

Référence Debian

Copyright © 2013 Osamu Aoki

Ce guide de référence Debian (v2) (2017-05-14 15:39:44 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	14 mai 2017	

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	Invite de l'interpréteur de commandes sous X	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	4
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	7
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	11
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	15
1.2.11	Fichiers spéciaux de périphériques	15
1.2.12	procfs et sysfs	16
1.2.13	tmpfs	16
1.3	Midnight Commander (MC)	17

1.3.1	Personnalisation de MC	17
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	18
1.3.5	Éditeur interne de MC	18
1.3.6	Visualisateur interne de MC	19
1.3.7	Possibilités de démarrage automatique de MC	19
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	20
1.4.3	Combinaisons particulières de touches	21
1.4.4	Opérations de style UNIX avec la souris	21
1.4.5	Le visualisateur de fichiers	22
1.4.6	L'éditeur de texte	22
1.4.7	Définir un éditeur de texte par défaut	22
1.4.8	Personnaliser vim	23
1.4.9	Enregistrer les actions de l'interpréteur de commandes	23
1.4.10	Commandes UNIX de base	23
1.5	La commande simple de l'interpréteur de commandes	25
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 »	28
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	30
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	36

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	<code>apt-get</code> / <code>apt-cache</code> comparés à <code>aptitude</code>	47
2.2.2	Opérations de base de gestion des paquets en ligne de commandes	48
2.2.3	Utilisation interactive d' <code>aptitude</code>	49
2.2.4	Raccourcis clavier d' <code>aptitude</code>	49
2.2.5	Vues des paquets sous <code>aptitude</code>	50
2.2.6	Options de la méthode de recherche avec <code>aptitude</code>	51
2.2.7	Les formules d'expressions rationnelles d' <code>aptitude</code>	52
2.2.8	Résolution des dépendances par <code>aptitude</code>	52
2.2.9	Journaux d'activité des paquets	54
2.3	Exemples d'opérations avec <code>aptitude</code>	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	58
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 <code>aptitude</code>	61
2.5.7	Copies locales des paquets téléchargés	61

2.5.8	Nom de fichier d'un paquet Debian	62
2.5.9	La commande <code>dpkg</code>	62
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	74
2.7.14	Serveur mandataire (proxy) pour APT	75
2.7.15	Petite archive publique de paquets	75
2.7.16	Enregistrer et copier la configuration du système	77
2.7.17	Convertir ou installer un paquet binaire non Debian	77
2.7.18	Extraire un paquet sans <code>dpkg</code>	78
2.7.19	Autres lectures concernant la gestion des paquets	78
3	Initialisation du système	79
3.1	Aperçu du processus d'amorçage du système	79
3.1.1	Étage 1 : le BIOS	80
3.1.2	Étage 2 : le chargeur initial	80
3.1.3	Étage 3 : le système mini-Debian	82
3.1.4	Étage 4 : le système Debian normal	82
3.2	Initialisation façon SysV	84

3.2.1	Signification du niveau de fonctionnement (« runlevel »)	84
3.2.2	Configuration des niveaux de fonctionnement (« runlevel »)	85
3.2.3	Exemple de gestion de niveau de fonctionnement	85
3.2.4	Paramètre par défaut de chaque script init	86
3.2.5	Nom de machine (« hostname »)	86
3.2.6	Le système de fichiers	86
3.2.7	Initialisation de l'interface réseau	87
3.2.8	Initialisation des services réseau	87
3.2.9	Messages du système	87
3.2.10	Messages du noyau	87
3.3	Le système udev	88
3.3.1	Initialisation des modules du noyau	88
4	Authentication	90
4.1	Authentication normale d'UNIX	90
4.2	Gestion des informations des comptes et des mots de passes	92
4.3	Mot de passe de qualité	92
4.4	Créer un mot de passe chiffré	93
4.5	PAM et NSS	93
4.5.1	Fichiers de configuration auxquels accèdent PAM et NSS	94
4.5.2	Le système de gestion centralisée moderne	94
4.5.3	« Pourquoi la commande su de GNU ne gère-t-elle pas le groupe wheel »	95
4.5.4	Règle de mots de passe plus stricte	95
4.6	Autres contrôles d'accès	96
4.6.1	sudo	96
4.6.2	PolicyKit	96
4.6.3	SELinux	96
4.6.4	Restreindre l'accès à certains services du serveur	96
4.7	Sécurité de l'authentification	97
4.7.1	Mot de passe sûr avec Internet	97
4.7.2	Le shell sûr (Secure Shell)	98
4.7.3	Mesures de sécurité supplémentaires pour Internet	98
4.7.4	sécuriser le mot de passe de l'administrateur	98
5	Configuration du réseau	100
5.1	L'infrastructure de base du réseau	100
5.1.1	Résolution du nom d'hôte	102
5.1.2	Nom de l'interface réseau	103
5.1.3	Plage d'adresses réseau du réseau local (« LAN »)	103

5.1.4	La gestion du périphérique réseau	104
5.2	Configuration moderne de réseau pour ordinateur de bureau	104
5.2.1	Outils graphiques de configuration du réseau	104
5.3	Ancienne méthode de configuration et de connexion réseau	105
5.4	Méthode de connexion réseau (ancienne)	106
5.4.1	La connexion DHCP avec Ethernet	107
5.4.2	Connexion Ethernet avec une adresse IP statique	107
5.4.3	La connexion PPP avec pppconfig	107
5.4.4	Autre connexion PPP avec wvdialconf	108
5.4.5	La connexion PPPoE avec pppoeconf	109
5.5	Configuration de base du réseau avec ifupdown (ancienne)	109
5.5.1	La syntaxe de commande simplifiée	109
5.5.2	Syntaxe de base de « /etc/network/interfaces »	110
5.5.3	L'interface réseau de rebouclage (« loopback »)	110
5.5.4	Interfaces réseau gérées par DHCP	111
5.5.5	Interface réseau avec une adresse IP fixe	111
5.5.6	Les bases de l'interface réseau sans fil	111
5.5.7	L'interface de réseau local sans fil avec WPA/WPA2	112
5.5.8	L'interface réseau local sans fil avec WEP	112
5.5.9	La connexion PPP	113
5.5.10	La connexion alternative PPP	113
5.5.11	La connexion PPPoE	113
5.5.12	État de configuration réseau de ifupdown	113
5.5.13	Configuration réseau de base	114
5.5.14	La paquet ifupdown-extra	114
5.6	Configuration réseau avancée avec ifupdown (ancienne)	114
5.6.1	La paquet ifplugd	115
5.6.2	Le paquet ifmetric	115
5.6.3	L'interface virtuelle	115
5.6.4	Syntaxe de commande avancée	116
5.6.5	L'entrée « mapping »	117
5.6.6	Configuration réseau commutable manuellement	117
5.6.7	Scripts avec le système ifupdown	119
5.6.8	Cartographie réseau avec guessnet	120
5.7	Configuration réseau de bas niveau	120
5.7.1	Commandes Iproute2	120
5.7.2	Opérations sûres de bas niveau sur le réseau	120
5.8	Optimisation du réseau	121
5.8.1	Rechercher le MTU optimum	121
5.8.2	Définir le MTU	122
5.8.3	Optimisation de TCP sur le réseau Internet	123
5.9	Infrastructure de netfilter	123

6 Applications réseau	125
6.1 Navigateurs Web	125
6.1.1 Configuration du navigateur	126
6.2 Le système de courrier électronique	126
6.2.1 Bases du courrier électronique	127
6.2.2 Bases du service de courrier électronique moderne	127
6.2.3 Stratégie de configuration du courrier électronique pour une station de travail	128
6.3 Agent de transport de courrier électronique (« MTA »)	128
6.3.1 Configuration d'exim4	130
6.3.2 Configuration de postfix avec SASL	131
6.3.3 Configuration de l'adresse de courriel	132
6.3.4 Opération de base du MTA	133
6.4 Agent de courrier électronique de l'utilisateur (« MUA »)	133
6.4.1 Le MUA de base —Mutt	134
6.5 Utilitaire de récupération et de rediffusion du courriel distant	135
6.5.1 configuration de getmail	135
6.5.2 configuration de fetchmail	136
6.6 Agent de distribution du courriel (MDA) avec filtre	137
6.6.1 configuration de maildrop	137
6.6.2 configuration de procmail	138
6.6.3 Redistribuer le contenu d'une « mbox »	139
6.7 serveur POP3/IMAP4	139
6.8 Le serveur et les utilitaires d'impression	140
6.9 Le serveur et les utilitaires d'accès à distance (SSH)	140
6.9.1 Bases de SSH	141
6.9.2 Redirection de port pour un tunnel SMTP/POP3	143
6.9.3 Se connecter sans mot de passe distant	143
6.9.4 Clients SSH exotiques	144
6.9.5 Configurer ssh-agent	144
6.9.6 Comment arrêter le système distant par SSH	144
6.9.7 Résoudre les problèmes avec SSH	145
6.10 Autres serveurs d'applications réseau	145
6.11 Autres clients d'applications réseau	146
6.12 Le diagnostic des démons du système	146

7	Le système X Window	148
7.1	Paquets clés	148
7.2	Configurer l'environnement de bureau	148
7.2.1	Menu Debian	149
7.2.2	Menu de Freedesktop.org	149
7.2.3	Menu Debian sous « Freedesktop.org menu »	149
7.3	La relation client - serveur	149
7.4	Le serveur X	150
7.4.1	La (re-)configuration du serveur X	150
7.4.2	Les méthodes de connexion au serveur X	150
7.5	Démarrer le système X Window	151
7.5.1	Démarrer une session X avec gdm3	152
7.5.2	Personnaliser la session X (méthode classique)	152
7.5.3	Personnaliser la session X (nouvelle méthode)	152
7.5.4	Connexion à un client X distant via SSH	153
7.5.5	Sécuriser le terminal X au travers d'Internet	153
7.6	Polices dans X Window	153
7.6.1	Polices de base	154
7.6.2	Polices supplémentaires	154
7.6.3	Polices CJK	156
7.7	Applications X	156
7.7.1	Applications X de bureautique	156
7.7.2	Applications utilitaires pour X	158
7.8	Les infos de X	158
7.8.1	Presse-papier	158
7.8.2	Cartographie des claviers et affectation des boutons de souris sous X	159
7.8.3	Clients X classiques	159
7.8.4	Émulateur de terminal X —xterm	159
7.8.5	Faire tourner X avec le compte de l'administrateur	159
8	I18N et L10N	161
8.1	L'entrée clavier	161
8.1.1	Prise en charge de la méthode d'entrée avec iBus	162
8.1.2	Un exemple pour le japonais	162
8.1.3	Désactiver la méthode d'entrée	163
8.2	L'affichage de sortie	163
8.3	Les paramètres linguistiques (« locale »)	163
8.3.1	Bases du codage	163
8.3.2	Justification de l'utilisation d'UTF-8 dans les paramètres linguistiques	164

8.3.3	Reconfiguration des paramètres linguistiques	164
8.3.4	Valeur de la variable d'environnement « \$LANG »	165
8.3.5	Paramètre linguistique spécifique sous X Window	165
8.3.6	Coder les noms de fichiers	166
8.3.7	Messages et documentation traduits	166
8.3.8	Effet des paramètres linguistiques	166
9	Astuces du système	167
9.1	Le programme screen	167
9.1.1	Scénario d'utilisation de screen(1)	167
9.1.2	Raccourcis clavier de la commande screen	168
9.2	Enregistrer et présenter des données	168
9.2.1	Le démon de journal	168
9.2.2	Analyseur de journaux	168
9.2.3	Enregistrer proprement l'activité de la console	169
9.2.4	Affichage personnalisé des données de texte	170
9.2.5	Affichage personnalisé de la date et de l'heure	170
9.2.6	Écho colorisé de l'interpréteur de commandes	170
9.2.7	Commandes colorisées	171
9.2.8	Enregistrer l'activité de l'éditeur pour des répétitions complexes	171
9.2.9	Enregistrer l'image graphique d'une application X	171
9.2.10	Enregistrer les modifications dans des fichiers de configuration	172
9.3	Surveiller, contrôler et démarrer l'activité des programmes	172
9.3.1	Temps d'un processus	172
9.3.2	La priorité d'ordonnancement	174
9.3.3	La commande ps	174
9.3.4	La commande top	174
9.3.5	Afficher les fichiers ouverts par un processus	174
9.3.6	Tracer l'activité d'un programme	175
9.3.7	Identification des processus qui utilisent des fichiers ou des sockets	175
9.3.8	Répéter une commande avec un intervalle constant	175
9.3.9	Répéter une commande en bouclant entre des fichiers	175
9.3.10	Lancer un programme depuis l'interface graphique	176
9.3.11	Personnaliser le programme à lancer	177
9.3.12	Tuer un processus	177
9.3.13	Planifier des tâches qui s'exécutent une fois	178
9.3.14	Planifier des tâches qui s'exécutent régulièrement	178
9.3.15	touche Alt-SysRq	179
9.4	Astuces de maintenance du système	179

9.4.1	Qui se trouve sur le système ?	179
9.4.2	Prévenir tout le monde	180
9.4.3	Identification du matériel	180
9.4.4	Configuration matérielle	180
9.4.5	Heure système et matérielle	181
9.4.6	Configuration du terminal	182
9.4.7	L'infrastructure de gestion du son	182
9.4.8	Désactiver l'économiseur d'écran	183
9.4.9	Désactiver les bips	183
9.4.10	Utilisation de la mémoire	183
9.4.11	Vérification de la sécurité et de l'intégrité du système	184
9.5	Astuces relatives au stockage des données	184
9.5.1	Utilisation de l'espace disque	184
9.5.2	Configuration de la partition du disque	185
9.5.3	Accès à une partition en utilisant l'UUID	185
9.5.4	LVM2	186
9.5.5	Configuration de systèmes de fichiers	186
9.5.6	Création et vérification de l'intégrité d'un système de fichiers	187
9.5.7	Optimisation du système de fichiers à l'aide des options de montage	188
9.5.8	Optimisation du système de fichiers à l'aide du superbloc	188
9.5.9	Optimisation du disque dur	189
9.5.10	Optimisation du SSD	189
9.5.11	Utiliser SMART pour prédire les défaillances des disques durs	189
9.5.12	Indication du répertoire de stockage temporaire à l'aide de \$TMPDIR	190
9.5.13	Étendre l'espace de stockage utile à l'aide de LVM	190
9.5.14	Extension de l'espace de stockage en montant une autre partition	190
9.5.15	Extension de l'espace de stockage en remontant un autre répertoire	190
9.5.16	Extension de l'espace utilisable à l'aide de liens symboliques	191
9.5.17	Extension de l'espace de stockage utilisable à l'aide d'overlayfs	191
9.6	Le fichier image du disque	191
9.6.1	Créer le fichier image du disque	191
9.6.2	Écrire directement sur le disque	192
9.6.3	Monter le fichier image du disque	192
9.6.4	Nettoyage d'un fichier image du disque	193
9.6.5	Réaliser le fichier image d'un disque vide	194
9.6.6	Créer un fichier image ISO9660	194
9.6.7	Écriture directe sur CD/DVD-R/RW	195
9.6.8	Monter le fichier image ISO9660	195
9.7	Les données binaires	196

9.7.1	Afficher et éditer des données binaires	196
9.7.2	Manipulation des fichiers sans monter le disque	196
9.7.3	Redondance des données	196
9.7.4	Récupération de fichiers de données et analyse par autopsie	198
9.7.5	Éclater un gros fichier en petits fichiers	198
9.7.6	Effacer le contenu d'un fichier	198
9.7.7	Fichiers fictifs	198
9.7.8	Effacer l'ensemble du disque dur	199
9.7.9	Effacer l'ensemble du disque dur	199
9.7.10	Récupérer des fichiers supprimés mais encore ouverts	199
9.7.11	Rechercher tous les liens physiques	200
9.7.12	Consommation d'espace disque invisible	200
9.8	Astuces de chiffrement des données	201
9.8.1	Chiffrement des disques amovibles à l'aide de dm-crypt/LUKS	201
9.8.2	Partition d'échange chiffrée avec dm-crypt	202
9.8.3	Monter des disques amovibles chiffrés à l'aide de dm-crypt/LUKS	202
9.8.4	Chiffrer les fichiers automatiquement avec eCryptfs	203
9.8.5	Montage automatique de eCryptfs	203
9.9	Le noyau	204
9.9.1	Noyau Linux 2.6/3.x	204
9.9.2	Paramètres du noyau	204
9.9.3	En-têtes du noyau	204
9.9.4	Compiler le noyau et les modules associés	204
9.9.5	Compiler les sources du noyau : recommandations de l'équipe en charge du noyau Debian	205
9.9.6	Pilotes de matériel et microprogramme	206
9.10	Système virtualisé	206
9.10.1	Outils de virtualisation	206
9.10.2	Étapes de la virtualisation	208
9.10.3	Monter le fichier image du disque virtuel	208
9.10.4	Système protégé (chroot)	209
9.10.5	Systèmes de bureaux multiples	210
10	Gestion des données	211
10.1	Partager, copier et archiver	211
10.1.1	Outils d'archivage et de compression	213
10.1.2	Outils de copie et de synchronisation	213
10.1.3	Idiomes pour les archives	214
10.1.4	Idiomes pour la copie	214
10.1.5	Idiomes pour la sélection de fichiers	215

10.1.6	Support d'archive	216
10.1.7	Périphériques d'enregistrement amovibles	217
10.1.8	Choix de système de fichiers pour les données partagées	218
10.1.9	Partage de données au travers du réseau	219
10.2	Sauvegarde et restauration	219
10.2.1	Suites d'utilitaires de sauvegarde	221
10.2.2	Script d'exemple pour la sauvegarde du système	222
10.2.3	Un script de copie pour la sauvegarde des données	223
10.3	Infrastructure de sécurité des données	224
10.3.1	Gestion de clés pour GnuPG	224
10.3.2	Utilisation de GnuPG sur des fichiers	225
10.3.3	Utiliser GnuPG avec Mutt	225
10.3.4	Utiliser GnuPG avec Vim	225
10.3.5	La somme de contrôle MD5	227
10.4	Outils pour fusionner le code source	227
10.4.1	Extraire des différences pour des fichiers sources	227
10.4.2	Fusionner les mises à jour des fichiers source	227
10.4.3	Mise à jour par fusion de 3 sources	227
10.5	Systèmes de contrôle de version	229
10.5.1	Comparaison de commandes VCS	229
10.6	Git	231
10.6.1	Configuration du client Git	231
10.6.2	Références de Git	231
10.6.3	Commandes de Git	232
10.6.4	Git pour le dépôt Subversion	232
10.6.5	Git pour l'enregistrement de l'historique de la configuration	233
10.7	CVS	233
10.7.1	Configuration du dépôt CVS	234
10.7.2	Accès local à CVS	234
10.7.3	Accès distant à CVS avec pserver	234
10.7.4	Accès distant à CVS avec ssh	234
10.7.5	Importer une nouvelle source dans CVS	235
10.7.6	Permissions des fichiers du dépôt CVS	235
10.7.7	Flux de travail avec CVS	235
10.7.8	Derniers fichiers venant du CVS	237
10.7.9	Administration de CVS	238
10.7.10	Bit d'exécution pour la vérification (« checkout ») du CVS	238
10.8	Subversion	238
10.8.1	Configuration du dépôt Subversion	238

10.8.2	Accès à Subversion par l'intermédiaire d'un serveur Apache2	239
10.8.3	Accès local à Subversion par un groupe	239
10.8.4	Accès distant à Subversion avec SSH	239
10.8.5	Structure de répertoires de Subversion	239
10.8.6	Importer une nouvelle source vers Subversion	240
10.8.7	Flux de travail avec Subversion	240
11	Conversion de données	244
11.1	Outils de conversion de données textuelles	244
11.1.1	Convertir un fichier texte avec iconv	244
11.1.2	Vérifier que les fichiers sont codés en UTF-8 avec iconv	246
11.1.3	Convertir les noms de fichiers avec iconv	246
11.1.4	Convertir les fins de ligne (EOL)	246
11.1.5	Convertir les tabulations (TAB)	247
11.1.6	Éditeurs avec conversion automatique	247
11.1.7	Extraire du texte brut	248
11.1.8	Mettre en évidence et formater des données en texte brut	250
11.2	Données XML	250
11.2.1	Conseils de base pour XML	250
11.2.2	Traitement XML	251
11.2.3	Extraire des données XML	251
11.3	Composition	253
11.3.1	Composition roff	253
11.3.2	TeX/LaTeX	253
11.3.3	Imprimer convenablement une page de manuel	254
11.3.4	Créer une page de manuel	254
11.4	Données imprimables	254
11.4.1	Ghostscript	255
11.4.2	Fusionner deux fichiers PS ou PDF	255
11.4.3	Utilitaires pour les données imprimables	255
11.4.4	Imprimer avec CUPS	255
11.5	La conversion de données de courrier électronique	257
11.5.1	Bases concernant les données de courrier électronique	257
11.6	Outils de données graphiques	258
11.7	Diverses conversions de données	258

12 Programmation	261
12.1 Les scripts de l'interpréteur de commande	262
12.1.1 Compatibilité de l'interpréteur de commandes avec POSIX	262
12.1.2 Paramètres de l'interpréteur de commandes	263
12.1.3 Opérateurs conditionnels de l'interpréteur	263
12.1.4 Boucles de l'interpréteur de commandes	265
12.1.5 Séquence de traitement de la ligne de commandes de l'interpréteur	265
12.1.6 Programmes utilitaires pour les scripts de l'interpréteur de commandes	266
12.1.7 Dialogue de l'interpréteur de commandes	266
12.1.8 Exemple de script avec zenity	267
12.2 Make	268
12.3 C	269
12.3.1 Programme simple en C (gcc)	269
12.4 Déboguer	269
12.4.1 Exécution de base de gdb	270
12.4.2 Déboguer un paquet Debian	270
12.4.3 Obtenir une trace	271
12.4.4 Commandes avancées de gdb	271
12.4.5 Déboguer les erreurs de X	271
12.4.6 Vérifier les dépendances avec les bibliothèques	271
12.4.7 Outils de détection des fuites de mémoire	272
12.4.8 Outils d'analyse du code statique	272
12.4.9 Désassembler un binaire	272
12.5 Flex -- un meilleur Lex	273
12.6 Bison -- un meilleur Yacc	273
12.7 Autoconf	273
12.7.1 Compiler et installer un programme	273
12.7.2 Désinstaller un programme	274
12.8 Folie de courts scripts en Perl	274
12.9 Web	275
12.10 La conversion du code source	275
12.11 Créer un paquet Debian	275
A Annexe	276
A.1 Le labyrinthe de Debian	276
A.2 Historique du Copyright	276
A.3 Format du document	277

Liste des tableaux

1.1	Liste de paquets de programmes intéressants en mode texte	5
1.2	Liste de paquets de documentation	5
1.3	Utilisation des répertoires-clés	8
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	11
1.7	Liste des groupes importants fournis par le système pour l'accès aux fichiers	12
1.8	Liste des groupes importants fournis par le système pour l'exécution de commandes particulières	12
1.9	Liste des types d'horodatage	12
1.10	Liste des fichiers spéciaux de périphériques	16
1.11	Touches de raccourcis de MC	18
1.12	Réaction à la touche Entrée dans MC	19
1.13	Liste d'interpréteurs de commandes (« shells »)	20
1.14	Liste des raccourcis clavier de bash	21
1.15	Liste des opérations de style UNIX avec la souris	21
1.16	Liste des commandes UNIX de base	24
1.17	Les trois parties des paramètres linguistiques	26
1.18	Liste des recommandations de paramètres linguistiques	26
1.19	Afficher les valeurs de la variable « <code>\$HOME</code> »	28
1.20	Motifs génériques d'expansion du nom de fichier de l'interpréteur de commandes	29
1.21	Codes de retour de la commande	29
1.22	Idiomes des commandes de l'interpréteur	30
1.23	Descripteurs de fichier prédéfinis	31
1.24	Métacaractères pour BRE et ERE	33
1.25	Expressions de remplacement	34
1.26	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 <code>aptitude(8)</code> , <code>apt-get(8)</code> et <code>apt-cache(8)</code>	48
2.7	Options importantes de la commande <code>aptitude(8)</code>	49
2.8	Liste des raccourcis clavier d' <code>aptitude</code>	50
2.9	Liste des vues d' <code>aptitude</code>	51
2.10	Classement par catégories des vues de paquets standard	51
2.11	Liste des formules d'expressions rationnelles d' <code>aptitude</code>	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 <code>dpkg</code>	63
2.18	Liste de valeurs remarquables des priorités d'épinglage (Pin-Priority) pour la technique d'épinglage <code>apt</code> (« <code>apt-pinning</code> ») :	70
2.19	Liste des outils de proxy spécifiques à l'archive Debian	75
3.1	Liste des chargeurs initiaux	81
3.2	Signification des paramètres de GRUB	81
3.3	Liste d'utilitaires d'amorçage initial pour le système Debian :	83
3.4	Liste des niveaux de fonctionnement et description de leur utilisation	84
3.5	Liste des niveaux d'erreur du noyau	88
4.1	3 fichiers de configuration importants pour <code>pam_unix(8)</code>	90
4.2	Contenu de la seconde entrée de « <code>/etc/passwd</code> »	91
4.3	Liste des commandes servant à gérer les informations des comptes	92
4.4	Liste d'outils permettant de générer des mots de passe	93
4.5	Liste des paquets importants des systèmes PAM et NSS	93
4.6	Liste des fichiers de configuration auxquels PAM et NSS accèdent	94
4.7	Liste des services et ports sûrs et non sûrs	97
4.8	Liste des outils fournissant des mesures de sécurité supplémentaires	98
5.1	Liste des outils de configuration du réseau	101
5.2	Liste des plages d'adresses de réseau	103
5.3	Liste des méthodes de connexions réseau et des chemins de connexion	106
5.4	Liste des configurations de connexions réseau	106
5.5	Liste des acronymes de connexion au réseau	107
5.6	Liste des fichiers de configuration pour la connexion PPP avec <code>pppconfig</code>	108
5.7	Liste des fichiers de configuration pour une connexion PPP avec <code>wvdialconf</code>	108

5.8	Liste des fichiers de configuration pour une connexion PPPoE avec <code>pppoeconf</code>	109
5.9	Liste des commandes de base de configuration du réseau avec <code>ifupdown</code>	110
5.10	Liste des entrées de « <code>/etc/network/interfaces</code> »	110
5.11	Liste d'acronymes pour le WLAN	112
5.12	Terminologie des périphériques réseau	116
5.13	Liste des commandes avancées de configuration du réseau par <code>ifupdown</code>	116
5.14	Liste des variables d'environnement passées par le système <code>ifupdown</code>	119
5.15	Table de conversion depuis les commandes obsolètes <code>net-tools</code> vers les nouvelles commandes <code>iproute2</code>	120
5.16	Liste des commandes de réseau de bas niveau	121
5.17	Liste des outils d'optimisation du réseau.	121
5.18	Lignes directrices pour une valeur optimum de MTU	122
5.19	Liste d'outils de pare-feu	123
6.1	Liste de navigateurs web	125
6.2	Liste des paquets de greffons pour les navigateurs	126
6.3	Liste des paquets liés à l'agent de transport du courrier pour une station de travail	129
6.4	Liste des choix de paquets Debian d'agent de transport du courrier électronique (MTA)	129
6.5	Liste des pages de manuel importantes de postfix	131
6.6	Liste des fichiers de configuration liés aux adresses de courriel	132
6.7	Liste des opérations de base du MTA	133
6.8	Liste d'agents de courrier électronique de l'utilisateur (MUA)	134
6.9	Listes des utilitaires de récupération et de rediffusion du courriel	135
6.10	Liste de MDA avec filtrage	137
6.11	Liste de serveurs POP3/IMAP4	140
6.12	Liste des serveurs et utilitaires d'impression.	140
6.13	Liste des serveurs et des utilitaires d'accès à distance	141
6.14	Liste des protocoles et méthodes d'authentification SSH	141
6.15	Liste des fichiers de configuration de SSH	142
6.16	Liste d'exemples de démarrage du client SSH	143
6.17	Liste des clients SSH libres pour d'autres plateformes	144
6.18	Liste d'autres serveurs d'applications réseau	145
6.19	Liste de clients d'applications réseau	146
6.20	Liste des RFC courantes	147
7.1	Liste des (méta)paquets clés pour X Window	149
7.2	Liste de terminologie client/serveur	150
7.3	Liste des méthodes de connexion au serveur X	151
7.4	Table de paquets prenant en charge les systèmes de polices de X Window	154
7.5	Table de correspondance des polices PostScript Type 1	155

7.6	Table de correspondance des polices TrueType	155
7.7	Table de mots-clés utilisés dans les noms de polices CJK afin d'indiquer les types de polices	156
7.8	Liste d'applications de bureautique de base pour X	157
7.9	Liste d'applications utilitaires de base pour X	158
7.10	Liste des programmes basé sur X	159
8.1	Liste des méthodes d'entrées prises en charge par iBus	162
9.1	Liste de programmes gérant les interruptions de connexion au réseau	167
9.2	Liste des raccourcis clavier de screen	168
9.3	Liste des analyseurs de journaux système	169
9.4	Exemples d'affichage de la date et de l'heure pour la commande « <code>ls -l</code> » avec wheezy	170
9.5	Liste des outils de manipulation d'images	172
9.6	Liste de paquets pour enregistrer un historique de configuration dans VCS	172
9.7	Liste des outils de surveillance et de contrôle de l'activité des programmes	173
9.8	Liste des valeurs de politesse pour la priorité d'ordonnancement	174
9.9	Liste des styles de la commande ps	174
9.10	Liste des signaux couramment utilisés avec la commande kill	178
9.11	Listes des touches de commande SAK (« Secure attention keys »)	179
9.12	Listes des outils d'identification du matériel	180
9.13	Liste des outils de configuration du matériel	181
9.14	Liste des paquets son	182
9.15	Liste des commandes pour désactiver l'économiseur d'écran	183
9.16	Taille mémoire affichée	183
9.17	Liste d'outils pour la vérification de la sécurité et de l'intégrité du système	184
9.18	Listes de paquets de gestion de la partition du disque	185
9.19	Liste des paquets de gestion des systèmes de fichiers	187
9.20	Liste des paquets permettant de visualiser et d'éditer des données binaires	196
9.21	Liste des paquets pour manipuler les fichiers sans monter le disque	196
9.22	Liste d'outils pour ajouter des données de redondance aux fichiers	197
9.23	Liste de paquets pour la récupération de données et l'analyse par autopsie	197
9.24	Liste d'utilitaires de chiffrement des données	201
9.25	Liste des paquets-clés à installer pour la compilation du noyau sur un système Debian	205
9.26	Liste des outils de virtualisation	207
10.1	Liste des outils d'archivage et de compression	212
10.2	Liste des outils de copie et de synchronisation	214
10.3	Liste de choix de systèmes de fichiers pour des périphériques amovibles avec des scénarios typiques d'utilisation	218
10.4	Liste des services réseau à choisir avec le scénario typique d'utilisation	219
10.5	Liste de suites d'utilitaires de sauvegarde	221

10.6	Liste des outils d'une infrastructure de sécurité des données	224
10.7	Liste des commandes de GNU Privacy Guard pour la gestion des clés	224
10.8	Liste de la signification des codes de confiance	225
10.9	Liste des commandes de GNU Privacy Guard sur des fichiers	226
10.10	Liste d'outils destinés à fusionner du code source	228
10.11	Liste d'outils pour les systèmes de contrôle de version	229
10.12	Comparaison des commandes natives de VCS	230
10.13	Liste des paquets et des commandes relatifs à git	232
10.14	Options importantes des commandes CVS (à utiliser comme premier(s) paramètre(s) à cvs(1))	237
10.15	Options importantes des commandes de Subversion (à utiliser comme premier(s) paramètre(s) de svn(1))	243
11.1	Liste des outils de conversion de texte	244
11.2	Liste de valeurs de codage et leur utilisation	245
11.3	Liste des styles d'EOL pour différentes plateformes	247
11.4	Liste des commande de conversion de TAB des paquets <code>bsdmainutils</code> et <code>coreutils</code>	247
11.5	Liste d'outils pour extraite des données en texte brut	248
11.6	Liste des outils pour mettre en évidence des données de texte brut	249
11.7	Liste des entités XML prédéfinies	250
11.8	Liste d'outils XML	251
11.9	Liste des outils DSSSL	252
11.10	Liste d'outils d'extraction de données XML	252
11.11	Liste d'outils d'impression élégante du XML	252
11.12	Liste des outils de typographie	253
11.13	Liste de paquets facilitant la création de pages de manuel	254
11.14	Liste des interpréteurs Ghostscript PostScript	255
11.15	Liste des utilitaires pour les données imprimables	256
11.16	Liste de paquets facilitant la conversion de données de courrier électronique	257
11.17	Liste d'outils pour les données graphiques	259
11.18	Liste d'outils divers de conversion de données	260
12.1	Liste de paquets pour aider à la programmation	261
12.2	Liste de bashismes typiques	263
12.3	Liste des paramètres de l'interpréteur de commandes	263
12.4	Liste des expansions de paramètre de l'interpréteur	263
12.5	Liste des substitutions-clés de paramètres de l'interpréteur	264
12.6	Liste des opérateurs de comparaison dans les expressions conditionnelles	264
12.7	Liste des opérateurs de comparaison de chaîne de caractères dans les expressions conditionnelles	265
12.8	Listes des paquets comportant des petits programmes utilitaires pour les scripts de l'interpréteur de commandes	266
12.9	Liste d'interfaces utilisateur	267

12.10	Liste des variables automatiques de make	268
12.11	Liste de l'expansion des variables de make	268
12.12	Liste des commandes avancées de gdb	271
12.13	Liste des outils de détection des fuites de mémoire	272
12.14	Liste des outils d'analyse du code statique :	272
12.15	Liste de générateurs d'analyseur LALR compatible avec Yacc	273
12.16	Liste des outils de conversion de code source	275

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 [Référence Debian \(version 2\)](#) (2017-05-14 15:39:44 UTC) est destinée à fournir un large aperçu de l'administration d'un système Debian, sous la forme d'un guide « post-installation » de l'utilisateur.

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 les instructions d'installation, voir :

- [Debian GNU/Linux : Guide d'Installation et d'Utilisation](#)
- [Debian GNU/Linux Guide d'Installation pour la version testing actuelle](#)

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 \(DFSG\)](#) ;
- un effort distribué, volontaire et bénévole, basé sur Internet : <http://www.debian.org> ;
- un grand nombre de logiciels de grande qualité et déjà compilés ;
- l'accent sur la stabilité et la sécurité avec un accès facile aux mises à jour de sécurité ;
- l'accent 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**) ;
- être objectif (utiliser [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 Debian en <http://www.debian.org> pour les 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 <http://wiki.debian.org/> pour les sujets spécifiques ou changeants ;
- les HOWTOs du Projet de documentation Linux (TLPD) en <http://tldp.org/> et leurs traductions disponibles sur la partie francophone en <http://fr.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) :

```
# <commande avec le compte root>
$ <commande avec le compte d'un utilisateur>
```

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.

