dev/sda7 swap sw 0 0
```

Vous pouvez chiffrer la partition d'échange en utilisant [dm-crypt](#) par ce qui suit :

```
# aptitude install cryptsetup
# swapoff -a
# echo "cswap /dev/sda7 /dev/urandom swap" >> /etc/crypttab
# perl -i -p -e "s/\\dev\\/sda7/\\dev\\/mapper\\/cswap/" /etc/fstab
# systemctl restart cryptdisks
...
# swapon -a
```

9.9.3 Monter des disques amovibles chiffrés à l'aide de dm-crypt/LUKS

Une partition de disque chiffrée créée avec dm-crypt/LUKS sur « /dev/sdc5 » peut être montée sur « /mnt » de la façon suivante :

```
$ 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 Le noyau

Debian distribue des [noyaux Linux](#) modulaires sous forme de paquets pour les architectures prises en compte.

If you are reading this documentation, you probably don't need to compile Linux kernel by yourself.

9.10.1 Paramètres du noyau

De nombreuses fonctionnalités du noyau peuvent être configurées par l'intermédiaire de paramètres du noyau de la manière suivante :

- Paramètres du noyau initialisés par le gestionnaire d'amorçage (consultez Section [3.1.2](#))
- Paramètres du noyau modifiés par `sysctl(8)` lors du fonctionnement du système pour ceux auxquels on a accès par l'intermédiaire de `sysfs` (consultez Section [1.2.12](#))
- Paramètres des modules définis par les paramètres de `modprobe(8)` lors de l'activation d'un module (consultez Section [9.7.3](#))

See "The Linux kernel user's and administrator's guide » [The kernel's command-line parameters](#)" for the detail.

9.10.2 En-têtes du noyau

La plupart des **programmes normaux** n'ont pas besoin des en-têtes du noyau et peuvent de fait être cassés si vous les utilisez directement pour la compilation. Ils devront être compilés avec les en-têtes se trouvant dans « /usr/include/linux » et « /usr/include/asm » qui sont fournis, sur les systèmes Debian, par le paquet `libc6-dev` (créé à partir du paquet source `glibc`).

Note

For compiling some kernel-specific programs such as the kernel modules from the external source and the automounter daemon (amd), you must include path to the corresponding kernel headers, e.g. "-I/usr/src/linux-particular-version/include/", to your command line.

9.10.3 Compiler le noyau et les modules associés

Debian a sa propre manière de compiler le noyau et les modules associés.

paquet	popcon	taille	description
build-essential	I:500	20	paquets essentiels pour la construction de paquets Debian : make, gcc, ...
bzip2	V:147, I:973	122	utilitaires de compression et de décompression des fichiers bz2
libncurses5-dev	I:113	6	bibliothèques de développement et documentations pour ncurses
git	V:312, I:494	36515	git :: système distribué de contrôle de version utilisé par le noyau de Linux
fakeroot	V:32, I:518	228	fournit l'environnement fakeroot pour construire le paquet sans être administrateur (« root »)
initramfs-tools	V:442, I:989	112	outil pour construire une image mémoire initiale (« initramfs ») (spécifique à Debian)
dkms	V:99, I:210	295	prise en charge dynamique des modules du noyau (dynamic kernel module support : DKMS) (générique)
module-assistant	V:1, I:30	391	helper tool to make module package (Debian specific)
devscripts	V:8, I:54	2654	scripts d'assistance pour un responsable de paquet Debian (spécifique à Debian)

Table 9.26 – Liste des paquets-clés à installer pour la compilation du noyau sur un système Debian

Si vous utilisez un `initrd` dans Section 3.1.2, veuillez lire les informations correspondantes dans `initramfs-tools(8)`, `update-initramfs(8)`, `mkinitramfs(8)` et `initramfs.conf(5)`.

**AVERTISSEMENT**

Ne mettez pas de liens symboliques vers le répertoire de l'arborescence des source (par exemple « /usr/src/linux* ») depuis « /usr/include/linux » et « /usr/include/asm » lors de la compilation des sources du noyau de Linux. (Certains documents périmés le suggèrent).

Note

Lors de la compilation du dernier noyau de Linux sous un système Debian stable, l'utilisation des derniers outils rétroportés depuis la distribution Debian unstable peuvent être nécessaires.

`module-assistant(8)` (or its short form `m-a`) helps users to build and install module package(s) easily for one or more custom kernels.

Le [gestionnaire de modules dynamique du noyau](#) (« dynamic kernel module support (DKMS) ») est une nouvelle architecture indépendante de la distribution conçue pour permettre la mise à jour de modules individuels du noyau sans modifier l'ensemble du noyau. Cela est utilisé pour la maintenance de modules hors arborescence. Cela rend aussi très facile la reconstruction des modules après la mise à niveau des noyaux.

9.10.4 Compiler les sources du noyau : recommandations de l'équipe en charge du noyau Debian

Pour construire des paquets binaires d'un noyau personnalisé à partir des sources du noyau amont, vous devriez utiliser la cible « `deb-pkg` » fournie pour cela.

```
$ sudo apt-get build-dep linux
$ cd /usr/src
$ wget http://www.kernel.org/pub/linux/kernel/v3.11/linux-version.tar.bz2
$ tar -xjvf linux-version.tar.bz2
$ cd linux-version
$ cp /boot/config-version .config
$ make menuconfig
...
$ make deb-pkg
```

ASTUCE

Le paquet `linux-source-version` fournit les sources du noyau Linux avec les correctifs Debian en tant que « `/usr/src/linux-version.tar.bz2` ».

Pour construire des paquets binaires particuliers à partir du paquet source Debian, vous devriez utiliser les cibles « `binary-arch_arch` » dans « `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
```

Consultez les renseignements complémentaires :

- wiki Debian : [KernelFAQ](#) ;
- wiki Debian : [DebianKernel](#) ;
- Debian Linux Kernel Handbook: <https://kernel-handbook.debian.net>

9.10.5 Pilotes de matériel et microprogramme

The hardware driver is the code running on the main CPUs of the target system. Most hardware drivers are available as free software now and are included in the normal Debian kernel packages in the `main` area.

- Pilote de [processeur graphique](#)
 - Pilote Intel (`main`)
 - Pilote AMD ou ATI (`main`)
 - Pilote NVIDIA (`main` pour le pilote [nouveau](#) et `non-free` pour les pilotes binaires pris en charge par le constructeur)
- Pilote de [Softmodem](#)
 - Paquets `martian-modem` et `sl-modem-dkms` (`non libres`)

The firmware is the code or data loaded on the device attach to the target system (e.g., CPU [microcode](#), rendering code running on GPU, or [FPGA](#) / [CPLD](#) data, ...). Some firmware packages are available as free software but many firmware packages are not available as free software since they contain sourceless binary data. Installing these firmware data is essential for the device to function as expected.

- The firmware data packages containing data loaded to the volatile memory on the target device.
 - `firmware-linux-free` (`main`)
 - `firmware-linux-nonfree` (`non libre`)
 - `firmware-linux-*` (`non libre`)
-

- *-firmware (non libre)
- intel-microcode (non libre)
- amd64-microcode (non libre)
- The firmware update program packages which update data on the non-volatile memory on the target device.
 - [fwupd](#) (main): Firmware update daemon which downloads firmware data from [Linux Vendor Firmware Service](#).
 - [gnome-firmware](#) (main): GTK front end for fwupd
 - [plasma-discover-backend-fwupd](#) (main): Qt front end for fwupd

Veuillez remarquer que les paquets de `non-free` et `contrib` ne font pas partie du système Debian. La configuration pour activer et désactiver les sections `non-free` et `contrib` est décrite en Section 2.1.4. Vous devriez être conscient des aspects négatifs associés à l'utilisation de paquets de `non-free` et `contrib` décrits dans Section 2.1.5.

Please also note that the firmware data downloaded by [fwupd](#) from [Linux Vendor Firmware Service](#) and loaded to the running Linux kernel may be `non-free`.

9.11 Système virtualisé

L'utilisation d'un système virtualisé permet de faire tourner simultanément plusieurs instances du système sur une plateforme unique.

ASTUCE

Consultez <http://wiki.debian.org/fr/SystemVirtualization> .

9.11.1 Virtualization and emulation tools

There are several [virtualization](#) and emulation tool platforms.

- Complete [hardware emulation](#) packages such as ones installed by the [games-emulator](#) metapackage
- Mostly CPU level emulation with some I/O device emulations such as [QEMU](#)
- Mostly CPU level virtualization with some I/O device emulations such as [Kernel-based Virtual Machine \(KVM\)](#)
- OS level container virtualization with the kernel level support such as [LXC \(Linux Containers\)](#), [Docker](#), ...
- OS level filesystem access virtualization with the system library call override on the file path such as [chroot](#)
- OS level filesystem access virtualization with the system library call override on the file ownership such as [fakeroot](#)
- OS API emulation such as [Wine](#)
- Interpreter level virtualization with its executable selection and run-time library overrides such as [virtualenv](#) and [venv](#) for Python

The container virtualization uses Section 4.7.4 and it is the backend technology of Section 7.6.

Here are some packages to help you to setup the virtualized system.

Consultez l'article de Wikipedia [Comparaison de machines pour plateforme virtuelle](#) pour une comparaison détaillée entre les différentes solutions de plateformes de virtualisation.

9.11.2 Étapes de la virtualisation

Note

Les noyaux par défaut de Debian prennent en charge [KVM](#) depuis Lenny.

La [virtualisation](#) met en œuvre plusieurs étapes :

paquet	popcon	taille	description
schroot	V:7, I:9	2708	outil spécialisé pour l'exécution d'un paquet binaire de Debian dans un chroot
sbuid	V:1, I:4	271	outil pour construire des paquets binaires de Debian depuis les sources Debian
debootstrap	V:5, I:62	298	amorcer un système Debian de base (écrit en sh)
cdebootstrap	V:0, I:2	116	amorcer un système Debian (écrit en C)
virt-manager	V:10, I:43	2298	Gestionnaire de machine virtuelle : application du gestionnaire de bureau pour la gestion des machines virtuelles
libvirt-clients	V:46, I:64	1130	programmes pour la bibliothèque libvirt
games-emulator	I:0	26	games-emulator : Debian's emulators for games
bochs	V:0, I:1	7194	Bochs : émulateur PC IA-32
qemu	I:31	96	QEMU : émulateur de processeur générique rapide
qemu-system	I:21	97	QEMU : binaires pour l'émulation d'un système complet
qemu-user	V:0, I:11	89683	QEMU : binaires pour l'émulation en mode utilisateur
qemu-utils	V:11, I:108	6077	QEMU : utilitaires
qemu-kvm	V:8, I:55	107	KVM : virtualisation complète sur les plateformes x86 ayant une virtualisation assistée par le matériel
virtualbox	V:11, I:14	107018	VirtualBox : solution de virtualisation x86 sur i386 et amd64
xen-tools	V:0, I:4	727	outils pour gérer le serveur virtuel XEN de Debian
wine	V:16, I:76	191	Wine : implémentation de l'API Windows (suite standard)
dosbox	V:2, I:18	2702	DOSBox : émulateur x86 avec graphisme Tandy/Herc/CGA/EGA/VGA/SVGA, son et DOS
dosemu	V:0, I:2	4891	DOSEMU : l'émulateur DOS de Linux
lxc	V:11, I:14	18971	Conteneurs Linux outils de l'espace utilisateur
python3-venv	V:2, I:50	6	venv for creating virtual python environments (system library)
python3-virtualenv	V:9, I:64	410	virtualenv for creating isolated virtual python environments
python3-pipx	I:0	NOT_FOUND	pipx for installing python applications in isolated environments

Table 9.27 – Liste des outils de virtualisation

- Créer un système de fichiers vide (une arborescence de fichiers ou une image disque).
 - L'arborescence de fichiers peut être créée par « `mkdir -p /path/to/chroot` ».
 - L'image disque brute peut être créée à l'aide de `dd(1)` (consultez Section 9.7.1 et Section 9.7.5).
 - `qemu-img(1)` peut être utilisé pour créer et convertir des fichiers d'image disque pris en charge par [QEMU](#).
 - Les formats de fichier brut et [VMDK](#) peuvent être utilisés en tant que formats courants par les outils de virtualisation.
- Monter l'image disque dans le système de fichiers avec `mount(8)` (optionnel).
 - Pour l'image disque brute, le montage doit être fait avec un [périphérique de rebouclage](#) ou des périphériques [device mapper](#) (consultez Section 9.7.3).
 - Les images disques prises en charge par [QEMU](#) seront montées en tant que [périphériques réseau en mode bloc](#) (consultez Section 9.11.3).
- Peupler le système de fichiers cible avec les données requises.
 - L'utilisation de programmes tels que `debootstrap` et `cdebootstrap` facilite ce processus (consultez Section 9.11.4).
 - Utiliser les installateurs des systèmes d'exploitation sous l'émulation du système complet.
- Lancer un programme dans l'environnement virtualisé.
 - [chroot](#) fournit un environnement virtualisé de base, suffisant pour y compiler des programmes, y faire tourner des applications en mode console et des démons.
 - [QEMU](#) fournit une émulation de processeur interplateformes.
 - [QEMU](#) avec [KVM](#) fournit une émulation système complète avec la [virtualisation assistée par le matériel](#).
 - [VirtualBox](#) fournit une émulation du système complet sur i386 amd64 avec ou sans la [virtualisation assistée par le matériel](#).

9.11.3 Monter le fichier image du disque virtuel

Pour le fichier image disque brut, consultez Section 8.1.

Pour d'autres fichiers d'images disques virtuels, vous pouvez utiliser `qemu-nbd(8)` pour les exporter en utilisant le protocole [network block device](#) et en les montant à l'aide du module `nbd` du noyau.

`qemu-nbd(8)` gère les formats de disques pris en compte par [QEMU](#) : [QEMU](#) gère les formats de disques suivants `raw`, [qcow2](#), [qcow](#), [vmdk](#), [vdi](#), [bochs](#), `cow` (mode utilisateur de Linux copy-on-write), [parallels](#), [dmg](#), [cloop](#), [vpc](#), `vfat` (VFAT virtuelle) et `host_device`.

Le [network block device](#) peut gérer des partitions de la même manière que le [périphérique de rebouclage](#) (« `loop device` ») (consultez Section 9.7.3). Vous pouvez monter la première partition de « `disk.img` » de la manière suivante :

```
# modprobe nbd max_part=16
# qemu-nbd -v -c /dev/nbd0 disk.img
...
# mkdir /mnt/part1
# mount /dev/nbd0p1 /mnt/part1
```

ASTUCE

Vous ne pouvez exporter que la première partition de « `disk.img` » en utilisant l'option « `-P 1` » de `qemu-nbd(8)`.

9.11.4 Système protégé (chroot)

If you wish to try a new Debian environment from a terminal console, I recommend you to use [chroot](#). This enables you to run console applications of Debian unstable and `testing` without usual risks associated and without rebooting. `chroot(8)` is the most basic way.

**Attention**

Examples below assumes both parent system and chroot system share the same amd64 CPU architecture.

Although you can manually create a `chroot(8)` environment using `debootstrap(1)`. But this requires non-trivial efforts.

The [sbuild](#) package to build Debian packages from source uses the chroot environment managed by the [schroot](#) package. It comes with helper script `sbuild-createchroot(1)`. Let's learn how it works by running it under `script(1)` as follows.

```
$ sudo mkdir -p /srv/chroot
$ sudo sbuild-createchroot -v --include=eatmydata,ccache unstable /srv/chroot/unstable- ↵
amd64-sbuild http://deb.debian.org/debian
```

You see how `debootstrap(8)` populates system data for `unstable` environment under `"/srv/chroot/unstable-amd64-sbuild"` for a minimal build system.

You can login to this environment using `schroot(1)`.