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/init.d/` ». Vous êtes encouragé à explorer son contenu en entrant :

```
$ mc "/etc/init.d/"
```

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.html` » ou par son **URL**, comme par exemple <http://www.debian.org>. Vous êtes encouragé à lire la documentation en entrant :

```
$ zcat "/usr/share/doc/sysv-rc/README.runlevels.gz" | sensible-pager
$ sensible-browser "/usr/share/doc/base-passwd/users-and-groups.html"
$ sensible-browser "http://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 2017-05-07 12:05:21 UTC et contiennent un total de 197536 soumissions de rapports pour 161289 paquets binaires et 26 architectures.

Note

Vous remarquerez que l'archive `amd64 unstable` ne contient actuellement que 55089 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.

Quelques citations pour les nouveaux utilisateurs

Voici quelques citations intéressantes venant de la liste de diffusion Debian et qui peuvent 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

Au démarrage du système, si vous n'avez pas installé le [système X Window](#) avec un gestionnaire d'affichage tel que `gdm3`, un écran d'identification en mode caractères vous est présenté. Supposons que le nom d'hôte de votre machine soit `toto`, l'invite d'identification ressemblera alors à ceci :

```
toto login:
```

Si vous avez installé un [environnement graphique](#) (GUI) tel que [GNOME](#) ou [KDE](#), vous pouvez alors obtenir une telle invite d'identification par `Ctrl-Alt-F1`, vous pourrez ensuite revenir à l'environnement graphique par `Alt-F7` (consultez [Section 1.1.6](#) ci-dessous pour davantage d'informations).

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 jessie/sid toto tty1
toto login: pingouin
Password:
Last login: Mon Sep 23 19:36:44 JST 2013 on tty3
Linux snoopy 3.11-1-amd64 #1 SMP Debian 3.11.6-2 (2013-11-01) 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.
toto:~$
```

Dans cet exemple, la partie principale du message d'accueil peut être personnalisée en éditant le fichier « `/etc/motd.tail` ». La première ligne est générée à partir des informations du système obtenues par la commande « `uname -snrv` ».

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

1.1.2 Invite de l'interpréteur de commandes sous X

Si, lors de l'installation, vous avez installé, avec le [système X Window](#), un gestionnaire graphique de session tel que `gdm3` de [GNOME](#) en sélectionnant la tâche « Environnement de bureau », un écran d'identification graphique vous sera présenté au démarrage de votre système. Entrez votre nom d'utilisateur et votre mot de passe pour vous connecter à un compte d'utilisateur sans privilège. Utilisez la touche de tabulation pour passer du champ d'entrée de l'utilisateur à celui du mot de passe ou utilisez la souris et un clic-gauche.

Sous X, vous pouvez obtenir une invite de l'interpréteur de commandes en lançant un programme d'émulation de terminal - `x` comme `gnome-terminal(1)`, `rxvt(1)` ou `xterm(1)`. Sous l'environnement de bureau GNOME, il suffit de cliquer « Applications » → « Accessoires » → « Terminal ».

Vous pouvez aussi consulter la Section [1.1.6](#) ci-dessous.

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 ;
- définir le propriétaire 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 ;
- sous l'environnement de bureau GNOME, cliquez « Applications » → « Accessoires » → « Terminal administrateur » ;
- 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

Lorsque le menu de votre environnement de bureau ne démarre pas automatiquement, avec les droits appropriés, l'interface graphique des outils d'administration du système, vous pouvez les démarrer depuis l'invite de l'interpréteur du compte de l'administrateur dans un émulateur de terminal X, tel que `gnome-terminal(1)`, `rxvt(1)` ou `xterm(1)`. Consultez Section [1.1.4](#) et Section [7.8.5](#).

**AVERTISSEMENT**

Ne jamais lancer un affichage X ou un gestionnaire de session avec le compte de l'administrateur en entrant `root` à l'invite d'un gestionnaire d'écran tel que `gdm3(1)`.

**AVERTISSEMENT**

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.

Depuis le système X Window, vous avez accès à la console 1 en pressant la combinaison de touches **Ctrl-Alt-F1**, c'est-à-dire que la touche **Ctrl** de gauche, la touche **Alt** de gauche et la touche **F1** sont pressées simultanément. Vous pouvez revenir au système X Window, qui tourne normalement sur la console virtuelle 7, en pressant **Alt-F7**.

Vous pouvez aussi changer de console virtuelle, par exemple pour la console 1, par la commande :

```
# chvt 1
```


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
```

Pour arrêter le système, vous pouvez aussi utiliser `Ctrl-Alt-Suppr` (la touche `Ctrl` de gauche, la touche `Alt` de gauche et la touche `Suppr` pressées simultanément) si la ligne `ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -h now` se trouve dans le fichier « `/etc/inittab` ». Consultez `inittab(5)` pour davantage d'informations.

Consultez Section [6.9.6](#).

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.

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 nom_paquet
```

paquet	popcon	taille	description
mc	V:69, I:237	1431	gestionnaire de fichiers plein écran en mode texte
sudo	V:347, I:708	3033	programme donnant aux utilisateurs des privilèges d'administration limités
vim	V:118, I:393	2374	éditeur de texte UNIX Vi amélioré (Vi IMproved), éditeur de texte pour programmeurs (version standard)
vim-tiny	V:65, I:967	1067	éditeur de texte UNIX Vi amélioré (Vi IMproved), éditeur de texte pour programmeurs (version compacte)
emacs25	V:2, I:4	19225	projet GNU Emacs, éditeur de texte extensible basé sur Lisp
w3m	V:212, I:813	2294	navigateurs WWW en mode texte
gpm	V:13, I:21	489	couper-coller à la mode UNIX sur une console texte (démon)

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

paquet	popcon	taille	description
doc-debian	I:856	166	projet de documentation Debian, (FAQ Debian) et autres documents
debian-policy	I:107	3744	Charte Debian et documents associés
developers-reference	I:7	1299	Guides et informations pour les responsables Debian
maint-guide	I:4	1005	Guide des nouveaux responsables Debian
debian-history	I:1	5034	Histoire du projet Debian
debian-faq	I:843	1277	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 poisson
```

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 poisson
```

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 "pingouin 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 "pingouin 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.

**Attention**

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).

Note

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 « `/share/doc/linux-doc-*/Documentation/filesystems/` ».

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.map.gz ». 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. Si vous êtes vraiment intéressé, consultez « /usr/include/linux/fs.h » la définition exacte de « struct inode » dans le système Debian GNU/Linux. 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 répertoire ;
- 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 <nouvel_utilisateur> toto
# chgrp <nouveau_groupe> toto
# chmod [ugoa][+ -=][rwxXst][, ...] toto
```

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 /un/emplacement/
# chown -R toto:titi .
# 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----- 1 root root 108, 0 oct. 20 07:12 /dev/ppp
-rw-r--r-- 1 root root 2718 oct. 5 22:09 /etc/passwd
-rw-r----- 1 root shadow 1741 oct. 5 22:09 /etc/shadow
-rwsr-xr-x 1 root root 973824 sept. 23 07:04 /usr/sbin/exim4
$ ls -ld /tmp /var/tmp /usr/local /var/mail /usr/src
drwxrwxrwt 17 root root 102400 oct. 20 16:42 /tmp
drwxrwsr-x 10 root staff 4096 nov. 29 2011 /usr/local
drwxr-xr-x 13 root root 4096 sept. 30 11:24 /usr/src
drwxrwsr-x 2 root mail 4096 sept. 27 11:07 /var/mail
drwxrwxrwt 5 root root 4096 oct. 20 09:14 /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 « `rw` » 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 toto titi
$ chmod u=rw,go=r toto
$ chmod 644 titi
$ ls -l toto titi
-rw-r--r-- 1 pingouin pingouin 0 oct. 20 16:46 titi
-rw-r--r-- 1 pingouin pingouin 0 oct. 20 16:46 toto
```

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)`.

```
(permissions du fichier) = (permissions demandées pour le fichier) & ~(valeur de umask)
```

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 new grp` »).

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.

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

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

groupe	commandes accessibles
sudo	exécuter SUDO sans son mot de passe.
lpadmin	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.html » 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 commande 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
mtime	date de modification du fichier (ls -l)
ctime	date de changement d'état du fichier (ls -lc)
atime	date de dernier accès au fichier (ls -lu)

Table 1.9 – Liste des types d'horodatage

Note

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

- 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.

Note

Sur un système Debian, même la simple lecture d'un fichier provoque normalement une opération d'écriture afin de mettre à jour les informations **atime** dans l'**inœud**. Monter un système de fichiers avec l'option « **noatime** » ou « **relatime** » fait que le système n'effectuera pas cette opération et, en conséquence, les accès aux fichiers en lecture seront plus rapides. C'est souvent recommandé pour les portables, parce que cela réduit l'activité disque et préserve l'autonomie de la batterie. Consultez **mount(8)**.

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

En ce qui concerne l'horodatage, la commande `ls` affiche des chaînes différentes avec les paramètres linguistiques non anglais (« `fr_FR.UTF-8` ») qu'avec le vénérable (« `C` »).

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

ASTUCE

Consultez Section 9.2.5 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 "Contenu d'origine" > toto
$ ls -li toto
1449840 -rw-rw-r-- 1 pingouin pingouin 18 oct. 20 16:50 toto
$ ln toto titi # lien physique
$ ln -s toto tutu # lien symbolique
$ ls -li toto titi tutu
1449840 -rw-rw-r-- 2 pingouin pingouin 18 oct. 20 16:50 titi
1449840 -rw-rw-r-- 1 pingouin pingouin 18 oct. 20 16:50 toto
1450180 lrwxrwxrwx 2 pingouin pingouin 18 oct. 20 16:55 tutu -> toto
$ rm toto
$ echo "Nouveau contenu" > toto
$ ls -li toto titi tutu
1450183 -rw-rw-r-- 2 pingouin pingouin 18 oct. 20 16:56 titi
1449840 -rw-rw-r-- 1 pingouin pingouin 18 oct. 20 16:50 toto
1450180 lrwxrwxrwx 2 pingouin pingouin 18 oct. 20 16:55 tutu -> toto
$ cat titi
Contenu d'origine
$ cat tutu
Nouveau contenu
```

Un lien physique peut être mis en place à l'intérieur du même système de fichiers, il partage le même numéro d'inœud, ce que montre l'option « `-i` » de `ls(1)`.

Le lien symbolique a les permissions d'accès nominales au fichier de « `rw-rw-rw-` » 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 montube
$ echo "salut" >montube & # mettre en arrière-plan
[1] 8022
$ ls -l montube
prw-r--r-- 1 pingouin pingouin 0 oct. 20 17:02 montube
$ cat montube
salut
[1]+  Fini                  echo "salut" > montube
$ ls montube
montube
$ rm montube
```

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---- 1 root disk      8,  0 20 oct.  07:13 /dev/sda
brw-rw----+ 1 root cdrom    11,  0 20 oct.  07:13 /dev/sr0
crw-rw---- 1 root dialout   4, 64 20 oct.  07:13 /dev/ttyS0
crw-rw-rw- 1 root root       1,  5 20 oct.  07:13 /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.

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

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

Table 1.10 – Liste des fichiers spéciaux de périphériques

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/filesystems/* ») 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

Sur certaines consoles telles que `gnome-terminal(1)`, les actions sur les touches de fonctions peuvent être récupérées par le programme de console. Vous pouvez désactiver cette fonctionnalité par « Éditer » → « Raccourcis clavier » pour le `terminal gnome`.

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 [9.4.6](#).

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 fichier_à_éditer
```

```
$ mcedit fichier_à_éditer
```

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-F<n>** 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 chemin/vers/fichier_à_voir
```

```
$ mcview chemin/vers/fichier_à_voir
```

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 9.3.11). 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 « <code>*.tar.gz</code> » et « <code>*.deb</code> »	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.nomdomaine` » ce 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 comment 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)`.

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`.

paquet	popcon	taille	Interpréteur POSIX	description
bash	V:853, I:999	5799	Oui	Bash : Shell GNU Bourne Again. (standard de fait)
tcsh	V:11, I:41	1306	Non	Shell TENEX C : version améliorée de csh de Berkeley
dash	V:914, I:980	204	Oui	Le Shell Almquist de Debian. Bon pour les scripts en shell
zsh	V:34, I:67	2150	Oui	Z shell : interpréteur standard avec de nombreuses améliorations
mksh	V:3, I:13	1281	Oui	Une version de Korn shell
csh	V:2, I:12	336	Non	C Shell OpenBSD , une version de csh de Berkeley
sash	V:0, I:5	969	Oui	Interpréteur de commandes autonome avec des commandes intégrées. (Ne convient pas en tant que « <code>/bin/sh</code> » standard)
ksh	V:5, I:23	3277	Oui	la vraie version de AT&T du shell Korn
rc	V:0, I:4	154	Non	implémentation du shell rc Plan 9 de AT&T
posh	V:0, I:0	187	Oui	Policy-compliant Ordinary SHell (dérivé de <code>pdksh</code>)

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

1.4.2 Personnaliser bash

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

Essayez, par exemple, ce qui suit :

```
# activer 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 en quittant MC
. /usr/lib/mc/mc.sh

# définir une valeur valable de CDPATH
CDPATH=./usr/share/doc:~/Desktop:~
export CDPATH

PATH="${PATH:$PATH:}/usr/sbin:/sbin"
# définir le PATH pour qu'il comporte un répertoire bin privé de l'utilisateur s'il existe
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 9.2.7, 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 :

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

ASTUCE

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

1.4.4 Opérations de style UNIX avec la souris

Les opérations de style UNIX avec la souris sont basées sur une souris ayant 3 boutons.

action	réponse
Clic-gauche et glisser de la souris	sélectionner et copier dans le presse-papiers
Clic-gauche	sélectionner le début de la sélection
Clic-droit	sélectionner la fin de la sélection et placez-là dans le presse-papiers
Clic du milieu	coller le contenu du presse-papiers à l'emplacement du curseur

Table 1.15 – Liste des opérations de style UNIX avec la souris

La molette centrale des souris à molette modernes est considérée comme le bouton du milieu et peut être utilisée pour les clics-milieu. Cliquer simultanément le bouton de gauche et le bouton de droite sert à émuler le bouton du milieu sur les systèmes ayant une souris à 2 boutons. Pour pouvoir utiliser une souris avec les consoles Linux en mode caractère, il faut que `gpm(8)` tourne en tant que démon.

1.4.5 Le visualisateur de fichiers

La commande `less(1)` invoque le visualisateur avancé (afficheur de contenu de fichier). Il affiche le contenu du fichier passé en argument ou de l'entrée standard. Pressez « h » si vous avez besoin d'une aide en ligne. Il peut faire beaucoup plus de choses que `more(1)` et peut être surchargé en exécutant « `eval $(lesspipe)` » ou « `eval $(lessfile)` » dans le script de démarrage de l'interpréteur de commandes. Vous trouverez davantage d'informations dans « `/usr/share/doc/lessf/LES SOPEN` ». L'option « -R » permet la sortie en mode caractères bruts et permet les séquences d'échappement de couleurs ANSI. Consultez `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

De bons éditeurs, tels que Vim et Emacs, traitent correctement les textes codés en UTF-8 et autres codages exotiques. Il est préconisé d'utiliser l'environnement X en utilisant les paramètres linguistiques UTF-8 et d'installer les programmes requis ainsi que les polices assorties. Les éditeurs permettent de régler le codage des fichiers indépendamment de celui de l'environnement X. Veuillez vous référer à leur documentation relative au texte multi-octets.

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 [9.3.11](#)). 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 Personnaliser vim

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

Essayez, par exemple, ce qui suit :

```
" -----
" Configuration des paramètres linguistiques
"
set nocompatible
set nopaste
set pastetoggle=<f2>
syn on
if $USER == "root"
    set nomodeline
    set noswapfile
else
    set modeline
    set swapfile
endif
" remplissage pour éviter que la ligne du dessus ne soit identifiée comme une ligne de mode ←
  (modeline)
" remplissage
" remplissage
```

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 une tâche d'administration quelconque.

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 [9.2.3](#) .

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.

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

Note

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

Note

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.

Note

« `[` » 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 builtins
$ 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) :

```
$ <nom-de-commande> --help
$ <nom-de-commande> -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>`, `;`, `&`, `(,)`).
-

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 :

- le gestionnaire graphique de session tel que `gdm3` réinitialise les variables d'environnement ;
- dans son code de démarrage, l'interpréteur de commandes réinitialise les variables d'environnement dans « `~/ .bash_profile` » et « `~/ .bashrc` ».

1.5.2 La variable « \$LANG »

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 « <code>fr</code> »
yy	codes de pays ISO 3166 (en majuscules), par exemple « <code>FR</code> »
zzzz	le jeu de caractères, toujours défini à « <code>UTF-8</code> »

Table 1.17 – Les trois parties des paramètres linguistiques

Pour les codes de langues et de pays, consultez la description dans « `info gettext` ».

Vous devriez toujours définir le codage des caractères d'un système Debian moderne en **UTF-8** à moins que vous ne vouliez spécifiquement en utiliser un ancien avec de bonnes raisons et les connaissances de bases associées.

Pour des informations détaillées sur la configuration des paramètres linguistiques, consultez Section [8.3](#).

Note

« `LANG=en_US` » n'est pas « `LANG=C` » ni « `LANG=en_US.UTF-8` ». C'est « `LANG=en_US.ISO-8859-1` » (consultez Section [8.3.1](#)).

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

Table 1.18 – Liste des recommandations de paramètres linguistiques

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

```
$ date
lundi 29 octobre 2012, 10:38:53 (UTC-0400)
$ LANG=en_US.UTF-8 date
Mon Oct 29 10:39:00 AST 2012
```

Ici, le programme `date(1)` est exécuté comme tâche au premier plan avec différentes valeurs de la variable d'environnement « `$LANG` ».

- 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=en_US.UTF-8
$ date
Mon Oct 29 10:40:50 AST 2012
```

Comme vous pouvez le voir ici, la sortie de la commande est affectée par la variable d'environnement afin que la sortie s'effectue en anglais. Si vous désirez que les sous-processus héritent de la variable d'environnement (par exemple en appelant un script shell), vous devrez alors l'**exporter** de la manière suivante :

```
$ export LANG
```

Note

Lorsque vous utilisez un émulateur de terminal classique, la variable d'environnement « `$LANG` » est en général réglée pour être **exportée** par l'environnement de bureau. De sorte que ce qui précède n'est pas vraiment un bon exemple pour tester l'effet de `export`.

ASTUCE

En déposant un signalement de bogue, c'est une bonne idée de lancer et de vérifier la commande « `LANG=en_US.UTF-8` » si vous utilisez un environnement autre que l'environnement anglais.

Consultez `locale(5)` et `locale(7)` pour les variables d'environnement « `$LANG` » et associées.

Note

Je vous recommande de configurer l'environnement du système en ne touchant qu'à la variable « `$LANG` » en laissant de côté les variables « `$LC_*` » à moins que ce ne soit absolument nécessaire.

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 ».

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 1.19 – Afficher les valeurs de la variable « \$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/ ».

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
lundi 29 octobre 2012, 10:34:38 (UTC-0400)
$ date -R
Mon, 29 Oct 2012 10:34:40 -0400
```

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**).

Essayez, par exemple, ce qui suit :

```
$ mkdir poubelle; cd poubelle; 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
```

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

```
$ echo .*[^.]*
.5.txt ..6.txt
$ echo [^1-3]*
4.h
$ cd ../; rm -rf poubelle
```

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 1.21 – 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.

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 (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 1.22 – Idiomes des commandes 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 `builtins(1)`.

Essayez, par exemple, ce qui suit :

```
$ </etc/motd visualisateur
```

```
$ visualisateur </etc/motd
```

```
$ visualisateur /etc/motd
```

```
$ cat /etc/motd | visualisateur
```

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 >toto
$ exec 3<toto 4>titi # ouvrir les fichiers
$ cat <&3 >&4        # rediriger stdin vers 3, stdout vers 4
$ exec 3<&- 4>&-    # fermer les fichiers
$ cat titi
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 1.23 – 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 [9.2.7](#).

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.

— Une expression rationnelle de base (« Basic regular expression - BRE ») est utilisée :

- `grep(1)` analyse la correspondance d'un texte avec des motifs ;
- `ed(1)` est un éditeur par ligne primitif ;
- `sed(1)` est un éditeur de flux ;
- `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 :

- `egrep(1)` fait correspondre du texte avec des motifs ;
- `awk(1)` effectue un traitement simple du texte ;
- `tcl(3tcl)` peut effectuer tous les traitements possibles du texte : voir `re_syntax(3)`. Souvent utilisé avec `tk(3tk)` ;
- `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/html/index.html` ».

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

L'ordre de tri et la plage d'une expression dépendent des paramètres linguistiques. Si vous désirez obtenir le comportement traditionnel d'une commande, utilisez alors **C** plutôt que **UTF-8** en faisant précéder la commande de « `LANG=C` » (consultez [Section 1.5.2](#) et [Section 8.3](#)).

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 « c »
<code>\c</code>	<code>\c</code>	correspond au caractère littéral « c » même si « c » 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 « abc... »
<code>[^abc...]</code>	<code>[^abc...]</code>	correspond à n'importe quel caractère sauf ceux se trouvant dans « abc... »
<code>r*</code>	<code>r*</code>	correspond à aucune ou plusieurs instances de l'expression rationnelle identifiée par « r »
<code>r\+</code>	<code>r+</code>	correspond à une ou plusieurs instances de l'expression rationnelle identifiée par « r »
<code>r\?</code>	<code>r?</code>	correspond à aucune ou une instance de l'expression rationnelle identifiée par « r »
<code>r1\ r2</code>	<code>r1 r2</code>	correspond à une instance de l'expression rationnelle identifiée par « r1 » ou « r2 ».
<code>\(r1\ r2\)</code>	<code>(r1 r2)</code>	correspond à une des expressions rationnelles identifiées par « r1 » ou « r2 » et la traite comme une expression rationnelle entre crochets

Table 1.24 – 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 [9.2.7](#).

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
&	ce à quoi correspond l'expression rationnelle (utilisez \& avec emacs)
\n	ce à quoi la nième expression rationnelle entre crochets correspond (« n » étant un nombre)

Table 1.25 – Expressions de remplacement

Pour les chaînes de remplacement en Perl « \$n » est utilisé à la place de « \n » et « & » n'a pas de signification particulière.

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

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 fichier <<EOF
,s/REGEX_SOURCE/TEXTE_DESTINATION/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/REGEX_SOURCE/TEXTE_DESTINATION/g' fichier
```

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/REGEX_SOURCE/TEXTE_DESTINATION/gc' '+w' '+q' fichier
```

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/REGEX_SOURCE/TEXT_DESTINATION/ge|update' '+q' fichier1 fichier2 fichier3
```

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/REGEX_SOURCE/TEXTE_DESTINATION/g;' fichier1 fichier2 fichier3
```

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                                # mois d'investiture
August
April
January
January
```



```

April
April
March
$ awk '($1=="Ian") { print }' <DPL          # DPL appelé Ian
Ian      Murdock      August 1993
Ian      Jackson      January 1998
$ awk '($2=="Perens") { print $3,$4 }' <DPL # quand Perens a été investi
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
... même sortie que le premier exemple Awk

```

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" # préserver l'ancienne valeur
$ IFS=":"
$ while read user password uid gid rest_of_line; do
    if [ "$user" = "osamu" ]; then
        echo "ID de $user est $uid"
    fi
done < /etc/passwd
ID de osamu est 1001
$ IFS="$oldIFS" # remet l'ancienne valeur

```

(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=":," # utiliser « : » et « , » pour IFS
$ echo IFS=$IFS, IFS="$IFS" # echo est une commande interne de Bash
IFS= , IFS=,
$ date -R # simplement une sortie de commande
Sat, 23 Aug 2003 08:30:15 +0200
$ echo $(date -R) # sous-shell --> entrée du shell principal
Sat 23 Aug 2003 08 30 36 +0200
$ unset IFS # réinitialiser IFS à sa valeur par défaut
$ 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.

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 9.3.9](#).

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](#)).

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 <commande></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 <motif_expression_rationnelle></code>	extraire du tube les lignes contenant <motif_expression_rationnelle>
<code> grep -v -e <motif_expression_rationnelle></code>	extraire du tube les lignes ne contenant pas <motif_expression_rationnelle>
<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 1.26 – Liste de parties de scripts pour enchaîner (piping) les commandes

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 `Stretch`.

[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](#).

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 55089 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-get(8)` pour toutes les opérations en ligne de commandes, y compris, l'installation et la suppression de paquets et `dist-upgrade`.
- `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 ésoériques peuvent être activées manuellement par l'administrateur du système. Cela peut créer des interférences avec les programmes génériques d'assistance à la configuration du système.
-

paquet	popcon	taille	description
apt	V:871, I:999	3531	Advanced Packaging Tool (APT) (« outil avancé de paquetage »), frontal de <code>dpkg</code> permettant les méthodes d'accès à l'archive « <code>http</code> », « <code>ftp</code> » et « <code>file</code> » (les commandes <code>apt-get</code> et <code>apt-cache</code> sont comprises)
aptitude	V:151, I:906	4452	gestionnaire de paquets interactif en mode terminal avec <code>aptitude</code> (8)
tasksel	V:39, I:972	374	outil de sélection de tâches pour l'installation sur un système Debian (frontal d'APT)
unattended-upgrades	V:132, I:302	244	paquet d'amélioration d'APT permettant une installation automatique des mises à niveau de sécurité
dselect	V:5, I:66	2485	gestionnaire de paquets en mode terminal (standard précédent, frontal d'APT et d'autres anciennes méthodes d'accès)
dpkg	V:933, I:999	6704	système de gestion des paquets pour Debian
synaptic	V:68, I:451	7793	gestionnaire de paquets graphique (interface graphique GNOME pour APT)
apt-utils	V:366, I:997	1099	Programmes utilitaires d'APT : <code>apt-extracttemplates</code> (1), <code>apt-ftparchive</code> (1) et <code>apt-sortpkgs</code> (1)
apt-listchanges	V:324, I:830	368	outil de notification des modifications d'un paquet
apt-listbugs	V:7, I:12	451	affiche la liste des bogues critiques avant chaque installation par APT
apt-file	V:12, I:78	82	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

En dépit de mes avertissements précédents, je sais que de nombreux lecteurs de ce document voudront utiliser les versions **testing** ou **unstable** de Debian comme système principal pour des **environnements de bureau autoadministrés**. Cela parce qu'elles fonctionnent très bien, sont fréquemment mises à jour et offrent les fonctionnalités les plus récentes.



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, par exemple le PC de votre belle-mère.

Cela ne demande rien d'autre que de définir la chaîne de la distribution dans « `/etc/apt/sources.list` » avec le nom de suite « **testing** » ou « **unstable** », ou le nom de code « **buster** » ou « **sid** ». Cela vous permettra de vivre **la vie des mises à jour éternelles**.

Il y a **beaucoup de plaisir** à prendre en utilisant **testing** ou **unstable** mais aussi quelques risques. Même si la version **unstable** du système Debian semble très stable la plupart du temps, il y a eu certains problèmes de paquets sur les versions **testing** et **unstable** et certains d'entre-eux n'ont pas été aisés à résoudre. Ce peut être **assez douloureux** pour vous. Quelquefois, vous pouvez avoir un paquet cassé ou une fonctionnalité manquante pendant quelques semaines.

Voici quelques idées pour vous assurer une récupération rapide et facile lors de bogues dans les paquets Debian :

- 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 ;
- Créez un environnement isolé (« **chroot** ») ou similaire et faites-y tourner à l'avance la dernière version du système (consultez Section [9.10](#)).

(si vous ne savez pas faire l'une quelconque de ces actions de précaution, vous n'êtes probablement pas prêt pour les versions **testing** et **unstable**).

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.

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** = **stretch** actuel :

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

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

Ici, j'ai tendance à utiliser le nom de code « **stretch** » 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 « **stretch** », 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 (stretch)	édition stable (Stretch)
http://deb.debian.org/debian/	testing (buster)	édition testing (Buster)
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/	stable/updates	mises à jour de sécurité pour la version stable (important)
http://security.debian.org/	testing/updates	mises à jour de sécurité pour la version testing (important)
http://deb.debian.org/debian/	stretch-updates	mises à jour compatibles pour le filtrage de spam, les clients de messagerie instantanée, etc. pour Stretch
http://deb.debian.org/debian/	stretch-backports	paquets plus récents rétroportés pour Stretch (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 [stretch-updates](https://stretch-updates.debian.org) 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	54202	conforme à DFSG sans dépendance vers non-free
contrib	342	conforme à DFSG mais avec des dépendances vers non-free
non-free	545	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 stretch	nom de code = stretch	nom de code = buster	nom de code = sid
après la diffusion de buster	nom de code = buster	nom de code = bullseye	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](#)) comme suit.

- 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 permettrons, 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 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.

Afin de respecter ces normes strictes du logiciel libre nécessaires pour `main`, Debian [démarque les paquets des programmes de Mozilla](#) comme Firefox, Thunderbird et Seamonkey en retirant leurs logos et certaines données artistiques, et les distribue respectivement sous les noms Iceweasel, Icedove et Iceape.

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 [9.9.6](#) 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 »)** (« 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 »)** (« `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 (« 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 (« 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 (« 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.Debian` » 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 <nom_paquet>
$ 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 \(BTS\)](https://bugs.debian.org/).

Rechercher sur [Google](https://www.google.com/) 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 décrirons ici les outils de base de gestion des paquets : `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-get` / `apt-cache` comparés 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-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-get dist-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.

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 `aptitude(8)`, `apt-get(8)` et `apt-cache(8)` :

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

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

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-get` et `apt-cache` plutôt que la commande `aptitude`.

Note

Puisqu'`apt-get` et `aptitude` partagent l'état de paquet installé automatiquement (consultez Section 2.5.5) après Lenny, vous pouvez mélanger ces outils sans trop de problème (consultez le [Bogue #594490](#)).

« `aptitude why <expression-rationnelle>` » peut afficher plus d'informations par « `aptitude -v why <expression_rationnelle>` ». On peut obtenir des informations similaires par « `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 passer en mode plein écran en pressant ensuite la touche « `e` » à l'invite de commande

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

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)`

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

ASTUCE

Le paquet `dselect` est encore disponible et était l'outil de gestion des paquets en mode plein écran préféré des versions précédentes.

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.

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 :

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.

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

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   libsmclient          -2220ko 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	état	description de la vue
Vue des paquets	Bon	consultez Tableau 2.10 (défaut)
Recommandations d'audit	Bon	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 »	Bon	liste des paquets sans regroupement par catégories (pour l'utilisation avec des expressions rationnelles)
Navigateur de Debtags	Très utilisable	liste des paquets classés selon leur entrée debtags
Navigateur par catégories	Obsolète	liste des paquets classés selon leur catégorie (utiliser plutôt Debtags Browser)

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.

- « ~P<term> » == « ~Dprovides:<term> »
- « ~C<term> » == « ~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/README »).

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 ».

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**.

description des règles étendues de correspondance	formules d'expressions rationnelles
correspond au nom du paquet	<code>~n<expression_rationnelle_nom></code>
correspond à la description	<code>~d<expression_rationnelle_description></code>
correspond au nom de la tâche	<code>~t<expression_rationnelle_tâche></code>
correspond à l'étiquette debtag	<code>~G<expression_rationnelle_debtag></code>
correspond au responsable du paquet	<code>~m<expression_rationnelle_responsable></code>
correspond à la section du paquet	<code>~s<expression_rationnelle_section></code>
correspond à la version du paquet	<code>~V<expression_rationnelle_version></code>
correspond à l'archive	<code>~A{stretch,buster,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 <type>	<code>~B<type></code>
correspond aux paquets filtrés par <motif> ayant une dépendance de <type>	<code>~D[<type>:]<motif></code>
correspond aux paquets filtrés par <motif> ayant une dépendance cassée de <type>	<code>~DB[<type>:]<motif></code>
correspond aux paquets vers lesquels le paquet filtré par <motif> déclare une dépendance de <type>	<code>~R[<type>:]<motif></code>
correspond aux paquets vers lesquels le paquet filtré par <motif> a une dépendance cassée de <type>	<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 <motif> 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

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

Table 2.12 – Fichiers journaux de l'activité des paquets

2.2.9 Journaux d'activité des paquets

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

En réalité, il n'est pas aussi facile de comprendre la signification de ces journaux. Consultez Section 9.2.10 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 « 1 » 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 « 1 ». 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-get update; apt-get dist-upgrade* ».

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

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 libglib2.0-0:amd64` » pour obtenir une liste du contenu du paquet `libglib2.0-0` 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.

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 <nom_paquet></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 pour le fichier dsc></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>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

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 9.9).
- Consultez Section 12.11 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 <http://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 « Release » de plus haut niveau, par exemple, « `http://deb.debian.org/debian/dists/unstable/Release` », 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 « Release » de plus haut niveau est vérifiée par une infrastructure cryptographique appelée [secure apt](#).

— Le fichier de signature chiffré « Release.gpg » est créé à partir du fichier « Release » 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 « Release.gpg » et de la clé publique de l'archive Debian de « /etc/apt/trusted.gpg », l'intégrité du fichier « Release » 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 « Release ». 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 « Release » 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 « Release » au niveau de l'archive sont utilisés pour la règle d'`apt_preferences(5)`.

Il y a des fichiers « Release » 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/binary-amd64/Release` » 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 (« dapper », « feisty », « gutsy », « hardy », « intrepid », ...) sont utilisés dans [l'archive Ubuntu](#).

Pour certaines archives, comme `experimental` et `stretch-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/main/binary-amd64/Release` » 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>.gz` » (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/archives` » 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)	<nom_paquet>_<epoch>:<version_amont>-<version_debian>-<architecture>.deb
Le paquet binaire de l'installateur de Debian (connu sous le nom udeb)	<nom_paquet>_<epoch>:<version_amont>-<version_debian>-<architecture>.udeb
paquet source (source amont)	<nom_paquet>_<epoch>:<version_amont>-<version_debian>.orig.tar.gz
Paquet source 1.0 (modifications Debian)	<nom_paquet>_<epoch>:<version_amont>-<version_debian>.diff.gz
Paquet source 3.0 (quilt) (modifications Debian)	<nom_paquet>_<epoch>:<version_amont>-<version_debian>.debian.tar.gz
paquet source (description)	<nom_paquet>_<epoch>:<version_amont>-<version_debian>.dsc

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	caractères utilisables (expression rationnelle)	existence
<nom_paquet>	[a-z,A-Z,0-9,.,+,-]+	nécessaire
<epoch>:	[0-9]+:	optionnel
<version_amont>	[a-z,A-Z,0-9,.,+,-,~]+	nécessaire
<version_debian>	[a-z,A-Z,0-9,.,+,-,~]+	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>.conffiles</code>	liste de 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>.config</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> command
<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(8)`. 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

Appuyez sur <Entrée> pour conserver la valeur par défaut[*] ou choisissez le numéro ↔
sélectionné :
```

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 <ancien_paquet>
```

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.

Vous pouvez trouver des copies de l'ancienne version du paquet `foo` sans bogue dans le répertoire de cache des paquets « `/var/cache/apt/archives/` ». (Si ce n'est pas le cas, vous pouvez télécharger l'archive depuis <http://snapshot.debian.net/> ou la copier depuis le cache des paquets d'une machine qui fonctionne).

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

```
# dpkg -i /chemin/vers/toto_<ancienne-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 /chemin/vers/toto_<ancienne-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.*` ».

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 /chemin/vers/ancien/système
```

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

```
# cd /chemin/vers/ancien/système/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 »
- Les paquets sélectionnés par le paquet de dépendances (par exemple, `python2.4` par `python`)
- 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 [stretch-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 [9.10](#)) afin de faire tourner des programmes en provenance d'archives différentes.

2.7.3 Ajuster la version candidate

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` »).

**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**

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/stretch-backports/main Packages** ».

2.7.4 Mises à jour et rétroportages

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

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/ stretch main contrib non-free
deb http://security.debian.org/ stretch/updates main contrib
deb http://deb.debian.org/debian/ stretch-updates main contrib non-free
deb http://deb.debian.org/debian/ stretch-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 **stretch-updates**.
- Seuls les paquets les plus anciens installés à partir de **stretch-backports** sont mis à niveau vers les plus récents à partir de **stretch-backports**.

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

```
$ sudo apt-get install -t stretch-backports <nom_paquet>
```

2.7.5 Blocage des paquets installés par « Recommends »

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: <paquet-1>
Pin: version *
Pin-Priority: -1
```

```
Package: <paquet-2>
Pin: version *
Pin-Priority: -1
```

2.7.6 Suivre testing avec quelques paquets d'unstable

Voici un exemple de technique d'**épinglage 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/updates 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'épinglage de `unstable` devient 990).

```
$ sudo apt-get install -t unstable <nom_paquet>
```

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: <paquet-1>
Pin: release a=unstable
Pin-Priority: 700
```

```
Package: <paquet-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-fr
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/updates main contrib
```

La valeur de Pin-Priority par défaut pour l'archive `experimental` est toujours de 1 ($\ll 100$) 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

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)



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/ buster main contrib non-free
```

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

```
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/changelog` » 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: extra
Section: admin
Description: Passer outre les dépendances de paquets.
 Ce paquet factice peut être utilisé pour créer des paquets Debian
 qui ne contiennent que des informations sur les dépendances.
```

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 toto
$ apt-get source toto
$ cd toto*
```

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 toto*.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 [6.10](#)) 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:1	4735	serveur proxy avec cache pour les fichiers de l'archive Debian (programme OCaml compilé)
apt-cacher	V:1, I:1	287	proxy avec cache pour les paquets et les fichiers source Debian (programme Perl)
apt-cacher-ng	V:4, I:5	1418	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 Petite archive publique de paquets

Voici un exemple de création d'une petite archive publique de paquets compatible avec le système moderne d'**APT sécurisée** (consultez Section [2.5.2](#)). Définissons un certain nombre de choses :

- Nom du compte : « `toto` »
- Nom de machine : « `www.example.com` »
- Paquets nécessaires : `apt-utils`, `gnupg`, et autres paquets
- URL : « `http://www.example.com/~toto/` » (→ « `/home/foo/public_html/index.html` »)
- Architecture des paquets : « `amd64` »

Créez une clé d'archive APT de Toto sur votre serveur en faisant ce qui suit :

```
$ ssh foo@www.example.com
$ gpg --gen-key
...
$ gpg -K
...
sec 1024D/3A3CB5A6 2008-08-14
uid      Toto (ARCHIVE KEY) <toto@www.example.com>
ssb 2048g/6856F4A7 2008-08-14
$ gpg --export -a 3A3CB5A6 >toto.public.key
```

Publiez le fichier contenant la clé de l'archive « `toto.public.key` » avec l'ID de clé « `3A3CB5A6` » pour toto

Créez une arborescence d'archive appelée « Origin: Toto » comme suit :


```

$ umask 022
$ mkdir -p ~/public_html/debian/pool/main
$ mkdir -p ~/public_html/debian/dists/unstable/main/binary-amd64
$ mkdir -p ~/public_html/debian/dists/unstable/main/source
$ cd ~/public_html/debian
$ cat > dists/unstable/main/binary-amd64/Release << EOF
Archive: unstable
Version: 4.0
Component: main
Origin: Foo
Label: Foo
Architecture: amd64
EOF
$ cat > dists/unstable/main/source/Release << EOF
Archive: unstable
Version: 4.0
Component: main
Origin: Foo
Label: Foo
Architecture: source
EOF
$ cat > aptftp.conf << EOF
APT::FTPArchive::Release {
    Origin "Foo";
    Label "Foo";
    Suite "unstable";
    Codename "sid";
    Architectures "amd64";
    Components "main";
    Description "Archive publique de Toto";
};
EOF
$ cat > aptgenerate.conf << EOF
Dir::ArchiveDir ".";
Dir::CacheDir ".";
TreeDefault::Directory "pool/";
TreeDefault::SrcDirectory "pool/";
Default::Packages::Extensions ".deb";
Default::Packages::Compress ". gzip bzip2";
Default::Sources::Compress "gzip bzip2";
Default::Contents::Compress "gzip bzip2";

BinDirectory "dists/unstable/main/binary-amd64" {
    Packages "dists/unstable/main/binary-amd64/Packages";
    Contents "dists/unstable/Contents-amd64";
    SrcPackages "dists/unstable/main/source/Sources";
};

Tree "dists/unstable" {
    Sections "main";
    Architectures "amd64 source";
};
EOF

```

Vous pouvez automatiser les mises à jour répétitives du contenu de l'archive APT sur votre serveur en configurant `dupload`.

Placez tous les fichiers de paquets dans « `~/toto/public_html/debian/pool/main/` » en exécutant « `dupload -t toto changes_file` » dans le client avec « `~/dupload.conf` » contenant ce qui suit :

```

$cfg{'toto'} = {
    fqdn => "www.example.com",
    method => "scpb",

```

```
incoming => "/home/toto/public_html/debian/pool/main",
# dinstall de ftp-master s'envoie lui-même des courriels
dinstall_runs => 1,
};

$cfg{'toto'}{postupload}{'changes'} = "
echo 'cd public_html/debian ;
apt-ftparchive generate -c=aptftp.conf aptgenerate.conf;
apt-ftparchive release -c=aptftp.conf dists/unstable >dists/unstable/Release ;
rm -f dists/unstable/Release.gpg ;
gpg -u 3A3CB5A6 -bao dists/unstable/Release.gpg dists/unstable/Release' |
ssh toto@www.example.com 2>/dev/null ;
echo 'Archive de paquets créée.'";
```

Le script automatique (« hook » **postupload** lancé par `dupload(1)` crée une archive mise à jour des fichiers lors de chaque dépôt (« upload »).

Vous pouvez ajouter cette petite archive publique à la ligne `apt` de votre système client par ce qui suit :

```
$ sudo bash
# echo "deb http://www.example.com/~foo/debian/ unstable main" \
  >> /etc/apt/sources.list
# apt-key add toto.public.key
```

ASTUCE

Si l'archive est située sur le système de fichiers local, vous pouvez utiliser à la place « `deb file:///home/toto/debian/...` ».

2.7.16 Enregistrer et copier la configuration du système

Vous pouvez faire une copie locale de l'état de la sélection des paquets et de `debconf` à l'aide des commandes suivantes :

```
# dpkg --get-selections '*' > selection.dpkg
# debconf-get-selections > selection.debconf
```

Ici, « `*` » fait que « `selection.dpkg` » comportera aussi les entrées de paquets à « purger ».

Vous pouvez transférer ces 2 fichiers vers un autre ordinateur, et les y installer avec :

```
# dselect update
# debconf-set-selections < myselection.debconf
# dpkg --set-selections < myselection.dpkg
# apt-get -u dselect-upgrade # ou dselect install
```

Si vous envisagez de gérer de nombreux serveurs dans une grappe avec pratiquement la même configuration, vous devriez envisager d'utiliser un paquet spécialisé tel que `fai` pour gérer l'ensemble du système.

2.7.17 Convertir ou installer un paquet binaire non Debian

`alien(1)` permet de convertir des paquets binaires fournis dans les fichiers de format `rpm` de Red Hat, `slp`, de Stampede, `tgz` de Slackware et `pkg` de Solaris sous forme de paquet `deb` de Debian. Si vous voulez utiliser des paquets d'une autre distribution Linux que celle sur laquelle vous avez installé sur votre système, vous pouvez utiliser `alien` pour les convertir depuis votre format de paquets préféré et les installer. `alien` prend aussi en charge les paquets `LSB`.

**AVERTISSEMENT**

alien(1) ne devrait pas être utilisé pour remplacer des paquets essentiels du système, tels que sysvinit, libc6, libpam-modules, etc. En pratique, alien(1) ne devrait être utilisé que pour des paquets **non libres** uniquement binaires qui soient conformes à LSB et liés statiquement. Pour les logiciels libres, vous devriez utiliser leur paquet source pour en faire de vrais paquets Debian.

2.7.18 Extraire un paquet sans dpkg

Le contenu du paquet « dpkg*.deb » peut être extrait sans utiliser dpkg(1) sur un environnement quelconque [UNIX-like](#) en utilisant les commandes standards ar(1) et tar(1).

```
# ar x /path/to/dpkg_<version>_<arch>.deb
# ls
total 24
-rw-r--r-- 1 bozo bozo 1320 2007-05-07 00:11 control.tar.gz
-rw-r--r-- 1 bozo bozo 12837 2007-05-07 00:11 data.tar.gz
-rw-r--r-- 1 bozo bozo 4 2007-05-07 00:11 debian-binary
# mkdir control
# mkdir data
# tar xvzf control.tar.gz -C control
# tar xvzf data.tar.gz -C data
```

Le contenu de l'autre paquet « *.deb » peut être extrait avec la commande dpkg-deb(1), elle-même extraite du paquet "dpkg*.deb" évoqué au paragraphe précédent ; ou en utilisant les outils standards ar(1) et son descendant GNU tar(1) avec les fonctions xz(1) de désarchivage/décompression, comme dans les exemples ci-dessus.

Vous pouvez aussi parcourir le contenu du paquet en utilisant la commande mc.

2.7.19 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-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/index.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 du nouveau responsable 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.

J'ai fait de mon mieux pour fournir un aperçu rapide des points-clés du système Debian et de sa configuration pour vous servir de référence en me basant sur mes propres connaissances actuelles et antérieures et celles des autres. Comme le système Debian est une cible changeante, la situation sur le système peut avoir changé. Avant de faire une quelconque modification au système, vous devrez vous référer à la dernière documentation de chacun des paquets.



AVERTISSEMENT

Ce chapitre est obsolète car il est basé sur Debian 7.0 (1Wheezy) sorti en 2013.

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.

Note

Pour les plateformes autres que les PC traditionnels tels que les systèmes SUN ou Macintosh, le BIOS sur la ROM et la partition sur le disque peuvent être assez différents ([Section 9.5.2](#)). Veuillez dans ce cas rechercher ailleurs une documentation spécifique à votre plateforme.

3.1.1 Étage 1 : le BIOS

Le **BIOS** est la première étape du processus d'amorçage, il est démarré par l'événement de mise sous tension. Le **BIOS**, qui est implanté sur une **mémoire à lecture seule (ROM)**, est exécuté depuis l'adresse mémoire particulière à laquelle le pointeur de programme du processeur est initialisé par l'événement de mise sous tension.

Ce BIOS effectue l'initialisation de base du matériel (**POST : autotest à la mise sous tension**) (« power on self test ») et passe le contrôle du système à l'étape suivante que vous lui indiquez. Le BIOS est habituellement fourni avec le matériel.

L'écran de démarrage du BIOS indique en général quelle(s) touche(s) presser pour entrer dans l'écran de configuration du BIOS afin de paramétrer son comportement. Des touches courantes sont F1, F2, F10, Échap, Ins et Suppr. Si l'écran de démarrage de votre BIOS est caché par un bel écran graphique, vous pouvez essayer de presser différentes touches comme Échap pour désactiver cet écran. Ces touches dépendent beaucoup du matériel.

L'emplacement du matériel et la priorité du code lancé par le BIOS peuvent être définis depuis l'écran de configuration du BIOS. Typiquement, quelques secteurs de début du premier périphérique sélectionné qui est trouvé (disque dur, disquette, CD-ROM, ...) sont chargés en mémoire et le code initial est exécuté. Ce code peut être l'un des suivants :

- code du chargeur initial ;
- code du noyau d'un OS de transition comme **FreeDOS** ;
- code du système d'exploitation cible s'il peut tenir dans cet espace réduit.

Typiquement, le système est amorcé depuis la partition spécifiée du disque dur primaire. Les deux premiers secteurs du disque dur d'un PC traditionnel contiennent l'**enregistrement maître d'amorçage (MBR)**. (« master boot record »). Les informations de partition du disque, y compris la sélection de l'amorçage, sont enregistrées à la fin de ce MBR. Le code initial du chargeur d'amorçage exécuté depuis le BIOS occupe le reste de ce MBR.

3.1.2 Étage 2 : le chargeur initial

Le **chargeur initial** (« boot loader ») est le deuxième étage du processus d'amorçage, il est lancé depuis le BIOS. Il charge en mémoire l'image du noyau du système et l'image **initrd** et leur passe le contrôle. Cette image mémoire initiale (« initrd image ») est l'image du système de fichiers racine et sa prise en compte dépend du chargeur initial utilisé.

Le système Debian utilise normalement le noyau de Linux comme noyau de système par défaut. L'image mémoire initiale de l'actuel noyau de Linux 2.6/3.x est techniquement l'image **initramfs** (initial RAM filesystem : système de fichiers initial en mémoire). L'image **initramfs** est une archive **cpio**, compressée à l'aide de **gzip**, des fichiers du système de fichiers racine.



AVERTISSEMENT

Ce qui précède n'est plus vrai avec les nouveaux multi-segments **initramfs**. Voir [Bug #790100](#).

L'installation par défaut du système Debian place, pour la plateforme PC, le premier étage du code du chargeur initial GRUB sur le **MBR**. Il existe de nombreux chargeurs d'amorçage et d'options de configuration possibles.



AVERTISSEMENT

Ne jouez pas avec les chargeurs d'amorçage sans avoir un support de secours amorçable (CD ou disquette) créé à partir des images du paquet **grub-rescue-pc**. Il vous permettra de démarrer votre système sans même avoir de chargeur initial fonctionnel sur le disque dur.

Le menu de configuration de GRUB Legacy est situé en « **/boot/grub/menu.lst** ». Il peut, par exemple, avoir les entrées suivantes :

title	Debian GNU/Linux
root	(hd0,2)
kernel	/vmlinuz root=/dev/hda3 ro
initrd	/initrd.img

paquet	popcon	taille	initrd	chargeur initial	description
grub-legacy	V:0, I:2	710	Pris en charge	GRUB Legacy	assez intelligent pour comprendre le partitionnement du disque et des systèmes de fichiers tels que vfat, ext3, ...
grub-pc	V:32, I:872	561	Pris en charge	GRUB 2	assez intelligent pour comprendre le partitionnement du disque et des systèmes de fichiers tels que vfat, ext4... (défaut).
grub-rescue-pc	V:0, I:1	5239	Pris en charge	GRUB 2	images de secours amorçables de GRUB 2 (CD et disquettes) (versions PC/BIOS)
lilo	V:0, I:4	694	Pris en charge	Lilo	réside sur les emplacements des secteurs de données du disque dur (Ancien).
syslinux	V:6, I:64	300	Pris en charge	Isolinux	il comprend le système de fichiers ISO9660. C'est utilisé pour le CD d'amorçage.
syslinux	V:6, I:64	300	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	84	Pris en charge	Loadlin	un nouveau système est démarré depuis le système FreeDOS/MSDOS.
mbr	V:1, I:11	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

Le fichier de menu de configuration de GRUB 2 est situé en « `/boot/grub/grub.cfg` ». Il est automatiquement créé par « `/usr/sbin/update-grub` » depuis les modèles situés dans « `/etc/grub.d/*` » et les paramètres situés en « `/etc/default/grub` ». Il peut, par exemple, avoir les entrées suivantes :

```
menuentry "Debian GNU/Linux" {
    set root=(hd0,3)
    linux /vmlinuz root=/dev/hda3
    initrd /initrd.img
}
```

Les paramètres de GRUB pour ces exemples ont la signification suivante :

Paramètres de GRUB	signification
<code>root</code>	utiliser la 3ème partition du premier disque en la désignant par « <code>(hd0,2)</code> » avec GRUB legacy ou par « <code>(hd0,3)</code> » dans GRUB 2
<code>kernel</code>	utiliser le noyau se trouvant en « <code>/vmlinuz</code> » avec le paramètre du noyau : « <code>root=/dev/hda3 ro</code> »
<code>initrd</code>	utiliser l'image initrd/initramfs située en « <code>/initrd.img</code> »

Table 3.2 – Signification des paramètres de GRUB

Note

La valeur du numéro de partition utilisé par le programme « GRUB legacy » est inférieure d'un unité à celle normalement utilisée par le noyau de Linux et les outils utilitaires. Le programme GRUB 2 corrige ce problème.

ASTUCE

Un **UUID** (consultez Section 9.5.3) peut être utilisé pour identifier un périphérique spécial par bloc plutôt que son nom de fichier comme « /dev/hda3 », par exemple, « root=UUID=81b289d5-4341-4003-9602-e254a17ac232 ro ».

ASTUCE

Si **GRUB** est le chargeur en usage, le paramètre de démarrage du noyau est spécifié dans /boot/grub/grub.cfg. Sur les systèmes Debian, il ne faut pas modifier soi-même /boot/grub/grub.cfg. Il faut modifier la valeur de GRUB_CMDLINE_LINUX_DEFAULT dans /etc/default/grub puis exécuter update-grub(8) qui se charge de la mise à jour de /boot/grub/grub.cfg.

ASTUCE

Vous pouvez démarrer un chargeur d'amorçage depuis un autre chargeur d'amorçage en utilisant une technique appelée **chargement en chaîne** (« chain loading »).

Consultez « info grub » et grub-install(8).

3.1.3 Étape 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**.

Le script « /init » est exécuté en tant que premier programme sur le système de fichiers racine en mémoire. C'est un script de l'interpréteur de commandes qui initialise le noyau dans l'espace utilisateur et passe le contrôle au reste de l'étape. Ce système mini-Debian offre au système d'amorçage une flexibilité comme l'ajout de modules du noyau avant le processus de démarrage principal ou le montage du système de fichiers racines en mode chiffré.

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).

**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` ».

Le programme d'initialisation par défaut a subi des changements :

- Les versions de Debian antérieures à **squeeze** utilisent une simple initialisation façon [SysV](#).
- Debian **wheezy** améliore le système d'init de style [SysV](#) en ordonnant la séquence de démarrage avec les en-têtes LSB et en lançant l'exécution parallèle de scripts de démarrage.
- Debian **jessie** change de système d'initialisation par défaut au profit de [systemd](#) pour une initialisation parallélisée et pilotée par événements.

ASTUCE

Tous les mécanismes de redémarrage sont compatibles par l'intermédiaire des scripts « `/etc/init.d/rc` », « `/etc/init.d/rcS` », « `/usr/sbin/update-rc.d` » et « `/usr/sbin/invoke-rc.d` ».

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
initscripts	V:847, I:956	208	scripts pour initialiser et arrêter le système
sysvinit-core	V:18, I:24	225	Utilitaires <code>init(8)</code> de type System-V
sysv-rc	V:847, I:956	123	Mécanisme de changement de niveau de fonctionnement de type System-V
sysvinit-utils	V:875, I:999	110	Utilitaires de type System-V (<code>startpar(8)</code> , <code>bootlogd(8)</code> , ...)
lsb-base	V:881, I:999	49	Linux Standard Base 3.2 fonctionnalité de script d'initialisation
insserv	V:792, I:936	140	outil pour organiser la séquence de démarrage en utilisant les dépendances du script LSB de <code>init.d</code>
systemd	V:648, I:730	9334	démon <code>init(8)</code> basé sur des événements pour la concurrence (remplacement de <code>sysvinit</code>)
uswsusp	V:4, I:13	699	outil permettant d'utiliser le programme de l'espace utilisateur de mise en veille fourni par Linux
kexec-tools	V:1, I:8	267	outil <code>kexec</code> pour le redémarrage par <code>kexec(8)</code> (redémarrage à chaud)
bootchart2	V:0, I:1	94	analyseur des performances du processus de démarrage
pybootchartgui	V:0, I:1	177	analyseur des performances du processus de démarrage (virtualisation)
mingetty	V:0, I:3	35	<code>getty(8)</code> en mode console uniquement
mgetty	V:0, I:1	302	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 Initialisation façon SysV



Attention

Le système Debian actuel par défaut n'utilise pas système d'initialisation (init) de type SysV. Veuillez lire d'autres documents sur le système d'initialisation systemd plus moderne. Voir [Le Guide de l'Administrateur Debian](#)

Cette section explique de quelle manière le système était anciennement démarré avec la bonne vieille initialisation façon SysV. Votre système Debian ne fonctionne pas exactement de cette manière mais il est très instructif de connaître ces bases, puisque les systèmes d'initialisation plus récents tendent à offrir des fonctionnalités équivalentes.

Le processus de démarrage façon SysV franchit essentiellement les étapes suivantes :

1. Le système Debian passe au **niveau de fonctionnement N** (« runlevel N ») (aucun) pour initialiser le système selon la description de « `/etc/inittab` ».
2. Le système Debian passe au **niveau de fonctionnement S** pour initialiser le système en mode utilisateur unique afin de terminer l'initialisation matérielle, etc.
3. Pour démarrer les services du système, le système Debian passe alors à l'un des **niveaux de fonctionnement multi-utilisateurs (2 à 5)** indiqués.

Le niveau de fonctionnement initial utilisé pour le mode multi-utilisateurs est indiqué par le paramètre de démarrage « `init=` » du noyau ou par la ligne « `initdefault` » du fichier « `/etc/inittab` ». Le système Debian tel qu'il est installé démarre au **niveau de fonctionnement 2**.

Tous les fichiers de scripts réels exécutés par le système init sont situés dans le répertoire « `/etc/init.d/` ».

Consultez `init(8)`, `inittab(5)`, et « `/usr/share/doc/sysv-rc/README.runlevels.gz` » pour une explication précise.

3.2.1 Signification du niveau de fonctionnement (« runlevel »)

Chaque [niveau de fonctionnement](#) utilise un répertoire pour sa configuration, il possède la signification particulière décrite ci-dessous :

niveau de fonctionnement	répertoire	description de l'utilisation du niveau de fonctionnement (runlevel)
N	aucun	niveau d'amorçage du système (NONE) (pas de répertoire « <code>/etc/rcN.d/</code> »)
0	<code>/etc/rc0.d/</code>	arrêter le système
S	<code>/etc/rcS.d/</code>	mode utilisateur unique lors du démarrage (alias : « S »)
1	<code>/etc/rc1.d/</code>	mode utilisateur unique commuté depuis le mode multi-utilisateurs
2	<code>/etc/rc2.d/</code>	mode multi-utilisateurs
3	<code>/etc/rc3.d/</code>	„
4	<code>/etc/rc4.d/</code>	„
5	<code>/etc/rc5.d/</code>	„
6	<code>/etc/rc6.d/</code>	redémarrer le système
7	<code>/etc/rc7.d/</code>	mode multi-utilisateurs valable mais normalement non utilisé
8	<code>/etc/rc8.d/</code>	„
9	<code>/etc/rc9.d/</code>	„

Table 3.4 – Liste des niveaux de fonctionnement et description de leur utilisation

Vous pouvez aussi changer de niveau de fonctionnement depuis la console, par exemple 4 en faisant ce qui suit :

```
$ sudo telinit 4
```

**Attention**

Le système Debian n'assigne pas par avance de différence spéciale de signification entre les **niveaux de fonctionnement** compris entre 2 et 5.. L'administrateur du système d'un système Debian peut modifier ce comportement. (C'est-à-dire que Debian n'est ni [Red Hat Linux](#), ni [Solaris de Sun Microsystems](#), ni [HP-UX de Hewlett Packard](#), ni [AIX d'IBM](#), ni...)

**Attention**

Le système Debian ne remplit pas les répertoires des **niveaux d'exécution** entre 7 et 9 lors de l'installation. Les **variantes d'UNIX** traditionnelles n'utilisent pas ces **niveaux d'exécution**.

3.2.2 Configuration des niveaux de fonctionnement (« runlevel »)

Lorsque la commande `init(8)` ou `telinit(8)` arrive au niveau « <n> », le système exécute essentiellement les scripts d'initialisation comme suit.

1. Les noms de scripts de « `/etc/rc<n>.d/` » commençant par un « K » sont exécutés dans l'ordre alphabétique avec le paramètre unique « `stop` » (arrêt des services).
2. Les noms de scripts de « `/etc/rc<n>.d/` » commençant par un « S » sont exécutés dans l'ordre alphabétique avec le paramètre unique « `start` » (lancement des services).

Par exemple, si vous avez les liens « `S10syslogd` » et « `S20exim4` » dans un répertoire de niveau de fonctionnement, « `S10syslogd` » qui est lié symboliquement à « `../init.d/syslogd` » sera lancé avant « `S20exim4` » qui est lié symboliquement à « `../init.d/exim4` ».

Ce simple système séquentiel d'initialisation est le système de démarrage classique de type [System V](#) tel qu'il a été utilisé jusqu'au système Debian Lenny.

Les systèmes Debian récents sont optimisés pour exécuter les scripts simultanément.

- La commande `insserv(8)` utilise les renseignements LSB de dépendances du paquet pour calculer les dépendances entre les scripts.
 - Consultez « `/usr/share/doc/insserv/README.Debian` ».
- Les renseignements LSB de dépendances du paquet définis dans l'en-tête des scripts d'initialisation assurent la disponibilité des ressources nécessaires.
 - Consultez la [spécification principale de Linux Standard Base 3.1, VII. Initialisation du système](#).
- [Annonce du « démarrage en parallèle activé par défaut »](#).

**AVERTISSEMENT**

Faire des modifications aux liens symboliques se trouvant dans « `/etc/rcS.d/` » est déconseillé à moins d'en savoir davantage que le responsable du paquet.

3.2.3 Exemple de gestion de niveau de fonctionnement

Définissons, par exemple, définissons un niveau de fonctionnement du système comme [Red Hat Linux](#) de la manière suivante :

- `init` démarre le système au niveau 3 qui est la valeur par défaut ;
- `init` ne démarre pas `gdm3(1)` aux niveaux 0,1,2,6 ;
- `init` démarre `gdm3(1)` aux niveaux 3,4,5 ;

Cela peut être fait en éditant le fichier « `/etc/inittab` » afin de modifier les niveaux de lancement et en utilisant des outils conviviaux de gestion des niveaux de fonctionnement tels que `sysv-rc-conf` ou `bum` pour éditer le niveau de fonctionnement. Si vous ne devez utiliser que la ligne de commande, voici comment faire (après l'installation par défaut du paquet `gdm3` et sa sélection comme gestionnaire d'affichage).

```
# cd /etc/rc2.d ; mv S21gdm3 K21gdm3
# cd /etc ; perl -i -p -e 's/^id:./id:3:/' inittab
```

Veuillez noter que le fichier « `/etc/X11/default-display-manager` » est vérifié lors du démarrage des démons de gestion d'affichage : `xdm`, `gdm3`, `sddm`, et `wdm`.

Note

Vous pouvez toujours lancer X depuis l'interpréteur de commandes de n'importe quelle console avec la commande `startx(1)`.

3.2.4 Paramètre par défaut de chaque script init

Le paramètre par défaut de chacun des scripts d'initialisation de « `/etc/init.d/` » est donné par le fichier correspondant se trouvant dans « `/etc/default/` » qui ne contient **que** des assignations de variables d'environnement. Ce choix de nom de répertoire est spécifique au système Debian. Il est en gros l'équivalent du répertoire « `/etc/sysconfig` » qu'on trouve dans [Red Hat Linux](#) et d'autres distributions. Par exemple, « `/etc/default/cron` » peut être utilisé pour contrôler la manière dont fonctionne « `/etc/init.d/cron` ».

Le fichier « `/etc/default/rcS` » peut être utilisé pour personnaliser les valeurs par défaut au moment du démarrage de `motd(5)`, `sulogin(8)`, etc.

Si vous ne pouvez pas obtenir le comportement que vous souhaitez en modifiant ces variables, alors vous pouvez [modifier les scripts init](#) eux-mêmes. Ce sont des fichiers de configuration qui peuvent être édités par les administrateurs du système.

3.2.5 Nom de machine (« `hostname` »)

Le noyau conserve le **nom de machine** du système. Le script d'initialisation du niveau de fonctionnement S qui est lié symboliquement à « `/etc/init.d/hostname.sh` » définit le nom de machine « `hostname` » au moment du démarrage (en utilisant la commande `hostname`) avec le nom qui est enregistré dans « `/etc/hostname` ». Ce fichier ne devra contenir **que** le nom de machine du système et pas un nom de domaine pleinement qualifié.

Pour afficher le nom de la machine utilisée, lancez la commande `hostname (1)` sans paramètre.

3.2.6 Le système de fichiers

Alors que le système de fichiers racine est monté par le noyau lors de son démarrage, les autres systèmes de fichiers sont montés dans le niveau de fonctionnement (runlevel) S à l'aide des scripts d'initialisation suivants :

- « `/etc/init.d/mountkernfs.sh` » pour les systèmes de fichiers du noyau se trouvant en « `/proc` », « `/sys` », etc.
- « `/etc/init.d/mountdevsubfs.sh` » pour les systèmes de fichiers virtuels de « `/dev` »
- « `/etc/init.d/mountall.sh` » pour les systèmes de fichiers normaux qui utilisent « `/etc/fstab` »
- « `/etc/init.d/mountnfs.sh` » pour les systèmes de fichiers à accès par le réseau en utilisant « `/etc/fstab` »

Les options de montage des systèmes de fichiers du noyau (`procfs`, `sysfs`, et `tmpfs` pour `/proc`, `/sys`, `/tmp`, `/run`, etc.) sont configurés dans « `/etc/default/rcS` ». Consultez `rcS(5)`.

Pour les systèmes de fichiers ordinaires, disque ou réseau, les options de montage sont définies dans « `/etc/fstab` ». Consultez [Section 9.5.7](#).

Note

Le montage effectif des systèmes de fichiers à accès par le réseau attendra le démarrage de l'interface réseau.

**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.

3.2.7 Initialisation de l'interface réseau

Les interfaces réseau sont initialisées dans le niveau de fonctionnement S par le script d'initialisation ayant un lien symbolique vers « /etc/init.d/ifupdown-clean » et « /etc/init.d/ifupdown ». Consultez Chapitre 5 pour la manière de les configurer.

3.2.8 Initialisation des services réseau

De nombreux services réseau (consultez Chapitre 6) sont directement démarrés en tant que démons sous le mode multi-utilisateurs lors du démarrage du système par le script d'initialisation, par exemple « /etc/rc2.d/S20exim4 » (pour RUNLEVEL=2) qui est un lien symbolique vers « /etc/init.d/exim4 ».

Certains services réseau peuvent être démarrés à la demande en utilisant le [super-serveur inetd](#) (ou un équivalent). `inetd` est lancé lors du démarrage du système par « /etc/rc2.d/S20inetd » (pour RUNLEVEL=2) qui est un lien symbolique vers « /etc/init.d/inetd ». Fondamentalement, `inetd` permet à un démon en cours de fonctionnement d'en appeler plusieurs autres, ce qui réduit la charge du système.

Lorsqu'une requête pour un service arrive au [super-serveur inetd](#), le protocole et le service sont identifiés par une recherche dans les bases de données se trouvant dans « /etc/protocols » et « /etc/services ». `inetd` recherche ensuite un service Internet normal dans la base de données « /etc/inetd.conf » ou un service basé sur [Open Network Computing Remote Procedure Call \(ONC RPC\)/Sun RPC](#) dans « /etc/rpc.conf ».

Parfois, `inetd` ne lance pas directement le serveur voulu mais exécute le [programme d'enveloppe du démon TCP/IP](#) `tcpd`(8) avec, en paramètre dans « /etc/inetd.conf », le service demandé. Dans ce cas, `tcpd` lance le serveur approprié après avoir enregistré la requête dans le journal et avoir fait quelques autres vérifications à l'aide de « /etc/hosts.deny » et « /etc/hosts.allow ».

Afin d'assurer la sécurité du système, désactivez autant de services réseau que possible. Consultez Section 4.6.4.

Consultez `inetd`(8), `inetd.conf`(5), `protocols`(5), `services`(5), `tcpd`(8), `hosts_access`(5), `hosts_options`(5), `rpcinfo`(8), `portmap`(8) et « /usr/share/doc/portmap/portmapper.txt.gz ».

3.2.9 Messages du système

Les messages du système peuvent être personnalisés au moyen de « /etc/default/rsyslog » et « /etc/rsyslog.conf » à la fois pour le fichier journal et pour l'affichage à l'écran. Consultez `rsyslogd`(8) et `rsyslog.conf`(5). Consultez aussi Section 9.2.2.

3.2.10 Messages du noyau

Les messages du noyau peuvent être personnalisés au moyen de « /etc/init.d/klogd » à la fois pour le fichier journal et l'affichage à l'écran. Positionnez « `KLOGD='-c 3'` » dans ce fichier et lancez « /etc/init.d/klogd restart ». Consultez `klogd`(8).

Vous pouvez modifier directement le niveau d'erreur des messages de la manière suivante :

```
# 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 3.5 – Liste des niveaux d'erreur du noyau

3.3 Le système udev

À partir de la version 2.6 du noyau Linux, [le système udev](#) fournit un mécanisme de découverte et d'initialisation automatique du matériel (consultez [udev\(7\)](#)). Lors de la découverte de chaque périphérique par le noyau, le système udev lance un processus utilisateur qui utilise les informations provenant du système de fichiers [sysfs](#) (consultez [Section 1.2.12](#)), charge les modules du noyau nécessaires pour sa prise en charge en utilisant le programme [modprobe\(8\)](#) (consultez [Section 3.3.1](#)) et crée les nœuds de périphériques en conséquence.