```
$ sudo schroot -v -c chroot:unstable-amd64-sbuild
```

You see how a system shell running under `unstable` environment is created.

Note

The `"/usr/sbin/policy-rc.d"` file which always exits with 101 prevents daemon programs to be started automatically on the Debian system. See `"/usr/share/doc/sysv-rc/README.policy-rc.d.gz"`.

Note

Some programs under chroot may require access to more files from the parent system to function than `sbuild-createchroot` provides as above. For example, `"/sys"`, `"/etc/passwd"`, `"/etc/group"`, `"/var/run/utmp"`, `"/var/log/wtmp"`, etc. may need to be bind-mounted or copied.

ASTUCE

The `sbuild` package helps to construct a chroot system and builds a package inside the chroot using `schroot` as its backend. It is an ideal system to check build-dependencies. See more on [sbuild at Debian wiki](#) and [sbuild configuration example in "Guide for Debian Maintainers"](#).

9.11.5 Systèmes de bureaux multiples

If you wish to try a new GUI Desktop environment of any OS, I recommend you to use [QEMU](#), [KVM](#), or [VirtualBox](#) on a Debian `stable` system to run multiple desktop systems safely using [virtualization](#). These enable you to run any desktop applications including ones of Debian `unstable` and `testing` without usual risks associated with them and without rebooting. The configuration of these tools are relatively straight forward.

Since pure [QEMU](#) is very slow, it is recommended to accelerate it with [KVM](#) when the host system supports it.

L'image disque virtuelle « `virtdisk.qcow2` » qui contient un système Debian pour [QEMU](#) peut être créée en utilisant [un CD minimal de debian-installer](#) de la manière suivante :

```
$ wget http://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
...
```

Vous trouverez des astuces supplémentaires sur le [wiki Debian : QEMU](#).

[VirtualBox](#) est fourni avec des outils ayant une interface graphique [Qt](#) et est assez intuitif. Son interface graphique et ses outils en ligne de commandes sont expliqués dans le [Manuel de l'utilisateur de VirtualBox](#) et le [Manuel de l'utilisateur VirtualBox \(PDF\)](#).

ASTUCE

Faire tourner d'autres distributions de GNU/Linux comme [Ubuntu](#) et [Fedora](#) sous une [virtualisation](#) est une bonne manière d'en étudier les astuces de configuration. D'autres systèmes d'exploitation propriétaires peuvent aussi tourner de manière agréable sous la [virtualisation](#) GNU/Linux.

Chapitre 10

Gestion des données

Des outils et astuces pour gérer les données binaires ou textuelles sur le système sont décrits.

10.1 Partager, copier et archiver



AVERTISSEMENT

Il ne faut pas accéder de manière non coordonnée en écriture à des périphériques et des fichiers à haut trafic depuis différents processus pour éviter une [compétition d'accès](#) (« [race condition](#) ». Pour l'éviter, on peut utiliser les mécanismes de [verrouillage de fichier](#) (« [File locking](#) » en utilisant `flock(1)`).

La sécurité des données et leur partage contrôlé présentent plusieurs aspects.

- La création d'une archive des données
- L'accès à un stockage distant
- La duplication
- Le suivi de l'historique des modifications
- La facilité de partage des données
- La prévention de l'accès non autorisé aux données
- La détection des modifications de fichier non autorisées

Cela peut être réalisé avec certaines combinaisons d'outils.

- Outils d'archivage et de compression
 - Outils de copie et de synchronisation
 - Systèmes de fichiers par le réseau
 - Supports d'enregistrement amovibles
 - L'interpréteur de commandes sécurisé
 - Le système d'authentification
 - Outils de système de contrôle de version
 - Outils de hachage et de chiffrement
-

paquet	popcon	taille	extension	commande	commentaire
tar	V:914, I:999	3152	.tar	tar(1)	archivageur standard (standard de fait)
cpio	V:489, I:998	1144	.cpio	cpio(1)	archivageur de style UNIX System V, utiliser avec find(1)
binutils	V:164, I:673	97	.ar	ar(1)	archivageur pour la création de bibliothèques statiques
fastjar	V:2, I:25	183	.jar	fastjar(1)	archivageur pour Java (semblable à zip)
pax	V:12, I:24	170	.pax	pax(1)	nouvel archivageur standard POSIX, compromis entre tar et cpio
gzip	V:891, I:999	242	.gz	gzip(1) , zcat(1) , ...	utilitaire de compression GNU LZ77 (standard de fait)
bzip2	V:147, I:973	122	.bz2	bzip2(1) , bzcat(1) , ...	utilitaire de transformée par tri de blocs de Burrows-Wheeler permettant un taux de compression plus élevé que gzip(1) (plus lent que gzip avec une syntaxe similaire)
lzma	V:2, I:27	149	.lzma	lzma(1)	LZMA compression utility with higher compression ratio than gzip(1) (deprecated)
xz-utils	V:453, I:980	612	.xz	xz(1) , xzdec(1) , ...	utilitaire de compression XZ avec un plus haut taux de compression que bzip2(1) (plus lent que gzip mais plus rapide que bzip2 ; le remplaçant de LZMA utilitaire de compression)
zstd	V:4, I:24	1902	.zstd	zstd(1) , zstdcat(1) , ...	Zstandard fast lossless compression utility
p7zip	V:79, I:454	987	.7z	7zr(1) , p7zip(1)	archivageur de fichiers 7-Zip avec un haut taux de compression (compression LZMA)
p7zip-full	V:102, I:469	4664	.7z	7z(1) , 7za(1)	archivageur de fichiers 7-Zip avec un haut taux de compression (compression LZMA et autres)
lzop	V:10, I:85	164	.lzo	lzop(1)	utilitaire de compression LZO avec de plus hautes vitesses de compression et de décompression que gzip(1) (plus faible taux de compression que gzip avec une syntaxe similaire)
zip	V:49, I:427	623	.zip	zip(1)	InfoZIP : outil d'archive et de compression DOS
unzip	V:132, I:792	385	.zip	unzip(1)	InfoZIP : outil de désarchivage et de décompression DOS

Table 10.1 – Liste des outils d'archivage et de compression

10.1.1 Outils d'archivage et de compression

Voici un résumé des outils d'archivage et de compression disponible sur le système Debian :



AVERTISSEMENT

Ne positionnez pas la variable « \$TAPE » à moins que vous ne sachiez à quoi vous attendre. Elle modifie le comportement de `tar(1)`.

- L'archive `tar(1)` utilise l'extension de fichier « `.tgz` » ou « `.tar.gz` ».
- L'archive `tar(1)` avec compression `xz` utilise l'extension de fichier « `.txz` » ou « `.tar.xz` ».
- La méthode de compression habituelle des outils FOSS tels que `tar(1)` a été modifiée de la manière suivante : `gzip` → `bzip2` → `xz`
- `cp(1)`, `scp(1)` et `tar(1)` peuvent avoir certaines limitations pour des fichiers spéciaux. `cpio(1)` est plus souple.
- `cpio(1)` est conçu pour être utilisé avec `find(1)` et d'autres commandes et est adapté à la création de scripts de sauvegarde car la partie correspondant à la sélection de fichier du script peut être testée indépendamment.
- La structure interne des fichiers de données de Libreoffice est composée de fichiers « `.jar` » qui peut être ouvert également par `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 Outils de copie et de synchronisation

Voici une liste d'outils simples de copie et de sauvegarde sur le système Debian :

paquet	popcon	taille	outil	fonction
coreutils	V:901, I:999	17478	GNU cp	copier localement des fichiers et répertoires (« -a » pour récursif)
openssh-client	V:809, I:997	4298	scp	copier à distance des fichiers et des répertoires (client, « -r » pour récursif)
openssh-server	V:701, I:844	1567	sshd	copier à distance des fichiers et des répertoires (serveur distant)
rsync	V:311, I:564	677		synchronisation et sauvegarde distantes unidirectionnelles
unison	V:3, I:16	14		synchronisation et sauvegarde distantes bidirectionnelles

Table 10.2 – Liste des outils de copie et de synchronisation

Copier des fichiers avec `rsync(8)` offre des fonctionnalités plus riches que les autres méthodes.

- algorithme de transfert delta qui n'envoie que la différence entre les fichiers source et les fichiers existants sur la destination
- algorithme de vérification rapide (par défaut) recherchant les fichiers dont la taille ou l'heure de dernière modification a été modifiée
- Les options « `--exclude` » et « `--exclude-from` » sont semblables à celles de `tar(1)`
- La syntaxe « un slash en fin de répertoire source » qui évite la création d'un niveau de répertoire supplémentaire à la destination.

ASTUCE

Les outils de système de contrôle de version (VCS) de Tableau 10.11 peuvent fonctionner comme outils de synchronisation et de copie multi-voies.

10.1.3 Idiomes pour les archives

Voici quelques manières d'archiver et de désarchiver le contenu entier du répertoire « `./source` » en utilisant différents outils.

GNU tar(1):

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

ou encore, comme suit :

```
$ find ./source -xdev -print0 | tar -cvJf archive.tar.xz --null -F -
```

cpio(1):

```
$ find ./source -xdev -print0 | cpio -ov --null > archive.cpio; xz archive.cpio
$ zcat archive.cpio.xz | cpio -i
```

10.1.4 Idiomes pour la copie

Voici quelques manières d'archiver et de désarchiver le contenu entier du répertoire « `./source` » en utilisant différents outils.

— Copie locale du répertoire « `./source` » → répertoire « `/dest` »

— Faire le copie distante du répertoire « `./source` » de la machine locale → répertoire « `/dest` » situé sur la machine « `user@host.dom` »

rsync(8):

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

Vous pouvez, en remplacement, utiliser la syntaxe « un slash en fin du répertoire source ».

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

ou encore, comme suit :

```
# cd ./source; find . -print0 | rsync -aHAXSv0 --files-from=- . /dest
# cd ./source; find . -print0 | rsync -aHAXSv0 --files-from=- . user@host.dom:/dest
```

GNU cp(1) et 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
```

Vous pouvez remplacer « `.` » par « `truc` » dans tous les exemples comportant « `.` » pour copier les fichiers du répertoire « `./source/truc` » vers le répertoire « `/dest/truc` ».

Vous pouvez remplacer « `.` » par le chemin absolu « `/chemin/vers/source/truc` » dans tous les exemples comportant « `.` » pour éviter « `cd ./source;` ». Cela permet de copier les fichiers vers différents emplacements selon les outils utilisés de la manière suivante :

- « /dest/truc » : `rsync(8)`, `GNU cp(1)` et `scp(1)`
- « /dest/chemin/vers/source/truc » : `GNU tar(1)` et `cpio(1)`

ASTUCE

`rsync(8)` et `GNU cp(1)` possèdent l'option « `-u` » pour sauter les fichiers qui sont plus récents sur la destination.

10.1.5 Idioms pour la sélection de fichiers

`find(1)` est utilisé pour la sélection de fichiers pour les commandes d'archive et de copie (consultez Section 10.1.3 et Section 10.1.4) ou pour `xargs(1)` (consultez Section 9.4.9). Cela peut être amélioré en utilisant ces paramètres de commande.

La syntaxe de base de `find(1)` peut être résumée comme suit :

- Ses paramètres conditionnels sont évalués de gauche à droite.
- L'évaluation s'arrête lors que son résultat est déterminé.
- Le « **OU** logique » (indiqué par « `-o` » entre les éléments conditionnels) a une plus faible priorité que le « **ET** logique » (indiqué par « `-a` » ou rien entre éléments conditionnels).
- Le « **NON** logique » (indiqué par « `!` » avant un élément conditionnel) a une priorité plus élevée que le « **ET** logique ».
- « `-prune` » retourne toujours un **VRAI** logique et, si c'est un répertoire, la recherche de fichier est arrêtée au-delà de ce point.
- « `-name` » correspond à la base du nom de fichier avec les motifs génériques de l'interpréteur de commandes (voir Section 1.5.6) mais il correspond aussi à son « `.` » de début avec des métacaractères comme « `*` » et « `?` » (nouvelle fonctionnalité **POSIX**).
- « `-regex` » correspond au chemin complet par défaut dans le style **BRE** Emacs (consultez Section 1.6.2).
- « `-size` » correspond au fichier en se basant sur la taille du fichier (valeur précédée de « `+` » pour plus grand et précédée de « `-` » pour plus petit)
- « `-newer` » correspond au fichier plus récent que celui indiqué comme paramètre.
- « `-print0` » retourne toujours la valeur logique **VRAI** et affiche sur la sortie standard le nom de fichier en entier (terminé par le caractère null).

`find(1)` est souvent utilisé dans un style idiomatique comme ce qui suit :

```
# 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
```

Cela signifie que les actions suivantes doivent être effectuées :

1. rechercher tous les fichiers en partant de « /chemin/vers » ;
2. limiter globalement sa recherche à l'intérieur du système de fichiers et utiliser **ERE** (consultez Section 1.6.2) ;
3. exclure les fichiers correspondant à l'expression rationnelle de « `.*\.cpio` » ou « `.*~` » de la recherche en arrêtant le traitement ;
4. exclure les répertoires qui correspondent à l'expression rationnelle de « `.*\/\.git` » de la recherche en arrêtant le traitement ;
5. exclure les fichiers plus gros que 99 Megaoctets (unités de 1048576 octets) de la recherche en arrêtant le traitement ;
6. Afficher les noms de fichiers qui satisfont aux conditions de recherche ci-dessus et qui sont plus récents que « /chemin/vers/horodatage ».

Remarquez l'utilisation idiomatique de « `-prune -o` » pour exclure les fichiers dans l'exemple ci-dessus.

Note

Pour les systèmes **UNIX-like** autre que Debian, certaines options peuvent ne pas être prises en compte par `find(1)`. Dans un tel cas, essayez d'adapter la méthode de correspondance et remplacez « `-print0` » par « `-print` ». Vous devrez aussi ajuster les commandes associées.

10.1.6 Support d'archive

Lors du choix d'un [support d'enregistrement de données informatiques](#) destiné à l'archivage de données importantes, il faut faire attention à leurs limitations. Pour des petites sauvegardes de données personnelles, j'utilise des CD-R et des DVD-R provenant d'une grande marque et je les range dans un endroit frais, à l'ombre, sec et propre. (Les supports d'archive sur bande semblent être populaires pour les utilisations professionnelles).

Note

Un [coffre-fort anti-feu](#) est destiné aux documents sur papier. La plupart des supports de stockage de données informatiques ont une tolérance en température inférieure à celle du papier. J'utilise en général plusieurs copies chiffrées stockées dans différents endroits sûrs.

Durées de vie optimistes des moyens d'archivage trouvées sur le net (la plupart à partir d'informations des constructeurs).

- 100 ans et plus : papier non acide et encre
- 100 ans : stockage optique (CD/DVD, CD/DVD-R)
- 30 ans : supports magnétiques (bande, disquette)
- 20 ans : disque optique à changement de phase (CD-RW)

Cela ne prend pas en compte les défaillances mécaniques dues aux manipulations, etc.

Nombre de cycles d'écriture optimistes des moyens d'archivage trouvées sur le net (la plupart à partir d'informations des constructeurs).

- plus de 250 000 : disque dur
- plus de 10 000 cycles : mémoires Flash
- 1000 cycles : CD/DVD-RW
- 1 cycle : CD/DVD-R, papier

**Attention**

Ces chiffres de durée de vie et de nombre de cycles ne devront pas être utilisés pour des décisions concernant l'enregistrement de données critiques. Veuillez consulter les informations spécifiques au produit fournies par le constructeur.

ASTUCE

Comme les CD/DVD-R et le papier n'ont qu'un cycle d'écriture de 1, ils évitent de manière inhérente le risque de perte de données par écrasement. C'est un avantage !

ASTUCE

Si vous devez faire des sauvegardes fréquentes et rapides d'un gros volume de données, un disque dur sur une liaison réseau à haute vitesse peut être la seule option réaliste.

10.1.7 Périphériques d'enregistrement amovibles

Les périphériques d'enregistrement amovibles possibles sont les suivants.

- [Clé USB](#)
 - [Disque dur](#)
 - [Graveur de disque optique](#)
 - Appareil photographique numérique
-

— Lecteur de musique numérique

Ils peuvent être connectés à l'aide de n'importe quel moyen suivant.

— [USB](#)

— [FireWire](#)

— [PC-Card](#)

Les environnements de bureau modernes comme GNOME et KDE peuvent monter ces périphériques amovibles automatiquement sans entrée correspondante dans « `/etc/fstab` »

— Le paquet `udisks` fournit un démon et les utilitaires associés pour monter et démonter ces périphériques.

— [D-bus](#) crée les événements pour initialiser les processus automatiques.

— [PolicyKit](#) fournit les droits nécessaires.

ASTUCE

Les périphériques montés automatiquement pourraient avoir l'option de montage « `uhelper=` » qui est utilisée par `umount(8)`.

ASTUCE

Le montage automatique sous les environnements de bureau modernes ne se produit que lorsque ces périphériques amovibles ne se trouvent pas dans « `/etc/fstab` ».

Le point de montage sous les environnements de bureau modernes est choisi avec le nom « `/media/étiquette_disque` », il peut être personnalisé avec les outils suivants :

— `mlabel(1)` pour le système de fichiers FAT ;

— `genisoimage(1)` avec l'option « `-V` » pour le système de fichiers ISO9660 ;

— `tune2fs(1)` avec l'option « `-L` » pour le système de fichiers ext2, ext3 ou ext4.

ASTUCE

Le choix du codage doit être fourni comme option de montage (consultez Section [8.1.3](#)).

ASTUCE

L'utilisation d'une interface graphique pour démonter un système de fichiers peut supprimer son point de montage créé dynamiquement tel que « `/dev/sdc` ». Si vous souhaitez conserver son point de montage, démontez-le à l'aide de la commande `umount(8)` lancée depuis un interpréteur de commandes.

10.1.8 Choix de système de fichiers pour les données partagées

Lors du partage de données avec d'autres systèmes à l'aide de périphériques de stockage amovibles, vous devez les formater avec un [système de fichiers](#) pris en charge par les deux systèmes. Voici une liste de choix de systèmes de fichiers :

ASTUCE

Consultez Section [9.9.1](#) pour le partage de données entre plateformes en utilisant le chiffrement au niveau du périphérique.

Le système de fichiers FAT est pris en charge par la plupart des systèmes d'exploitation modernes et est assez utile pour l'échange des données par l'intermédiaire de supports du type disque dur amovible.

Pour le formatage de périphériques de type disque dur amovible pour l'échange de données entre plateformes avec un système de fichiers FAT, ce qui suit peut être un choix sûr :

filesystem name	typical usage scenario
FAT12	partage de données sur disquettes entre plateformes (<32Mio)
FAT16	partage entre plateformes de données sur des périphériques semblables à des disques durs de faible capacité (<2Gio)
FAT32	partage entre plateformes de données sur périphériques semblables à des disques durs de grande capacité (<8Tio, pris en charge par plus récent que 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	partage entre plateformes de données sur périphériques semblables à des disques durs de grande capacité (pris en charge de manière native par MS Windows NT et versions plus récentes; et pris en charge par NTFS-3G par l'intermédiaire de FUSE sous Linux)
ISO9660	partage entre plateformes de données sur CD-R et DVD+/-R
UDF	écriture incrémentale de CD-R et de DVD+/-R (nouveau)
MINIX	enregistrement efficace en matière d'espace disque de fichiers de données unix sur disquette
ext2	partage de données sur disque dur avec les anciens systèmes Linux
ext3	partage de données sur disque dur avec les anciens systèmes Linux
ext4	partage de données sur disque dur avec les systèmes Linux actuels
btrfs	sharing of data on the hard disk like device with current Linux systems with read-only snapshots

Table 10.3 – Liste de choix de systèmes de fichiers pour des périphériques amovibles avec des scénarios typiques d'utilisation

- les partitionner avec `fdisk(8)`, `cfdisk(8)` ou `parted(8)` (consultez Section [9.6.2](#)) en une seule partition primaire et la marquer comme suit :
 - type « 6 » pour FAT16 pour les supports faisant moins de 2Go.
 - type « c » pour FAT32 (LBA) pour les supports plus gros.
 - formater la partition primaire avec `mkfs.vfat(8)` comme suit :
 - simplement son nom de périphérique, par exemple « `/dev/sda1` », pour la FAT16
 - L'option explicite et le nom de périphérique, par exemple « `-F 32 /dev/sda1` », pour la FAT32
- Lors de l'utilisation des systèmes de fichiers FAT ou ISO9660 pour le partage de données ce qui suit sera une précaution sûre :
- Archiver d'abord les fichiers dans un fichier d'archive en utilisant `tar(1)` ou `cpio(1)` afin de conserver les noms de fichiers longs, les permissions de fichiers d'origine d'UNIX et les informations de propriétaire.
 - Découper le fichier d'archive en éléments de moins de 2 Gio à l'aide de la commande `split(1)` afin de le protéger contre les limitations de taille de fichier.
 - Chiffrer le fichier d'archive afin de sécuriser son contenu contre un accès non autorisé.