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.

Le nom des nœuds de périphériques peut être configuré par les fichiers de règle de udev se trouvant dans « `/etc/udev/rules.d/` ». Les règles actuelles par défaut tendent à créer des noms générés dynamiquement ce qui donne des noms de périphériques non statiques excepté pour les périphériques cd et réseau. En ajoutant vos règles personnalisées semblables à celles existantes pour les périphériques cd et réseau, vous pouvez aussi créer des noms de périphériques statiques pour les autres périphériques comme les clés USB. Consultez « [Écrire des règles udev](#) » ou « `/usr/share/doc/udev/writing_udev_rules/index.html` ».

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.

ASTUCE

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 9.5.3](#).

3.3.1 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 3.3](#)) 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 5.9](#)) ;
 - 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 [9.4.3](#).

ASTUCE

Vous pouvez configurer le matériel au moment du démarrage pour activer les fonctionnalités désirées de ce matériel. Consultez Section [9.4.4](#).

ASTUCE

Vous pouvez probablement ajouter la prise en charge d'un périphérique particulier en recompilant le noyau. Consultez Section [9.9](#).

Chapitre 4

Authentification

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.



AVERTISSEMENT

Ce chapitre est obsolète car il est basé sur Debian 7.0 (1Wheezy) sorti en 2013.

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 :

```
...
utilisateur1:x:1000:1000:Nom Utilisateur1,,,:/home/utilisateur1:/bin/bash
utilisateur2:x:1001:1001:Nom Utilisateur2,,,:/home/utilisateur2:/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 :

```
...
utilisateur1:$1$Xop0FYH9$IfxyQwBe9b8tiyIkt2P4F/:13262:0:99999:7:::
utilisateur2:$1$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:utilisateur1,utilisateur2
```

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 « <code><nom_utilisateur></code> »
<code>getent shadow <nom_utilisateur></code>	consulter les informations cachées du compte « <code><nom_utilisateur></code> »
<code>getent group <nom_groupe></code>	consulter les informations du groupe « <code><nom_groupe></code> »
<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

Sur les systèmes configurés avec PAM et NSS comme la machine [alioth](#) de Debian, le contenu des fichiers locaux « `/etc/passwd` », « `/etc/group` » et « `/etc/shadow` » peut ne pas être utilisé de manière active par le système. Les commandes ci-dessus restent valables même sous un tel environnement.

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:60, I:871	338	mkpasswd	frontal de la bibliothèque crypt(3) avec fonctionnalités surabondantes
openssl	V:785, I:988	1255	openssl passwd	calculer le hachage du mot de passe (OpenSSL). passwd(1ssl)

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:843, I:999	870	PAM (« Pluggable Authentication Modules »), Modules attachables d'authentification (service de base)
libpam-ldap	V:2, I:17	242	Modules attachables d'authentification permettant l'utilisation d'interfaces LDAP
libpam-cracklib	V:1, I:17	113	Modules attachables d'authentification permettant la prise en compte de cracklib
libpam-doc	I:2	1019	Modules attachables d'authentification (documentation en html et texte)
libc6	V:933, I:999	10679	bibliothèque GNU C Library : bibliothèques partagées qui fournissent aussi le service « Name Service Switch »
glibc-doc	I:14	3129	bibliothèque GNU C : pages de manuel
glibc-doc-reference	I:5	11934	bibliothèque GNU C : manuel de référence dans les formats info, pdf et html (non libre)
libnss-mdns	I:561	107	module NSS pour la résolution des noms DNS Multicast
libnss-ldap	I:15	255	module NSS pour l'utilisation de LDAP comme service de nommage
libnss-ldapd	I:20	143	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 NNS

- « 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.

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
/etc/pam.d/<nom_programme>	définir la configuration de PAM pour le « <nom_programme> », consultez <code>pam(7)</code> et <code>pam.d(5)</code>
/etc/nsswitch.conf	définir la configuration de NSS avec une entrée pour chaque service. Consultez <code>nsswitch.conf(5)</code>
/etc/nologin	limiter la connexion de l'utilisateur à l'aide du module <code>pam_nologin(8)</code>
/etc/securetty	limiter l'accès de l'administrateur à certains tty à l'aide du module <code>pam_securetty(8)</code>
/etc/security/access.conf	limiter les accès à l'aide du module <code>pam_access(8)</code>
/etc/security/group.conf	définir les limitations en fonction du groupe à l'aide du module <code>pam_group(8)</code>
/etc/security/pam_env.conf	définir des variables d'environnement à l'aide du module <code>pam_env(8)</code>
/etc/environment	définir des variables d'environnement supplémentaires à l'aide du module <code>pam_env(8)</code> avec le paramètre « <code>readenv=1</code> »
/etc/default/locale	définir les paramètres linguistiques (« locale ») à l'aide du module <code>pam_env(8)</code> avec le paramètre « <code>readenv=1 envfile=/etc/default/locale</code> » (Debian)
/etc/security/limits.conf	définir les limitations de ressources (<code>ulimit</code> , <code>core</code> , ...) à l'aide du module <code>pam_limits(8)</code>
/etc/security/time.conf	définir les restrictions de temps à l'aide du module <code>pam_time(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.secret` » 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 `libc-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 <http://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

L’installation du paquet `libpam-cracklib` vous permet de forcer des règles plus strictes concernant les mots de passe en mettant, par exemple, les lignes suivantes dans « `/etc/pam.d/common-password` » :

Pour `squeeze` :

```
password required pam_cracklib.so retry=3 minlen=9 difok=3
password [success=1 default=ignore] pam_unix.so use_authtok nullok md5
password requisite pam_deny.so
password required pam_permit.so
```

4.6 Autres contrôles d'accès

Note

Consultez Section 9.3.15 pour restreindre la fonctionnalité de [touche d'appel sécurisée \(SAK\)](#) (« [secure attention key](#) ») du noyau.

ASTUCE

Sous [systemd](#), [logind](#) peut être utilisé pour gérer les connexions des utilisateurs. Consultez [systemd-logind\(8\)](#).

4.6.1 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 <mon_nom> <un_fichier>
```

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](#) ».

4.6.2 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\(8\)](#).

4.6.3 SELinux

[Security-Enhanced Linux \(SELinux\)](#) est une infrastructure pour resserrer davantage le modèle des privilèges que le modèle de sécurité ordinaire des systèmes semblables à UNIX avec les règles de [contrôle d'accès obligatoire \(MAC\)](#) (« [mandatory access control](#) »). La puissance de l'administrateur peut être restreinte sous certaines conditions.

4.6.4 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>](#) » ;

- configuration du niveau de fonctionnement (« runlevel » du [démon](#) ;
- [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](#).

Consultez Section 3.2.3, Section 3.2.4, Section 4.5.1, Section 3.2.8 et Section 5.9.

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 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**.

4.7.1 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).

4.7.2 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 [6.9](#).

4.7.3 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 [5.9](#)) conjointement avec les outils sûrs qui suivent peut améliorer la sécurité :

paquet	popcon	taille	description
knockd	V:1, I:3	89	démon de serrure électronique IP knockd(1) et client konck(1)
fail2ban	V:96, I:108	1290	bannir les IP qui provoquent des erreurs d'authentification multiples
libpam-shield	V:0, I:0	107	verrouiller les attaquants distants cherchant à deviner le mot de passe

Table 4.8 – Liste des outils fournissant des mesures de sécurité supplémentaires

4.7.4 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 ;
- verrouiller le BIOS et interdire l'amorçage depuis un support amovible ;
- 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. Déplacer le disque dur sur un PC dont le BIOS permet le démarrage depuis un CD.
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 [9.4.11](#)). 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 [9.8](#)). Vous aurez alors toujours besoin d'un mot de passe pour vous connecter au système.

Chapitre 5

Configuration du réseau

ASTUCE

Pour un guide d'ordre général concernant le réseau sous GNU/Linux, lire le [Guide de l'administrateur réseau de Linux](#) (« Linux Network Administrators Guide »).

ASTUCE

Pour un guide actualisé de la gestion réseau sous Debian, lire [Le Guide de l'administrateur Debian —Configuration du réseau](#).



AVERTISSEMENT

Au lieu d'utiliser le schéma de nom d'interface réseau traditionnel ("eth0", "eth1", "wlan0", ...), le nouveau [systemd](#) utilise "[des Noms d'interface réseau prévisibles \(Predictable Network Interface Names\)](#)" semblable à "enp0s25".



AVERTISSEMENT

Ce chapitre est obsolète car il est basé sur Debian 7.0 (Wheezy) sorti en 2013.

ASTUCE

Bien que ce document utilise encore l'ancien `ifconfig(8)` avec IPv4 comme exemples de configuration réseau, Debian se dirige vers `ip(8)` avec IPv4 et IPv6 à partir de Wheezy. Les correctifs pour mettre à jour ce document sont les bienvenus.

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.

paquets	popcon	taille	type	description
ifupdown	V:561, I:995	195	config::ifupdown	outil standard pour activer ou désactiver le réseau (spécifique à Debian)
ifplugd	V:4, I:20	214	, ,	gérer automatiquement le réseau filaire
ifupdown-extra	V:0, I:1	96	, ,	script de test du réseau pour améliorer le paquet « ifupdown »
ifmetric	V:0, I:1	36	, ,	définir les métriques de routage d'une interface réseau
guessnet	V:0, I:0	422	, ,	script de cartographie réseau améliorant le paquet « ifupdown » par l'intermédiaire du fichier « <code>/etc/network/interfaces</code> »
ifscheme	V:0, I:0	58	, ,	script de cartographie réseau pour améliorer le paquet « ifupdown »
ifupdown-scripts-zg2	V:0, I:0	147	, ,	scripts d'interface de Zugschluss fournissant une méthode manuelle à ifupdown
network-manager	V:366, I:469	10172	config::NM	NetworkManager (démon) : gère automatiquement le réseau
network-manager-gnome	V:236, I:411	6281	, ,	NetworkManager (frontal de GNOME)
wicd	I:35	35	config::wicd	gestionnaire de réseau filaire ou sans fil (méta paquet)
wicd-cli	V:0, I:2	59	, ,	gestionnaire de réseau filaire ou sans fil (client en ligne de commandes)
wicd-curses	V:1, I:5	175	, ,	gestionnaire de réseau filaire ou sans fil (client Curses)
wicd-daemon	V:30, I:40	954	, ,	gestionnaire de réseau filaire ou sans fil (démon)
wicd-gtk	V:24, I:37	574	, ,	gestionnaire de réseau filaire ou sans fil (client GTK+)
iptables	V:240, I:993	1528	config::Netfilter	outils d'administration pour le filtrage des paquets et NAT (Netfilter)
iproute	V:107, I:475	23	config::iproute2	iproute2 , IPv6 et autres configurations avancées du réseau : ip(8), tc(8), etc
ifrename	V:0, I:2	122	, ,	renommer une interface réseau en fonction de divers critères statiques : ifrename(8)
ethtool	V:94, I:266	372	, ,	afficher ou modifier les paramètres d'un périphérique Ethernet
iputils-ping	V:257, I:997	111	test::iproute2	tester l'accessibilité d'une machine distante par nom de machine ou adresse IP (iproute2)
iputils-arping	V:66, I:405	58	, ,	tester l'accessibilité réseau d'une machine distante spécifiée par une adresse ARP
iputils-tracepath	V:14, I:264	75	, ,	tracer le chemin du réseau vers une machine distante
net-tools	V:395, I:976	940	config::net-tools	boîte à outils réseau NET-3 (net-tools , configuration réseau IPv4) : ifconfig(8), etc.
inetutils-ping	V:0, I:1	337	test::net-tools	tester l'accessibilité réseau d'une machine distante par nom de machine ou adresse IP (ancien, GNU)
arping	V:1, I:28	67	, ,	tester l'accessibilité réseau d'une machine distante spécifiée par une adresse ARP (ancien)
traceroute	V:68, I:992	154	, ,	tracer le chemin réseau vers une machine distante (ancien, console)
isc-dhcp-client	V:406, I:963	649	config::low-level	client DHCP
wpasupplicant	V:294, I:544	2528	, ,	client prenant en charge WPA et WPA2 (IEEE 802.11i)
wpaui	V:0, I:3	780	, ,	client graphique (Qt) pour wpa_supplicant
wireless-tools	V:94, I:274	286	, ,	outils pour manipuler les « Extensions Linux sans fil » (Linux Wireless Extensions)
ppp	V:186, I:515	927	, ,	connexion PPP/PPPoE avec chat
pppoeconf	V:0, I:11	290	config::helper	assistant de configuration d'une connexion PPPoE
pppconfig	V:1, I:3	805	, ,	assistant de configuration pour une connexion PPP avec chat
wvdial	V:0, I:7	276	, ,	assistant de configuration pour une connexion PPP avec wvdial et ppp
mtr-tiny	V:7, I:58	138	test::low-level	tracer le chemin réseau vers une machine distante (curses)
mtr	V:5, I:40	190	, ,	tracer le chemin réseau vers une machine distante (curses et GTK+)
gnome-nettool	V:11, I:257	2116	, ,	outils pour des opérations d'informations habituelles sur le réseau (GNOME)

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 <nom_hote>

# Les lignes suivantes servent pour les machines pouvant utiliser IPv6
::1      ip6-localhost ip6-loopback
fe00::0  ip6-localnet
ff00::0  ip6-mcastprefix
ff02::1  ip6-allnodes
ff02::2  ip6-allrouters
ff02::3  ip6-allhosts
```

Chaque ligne commence par une [adresse IP](#) et est suivie du [nom d'hôte](#) associé.

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

Le nom de l'interface réseau, par exemple `eth0`, est assigné dans le noyau Linux à chaque matériel par le mécanisme `udev` de configuration de l'espace utilisateur (consultez [Section 3.3](#)), lorsqu'il est trouvé. On appelle l'interface réseau **interface physique** (« `physical interface` ») dans `ifup(8)` et `interfaces(5)`.

De manière à ce que les interfaces réseau aient un nom homogène lors des redémarrages en utilisant l'[adresse MAC](#), etc., il existe un fichier de règles « `/etc/udev/rules.d/70-persistent-net.rules` ». Ce fichier est créé automatiquement par le programme « `/lib/udev/write_net_rules` », probablement lancé par le fichier de règles « `persistent-net-generator.rules` ». Vous pouvez le modifier pour changer la règle de nommage.



Attention

En éditant le fichier de règles « `/etc/udev/rules.d/70-persistent-net.rules` », vous devrez conserver chaque règle sur une seule ligne et l'[adresse MAC](#) en minuscule. Par exemple, si vous trouvez « `FireWire device` » et « `PCI device` » dans ce fichier, vous voudrez probablement nommer `eth0` le « `PCI device` » et le configurer comme interface réseau primaire.

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.

Classe	adresses de réseau	masque de réseau	masque de réseau /bits	nombre de sous-réseaux
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 [9.9.6](#).

5.2 Configuration moderne de réseau pour ordinateur de bureau

À partir de la version *squeeze*, les systèmes Debian peuvent gérer la connexion au réseau à l'aide de [démons](#) tels que [Network-Manager \(NM\)](#) (paquet `network-manager` et les paquets associés) ou [Wicd](#) (paquet `wicd` et les paquets associés).

- 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` ».

ASTUCE

Sous [systemd](#), le réseau peut aussi être configuré dans `/etc/systemd/network/`. Voir : `systemd-resolved(8)`, `resolved.conf(5)`, et `systemd-networkd(8)`.

Note

Certaines fonctionnalités de ces outils de configuration automatique du réseau peuvent souffrir de régressions. Ils ne sont pas aussi robustes que l'ancien paquet `ifupdown`. Consultez le [BTS de network-manager](#) et le [BTS de wicd](#) pour en savoir plus sur les problèmes et les limitations actuelles.

5.2.1 Outils graphiques de configuration du réseau

Les documentations officielles de NM et Wicd sous Debian sont fournies respectivement par « `/usr/share/doc/network-manager/README.Debian` » et « `/usr/share/doc/wicd/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 toto netdev
```

2. Gardez la configuration de « `/etc/network/interfaces` » aussi simple que possible comme ce qui suit :

```
auto lo
iface lo inet loopback
```

3. Redémarrez NM ou Wicd de la manière suivante :

```
$ sudo /etc/init.d/network-manager restart
```

```
$ sudo /etc/init.d/wicd restart
```

4. Configurez votre réseau à l'aide d'une interface graphique.

Note

Afin d'éviter les conflits avec `ifupdown`, seules les interfaces qui ne sont **pas** listées dans « `/etc/network/interfaces` » sont gérées par NM ou Wicd.

ASTUCE

Si vous désirez étendre les possibilités de configuration de NM, veuillez récupérer les modules d'extension appropriés et les paquets supplémentaires tels que `network-manager-openconnect`, `network-manager-openvpn-gnome`, `network-manager-pptp-gnome`, `mobile-broadband-provider-info`, `gnome-bluetooth`, etc. Il en va de même pour ceux de Wicd.

**Attention**

Ces outils de configuration automatique peuvent ne pas être compatibles avec des configurations ésothériques de l'ancien `ifupdown` dans « `/etc/network/interfaces` » telles que celles se trouvant en Section 5.5 et Section 5.6. Consultez le [BTS de network-manager](#) et le [BTS de wicd](#) pour les problèmes et les limitations actuels.

5.3 Ancienne méthode de configuration et de connexion réseau

Lorsque la méthode décrite dans Section 5.2 ne suffit pas à vos besoins, vous pouvez utiliser les anciennes connexion au réseau et méthode de configuration qui combinent de nombreux outils simples.

L'ancienne connexion au réseau est spécifique pour chacune des méthodes (consultez Section 5.4).

Il existe deux types de programmes de bas niveau pour la configuration du réseau sous un système Linux (consultez Section 5.7.1).

— Les programmes [net-tools](#) anciens (`ifconfig(8)`, ...) proviennent du système de réseau NET-3 de Linux. La plupart d'entre-eux sont aujourd'hui obsolètes.

— Les nouveaux programmes [Linux iproute2](#) (`ip(8)`, ...) représentent le système actuel de gestion de réseau sous Linux.

Bien que ces programmes de bas niveau soient puissants, ils sont lourds à utiliser. Des systèmes de haut niveau de configuration du système ont donc été créés.

Le paquet `ifupdown` est le standard de fait pour un tel système de configuration de haut niveau du réseau sur Debian. Il vous permet d'activer un réseau en utilisant simplement, par exemple, « `ifup eth0` ». Son fichier de configuration est le fichier « `/etc/network/interfaces` » et son contenu typique est le suivant :

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp
```

Le paquet `resolvconf` a été créé pour complément au système `ifupdown` afin de prendre en charge une reconfiguration en douceur de la résolution d'adresse du réseau en automatisant la réécriture du fichier de configuration de la résolution « `/etc/resolv.conf` ». Maintenant, la plupart des paquets de Debian servant à la configuration du réseau sont modifiés pour utiliser le paquet `resolvconf` (consultez « `/usr/share/doc/resolvconf/README.Debian` »).

Des scripts d'aide pour le paquet `ifupdown` comme `ifplugd`, `guessnet`, `ifscheme`, etc. ont été créés pour automatiser la configuration de l'environnement réseau comme dans le cas d'un PC mobile sur un réseau local câblé. Ils sont relativement difficiles à utiliser mais fonctionnent bien en conjonction avec un système `ifupdown` existant.

Cela est expliqué en détail et avec des exemples (consultez Section 5.5 et Section 5.6).

5.4 Méthode de connexion réseau (ancienne)



Attention

La méthode de test de connexion décrite dans cette section est destinée à des fins de test. Elle n'est pas destinée à être directement utilisée pour la connexion réseau de tous les jours. Vous êtes invité à utiliser à la place NM, Wicd ou le paquet `ifupdown` (consultez Section 5.2 et Section 5.5).

La connexion typique au réseau et le chemin de connexion pour un PC peuvent être résumés comme suit :

PC	méthode de connexion	chemin de connexion
Port série (<code>ppp0</code>)	PPP	modem POTS point d'accès réseau commuté FAI
Port Ethernet (<code>eth0</code>)	PPPoE/DHCP/Statique	BB-modem BB service BB access point FAI
Port Ethernet (<code>eth0</code>)	DHCP/Statique	LAN routeur BB avec traduction d'adresse (NAT) (modem BB ...)

Table 5.3 – Liste des méthodes de connexions réseau et des chemins de connexion

Voici un résumé des scripts de configuration de chaque méthode de connexion :

méthode de connexion	configuration	paquets(s) de support
PPP	<code>pppconfig</code> pour créer un chat déterministe	<code>pppconfig</code> , <code>ppp</code>
PPP (alternative)	<code>wvdialconf</code> pour créer un chat heuristique	<code>ppp</code> , <code>wvdial</code>
PPPoE	<code>pppoeconf</code> pour créer un chat déterministe	<code>pppoeconf</code> , <code>ppp</code>
DHCP	décrit dans « <code>/etc/dhcp/dhclient.conf</code> »	<code>isc-dhcp-client</code>
IP statique (IPv4)	décrit dans « <code>/etc/network/interfaces</code> »	<code>iproute</code> ou <code>net-tools</code> (obsolète)
IP statique (IPv6)	décrit dans « <code>/etc/network/interfaces</code> »	<code>iproute</code>

Table 5.4 – Liste des configurations de connexions réseau

Les acronymes de connexion au réseau ont la signification suivante :

Note

Les services de connexion au réseau étendu (WAN) par l'intermédiaire du câble du réseau de télévision sont en général servis par DHCP ou PPPoE. Ceux qui sont connectés en ADSL et FTTP sont en général servis par PPPoE. Vous devrez consulter votre FAI afin de connaître les exigences exactes de configuration de la connexion WAN.

Note

Lorsqu'un routeur large bande (« BB-router ») est utilisé pour créer un environnement de réseau local domestique, les PC sur le réseau local sont connectés au réseau étendu (WAN) par l'intermédiaire du routeur large bande par [traduction d'adresse réseau \(NAT\)](#). Dans ce cas, les interfaces réseau des PC sur le réseau local sont servies avec des adresses IP statiques ou par DHCP depuis le routeur large bande. Le routeur large bande, pour se connecter au réseau étendu, doit être configuré en suivant les instructions de votre FAI.

acronyme	signification
POTS	service téléphonique analogique classique (« plain old telephone service »)
BB	large bande
BB-service	par exemple, l'ADSL, la télévision par câble ou les services sur fibre optique (FTTP)
BB-modem	par exemple, le modem ADSL , le modem sur réseau câblé ou le terminal de réseau optique (ONT)
LAN	réseau local (« local area network »)
WAN	réseau étendu
DHCP	protocole de configuration dynamique des machines (« dynamic host configuration protocol »)
PPP	protocole point à point (« point to point protocol »)
PPPoE	protocole point à point avec encapsulation Ethernet
FAI (« ISP »)	fournisseur d'accès à Internet

Table 5.5 – Liste des acronymes de connexion au réseau

5.4.1 La connexion DHCP avec Ethernet

Le réseau moderne domestique ou pour une petite entreprise typique (réseau local, LAN) est connecté au réseau étendu (« WAN ») (Internet) en utilisant un routeur à large bande grand public. Le réseau local derrière ce routeur est habituellement configuré par un serveur de [protocole de configuration dynamique de l'hôte \(DHCP\)](#) (« dynamic host configuration protocol ») tournant sur le routeur.

Installez simplement le paquet `isc-dhcp-client` pour Ethernet desservi par le [protocole de configuration dynamique de l'hôte \(DHCP\)](#).

Consultez `dhclient.conf(5)`.

5.4.2 Connexion Ethernet avec une adresse IP statique

Il n'y a pas d'action spécifique nécessaire pour la configuration Ethernet avec une adresse IP statique.

5.4.3 La connexion PPP avec pppconfig

Le script de configuration `pppconfig` configure la connexion [PPP](#) de manière interactive en sélectionnant simplement les éléments suivants :

- numéro de téléphone ;
- identifiant chez le fournisseur d'accès (FAI) ;
- mot de passe chez le FAI ;
- vitesse du port ;
- port de communication avec le modem ;
- méthode d'authentification.



Attention

La valeur « `<nom_fai>` » du champ « provider » est supposée si les commandes `pon` et `poff` sont appelées sans paramètre.

Vous pouvez tester la configuration de la manière suivante, en utilisant des outils de configuration de réseau de bas niveau :

fichier	fonction
/etc/ppp/peers/<nom_fai>	fichier de configuration spécifique à <nom_fai> créé par <code>pppconfig</code> pour <code>pppd</code>
/etc/chatscripts/<nom_fai>	fichier de configuration spécifique à <nom_fai> créé par <code>pppconfig</code> pour <code>chat</code>
/etc/ppp/options	paramètres généraux d'exécution de <code>pppd</code>
/etc/ppp/pap-secret	données d'authentification pour PAP (risque de sécurité)
/etc/ppp/chap-secret	données d'authentification pour CHAP (plus sécurisé)

Table 5.6 – Liste des fichiers de configuration pour la connexion [PPP](#) avec `pppconfig`

```
$ sudo pon <nom_fai>
...
$ sudo poff <nom_fai>
```

Consultez « `/usr/share/doc/ppp/README.Debian.gz` ».

5.4.4 Autre connexion PPP avec `wvdialconf`

Une approche différente de l'utilisation de `pppd(8)` est de le lancer depuis `wvdial(1)` qui est fourni par le paquet `wvdial`. Plutôt que `pppd` faisant tourner `chat` (8) pour numéroter et négocier la connexion, `wvdial` effectue la numérotation et la négociation initiale puis démarre `pppd` pour effectuer le reste.

Le script de configuration `wvdialconf` configure la connexion PPP de manière interactive en sélectionnant simplement ce qui suit :

- numéro de téléphone ;
- identifiant chez le fournisseur d'accès (FAI) ;
- mot de passe chez le FAI ;

`wvdial` réussit à réaliser la connexion dans la plupart des cas et conserve automatiquement la liste des données d'authentification.

fichier	fonction
/etc/ppp/peers/wvdial	fichier de configuration spécifique à <code>wvdial</code> créé par <code>wvdialconf</code> pour <code>pppd</code>
/etc/wvdial.conf	fichier de configuration créé par <code>wvdialconf</code>
/etc/ppp/options	paramètres généraux d'exécution de <code>pppd</code>
/etc/ppp/pap-secret	données d'authentification pour PAP (risque de sécurité)
/etc/ppp/chap-secret	données d'authentification pour CHAP (plus sécurisé)

Table 5.7 – Liste des fichiers de configuration pour une connexion PPP avec `wvdialconf`

Vous pouvez tester la configuration de la manière suivante, en utilisant des outils de configuration de réseau de bas niveau :

```
$ sudo wvdial
...
$ sudo killall wvdial
```

Consultez `wvdial(1)` et `wvdial.conf(5)`.

5.4.5 La connexion PPPoE avec pppoeconf

Lorsque votre FAI vous propose une connexion avec PPPoE et que vous décidez de connecter directement votre PC au WAN, le réseau de votre PC doit être configuré avec PPPoE. PPPoE signifie « PPP over Ethernet » (PPP encapsulé par Ethernet). Le script de configuration `pppoeconf` configure la connexion PPPoE de manière interactive.

Les fichiers de configuration sont les suivants :

fichier	fonction
<code>/etc/ppp/peers/fournisseur-accès-adsl</code>	fichier de configuration spécifique à pppoe créé par <code>pppoeconf</code> pour <code>pppd</code>
<code>/etc/ppp/options</code>	paramètres généraux d'exécution de <code>pppd</code>
<code>/etc/ppp/pap-secret</code>	données d'authentification pour PAP (risque de sécurité)
<code>/etc/ppp/chap-secret</code>	données d'authentification pour CHAP (plus sécurisé)

Table 5.8 – Liste des fichiers de configuration pour une connexion PPPoE avec `pppoeconf`

Vous pouvez tester la configuration de la manière suivante, en utilisant des outils de configuration de réseau de bas niveau :

```
$ sudo /sbin/ifconfig eth0 up
$ sudo pon fournisseur-accès-adsl
...
$ sudo poff fournisseur-accès-adsl
$ sudo /sbin/ifconfig eth0 down
```

Consultez « `/usr/share/doc/pppoeconf/README.Debian` ».

5.5 Configuration de base du réseau avec ifupdown (ancienne)

La configuration traditionnelle du [réseau TCP/IP](#) sur un système Debian utilise le paquet `ifupdown` comme outil de haut niveau. Il y a deux cas typiques :

- Pour les systèmes utilisant une **adresse IP dynamique** tels que les PC portables, vous devrez configurer le réseau TCP/IP **avec** le paquet `resolvconf` et ce qui vous permettra changer facilement la configuration de votre réseau (consultez Section [5.5.4](#)).
- Pour les systèmes ayant une **adresse IP statique** tels que les serveurs, vous devrez configurer le réseau TCP/IP **sans** le paquet `resolvconf` et garder un système simple (see Section [5.5.5](#)).

Ces méthodes traditionnelles de configuration sont assez utiles si vous désirez définir une configuration avancée ; plus de détails dans ce qui suit :

Le paquet `ifupdown` fournit l'ossature standardisée pour la configuration du réseau de haut niveau sur un système Debian. Dans cette section, nous apprenons les bases de la configuration du réseau avec `ifupdown` avec une introduction simplifiée et de nombreux exemples typiques.

5.5.1 La syntaxe de commande simplifiée

Le paquet `ifupdown` contient 2 commandes : `ifup(8)` et `ifdown(8)`. Elles offrent une configuration de réseau de haut niveau à l'aide du fichier de configuration « `/etc/network/interfaces` ».



AVERTISSEMENT

Ne pas utiliser les outils de configuration de bas niveau tels que `ifconfig(8)` ni les commandes `ip(8)` pour configurer une interface se trouvant dans l'état **actif** (« up »).

Note

Il n'y a pas de commande `ifupdown`.

commande	action
ifup eth0	activer l'interface réseau eth0 avec la configuration eth0 si l'entrée « iface eth0 » existe
ifdown eth0	désactiver l'interface réseau eth0 avec la configuration eth0 si l'entrée « iface eth0 »

Table 5.9 – Liste des commandes de base de configuration du réseau avec ifupdown

5.5.2 Syntaxe de base de « /etc/network/interfaces »

La syntaxe clé de « /etc/network/interfaces » telle qu'expliquée dans interfaces(5) peut être résumée comme suit :

entrée	signification
« auto <nom_interface> »	démarrer l'interface <nom_interface> lors du démarrage du système
« allow-auto <nom_interface> »	, ,
« allow-hotplug <nom_interface> »	démarrer l'interface <nom_interface> lorsque le noyau détecte un événement « à chaud » depuis cette interface
Les lignes qui commencent par « iface <nom_config> ... »	définissent la configuration de réseau <nom_config>
Les lignes qui commencent par « mapping <nom_interface_glob> »	définissent une valeur de correspondance de <nom_config> pour l'interface correspondant à <nom_interface>
Une ligne commençant par le signe « # »	est traitée comme commentaire et ignorée (les commentaires de fin de ligne ne sont pas pris en charge)
Une barre oblique inversée (« \ », « back slash » en anglais) en fin de ligne	étend la configuration à la ligne suivante

Table 5.10 – Liste des entrées de « /etc/network/interfaces »

Les lignes commençant par l'entrée **iface** ont la syntaxe suivante :

```
iface <nom_config> <famille_adresse> <nom_méthode>
<option1> <valeur1>
<option2> <valeur2>
...
```

Pour la configuration de base, l'entrée de **mapping** n'est pas utilisée si vous utilisez le nom de l'interface réseau comme nom de configuration du réseau (consultez Section 5.6.5).



AVERTISSEMENT

Ne pas définir dans « /etc/network/interfaces » d'entrée « iface » en double pour une interface réseau

5.5.3 L'interface réseau de rebouclage (« loopback »)

L'entrée de configuration suivante du fichier « /etc/network/interfaces » active l'interface réseau « loopback » 10 lors du démarrage du système (grâce à l'entrée **auto**) :

```
auto lo
iface lo inet loopback
```

Elle existe toujours dans le fichier « /etc/network/interfaces ».

5.5.4 Interfaces réseau gérées par DHCP

Après avoir préparé le système selon Section 5.4.1, l'interface réseau gérée par le DHCP est configurée en créant l'entrée de configuration suivante dans le fichier « `/etc/network/interfaces` » :

```
allow-hotplug eth0
iface eth0 inet dhcp
```

Lorsque le noyau de Linux détecte l'interface physique `eth0`, l'entrée **allow-hotplug** permet à `ifup` d'activer l'interface et l'entrée **iface** demande à `ifup` d'utiliser DHCP pour configurer l'interface.

5.5.5 Interface réseau avec une adresse IP fixe

Une interface réseau avec une adresse IP fixe est configurée en créant de la manière suivante une entrée de configuration du fichier « `/etc/network/interfaces` » :

```
allow-hotplug eth0
iface eth0 inet static
    address 192.168.11.100
    netmask 255.255.255.0
    gateway 192.168.11.1
    dns-domain example.com
    dns-nameservers 192.168.11.1
```

Lorsque le noyau de Linux détecte l'interface physique `eth0`, l'entrée **allow-hotplug** permet à `ifup` d'activer l'interface et l'entrée **iface** permet à `ifup` d'utiliser l'IP statique pour la configurer.

Ici, je suppose ce qui suit :

- plage d'adresses IP du réseau local : `192.168.11.0 - 192.168.11.255`
- adresse IP de la passerelle : `192.168.11.1`
- adresse IP du PC : `192.168.11.100`
- paquet `resolvconf` : installé
- nom de domaine : « `example.com` »
- adresse IP address du serveur DNS : `192.168.11.1`

Si le paquet `resolvconf` n'est pas installé, vous devrez faire vous-même la configuration associée au DNS en éditant le fichier « `/etc/resolv.conf` » comme suit :