Note

La taille maximum d'un fichier FAT, par conception, est de $(2^{32} - 1)$ octets = (4GiB - 1 octet). Pour certaines applications sur le système 32 bits plus ancien, la taille maximum était même plus faible $(2^{31} - 1)$ octets = (2Gio - 1 octet). Debian ne souffre pas de ce dernier problème.

Note

Microsoft lui-même ne recommande pas l'utilisation de FAT pour des disques ou des partitions de plus de 200 Mo. Microsoft met en avant ces limitations comme une utilisation inefficace de l'espace disque dans ses « [Informations générales sur les systèmes de fichiers FAT, HPFS et NTFS](#) ». Bien sûr, on peut normalement utiliser le système de fichiers ext4 pour Linux.

ASTUCE

Pour davantage d'informations sur les systèmes de fichiers et les accès aux systèmes de fichiers, veuillez consulter « [Filesystems HOWTO](#) ».

10.1.9 Partage de données au travers du réseau

Lors du partage de données avec d'autres systèmes au travers du réseau, vous devrez utiliser un service commun. Voici quelques éléments :

service réseau	description d'un scénario typique d'utilisation
SMB/CIFS système de fichiers monté avec Samba	partage de fichiers par l'intermédiaire de « Microsoft Windows Network », consultez <code>smb.conf(5)</code> et le HOWTO et guide de référence officiel de Samba 3.x.x ou le paquet <code>samba-doc</code>
système de fichiers monté au travers du réseau NFS avec le noyau Linux	partager des fichiers par « UNIX/Linux Network », consultez <code>exports(5)</code> et Linux NFS-HOWTO
service HTTP	partager des fichiers entre client et serveur web
service HTTPS	partager des fichiers entre le client et le serveur web avec un chiffrement Secure Sockets Layer (SSL) ou Transport Layer Security (TLS)
service FTP	partager des fichiers entre serveur et client FTP

Table 10.4 – Liste des services réseau à choisir avec le scénario typique d'utilisation

Bien que ces systèmes de fichiers montés au travers du réseau et les méthodes de transfert au travers du réseau soient assez pratiques pour partager des données, elles peuvent être non sûres. Leur connexion réseau doit être sécurisée par ce qui suit :

- chiffrez-la avec [SSL/TLS](#)
- tunnelisez-la par [SSH](#)
- tunnelisez-la par [VPN](#)
- limitez-la derrière un pare-feu sûr

consultez aussi [Section 6.5](#) et [Section 6.6](#).

10.2 Sauvegarde et restauration

Nous savons tous que les ordinateurs sont parfois victimes de pannes ou que des erreurs humaines provoquent des dommages au système et aux données. Les opérations de sauvegarde et de restauration sont les parties essentielles d'une administration système réussie. Vous serez victime, un jour ou l'autre, de tous les modes de défaillance possibles.

ASTUCE

Mettez en place un système de sauvegardes simple et faites une sauvegarde fréquente de votre système. Avoir des données de sauvegarde est plus important que la qualité technique de votre méthode de sauvegarde.

Il y a 3 facteurs-clé qui permettent de définir une méthode pratique de sauvegarde et de restauration.

1. Ce qu'il faut sauvegarder et restaurer :
 - les fichiers que vous avez directement créés : données de « `~/` » ;
 - les fichiers de données créés par les applications que vous utilisez : données de « `/var/` » (sauf « `/var/cache/` », « `/var/run/` » et « `/var/tmp/` ») ;
 - les fichiers de configuration du système : fichiers de « `/etc/` » ;
 - logiciels locaux : données se trouvant dans « `/usr/local/` » ou « `/opt/` » ;
 - informations concernant l'installation du système : un mémo en texte concernant les étapes-clés ((partition, ...)) ;
 - un jeu de données testé : confirmé par des opérations de restauration expérimentales réalisées à l'avance.
2. Comment sauvegarder et restaurer :
 - entreposer les données de manière sûre : protection des données contre la réécriture et les défaillances du système ;

- sauvegardes fréquentes ! sauvegardes planifiées ;
- sauvegardes redondantes : duplication (miroir) des données ;
- processus indéréglable : sauvegarde facile en une seule commande.

3. Risques et coûts :

- valeur des données perdues ;
- ressources nécessaires pour effectuer les sauvegardes : humaines, matérielles, logicielles, ...
- mode de défaillance avec leur probabilité.

Note

Ne sauvegardez pas le contenu des pseudo systèmes de fichiers se trouvant dans `/proc`, `/sys`, `/tmp`, et `/run` (voir Section 1.2.12 et Section 1.2.13). À moins que vous ne sachiez exactement ce que vous faites, ce ne sont que d'énormes quantités de données inutiles.

Comme pour l'enregistrement sécurisé de données, les données doivent se trouver de préférences sur différentes partitions de disque ou sur des disques différents ou des machines différentes afin de résister à une corruption du système de fichiers. Les données importantes seront, de préférence écrites sur des supports à écriture unique tels que les CD/DVD-R afin de prévenir l'écrasement accidentel des données (consultez Section 9.8 pour la manière d'écrire sur le support d'enregistrement depuis la ligne de commandes de l'interpréteur. L'interface graphique de l'environnement de bureau GNOME vous donne un accès facile depuis le menu : « Places → CD/DVD Creator »).

Note

Il faudra peut-être arrêter certains démons d'applications comme le MTA (consultez Section 6.2.4) lors de la sauvegarde des données.

Note

Vous devez prendre des précautions supplémentaires lors de la sauvegarde et de la restauration des fichiers de données relatifs à l'identité comme « `/etc/ssh/ssh_host_dsa_key` », « `/etc/ssh/ssh_host_rsa_key` », « `~/.gnupg/*` », « `~/.ssh/*` », « `/etc/passwd` », « `/etc/shadow` », « `/etc/ftpmailrc` », « `popularity-contest.conf` », « `/etc/ppp/pap-secrets` » et « `/etc/exim4/passwd.client` ». Certaines de ces données ne peuvent pas être recrées en entrant la même chaîne de caractères d'entrée sur le système.

Note

Si vous faites tourner une tâche programmée (« cron » en tant que processus d'utilisateur, vous devrez restaurer les fichiers de « `/var/spool/cron/crontabs` » et redémarrer `cron(8)`. Consultez Section 9.4.14 pour `cron(8)` et `crontab(1)`.

10.2.1 Suites d'utilitaires de sauvegarde

Voici une liste d'utilitaires de sauvegarde notables disponibles sur le système Debian :

Les outils de sauvegarde ont chacun des objectifs particuliers.

- [Mondo Rescue](#) est un système de sauvegarde qui facilite la restauration rapide d'un système complet depuis de CD/DVD etc. sans passer par le processus normal d'installation d'un système.
 - [Bacula](#), [Amanda](#) et [BackupPC](#) sont des suites de sauvegardes ayant des fonctionnalités avancées qui sont orientées vers les sauvegardes fréquentes au travers du réseau.
 - Regular backups of user data can be realized by a simple script (Section 10.2.2).
-

paquet	popcon	taille	description
dump	V:1, I:6	352	BSD 4.4 dump(8) et restore(8) pour les systèmes de fichiers ext2/ext3/ext4 .
xfsdump	V:0, I:9	854	sauvegarder et restaurer avec xfsdump(8) et xfsrestore(8) pour le système de fichiers XFS sous GNU/Linux et IRIX
backupninja	V:3, I:4	367	système de sauvegarde meta-backup léger et extensible
bacula-common	V:10, I:14	2158	Bacula : sauvegarde, restauration et vérification par le réseau - fichiers communs
bacula-client	I:3	183	Bacula : sauvegarde, restauration et vérification par le réseau - métapaquet du client
bacula-console	V:1, I:4	107	Bacula : sauvegarde, restauration et vérification par le réseau - console en mode texte
bacula-server	I:1	183	Bacula : sauvegarde, restauration et vérification par le réseau - métapaquet du serveur
amanda-common	V:1, I:2	10030	Amanda : Advanced Maryland Automatic Network Disk Archiver (Libs). (Archiveur de disque par le réseau de Maryland avancé et automatique.
amanda-client	V:1, I:2	1088	Amanda : Advanced Maryland Automatic Network Disk Archiver (Client)
amanda-server	V:0, I:0	1075	Amanda : Advanced Maryland Automatic Network Disk Archiver (Serveur)
backup-manager	V:1, I:1	571	outil de sauvegarde en ligne de commandes
backup2l	V:0, I:1	114	outil de sauvegarde et restauration de faible maintenance pour des supports pouvant être montés (basé sur disque)
backuppc	V:3, I:3	3183	BackupPC est un système de hautes performances pour effectuer la sauvegarde de PC au niveau de l'entreprise (basé sur disques)
duplicity	V:9, I:19	1834	sauvegarde incrémentale (distante)
flexbackup	V:0, I:0	243	sauvegarde incrémentale (distante)
rdiff-backup	V:6, I:14	733	sauvegarde incrémentale (distante)
restic	V:1, I:4	22540	sauvegarde incrémentale (distante)
slbackup	V:0, I:0	151	sauvegarde incrémentale (distante)

Table 10.5 – Liste de suites d'utilitaires de sauvegarde

Les outils de base décrits dans Section 10.1.1 et Section 10.1.2 peuvent être utilisés pour faciliter la sauvegarde du système au moyen de scripts personnalisés. De tels scripts peuvent être améliorés comme suit :

- Le paquet `restic` permet les sauvegardes incrémentales (distantes) .
- le paquet `rdiff-backup` permet les sauvegardes incrémentales (distantes) ;
- le paquet `dump` facilite l’archivage et la restauration d’un système de fichiers entier de manière incrémentielle et efficace.

ASTUCE

Consultez les fichiers dans « `/usr/share/doc/dump/` » et « [« Est-ce que dump est vraiment obsolète ? »](#) » pour en savoir plus sur le paquet `dump`.

10.2.2 Personal backup

For a personal Debian desktop system running `testing` suite, I only need to protect personal and critical data. I reinstall system once a year anyway. Thus I see no reason to backup the whole system or to install a full featured backup utility.

At the same time, it is very valuable to have frequent recent snapshots of personal data and system configuration, and occasional full backups of personal data.

I usually make these snapshots and backups with a simple shell script [bss](#). This script is a short shell which uses standard utilities: `btrfs subvolume snapshot`, `rsync`, `cp`.

ASTUCE

Vous pouvez récupérer les données de configuration de `debconf` avec « `debconf-set-selections` `debconf-selections` » et les données de sélection de `dpkg` avec « `dpkg --set-selection <dpkg-selections.list` ».

10.3 Infrastructure de sécurité des données

L’infrastructure de sécurité des données est fournie par la combinaison d’un outil de chiffrement des données, d’un outil de condensé de messages et d’un outil de signature.

paquet	popcon	taille	commande	description
gnupg	V:498, I:942	864	<code>gpg(1)</code>	GNU Privacy Guard - outil de signature et de chiffrement OpenPGP
gpgv	V:890, I:999	882	<code>gpgv(1)</code>	GNU Privacy Guard - outil de vérification de signature
paperkey	V:1, I:13	58	<code>paperkey(1)</code>	extraire seulement l’information secrète de clés secrètes OpenPGP
cryptsetup	V:41, I:77	407	<code>cryptsetup(8)</code> ...	utilities for dm-crypt block device encryption supporting LUKS
coreutils	V:901, I:999	17478	<code>md5sum(1)</code>	calculer et vérifier un condensé MD5 de message
coreutils	V:901, I:999	17478	<code>sha1sum(1)</code>	calculer et vérifier un condensé SHA1 de message
openssl	V:814, I:994	1465	<code>openssl(1ssl)</code>	calculer un condensé de message avec « <code>openssl dgst</code> » (OpenSSL)

Table 10.6 – Liste des outils d’une infrastructure de sécurité des données

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

commande	description
gpg --gen-key	générer une nouvelle clé
gpg --gen-revoke ID_de_mon_utilisateur	générer une clé de révocation pour ID_de_mon_utilisateur
gpg --edit-key ID_utilisateur	éditer la clé de manière interactive, « help » pour obtenir de l'aide
gpg -o fichier --export	exporter toutes les clés vers fichier
gpg --import fichier	importer toutes les clés depuis fichier
gpg --send-keys ID_utilisateur	envoyer la clé de ID_utilisateur vers le serveur de clés
gpg --recv-keys ID_utilisateur	recevoir la clé de ID_utilisateur du serveur de clés
gpg --list-keys ID_utilisateur	afficher la liste des clés de ID_utilisateur
gpg --list-sigs ID_utilisateur	afficher la liste des signatures de ID_utilisateur
gpg --check-sigs ID_utilisateur	vérifier la signature de ID_utilisateur
gpg --fingerprint ID_utilisateur	vérifier l'empreinte de ID_utilisateur
gpg --refresh-keys	mettre à jour le porte-clé local

Table 10.7 – Liste des commandes de GNU Privacy Guard pour la gestion des clés

code	description de la confiance
-	pas de confiance d'utilisateur assignée/pas encore calculée
e	échec du calcul de la confiance
q	pas assez d'informations pour le calcul
n	ne jamais faire confiance à cette clé
m	confiance marginale
f	confiance complète
u	confiance ultime

Table 10.8 – Liste de la signification des codes de confiance

10.3.1 Gestion de clés pour GnuPG

Voici les commandes de [GNU Privacy Guard](#) pour la gestion de base des clés :

Voici la signification du code de confiance

Ce qui suit permet d'envoyer ma clé « 1DD8D791 » vers le serveur de clé populaire « `hkp://keys.gnupg.net` » :

```
$ gpg --keyserver hkp://keys.gnupg.net --send-keys 1DD8D791
```

Une bonne configuration de serveur de clés dans « `~/.gnupg/gpg.conf` » (ou à l'ancien emplacement « `~/.gnupg/options` ») contient ce qui suit :

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

Ce qui suit obtient les clés inconnues du serveur de clés :

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

Il y avait un bogue dans [OpenPGP Public Key Server](#) (pre version 0.9.6) qui corrompait les clés ayant plus de 2 sous-clés. Le paquet du serveur gnupg (>1.2.1-2) peut gérer ces sous-clés corrompues. Consultez `gpg(1)` sous l'option « `--repair-pks-subkey-bug` »

10.3.2 Utilisation de GnuPG sur des fichiers

Voici des exemples d'utilisation des commandes de [GNU Privacy Guard](#) sur des fichiers :

10.3.3 Utiliser GnuPG avec Mutt

Ajoutez ce qui suit à « `~/.muttrc` » afin d'éviter que GnuPG qui est lent ne démarre automatiquement, tout en permettant son utilisation en entrant « S » depuis l'index du menu :

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

10.3.4 Utiliser GnuPG avec Vim

The `gnupg` plugin let you run GnuPG transparently for files with extension « `.gpg` », « `.asc` », and « `.ppg` ». ¹

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

10.3.5 La somme de contrôle MD5

`md5sum(1)` fournit un utilitaire permettant de créer un fichier de résumé en utilisant la méthode se trouvant dans [rfc1321](#) et en l'utilisant pour vérifier chaque fichier qu'il contient.

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

1. If you use « `~/.vimrc` » instead of « `~/.vim/vimrc` », please substitute accordingly.

commande	description
<code>gpg -a -s fichier</code>	signer « fichier » dans un fichier ASCII blindé fichier.asc
<code>gpg --armor --sign fichier</code>	, ,
<code>gpg --clearsign fichier</code>	signer un fichier en clair
<code>gpg --clearsign fichier mail truc@example.org</code>	envoyer un message signé en clair à truc@example.org
<code>gpg --clearsign --not-dash-escaped fichier_rustine</code>	signer en clair un fichier_rustine
<code>gpg --verify fichier</code>	vérifier fichier signé en texte clair
<code>gpg -o fichier.sig -b fichier</code>	créer une signature détachée
<code>gpg -o fichier.sig --detach-sig fichier</code>	, ,
<code>gpg --verify fichier.sig fichier</code>	vérifier un fichier avec fichier.sig
<code>gpg -o fichier_chiffré.gpg -r nom -e fichier</code>	chiffrement par clé publique destiné au « nom » depuis le « fichier » vers « fichier_chiffré.gpg » binaire
<code>gpg -o fichier_chiffré.gpg --recipient nom --encrypt fichier</code>	, ,
<code>gpg -o fichier_chiffré.asc -a -r nom -e fichier</code>	chiffrement par clé publique destiné au « nom » depuis le « fichier » vers le fichier ASCII blindé « fichier_chiffré.asc »
<code>gpg -o fichier_chiffré.gpg -c fichier</code>	chiffrement symétrique depuis « fichier » vers « fichier_chiffré.gpg »
<code>gpg -o fichier_chiffré.gpg --symmetric fichier</code>	, ,
<code>gpg -o fichier_chiffré.asc -a -c fichier</code>	chiffrement symétrique prévu destiné au « nom » depuis le « fichier » vers le fichier ASCII blindé « fichier_chiffré.asc »
<code>gpg -o fichier -d fichier_crypt.gpg -r nom</code>	déchiffrement
<code>gpg -o fichier --decrypt fichier_chiffré.gpg</code>	, ,

Table 10.9 – Liste des commandes de GNU Privacy Guard sur des fichiers

Note

Le calcul de somme de contrôle [MD5](#) consomme moins de ressources processeur que celles utilisées pour le chiffrement des signatures en utilisant [GNU Privacy Guard \(GnuPG\)](#). Habituellement, pour s'assurer de l'intégrité des données, seul le résumé de plus haut niveau est signé par chiffrement.

10.4 Outils pour fusionner le code source

Il existe de nombreux outils pour fusionner du code source. Les commandes qui suivent ont attiré mon attention :

paquet	popcon	taille	commande	description
patch	V:123, I:721	248	<code>patch(1)</code>	appliquer un fichier de différences (« diff ») à un original
vim	V:102, I:404	3286	<code>vimdiff(1)</code>	comparer deux fichiers côte à côte dans vim
imediff	V:0, I:0	170	<code>imediff(1)</code>	interactive full screen 2/3-way merge tool
meld	V:14, I:38	3065	<code>meld(1)</code>	comparer et fusionner des fichiers (GTK)
wiggle	V:0, I:0	174	<code>wiggle(1)</code>	appliquer les rustines rejetées
diffutils	V:883, I:993	1598	<code>diff(1)</code>	comparer des fichiers ligne à ligne
diffutils	V:883, I:993	1598	<code>diff3(1)</code>	comparer et fusionner trois fichiers ligne par ligne
quilt	V:3, I:32	788	<code>quilt(1)</code>	gérer une série de rustines
wdiff	V:8, I:69	644	<code>wdiff(1)</code>	afficher les différences de mots entre deux fichiers texte
diffstat	V:14, I:146	81	<code>diffstat(1)</code>	afficher un histogramme des modifications apportées par le fichier de différences
patchutils	V:16, I:143	232	<code>combinediff(1)</code>	créer une rustine (patch) cumulative à partir de deux rustines incrémentales
patchutils	V:16, I:143	232	<code>dehtmldiff(1)</code>	extraire un fichier de différences d'une page HTML
patchutils	V:16, I:143	232	<code>filterdiff(1)</code>	extraire ou exclure des différences d'un fichier de différences
patchutils	V:16, I:143	232	<code>fixcvsdiff(1)</code>	corriger les fichiers de différences créés par CVS que <code>patch(1)</code> interprète mal
patchutils	V:16, I:143	232	<code>flipdiff(1)</code>	échanger l'ordre de deux rustines
patchutils	V:16, I:143	232	<code>grepdiff(1)</code>	afficher quels sont les fichiers modifiés par une rustine correspondant une expression rationnelle
patchutils	V:16, I:143	232	<code>interdiff(1)</code>	afficher les différences entre deux fichiers de différence unifiés
patchutils	V:16, I:143	232	<code>lsdiff(1)</code>	afficher quels sont les fichiers modifiés par une rustine
patchutils	V:16, I:143	232	<code>recountdiff(1)</code>	recalculer les nombres et les décalages dans un contexte unifié de fichiers de différences
patchutils	V:16, I:143	232	<code>rediff(1)</code>	corriger les décalages et les nombres d'un fichier de différences édité manuellement
patchutils	V:16, I:143	232	<code>splitdiff(1)</code>	séparer les rustines incrémentales
patchutils	V:16, I:143	232	<code>unwrapdiff(1)</code>	réparer les correctifs dont les mots ont été coupés
dirdiff	V:0, I:2	166	<code>dirdiff(1)</code>	afficher les différences et fusionner les modifications entre deux arbres de répertoires
docdiff	V:0, I:0	555	<code>docdiff(1)</code>	comparer deux fichiers mot par mot ou caractère par caractère
makepatch	V:0, I:0	102	<code>makepatch(1)</code>	créer des fichiers de rustines étendus
makepatch	V:0, I:0	102	<code>applypatch(1)</code>	appliquer des fichiers de rustines étendus

Table 10.10 – Liste d'outils destinés à fusionner du code source

10.4.1 Extraire des différences pour des fichiers sources

La procédure suivante vous permet d'extraire les différences entre deux fichiers sources et créer un fichier de différences unifié « `fichier.patch0` » ou « `fichier.patch1` » selon l'emplacement du fichier.

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

10.4.2 Fusionner les mises à jour des fichiers source

Le fichier de différences (« `diff` » (encore appelé fichier « `patch` » ou rustine) est utilisé pour envoyer une mise à jour de programme. Celui qui reçoit applique cette mise à jour à un autre fichier de la manière suivante :

```
$ 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 `imdiff(1)` by the following.

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

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

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

10.5 Systèmes de contrôle de version

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

Git is the tool of choice these days 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 is a summary of the notable VCS on the Debian system.

10.5.1 Configuration du client Git

Vous pourrez définir certains éléments de configuration globaux, comme votre nom et votre adresse de courriel utilisée par Git, dans « `~/.gitconfig` » de la manière suivante :

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

Si vous avez l'habitude d'utiliser les commandes de CVS ou de Subversion, vous pourrez définir certains alias de commandes comme suit :

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

Vous pouvez vérifier votre configuration globale de la manière suivante :

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

paquet	popcon	taille	outil	type du VCS	commentaire
git	V:312, I:494	36515	Git	distribué	DVCS rapide en C (utilisé par le noyau de Linux et d'autres)
gitk	V:5, I:41	1749	Git	distribué	Interface graphique d'affichage d'une arborescence de dépôt VCS (Git)
mercurial	V:6, I:44	1053	Mercurial	distribué	DVCS en Python avec un peu de C
darcs	V:0, I:7	23159	Darcs	distribué	DVCS avec une algèbre intelligente des rustines (lent)
bazaar	V:1, I:15	28	Bazaar	distribué	DVCS influenced by t la written in Python (historic)
tla	V:0, I:2	1011	GNU arch	distribué	DVCS mainly by Tom Lord (historic)
subversion	V:18, I:104	4873	Subversion	distant	"CVS done right", newer standard remote VCS (historic)
cvs	V:4, I:38	4620	CVS	distant	previous standard remote VCS (historic)
tkcvs	V:0, I:1	1498	CVS, ...	distant	Interface graphique d'affichage d'une arborescence de dépôt VCS (CVS, Subversion, RCS)
rcs	V:3, I:19	562	RCS	local	" Unix SCCS done right" (historic)
cssc	V:0, I:2	2044	CSSC	local	clone of the Unix SCCS (historic)

Table 10.11 – Liste d'outils pour les systèmes de contrôle de version

Git command	fonction
<code>git init</code>	créer le dépôt (local)
<code>git clone</code>	clone the remote repository as the working tree
<code>git pull</code>	mettre à jour l'arborescence de travail depuis le dépôt distant
<code>git add .</code>	ajouter des fichiers de l'arborescence de travail vers le VCS
<code>git rm</code>	supprimer du VCS des fichiers de l'arborescence de travail
<code>git commit -a</code>	déposer des modifications sur le dépôt local
<code>git push</code>	mettre à jour le dépôt distant à l'aide du dépôt local
<code>git status</code>	afficher l'état de l'arborescence de travail d'après le VCS
<code>git diff</code>	<i>diff dépôt_de_référence arborescence_de_travail</i>
<code>git repack -a -d; git prune</code>	réempaqueter le dépôt local en un seul paquet
<code>gitk</code>	Interface graphique d'affichage de l'arborescence d'un dépôt VCS

Table 10.12 – Comparaison des commandes natives de VCS

paquet	popcon	taille	commande	description
git-doc	I:14	12063	N/A	Documentation officielle de git
gitmagic	I:1	721	N/A	« Git Magic », le guide le plus facile à comprendre pour Git
git	V:312, I:494	36515	git(7)	Git, système de contrôle de version rapide, évolutif et distribué
gitk	V:5, I:41	1749	gitk(1)	Interface graphique de navigateur de dépôt Git avec historique
git-gui	V:1, I:23	2343	git-gui(1)	Interface graphique pour Git (pas d'historique)
git-email	V:0, I:11	993	git-send-email(1)	envoyer une série de rustines sous forme de courriel à partir de Git
git-buildpackage	V:1, I:11	4187	git-buildpackage(1)	la mise en paquet Debian avec Git
dgit	V:0, I:1	496	dgit(1)	git interoperability with the Debian archive
imediff	V:0, I:0	170	git-ime(1)	interactive git commit split helper tool
stgit	V:0, I:0	603	stg(1)	quilt par-dessus git (Python)

Table 10.13 – Liste des paquets et des commandes relatifs à git

10.5.2 Commandes de Git

Here is an overview of main Git commands.



Attention

L'appel d'une sous-commande git directement avec « git-xyz » depuis la ligne de commandes est devenu obsolète depuis début 2006.

ASTUCE

With [git\(1\)](#), you can work on a local topic branch with many commits with "git rebase -i master" to reorder/drop/squish commits and with occasional "git ime" to split the squashed commit into multiple commits. Then you can push commits with clean change history to the remote repository. See [git-rebase\(1\)](#) and [git-ime\(1\)](#).

ASTUCE

When you want to go back to the clean working directory without losing the current state of the working directory, you can use "git stash". See [git-stash\(1\)](#).

ASTUCE

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.

ASTUCE

GUI tools such as [gitk\(1\)](#) really help you with tracking revision history of files and operating on the repository such as changing between checked out branches and cherry-picking patches. The web interface provided by many public archives for browsing their repositories is also quite useful, too.

10.5.3 Git pour l'enregistrement de l'historique de la configuration

Vous pouvez enregistrer vous-même un historique chronologique de la configuration en utilisant les outils [Git](#). Voici un exemple simple, pour vous exercer, d'enregistrement du contenu de « `/etc/apt/` ».

```
$ cd /etc/apt/  
$ sudo git init  
$ sudo chmod 700 .git  
$ sudo git add .  
$ sudo git commit -a
```

Dépôt (« commit ») d'une configuration avec sa description

Effectuez les modifications dans les fichiers de configuration.

```
$ cd /etc/apt/  
$ sudo git commit -a
```

Déposez (« commit ») la configuration avec sa description et retournez à vos occupations habituelles.

```
$ cd /etc/apt/  
$ sudo gitk --all
```

Vous avez avec vous l'historique complet de la configuration.

Note

`sudo(8)` est nécessaire pour travailler avec n'importe quelles permissions sur les données de configuration. Pour les données de configuration de l'utilisateur, `sudo` peut être omis.

Note

La commande « `chmod 700 .git` » de l'exemple ci-dessus est nécessaire pour protéger les données de l'archive d'un accès en écriture non autorisé.

ASTUCE

Pour une configuration plus complète de l'enregistrement de l'historique de configuration, voyez le paquet `etckeeper` package : Section [9.3.9](#).

10.5.4 Références de Git

Consultez ce qui suit.

- [page de manuel : git\(1\)](#) (`/usr/share/doc/git-doc/git.html`)
- [Manuel de l'utilisateur de Git](#) (`/usr/share/doc/git-doc/user-manual.html`)
- [Un tutoriel d'introduction à git](#) « A tutorial introduction to git » (`/usr/share/doc/git-doc/gittutorial.html`)
- [Un tutoriel d'introduction à git : deuxième partie](#) « A tutorial introduction to git: part two » (`/usr/share/doc/git-doc/gittutorial2.html`)
- [Everyday GIT With 20 Commands Or So](#) (`/usr/share/doc/git-doc/giteveryday.html`)
- [La magie de Git](#) « Git Magic » (`/usr/share/doc/gitmagic/html/index.html`)

Les commandes `git-gui(1)` et `gitk(1)` rendent très facile l'utilisation de Git.

**AVERTISSEMENT**

Ne pas utiliser d'espaces dans la chaîne de balise même si certains outils comme `gitk(1)` vous permettent de le faire. Cela peut perturber d'autres commandes de `git`.

Chapitre 11

Conversion de données

Description des outils et astuces pour convertir différents formats de données sur un système Debian.

Les outils basés sur des standards sont de très bonne qualité mais la prise en charge des formats propriétaires est limitée.

11.1 Outils de conversion de données textuelles

Mes yeux ont été attirés par les paquets suivants de conversions de données textuelles :

paquet	popcon	taille	mot clé	description
libc6	V:946, I:999	12772	jeu de caractères	convertisseur de codage de texte entre différents paramètres linguistiques à l'aide d' <code>iconv(1)</code> (fondamental)
recode	V:3, I:25	603	charset+eol	convertisseur de codage de texte entre différents paramètres linguistiques (flexible avec plus d'alias et de fonctionnalités)
konwert	V:1, I:52	134	jeu de caractères	convertisseur de codage de texte entre différents paramètres linguistiques (le luxe)
nkf	V:0, I:11	358	jeu de caractères	jeux de caractères pour le japonais
tcs	V:0, I:0	518	jeu de caractères	jeu de caractères du traducteur
unaccent	V:0, I:0	29	jeu de caractères	remplacer les lettres accentuées par leur équivalent accentué
tofrodos	V:1, I:23	55	eol (fin de ligne)	convertisseur de format de texte entre DOS et UNIX : <code>fromdos(1)</code> et <code>todos(1)</code>
macutils	V:0, I:0	298	eol (fin de ligne)	convertisseur de format de texte entre Macintosh et UNIX : <code>frommac(1)</code> et <code>tomac(1)</code>

Table 11.1 – Liste des outils de conversion de texte

11.1.1 Convertir un fichier texte avec `iconv`

ASTUCE

`iconv(1)` fait partie du paquet `libc6` et est toujours disponible sur pratiquement tous les systèmes de type Unix pour permettre la conversion des codages de caractères.

Vous pouvez convertir les codages de caractères d'un fichier texte par `iconv(1)` en effectuant ce qui suit :

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

La valeur des codages n'est pas sensible à la casse et ignore « - » et « _ » pour la correspondance. On peut vérifier quels sont les codages pris en charge à l'aide de la commande « `iconv -l` ».

valeur de codage	utilisation
ASCII	American Standard Code for Information Interchange (Code américain standard pour l'échange d'informations), code sur 7 bits sans caractère accentué
UTF-8	norme actuelle multi-langues pour tous les systèmes d'exploitation modernes
ISO-8859-1	ancienne norme pour les langues d'Europe de l'ouest, ASCII + lettres accentuées
ISO-8859-2	ancienne norme pour les langues d'Europe de l'est, ASCII + lettres accentuées
ISO-8859-15	ancienne norme pour les langues d'Europe de l'ouest, ISO-8859-1 avec le signe euro
CP850	page de code 850, caractères de DOS Microsoft avec graphiques pour les langues de l'Europe de l'est, variante de ISO-8859-1
CP932	page de code 932, variante style Microsoft Windows de Shift-JIS pour le japonais
CP936	page de code 936, variante style Microsoft Windows de GB2312 , GBK ou GB18030 pour le chinois simplifié
CP949	page de code 949, variante style Microsoft Windows de EUC-KR de « Unified Hangul Code » pour le coréen
CP950	page de code 950, variante style Microsoft Windows de Big5 pour le chinois traditionnel
CP1251	page de code 1251, encodage de style Microsoft Windows pour l'alphabet cyrillique
CP1252	page de code 1252, variante style Microsoft Windows de ISO-8859-15 pour les langues d'Europe de l'ouest
KOI8-R	ancienne norme UNIX de russe pour l'alphabet cyrillique
ISO-2022-JP	codage standard du courrier électronique japonais n'utilisant que des codes à 7 bits
eucJP	ancien code UNIX de japonais sur 8 bits et complètement différent de Shift-JIS
Shift-JIS	norme JIS X 0208 Annexe 1 pour le japonais (consultez CP932)

Table 11.2 – Liste de valeurs de codage et leur utilisation

Note

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

Pour les jeux de caractères qui tiennent dans un seul octet tels que les jeux de caractères [ASCII](#) et [ISO-8859](#), le [codage des caractères](#) signifie à peu près la même chose que le jeu de caractères.

Pour les jeux de caractères ayant de nombreux caractères tels que [JIS X 0213](#) pour le japonais ou [Universal Character Set \(UCS, Unicode, ISO-10646-1\)](#) (jeu de caractère universel) pour pratiquement toutes les langues, il y a de nombreux schémas de codage pour les insérer dans les séquences d'octets de données.

- [EUC](#) et [ISO/IEC 2022](#) (connu aussi en tant que [JIS X 0202](#)) pour le japonais
- [UTF-8](#), [UTF-16/UCS-2](#) et [UTF-32/UCS-4](#) pour l'Unicode

Pour ceux-ci, il y a une différence claire entre le jeu de caractères et le codage des caractères.

[Page de code](#) est utilisée comme synonyme de table de codage de caractères pour certaines d'entre-elles spécifiques au fournisseur.

Note

Veillez remarquer que la plupart des systèmes de codage partagent le même code avec ASCII pour les caractères sur 7 bits. Mais il y a quelques exceptions, lors de la conversion de programmes C et des données d'URL anciens en japonais depuis ce qui est parfois appelé format de codage shift-JIS vers le format UTF-8, utilisez « CP932 » comme nom de codage plutôt que « shift-JIS » afin d'obtenir le résultat attendu : 0x5C → « \ » et 0x7E → « ~ ». Sinon, ils seront convertis vers les mauvais caractères.

ASTUCE

On peut aussi utiliser `recode(1)` qui offre plus de fonctionnalités que celles combinées de `iconv(1)`, `fromdos(1)`, `todos(1)`, `frommac(1)` et `tomac(1)`. Pour plus de détails, consultez « `info recode` ».

11.1.2 Vérifier que les fichiers sont codés en UTF-8 avec `iconv`

Vous pouvez vérifier si un fichier texte est codé en UTF-8 à l'aide d'`iconv(1)` en effectuant ce qui suit :

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

ASTUCE

Utilisez l'option « `--verbose` » dans les exemples ci-dessus pour trouver le premier caractère non UTF-8.

11.1.3 Convertir les noms de fichiers avec `iconv`

Voici un exemple de script pour convertir le codage des noms de fichiers, dans un seul répertoire, depuis celui créé par un ancien système d'exploitation vers celui d'un système d'exploitation moderne en UTF-8.

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

La variable « `$ENCDN` » spécifie l'encodage d'origine des noms de fichier sous d'anciens OS de la même façon que dans [Tableau 11.2](#).

Pour les cas plus compliqués, veuillez monter le système de fichiers (par exemple une partition d'un disque dur) contenant de tels noms de fichiers avec le codage correct comme option de `mount(8)` (consultez [Section 8.1.3](#)) et copier son contenu complet vers un autre système de fichiers monté en UTF-8 avec la commande « `cp -a` ».

11.1.4 Convertir les fins de ligne (EOL)

Le format de fichier texte, particulièrement le code de fin de ligne (EOL), dépend de la plateforme.

Les programmes de conversion du format des fins de lignes (EOL), `fromdos(1)`, `todos(1)`, `frommac(1)` et `tomac(1)` sont assez pratiques. `recode(1)` peut aussi être utile.

plateforme	code pour EOL	contrôle	décimal	hexadécimal
Debian (unix)	LF	^J	10	0A
MSDOS et Windows	CR-LF	^M^J	13 10	0D 0A
Apple Macintosh	CR	^M	13	0D

Table 11.3 – Liste des styles d’EOL pour différentes plateformes

Note

Certaines données sur un système Debian, telles que les données de la page wiki du paquet `python-moinmoin`, utilisent le CR-LF du style MSDOS comme code de fin de ligne (EOL). La règle précédente n'est donc que générale.

Note

La plupart des éditeurs (par exemple `vim`, `emacs`, `gedit`...) peuvent prendre en compte de manière transparente les fichiers ayant une fin de ligne (EOL) de style MSDOS.

ASTUCE

L'utilisation de « `sed -e '/\r$/!s$/\r/'` » en remplacement de `todos(1)` est préférable lorsque vous désirez unifier le style de caractère de fin de ligne vers le style MSDOS depuis un style mixte MSDOS et UNIX (par exemple, après avoir fusionné deux fichiers de style MSDOS avec `diff3(1)`). Cela parce que `todos` ajoute un retour charriot (CR) à toutes les lignes.

11.1.5 Convertir les tabulations (TAB)

Il y a quelques programmes spécialisés dans la conversion des codes de tabulations.

fonction	<code>bsdmainutils</code>	<code>coreutils</code>
étendre les tabulations en espaces	« <code>col -x</code> »	<code>expand</code>
convertir les espaces en tabulation	« <code>col -h</code> »	<code>unexpand</code>

Table 11.4 – Liste des commande de conversion de TAB des paquets `bsdmainutils` et `coreutils`

`indent(1)` du paquet `indent` reformate complètement les espaces dans un programme en C.

Des programmes d'édition tels que `vim` et `emacs` peuvent aussi être utilisés pour la conversion des tabulations. Par exemple avec `vim`, vous pouvez étendre les tabulation avec la séquence de commandes « `:set expandtab` » et « `:%retab` ». Vous pouvez revenir en arrière de cette conversion par la séquence de commandes « `:set noexpandtab` » et « `:%retab!` ».

11.1.6 Éditeurs avec conversion automatique

Les éditeurs de textes modernes et intelligents comme le programme `vim` sont assez habiles et prennent assez bien en compte les systèmes de codage de caractères et tous les formats de fichiers. Vous devriez utiliser ces éditeurs avec les paramètres linguistiques UTF-8 dans une console compatible avec UTF-8 pour une meilleure compatibilité.

Un ancien fichier texte UNIX d'Europe de l'Ouest, « `u-file.txt` » enregistré dans le codage latin1 (iso-8859-1) peut être édité avec `vim` de la manière suivante :

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

C'est possible car le mécanisme d'autodétection du codage du fichier dans `vim` suppose d'abord que le codage est UTF-8 et, s'il échoue, suppose qu'il est latin1.

Un ancien fichier texte en polonais, « `pu-file.txt` », enregistré avec le codage latin2 (iso-8859-2) peut être édité avec `vim` de la manière suivante :

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

Un ancien fichier texte UNIX en japonais, « `ju-file.txt` », enregistré avec le codage eucJP peut être édité avec `vim` de la manière suivante :

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

Un ancien fichier MS-Windows en japonais, « `jw-file.txt` », enregistré dans le codage appelé shift-JIS (plus précisément : CP932) peut être édité avec `vim` de la manière suivante :

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

Lorsqu'un fichier est ouvert avec les options « `++enc` » et « `++ff` », « `:w` » sur la ligne de commandes de Vim l'enregistre dans son format d'origine et écrase le fichier d'origine. Vous pouvez aussi indiquer le format d'enregistrement et le nom du fichier sur la ligne de commandes de Vim, par exemple, « `:w ++enc=utf8 nouveau.txt` ».

Veuillez vous rapporter à `mbyte.txt` « multi-byte text support » dans l'aide en ligne de `vim` et Tableau 11.2 pour les valeurs de paramètres linguistiques utilisés avec « `++enc` ».

La famille de programmes `emacs` peut effectuer des fonctions équivalentes.

11.1.7 Extraire du texte brut

Ce qui suit permet de lire une page web sous forme de fichier texte. C'est très utile pour copier des informations de configuration depuis le Web ou pour appliquer des outils textuels de base d'UNIX comme `grep(1)` à la page web .

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

De la même manière, vous pouvez extraire des données en texte brut vers d'autres formats en utilisant ce qui suit :

paquet	popcon	taille	mot clé	fonction
w3m	V:25, I:252	2367	html → texte	convertisseur HTML vers texte avec la commande « <code>w3m -dump</code> »
html2text	V:2, I:29	274	html → texte	convertisseur avancé HTML vers texte (ISO 8859-1)
lynx	V:11, I:97	1948	html → texte	convertisseur HTML vers texte avec la commande « <code>lynx -dump</code> »
elinks	V:5, I:27	1767	html → texte	convertisseur HTML vers texte avec la commande « <code>elinks -dump</code> »
links	V:5, I:38	2250	html → texte	convertisseur HTML vers texte avec la commande « <code>links -dump</code> »
links2	V:1, I:14	5418	html → texte	convertisseur HTML vers texte avec la commande « <code>links2 -dump</code> »
catdoc	V:22, I:125	675	MSWord → texte	convertir les fichier MSWord en texte brut ou en TeX
antiword	V:1, I:10	589	MSWord → texte	convertir des fichiers MSWord en texte brut ou en ps
pstotext	V:0, I:2	126	ps/pdf → texte	extraire du texte depuis des fichiers PostScript et PDF
unhtml	V:0, I:0	43	html → texte	supprimer les balise d'un fichier HTML
odt2txt	V:1, I:7	60	odt → texte	convertisseur du texte OpenDocument vers texte

Table 11.5 – Liste d'outils pour extraire des données en texte brut

paquet	popcon	taille	mot clé	description
vim-runtime	V:18, I:439	32134	mise en évidence	MACRO Vim pour convertir du code source en HTML avec « :source \$VIMRUNTIME/syntax/html.vim »
cxref	V:0, I:0	1193	c → html	convertisseur pour les programmes C vers latex et HTML (langage C)
src2tex	V:0, I:0	622	mise en évidence	convertit de nombreux codes sources en TeX (langage C)
source-highlight	V:0, I:7	1992	mise en évidence	convertit de nombreux codes source vers des fichiers HTML, XHTML, LaTeX, Texinfo, séquences d'échappement en couleur ANSI et DocBook files avec mise en évidence (C++)
highlight	V:1, I:10	1083	mise en évidence	convertit de nombreux codes sources en fichiers HTML, XHTML, RTF, LaTeX, TeX ou XSL-FO avec mise en évidence (C++)
grc	V:0, I:3	208	texte → couleur	coloriseur générique pour n'importe quoi (Python)
python-docutils	V:9, I:112	1771	texte → tout format	formateur de documents ReStructured Text vers XML (Python)
pandoc	V:8, I:48	151714	texte → tout format	Marqueur de conversion global (Haskell)
markdown	V:0, I:8	57	texte → html	formateur de documents textes Markdown vers (X)HTML (Perl)
txt2html	V:0, I:3	259	texte → html	convertisseur de texte vers HTML (Perl)
asciidoc	I:12	81	texte → tout format	formateur de documents textes AsciiDoc vers XML/HTML (Python)

Table 11.6 – Liste des outils pour mettre en évidence des données de texte brut

11.1.8 Mettre en évidence et formater des données en texte brut

Vous pouvez mettre en évidence et formater des données en texte brut de la manière suivante :

11.2 Données XML

Le langage de balisage extensible (« The Extensible Markup Language (XML) ») est un langage de balisage de documents dont les informations sont structurées.

Consultez une introduction sur [XML.COM](#).

- « Qu'est-ce qu'XML ? »
- « Qu'est-ce qu'XSLT ? »
- « Qu'est-ce qu'XSL-FO ? »
- « Qu'est-ce qu'XLink ? »

11.2.1 Conseils de base pour XML

Le texte en XML ressemble un peu à [HTML](#). Il vous permet de gérer de nombreux formats de sortie pour un document. Un système XML facile est le paquet `docbook-xsl` qui est utilisé ici.

Chaque fichier XML commence par la déclaration XML standard suivante :

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

La syntaxe de base d'un élément XML est balisée de la manière suivante :

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

Un élément XML dont le contenu est vide est balisé de la façon raccourcie suivante :