```
nameserver 192.168.11.1
domain example.com
```



Attention

Les adresses IP utilisées dans l'exemple ci-dessus ne sont pas destinées à être recopiées littéralement. Vous devrez adapter les adresses IP à la configuration réelle de votre réseau.

5.5.6 Les bases de l'interface réseau sans fil

Le **réseau local sans fil** (« wireless LAN ou WLAN ») permet une connexion sans fil à haute vitesse par une communication en spectre étalé « spread-spectrum » sur des bandes radio ne nécessitant pas de licence en utilisant un ensemble de normes appelées [IEEE 802.11](#).

Les interfaces WLAN se comportent sensiblement de la même manière que les interfaces Ethernet normales mais elles demandent un identifiant de réseau et des données de clé de chiffrement lors de leur initialisation. Leurs outils réseau de haut niveau sont exactement les mêmes que pour les interfaces Ethernet mais les noms des interfaces sont un peu différents, comme `eth1`, `wlan0`, `ath0`, `wifi0`, ...selon les pilotes du noyau utilisés.

ASTUCE

Le périphérique `wmaster0` est le périphérique-maître interne utilisé uniquement par [SoftMAC](#) avec la nouvelle [API mac80211 de Linux](#).

Voici quelques mot-clés à se souvenir pour le WLAN :

acronyme	en entier	signification
NWID	ID du réseau	ID du réseau sur 16 bits utilisées par les réseaux WaveLAN pre-802.11 (complètement dépassé)
(E)SSID	Service Set Identifier (étendu)	nom de réseau des Points d'accès sans fil (AP) interconnectés pour former un réseau local sans fil 802.11 , identifiant de domaine
WEP, (WEP2)	Wired Equivalent Privacy (confidentialité équivalente à un réseau câblé)	première génération de norme de chiffrement sans fil sur 64 bits (128 bits) avec une clé sur 40 bits (dépassé)
WPA	Wi-Fi Protected Access (Accès Wi-Fi protégé)	seconde génération de norme de chiffrement sans fil (la plus grande partie de 802.11i), compatible avec WEP
WPA2	Wi-Fi Protected Access 2 (Accès protégé Wi-Fi 2)	troisième génération de norme de chiffrement sans fil (802.11i complète), non compatible avec WEP

Table 5.11 – Liste d'acronymes pour le WLAN

Le choix du protocole est normalement restreint par le routeur sans fil déployé.

5.5.7 L'interface de réseau local sans fil avec WPA/WPA2

Vous devrez installer le paquet `wpa_supplicant` afin de prendre en compte le WLAN avec les nouveaux protocoles WPA/WPA2. Dans le cas d'IP fournies par [DHCP](#) sur une connexion WLAN, l'entrée du fichier « `/etc/network/interfaces` » doit être similaire à ce qui suit :

```
allow-hotplug ath0
iface ath0 inet dhcp
    wpa-ssid zonemaison
    # la clé psk hexadécimale est encodée depuis une phrase de passe en texte clair
    wpa-psk 000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f
```

Consultez « `/usr/share/doc/wpa_supplicant/README.modes.gz` ».

5.5.8 L'interface réseau local sans fil avec WEP

Vous devrez installer le paquet `wireless-tools` pour prendre en charge le WLAN avec l'ancien protocole WEP. (Votre routeur grand public peut encore utiliser cette infrastructure non sûre mais c'est mieux que rien).



Attention

Remarquez que le trafic réseau sur votre WLAN avec WEP peut être espionné par d'autres.

Dans le cas d'IP fournies par [DHCP](#) sur une connexion WLAN, l'entrée du fichier « `/etc/network/interfaces` » doit être similaire à ce qui suit :

```
allow-hotplug eth0
iface eth0 inet dhcp
    wireless-essid Maison
    wireless-key1 0123-4567-89ab-cdef
    wireless-key2 12345678
    wireless-key3 s:mot_de_passe
    wireless-defaultkey 2
    wireless-keymode open
```

Consultez « `/usr/share/doc/wireless-tools/README.Debian` ».

5.5.9 La connexion PPP

Vous devrez d’abord configurer la connexion PPP comme décrit précédemment (consultez Section 5.4.3). Ensuite, ajoutez dans le fichier « `/etc/network/interfaces` » une entrée pour le périphérique PPP primaire `ppp0` comme suit :

```
iface ppp0 inet ppp
    provider <nom_fai>
```

5.5.10 La connexion alternative PPP

Vous devrez d’abord configurer la connexion PPP alternative avec `wvdial` comme décrit précédemment (consultez Section 5.4.4). Ajoutez ensuite dans le fichier « `/etc/network/interfaces` » une entrée pour le périphérique primaire PPP `ppp0` comme suit :

```
iface ppp0 inet wvdial
```

5.5.11 La connexion PPPoE

Pour le PC directement connecté au WAN et utilisant PPPoE, vous devez configurer votre système pour une connexion PPPoE comme cela a été décrit précédemment (consultez Section 5.4.5). Ajoutez ensuite dans le fichier « `/etc/network/interfaces` » une entrée pour le premier périphérique PPPoE `eth0` comme suit :

```
allow-hotplug eth0
iface eth0 inet manual
    pre-up /sbin/ifconfig eth0 up
    up ifup ppp0=dsl
    down ifdown ppp0=dsl
    post-down /sbin/ifconfig eth0 down
# Ce qui suit est utilisé uniquement de manière interne
iface dsl inet ppp
    provider fournisseur-internet
```

5.5.12 État de configuration réseau de ifupdown

Le fichier « `/etc/network/run/ifstate` » enregistre l’état **désiré** de la connexion réseau pour toutes les interfaces réseau actuellement actives et gérées par le paquet `ifupdown`. Malheureusement, même si le système `ifupdown` échoue dans l’activation de l’interface comme désiré, le fichier « `/etc/network/run/ifstate` » l’affichera comme active.

Si la sortie de la commande `ifconfig(8)` d’une interface n’a pas de ligne telle que la suivante, elle ne peut pas être utilisée en tant qu’élément d’un [réseau IPv4](#).

```
inet addr:192.168.11.2 Bcast:192.168.11.255 Mask:255.255.255.0
```

Note

Pour un périphérique Ethernet connecté en PPPoE, la sortie de la commande `ifconfig(8)` ne produit pas de ligne semblable à celle de l'exemple ci-dessus.

5.5.13 Configuration réseau de base

Lorsque vous tentez de reconfigurer une interface, par exemple `eth0`, vous devez d'abord la désactiver par la commande « **sudo ifdown eth0** ». Cela supprime l'entrée `eth0` du fichier « `/etc/network/run/ifstate` » (quelques messages d'erreur peuvent être émis si `eth0` n'est pas active ou a été mal configurée auparavant. Jusqu'à présent, il semble qu'il soit sans danger de le faire n'importe quand sur une simple station de travail n'ayant qu'un seul utilisateur).

Vous pouvez maintenant réécrire le contenu du fichier « `/etc/network/interfaces` » selon vos besoins pour reconfigurer l'interface réseau `eth0`.

Ensuite, vous pourrez réactiver `eth0` avec la commande **sudo ifup eth0** ».

ASTUCE

Vous pouvez (ré)initialiser l'interface réseau simplement par « **sudo ifdown eth0;sudo ifup eth0** ».

5.5.14 La paquet ifupdown-extra

Le paquet `ifupdown-extra` fournit des tests de connexion au réseau faciles à utiliser avec le paquet `ifupdown`.

— La commande `network-test(1)` peut être utilisée depuis l'interpréteur de commandes.

— Les scripts automatiques sont lancés pour chaque exécution de la commande `ifup`.

La commande `network-test` vous libère de l'exécution d'ennuyeuses commandes de bas niveau pour analyser les problèmes du réseau.

Les scripts automatiques sont installés dans « `/etc/network/*/` » et effectuent ce qui suit :

- vérifier la connexion du câble réseau
- vérifier l'utilisation d'une adresse IP dupliquée
- définir les routes statiques du système basées sur la définition de « `/etc/network/routes` »
- vérifier si la passerelle réseau peut être atteinte
- enregistrer les résultats dans le fichier « `/var/log/syslog` »

L'enregistrement dans `syslog` est assez utile pour l'administration des problèmes de réseau sur un système distant.

ASTUCE

Le comportement automatique du paquet `ifupdown-extra` est configurable par « `/etc/default/network-test` ». Certaines de ces vérifications automatiques ralentissent un peu le démarrage du système car elles prennent un certain temps à l'écoute des réponses [ARP](#).

5.6 Configuration réseau avancée avec ifupdown (ancienne)

Les fonctionnalités du paquet `ifupdown` peuvent être améliorées au-delà de ce qui est décrit dans Section [5.5](#) avec des connaissances avancées.

Les fonctionnalités décrites ici sont entièrement facultatives. Étant paresseux et minimaliste, je me suis rarement ennuyé à les utiliser.

**Attention**

Si vous n'êtes pas arrivé à établir une connexion réseau à l'aide des informations de Section [5.5](#), vous ne ferez qu'empirer la situation en utilisant les informations qui suivent :

5.6.1 La paquet ifplugd

Le paquet `ifplugd` est un ancien outil de configuration automatique de réseau qui ne peut gérer que les connexions Ethernet. Il permet de régler le problème des connexions et déconnexions des câbles Ethernet avec les PC mobiles, etc. Si vous avez installé [NetworkManager](#) ou [Wicd](#) (consultez Section 5.2), vous n'avez pas besoin de ce paquet.

Ce paquet fait tourner un [démon](#) qui remplace les fonctionnalités `auto` ou `allow-hotplug` (consultez Tableau 5.10) et qui active les interfaces lors de leur connexion au réseau.

Voici comment utiliser le paquet `ifplugd` pour le port Ethernet interne, par exemple, `eth0`.

1. Supprimez l'entrée « `auto eth0` » ou « `allow-hotplug eth0` » de « `/etc/network/interfaces` »
2. Conservez les entrées « `iface eth0 inet ...` » et « `mapping ...` » de « `/etc/network/interfaces` »
3. Installez le paquet `ifplugd`.
4. Lancez « `sudo dpkg-reconfigure ifplugd` »
5. Mettez `eth0` comme interface « statique devant être surveillée par `ifplugd` ».

Le réseau fonctionne maintenant comme vous le désirez.

- Lors de la mise sous tension ou lors de la découverte du matériel, l'interface n'est pas activée par elle-même.
 - Processus de démarrage rapide sans le long délai de dépassement de temps de DHCP.
 - Pas d'interface bizarre activée sans une adresse IPV4 correcte (consultez Section 5.5.12).
- L'interface est activée lors de la découverte du câble Ethernet.
- L'interface est désactivée quelque temps après que le câble Ethernet ait été débranché.
- L'interface est activée dans le nouvel environnement de réseau lors de la connexion d'un autre câble Ethernet.

ASTUCE

Les paramètres de la commande `ifplugd(8)` peuvent définir son comportement tel que le délai de reconfiguration des interfaces.

5.6.2 Le paquet ifmetric

Le paquet `ifmetric` permet de manipuler a posteriori la métrique de routes même avec DHCP.

Ce qui suit permettra de rendre l'interface `eth0` prioritaire sur l'interface `wlan0` :

1. installer le paquet `ifmetric` :
2. ajouter l'option avec « `metric 0` » juste en-dessous de la ligne « `iface eth0 inet dhcp` » dans « `/etc/network/interfaces` ».
3. ajouter une ligne d'option avec « `metric 1` » juste en-dessous de la ligne « `iface wlan0 inet dhcp` » dans « `/etc/network/interfaces` ».

« `metric 0` » indique la route de plus haute priorité et c'est celle par défaut. La plus grande valeur de « `metric` » indique une route de moindre priorité. L'adresse IP de l'interface active ayant la valeur de « `metric` » la plus basse devient celle d'origine. Consultez `ifmetric(8)`.

5.6.3 L'interface virtuelle

Une seule interface Ethernet physique peut être configurée en tant que de multiples interfaces virtuelles avec différentes adresses IP. Habituellement, le but est de connecter une interface à plusieurs sous-réseaux ayant des IP différentes. Par exemple, l'hébergement web virtuel basé sur l'IP avec une seule interface réseau est une application de ce type.

Supposons, par exemple, ce qui suit :

- une seule interface Ethernet de votre machine est raccordée à un « hub Ethernet » (pas au routeur à large bande) :
 - le concentrateur Ethernet est connecté à la fois à Internet et au réseau local :
-

- le réseau local utilise le sous-réseau `192.168.0.x/24` :
- votre machine utilise une adresse IP fournie par DHCP avec l'interface physique `eth0` pour Internet :
- votre machine utilise `192.168.0.1` avec l'interface virtuelle `eth0:0` pour le réseau local.

Les entrées suivantes de « `/etc/network/interfaces` » permettent de configurer votre réseau :

```
iface eth0 inet dhcp
    metric 0
iface eth0:0 inet static
    address 192.168.0.1
    netmask 255.255.255.0
    network 192.168.0.0
    metric 1
```



Attention

Bien que cet exemple de configuration avec [traduction d'adresse réseau \(NAT\)](#) en utilisant [netfilter/iptables](#) (consultez Section 5.9) puisse fournir un routeur économique pour le réseau local avec une seule interface, il n'y a pas, avec votre configuration, de possibilité réelle de pare-feu. Vous devriez utiliser deux interfaces physiques avec NAT afin de sécuriser le réseau local contre ce qui provient d'Internet.

5.6.4 Syntaxe de commande avancée

Le paquet `ifupdown` fournit une configuration avancée du réseau en utilisant le nom **configuration réseau** et le nom **interface réseau**. J'utilise une terminologie un peu différente de celle utilisée dans `ifup(8)` et `interfaces(5)`.

terminologie de la page de manuel	ma terminologie	exemple dans le texte suivant :	description
nom de l' interface physique	nom de l' interface réseau	<code>lo, eth0,</code> <code><nom_interface></code>	nom donné dans le noyau Linux (utilisant le mécanisme <code>udev</code>)
nom de l' interface logique	nom de la configuration réseau	<code>config1, config2,</code> <code><nom_config></code>	noms suivants iface dans le fichier « <code>/etc/network/interfaces</code> »

Table 5.12 – Terminologie des périphériques réseau

Les commandes élémentaires de configuration du réseau se trouvant dans Section 5.5.1 exigent que le nom de la **configuration réseau** de l'entrée **iface** corresponde au nom de l'**interface réseau** dans le fichier « `/etc/network/interfaces` ».

Les commandes avancées de configuration du réseau permettent de séparer de la manière suivante le nom de la **configuration réseau** et le nom de l'**interface réseau** dans le fichier « `/etc/network/interfaces` » :

commande	action
<code>ifup eth0=config1</code>	activer une interface réseau <code>eth0</code> avec la configuration <code>config1</code>
<code>ifdown eth0=config1</code>	désactiver une interface réseau <code>eth0</code> avec la configuration <code>config1</code>
<code>ifup eth0</code>	activer une interface réseau <code>eth0</code> avec la configuration sélectionnée par l'entrée mapping
<code>ifdown eth0</code>	désactiver une interface réseau <code>eth0</code> avec la configuration sélectionnée par l'entrée mapping

Table 5.13 – Liste des commandes avancées de configuration du réseau par `ifupdown`

5.6.5 L'entrée « mapping »

Nous avons sauté l'explication de l'entrée **mapping** du fichier « `/etc/network/interfaces` » dans la Section 5.5.2 pour éviter des complications. Cette entrée possède la syntaxe suivante :

```
mapping <motif_nom_interface>
  script <nom_script>
  map <script_entrée1>
  map <script_entrée2>
  map ...
```

Cela fournit des fonctionnalités avancées au fichier « `/etc/network/interfaces` » en automatisant le choix de la configuration avec le script de « mapping » indiqué par `<nom_script>`.

Suivons l'exécution de ce qui suit :

```
$ sudo ifup eth0
```

Lorsque « `<motif_nom_interface>` » correspond à « `eth0` », cette exécution lance l'exécution de la commande suivante afin de configurer automatiquement `eth0` :

```
$ sudo ifup eth0=$(echo -e '<script_entrée1> \n <script_entrée2> \n ...' | <nom_script> <- eth0)
```

Ici, les lignes d'entrée du script contenant « `map` » sont facultatives et peuvent être répétées.

Note

Le motif générique (glob) pour l'entrée **mapping** fonctionne comme les motifs génériques des noms de fichiers de l'interpréteur de commandes (consultez Section 1.5.6).

5.6.6 Configuration réseau commutable manuellement

Voici comment commuter manuellement entre plusieurs configurations de réseau sans réécrire le fichier « `/etc/network/interfaces` » comme dans Section 5.5.13.

Créez un paragraphe distinct dans le fichier « `/etc/network/interfaces` » pour chaque configuration de réseau auquel vous souhaitez accéder, comme dans les exemples suivants :

```
auto lo
iface lo inet loopback

iface config1 inet dhcp

iface config2 inet static
  address 192.168.11.100
  netmask 255.255.255.0
  gateway 192.168.11.1
  dns-domain example.com
  dns-nameservers 192.168.11.1

iface pppoe inet manual
  pre-up /sbin/ifconfig eth0 up
  up ifup ppp0=dsl
  down ifdown ppp0=dsl
  post-down /sbin/ifconfig eth0 down

# Ce qui suit n'est utilisé que de manière interne
iface dsl inet ppp
  provider fai_adsl
```

```
iface pots inet ppp
provider fai
```

Vous remarquez que le **nom de configuration de réseau** qui est l'élément suivant **iface** n'utilise pas l'élément pour le **nom d'interface réseau**. Notez aussi qu'il n'y a pas d'entrée **auto** ni d'entrée **allow-hotplug** pour démarrer automatiquement l'interface réseau **eth0** sur événement.

Vous êtes maintenant prêt à commuter la configuration de votre réseau.

Déplaçons votre PC vers un réseau local desservi par DHCP. Vous activez l'**interface réseau** (l'interface physique) **eth0** en lui assignant le nom de **configuration réseau** (le nom de l'interface logique) **config1** de la manière suivante :

```
$ sudo ifup eth0=config1
Password:
...
```

L'interface **eth0** est active, configurée par DHCP et connectée au réseau local.

```
$ sudo ifdown eth0=config1
...
```

L'interface **eth0** est inactive et déconnectée du réseau local.

Déplaçons votre PC vers un réseau local avec une adresse IP fixe. Vous activez l'**interface réseau** **eth0** en lui assignant le nom de **configuration réseau** **config2** de la manière suivante :

```
$ sudo ifup eth0=config2
...
```

L'interface **eth0** est active, configurée avec une IP statique et connectée au réseau local. Les paramètres supplémentaires donnés sous la forme **dns-*** configurent le contenu de « **/etc/resolv.conf** ». Ce fichier « **/etc/resolv.conf** » est mieux géré si le paquet **resolvconf** est installé.

```
$ sudo ifdown eth0=config2
...
```

L'interface **eth0** est de nouveau inactive et déconnectée du réseau local.

Déplaçons votre PC sur le port d'un modem large bande connecté au service par PPPoE. Vous activez l'**interface réseau** **eth0** en lui assignant le nom de **configuration réseau** **pppoe** de la manière suivante :

```
$ sudo ifup eth0=pppoe
...
```

L'interface **eth0** est active, configurée avec une connexion PPPoE, directement vers le FAI.

```
$ sudo ifdown eth0=pppoe
...
```

L'interface **eth0** est de nouveau désactivée et déconnectée.

Déplaçons votre PC à un emplacement sans réseau local et sans modem large bande mais avec un POTS et un modem (modem sur le réseau commuté). Vous activez l'**interface réseau** **ppp0** en lui assignant le nom de **configuration réseau** **pots** de la manière suivante :

```
$ sudo ifup ppp0=pots
...
```

L'interface **ppp0** est active et connectée à Internet par PPP.

```
$ sudo ifdown ppp0=pots
...
```

L'interface `ppp0` est inactive et déconnectée d'Internet.

Vous devriez vérifier l'état actuel de la configuration du réseau du système `ifupdown` dans le fichier « `/etc/network/run/ifstate` ».



AVERTISSEMENT

Il se peut que vous ayez besoin d'ajuster le nombre à la fin de `eth*`, `ppp*`, etc. si vous avez plusieurs interfaces réseau.

5.6.7 Scripts avec le système ifupdown

Le système `ifupdown` lance automatiquement les scripts qui sont installés dans « `/etc/network/*/` » en exportant les variables d'environnement vers les scripts.

variable d'environnement	valeur passée
« <code>\$IFACE</code> »	nom physique (nom d'interface) de l'interface en cours de traitement
« <code>\$LOGICAL</code> »	nom logique (nom de configuration) de l'interface en cours de traitement
« <code>\$ADDRFAM</code> »	<famille_d_adresses> de l'interface
« <code>\$METHOD</code> »	<nom_méthode> de l'interface. (par exemple « <code>static</code> »)
« <code>\$MODE</code> »	« <code>up</code> » si lancé depuis <code>ifup</code> , « <code>stop</code> » si lancé depuis <code>ifdown</code>
« <code>\$PHASE</code> »	comme pour « <code>\$MODE</code> » mais avec une plus faible granularité permettant de distinguer les phases <code>pre-up</code> , <code>post-up</code> , <code>pre-down</code> et <code>post-down</code>
« <code>\$VERBOSITY</code> »	indique si « <code>--verbose</code> » a été utilisé. Positionnée à 1 dans ce cas, à 0 sinon
« <code>\$PATH</code> »	chemin de recherche de la commande : « <code>/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin</code> »
« <code>\$IF_<OPTION></code> »	valeur de l'option correspondante de l'entrée iface

Table 5.14 – Liste des variables d'environnement passées par le système ifupdown

Ici, chaque variable d'environnement, « `$IF_<OPTION>` », est créée à partir du nom de l'option correspondante comme <option1> et <option2> en les préfixant avec « `$IF_` », et en convertissant la casse en majuscules, en remplaçant les tirets (« `-` ») par des tirets soulignés (« `_` ») et en supprimant les caractères non alphanumériques.

ASTUCE

Consultez Section 5.5.2 pour <famille_adresse>, <nom_méthode>, <option1> et <option2>.

Le paquet `ifupdown-extra` (consultez Section 5.5.14) utilise ces variables d'environnement pour étendre les fonctionnalités du paquet `ifupdown`. Le paquet `ifmetric` (consultez Section 5.6.2) installe le script « `/etc/network/if-up.d/ifmetric` » qui définit la métrique par l'intermédiaire de la variable « `$IF_METRIC` ». Le paquet `guessnet` (consultez Section 5.6.8), qui fournit une ossature simple et puissante pour la sélection automatique de la configuration du réseau par le mécanisme de cartographie de réseau (« `mapping` »), les utilise aussi.

Note

Pour des exemples plus spécifiques de scripts personnalisés de configuration du réseau en utilisant les variables d'environnement, vous pouvez consulter les scripts d'exemples se trouvant dans « `/usr/share/doc/ifupdown/examples/*` » et les scripts utilisés dans les paquets `ifscheme` et `ifupdown-scripts-zg2`. Ces scripts supplémentaires ont des fonctionnalités qui se chevauchent un peu avec les paquets de base `ifupdown-extra` et `guessnet`. Si vous installez ces scripts supplémentaires, vous devrez les personnaliser afin d'éviter des interférences.

5.6.8 Cartographie réseau avec guessnet

Plutôt que de choisir manuellement la configuration comme il est décrit dans la Section 5.6.6, vous pouvez utiliser le mécanisme de cartographie réseau décrit dans la Section 5.6.5 pour sélectionner automatiquement la configuration du réseau à l'aide de scripts personnalisés.

La commande `guessnet-ifupdown(8)` fournie par le paquet `guessnet` est conçue pour être utilisée comme script de cartographie réseau et fournit une infrastructure puissante pour améliorer le système `ifupdown`.

- Vous listez la condition de test comme valeur des options de **guessnet** pour chacune des configurations de réseau de l'entrée **iface**.
- La cartographie du réseau choisit la première entrée **iface** donnant un résultat qui ne soit pas en erreur comme configuration du réseau.

Cette double utilisation du fichier « `/etc/network/interfaces` » par le script de mapping, `guessnet-ifupdown` et l'infrastructure d'origine de configuration du réseau, `ifupdown`, n'a pas d'impact négatif car les options de **guessnet** n'exportent que des variables d'environnement supplémentaires vers les scripts lancés par le système `ifupdown`. Consultez `guessnet-ifupdown(8)` pour davantage d'informations.

Note

Lorsqu'il est nécessaire d'avoir plusieurs lignes d'options de **guessnet** dans « `/etc/network/interfaces` », utilisez les lignes d'options commençant par **guessnet1**, **guessnet2** et ainsi de suite, car le paquet `ifupdown` n'autorise pas que les chaînes de caractères de début des lignes d'options soient dupliquées.

5.7 Configuration réseau de bas niveau

5.7.1 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.

net-tools obsolètes	nouveau Iproute2, etc.	manipulation
<code>ifconfig(8)</code>	<code>ip addr</code>	adresse de protocole (IP ou IPv6) d'un périphérique
<code>route(8)</code>	<code>ip route</code>	entrée de la table de routage
<code>arp(8)</code>	<code>ip neigh</code>	entrée de cache ARP ou NDISC
<code>ipmaddr</code>	<code>ip maddr</code>	adresse multicast
<code>iptunnel</code>	<code>ip tunnel</code>	tunnel sur IP
<code>nameif(8)</code>	<code>ifrename(8)</code>	nommer les interfaces réseau en se basant sur l'adresse MAC
<code>mii-tool(8)</code>	<code>ethtool(8)</code>	paramétrage du périphérique Ethernet

Table 5.15 – Table de conversion depuis les commandes obsolètes `net-tools` vers les nouvelles commandes `Iproute2`

Consultez `ip(8)` et [Howto de la suite utilitaire IPROUTE2](#).

5.7.2 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 :

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` »

commande	description
ifconfig	afficher l'état et l'adresse du lien des interfaces actives
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 5.16 – Liste des commandes de réseau de bas niveau

5.8 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:9, I:111	93	afficher l'utilisation de la bande passante d'une interface réseau
iperf	V:4, I:54	160	outil de mesure de la bande passante du protocole Internet
apt-spy	V:0, I:3	105	écrire un fichier « /etc/apt/sources.list » en fonction de tests de bande passante
ifstat	V:1, I:10	55	InterFace STATistics Monitoring (surveillance des statistiques de l'interface)
bmon	V:1, I:12	139	surveillance portable de la bande passante et estimation du débit
ethstatus	V:0, I:4	39	script qui mesure rapidement le débit d'une interface réseau
bing	V:0, I:2	62	testeur de bande passante empirique et stochastique
bwm-ng	V:2, I:17	88	moniteur de bande passante simple en mode console
ethstats	V:0, I:1	20	moniteur de statistiques Ethernet en mode console
ipfm	V:0, I:0	78	outil d'analyse de bande passante

Table 5.17 – Liste des outils d'optimisation du réseau.

5.8.1 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
```

Essayez 1454 à la place de 1500

Vous voyez que ping(8) a réussi avec 1454.

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.

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)	consultez une justification dans « Configuration optimum du MTU pour les connexions ADSL PPPoE »

Table 5.18 – Lignes directrices pour une valeur optimum de MTU

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.

5.8.2 Définir le MTU

Voici des exemples de définition de la valeur de MTU à partir de ses valeurs par défaut 1500 à 1454.

Pour le DHCP (consultez Section [5.5.4](#)), vous pouvez remplacer les lignes d'entrée **iface** pertinentes du fichier « `/etc/network/interfaces` » avec ce qui suit :

```
iface eth0 inet dhcp
pre-up /sbin/ifconfig $IFACE mtu 1454
```

Pour une IP statique (consultez Section [5.5.5](#)), vous pouvez remplacer les lignes d'entrées **iface** pertinentes du fichier « `/etc/network/interfaces` » avec ce qui suit :

```
iface eth0 inet static
address 192.168.11.100
netmask 255.255.255.0
```

```
gateway 192.168.11.1
mtu 1454
dns-domain example.com
dns-nameservers 192.168.11.1
```

Pour une connexion PPPoE directe (consultez Section 5.4.5), vous pouvez remplacer la ligne « `mtu` » pertinente du fichier « `/etc/ppp/peers/dsl-provider` » par ce qui suit :

```
mtu 1454
```

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 5.9) peuvent limiter la taille des paquets au MSS, ce qui est utile pour le routeur. Consultez "TCP MSS" dans `iptables(8)`.

5.8.3 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.

5.9 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 3.3.1).

paquets	popcon	taille	description
iptables	V:240, I:993	1528	outils d'administration pour <code>netfilter</code> (<code>iptables(8)</code> pour IPv4, <code>ip6tables(8)</code> for IPv6)
arptables	V:0, I:2	84	outils d'administration pour <code>netfilter</code> (<code>arptables(8)</code> pour ARP)
ebtables	V:30, I:56	351	outils d'administration pour <code>netfilter</code> (<code>ebtables(8)</code> pour le pontage Ethernet)
iptstate	V:0, I:4	116	surveillance continue de l'état de <code>netfilter</code> (semblable à <code>top(1)</code>)
shorewall-init	V:0, I:0	80	initialisation de Shoreline Firewall
shorewall	V:6, I:16	2343	Shoreline Firewall , générateur de fichier de configuration pour <code>netfilter</code>
shorewall-lite	V:0, I:0	85	Shoreline Firewall , générateur de fichier de configuration pour <code>netfilter</code> (version légère)
shorewall6	V:0, I:2	895	Shoreline Firewall , générateur de fichier de configuration pour <code>netfilter</code> (version IPv6)
shorewall6-lite	V:0, I:0	83	Shoreline Firewall , générateur de fichier de configuration pour <code>netfilter</code> (version légère, IPv6)

Table 5.19 – Liste d'outils de pare-feu

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/html/ »).

- [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.

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](#).



AVERTISSEMENT

Ce chapitre est obsolète car il est basé sur Debian 7.0 (1Wheezy) sorti en 2013.

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:67, I:148	191705	X	Chromium (navigateur libre de Google)
iceweasel	V:210, I:450	232	, ,	Mozilla Firefox démarqué
iceape-browser	N/A	N/A	, ,	Mozilla , supprimé en raisons de problèmes de sécurité bogue n° 505565
epiphany-browser	V:10, I:41	2145	, ,	GNOME, Epiphany respectant HIG
konqueror	V:28, I:114	3543	, ,	KDE, Konqueror
dillo	V:0, I:5	1504	, ,	Dillo (navigateur léger, basé sur FLTK)
w3m	V:212, I:813	2294	texte	w3m
lynx	V:32, I:104	1901	, ,	Lynx
elinks	V:14, I:32	1587	, ,	ELinks
links	V:17, I:46	2149	, ,	Links (texte uniquement)
links2	V:2, I:17	5392	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
icedtea-plugin	I:24	19	main	Greffon Java basé sur OpenJDK et IcedTea
mozilla-plugin-gnash	I:2	39	main	greffon Flash basé sur Gnash
flashplugin-nonfree	V:9, I:148	71	contrib	assistant à l'installation du greffon Flash pour installer le lecteur Flash d'Adobe (i386 et amd64 uniquement)
browser-plugin-vlc	V:5, I:17	202	main	greffon multimédia basé sur VLC media player

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/*.default/user.js` » :

```
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



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.

Le système de courriels inclus de nombreux logiciels serveurs et clients fonctionnant sur de nombreuses machines hôtes. En termes de fonctionnalités, il y a 3 sortes de logiciels d'agents de courriel :

- L'agent de transport de courriel ([MTA](#), voir Section 6.3) est un programme dont le but est le transfert de courriels entre différents hôtes.
- L'agent de distribution de courriel ([MDA](#), voir Section 6.6) est un programme dont le but est la distribution de messages dans les boîtes aux lettres des utilisateurs au sein d'un hôte.
- Le client de messagerie (MUA, ou encore [email client](#), voir Section 6.4) est le programme utilisé pour créer des messages et accéder aux messages reçus.

Note

Les exemples de configuration suivants ne sont valables que pour une station de travail mobile typique avec des connexions internet de type grand public.

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).

Le [client de messagerie](#) (MUA) a besoin d'interpréter l'en-tête du message et les données du corps en utilisant les [Multipurpose Internet Mail Extensions \(MIME\)](#) pour gérer le type de données et l'encodage du contenu.

6.2.2 Bases du service de courrier électronique moderne

De manière à limiter les risques dus au problème des pourriels (courriels non désirés et non sollicités), de nombreux fournisseurs d'accès à Internet qui fournissent une connexion Internet pour le grand public mettent en place des contre-mesures.

- Le service smarthost pour l'envoi des messages de leurs clients utilise le port (587) défini dans la [rfc4409](#) avec le service de mot de passe ([SMTP AUTH](#)) défini dans la [rfc4954](#).
- La connexion sur le port [SMTP](#) (25) depuis les machines de leur réseau interne (à l'exception de la sortie propre aux serveurs du FAI lui-même) vers Internet est bloquée.
- La connexion au serveur de courrier entrant du FAI sur le port [SMTP](#) (25) depuis certaines machines d'une source externe douteuse est bloquée. (La connexion des machines dans la plage d'adresses IP dynamiques utilisée par les clients du réseau commuté ou d'autres connexions Internet grand public sont les premières à être bloquées).
- La [lutte anti-spam](#) en utilisant [DomainKeys Identified Mail \(DKIM\)](#) et [Sender Policy Framework \(SPF\)](#) est couramment utilisée pour le [filtrage du courrier](#).
- Le service de [DomainKeys Identified Mail](#) peut être fourni pour vos messages envoyés par l'intermédiaire du smarthost.

Lors de la configuration de votre système de courrier électronique, ou lors de la résolution de problèmes de diffusion de courrier, vous devez tenir compte de ces nouvelles limitations.

À la lumière de cette situation hostile d'Internet, certains fournisseurs de service de courrier électronique comme Yahoo.com et Gmail.com offrent un service de courrier électronique sécurisé que l'on peut utiliser depuis n'importe où sur Internet en utilisant [Transport Layer Security \(TLS\)](#) et son prédécesseur, [Secure Sockets Layer \(SSL\)](#).

- Le service de smarthost sur le port 465 avec l'obsolète SMTP sur SSL ([SMTPS](#)).
 - Le service de smarthost sur le port 587 avec [STARTTLS](#).
 - Le courrier électronique entrant est accessible par le port TLS/POP3 (995) par [POP3](#).
-

**Attention**

Il n'est pas réaliste de mettre en œuvre un serveur SMTP sur un réseau grand public pour envoyer de manière fiable le courrier électronique directement vers une machine distante. Il sera probablement rejeté. Vous devez utiliser les services de smarthost offerts par votre fournisseur d'accès ou par un fournisseur de courrier électronique indépendant.

Pour simplifier, supposons que le smarthost situé à « `smtp.hostname.dom` » exige l'[authentification SMTP](#) et utilise le port de soumission de messages (587) avec [STARTTLS](#) dans le texte qui suit.

6.2.3 Stratégie de configuration du courrier électronique pour une station de travail

La configuration de courrier électronique la plus simple est que le courrier électronique soit émis vers le smarthost du fournisseur d'accès et reçu du serveur POP3 de ce fournisseur d'accès par le MUA (consultez Section 6.4) lui-même. Ce type de configuration est courant avec les MUA entièrement basés sur une interface graphique tels que `icedove(1)`, `evolution(1)`, etc. Si vous devez filtrer le courriel selon son type, vous utiliserez les fonctions de filtrage du MUA. Dans ce cas, l'agent de transport du courrier (MTA) local (consultez Section 6.3) n'aura que la distribution locale à effectuer (dans la mesure où expéditeur et destinataire sont sur le même hôte).

Veuillez noter que le système Debian est un système multi-utilisateur. Même si vous êtes l'unique usager, il y a de nombreux logiciels exécutés en tant que superutilisateur qui peuvent vous envoyer un courriel.

La configuration de courriel de remplacement est que le courriel soit envoyé via un MTA local vers le smarthost du FAI et reçu du serveur POP3 du FAI par le logiciel de récupération du courriel (consultez Section 6.5) vers une boîte à lettres locale. Si vous avez besoin de filtrer le courriel en fonction de son type, vous pouvez utiliser un MDA (agent de distribution du courriel) avec filtres (consultez Section 6.6) pour filtrer les courriels vers des boîtes à lettres séparées. Ce type de configuration est populaire avec un simple MUA en mode console tel que `mutt(1)`, `mew(1)`, etc., bien que ce soit possible avec n'importe quel MUA (consultez Section 6.4). Dans ce cas, le MTA local (consultez Section 6.3) doit faire à la fois la diffusion vers le smarthost et la diffusion locale). Comme les stations de travail mobiles n'ont pas de nom de domaine pleinement qualifié (FDQN) valable, vous devez configurer le MTA local pour masquer et simuler le nom de courrier local réel pour le courriel sortant afin d'éviter des erreurs de distribution (consultez Section 6.3.3).

ASTUCE

Vous pourrez avoir envie de configurer MUA/MDA afin d'utiliser [Maildir](#) pour stocker les messages de courrier électronique quelque part dans l'arborescence de votre répertoire personnel.

6.3 Agent de transport de courrier électronique (« MTA »)

Pour une station de travail normale le choix habituel pour l'agent de transport du courrier électronique (MTA) est soit le paquet `exim4-*`, soit `postfix`. C'est vraiment laissé à votre choix.

Bien que suivant le nombre de votes `popcon` `exim4-*` semble de beaucoup plus populaire que `postfix`, cela ne signifie pas que `postfix` n'est pas populaire chez les développeurs de Debian. Le système de serveur de Debian utilise à la fois `exim4` et `postfix`. L'[analyse des en-têtes de courriels](#) vers les listes de diffusion par les développeurs principaux de Debian montre aussi que ces deux MTA ont la même popularité.

Les paquets `exim4-*` sont connus pour consommer très peu de mémoire et avoir une configuration très souple. Le paquet `postfix` est connu pour être compact, rapide, simple et sûr. Les deux sont livrés avec une riche documentation, sont d'aussi bonne qualité et ont une aussi bonne licence.

Il existe dans l'archive Debian un grand choix de paquets d'agents de transport du courrier électronique (MTA) avec des possibilités et des objectifs différents.

paquet	popcon	taille	description
exim4-daemon-light	V:604, I:691	1167	Agent de transport de courrier électronique Exim4 (MTA : par défaut dans Debian)
exim4-base	V:614, I:703	1561	Documentation d'Exim4 (texte) et fichiers communs
exim4-doc-html	I:1	3454	Documentation d'Exim4 (html)
exim4-doc-info	I:1	588	Documentation d'Exim4 (info)
postfix	V:149, I:165	3916	Agent de transport de courrier électronique Postfix (MTA : remplacement)
postfix-doc	I:10	4038	Documentation de Postfix (html+texte)
sasl2-bin	V:7, I:25	410	Implémentation de l'API Cyrus SASL (complément à Postfix pour SMTP AUTH)
cyrus-sasl2-doc	I:2	247	Cyrus SASL - documentation

Table 6.3 – Liste des paquets liés à l'agent de transport du courrier pour une station de travail

paquet	popcon	taille	possibilités et objectif
exim4-daemon-light	V:604, I:691	1167	complet
postfix	V:149, I:165	3916	complet (sécurité)
exim4-daemon-heavy	V:9, I:10	1300	complet (souple)
sendmail-bin	V:10, I:11	1806	complet (uniquement lorsque vous serez déjà familiarisé)
nullmailer	V:5, I:8	372	limité, pas de courrier local
ssmtp	V:13, I:20	2	limité, pas de courrier local
courier-mta	V:0, I:0	1950	très complet (interface web, etc.)
masqmail	V:0, I:0	337	léger
esmtplib	V:0, I:0	124	léger
esmtplib-run	V:0, I:0	32	léger (extension de compatibilité avec sendmail vers esmtplib)
msmtplib	V:2, I:7	260	léger
msmtplib-mta	V:1, I:1	35	léger (extension de compatibilité avec sendmail vers msmtplib)

Table 6.4 – Liste des choix de paquets Debian d'agent de transport du courrier électronique (MTA)

6.3.1 Configuration d'exim4

Pour le courrier d'Internet par l'intermédiaire d'un smarthost, vous (re)configurerez les paquets `exim4-*` comme suit :

```
$ sudo /etc/init.d/exim4 stop
$ 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 ».