```
name attribute="value"/
```

« `attribute="value"` » dans les exemples ci-dessus est optionnel.

L'action commentaire en XML est balisée comme suit :

```
!-- comment --
```

En plus d'ajouter des balises, XML demande des conversions mineures de contenu en utilisant des entités prédéfinies pour les caractères suivants :

entité prédéfinie	caractère devant être converti
"	" : double apostrophe
'	' : apostrophe
<	< : inférieur à
>	> : supérieur à
&	& : esperluette

Table 11.7 – Liste des entités XML prédéfinies



Attention

« < » ou « & » ne peuvent pas être utilisés dans des attributs ni des éléments.

Note

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"`".

Note

Tant que le balisage XML est fait de manière cohérente avec un jeu particulier de nom de balises (soit certaines données comme valeur de contenu ou d'attribut), la conversion vers un autre XML est une tâche triviale en utilisant [Extensible Stylesheet Language Transformations \(XSLT\)](#).

11.2.2 Traitement XML

Il existe de nombreux outils pour traiter les fichiers XML tels que [le langage extensible de feuilles de style](#) (« [the Extensible Stylesheet Language XSL](#) »).

En gros, une fois créé un fichier XML correctement formaté, vous pouvez le convertir vers n'importe quel format en utilisant le [langage extensible de transformation des feuilles de style \(XSLT\)](#) (« [Extensible Stylesheet Language Transformations](#) »).

Le [Langage extensible de feuilles de style pour le formatage des objets \(XSL-FO\)](#) (« [Extensible Stylesheet Language for Formatting Objects](#) ») est censé être la solution au formatage. Le paquet `fop` est nouveau dans l'archive `main` de Debian du fait de ses dépendances vers le [langage de programmation Java](#). Le code LaTeX est donc habituellement créé depuis XML en utilisant XSLT, et le système LaTeX est utilisé pour créer des fichiers imprimables comme DVI, PostScript et PDF.

Comme XML est un sous-ensemble du [Langage généralisé de balisage \(Standard Generalized Markup Language SGML\)](#) », il peut être traité par les nombreux outils disponibles pour SGML, comme [Document Style Semantics and Specification Language \(DSSSL\)](#).

paquet	popcon	taille	mot clé	description
docbook-xml	V:14, I:264	2133	xml	Définition de type de document XML (DTD) pour DocBook
docbook-xsl	V:12, I:154	14870	xml/xslt	feuilles de style XSL pour le traitement de DocBook XML vers divers formats de sortie avec XSLT
xsltproc	V:14, I:100	160	xslt	processeur en ligne de commandes XSLT (XML → XML, HTML, texte brut, etc.)
xmlto	V:1, I:22	130	xml/xslt	convertisseur XML-vers-tout avec XSLT
fop	V:1, I:20	291	xml/xsl-fo	convertir les fichiers Docbook XML en PDF
dblatex	V:2, I:15	4643	xml/xslt	convertir les fichiers Docbook en documents DVI, PostScript, PDF avec XSLT
dbtoepub	V:0, I:0	37	xml/xslt	Convertisseur Docbook XML vers .epub

Table 11.8 – Liste d’outils XML

paquet	popcon	taille	mot clé	description
openjade	V:2, I:36	1019	dsssl	processeur DSSSL à la norme ISO/IEC 10179:1996 DSSSL (le plus récent)
docbook-dsssl	V:1, I:21	2604	xml/dsssl	feuilles de style DSSSL pour le traitement des DocBook XML vers divers formats de sortie avec DSSSL
docbook-utils	V:0, I:15	281	xml/dsssl	utilitaires pour les fichiers DocBook y compris la conversion avec DSSSL vers d’autres formats (HTML, RTF, PS, man, PDF) avec des commandes docbook2*
sgml2x	V:0, I:0	90	SGML/dsssl	convertisseur depuis SGML et XML utilisant les feuilles de style DSSSL

Table 11.9 – Liste des outils DSSSL

ASTUCE

Le `lp` de [GNOME](#) est parfois pratique pour lire les fichiers XML [DocBook](#) directement car il effectue un rendu propre sous X.

11.2.3 Extraire des données XML

Vous pouvez extraire des données HTML ou XML depuis d'autres formats en utilisant ce qui suit :

paquet	popcon	taille	mot clé	description
man2html	V:0, I:2	138	manpage → html	convertisseur de page de manuel en HTML (prise en charge de CGI)
doclifter	V:0, I:0	451	troff → xml	convertisseur de troff vers DocBook XML
texi2html	V:0, I:7	1833	texi → html	convertisseur de Texinfo vers HTML
info2www	V:1, I:3	76	info → html	convertisseur depuis GNU info vers HTML (prise en charge de CGI)
wv	V:0, I:6	717	MSWord → n'importe quoi	convertisseur de document depuis Microsoft Word vers HTML, LaTeX, etc.
unrtf	V:0, I:4	148	rtf → html	convertisseur de documents de RTF vers HTML, etc
ooo2dbk	V:0, I:0	217	sxw → xml	convertisseur de documents OpenOffice.org SXW en DocBook XML
wp2x	V:0, I:0	202	WordPerfect → any	fichiers WordPerfect 5.0 et 5.1 vers TeX, LaTeX, troff, GML et HTML

Table 11.10 – Liste d'outils d'extraction de données XML

11.2.4 The XML data lint

Vous pouvez convertir les fichiers HTML non-XML en XHTML qui est une instance XML correctement formatée. XHTML peut être traité par les outils XML.

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

paquet	popcon	taille	fonction	description
libxml2-utils	V:21, I:234	181	xml ↔ html ↔ xhtml	outil XML en ligne de commandes <code>xmlLint(1)</code> (vérification de la syntaxe, reformatage, décomposition...)
tidy	V:1, I:13	84	xml ↔ html ↔ xhtml	vérificateur de syntaxe et reformateur HTML
weblint-perl	V:0, I:1	32	lint	vérificateur de syntaxe et de style minimal pour HTML
linklint	V:0, I:0	344	link check	vérificateur rapide de liens et outils de maintenance de sites web

Table 11.11 – Liste d'outils d'impression élégante du XML

Une fois qu'un fichier XML propre est créé, vous pouvez utiliser la technologie XSLT pour extraire des données dans le contexte de balisage, etc.

11.3 Composition

Le programme UNIX [troff](#), développé à l'origine par AT&T, peut être utilisé pour une composition simple. Il est habituellement utilisé pour créer des pages de manuel.

[TeX](#), créé par Donald Knuth, est un outil très puissant de composition et c'est le standard de fait. [LaTeX](#), écrit à l'origine par Leslie Lamport permet un accès de haut niveau à la puissance de TeX.

paquet	popcon	taille	mot clé	description
texlive	V:3, I:48	72	(La)TeX	système TeX pour la composition, l'aperçu et l'impression
groff	V:3, I:59	11870	troff	système de formatage de texte GNU troff

Table 11.12 – Liste des outils de typographie

11.3.1 Composition roff

Traditionnellement, [roff](#) est le système de traitement de texte principal sous UNIX. Consultez [roff\(7\)](#), [groff\(7\)](#), [groff\(1\)](#), [grotty\(1\)](#), [troff\(1\)](#), [groff_mdoc\(7\)](#), [groff_man\(7\)](#), [groff_ms\(7\)](#), [groff_me\(7\)](#), [groff_mm\(7\)](#), et « [info groff](#) ».

Vous pouvez lire ou imprimer un bon didacticiel et document de référence sur la [macro](#) « -me » dans « [/usr/share/doc/groff/](#) » en installant le paquet [groff](#).

ASTUCE

« [groff -Tascii -me -](#) » produit une sortie en texte brut avec du [code d'échappement ANSI](#). Si vous désirez obtenir une sortie semblable à une page de manuel avec de nombreux « ^H » et « _ », utilisez plutôt « [GROFF_NO_SGR=1 groff -Tascii -me -](#) ».

ASTUCE

Pour supprimer les « ^H » et les « _ » d'un fichier texte issu de [groff](#), filtrez le par « [col -b -x](#) ».

11.3.2 TeX/LaTeX

La distribution logicielle [TeX Live](#) offre un système TeX complet. Le métapaquet [texlive](#) fournit une bonne sélection de paquets de [TeX Live](#) qui devraient suffire aux tâches les plus courantes.

De nombreuses références sont disponibles pour [TeX](#) et [LaTeX](#) :

- [The teTeX HOWTO : le guide local de teTeX sous Linux](#) ;
- [tex\(1\)](#) ;
- [latex\(1\)](#) ;
- [texdoc\(1\)](#) ;
- [texdoctk\(1\)](#) ;
- « [The TeXbook](#) », par Donald E. Knuth, (Addison-Wesley) ;
- « [LaTeX - A Document Preparation System](#) », par Leslie Lamport, (Addison-Wesley) ;
- « [The LaTeX Companion](#) », par Goossens, Mittelbach, Samarin, (Addison-Wesley).

C'est l'environnement de composition typographique le plus puissant. De nombreux processeurs [SGML](#) l'utilisent comme processeur de texte en « backend ». [Lyx](#) qu'on trouvera dans le paquet [lyx](#) et [GNU TeXmacs](#) qui provient du paquet [texmacs](#) offrent un environnement d'édition agréable en mode [WYSIWYG](#) pour [LaTeX](#) bien que nombreux sont ceux qui choisissent d'utiliser [Emacs](#) et [Vim](#) pour éditer les sources.

De nombreuses ressources sont disponibles en ligne :

- [The TEX Live Guide - TEX Live 2007](#) ([/usr/share/doc/texlive-doc-base/english/texlive-en/live.html](#) » (paquet [texlive-doc-base](#)) ;
 - [Un guide simple pour Latex et Lyx](#) (« [A Simple Guide to Latex/Lyx](#) »)
 - [Traitement de texte avec LaTeX](#) (« [Word Processing Using LaTeX](#) »)
 - [Guide local de l'utilisateur de teTeX et LaTeX](#) (« [Local User Guide to teTeX/LaTeX](#) »)
-

Lorsque les documents deviennent plus importants, TeX cause parfois des erreurs. Vous devez augmenter la taille de l'espace dans « `/etc/texmf/texmf.cnf` » (ou, de manière plus rigoureuse, éditez « `/etc/texmf/texmf.d/95NonPath` » et lancez `update-texmf(8)`) afin de corriger cela.

Note

Les sources au format TeX de « The TeXbook » sont disponibles à l'adresse <http://tug.ctan.org/tex-archive/systems/knuth/dist/tex/texbook.tex>. Ce fichier contient la plupart des macros nécessaires. J'ai entendu dire que vous pouvez traiter ce document avec `tex(1)` après avoir commenté les lignes 7 à 10 et ajouté « `\input manmac \proofmodefalse` ». Je recommande vivement l'achat de ce livre (et de tous les autres livres de Donald E. Knuth) plutôt que d'utiliser la version en ligne mais la version source est un très bon exemple d'utilisation de TeX !

11.3.3 Imprimer convenablement une page de manuel

Vous pouvez imprimer convenablement une page de manuel en PostScript en utilisant l'une des commandes suivantes :

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

11.3.4 Créer une page de manuel

Bien que l'écriture d'une page de manuel (manpage) dans le format [troff](#) brut soit possible, il existe quelques paquets facilitant cette tâche :

paquet	popcon	taille	mot clé	description
docbook-to-man	V:0, I:12	191	SGML → manpage	macros de conversion de DocBook SGML vers roff man
help2man	V:0, I:10	491	text → manpage	générateur automatique d'une pages de manuel depuis --help
info2man	V:0, I:0	135	info → manpage	convertisseur depuis GNU info vers POD ou page de manuel
txt2man	V:0, I:1	114	text → manpage	convertir du texte brut ASCII au format d'une page de manuel

Table 11.13 – Liste de paquets facilitant la création de pages de manuel

11.4 Données imprimables

Sur un système Debian, les données imprimables sont définies dans le format [PostScript. Common UNIX Printing System \(CUPS\)](#) utilise Ghostscript en tant que programme de tramage « rasterisation » pour les imprimantes non-PostScript.

11.4.1 Ghostscript

Le cœur de la manipulation des données imprimables est l'interpréteur [Ghostscript PostScript \(PS\)](#) qui génère une image tramée (« raster image »).

ASTUCE

« `gs -h` » permet d'afficher la configuration de Ghostscript.

paquet	popcon	taille	description
ghostscript	V:256, I:576	231	L'interpréteur GPL Ghostscript PostScript/PDF
ghostscript-x	V:12, I:63	223	interpréteur GPL Ghostscript PostScript/PDF - prise en charge de l'affichage X
libpoppler102	V:55, I:89	4274	bibliothèque de rendu PDF dérivée du visualisateur PDF xpdf
libpoppler-glib8	V:228, I:463	445	bibliothèque de rendu PDF (bibliothèque partagée basée sur GLib)
poppler-data	V:102, I:621	13090	CMaps pour PDF la bibliothèque de rendu (pour la prise en charge de CJK : Adobe-*)

Table 11.14 – Liste des interpréteurs Ghostscript PostScript

11.4.2 Fusionner deux fichiers PS ou PDF

Vous pouvez fusionner deux fichiers [PostScript \(PS\)](#) ou [Portable Document Format \(PDF\)](#) en utilisant `gs(1)` de 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
```