Sélectionner « <Non> » pour « Faut-il cacher le nom local de courriel dans les courriels sortants ? ». (Utiliser plutôt « /etc/email-addresses » comme dans Section 6.3.3).

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.

Définir « Méthode de distribution du courrier local : » à « Format « mbox » dans /var/mail ».

Sélectionner « Oui » pour « Faut-il séparer la configuration dans plusieurs fichiers ? ».

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.*\.nommachine\.dom:nomutilisateur@nommachine.dom:motdepasse
```

Lancer `exim4` par la commande suivante :

```
$ sudo /etc/init.d/exim4 start
```

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.template » (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).

— Lancer « `invoke-rc.d exim4 reload` ».

Veuillez lire le guide officiel se trouvant à « /usr/share/doc/exim4-base/README.Debian.gz » et `update-exim4.conf(8)`.

**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.

**AVERTISSEMENT**

L'utilisation de mot de passe en texte non chiffré n'est pas sécurisée, même si votre FAI le permet.

ASTUCE

Même si l'utilisation de [SMTP](#) avec [STARTTLS](#) sur le port 587 est préférable, certains FAI utilisent encore l'obsolète [SMTPS](#) (SSL sur le port 465). Exim4, depuis la version 4.77, permet d'utiliser ce protocole SMTPS obsolète autant en tant que client que serveur.

ASTUCE

Si vous recherchez un MTA léger qui respecte « /etc/aliases » pour votre PC de bureau, vous devriez envisager de configurer exim4(8) avec « QUEUERUNNER='queueonly' », « QUEUERUNNER='nodaemon' », etc. dans « /etc/default/exim4 ».

6.3.2 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
postfix(1)	Programme de contrôle de postfix
postconf(1)	Utilitaire de configuration de postfix
postconf(5)	Paramètres de configuration de postfix
postmap(1)	Maintenance des tables de consultation de postfix
postalias(1)	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 /etc/init.d/postfix stop
$ 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      nomutilisateur:motdepasse
$ sudo postmap hush:/etc/postfix/sasl_passwd
```

Lancez `postfix` comme suit :

```
$ sudo /etc/init.d/postfix start
```

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` ».

6.3.3 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
<code>/etc/mailname</code>	nom de machine par défaut pour le courrier (sortant)	Spécifique à Debian, <code>mailname(5)</code>
<code>/etc/email-addresses</code>	usurpation du nom de machine pour le courriel sortant	Spécifique à <code>exim(8)</code> , <code>exim4-config_files(5)</code>
<code>/etc/postfix/generic</code>	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> .
<code>/etc/aliases</code>	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.

6.3.4 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/*` ».

6.4 Agent de courrier électronique de l'utilisateur (« MUA »)

Si vous vous abonnez à une liste de diffusion relative à Debian, ce peut être une bonne idée d'utiliser un MUA tel que `mutt` ou `mew` qui sont le standard de fait pour le participant et sont connus pour se comporter comme on le souhaite.

paquet	popcon	taille	type
evolution	V:43, I:298	428	programme X avec une interface graphique (GNOME 3, suite « groupware »)
icedove	V:55, I:142	292	programme X avec une interface graphique (GNOME 2, version démarquée de Mozilla Thunderbird)
kmail	V:43, I:98	6419	programme X avec une interface graphique (KDE)
mutt	V:224, I:820	6163	programme de terminal en mode caractère, probablement utilisé avec vim
mew	V:0, I:0	2328	programme de terminal en mode caractères sous (x)emacs

Table 6.8 – Liste d’agents de courrier électronique de l’utilisateur (MUA)

6.4.1 Le MUA de base —Mutt

Configurez « `~/muttrc` » comme suit afin d’utiliser `mutt` comme agent utilisateur de courrier électronique (MUA) en conjonction avec `Vim`.

```
#
# Fichier de configuration de l'utilisateur pour prendre le pas sur /etc/Muttrc
#
# modifier l'adresse électronique d'origine
set use_from
set hostname=example.dom
set from="Prénom Nom <prenom-nom@example.dom>"
set signature="~/signature"

# vim : « gq » pour reformater les citations
set editor="vim -c 'set tw=72 et ft=mail'"

# « mutt » utilise Inbox, alors que « mutt -y » donne la liste des boîtes de messagerie
set mbox_type=Maildir      # utiliser le format Maildir de gmail pour créer les mbox
set mbox=~/.Mail          # conserver toutes les boîtes de messagerie dans $HOME/Mail/
set spoolfile=+Inbox       # message délivré vers $HOME/Mail/Inbox
set record=+Outbox        # sauvegarder les messages envoyés vers $HOME/Mail/Outbox
set postponed=+Postponed  # garder les messages ajournés dans $HOME/Mail/postponed
set move=no               # ne pas déplacer les messages lus de Inbox vers mbox
set quit=ask-yes          # ne pas quitter avec seulement « q »
set delete=yes            # toujours effacer sans confirmation à la fin
set fcc_clear              # garder les messages envoyés non chiffrés

# boîtes de messagerie dans Maildir (mise à jour automatique)
mailboxes 'cd ~/.Mail; /bin/ls -l|sed -e 's/^/+/' | tr "\n" " "'
unmailboxes Maillog *.ev-summary

## par défaut
#set index_format="%4C %Z %{%b %d} %-15.15L (%4l) %s"
## index des enfilades avec expéditeurs (quand elles sont cachées)
set index_format="%4C %Z %{%b %d} %-15.15n %?M?(<#03M>)&(%4l)? %s"

## par défaut
#set folder_format="%2C %t %N %F %2l %-8.8u %-8.8g %8s %d %f"
## simplement les noms de dossier
set folder_format="%2C %t %N %f"
```

Ajoutez ce qui suit dans « `/etc/mailcap` » ou « `~/mailcap` » afin d’afficher le courriel en HTML et les attachements en ligne de documents MS Word.

```
text/html; lynx -force_html %s; needsterminal;
application/msword; /usr/bin/antiword '%s'; copiousoutput; description="Microsoft Word Text ←
"; nametemplate=%s.doc
```

ASTUCE

Mutt peut être utilisé comme client [IMAP](#) et comme convertisseur de format de boîtes à lettres. Vous pouvez marquer les messages avec « t », « T », etc. Ces messages marqués peuvent être copiés avec « ;C » entre différentes boîtes à lettres et supprimés à l'aide de « ;d » en une seule action.

6.5 Utilitaire de récupération et de rediffusion du courriel distant

Plutôt que d'utiliser un client de messagerie pour accéder à des courriels distants afin de les traiter manuellement, vous pourriez vouloir automatiser ce processus afin d'avoir tous les courriels distribués à la machine hôte. L'utilitaire de récupération et de transfert est l'outil qu'il vous faut.

Bien que `fetchmail(1)` ait été le standard de fait pour la récupération du courriel sur GNU/Linux, l'auteur préfère maintenant `getmail(1)`. Si vous désirez rejeter du courriel avant de le télécharger pour économiser de la bande passante, `mailfilter` ou `mpop` peuvent être utiles. Quels que soient les utilitaires de récupération du courrier électronique utilisés, c'est une bonne idée de configurer le système afin de diffuser les courriels récupérés à un MDA, tel que `maildrop`, par l'intermédiaire d'un tube.

paquet	popcon	taille	description
fetchmail	V:7, I:21	2343	récupération du courriel (POP3, APOP, IMAP) (ancien)
getmail4	V:3, I:11	637	récupération du courriel (POP3, IMAP4 et SDPS) (simple, sûr et fiable)
mailfilter	V:0, I:0	274	récupération du courriel (POP3) avec une possibilité de filtrage par expressions rationnelles
mpop	V:0, I:0	222	récupération du courriel (POP3) et MDA avec possibilité de filtrage

Table 6.9 – Listes des utilitaires de récupération et de rediffusion du courriel

6.5.1 configuration de getmail

La configuration de `getmail(1)` est décrite dans la [documentation de getmail](#). voici ma configuration permettant l'accès à plusieurs comptes POP3 en tant qu'utilisateur.

Créez « `/usr/local/bin/getmails` » comme suit :

```
#!/bin/sh
set -e
if [ -f $HOME/.getmail/running ]; then
    echo "getmail est déjà en fonctionnement... (dans le cas contraire, supprimer $HOME/. ←
        getmail/running)" >&2
    pgrep -l "getmai[l]"
    exit 1
else
    echo "getmail n'a pas tourné... " >&2
fi
if [ -f $HOME/.getmail/stop ]; then
    echo "ne pas lancer getmail ... (dans le cas contraire, supprimer $HOME/.getmail/stop)" ←
        >&2
    exit
fi
if [ "$1" = "x-l" ]; then
    exit
fi
rcfiles="/usr/bin/getmail"
for fichier in $HOME/.getmail/config/* ; do
    rcfiles="$rcfiles --rcfile $fichier"
done
```

```
date -u > $HOME/.getmail/running
eval "$srcfiles $@"
rm $HOME/.getmail/running
```

Configurez-le comme suit :

```
$ sudo chmod 755 /usr/local/bin/getmails
$ mkdir -m 0700 $HOME/.getmail
$ mkdir -m 0700 $HOME/.getmail/config
$ mkdir -m 0700 $HOME/.getmail/log
```

Créez les fichiers de configuration « `$HOME/.getmail/config/pop3_name` » pour chacun des comptes POP3 comme suit :

```
[retriever]
type = SimplePOP3SSLRetriever
server = pop.example.com
username = nom_pop3@example.com
password = <votre_mot-de-passe>

[destination]
type = MDA_external
path = /usr/bin/maildrop
unixfrom = True

[options]
verbose = 0
delete = True
delivered_to = False
message_log = ~/.getmail/log/nom_pop3.log
```

Configurez-le comme suit :

```
$ chmod 0600 $HOME/.getmail/config/*
```

Programmez « `/usr/local/bin/getmails` » pour qu'il soit lancé toutes les 15 minutes avec cron(8) en exécutant « `sudo crontab -e -u <nom_utilisateur>` » et en ajoutant l'entrée cron de l'utilisateur suivante :

```
5,20,35,50 * * * * /usr/local/bin/getmails --quiet
```

ASTUCE

Les problèmes d'accès POP3 peuvent ne pas venir de getmail. Certains services POP3 gratuits courants violent le protocole POP3 et leur filtre de SPAM peut ne pas être parfait. Par exemple, ils peuvent effacer les messages juste après la commande RETR et avant de recevoir la commande DELE et peuvent mettre en quarantaines des messages dans la boîte à lettres des Spam. Vous pourrez minimiser les dommages en les configurant de manière à archiver les messages auxquels ils ont accédé et non les supprimer. Consultez aussi « [Certains courriels n'ont pas été téléchargés](#) ».

6.5.2 configuration de fetchmail

La configuration de fetchmail(1) est effectuée par l'intermédiaire de « `/etc/default/fetchmail` », « `/etc/fetchmailrc` » et « `$HOME/.fetchmailrc` ». Consultez les exemples dans « `/usr/share/doc/fetchmail/examples/fetchmailrc.example` ».

6.6 Agent de distribution du courriel (MDA) avec filtre

La plupart des programmes de MTA tels que `postfix` et `exim4` fonctionnent comme des MDA (agent de distribution du courriel, « mail delivery agent »). Il existe des MDA spécialisés ayant des possibilités de filtrage.

Bien que `procmail(1)` ait été le standard de fait comme MDA sous GNU/Linux, l'auteur préfère maintenant `maildrop(1)`. Quel que soit l'utilitaire de filtrage utilisé, c'est une bonne idée que de configurer le système pour distribuer le courriel filtré vers une [Maildir de style qmail](#).

paquet	popcon	taille	description
procmail	V:85, I:808	300	MDA avec filtrage (ancien)
mailagent	V:0, I:14	1283	MDA avec filtre en Perl
maildrop	V:1, I:3	1091	MDA avec langage de filtrage structuré

Table 6.10 – Liste de MDA avec filtrage

6.6.1 configuration de maildrop

La configuration de `maildrop(1)` est décrite dans la [documentation de maildropfilter](#). Voici un exemple de configuration pour « `$HOME/.mailfilter` » :

```
# Configuration locale
MAILROOT="$HOME/Mail"
# configurer conformément au contenu de /etc/mailname
MAILHOST="example.dom"
logfile $HOME/.maildroplog

# les règles sont faites pour que la dernière valeur prenne le pas sur la précédente.

# messages de liste de diffusion ?
if (    /^Precedence:.*list/:h || /^Precedence:.*bulk/:h )
{
    # règles pour les messages de liste de diffusion
    # boîte à lettres par défaut pour les messages de liste de diffusion
    MAILBOX="Inbox-list"
    # boîte à lettres par défaut pour les messages de debian.org
    if ( /^(Sender|Resent-From|Resent-Sender): .*debian.org/:h )
    {
        MAILBOX="service.debian.org"
    }
    # boîte à lettres par défaut pour les messages de bugs.debian.org (BTS)
    if ( /^(Sender|Resent-From|Resent-sender): .*@bugs.debian.org/:h )
    {
        MAILBOX="bugs.debian.org"
    }
    # boîte à lettres pour chaque liste de diffusion proprement configurée avec « List-Id: ↵
    #      truc » ou « List-Id: ...<truc.bidule> »
    if ( /^List-Id: ([^<]*<)?([^>]*)>?:h )
    {
        MAILBOX="$MATCH2"
    }
}
else
{
    # règles pour les messages ne provenant pas de liste de diffusion
    # boîte à lettres de courrier entrant par défaut
    MAILBOX="Inbox-unusual"
    # local mails
```

```

if ( /Envelope-to: .*@$MAILHOST/:h )
{
    MAILBOX="Inbox-local"
}
# Messages en HTML (99% d'indésirables)
if ( /DOCTYPE html/:b ||\
    /^Content-Type: text\/html/ )
{
    MAILBOX="Inbox-html"
}
# règle de liste noire pour les indésirables
if ( /^X-Advertisement/:h ||\
    /^Subject:.*BUSINESS PROPOSAL/:h ||\
    /^Subject:.*URGENT.*ASISSTANCE/:h ||\
    /^Subject: *I NEED YOUR ASSISTANCE/:h )
{
    MAILBOX="Inbox-trash"
}
# règle de liste blanche pour les messages normaux
if ( /^From: .*@debian.org/:h ||\
    /^(Sender|Resent-From|Resent-Sender): .*debian.org/:h ||\
    /^Subject: .*(debian|bug|PATCH)/:h )
{
    MAILBOX="Inbox"
}
# règle de liste blanche pour les messages relatifs au BTS
if ( /^Subject: .*Bug#.*/:h ||\
    /^(To|Cc): .*@bugs.debian.org/:h )
{
    MAILBOX="bugs.debian.org"
}
# règle de liste blanche pour les messages « getmails » de cron
if ( /^Subject: Cron .*getmails/:h )
{
    MAILBOX="Inbox-getmails"
}
}

# vérifier l'existence de $MAILBOX
'test -d $MAILROOT/$MAILBOX'
if ( $RETURNCODE == 1 )
{
    # créer la boîte à lettres au format maildir pour $MAILBOX
    'maildirmake $MAILROOT/$MAILBOX'
}
# délivrer vers le répertoire maildir $MAILBOX
to "$MAILROOT/$MAILBOX/"
exit

```

**AVERTISSEMENT**

À l'inverse de procmail, maildrop ne crée pas automatiquement les répertoires maildir manquants. Vous devez les créer vous-même à l'avance en utilisant maildirmake(1) comme cela a été fait pour « \$HOME/.mailfilter » dans l'exemple.

6.6.2 configuration de procmail

Voici une configuration équivalente avec « \$HOME/.procmailrc » pour procmail(1) :

```
MAILDIR=$HOME/Maildir
DEFAULT=$MAILDIR/Inbox/
LOGFILE=$MAILDIR/Maillog
# courriels qui apparaissent clairement mauvais : les mettre dans X-trash et quitter
:0
* 1^0 ^X-Advertisement
* 1^0 ^Subject:.*BUSINESS PROPOSAL
* 1^0 ^Subject:.*URGENT.*ASISSTANCE
* 1^0 ^Subject:.*I NEED YOUR ASSISTANCE
X-trash/

# Diffusion des messages des listes de diffusion
:0
* 1^0 ^Precedence:.*list
* 1^0 ^Precedence:.*bulk
* 1^0 ^List-
* 1^0 ^X-Distribution:.*bulk
{
:0
* 1^0 ^Return-path:.*debian-devel-admin@debian.or.jp
jp-debian-devel/

:0
* ^Resent-Sender.*debian-user-request@lists.debian.org
debian-user/

:0
* ^Resent-Sender.*debian-devel-request@lists.debian.org
debian-devel/

:0
* ^Resent-Sender.*debian-announce-request@lists.debian.org
debian-announce

:0
mailing-list/
}

:0
Inbox/
```

6.6.3 Redistribuer le contenu d'une « mbox »

Vous devrez distribuer vous-même les courriels se trouvant des les boîtes à lettres de votre répertoire personnel depuis « /var/mail/<nom_utilisateur> » si votre répertoire personnel est plein et que procmail(1) échoue. Après avoir fait de la place sur le disque dans votre répertoire personnel, lancez ce qui suit :

```
# /etc/init.d/${MAILDAEMON} stop
# formail -s procmail </var/mail/<nom_utilisateur>
# /etc/init.d/${MAILDAEMON} start
```

6.7 serveur POP3/IMAP4

Si vous voulez faire tourner un serveur privé sur votre réseau local, vous pouvez envisager de faire tourner un serveur [POP3](#) / [IMAP4](#) pour la distribution du courrier électronique aux clients du réseau local.

paquet	popcon	taille	type	description
courier-pop	V:4, I:5	238	POP3	serveur de courriel Courier - serveur POP3 (format maildir uniquement)
cyrus-pop3d	V:0, I:0	146	POP3	système de courrier Cyrus (prise en charge de POP3)
courier-imap	V:6, I:8	544	IMAP	serveur de courriel Courier - serveur IMAP (format maildir uniquement)
cyrus-imapd	V:0, I:1	331	IMAP	système de courrier Cyrus (prise en charge d'IMAP)

Table 6.11 – Liste de serveurs POP3/IMAP4

6.8 Le serveur et les utilitaires d'impression

Sur l'ancien système « UNIX-like » ; le standard était le [démon d'impression](#) BSD. Comme le format d'impression standard du logiciel libre est PostScript sur les systèmes semblables à UNIX, certains systèmes de filtres ont été utilisés conjointement à [Ghostscript](#) afin de permettre l'impression sur des imprimantes qui ne sont pas PostScript.

Récemment, le [système commun d'impression sous UNIX](#) (« Common UNIX Printing System (CUPS) ») est devenu le nouveau standard de fait. CUPS utilise le [Protocole d'impression Internet](#) (« Internet Printing Protocol (IPP) »). IPP est maintenant pris en charge par d'autres systèmes d'exploitation comme Windows XP et Mac OS X, il est devenu le nouveau standard cross-plateforme de fait pour l'impression à distance avec des possibilités de communications bi-directionnelles.

Le format de données imprimables standard pour une application sur le système Debian est [PostScript \(PS\)](#) qui est un langage de description de page. Les données au format PS sont passées à l'interpréteur Ghostscript PostScript afin de produire les données imprimables spécifiques à l'imprimante. Consultez Section [11.4.1](#).

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:8, I:10	324	imprimante (515)	BSD lpr/lpd (démon d'impression)
lprng	V:1, I:2	3852	, ,	, , (Amélioré)
cups	V:217, I:451	760	IPP (631)	Serveur Internet d'impression CUPS
cups-client	V:80, I:500	464	, ,	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:42, I:423	110	, ,	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:80, I:428	832	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/> ».

6.9 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`.

paquet	popcon	taille	outil	description
openssh-client	V:818, I:996	4106	ssh(1)	client de l'interpréteur de commandes sécurisé
openssh-server	V:677, I:807	883	sshd(8)	serveur de l'interpréteur de commandes sécurisé
ssh-askpass-fullscreen	V:0, I:1	42	ssh-askpass-fullscreen(1)	demande à l'utilisateur une phrase de passe pour ssh-add (GNOME2)
ssh-askpass	V:3, I:44	102	ssh-askpass(1)	demande à l'utilisateur une phrase de passe pour ssh-add (X natif)

Table 6.13 – Liste des serveurs et des utilitaires d'accès à distance

**Attention**

Consultez Section [4.7.3](#) si votre serveur SSH est accessible depuis Internet.

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](#)).

6.9.1 Bases de SSH

**AVERTISSEMENT**

Il ne faut pas que « `/etc/ssh/sshd_not_to_be_run` » soit présent si l'on souhaite faire tourner le serveur OpenSSH.

SSH possède deux protocoles d'identification :

Protocole SSH	Méthode SSH	description
SSH-1	« RSAAuthentication »	authentification de l'utilisateur basée sur une clé d'identité RSA
, ,	« RhostsAuthentication »	authentification de l'hôte basée sur « <code>.rhosts</code> » (non sûre, désactivée)
, ,	« RhostsRSAAuthentication »	authentification de l'hôte basée sur « <code>.rhosts</code> » combinée avec clé d'hôte RSA (désactivée)
, ,	« ChallengeResponseAuthentication »	authentification par « challenge-response » RSA
, ,	« PasswordAuthentication »	authentification basée sur un mot de passe
SSH-2	« PubkeyAuthentication »	authentification de l'utilisateur basée sur une clé publique
, ,	« HostbasedAuthentication »	authentification de la machine basée sur « <code>~/ .rhosts</code> » ou « <code>/etc/hosts.equiv</code> » combiné à une authentification par la clé publique de la machine (désactivée)
, ,	« ChallengeResponseAuthentication »	authentification par « challenge-response »
, ,	« PasswordAuthentication »	authentification basée sur un mot de passe

Table 6.14 – Liste des protocoles et méthodes d'authentification SSH

**Attention**

Faites attention à ces différences si vous utilisez un système autre que Debian.

Consultez « `/usr/share/doc/ssh/README.Debian.gz` », `ssh(1)`, `sshd(8)`, `ssh-agent(1)`, et `ssh-keygen(1)` pour davantage d'informations.

Les fichiers suivants sont les fichiers de configuration importants :

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/identity</code>	clé secrète SSH-1 RSA de l'utilisateur
<code>~/.ssh/id_rsa</code>	clé secrète SSH-2 RSA de l'utilisateur
<code>~/.ssh/id_dsa</code>	clé secrète SSH-2 DSA de l'utilisateur

Table 6.15 – Liste des fichiers de configuration de SSH

ASTUCE

Consultez `ssh-keygen(1)`, `ssh-add(1)` et `ssh-agent(1)` pour la manière d'utiliser les clés publiques et secrètes de SSH.

ASTUCE

Assurez-vous de la validité de la configuration en testant la connexion. En cas de problème, utilisez « `ssh -v` ».

ASTUCE

Vous pouvez changer la phrase de passe pour chiffrer les clés secrètes SSH plus tard avec « `ssh-keygen -p` ».

ASTUCE

Vous pouvez ajouter des options aux entrées de « `~/.ssh/authorized_keys` » afin de limiter les accès à certaines machines et les commandes autorisées. Consultez `sshd(8)` pour obtenir plus de renseignements.

Ce qui suit permettra de démarrer une connexion `ssh(1)` depuis un client :

Si vous utilisez le même nom d'utilisateur sur la machine locale et la machine distante, il n'est pas nécessaire d'entrer « `nomutilisateur@` ». Même si vous utilisez un nom d'utilisateur différent sur la machine locale et la machine distante, vous pouvez l'éliminer en utilisant « `~/.ssh/config` ». Pour le [service Debian Alioth](#) avec le nom de compte « `toto-guest` », vous devrez configurer « `~/.ssh/config` » afin qu'il contienne ce qui suit :

```
Host alioth.debian.org svn.debian.org git.debian.org
  User toto-guest
```

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).

commande	description
<code>ssh nomutilisateur@nommachine..domaine.ext</code>	connexion avec le mode par défaut
<code>ssh -v nomutilisateur@nommachine..domaine.ext</code>	connexion avec le mode par défaut et les messages de débogage
<code>ssh -1 nomutilisateur@nommachine..domaine.ext</code>	forcer la connexion avec SSH version 1
<code>ssh -1 -o RSAAuthentication=no -l nomutilisateur@nommachine..domaine.ext</code>	forcer l'utilisation d'un mot de passe avec SSH version 1
<code>ssh -o PreferredAuthentications=password -l nomutilisateur@nommachine.domaine.ext</code>	forcer l'utilisation d'un mot de passe avec SSH version 2

Table 6.16 – Liste d'exemples de démarrage du client SSH

6.9.2 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:serveur-distant:25 4110:serveur-distant:110 nomutilisateur@serveur-distant
```

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.

6.9.3 Se connecter sans mot de passe distant

On peut éviter de devoir se souvenir des mots de passe des systèmes distants en utilisant « RSAAuthentication » (protocole SSH-1) ou « PubkeyAuthentication » (protocole SSH-2).

Sur le système distant, définissez les entrées respectives suivantes : « RSAAuthentication yes » ou « PubkeyAuthentication yes » dans « /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 :

— « RSAAuthentication » : clé RSA pour SSH-1 (obsolète parce qu'il a été remplacé).

```
$ ssh-keygen
$ cat .ssh/identity.pub | ssh user1@remote "cat - >>.ssh/authorized_keys"
```

— « PubkeyAuthentication » : clé RSA pour SSH-2

```
$ ssh-keygen -t rsa
$ cat .ssh/id_rsa.pub | ssh user1@remote "cat - >>.ssh/authorized_keys"
```

— « PubkeyAuthentication » : clé DSA pour SSH-2 (obsolète parce qu'elle est lente).

```
$ ssh-keygen -t dsa
$ cat .ssh/id_dsa.pub | ssh user1@remote "cat - >>.ssh/authorized_keys"
```

ASTUCE

L'utilisation de clés DSA pour SSH-2 est obsolète parce que la clé est plus courte et lente. Il n'y a plus de raison de contourner le brevet de RSA en utilisant DSA parce qu'il est arrivé en fin de validité. DSA signifie [Digital Signature Algorithm](#) (Algorithme de signature numérique) et il est lent. Consultez aussi [DSA-1571-1](#).

Note

Pour que « HostbasedAuthentication » fonctionne avec SSH-2, vous devez définir les paramètres de « HostbasedAuthentication » à « yes » à la fois dans le fichier « /etc/ssh/sshd_config » sur la machine serveur et dans le fichier « /etc/ssh/ssh_config » ou « ~/.ssh/config » sur la machine client.

6.9.4 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 6.17 – Liste des clients SSH libres pour d'autres plateformes

6.9.5 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/<nom_utilisateur>/.ssh/id_rsa:
Identity added: /home/<nom_utilisateur>/.ssh/id_rsa (/home/<nom_utilisateur>/.ssh/id_rsa)
```

Il n'y a plus besoin de mot de passe distant, à partir de maintenant, pour la commande suivante :

```
$ scp toto <utilisateur>@remote.host:toto
```

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)`.

6.9.6 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 [9.3.13](#)) comme suit :

```
# echo "shutdown -h now" | at now
```

Lancer « `shutdown -h now` » dans une `sessionscreen(1)` (consultez Section [9.1](#)) est une autre manière d'effectuer la même chose.

6.9.7 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 ».

6.10 Autres serveurs d'applications réseau

Voici d'autres serveurs d'applications réseau :

paquet	popcon	taille	protocole	description
telnetd	V:1, I:3	103	TELNET	Serveur TELNET
telnetd-ssl	V:0, I:0	154	, ,	, , (prise en charge de SSL)
nfs-kernel-server	V:28, I:95	326	NFS	Partage de fichiers UNIX
samba	V:116, I:207	10928	SMB	Partage de fichiers et d'imprimantes Windows
netatalk	V:3, I:6	3917	ATP	Partage de fichiers et d'imprimantes Apple/Mac (AppleTalk)
proftpd-basic	V:35, I:44	4431	FTP	Téléchargement généraliste de fichiers
apache2	V:315, I:358	577	HTTP	Serveur Web généraliste
squid	V:11, I:14	7093	, ,	Serveur mandataire (proxy) web généraliste
squid3	V:17, I:20	161	, ,	, ,
slpd	V:0, I:0	NOT_FOSMID		Serveur OpenSLP en tant que serveur LDAP
bind9	V:56, I:86	1773	DNS	adresses IP des autres machines
isc-dhcp-server	V:25, I:63	1392	DHCP	adresse IP du client lui-même

Table 6.18 – Liste 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](#).

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.