Note

Le format [PDF](#) qui est un format imprimable multi-plateformes largement utilisé, est essentiellement un format [PS](#) compressé avec quelques fonctionnalités et extensions supplémentaires .

ASTUCE

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 Utilitaires pour les données imprimables

Les paquets suivants fournissant des utilitaires pour les données imprimables ont attiré mon attention :

11.4.4 Imprimer avec CUPS

Les commandes `lp(1)` et `lpr(1)` proposées par le [Common UNIX Printing System \(CUPS\)](#) fournissent toutes deux des options de personnalisation des données imprimables.

Vous pouvez imprimer 3 copies d'un fichier en utilisant une des commandes suivantes :

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

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

Vous pouvez personnaliser davantage le fonctionnement de l'imprimante en utilisant des options d'impression telles que « `-o number-up=2` », « `-o page-set=even` », « `-o page-set=odd` », « `-o scaling=200` », « `-o natural-scaling=200` » etc., documentées sur [Impression et options en ligne de commande](#).

11.5 La conversion de données de courrier électronique

Les paquets suivants, destinés à la conversion de données de courrier électronique, ont attiré mon attention :

paquet	popcon	taille	mot clé	description
poppler-utils	V:245, I:413	689	pdf → ps,text, ...	utilitaires PDF :: pdftops , pdfinfo , pdfimages , pdftotext , pdffonts
psutils	V:6, I:98	219	ps → ps	outils de conversion de document PostScript
poster	V:0, I:4	58	ps → ps	créer des posters de grande dimension à partir de pages PostScript
enscript	V:1, I:19	2132	text → ps, html, rtf	convertir du text ASCII en PostScript, HTML, RTF ou Pretty-Print
a2ps	V:0, I:14	3648	text → ps	Convertisseur de « N'importe quoi vers PostScript » et imprimeur élégant
pdftk	I:50	28	pdf → pdf	outil de conversion de documents PDF : pdftk
html2ps	V:0, I:3	249	html → ps	convertisseur de HTML vers PostScript
gnuhtml2latex	V:0, I:1	27	html → latex	convertisseur de html vers latex
latex2rtf	V:0, I:6	480	latex → rtf	convertir des documents de LaTeX en RTF qui peuvent être lus par MS Word
ps2eps	V:3, I:62	99	ps → eps	convertisseur de PostScript vers EPS (PostScript encapsulé)
e2ps	V:0, I:0	109	text → ps	convertisseur de texte vers PostScript avec la prise en charge du codage japonais
impose+	V:0, I:0	119	ps → ps	Utilitaires PostScript
trueprint	V:0, I:0	146	text → ps	imprime élégamment de nombreux codes source (C, C++, Java, Pascal, Perl, Pike, Sh, et Verilog) vers PostScript. (langage C)
pdf2svg	V:0, I:4	30	ps → svg	convertisseur de PDF vers le format Scalable vector graphics (« SVG »)
pdftoipe	V:0, I:0	71	ps → ipe	convertisseur de PDF vers le format IPE d'XML

Table 11.15 – Liste des utilitaires pour les données imprimables

paquet	popcon	taille	mot clé	description
sharutils	V:4, I:52	1421	mail	shar (1), unshar (1), uuencode (1), uudecode (1)
mpack	V:1, I:17	106	MIME	encoder et décoder des messages MIME : mpack (1) et munpack (1)
tnef	V:1, I:10	110	ms-tnef	dépaqueter des attachements MIME de type « application/ms-tnef » qui est un format propre à Microsoft
uudeview	V:0, I:4	109	mail	encodeur et décodeur pour les formats suivants : uuencode , xxencode , BASE64 , quoted printable et BinHex

Table 11.16 – Liste de paquets facilitant la conversion de données de courrier électronique

ASTUCE

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 Bases concernant les données de courrier électronique

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 [Tableau 11.2](#)).

Le format standard d'enregistrement du courrier électronique est « mbox » selon la [RFC2822 \(RFC822 mise à jour\)](#). Consultez `mbox(5)` (fourni par le paquet `mutt`).

Pour les langues européennes, on utilise habituellement pour le courriel « Content-Transfer-Encoding: quoted-printable » avec le jeu de caractères ISO-8859-1 car il n'y a pas beaucoup de caractères de 8 bits. Si le texte européen est codé en UTF-8, on préférera utiliser « Content-Transfer-Encoding: quoted-printable » car ce sont essentiellement des données sur 7 bits.

Pour le japonais, « Content-Type: text/plain; charset=ISO-2022-JP » est habituellement utilisé pour le courriel afin de conserver le texte sur 7 bits. Mais les anciens systèmes Microsoft peuvent envoyer des données de courriel en Shift-JIS sans le déclarer proprement. Si le texte japonais est codé en UTF-8, on utilisera de préférence [Base64](#) car il comporte de nombreuses données sur 8 bits. La situation des autres langues asiatiques est similaire.

Note

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.

Note

Si vous utilisez d'autres formats d'enregistrement de courriel, les mettre dans le format mbox est une bonne première étape. Un programme client souple comme `mutt(1)` peut être pratique pour le faire.

Vous pouvez éclater le contenu d'une boîte à lettre en messages séparés en utilisant `procmail(1)` et `formail(1)`.

Chaque message de courrier électronique peut être dépaqueté en utilisant `munpack(1)` qui provient du paquet `mpack` (ou d'autres outils spécialisés) afin d'en obtenir le contenu codé en MIME.

11.6 Outils de données graphiques

Les paquets suivants contenant des outils pour la conversion, l'édition et l'organisation de données graphiques ont attiré mon attention :

ASTUCE

Recherchez d'autres outils pour les images en utilisant l'expression rationnelle « `~Gworks-with::image` » dans `aptitude(8)` (consultez [Section 2.2.6](#)).

Bien que des programmes ayant une interface graphique comme `gimp(1)` soient très puissants, des outils en ligne de commandes comme `imagemagick(1)` sont assez utiles pour la manipulation automatique d'images au moyen de scripts.

Le format standard de fait pour les images d'appareils photo numériques est [Exchangeable Image File Format \(EXIF\)](#) qui est composé d'une image au format [JPEG](#) à laquelle sont ajoutées des balises de métadonnées. Il peut contenir des informations telles que la date, l'heure ou les paramètres de l'appareil photo.

Le brevet de compression de données sans perte [Lempel-Ziv-Welch \(LZW\)](#) est arrivé en fin de validité. Les utilitaires du format [Graphics Interchange Format \(GIF\)](#) qui utilise la méthode de compression LZW peuvent être maintenant librement disponibles sur un système Debian.

paquet	popcon	taille	mot clé	description
gimp	V:58, I:315	19706	image (bitmap)	Programme de manipulation d'images GNU (GNU Image Manipulation Program »)
imagemagick	I:373	220	image (bitmap)	programmes de manipulation d'images
graphicsmagick	V:2, I:17	5242	image (bitmap)	programmes de manipulation d'images (dérivés d' imagemagick)
xsane	V:14, I:165	2346	image (bitmap)	Interface graphique X11 de SANE basée sur GTK (Scanner Access Now Easy)
netpbm	V:29, I:382	5056	image (bitmap)	outils de conversion graphique
icoutils	V:18, I:115	221	png ↔ ico (bitmap)	convertit les icônes et curseurs de MS Windows de et vers des formats PNG (favicon.ico)
scribus	V:1, I:22	30370	ps/pdf/SVG/...	éditeur DTP Scribus
libreoffice-draw	V:150, I:416	14596	image (vectorielle)	suite de bureautique LibreOffice - dessin
inkscape	V:42, I:188	84892	image (vectorielle)	éditeur SVG (Scalable Vector Graphics)
dia	V:4, I:30	3727	image (vectorielle)	éditeur de diagrammes (Gtk)
xfig	V:2, I:14	1793	image (vectorielle)	Outil de génération interactive de figures sous X11
pstoedit	V:3, I:87	988	ps/pdf → image (vectorielle)	convertisseur de fichiers PostScript et PDF en graphiques vectoriels éditables (SVG)
libwmf-bin	V:8, I:191	113	Windows/image (vectorielle)	outils de conversion de métafichiers Windows (données graphiques vectorielles)
fig2sxd	V:0, I:0	149	fig → sxd (vectorielle)	conversion de fichiers XFig dans le format OpenOffice.org Draw
unpaper	V:2, I:19	460	image → image	outil de post-traitement pour des pages numérisées pour OCR
tesseract-ocr	V:8, I:37	1507	image → texte	logiciel OCR libre basé sur le moteur commercial OCR de HP
tesseract-ocr-eng	V:6, I:37	4032	image → texte	moteur de données OCR : fichier de langue de tesseract-ocr pour le texte en anglais
gocr	V:1, I:11	531	image → texte	logiciel OCR libre
ocrad	V:0, I:4	303	image → texte	logiciel OCR libre
eog	V:63, I:253	10189	image (Exif)	Programme de visualisation d'images « Eye of Gnome » (l'œil de Gnome)
gthumb	V:5, I:21	5483	image (Exif)	visionneuse et gestionnaire de photos (GNOME)
geeqie	V:6, I:19	14643	image (Exif)	Visualiseur d'images utilisant GTK
shotwell	V:16, I:217	6451	image (Exif)	gestionnaire de photos numériques (GNOME)
gtkam	V:0, I:5	1154	image (Exif)	Application destinée à rapatrier des données numériques depuis les appareils photo numériques (GTK)
gphoto2	V:0, I:12	955	image (Exif)	Le client en ligne de commande pour appareil photo numérique gphoto2
gwenview	V:27, I:94	10651	image (Exif)	visionneuse d'images (KDE)
kamera	I:93	798	image (Exif)	gestion des appareils photo numériques dans les applications KDE
digikam	V:2, I:12	2646	image (Exif)	application de gestion photos numériques (KDE)
exiv2	V:2, I:40	321	image (Exif)	Outil pour manipuler les méta-données EXIF/IPTC
exiftran	V:1, I:19	70	image (Exif)	transformer les images JPEG des appareils photo numériques
jhead	V:1, I:10	110	image (Exif)	manipuler la partie non graphique des fichiers JPEG compatibles avec Exif (appareils photo numériques)
exif	V:1, I:12	339	image (Exif)	utilitaire en ligne de commandes pour afficher les informations EXIF contenues dans les fichiers JPEG
exiftags	V:0, I:4	292	image (Exif)	utilitaire pour lire les balises Exif depuis un fichier JPEG d'appareil photo numérique
exifprobe	V:0, I:4	499	image (Exif)	lire les métadonnées des images numériques
dcraw	V:2, I:18	535	image (Raw) → ppm	décoder les images brutes (« raw ») des appareils photo numériques
similarity	V:2, I:4	70	image (Raw) → ppm	rechercher des images visuellement similaires ou

ASTUCE

Tous les appareils photo numériques ou les scanners ayant un support d'enregistrement amovible fonctionnent sous Linux avec des lecteurs « [USB storage](#) » s'ils sont conformes à la [Design rule for Camera Filesystem](#) et utilisent un système de fichiers [FAT](#). Consultez Section [10.1.7](#).

11.7 Diverses conversions de données

Il y a de nombreux programmes pour convertir les données. Les paquets suivants ont attiré mon attention en utilisant l'expression rationnelle « `~Guse::converting` » avec `aptitude(8)` (consultez Section [2.2.6](#)).

paquet	popcon	taille	mot clé	description
alien	V:2, I:31	163	rpm/tgz → deb	convertisseur de paquets étrangers vers des paquets Debian
freepwing	V:0, I:0	421	EB → EPWING	convertisseur de « Electric Book » (Livre électronique - populaire au Japon) en un simple format JIS X 4081 (un sous-ensemble de EPWING V1)
calibre	V:8, I:35	55048	tout → EPUB	Convertisseur de livre numérique et gestion de bibliothèque.

Table 11.18 – Liste d'outils divers de conversion de données

Vous pouvez aussi extraire des données depuis le format RPM avec ce qui suit :

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

Chapitre 12

Programmation

Je donne quelques indications pour apprendre à programmer sous le système Debian, suffisantes pour suivre le code source mis en paquets. Voici les paquets importants correspondant aux paquets de documentation pour la programmation .

Une référence en ligne est accessible en entrant « `man name` » après l'installation des paquets `manpages` et `manpages-dev`. Les références en ligne des outils GNU tools sont disponibles en entrant « `info nom_programme` » après l'installation des paquets de documentation pertinents. Vous devrez peut-être inclure les archives `contrib` et `non-free` en plus de l'archive `main` car certaines documentations GFDL ne sont pas considérées comme conformes à DFSG.

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



AVERTISSEMENT

N'utilisez pas « `test` » comme nom d'un fichier exécutable. « `test` » fait partie de l'interpréteur de commandes.



Attention

Vous devrez installer les programmes directement compilés à partir des sources dans « `/usr/local` » ou « `/opt` » afin d'éviter des collisions avec les programmes du système.

ASTUCE

Les exemples de code pour la création de « [Song 99 Bottles of Beer](#) » devraient vous donner de bonnes indications sur pratiquement tous les langages de programmation.

12.1 Les scripts de l'interpréteur de commande

Le [script de l'interpréteur de commandes](#) (« shell script ») est un fichier texte dont le bit d'exécution est positionné et qui contient des commandes dans le format suivant :

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

La première ligne indique l'interpréteur qui sera utilisé pour lire et exécuter le contenu de ce fichier.

La lecture des scripts de l'interpréteur de commandes est la **meilleure** manière de comprendre comment fonctionne un système de type UNIX. Je donne ici quelques indications et rappels de la programmation avec l'interpréteur de commandes. Consultez « Erreurs en shell » (<http://www.greenend.org.uk/rjk/2001/04/shell.html>) pour apprendre à partir d'erreurs.

Contrairement à l'interpréteur de commandes en mode interactif (consultez Section [1.5](#) et Section [1.6](#)), les scripts de l'interpréteur de commandes utilisent souvent des paramètres, des conditions et des boucles.

12.1.1 Compatibilité de l'interpréteur de commandes avec POSIX

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

- The default non-interactive POSIX shell `"/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`.

Évitez d'écrire des scripts de l'interpréteur de commandes avec des **bashismes** ou des **zshismes** afin de les rendre portables entre tous les interpréteurs POSIX. Vous pouvez le vérifier en utilisant `checkbashisms(1)`.

Bon : POSIX	À éviter : bashisme
<code>if ["\$toto" = "\$titi"] ; then ...</code>	<code>if ["\$toto" == "\$titi"] ; then ...</code>
<code>diff -u fichier.c.orig fichier.c</code>	<code>diff -u fichier.c{.orig,}</code>
<code>mkdir /tototiti /tototutu</code>	<code>mkdir /toto{titi,tutu}</code>
<code>funcname() { ... }</code>	<code>function funcname() { ... }</code>
<code>format octal : « \377 »</code>	<code>format hexadécimal : « \xff »</code>

Table 12.1 – Liste de bashismes typiques

La commande « `echo` » doit être utilisée avec les précautions suivantes car son implémentation diffère selon que l'on utilise les commandes internes ou externes de l'interpréteur de commandes :

- Éviter d'utiliser toutes les options de commandes sauf « `-n` ».
- Éviter d'utiliser les séquences d'échappement dans les chaînes de caractères car leur prise en compte varie.

Note

Bien que l'option « `-n` » ne soit **pas** vraiment de la syntaxe POSIX, elle est généralement acceptée.

ASTUCE

Utilisez la commande « `printf` » plutôt que la commande « `echo` » si vous avez besoin d'intégrer des séquences d'échappement dans la chaîne de sortie.

12.1.2 Paramètres de l'interpréteur de commandes

Des paramètres spéciaux de l'interpréteur de commandes sont souvent utilisés dans les scripts de l'interpréteur de commandes.

paramètre de l'interpréteur de commandes	valeur
<code>\$0</code>	nom de l'interpréteur ou du script de l'interpréteur
<code>\$1</code>	premier (1er) paramètre de l'interpréteur
<code>\$9</code>	neuvième (9ème) paramètre de l'interpréteur
<code>\$#</code>	nombre de paramètres positionnels
<code>"\$*"</code>	<code>"\$1 \$2 \$3 \$4 ..."</code>
<code>"\$@"</code>	<code>"\$1" "\$2" "\$3" "\$4" ...</code>
<code>\$?</code>	valeur de retour de la commande la plus récente
<code>\$\$</code>	PID de ce script de l'interpréteur de commandes
<code>#!</code>	PID de la tâche de fond la plus récemment lancée

Table 12.2 – Liste des paramètres de l'interpréteur de commandes

Les **expansions de paramètre** les plus courantes à retenir sont mentionnées ci-dessous :

Ici, les deux points « `:` » dans tous ces opérateurs sont en fait optionnels.

forme de l'expression du paramètre	valeur si var est positionnée	valeur si var n'est pas positionnée
<code>\${var:-chaîne}</code>	« \$var »	« chaîne »
<code>\${var:+chaîne}</code>	« chaîne »	« null »
<code>\${var:=chaîne}</code>	« \$var »	« chaîne » (et lancer « var=chaîne »)
<code>\${var:?chaîne}</code>	« \$var »	echo « chaîne » vers stderr (et quitter avec une erreur)

Table 12.3 – Liste des expansions de paramètre de l'interpréteur

- **avec** « : » = opérateur de test pour **existe** et **différent de null**
- **sans** « : » = opérateur de test pour **existe** uniquement

forme de substitution de paramètre	résultat
<code>\${var%suffixe}</code>	supprimer le motif de suffixe le plus petit
<code>\${var%%suffixe}</code>	supprimer le motif de suffixe le plus grand
<code>\${var#préfixe}</code>	supprimer le motif de préfixe le plus petit
<code>\${var##préfixe}</code>	supprimer le motif de suffixe le plus grand

Table 12.4 – Liste des substitutions-clés de paramètres de l'interpréteur

12.1.3 Opérateurs conditionnels de l'interpréteur

Chaque commande retourne un **état de sortie** qui peut être utilisé pour des expressions conditionnelles.

- Succès : 0 (« Vrai »)
- Erreur : différent de 0 (« Faux »)

Note

« 0 » dans le contexte conditionnel de l'interpréteur signifie « Vrai » alors que « 0 » dans le contexte conditionnel de C signifie « Faux ».

Note

« [» est l'équivalent de la commande `test`, qui évalue, comme expression conditionnelle, les paramètres jusqu'à «] ».

Les **idiomes conditionnels** de base à retenir sont les suivants :

- « `commande && si_succès_lancer_aussi_cette_commande || true` »
- « `commande || en_cas_de_non_succès_lancer_aussi_cette_commande || true` »
- Un morceau de script sur plusieurs lignes comme le suivant :

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

Ici, le « `|| true` » était nécessaire pour s'assurer que ce script de l'interpréteur ne se termine pas accidentellement à cette ligne lorsque l'interpréteur est appelé avec l'indicateur « `-e` ».

Les opérateurs de comparaison **arithmétique** entière dans les expressions conditionnelles sont « `-eq` », « `-ne` », « `-lt` », « `-le` », « `-gt` » et « `-ge` ».

équation	condition pour retourner une valeur logique vraie
<code>-e fichier</code>	<i>fichier</i> existe
<code>-d fichier</code>	<i>fichier</i> existe et est un répertoire
<code>-f fichier</code>	<i>fichier</i> existe et est un fichier normal (« régulier »)
<code>-w fichier</code>	<i>fichier</i> existe et peut être écrit
<code>-x fichier</code>	<i>fichier</i> existe et est exécutable
<code>fichier1 -nt fichier2</code>	<i>fichier1</i> est plus récent que <i>fichier2</i> (modification)
<code>fichier1 -ot fichier2</code>	<i>fichier1</i> est plus ancien que <i>fichier2</i> (modification)
<code>fichier1 -ef fichier2</code>	<i>fichier1</i> et <i>fichier2</i> sont sur le même périphérique et le même numéro d'inœud

Table 12.5 – Liste des opérateurs de comparaison dans les expressions conditionnelles

équation	condition pour retourner une valeur logique vraie
<code>-z str</code>	la longueur de <i>str</i> est nulle
<code>-n str</code>	la longueur de <i>str</i> est non nulle
<code>str1 = str2</code>	<i>str1</i> et <i>str2</i> sont égales
<code>str1 != str2</code>	<i>str1</i> et <i>str2</i> ne sont pas égales
<code>str1 < str2</code>	<i>str1</i> est trié avant <i>str2</i> (dépendant des paramètres linguistiques)
<code>str1 > str2</code>	<i>str1</i> est trié après <i>str2</i> (dépendant des paramètres linguistiques)

Table 12.6 – Liste des opérateurs de comparaison de chaîne de caractères dans les expressions conditionnelles

12.1.4 Boucles de l'interpréteur de commandes

Il existe un certains nombre d'idiomes de boucles qu'on peut utiliser avec un interpréteur de commandes POSIX.

- « `for x in toto1 toto2 ... ; do commande ; done` » boucle en assignant les éléments de la liste « `toto1 toto2 ...` » à la variable « `x` » et en exécutant la « `commande` ».
- « `while condition ; do commande ; done` » répète la « `commande` » tant que la « `condition` » est vraie.
- « `until condition ; do commande ; done` » répète la « `commande` » tant que « `condition` » n'est pas vraie.
- « `break` » permet de quitter la boucle.
- « `continue` » permet de reprendre l'itération suivante de la boucle.

ASTUCE

Les itérations numériques semblables à celles du langage [C](#) peuvent être réalisées en utilisant `seq(1)` comme générateur de « `toto1 toto2 ...` ».

ASTUCE

Consultez Section [9.4.9](#).

12.1.5 Séquence de traitement de la ligne de commandes de l'interpréteur

En gros, l'interpréteur de commandes traite un script de la manière suivante :

- l'interpréteur de commandes lit une ligne :
 - l'interpréteur de commandes regroupe une partie de la ligne sous forme d'un **élément** (« token » si elle se trouve entre «`...`» ou «`'...'`» :
 - l'interpréteur de commandes découpe les autres parties de la ligne en **éléments** comme suit :
 - Espaces : *espace tabulation saut-de-ligne*
-

- Metacharacters: `| ; & ()`
 - l'interpréteur de commandes vérifie les **mots réservés** pour chacun des éléments et ajuste son comportement s'il ne se trouve pas entre `"..."` ou `'...'`.
 - **mot réservé**: `if then elif else fi for in while unless do done case esac`
 - L'interpréteur de commandes étend les **alias** s'ils ne se trouvent pas entre `"..."` ou `'...'`.
 - l'interpréteur de commandes étend les **tilde** s'ils ne se trouvent pas entre `"..."` ou `'...'`.
 - `« ~ »` → répertoire personnel de l'utilisateur actuel
 - `« ~utilisateur »` → répertoire personnel de l'utilisateur
 - l'interpréteur de commandes étend les **paramètres** en leur valeur s'ils ne sont pas entre `'...'`.
 - **paramètre** : `« $PARAMETRE »` ou `« ${PARAMETRE} »`
 - l'interpréteur de commandes étend la **substitution de commande** si elle n'est pas entre `'...'`.
 - `« $(commande) »` → sortie de la « commande »
 - `« ` commande ` »` → sortie de la « commande »
 - l'interpréteur de commandes étend les **motifs génériques du chemin** aux fichiers correspondants s'ils ne sont pas entre `"..."` ou `'...'`.
 - `*` → n'importe quel caractère
 - `?` → un caractère
 - `[...]` → un caractère quelconque parmi « ... »
 - l'interpréteur de commandes recherche la **commande** dans ce qui suit et l'exécute.
 - définition de **fonction**
 - commande **interne** (« builtin »)
 - **fichier exécutable** dans « \$PATH »
 - l'interpréteur de commandes passe à la ligne suivante et recommence ce traitement depuis le début de la séquence.
- Des guillemets simples dans des guillemets doubles n'ont pas d'effet.

Exécuter `« set -x »` dans le script de l'interpréteur ou l'appel du script avec l'option `« -x »` fait imprimer par l'interpréteur de commandes toutes les commandes exécutées. C'est assez pratique pour le débogage.

12.1.6 Programmes utilitaires pour les scripts de l'interpréteur de commandes

De façon à rendre vos programmes de l'interpréteur de commandes aussi portables que possible dans tous les systèmes Debian, c'est une bonne idée de limiter les programmes utilitaires à ceux fournis par les paquets **essentiels**.

- `« aptitude search ~E »` affiche la liste des **essentiels**.
- `« dpkg -L nom_paquet |grep '/man/man.*/' »` affiche la liste des pages de manuel pour les commandes que fournit le paquet `nom_paquet`.

paquet	popcon	taille	description
dash	V:913, I:993	221	small and fast POSIX-compliant shell for sh
coreutils	V:901, I:999	17478	utilitaires du cœur de GNU
grep	V:837, I:999	1091	GNU grep, egrep and fgrep
sed	V:834, I:999	883	GNU sed
mawk	V:392, I:997	242	small and fast awk
debianutils	V:932, I:999	230	divers utilitaires spécifiques à Debian
bsdutils	V:708, I:999	394	utilitaires de base provenant de 4.4BSD-Lite
bsdextrautils	V:80, I:103	384	extra utilities from 4.4BSD-Lite
moreutils	V:12, I:37	237	utilitaires supplémentaires d'UNIX

Table 12.7 – Lites des paquets comportant des petits programmes utilitaires pour les scripts de l'interpréteur de commandes

ASTUCE

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.

See Section 1.6 for examples.

12.2 Scripting in interpreted languages

paquet	popcon	taille	documentation
dash	V:913, I:993	221	sh : small and fast POSIX-compliant shell for <code>sh</code>
bash	V:796, I:999	6470	sh : "info bash" provided by <code>bash-doc</code>
mawk	V:392, I:997	242	AWK : small and fast <code>awk</code>
gawk	V:359, I:440	2456	AWK : "info gawk" provided by <code>gawk-doc</code>
perl	V:609, I:992	705	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:31	379	Perl extension for the GNU ReadLine/History Library: <code>perlsh(1)</code>
libreply-perl	V:0, I:0	170	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:676, I:901	89	Python : <code>python3(1)</code> and html pages provided by <code>python3-doc</code>
tcl	V:28, I:390	22	Tcl : <code>tcl(3)</code> and detail manual pages provided by <code>tcl-doc</code>
tk	V:28, I:382	22	Tk : <code>tk(3)</code> and detail manual pages provided by <code>tk-doc</code>
ruby	V:141, I:308	35	Ruby : <code>ruby(1)</code> , <code>erb(1)</code> , <code>irb(1)</code> , <code>rdoc(1)</code> , <code>ri(1)</code>

Table 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 -I ./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 Section 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.

paquet	popcon	taille	description
x11-utils	V:171, I:581	712	<code>xmessage(1)</code> : afficher un message ou une question dans une fenêtre (X)
whiptail	V:116, I:996	71	afficher des boîtes de dialogues conviviales depuis des scripts de l'interpréteur de commandes (<code>newt</code>)
dialog	V:15, I:118	1222	afficher des boîtes de dialogues conviviales depuis des scripts de l'interpréteur de commandes (<code>ncurses</code>)
zenity	V:79, I:393	384	display graphical dialog boxes from shell scripts (GTK)
ssft	V:0, I:0	75	Outil frontal de scripts de l'interpréteur de commandes (enrobeur pour <code>zenity</code> , <code>kdial</code> et <code>dialog</code> avec <code>gettext</code>)
gettext	V:47, I:302	5843	« <code>/usr/bin/gettext.sh</code> »: traduire des messages

Table 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 Section 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 Folie de courts scripts en Perl

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 }' |
```

Il est équivalent à l'une quelconque des lignes suivantes :

```
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' |
```

La dernière est une devinette. Elle tire parti des fonctionnalités suivantes de Perl :

- L'espace est optionnel.
- Il existe une conversion automatique des nombres en chaîne de caractères.
- 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.

For more crazy Perl scripts, [Perl Golf](#) may be interesting.

12.3 Coding in compiled languages

paquet	popcon	taille	description
gcc	V:185, I:598	45	GNU C compiler
libc6-dev	V:241, I:615	14358	GNU C Library: Development Libraries and Header Files
g++	V:57, I:519	15	GNU C++ compiler
libstdc++-10-dev	V:16, I:76	17752	GNU Standard C++ Library v3 (development files)
cpp	V:318, I:755	42	GNU C preprocessor
gettext	V:47, I:302	5843	GNU Internationalization utilities
glade	V:0, I:7	1730	GTK User Interface Builder
valac	V:0, I:7	684	C# like language for the GObject system
flex	V:8, I:91	1279	LEX-compatible fast lexical analyzer generator
bison	V:9, I:101	2858	YACC-compatible parser generator
susv2	I:0	16	aller chercher « The Single UNIX Specifications v2 »
susv3	I:0	16	aller chercher « The Single UNIX Specifications v3 »
golang	I:21	11	Go programming language compiler
rustc	V:1, I:12	5720	Rust systems programming language
haskell-platform	I:5	12	Standard Haskell libraries and tools
gfortran	V:10, I:97	16	GNU Fortran 95 compiler
fpc	I:3	122	Free Pascal

Table 12.10 – List of compiler related packages

Here, Section [12.3.3](#) and Section [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

Vous pouvez définir un environnement propre pour compiler des programmes écrits dans le [langage de programmation C](#) par ce qui suit :

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

Le paquet `libc6-dev`, c'est-à-dire la bibliothèque GNU C, fournit la [bibliothèque C standard](#) qui est une collection de fichiers d'en-têtes et de routines de bibliothèque utilisée par le langage de programmation C.

Consultez les références pour C comme suit :

- « `info libc` » (références des fonctions de la bibliothèque C)
- `gcc(1)` et « `info gcc` »
- `chaque_nom_de_fonction_de_la_bibliothèque_C(3)`
- Kernighan & Ritchie, « Le langage de programmation C », 2ème édition (Prentice Hall)

12.3.2 Programme simple en C (gcc)

Un exemple simple « `example.c` » peut être compilé avec la bibliothèque « `libc` » pour donner l'exécutable « `run_example` » par ce qui suit :

```
$ 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
```

Ici, « `-lm` » est nécessaire pour lier la bibliothèque « `/usr/lib/libm.so` » depuis le paquet `libc6` pour `sqrt(3)`. La bibliothèque réelle se trouve dans « `/lib/` » avec le nom de fichier « `libm.so.6` » avec un lien symbolique vers « `libm-2.7.so` ».

Regardez le dernier paramètre du texte en sortie. Il y a plus de 10 caractères bien que « `%10s` » soit indiqué.

L'utilisation de fonctions effectuant des opérations sur des pointeurs en mémoire sans vérification des limites, telles que `sprintf(3)` et `strcpy(3)` a été rendue obsolète afin d'éviter les exploits de débordements de tampons qui utilisent les effets des débordements ci-dessus. Utilisez `snprintf(3)` et `strncpy(3)` en remplacement..

12.3.3 Flex -- un meilleur Lex

[Flex](#) est un générateur d'[analyse lexicale](#) rapide compatible avec [Lex](#).

On trouve un didacticiel de `flex(1)` dans « `info flex` ».

Vous devez fournir vos propres « `main()` » et « `yywrap()` ». Sinon votre programme flex devrait ressembler à ce qui suit pour se compiler sans bibliothèque (cela parce que « `yywrap` » est une macro et que « `%option main` » active de manière implicite « `%option noyywrap` »).

```
%option main
%%
.|\\n      ECHO ;
%%
```

Sinon, vous pouvez compiler avec l'option de l'éditeur de liens « `-lfl` » à la fin de la ligne de commandes de `cc(1)` (comme AT&T-Lex avec « `-ll` »). L'option « `%option` » n'est pas nécessaire dans ce cas.

12.3.4 Bison -- un meilleur Yacc

Un certain nombre de paquets fournissent un [analyseur LR à lecture anticipée](#) (« lookahead ») compatible avec [Yacc](#) ou un générateur d'[analyseur LALR](#) sous Debian.

paquet	popcon	taille	description
bison	V:9, I:101	2858	générateur d'analyseur GNU LALR
byacc	V:0, I:5	160	générateur d'analyseur Berkeley LALR
btyacc	V:0, I:0	243	générateur d'analyseur avec retour arrière basé sur byacc

Table 12.11 – Liste de générateurs d'analyseur LALR compatible avec Yacc

On trouve un didacticiel de `bison(1)` dans « `info bison` ».

Vous devez fournir vos propre « `main()` » et « `yyerror()` ». « `main()` » appelle « `yyparse()` » qui appelle « `yylex()` », habituellement créé avec Flex.