6.11 Autres clients d'applications réseau

Voici d'autres clients d'applications réseau :

paquet	popcon	taille	protocole	description
netcat	I:34	16	TCP/IP	couteau de l'armée Suisse pour TCP/IP
openssl	V:785, I:988	1255	SSL	binaire Secure Socket Layer (SSL) et outils de chiffrement associés
stunnel4	V:4, I:20	456	, ,	enrobeur SSL universel
telnet	V:72, I:907	157	TELNET	Client TELNET
telnet-ssl	V:1, I:6	208	, ,	, , (prise en charge de SSL)
nfs-common	V:423, I:792	730	NFS	Partage de fichiers UNIX
smbclient	V:21, I:239	1460	SMB	Client de partage de fichiers et imprimantes MS Windows
cifs-utils	V:36, I:127	230	, ,	commande de montage et de démontage de fichiers MS Windows distants
ftp	V:42, I:813	132	FTP	Client FTP
lftp	V:6, I:41	2188	, ,	, ,
ncftp	V:4, I:30	1506	, ,	client FTP plein écran
wget	V:313, I:998	3053	HTTP et FTP	téléchargement web
curl	V:103, I:424	346	, ,	, ,
axel	V:0, I:5	153	, ,	accélérateur de téléchargement
aria2	V:2, I:15	4564	, ,	accélérateur de téléchargement avec prise en charge de BitTorrent et Metalink
bind9-host	V:426, I:945	311	DNS	host(1) de bind9, « <code>Priority:standard</code> »
dnsutils	V:80, I:876	518	, ,	dig(1) de bind, « <code>Priority:standard</code> »
isc-dhcp-client	V:406, I:963	649	DHCP	obtenir une adresse IP
ldap-utils	V:18, I:80	670	LDAP	obtenir des données d'un serveur LDAP

Table 6.19 – Liste de clients d'applications réseau

6.12 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.nom_isp.net pop3
```

Pour tester le service [POP3](#) ayant [TLS/SSL](#) activé de certains fournisseurs d'accès Internet, 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` ».

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 6.20 – Liste des RFC courantes

Chapitre 7

Le système X Window



AVERTISSEMENT

Ce chapitre est obsolète car il est basé sur Debian 7.0 (Wheezy) sorti en 2013.

Le [système X Window](#) d'un système Debian est basé sur les sources de [X.Org](#).

7.1 Paquets clés

Il existe certains paquets (ou métapaquets) prévus pour faciliter l'installation.

Pour les bases de X, veuillez vous référer à X(7) et [le LDP XWindow-User-HOWTO](#).

7.2 Configurer l'environnement de bureau

Un [environnement de bureau](#) est habituellement constitué de la combinaison d'un [gestionnaire X Window](#), d'un gestionnaire de fichiers et d'une suite compatible de programmes utilitaires.

Vous pouvez configurer un [environnement de bureau](#) complet tel que [GNOME](#), [KDE](#), [Xfce](#) ou [LXDE](#) depuis le menu des tâches d'aptitude.

ASTUCE

Sous les environnements Debian unstable et testing, le menu Tâches peut être désynchronisé de l'état des dernières transitions des paquets. Dans une telle situation, vous devrez, afin d'éviter des conflits de paquets, désélectionner certains paquets ou métapaquets du menu des tâches d'aptitude. Lors de la désélection de paquets ou de métapaquets, vous devrez sélectionner vous-même certains paquets qui fournissent leurs dépendances afin d'éviter qu'ils ne soient automatiquement supprimés.

Vous pouvez aussi mettre en place manuellement un environnement simple n'ayant qu'un [gestionnaire X Window](#) comme [Fluxbox](#).

Consultez [gestionnaires de fenêtres pour X](#) pour un guide du gestionnaire de fenêtres X et de l'environnement de bureau.

(méta)paquet	popcon	taille	description
xorg	I:510	52	bibliothèques X, un serveur X, un jeu de polices et un groupe de clients et d'utilitaires X de base (métapaquet)
xserver-xorg	V:171, I:551	238	suite complète de serveurs X avec leur configuration
xbase-clients	I:68	46	assortiment de clients X divers (méta-paquet)
x11-common	V:406, I:804	305	infrastructure de système de fichiers pour le système X Window
xorg-docs	I:8	2040	diverses documentations pour la suite logicielle X.Org
menu	V:116, I:318	1435	créer le menu Debian avec toutes les applications prenant en charge un menu
gksu	V:51, I:277	289	frontal Gtk+ à su(1) ou sudo(8)
menu-xdg	V:29, I:144	76	convertir la structure de menus de Debian vers la structure de menus xdg de freedesktop.org
xdg-utils	V:244, I:569	306	utilitaires pour intégrer l'environnement de bureau fourni par freedesktop.org
task-gnome-desktop	I:217	6	environnement de bureau standard GNOME (métapaquet)
task-kde-desktop	I:67	6	cœur de l'environnement de bureau KDE (métapaquet)
task-xfce-desktop	I:92	6	environnement de bureau léger Xfce (métapaquet)
task-lxde-desktop	I:45	6	environnement de bureau léger LXDE (métapaquet)
fluxbox	V:3, I:13	3860	Fluxbox : paquet pour un gestionnaire X window hautement configurable et faible consommateur de ressources

Table 7.1 – Liste des (méta)paquets clés pour X Window

7.2.1 Menu Debian

Le [système de menus Debian](#) fournit une interface aux programmes qu'ils soient en mode texte ou orientés X avec `update-menus(1)` du paquet `menu`. Chaque paquet installe ses données de menu dans le répertoire « `/usr/share/menu/` ». Consultez « `/usr/share/menu/README` ».

7.2.2 Menu de Freedesktop.org

Chaque paquet conforme au système de menu xdg du système Freedesktop.org installe ses données de menu fournies par « `*.desktop` » dans « `/usr/share/applications/` ». Les environnements de bureau modernes conformes au standard Freedesktop.org utilisent ces données afin de créer leur menu à l'aide du paquet `xdg-utils`. Consultez « `/usr/share/doc/xdg-utils/README` ».

7.2.3 Menu Debian sous « Freedesktop.org menu »

Pour pouvoir accéder au menu traditionnel de Debian depuis un gestionnaire d'environnement de bureau se conformant au [Freedesktop.org menu](#), tel que GNOME ou KDE, il faut installer le paquet `menu-xdg`.

7.3 La relation client - serveur

Le système X Window est activé sous la forme de la combinaison d'un programme serveur et d'un programme client. La signification des mots **server** et **client** comparés aux mots **local** et **distant** demande ici une attention particulière.

Les serveurs X modernes ont l'extension de mémoire partagée du MIT ([the MIT Shared Memory Extension](#)) et communiquent avec leurs clients locaux en utilisant la mémoire partagée locale. De cette manière le canal de communication inter-processus de réseau transparent de la Xlib est contourné, ce qui permet de gagner en performance avec des images de grande taille.

type	description
serveur X	programme tournant sur une machine locale connectée aux périphériques d’affichage et de saisie de l’utilisateur.
client X	programme tournant sur une machine distante qui traite des données et dialogue avec le serveur X.
serveur d’applications	programme tournant sur une machine distante qui traite des données et dialogue avec les clients de l’application.
client d’applications	programme tournant sur une machine locale connectée aux périphériques d’affichage et de saisie de l’utilisateur.

Table 7.2 – Liste de terminologie client/serveur

7.4 Le serveur X

Consultez `xorg(1)` pour des informations concernant le serveur X.

7.4.1 La (re-)configuration du serveur X

La (re-)configuration du serveur X s’effectue de la façon suivante.

```
# dpkg-reconfigure --priority=low x11-common
```

Note

Les noyaux Linux récents ont une bonne prise en charge des composants graphiques et des périphériques d’entrées avec [DRM](#), [KMS](#), et [udev](#). Le serveur X a été réécrit pour les utiliser. Ainsi `/etc/X11/xorg.conf` n’est habituellement pas présent sur votre système. Ces paramètres sont configurés par le noyau. Voir `fb/modedb.txt` dans la documentation du noyau Linux.

Pour les moniteurs à tube de grande taille, c’est une bonne idée de définir un taux de rafraîchissement aussi élevé que votre moniteur peut le supporter (85 Hz est excellent, 75 Hz est bon) afin de réduire le scintillement. Pour les moniteurs à cristaux liquides, le taux de rafraîchissement plus faible (60 Hz) convient habituellement bien en raison de leur temps de réponse qui est plus lent.

Note

Attention à ne pas utiliser un taux de rafraîchissement trop élevé qui peut entraîner une défaillance matérielle fatale à votre moniteur.

7.4.2 Les méthodes de connexion au serveur X

Il y a plusieurs manières de faire accepter au « serveur X » (côté **affichage**) les connexions en provenance d’un « client X » (côté **application**).



AVERTISSEMENT

Ne pas utiliser, pour la connexion X, de connexion [TCP/IP](#) distante sur un réseau **non sécurisé** à moins que vous n’ayez une très bonne raison de le faire telle que l’utilisation du chiffrement. Une socket de connexion TCP/IP sans chiffrement est susceptible d’une **attaque d’espionnage**, elle est désactivée par défaut sur le système Debian. Utilisez `ssh -X`.

paquet	popcon	taille	utilisateur	chiffrement	méthode	pertinence de l'utilisation
xbase-clients	I:68	46	non vérifié	non	Commande <code>xhost</code>	obsolète
xbase-clients	I:68	46	vérifié	non	Commande <code>xauth</code>	Connexion locale par un tube
openssh-client	V:818, I:996	4106	vérifié	oui	Commande <code>ssh -X</code>	Connexion à distante par le réseau
gdm3	V:197, I:295	4648	vérifié	non (XDMCP)	Gestionnaire d'affichage de GNOME	Connexion locale par un tube
sddm	V:15, I:25	1601	vérifié	non (XDMCP)	Gestionnaire d'affichage de KDE	Connexion locale par un tube
xdm	V:3, I:9	665	vérifié	non (XDMCP)	Gestionnaire d'affichage de X	Connexion locale par un tube
wdm	V:212, I:813	2294	vérifié	non (XDMCP)	Gestionnaire d'affichage Window-Maker	Connexion locale par un tube
ldm	V:0, I:1	465	vérifié	oui	Gestionnaire d'affichage LTSP	connexion réseau distante par SSH (client léger)

Table 7.3 – Liste des méthodes de connexion au serveur X

**AVERTISSEMENT**

Ne pas utiliser non plus de [connexion XDMCP](#) sur un réseau **non sécurisé**. Il envoie des données [UDP/IP](#) sans chiffrement et susceptible d'une **attaque d'espionnage**.

ASTUCE

LTSP signifie [Projet de serveur de terminal Linux](#) (« Linux Terminal Server Project »).

7.5 Démarrer le système X Window

Le système X Window est habituellement démarré en tant que [session X](#) qui est la combinaison d'un serveur X et de la connexion de clients X. Pour le système de bureau normal, les deux tournent sur la station de travail.

La [session X](#) peut être démarrée de l'une des façons suivantes :

- la commande `startx` lancée depuis la ligne de commande ;
- un des programmes démons [gestionnaire de l'affichage X](#) *`dm` démarré à la fin du script de démarrage du répertoire « `/etc/rc?.d/` » (« ? » correspond au niveau de fonctionnement).

ASTUCE

Le script de démarrage des démons de gestion d'affichage examine d'abord le contenu du fichier « `/etc/X11/default-display-manager` » avant leur exécution proprement dite. Cela permet de s'assurer qu'un seul programme démon [X display manager](#) est activé.

ASTUCE

Consultez Section 8.3.5 pour les variables d'environnement initiales du gestionnaire d'affichage X.

Essentiellement, tous ces programmes exécutent le script « `/etc/X11/Xsession` ». Ensuite, le script « `/etc/X11/Xsession` » effectue une action de type `run-parts(8)` pour exécuter les scripts se trouvant dans le répertoire « `/etc/X11/Xsession.d/` ». Le premier programme trouvé, dans l'ordre suivant, est exécuté par la commande « `exec` » intégrée :

1. le script indiqué comme paramètre de « `/etc/X11/Xsession` » par le gestionnaire d'affichage X, s'il est défini ;
2. le script « `~/Xsession` » ou « `~/Xsession` », s'il est défini ;
3. la commande « `/usr/bin/x-session-manager` », si elle est définie ;
4. la commande « `/usr/bin/x-window-manager` », si elle est définie ;
5. la commande « `/usr/bin/x-terminal-emulator` », si elle est définie.

Ce processus est affecté par le contenu de « `/etc/X11/Xsession.options` ». Les programmes exacts vers lesquels pointent les commandes « `/usr/bin/x-*` » sont déterminés par le système d'alternatives de Debian et sont modifiés par « `update-alternatives --config x-session-manager` », etc.

Consultez `Xsession(5)` pour plus d'informations.

7.5.1 Démarrer une session X avec gdm3

`gdm3(1)` vous permet de sélectionner le type de session (ou d'environnement de bureau : Section 7.2), la langue (ou locale: Section 8.3) de la session X depuis son menu. Il conserve les valeurs par défaut dans « `~/dmrc` » sous la forme suivante.

```
[Desktop]
Session=default
Language=fr_FR.UTF-8
```

7.5.2 Personnaliser la session X (méthode classique)

Sur un système où « `/etc/X11/Xsession.options` » contient une ligne « `allow-user-xsession` sans caractère « `#` » la précédant, tout utilisateur qui définit « `~/Xsession` » ou « `~/Xsession` » peut personnaliser l'action de « `/etc/X11/Xsession` » en passant outre le code du système. La dernière commande dans le fichier « `~/Xsession` » devrait toujours être de la forme « `exec un_gestionnaire_de_fenêtre_ou_de_session` » afin de démarrer vos gestionnaires favoris de fenêtres et de session X.

Si cette fonction est utilisée, la sélection du gestionnaire (DM) d'affichage (ou de connexion), du gestionnaire de session ou du gestionnaire de fenêtres (WM) par l'utilitaire système est ignorée.

7.5.3 Personnaliser la session X (nouvelle méthode)

Voici de nouvelles méthodes de personnalisation de la session X sans complètement outrepasser le code système comme précédemment.

- le gestionnaire d'affichage `gdm3` peut sélectionner une session particulière et la définir comme paramètre de « `/etc/X11/Xsession` ».
 - Les fichiers « `/etc/profile` », « `~/profile` », « `/etc/xprofile` », et « `~/xprofile` » sont exécutés comme partie (intégrante) du processus de démarrage `gdm3`.
- Le fichier « `~/Xsessionrc` » est exécuté dans le cadre du processus de démarrage (indépendant du bureau)
 - « `#allow-user-xsession` » dans « `/etc/X11/Xsession.options` » ne restreint pas l'exécution du fichier « `~/Xsessionrc` ».
- Le fichier « `~/gnomerc` » est exécuté dans le cadre du processus de démarrage (bureau GNOME uniquement)

La sélection du gestionnaire d'affichage (ou de connexion) (DM), du gestionnaire de session ou du gestionnaire de fenêtres (WM) par l'utilitaire système est respectée.

Ces fichiers de configuration ne devraient contenir ni « `exec ...` » ni « `exit` ».

7.5.4 Connexion à un client X distant via SSH

L'utilisation de `ssh -X` active une connexion sécurisée entre le serveur X local et une application sur un serveur distant.

Si on ne désire pas utiliser l'option `-X` de la ligne de commande, définir à `yes` les entrées `X11Forwarding` de `/etc/ssh/sshd_config` sur la machine distante.

Démarrez le serveur X sur la station locale.

Ouvrez un `xterm` sur la machine locale.

Lancez `ssh(1)` pour établir une connexion avec un site distant comme suit :

```
utilisateur_local @ localhost $ ssh -q -X -l nom_utilisateur@machine_distante.domaine
Password:
```

Lancez comme suit une application X, par exemple « `gimp` » sur le site distant :

```
nom_utilisateur @ machine_distante $ gimp &
```

Cette méthode permet l'affichage du client X distant comme s'il était connecté par une socket UNIX locale.

7.5.5 Sécuriser le terminal X au travers d'Internet

La sécurisation du terminal X par Internet, qui affiche localement un environnement de bureau qui tourne entièrement de manière distante, peut être facilement réalisée en utilisant des paquets spécialisés tels que `ldm`. Votre machine locale devient un client léger sécurisé du serveur d'applications connecté par SSH.

7.6 Polices dans X Window

[Fontconfig 2.0](#) a été créé en 2002 pour fournir une bibliothèque indépendante de la distribution destinée à configurer et personnaliser l'accès aux polices. Debian, pour les versions ultérieures à `squeeze`, utilise [Fontconfig 2.0](#) pour la configuration de ses polices.

La prise en compte des polices dans le système X Window peut être résumée comme suit :

- Ancien système de gestion des polices côté serveur X
 - Le système de polices de base d'origine de X11 fournit une rétrocompatibilité avec les anciennes versions des applications X clientes.
 - Les polices de base d'origine de X11 sont installées sur le serveur X.
- Système de prise en charge moderne des polices côté client X
 - Le système X moderne prend en charge toutes les polices dont la liste figure ci-dessous (Section 7.6.1, Section 7.6.2 et Section 7.6.3) avec des fonctionnalités avancées telles que l'anti-crênelage.
 - [Xft 2.0](#) connecte les applications X modernes telles que celles de [GNOME](#), [KDE](#) et [LibreOffice](#) avec la bibliothèque [FreeType 2.0](#)
 - [FreeType 2.0](#) fournit une bibliothèque de tramage des polices.
 - [Fontconfig](#) fournit une résolution de la spécification des polices pour [Xft 2.0](#). Consultez `fonts.conf(5)` pour sa configuration.
 - Toutes les applications X modernes qui utilisent [Xft 2.0](#) peuvent dialoguer avec un serveur X moderne en utilisant l'[extension de rendu X](#) (« X Rendering Extension »).
 - L'[extension de rendu X](#) déplace l'accès aux polices et la génération des images de glyphes du serveur vers le client X.

Vous pouvez vérifier les informations de configuration des polices comme suit :

- « `xset q` » pour le chemin des police X11 de base
- « `fc-match` » pour la police par défaut de `fontconfig`

paquet	popcon	taille	description
xfonts-utils	V:38, I:621	387	Programmes utilitaires des polices du système X Window
libxft2	V:169, I:739	122	Xft, bibliothèque permettant de connecter des applications X avec la bibliothèque de tramage des polices FreeType
libfreetype6	V:575, I:991	898	Bibliothèque de tramage des polices FreeType 2.0
fontconfig	V:395, I:882	530	Fontconfig , bibliothèque générique de configuration des polices — binaires de gestion
fontconfig-config	V:398, I:929	420	Fontconfig , bibliothèque générique de configuration des polices — données de configuration

Table 7.4 – Table de paquets prenant en charge les systèmes de polices de X Window

— « `fc-list` » pour les polices disponibles de fontconfig

ASTUCE

« [Unicode et le Pingouin](#) » (« The Penguin and Unicode ») est un bon aperçu du système X Window moderne. D'autres documentations se trouvant à <http://unifont.org/> devraient fournir de bonnes informations sur les polices Unicode, les logiciels fonctionnant avec Unicode, l'internationalisation et les problèmes d'utilisation d'Unicode sur les systèmes d'exploitation [free/libre/open source](#) (FLOSS).

7.6.1 Polices de base

Il y a deux types principaux de [polices de caractères pour ordinateurs](#).

- Polices par champs de bits (« bitmap ») (bonnes pour le tramage en faible résolution)
- Polices par traits/contours (bonnes pour le tramage en haute résolution)

Alors que le redimensionnement des polices bitmap donne des images hachées, le redimensionnement des polices par traits et contours donne des images lisses.

Les polices bitmap sur le système Debian sont habituellement fournies sous forme de [fichiers de police X11 pcf bitmap](#) compressés ayant l'extension de nom de fichier « `.pcf.gz` ».

Les polices de type contour sur le système Debian sont fournies de la manière suivante :

- Fichiers de polices [PostScript](#) Type 1 avec une extension de nom de fichier « `.pfb` » (fichier binaire de police) et « `.afm` » (fichier de métrique de la police).
- Fichiers de polices [TrueType](#) (ou [OpenType](#)) ayant habituellement l'extension « `.ttf` ».

ASTUCE

[OpenType](#) est prévu pour se substituer à la fois à [TrueType](#) et [PostScript](#) Type 1.

ASTUCE

Les polices [DejaVu](#) sont basées sur un sur-ensemble des polices [Bitstream Vera](#)

7.6.2 Polices supplémentaires

`aptitude(8)` peut vous aider à trouver facilement de nouvelles polices de caractères.

- La courte liste de paquets sous « Tâches » → « Localisation »
 - La liste des paquets de données de polices filtrée par l'expression rationnelle sur `debtags` : « `~Gmade-of::data:font` »
-

paquet de police	popcon	taille	police sans-sérf	police sérif	police monospace	source de la police
PostScript	N/A	N/A	Helvetica	Times	Courier	Adobe
gsfonts	I:703	4439	Nimbus Sans L	Nimbus Roman No9 L	Nimbus Mono L	URW (taille compatible avec Adobe)
gsfonts-x11	I:134	96	Nimbus Sans L	Nimbus Roman No9 L	Nimbus Mono L	gestion des polices X avec polices PostScript Type 1.
t1-cyrillic	I:23	4884	Helvetian libre	Times libre	Courier libre	URW étendu (taille compatible avec Adobe)
lmodern	I:148	33270	LMSans*	LMRoman*	LMTypewriter*	PostScript redimensionnable et polices OpenType basées sur « Computer Modern » (venant de TeX)

Table 7.5 – Table de correspondance des polices [PostScript](#) Type 1

paquet de police	popcon	taille	police sans-sérf	police sérif	police monospace	source de la police
ttf-mscorefonts-installer	V:1, I:90	125	Arial	Times New Roman	Courier New	Microsoft (taille compatible avec Adobe) (cela installe des données non libres)
fonts-liberation	I:575	2093	Liberation Sans	Liberation Serif	Liberation Mono	Projet de polices Liberation (taille compatible avec Microsoft)
fonts-freefont-ttf	V:38, I:382	10747	FreeSans	FreeSerif	FreeMono	GNU freefont (taille compatible avec Microsoft)
fonts-dejavu	I:489	39	DejaVu Sans	DejaVu Serif	DejaVu Sans Mono	DejaVu , Bitstream Vera avec couverture Unicode
fonts-dejavu-core	V:310, I:724	2954	DejaVu Sans	DejaVu Serif	DejaVu Sans Mono	DejaVu , Bitstream Vera avec couverture Unicode (sans, sans-bold, serif, serif-bold, mono, mono-bold)
fonts-dejavu-extra	I:510	7217	N/A	N/A	N/A	DejaVu , Bitstream Vera avec couverture Unicode (oblique, italic, bold-oblique, bold-italic, condensed)
ttf-unifont	I:23	16903	N/A	N/A	unifont	GNU Unifont , avec tous les codes de caractères imprimables d'« Unicode 5.1 Basic Multilingual Plane (BMP) » (Plan de base multilingue Unicode 5.1)

Table 7.6 – Table de correspondance des polices [TrueType](#)

- La liste des paquets de polices BDF (bitmap) filtrée par l'expression rationnelle sur le nom de paquet : « `~nxfonts-` »
- La liste des paquets de polices TrueType (outline) filtrée par l'expression rationnelle sur le nom de paquet : « `~nttf-|~nfont-` »

Comme les polices **libres** sont quelque peu limitées, l'installation ou le partage de certaines polices TrueType commerciales est possible pour les utilisateurs de Debian. De manière à faciliter ce processus pour l'utilisateur, certains paquets pratiques ont été créés.

- `mathematica-fonts`
- `fonts-mscorefonts-installer`

Vous obtiendrez une vraiment bonne sélection de polices TrueType en acceptant de contaminer votre système **Libre** avec des polices non libres.

7.6.3 Polices CJK

Voici quelques points ciblés sur les polices de [caractères CJK](#).

type de police	nom de police Japonaise	nom de police chinoise	nom de police coréenne
sans-sérif	gothic, ゴシック	hei, gothic	dodum, gulim, gothic
serif	mincho, 明朝	song, ming	batang

Table 7.7 – Table de mots-clés utilisés dans les noms de polices CJK afin d'indiquer les types de polices

Un nom de police tel que « VL PGothic » avec « P » est une police proportionnelle correspondant à la police de largeur fixe « VL Gothic ».

Par exemple, la table de code [Shift_JIS](#) comporte 7070 caractères. Ils peuvent être regroupés de la manière suivante :

- caractères sur un seul octet JIS X 0201 (191 caractères encore appelés caractères de demi-largeur)
- caractères sur deux octets JIS X 0208 (6879 caractères encore appelés caractères de pleine-largeur)

Les caractères sur deux octets occupent une largeur double sur les consoles qui utilisent des polices CJK de largeur fixe. Afin de s'accommoder d'une telle situation, le [fichier Hanzl Bitmap Font \(HBF\) File](#) ayant l'extension de nom de fichier « `.hbf` » peut être mis en œuvre pour les polices comportant des caractères sur un ou sur deux octets.

Afin d'économiser la place prise par les fichiers de polices [TrueType](#), on peut utiliser un fichier de collection de polices [TrueType](#) ayant l'extension du nom de fichier « `.ttc` ».

Afin de couvrir un espace de code de caractères compliqué, la police CID à clé [PostScript](#) Type 1 est utilisée avec des fichiers CMap qui commencent par « `%!PS-Adobe-3.0 Resource-CMap` ». C'est rarement utilisé pour l'affichage X normal mais est utilisé pour le rendu PDF, etc. (consultez Section [7.7.2](#)).

ASTUCE

Des [glyphes](#) multiples sont attendus de certains points du code [Unicode](#) en raison de l'[unification Han](#). Parmi les plus ennuyeux se trouvent « U+3001 IDEOGRAPHIC COMMA » et « U+3002 IDEOGRAPHIC FULL STOP » dont la position des caractères diffère selon les pays CJK. La priorité de configuration des polices japono-centrées sur les chinoises en utilisant « `~/font.conf` » devrait apaiser l'esprit des Japonais.

7.7 Applications X

7.7.1 Applications X de bureautique

Voici une liste des applications de bureautique de base (LO signifie LibreOffice) :

paquet	popcon	taille du pa- quet	type	description
libreoffice-writer	V:328, I:486	26111	LO	traitement de texte
libreoffice-calc	V:322, I:480	23839	LO	feuille de calcul
libreoffice-impress	V:319, I:477	3851	LO	présentation
libreoffice-base	V:308, I:462	6266	LO	gestion de base de données
libreoffice-draw	V:320, I:478	8970	LO	éditeur de graphiques vectoriels (draw)
libreoffice-math	V:323, I:483	1337	LO	éditeur d'équations et de formules mathématiques
abiword	V:9, I:18	4736	GNOME	traitement de texte
gnumeric	V:16, I:25	7758	GNOME	feuille de calcul
gimp	V:91, I:507	16278	GTK	éditeur de graphiques en champs de bits (« bitmap ») (paint)
inkscape	V:141, I:354	129840	GNOME	éditeur de graphiques vectoriels (draw)
dia-gnome	V:5, I:10	20	GNOME	éditeur d'organigrammes et de diagrammes
planner	V:4, I:8	1170	GNOME	gestion de projets
calligrawords	V:1, I:7	6059	KDE	traitement de texte
calligrasheets	V:1, I:7	14162	KDE	feuille de calcul
calligrastage	V:0, I:7	4814	KDE	présentation
calligraplan	V:0, I:7	8508	KDE	gestion de projets
calligraflow	V:0, I:7	478	KDE	éditeur d'organigrammes et de diagrammes
kexi	V:1, I:7	7678	KDE	gestion de base de données
karbon	V:1, I:8	3450	KDE	éditeur de graphiques vectoriels (draw)
calligra-gemini	V:0, I:0	1026	KDE	éditeur de graphiques en champs de bits (« bitmap ») (paint)

Table 7.8 – Liste d'applications de bureautique de base pour X

paquet	popcon	taille du pa- quet	type	description
evince	V:192, I:373	1363	GNOME	afficheur de documents (pdf)
okular	V:70, I:122	3925	KDE	afficheur de documents (pdf)
calibre	V:8, I:36	49341	KDE	Convertisseur de livre numérique et gestion de bibliothèque.
fbreader	V:3, I:18	3035	GTK	lectrice de e-book
evolution	V:43, I:298	428	GNOME	Gestion d'informations personnelles (logiciel de travail collaboratif et courriel)
kontakt	V:3, I:25	990	KDE	Gestion d'informations personnelles (logiciel de travail collaboratif et courriel)
scribus	V:13, I:29	19136	KDE	éditeur de mise en page de bureau
glabels	V:0, I:5	1376	GNOME	éditeur d'étiquettes
gnucash	V:4, I:15	9545	GNOME	gestion financière personnelle
homebank	V:0, I:3	779	GTK	gestion financière personnelle
kymoney	V:0, I:3	13145	KDE	gestion financière personnelle
shotwell	V:15, I:126	5762	GTK	gestionnaire de photos numériques
xsane	V:22, I:191	909	GTK	interface pour dispositifs de numérisation (scanner)

Table 7.9 – Liste d'applications utilitaires de base pour X

7.7.2 Applications utilitaires pour X

Voici une liste d'applications de base qui ont attiré mon attention :



Attention

Le paquet `poppler-data` (précédemment non libre, consultez Section [11.4.1](#)) doit être installé pour qu'`evince` et `okular` puissent afficher des documents CJK PDF qui utilisent des données Cmap (Section [7.6.3](#)).

Note

L'installation de logiciels tels que `scribus` (KDE) sous l'environnement de bureau GNOME est acceptable puisqu'il n'existe pas de fonctionnalité équivalente sous l'environnement de bureau de GNOME. Cependant, l'installation de nombreux paquets ayant les mêmes fonctionnalités pollue votre menu.

7.8 Les infos de X

7.8.1 Presse-papier

La sélection X à l'aide des 3 boutons de souris est la fonction de presse-papiers native de X (voir Section [1.4.4](#)).

ASTUCE

La combinaison des touches "SHIFT + INSERT" peut fonctionner comme l'équivalent du clic du milieu du bouton de la souris.

Les environnements de bureau modernes (GNOME, KDE, ...) offrent plusieurs presse-papier pour couper, copier, et coller en utilisant le bouton gauche de la souris et des touches du clavier (CTRL-X, CTRL-C et CTRL-V).

paquet	popcon	taille du pa- quet	type	description
xsel	V:10, I:36	48	X	Interface de ligne de commande pour les sélections X
xclip	V:7, I:33	55	X	Interface de ligne de commande pour les sélections X

Table 7.10 – Liste des programmes basé sur X

7.8.2 Cartographie des claviers et affectation des boutons de souris sous X

`xmodmap(1)` est un utilitaire permettant de modifier les cartographies de clavier et l'affectation des boutons de la souris sous le système X Window. Afin d'obtenir le **code de touche** (« keycode »), lancez `xeve(1)` sous X et pressez les touches. Afin d'obtenir la signification de **symbole de touche** (« keysym »), regardez la définition de la MACRO dans le fichier « `/usr/include/X11/keysymdef.h` » (paquet `x11proto-core-dev`). Toutes les instructions « `#define` » de ce fichier ont des noms préfixés par « `XK_` » suivi des noms de **symbole de touche** (« keysym »).

7.8.3 Clients X classiques

La plupart des programmes clients X traditionnels, tels que `xterm(1)`, peuvent être démarrés avec un ensemble d'options en ligne de commande afin de préciser leur géométrie, les polices et l'affichage.

Ils utilisent aussi la base de données de ressources X afin de configurer leur aspect. Les valeurs par défaut des ressources X pour l'ensemble du système se trouvent dans « `/etc/X11/Xresources/*` » et les valeurs par défaut de leurs applications dans « `/etc/X11/app-defaults/*` ». Utilisez ces paramètres comme point de départ.

Le fichier « `~/.Xresources` » est utilisé pour enregistrer les spécifications de ressources de l'utilisateur. Ce fichier est automatiquement fusionné avec les ressources X par défaut lors de la connexion. Pour effectuer des modifications à ces définitions et les rendre immédiatement effectives, fusionnez-les dans la base de données à l'aide de la commande suivante :

```
$ xrdp -merge ~/.Xresources
```

Consultez `x(7)` et `xrdb(1)`.

7.8.4 Émulateur de terminal X —xterm

Vous pourrez tout apprendre ce qui concerne `xterm(1)` sur <http://dickey.his.com/xterm/xterm.faq.html>.

7.8.5 Faire tourner X avec le compte de l'administrateur



AVERTISSEMENT

Ne lancez jamais le gestionnaire de session ou d'affichage X avec le compte de l'administrateur en entrant `root` à l'invite d'un gestionnaire d'affichage comme `gdm3`, même si vous prévoyez d'effectuer des tâches administratives car c'est considéré comme non sûr. L'ensemble de l'architecture de X est considérée comme non sûre si elle tourne avec le compte de l'administrateur. Vous devez toujours avoir le niveau de privilège le plus bas possible, comme avec un compte normal.

Une manière simple de faire tourner un client X particulier, par exemple « `toto` » en tant que superutilisateur est d'utiliser `sudo(8)` etc. de la manière suivante :

```
$ sudo toto &
```

```
$ sudo -s  
# toto &
```

```
$ gksu toto &
```

```
$ ssh -X root@localhost  
# toto &
```

**Attention**

N'utiliser ssh(1) que dans ce but est un gaspillage de ressources.

De manière à ce que le client X puisse se connecter au serveur X, veuillez noter ce qui suit :

- Les valeurs des variables d'environnement « `$XAUTHORITY` » et « `$DISPLAY` » de l'ancien utilisateur doivent être copiées dans celles du nouvel utilisateur.
- Le nouvel utilisateur doit avoir la permission de lecture du fichier vers lequel pointe la valeur de la variable d'environnement « `:$XAUTHORITY` ».

Le paquet `gksu` (popcon : V:51, I:277) est un paquet spécialisé de l'interface graphique GTK+ permettant d'obtenir les privilèges de l'administrateur. Il peut être configuré pour utiliser `su`(1) ou `sudo`(8) en tant que programme de support selon la valeur de la clé « `/apps/gksu/sudo-mode` » de `gconf`. Vous pouvez éditer la clé de `gconf` en utilisant `gconf-editor`(1) (menu : « Applications » → « Outils système » → « Éditeur de configuration »).

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

Il y a 17, 18, ou 10 lettres entre « m » et « n », « i » et « n » ou « l » et « n » dans « multilingualization », « internationalization » et « localization » ce qui correspond à M17N, I18N, et L10N.

Des logiciels modernes, tels que GNOME et KDE, gèrent le multilinguisme. Ils sont internationalisés en les faisant gérer les données [UTF-8](#) et localisés en leur donnant les messages traduits par l'intermédiaire de l'infrastructure gettext(1). Les messages traduits peuvent être offerts sous forme de paquets de localisation séparés. Ils sont simplement sélectionnés en donnant la valeur du paramètres linguistique approprié à des variables d'environnement pertinentes.

La représentation la plus simple d'un texte de données est l'**ASCII** qui suffit à l'anglais et utilise moins de 127 caractères (qui peuvent être représentés avec 7 bits). De manière à pouvoir prendre en compte bien plus de caractères pour l'internationalisation, de nombreux systèmes de codage des caractères ont été inventés. Le système moderne et préféré est **UTF-8** qui peut prendre en charge tous les caractères connus des êtres humains (consultez Section [8.3.1](#)).

Consultez [Introduction à i18n](#) pour davantage d'informations.

La prise en charge du matériel international est activée par des données de configuration du matériel localisées.



AVERTISSEMENT

Ce chapitre est obsolète car il est basé sur Debian 7.0 (1Wheezy) sorti en 2013.

8.1 L'entrée clavier

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
```

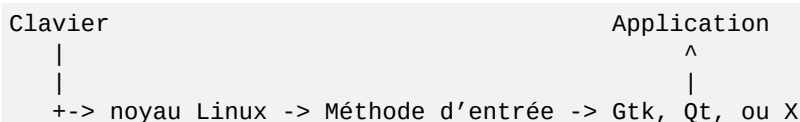
Cela configure le clavier pour la console Linux et X Window met à jour les paramètres de configuration dans `"/etc/default/keyboard"` et `"/etc/default/console-setup"`. Cela configure également la police de caractères de la console Linux.