```
%%
```

```
%%
```

12.4 Outils d'analyse du code statique

[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.

ASTUCE

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.

12.5 Déboguer

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

12.5.1 Exécution de base de gdb

Le [débogueur](#) primaire sous Debian est `gdb(1)`, il vous permet d'inspecter un programme alors qu'il tourne.

Installons `gdb` et les programmes associés par ce qui suit :

```
# 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](#)”

paquet	popcon	taille	description
vim-ale	I:0	2022	Asynchronous Lint Engine for Vim 8 and NeoVim
vim-syntastic	I:3	1240	Syntax checking hacks for vim
elpa-flycheck	V:0, I:1	792	modern on-the-fly syntax checking for Emacs
elpa-relint	V:0, I:0	133	Emacs Lisp regexp mistake finder
cppcheck-gui	V:0, I:1	5529	tool for static C/C++ code analysis (GUI)
shellcheck	V:2, I:10	17977	lint tool for shell scripts
pyflakes3	V:1, I:15	23	passive checker of Python 3 programs
pylint	V:2, I:14	1430	vérificateur de code statique Python
perl	V:609, I:992	705	interpréteur ayant un vérificateur de code statique interne : B: :Lint(3perl)
rubocop	V:0, I:0	2390	Ruby static code analyzer
clang-tidy	V:0, I:5	22	clang-based C++ linter tool
splint	V:0, I:3	2320	outil pour vérifier de manière statique les bogues d'un programme en C
flawfinder	V:0, I:0	181	outil pour examiner le code source en C/C++ et rechercher des faiblesses du point de vue de la sécurité
black	V:1, I:3	475	uncompromising Python code formatter
perltidy	V:0, I:5	1674	Perl script indenter and reformatter
indent	V:0, I:11	425	C language source code formatting program
astyle	V:0, I:4	761	Source code indenter for C, C++, Objective-C, C#, and Java
bcpp	V:0, I:1	111	C(++) beautifier
xmlindent	V:0, I:1	53	XML stream reformatter
global	V:0, I:3	1896	Source code search and browse tools
exuberant-ctags	V:4, I:34	349	build tag file indexes of source code definitions

Table 12.12 – Liste des outils d'analyse du code statique :

paquet	popcon	taille	documentation
gdb	V:12, I:113	9799	« info gdb » fourni par gdb-doc
ddd	V:0, I:10	4092	« info ddd » fourni par ddd-doc

Table 12.13 – List of debug packages

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
```

ASTUCE

De nombreuses commandes de `gdb(1)` possèdent une abréviation. L'expansion à l'aide de la touche de tabulation fonctionne comme avec l'interpréteur de commandes.

12.5.2 Déboguer un paquet Debian

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 ./
```

Corriger les bogues si nécessaire.

Modifier la version du paquet pour ne pas entrer en collision avec les versions officielles de Debian, par exemple, en ajoutant « `+debug1` » pour la compilation d'une version de paquet existante, ou « `~pre1` » pour la compilation d'une version de paquet qui n'est pas encore diffusée de la manière suivante :

```
$ dch -i
```

Compiler et installer les paquets avec les symboles de débogage comme suit :

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

Vous devrez vérifier les scripts de construction du paquet et vous assurer que les options « `CFLAGS=-g -Wall` » sont positionnées pour la compilation des binaires.

12.5.3 Obtenir une trace

Si vous rencontrez un plantage de programme, signaler le bogue avec un copier-coller des informations de trace est une bonne idée.

The backtrace can be obtained by `gdb(1)` using one of the following approaches:

— Crash-in-GDB approach:

- Run the program from GDB.
- Crash the program.
- Type `"bt"` at the GDB prompt.

— 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"`.
- Type `"bt"` at the GDB prompt.

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 Section [9.4.12](#))

ASTUCE

Souvent, vous voyez une trace où une ou plusieurs des lignes de départ se trouvent dans `« malloc() »` ou `« g_malloc() »`. Lorsque cela arrive, il y a des chances pour que votre trace ne soit pas très utile. La meilleure façon de trouver des informations utiles est de définir la variable d'environnement `« $MALLOCCHECK_ »` à la valeur 2 (`malloc(3)`). Vous pouvez le faire en lançant `gdb` de la manière suivante :

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

12.5.4 Commandes avancées de gdb

commande	description des objectifs des commandes
(gdb) thread apply all bt	obtenir une trace de tous les processus d'un programme multi-processus (multi-threaded)
(gdb) bt full	obtenir les paramètres qui se trouvent sur la pile d'appel des fonctions
(gdb) thread apply all bt full	obtenir une trace et les paramètres en combinant les options précédentes
(gdb) thread apply all bt full 10	obtenir une trace et les paramètres des 10 premiers appels en supprimant ce qui n'est pas significatif
(gdb) set logging on	écrire le journal de sortie de <code>gdb</code> dans un fichier (le fichier par défaut est <code>« gdb.txt »</code>)

Table 12.14 – Liste des commandes avancées de `gdb`

12.5.5 Vérifier les dépendances avec les bibliothèques

Utilisez `ldd(1)` pour trouver les dépendances d'un programme avec des bibliothèques :

```
$ ldd /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)
```

Pour que `ls(1)` fonctionne dans un environnement `chroot`é, les bibliothèques ci-dessus doivent être disponibles dans votre environnement `chroot`é.

Consultez Section [9.4.6](#).

12.5.6 Dynamic call tracing tools

There are several dynamic call tracing tools available in Debian. See Section [9.4](#).

12.5.7 Déboguer les erreurs de X

Si un programme `aperçu1` de GNOME a reçu une erreur X, vous devriez obtenir un message comme suit :

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

Dans ce cas, vous pouvez essayer de faire tourner le programme avec « `--sync` » et arrêter sur la fonction « `gdk_x_error` » de manière à obtenir une trace.

12.5.8 Outils de détection des fuites de mémoire

Il y a plusieurs outils de détection des fuites de mémoire disponibles sous Debian.

paquet	popcon	taille	description
libc6-dev	V:241, I:615	14358	<code>mtrace(1)</code> : fonctionnalité de débogage de <code>malloc</code> dans <code>glibc</code>
valgrind	V:5, I:45	80378	débogueur mémoire et optimiseur
electric-fence	V:0, I:5	70	débogueur <code>malloc(3)</code>
libdmalloc5	V:0, I:3	393	bibliothèque de débogage de l'allocation mémoire
duma	V:0, I:0	259	library to detect buffer overruns and under-runs in C and C++ programs
leaktracer	V:0, I:2	57	traceur de fuites de mémoire pour les programmes C++

Table 12.15 – Liste des outils de détection des fuites de mémoire

12.5.9 Désassembler un binaire

Vous pouvez désassembler du code binaire avec `objdump(1)` en faisant ce qui suit :

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

Note

`gdb(1)` peut être utilisé pour désassembler du code de manière interactive

paquet	popcon	taille	documentation
make	V:174, I:603	1592	« info make » fourni par make-doc
autoconf	V:42, I:280	1846	« info autoconf » fourni par autoconf-doc
automake	V:41, I:278	1830	« info automake » fourni par automake1.10-doc
libtool	V:35, I:264	1198	“ info libtool ” provided by libtool-doc
cmake	V:16, I:122	22944	cmake(1) cross-platform, open-source make system
ninja-build	V:4, I:30	347	ninja(1) small build system closest in spirit to Make
meson	V:2, I:18	2761	meson(1) high productivity build system on top of ninja
xutils-dev	V:1, I:11	1466	imake(1) , xmkmf(1) , etc.

Table 12.16 – List of build tool packages

12.6 Build tools

12.6.1 Make

[Make](#) est un utilitaire destiné à la maintenance d’un groupe de programmes. Lors de l’exécution de [make\(1\)](#), [make](#) lit le fichier de règles, « [Makefile](#) » et met à jour une cible si elle dépend de fichiers qui ont été modifiés depuis que la cible a été modifiée pour la dernière fois ou si la cible n’existe pas. L’exécution de ces mises à jour peut être faite simultanément.

La syntaxe du fichier de règles est la suivante :

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

Ici, « [TAB] » est un code de tabulation. Chaque ligne est interprétée par l’interpréteur de commandes après que [make](#) ait effectué la substitution des variables. Utilisez « \ » à la fin d’une ligne pour poursuivre le script. Utilisez « \$\$ » pour entrer un « \$ » pour les valeurs des variables d’environnement d’un script de l’interpréteur de commandes.

On peut écrire des règles implicites pour la cible et les prérequis, par exemple, de la manière suivante :

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

Ici, la cible contient le caractère « % » (exactement 1 caractère). Le caractère « % » peut correspondre à n’importe quelle sous-chaîne non vide des noms de fichiers de la cible actuelle. De même pour les prérequis, utilisez « % » pour afficher la manière dont leur nom est en relation avec le nom de la cible actuelle.

variables automatiques	valeur
\$@	cible
\$<	première exigence
\$?	toutes les exigences plus récentes
\$^	toutes les exigences
\$*	« % » correspond au radical dans le motif cible

Table 12.17 – Liste des variables automatiques de make

expansion de la variable	description
<code>toto := titi</code>	expansion à la volée
<code>toto2 = titi</code>	expression récursive
<code>toto3 += titi</code>	ajouter

Table 12.18 – Liste de l’expansion des variables de make

Exécutez « `make -p -f/dev/null` » afin de voir les règles automatiques internes.

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 Compiler et installer un programme



AVERTISSEMENT

Ne pas écraser les fichiers du système avec les programmes que vous avez compilés en les installant.

Debian ne touche pas aux fichiers se trouvant dans « /usr/local/ » ou « /opt ». Donc, si vous compilez un programme depuis ses sources, installez-le dans « /usr/local/ » de manière à ce qu'il n'interfère pas avec 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 Désinstaller un programme

Si vous avez les sources d'origine et s'ils utilisent [autoconf](#)(1) et [automake](#)(1) et si vous-vous souvenez comment vous l'avez configuré, exécutez-le comme suit pour désinstaller le programme :

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

Sinon, si vous êtes absolument certain que le processus d'installation n'a mis des fichiers que sous « /usr/local/ » et qu'il n'y a là rien d'important, vous pouvez supprimer tout son contenu avec :

```
# 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 still build on the top of inherently slow [Make](#).
- [Ninja](#) initially released in 2012 is meant to replace Make for the further improved build speed but is also 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

Des pages web dynamiques et interactives simples peuvent être faites de la manière suivante :

- Les requêtes sont présentées au navigateur de l'utilisateur en utilisant des formulaires [HTML](#).
- Remplir et cliquer sur les entrées de formulaires envoie une des chaînes d'[URL](#) suivantes avec des paramètres codés depuis le navigateur vers le serveur web.
 - « `http://www.foo.dom/cgi-bin/programme.pl?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3` »
 - « `http://www.foo.dom/cgi-bin/programme.py?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3` »
 - « `http://www.foo.dom/programme.php?VAR1=VAL1&VAR2=VAL2&VAR3=VAL3` »
- « %nn » dans l'URL est remplacé par le caractère dont la valeur hexadécimale est nn.
- La variable d'environnement est définie à : « `QUERY_STRING="VAR1=VAL1 VAR2=VAL2 VAR3=VAL3"` ».
- Le programme [CGI](#) (l'un quelconque des « `programme.*` ») sur le serveur web s'exécute lui-même avec la variable d'environnement « `$QUERY_STRING` ».
- La sortie standard (stdout) du programme CGI est envoyée au navigateur web et présentée sous forme d'une page web dynamique interactive.

Pour des raisons de sécurité, il est préférable de ne pas réaliser soi-même de nouvelles bidouilles pour analyser les paramètres CGI. Il existe des modules bien établis pour cela, en Perl et Python. [PHP](#) est fourni avec ces fonctionnalités. Lorsqu'il est nécessaire d'enregistrer des données du client, on utilise des [cookies HTTP](#). Lorsqu'un traitement de données est nécessaire côté client, on utilise fréquemment [Javascript](#).

Pour davantage d'informations, consultez [Common Gateway Interface](#), [The Apache Software Foundation](#) et [JavaScript](#).

Rechercher « CGI tutorial » sur Google en entrant l'URL encodée <http://www.google.com/search?hl=en&ie=UTF-8&q=CGI+tutorial> directement dans la barre d'adresse du navigateur est une bonne méthode pour voir un script CGI en action sur le serveur Google.

12.8 La conversion du code source

Il existe des programmes pour convertir les codes sources.

paquet	popcon	taille	mot clé	description
perl	V:609, I:992	705	AWK → PERL	convertir le code source de AWK vers PERL : a2p(1)
f2c	V:0, I:5	442	FORTRAN → C	convertir le code source de FORTRAN 77 vers C/C++ : f2c(1)
intel2gas	V:0, I:0	178	intel → gas	convertisseur depuis NASM (format Intel) vers l'Assembleur GNU (GAS)

Table 12.19 – Liste des outils de conversion de code source

12.9 Créer un paquet Debian

Si vous désirez créer un paquet Debian, lisez ce qui suit :

- Chapitre [2](#) pour comprendre les bases du système de paquets
- Section [2.7.13](#) pour comprendre les bases du processus de portage
- Section [9.11.4](#) pour comprendre les techniques de base d'un environnement isolé (« chroot »)
- [debuild\(1\)](#), and [sbuid\(1\)](#)
- Section [12.5.2](#) pour recompiler avec les informations de débogage
- [Guide pour les responsables Debian](#) (le paquet `debmake-doc`)
- [Référence du développeur Debian](#) (paquet `developers-reference`)
- [Charte Debian](#) (paquet `debian-policy`)

Il existe des paquets tels que `debmake`, `dh-make`, `dh-make-perl`, etc., qui facilitent la réalisation des paquets.

Annexe A

Annexe

Voici les fondements de ce document.

A.1 Le labyrinthe de Debian

Le système Linux est une plateforme informatique très performante pour un ordinateur connecté au réseau. Cependant, apprendre à utiliser toutes ses possibilités n'est pas si facile. Configurer LPR avec une imprimante qui ne soit pas PostScript en était un bon exemple. (Il n'y a plus de problème maintenant car les nouvelles installations utilisent CUPS).

Il existe une carte très détaillée appelée « CODE SOURCE ». Elle est très précise mais très difficile à comprendre. Il existe aussi des références appelées HOWTO et mini-HOWTO. Elles sont plus faciles à comprendre mais ont tendance à trop se concentrer sur des détails et perdent de vue les aspects généraux. J'ai de temps en temps des problèmes à trouver la bonne section dans un long HOWTO quand j'ai besoin d'exécuter certaines commandes.

I hope this "Debian Reference (version 2.83)" (2021-09-01 02:14:16 UTC) provides a good starting direction for people in the Debian maze.

A.2 Historique du Copyright

La Référence Debian fut lancée par moi-même, Osamu Aoki <osamu at debian dot org> en tant qu'aide-mémoire personnel pour l'administration système. De nombreuses parties proviennent des connaissances que j'ai acquises sur [la liste de diffusion debian-user](#) et d'autres ressources Debian.

En suivant les suggestions de Josip Rodin, qui a été très actif dans le [Projet de documentation Debian \(DDP\)](#), la « Référence Debian (version 1, 2001-2007) » a été créée en tant que partie des documents du DDP.

Au bout de 6 années, j'ai réalisé que la « Référence Debian (version 1) » était dépassée et ai commencé à en réécrire de nombreux passages. La nouvelle « Référence Debian (version 2) » a été diffusée en 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 2.83)" (2021-09-01 02:14:16 UTC) covers mostly Bullseye (=stable) and Bookworm (=testing) Debian releases.

Le contenu de ce tutoriel tire son origine et son inspiration dans ce qui suit :

- « [Linux User's Guide](#) » de Larry Greenfield (décembre 1996)
 - rendu obsolète par « Debian Tutorial »
- « [Debian Tutorial](#) » de Havoc Pennington. (11 décembre 1998)

- écrit partiellement par Oliver Elphick, Ole Tetlie, James Treacy, Craig Sawyer et Ivan E. Moore II
- rendu obsolète par « Debian GNU/Linux: Guide to Installation and Usage »
- « [Debian GNU/Linux: Guide to Installation and Usage](#) » de John Goerzen et Ossama Othman (1999)
- rendue obsolète par la « Référence Debian (version 1) »

Les descriptions des paquets et des archives peuvent trouver une partie de leur origine et de leur inspiration dans ce qui suit :

- « [FAQ Debian](#) » (version de mars 2002, alors qu’elle était maintenue par Josip Rodin)

Le reste du contenu peut trouver son origine et son inspiration dans ce qui suit :

- « [Référence Debian](#) (version 1) » de Osamu Aoki (2001–2007)
- obsoleted by the newer “Debian Reference (version 2)” in 2008.

La version précédente « Référence Debian (version 1) » avait été créée par de nombreux contributeurs :

- la contribution principale pour les sujets relatifs à la configuration du réseau par Thomas Hood :
- une importante contribution au contenu sur les sujets relatifs à X et VCS par Brian Nelson :
- l’aide pour les scripts de construction et de nombreuses corrections de contenu par Jens Seidel :
- une relecture intensive de David Sewell :
- de nombreuses contributions par les traducteurs, les contributeurs et ceux qui ont signalés des bogues.

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.

La description du débogueur gdb a été augmentée en utilisant le [contenu du wiki Debian sur les « backtrace »](#) avec le consentement d’Ari Pollak, Loïc Minier et Dafydd Harries.

Le contenu de « Référence Debian (version 2.83) » (2021-09-01 02:14:16 UTC) est principalement mon propre travail à l’exception de ce qui est mentionné ci-dessus. Il a aussi été mis à jour par les contributeurs.

Le document « Référence Debian (version 1) » a été traduit par Guillaume Erbs (gerbs chez free point fr) et al.

Le document « Référence Debian (version 2) » a été traduit par Jean-Luc Coulon (f5ibh) (jean-luc.coulon chez wanadoo.fr) à partir de la version 1, et est maintenu avec l’aide de David Prévot (david chez tilapin point org) et la liste de contributeurs (debian-l10n-french chez lists point debian point org).

L’auteur, Osamu Aoki, remercie tous ceux qui ont aidé à rendre possible ce document.

A.3 Format du document

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