De nombreux caractères non-ASCII, y compris les caractères accentués utilisés par de nombreuses langues européennes, peuvent être disponibles avec la [touche morte](#), la [touche AltGr](#), et la [touche compose](#).

Pour les langues asiatiques, il faut une gestion plus compliquée de la [méthode d'entrée](#) telle que [IBus](#) qui sera discutée ci-après.

8.1.1 Prise en charge de la méthode d'entrée avec iBus

Les entrées multilingues dans une application sont traitées comme suit :



La configuration de l'entrée multilingue du système Debian est simplifiée par l'utilisation de la famille de paquets [IBus](#) avec le paquet `im-config`. La liste des paquets IBus est la suivante :

paquet	popcon	taille	paramètres linguistiques pris en charge
ibus	V:6, I:9	1937	infrastructure de méthode d'entrée utilisant dbus
ibus-mozc	V:0, I:1	930	Japonais
ibus-anthy	V:1, I:2	734	, ,
ibus-kkc	V:0, I:0	229	, ,
ibus-skk	V:0, I:0	234	, ,
ibus-pinyin	V:0, I:2	1425	Chinois (pour zh_CN)
ibus-chewing	V:0, I:0	401	, , (pour zh_TW)
ibus-hangul	V:0, I:0	224	Coréen
ibus-table	V:0, I:1	908	moteur de tables pour iBus
ibus-table-thai	I:0	45	Thaï
ibus-unikey	V:0, I:0	256	Vietnamien
ibus-m17n	V:0, I:0	121	Multilingue : Indien, Arabe et autres

Table 8.1 – Liste des méthodes d'entrées prises en charge par iBus

La méthode `kinput2` et d'autres [méthodes d'entrée](#) dépendant des paramètres linguistiques asiatiques classiques existent encore mais ne sont pas recommandées avec l'environnement X UTF-8 moderne. Les chaînes d'outils [SCIM](#) et [uim](#) sont une approche légèrement plus ancienne de méthode d'entrée internationale de l'environnement X UTF-8 moderne.

8.1.2 Un exemple pour le japonais

J'ai trouvé très utile la méthode d'entrée du japonais lancée depuis un environnement anglais (« `en_US.UTF-8` »). Voici comment j'ai procédé avec iBus pour GNOME3 :

1. Installez le paquet d'outil d'entrée du japonais `ibus-anthy` avec ses paquets recommandés comme `im-config`.
2. Exécutez « `im-config` » depuis l'interpréteur de commandes de l'utilisateur et sélectionnez « `ibus` » comme méthode de saisie.
3. Sélectionnez "Settings" → "Keyboard" → "Input Sources" → click "+" dans "Input Sources" → "Japanese" → "Japanese (anthy)" et cliquez "Add".
4. Sélectionnez « Japonais » et cliquez sur « Ajouter » pour activer le support de la disposition de clavier japonais sans conversion de caractère. (Vous pouvez choisir autant de sources d'entrée que souhaité.)
5. Reconnectez-vous au compte utilisateur.
6. Vérifiez le paramétrage par « `im-config` ».

7. Spécifiez la source d'entrée par un clic-droit sur l'icône de barre d'outils de l'interface graphique.

8. Choisissez parmi les sources d'entrée installées avec SUPER+ESPACE. (en général, SUPER désigne la touche Windows)

Veuillez noter ce qui suit :

- 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.
- L'entrée du menu de l'interface graphique pour im-config(8) est désactivée par défaut pour éviter de l'encombrer.

8.1.3 Désactiver la méthode d'entrée

Si vous désirez effectuer une entrée sans passer par XIM (le mécanisme utilisé par X), définissez la valeur de « `$XMODIFIERS` » à « none » lors du lancement d'un programme. Ce peut être le cas si vous utilisez l'infrastructure d'entrée japonaise `egg` sous `emacs(1)` alors qu'`ibus` est désactivé. Depuis l'interpréteur de commandes, lancez ce qui suit :

```
$ XMODIFIERS=none emacs
```

Pour ajuster la commande exécutée par le menu Debian, placez la configuration personnalisée dans « `/etc/menu/` » en suivant la méthode décrite dans « `/usr/share/doc/menu/html` ».

8.2 L'affichage de sortie

La console Linux ne peut afficher qu'un nombre restreint de caractères. (Vous devrez avoir un programme de terminal particulier tel que `jfbterm(1)` pour afficher les langues non européennes sur des consoles autres que la console X).

Le système X Window peut afficher tous les caractères UTF-8 dès que les données de polices de caractères existent. (Le codage des données de police d'origine est pris en charge par le système X Window de manière transparente pour l'utilisateur).

8.3 Les paramètres linguistiques (« locale »)

Ce qui suit met l'accent sur les paramètres linguistiques pour les applications tournant sous l'environnement X Window lancé par `gdm3(1)`.

8.3.1 Bases du codage

La variable d'environnement « `LANG=xx_YY.ZZZZ` » définit les paramètres linguistiques avec le code de langue « `xx` », le code de pays « `yy` » et le codage « `ZZZZ` » (consultez Section 1.5.2).

Le système Debian actuel définit normalement les paramètres linguistiques avec « `LANG=xx_YY.UTF-8` ». Cela utilise le codage [UTF-8](#) avec le jeu de caractères [Unicode](#). Ce système de codage [UTF-8](#) est un système de code multi-octets qui utilise intelligemment les éléments du code. Les données [ASCII](#), qui sont uniquement constituées de code sur 7 bits, sont toujours des données UTF-8 valables qui ne comportent qu'un octet par caractère.

Le système Debian précédent définissait habituellement les paramètres linguistiques avec « `LANG=C` » ou « `LANG=xx_YY` » (sans « `.UTF-8` »).

- Le jeu de caractères [ASCII](#) est utilisé avec « `LANG=C` » ou « `LANG=POSIX` ».
- Le système de codage traditionnel sous UNIX est utilisé avec « `LANG=xx_YY` ».

Le système de codage traditionnel actuel utilisé pour « `LANG=xx_YY` » peut être identifié en vérifiant « `/usr/share/i18n/SUPPORTED` ». Par exemple, « `en_US` » utilise le codage « `ISO-8859-1` » et « `fr_FR@euro` » utilise le codage « `ISO-8859-15` ».

ASTUCE

Pour la signification des valeurs de codage, consultez Tableau [11.2](#).

8.3.2 Justification de l'utilisation d'UTF-8 dans les paramètres linguistiques

L'ensemble de caractères [Unicode](#) peut représenter pratiquement tous les caractères humainement connus avec une notation codée s'entendant de 0 à 10FFFF en notation hexadécimale. Son stockage requiert au moins 21 bits.

Le système d'encodage de texte [UTF-8](#) adapte le code des caractères Unicode dans un flux de données 8 bits adapté et compatible avec le système de traitement des données ASCII. **UTF** signifie Format de Transformation Unicode (« Unicode Transformation Format »).

Je recommande l'utilisation de paramètres linguistiques [UTF-8](#) pour votre bureau, par exemple « `LANG=fr_FR.UTF-8` ». La première partie du paramètre linguistique (« locale ») détermine comment les messages seront présentés par les applications. Par exemple, `gedit(1)` (éditeur de texte pour le Bureau GNOME) avec le paramètre de langue « `LANG=fr_FR.UTF-8` » peut afficher et éditer des données textuelles avec des caractères chinois tout en présentant les menus en français, dans la mesure où les polices et les méthodes d'entrée sont installées.

Je recommande aussi de ne définir que la variable d'environnement « `$LANG` », je ne vois pas beaucoup d'avantages à définir une combinaison compliquée de variables « `LC_*` » (consultez `locale(1)`) avec les paramètres linguistiques UTF-8.

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.

```
"texte entre doubles marques de citation" n'est pas "doubles marques de citation ASCII"
'texte entre simples marques de citation' n'est pas 'simples marques de citation ASCII'
```

Lorsque le texte brut [ASCII](#) est converti en [UTF-8](#), il a exactement le même contenu et la même taille que le texte original en ASCII. Il n'y a donc rien à perdre en mettant en œuvre des paramètres régionaux UTF-8.

Certains programmes utilisent davantage de mémoire lors de l'utilisation de `I18N`. 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.

Les anciens systèmes de codage propriétaires non UTF-8 tendent à avoir une différence mineure mais ennuyeuse pour certains caractères tels que les caractères graphiques pour de nombreux pays. La mise en œuvre du système UTF-8 sur les systèmes d'exploitation modernes a pratiquement résolu ces problèmes de codage conflictuels.

8.3.3 Reconfiguration des paramètres linguistiques

Pour accéder à un paramètre linguistique particulier, les données du paramètre linguistique doivent être compilées à partir de la base de données des paramètres linguistiques. (Le système Debian n'est **pas** installé avec tous les paramètres linguistiques précompilés à moins que vous n'ayez installé le paquet `locales-all`). La liste complète des paramètres linguistiques pris en charge pour être compilés se trouve dans « `/usr/share/i18n/SUPPORTED` ». On y trouve la liste de tous les noms des « locale ». La commande ci-après affiche la liste de tous les paramètres linguistiques UTF-8 déjà compilés sous forme binaire :

```
$ locale -a | grep utf8
```

L'exécution de la commande suivante va reconfigurer le paquet `locales` :

```
# dpkg-reconfigure locales
```

Ce processus se déroule en trois étapes :

1. mettre à jour la liste des paramètres linguistiques disponibles ;
2. les compiler sous forme binaire ;
3. définir la valeur des paramètres linguistiques par défaut pour l'ensemble du système dans « `/etc/default/locale` » pour une utilisation par PAM (consultez [Section 4.5](#)).

La liste des paramètres linguistiques disponibles devra comporter « `fr_FR.UTF-8` » et toutes les langues intéressantes avec « `UTF-8` ».

Pour l'anglais des USA, le paramètre linguistique recommandé est « `en_US.UTF-8` ». Pour les autres langues, assurez-vous de bien choisir une valeur avec « `UTF-8` » (NdT : prendre « `de_FR.UTF-8` » pour la langue française en France) . Les caractères internationaux sont tous pris en charge par l'un quelconque de ces paramétrages.

Note

Bien que le paramètre linguistique « C » utilise des messages en anglais, il ne prend en compte que les caractères ASCII.

8.3.4 Valeur de la variable d'environnement « \$LANG »

La valeur de la variable d'environnement « \$LANG » est définie et modifiée par de nombreuses applications.

- Elle est définie au départ par le mécanisme PAM de login(1) pour les programmes de console locale de Linux
- Elle est définie au départ par le mécanisme PAM du gestionnaire d'affichage pour tous les programmes X
- Elle est définie au départ par le mécanisme PAM dessh(1) pour les programmes de la console distante
- Modifiée par certains gestionnaires d'affichage tels que gdm3(1) pour tous les programmes X
- Modifiée par le code de démarrage de la session X par l'intermédiaire de « ~/.xsessionrc » pour tous les programmes X
- Modifiée par le code de démarrage de l'interpréteur de commandes, par exemple « ~/.bashrc », pour tous les programmes de console

ASTUCE

C'est une bonne idée de définir les paramètres linguistiques par défaut pour l'ensemble du système à « en_US.UTF-8 » pour un maximum de compatibilité.

8.3.5 Paramètre linguistique spécifique sous X Window

Vous pouvez choisir vos paramètres linguistiques spécifiques sous X Window indépendamment de leur valeur par défaut valable pour l'ensemble du système en utilisant une personnalisation de PAM (consultez Section 4.5) comme suit :

Cet environnement devrait vous apporter la meilleure expérience d'environnement de bureau stable. Vous avez accès à un terminal en mode caractères fonctionnel avec des messages lisibles même lorsque le système X Window ne fonctionne pas. Cela devient essentiel pour des langues qui utilisent des caractères non romains tels que le chinois, le japonais et le coréen.

Note

Il peut exister une autre manière comme l'amélioration du paquet de gestion de session de X mais veuillez lire ce qui suit qui est la méthode générique et basique pour définir les paramètres linguistiques. Pour gdm3(1), je sais que l'on peut sélectionner la « locale » de la session X par l'intermédiaire de son menu.

La ligne suivante définit l'emplacement du fichier de l'environnement de langue dans le fichier de configuration de PAM, tels que « /etc/pam.d/gdm3 » :

```
auth    required    pam_env.so read_env=1 envfile=/etc/default/locale
```

Modifiez-la avec ce qui suit :

```
auth    required    pam_env.so read_env=1 envfile=/etc/default/locale-x
```

Pour le japonais, créez un fichier « /etc/default/locale-x » avec les permissions « -rw-r--r--1 root root » et contenant ce qui suit :

```
LANG="ja_JP.UTF-8"
```

Conservez le fichier « /etc/default/locale » suivant par défaut pour les autres programmes :

```
LANG="en_US.UTF-8"
```

C'est la technique la plus générique pour personnaliser les paramètres linguistiques et faire que la fenêtre du menu de sélection de gdm3(1) lui-même soit affichée dans la bonne langue.

Pour ce cas, vous pouvez aussi simplement modifier les paramètres linguistiques en utilisant le fichier « ~/.xsessionrc ».

8.3.6 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.

8.3.7 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 <nom_programme>
```

8.3.8 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.UTF-8` » sont différents (consultez Section 9.2.5).

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.

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 Le programme screen

screen(1) est un outil très utile pour ceux qui accèdent à un site distant par l'intermédiaire d'une connexion non fiable ou intermittente car il gère les interruptions de connexion au réseau.

paquet	popcon	taille	description
screen	V:182, I:289	981	multiplexeur de terminal avec une émulation de terminal VT100/ANSI

Table 9.1 – Liste de programmes gérant les interruptions de connexion au réseau

9.1.1 Scénario d'utilisation de screen(1)

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 connexions 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.

9.1.2 Raccourcis clavier de la commande `screen`

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.

affectation	signification
<code>^A ?</code>	afficher l'aide de <code>screen</code> (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 <code>screen</code> du terminal
<code>^A DD</code>	détacher la session <code>screen</code> et se déconnecter

Table 9.2 – Liste des raccourcis clavier de `screen`

Consultez `screen(1)` pour davantage d'informations.

9.2 Enregistrer et présenter des données

9.2.1 Le démon de journal

De nombreux programmes enregistrent leur activité dans le répertoire « `/var/log/` ».

— Démon de journal du noyau : `klogd(8)`

— Démon de journal du système : `rsyslogd(8)`

Consultez Section 3.2.9 et Section 3.2.10.

9.2.2 Analyseur de journaux

Voici les principaux analyseurs de journaux (« `~Gsecurity::log-analyzer` » dans `aptitude(8)`).

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.

paquet	popcon	taille	description
logwatch	V:19, I:21	2214	analyseur de journal avec une sortie sympathique en Perl
fail2ban	V:96, I:108	1290	bannir les IP qui provoquent des erreurs d'authentification multiples
analog	V:5, I:127	3529	analyseur des journaux du serveur web
awstats	V:12, I:20	6799	analyseur des journaux du serveur web puissant ayant de nombreuses fonctionnalités
sarg	V:6, I:6	432	générateur de rapport d'analyse de squid
pflogsumm	V:1, I:4	111	résumer les entrées de journal de postfix
syslog-summary	V:1, I:4	30	résumer le contenu du fichier journal syslog
fwlogwatch	V:0, I:0	474	analyseur des journaux du pare-feu
squidview	V:0, I:2	192	surveiller et analyser les fichiers access.log de squid
swatch	V:0, I:0	112	visualisateur de fichier journal avec correspondance d'expressions rationnelles, mise en évidence et « hooks »
crm114	V:0, I:1	1099	analyseur et filtre de données diverses, dont les pourriels (CRM114)
icmpinfo	V:0, I:0	39	interpréter les messages ICMP

Table 9.3 – Liste des analyseurs de journaux système

9.2.3 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 >fichier_propre
$ vim fichier_propre
```

si vous n'avez pas `script` (par exemple pendant le processus d'amorçage dans l'`initramfs`), vous pouvez alors utiliser ce qui suit :

```
$ sh -i 2>&1 | tee typescript
```

ASTUCE

Certains émulateurs de terminaux X tels que `gnome-terminal` peuvent faire de l'enregistrement. Il faudra peut-être augmenter la taille du tampon de lignes pour le défilement vers l'arrière.

ASTUCE

Vous pouvez utiliser `screen(1)` avec « `^A H` » (consultez Section 9.1.2) pour effectuer des enregistrement de la console.

ASTUCE

Vous pouvez utiliser `emacs(1)` avec « `M-x shell` », « `M-x eshell` » ou « `M-x term` » pour effectuer l'enregistrement de la console. Vous pourrez ensuite utiliser « `C-x C-w` » pour enregistrer le tampon dans un fichier.

9.2.4 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 hlsl` » active la recherche avec mise en évidence.

9.2.5 Affichage personnalisé de la date et de l'heure

Le format par défaut de la date et de l'heure de la commande « `ls -l` » dépend des **paramètres régionaux** (consultez la valeur en Section 1.2.6). La variable « `$LANG` » est d'abord visée, elle peut être surchargée par la variable « `$LC_TIME` ».

Le format réel de l'affichage pour chaque paramètre linguistique dépend de la version de la bibliothèque C standard (paquet `libc6`) utilisée. Par exemple, les différentes versions de Debian ont des valeurs par défaut différentes.

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 9.4 – Exemples d'affichage de la date et de l'heure pour la commande « `ls -l` » avec wheezy

ASTUCE

Vous pouvez éviter d'entrer des options longues sur la ligne de commande en utilisant les alias de commande, par exemple « `alias ls='ls --time-style=+%d.%m.%y\%H:%M'` » (consultez Section 1.5.9).

ASTUCE

[ISO 8601](#) est suivie pour ces formats iso.

9.2.6 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 :

```
$ ROUGE=$(printf "\x1b[31m")
$ NORMAL=$(printf "\x1b[0m")
$ INVERSE=$(printf "\x1b[7m")
$ echo "${ROUGE}TEXTE-ROUGE${NORMAL} ${INVERSE}TEXTE-INVERSE${NORMAL}"
```

9.2.7 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` ».

9.2.8 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.

9.2.9 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	commande
xbase-clients	I:68	46	xwd(1)
gimp	V:91, I:507	16278	Interface graphique du menu
imagemagick	V:142, I:538	192	import(1)
scrot	V:8, I:87	50	scrot(1)

Table 9.5 – Liste des outils de manipulation d’images

9.2.10 Enregistrer les modifications dans des fichiers de configuration

Il existe des outils spécialisés pour enregistrer les modifications de la configuration avec l’aide du système DVCS.

paquet	popcon	taille	description
etckeeper	V:22, I:27	151	enregistrer les fichiers de configuration et leurs métadonnées avec Git (par défaut), Mercurial ou Bazaar (nouveau)
changeltrack	V:0, I:0	62	enregistrer les fichiers de configuration avec RCS (ancien)

Table 9.6 – Liste de paquets pour enregistrer un historique de configuration dans VCS

Je recommande l’utilisation avec `git(1)` du paquet `etckeeper`, il met l’ensemble de « `/etc` » sous le contrôle de VCS. Son guide d’installation et un tutoriel se trouvent dans « `/usr/share/doc/etckeeper/README.gz` ».

Essentiellement, l’exécution de « `sudo etckeeper init` » initialise le dépôt `git` pour « `/etc` » exactement comme le processus décrit dans Section 10.6.5 mais avec quelques scripts automatiques (« `hook` ») permettant une configuration plus approfondie.

Alors que vous effectuez des modifications de votre configuration, vous pouvez aussi utiliser `git(1)` normalement pour l’enregistrer. Il enregistre parfaitement les modifications chaque fois que vous lancez des commandes de gestion de paquet.

ASTUCE

Vous pouvez parcourir l’historique des modifications de « `/etc` » en exécutant « `sudo GIT_DIR=/etc/.git gitk` » avec une vue claire des nouveaux paquets installés et les changements de version des paquets.

9.3 Surveiller, contrôler et démarrer l’activité des programmes

L’activité des programmes peut être surveillée et contrôlée à l’aide d’outils spécialisés.

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.

9.3.1 Temps d’un processus

Afficher la durée du processus indiqué dans la commande.

```
# time une_commande >/dev/null
real    0m0.035s    # temps horloge (temps réel écoulé)
user    0m0.000s    # temps en mode utilisateur
sys     0m0.020s    # temps en mode noyau
```

paquet	popcon	taille	description
coreutils	V:881, I:999	15103	nice(1) : lancer un programme avec une priorité d’ordonnancement modifiée
bsdutils	V:851, I:999	238	renice(1) : modifier la priorité d’ordonnancement d’un programme en cours d’exécution
procps	V:801, I:999	690	utilitaires du système de fichiers « /proc » : ps(1), top(1), kill(1), watch(1), ...
psmisc	V:567, I:970	589	utilitaires du système de fichiers « /proc » : killall(1), fuser(1), peekfd(1), pstree(1)
time	V:39, I:819	82	time(1) : lancer un programme qui indique l’utilisation des ressources du système en fonction du temps
sysstat	V:100, I:119	1336	sar(1), iostat(1), mpstat(1), ... : outils de mesure des performances du système pour Linux
isag	V:0, I:4	106	Générateur interactif de graphes de l’activité système pour sysstat
lsof	V:355, I:945	440	lsof(8) : afficher la liste des fichiers ouverts par un processus en cours d’utilisation en utilisant l’option « -p »
strace	V:21, I:163	1363	strace(1) : tracer les appels système et les signaux
ltrace	V:1, I:22	360	ltrace(1) : tracer les appels de bibliothèque
xtrace	V:0, I:1	333	xtrace(1) : tracer la communication entre un client X11 et le serveur
powertop	V:6, I:227	612	powertop(1) : information concernant la puissance électrique utilisée
cron	V:877, I:997	251	faire tourner des processus en arrière plan selon un calendrier depuis le démon cron(8)
anacron	V:450, I:534	62	ordonnanceur de type cron pour les systèmes qui ne tournent pas 24 heures sur 24
at	V:482, I:828	145	at(1) ou batch(1) : lancer un travail à une heure déterminée ou en dessous d’un niveau de charge donné

Table 9.7 – Liste des outils de surveillance et de contrôle de l’activité des programmes

9.3.2 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 9.8 – Liste des valeurs de politesse pour la priorité d'ordonnancement

```
# nice -19 top # très poli
# nice --20 wodim -v -eject speed=2 dev=0,0 disk.img # très rapide
```

Parfois, une valeur extrême de politesse (« nice ») fait plus de mal que de bien au système. Utilisez cette commande avec précaution.

9.3.3 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 9.9 – 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 `pstree(1)` affiche une arborescence des processus.

9.3.4 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.

C'est un programme interactif en mode plein écran. Vous pouvez afficher l'aide sur son utilisation en tapant la touche « h » et la quitter en tapant la touche « q ».

9.3.5 Afficher les fichiers 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`.

9.3.6 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
```

9.3.7 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
                UTIL.      PID ACCÈS  COMMANDE
/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).

9.3.8 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.

9.3.9 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 "$x" ]; then commande "$x" ; fi; done
```

— combinaison de `find(1)` et de `xargs(1)` :

```
find . -type f -maxdepth 1 -name '*.ext' -print0 | xargs -0 -n 1 commande
```

— `find(1)` avec l'option « `-exec` » avec une commande :

```
find . -type f -maxdepth 1 -name '*.ext' -exec commande '{}' \;
```

— `find(1)` avec l'option « `-exec` » avec un court script de l'interpréteur :

```
find . -type f -maxdepth 1 -name '*.ext' -exec sh -c "commande '{}' && echo 'succès'" \;
```

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)`.

9.3.10 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](#)

Pour l'[interface utilisateur graphique \(GUI\)](#) conforme aux standards freedesktop.org, le fichier `*.desktop` présent dans le répertoire `/usr/share/applications` fournit les attributs nécessaires à l'affichage de chaque programme dans le menu graphique (GUI). Voir Section [7.2.2](#)

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=Navigateur Web Chromium
GenericName=Navigateur Web
Comment=Access 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)`.

9.3.11 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(8)` 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
```

9.3.12 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	fonction
1	HUP	redémarrer le démon
15	TERM	tuer normalement
9	KILL	tuer « dur »

Table 9.10 – Liste des signaux couramment utilisés avec la commande kill

9.3.13 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
```

9.3.14 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)`.

```
# utilise /bin/sh pour exécuter les commandes, quoi que dise /etc/passwd
SHELL=/bin/sh
# envoie un courrier électronique à Paul contenant tous les résultats,
# quelle que soit la personne à qui appartient la crontab
MAILTO=paul
# Minute Heure JourDuMois Mois JourDeLaSemaine commande
# est exécuté à 00:05 chaque jour
5 0 * * * $HOME/bin/daily.job >> $HOME/tmp/out 2>&1
# est exécuté à 14:15 le 1e de chaque mois -- le résultat est envoyé à Paul
# par courrier électronique
15 14 1 * * $HOME/bin/monthly
# est exécuté à 22:00 chaque jour de la semaine(1-5), ennuie Joe.
# % pour une nouvelle ligne, dernier % pour cc:
0 22 * * 1-5 mail -s "Il est 10h" joe%Joe,%Ou sont tes enfants ?%.%
23 */2 1 2 * echo "run 23 minutes after 0am, 2am, 4am ..., on Feb 1"
5 4 * * sun echo "run at 04:05 every sunday"
# est exécuté à 03:40 le premier lundi de chaque mois
40 3 1-7 * * [ "$(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` ».

9.3.15 touche Alt-SysRq

Une assurance contre le mauvais fonctionnement du système est fournie par l'option de compilation du noyau « Magic SysRq key » (touche **SAK**) qui est maintenant une valeur par défaut pour les noyaux de Debian. Presser Alt-SysRq (NDT : Alt-Sys ou Alt-Impr écran sur un clavier français) suivi par une des touches suivantes permet de reprendre par magie le contrôle du système.

touche suivant Alt-Sys	description de l'action
r	restaurer le clavier depuis le mode raw (brut) après un plantage de X
0	changer le niveau de journalisation de la console à 0 afin de réduire les messages d'erreurs
k	kill (tuer) tous les processus sur la console virtuelle actuelle
e	envoyer un SIGTERM à tous les processus, sauf init(8)
i	envoyer un SIGKILL à tous les processus, sauf init(8)
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)
b	reboot (redémarrer) le système sans effectuer de synchronisation ni de démontage

Table 9.11 – Listes des touches de commande SAK (« Secure attention keys »)

ASTUCE

Se référer aux pages de manuel de `signal(7)`, `kill(1)` et `sync(1)` afin de mieux comprendre la description ci-dessus.

La combinaison de « Alt-Sys s », « Alt-Sys u » et « Alt-Sys r » permet de se tirer de situation vraiment désastreuse et obtenir à nouveau l'accès à un clavier opérationnel sans avoir à arrêter le système.

Consultez « `/usr/share/doc/linux-doc-3.*/Documentation/sysrq.txt.gz` ».



Attention

la fonctionnalité Alt-SysRq peut être considérée comme un risque de sécurité permettant à des utilisateurs d'accéder à des fonctions ayant le privilège de l'administrateur. Placer « `echo 0 >/proc/sys/kernel/sysrq` » dans « `/etc/rc.local` » ou « `kernel.sysrq =0` » dans « `/etc/sysctl.conf` » désactive la fonctionnalité Alt-SysRq.

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.

9.4 Astuces de maintenance du système

9.4.1 Qui se trouve sur le système ?

Vous pouvez rechercher qui se trouve sur le système par les commandes suivantes :

— `who(1)` affiche qui est connecté.

- `w(1)` affiche qui est connecté et ce qu'il fait.
- `last(1)` affiche une liste des derniers utilisateurs connectés.
- `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)`.

9.4.2 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 "Nous allons arrêter le système dans une heure" | wall
```

9.4.3 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:167, I:992	1220	utilitaires PCI de Linux : <code>lspci(8)</code>
usbutils	V:97, I:862	721	utilitaires USB de Linux : <code>lsusb(8)</code>
pcmciautils	V:22, I:52	94	utilitaires PCMCIA pour Linux : <code>pccardctl(8)</code>
scsitools	V:0, I:4	366	collection d'outils pour la gestion des périphériques SCSI : <code>lsscsi(8)</code>
procinfo	V:1, I:16	186	informations sur le système obtenues dans « <code>/proc</code> » : <code>lsdev(8)</code>
lshw	V:9, I:85	671	informations concernant la configuration matérielle : <code>lshw(1)</code>
discover	V:39, I:927	90	système d'identification du matériel : <code>discover(8)</code>

Table 9.12 – Listes des outils d'identification du matériel

9.4.4 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`.

paquet	popcon	taille	description
console-setup	V:450, I:933	437	Police de la console Linux et utilitaires de table de caractères
x11-xserver-utils	V:364, I:585	516	utilitaires pour le serveur X : xset(1), xmodmap(1)
acpid	V:508, I:895	143	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:38, I:863	46	utilitaire d'affichage des informations des périphériques ACPI
apmd	V:0, I:9	132	démon de gestion des événements délivrés par la gestion avancée de l'alimentation APM (« Advanced Power Management »)
sleepd	V:0, I:0	79	démon permettant de mettre un ordinateur portable en veille lorsqu'il est inactif
hdparm	V:238, I:395	237	optimisation de l'accès aux disques durs (consultez Section 9.5.9)
smartmontools	V:112, I:194	1716	contrôle et surveillance des systèmes de stockage en utilisant S.M.A.R.T.
setserial	V:5, I:12	113	collection d'outils pour gérer les ports série
memtest86+	V:1, I:38	2391	collection d'outils pour gérer la mémoire physique
scsitools	V:0, I:4	366	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:0	27	curseurs de souris plus grands pour X

Table 9.13 – Liste des outils de configuration du matériel

9.4.5 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\(GMT\)](#).

Si l'heure matérielle (BIOS) est réglée en TUC, modifiez le réglage pour « UTC=yes » dans le fichier « /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](#), [ntpdate](#) ou [chrony](#).

ASTUCE

Under [systemd](#), use `systemd-timesyncd` for the network time synchronization instead. See `systemd-timesyncd(8)`.

Consultez ce qui suit :

- [HOWTO Gérer de manière précise la date et l'heure](#) (« Managing Accurate Date and Time HOWTO »)
- [Projet de services publics NTP](#)
- Le paquet `ntp-doc`

ASTUCE

`ntpdate(8)` du paquet `ntp` peut suivre la trace d'une chaîne de serveurs NTP jusqu'à la source primaire.

9.4.6 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` ».

9.4.7 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.

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

Les logiciels applicatifs peuvent être configurés pour non seulement accéder aux périphériques sonores directement mais aussi pour y accéder par l'intermédiaire d'un système de serveur de son.

paquet	popcon	taille	description
alsa-base	I:463	NOT_FOUND	Fichiers de configuration du pilote ALSA
alsa-utils	V:379, I:532	2103	utilitaires de configuration et d'utilisation d'ALSA
oss-compat	V:4, I:46	7	Compatibilité OSS sous ALSA évitant les erreurs « <code>/dev/dsp not found</code> »
jackd	V:3, I:27	32	serveur du kit de connexion audio JACK . (JACK) (faible latence)
libjack0	V:0, I:19	135	bibliothèque du kit de connexion JACK . (JACK) (faible latence)
nas	V:0, I:0	240	serveur du système audio réseau (NAS)
libaudio2	V:60, I:562	162	bibliothèque du serveur audio réseau (NAS)
pulseaudio	V:358, I:498	5858	serveur PulseAudio , remplaçant d'ESD
libpulse0	V:250, I:654	955	bibliothèque du client PulseAudio , remplace ESD
libgstreamer1.0-0	V:227, I:485	4742	GStreamer : moteur de son de GNOME
libphonon4	I:224	588	Phonon : moteur de son de KDE

Table 9.14 – Liste des paquets son

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.

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 9.15 – Liste des commandes pour désactiver l'économiseur d'écran

9.4.8 Désactiver l'économiseur d'écran

Pour désactiver l'écran de veille, utilisez les commandes suivantes :

9.4.9 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).

```
$ echo "set bell-style none">> ~/.inputrc
```

9.4.10 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 9.16 – Taille mémoire affichée

9.4.11 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)`.
- [FAQ de sécurité Debian](#).
- [Manuel de sécurisation de Debian](#).

paquet	popcon	taille	description
logcheck	V:12, I:14	152	démon pour poster à l'administrateur les anomalies des fichiers journaux du système
debsums	V:5, I:38	120	utilitaire pour vérifier les fichiers des paquets installés d'après leur somme de contrôle MD5
chkrootkit	V:7, I:28	951	détecteur de rootkit .
clamav	V:16, I:71	874	utilitaire anti-virus pour UNIX interface en ligne de commandes.
tiger	V:3, I:4	2599	signale les vulnérabilités du système pour la sécurité
tripwire	V:3, I:4	11480	vérificateur d'intégrité des fichiers et répertoires
john	V:3, I:14	449	outils de casse des mots de passe actifs
aide	V:1, I:2	2002	Environnement avancé de détection d'intrusion (« Advanced Intrusion Detection Environment ») — bibliothèque statique
integrit	V:0, I:0	313	programme de vérification de l'intégrité des fichiers
crack	V:0, I:0	128	programme pour deviner les mots de passe

Table 9.17 – Liste d'outils pour la vérification de la sécurité et de l'intégrité du système

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.

9.5 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.

9.5.1 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.

9.5.2 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.

La plupart des PC utilisent le principe de l'[Enregistrement Maître de démarrage \(MBR\)](#) (« Master Boot Record ») pour enregistrer les données de [partition du disque](#) sur le premier secteur, c'est-à-dire, le secteur 0 [LBA](#) (512 octets).

Note

Certains PC récent avec une [Extensible Firmware Interface \(EFI\)](#), ce qui comprend les Macs basés sur Intel, utilisent un principe de [GUID Partition Table \(GPT\)](#) pour enregistrer les données de [partitionnement du disque](#) ailleurs que sur le premier secteur.

Alors que `fdisk(8)` a été l'outil standard de partitionnement de disque, `parted(8)` le remplace maintenant.

paquet	popcon	taille	GPT	description
util-linux	V:879, I:999	3558	Non pris en charge	divers utilitaires systèmes dont <code>fdisk(8)</code> et <code>cfdisk(8)</code>
parted	V:340, I:544	278	Pris en charge	programme GNU de redimensionnement des partitions <code>Parted</code>
gparted	V:25, I:147	6788	Pris en charge	éditeur de partitions de GNOME basé sur <code>libparted</code>
gdisk	V:16, I:471	746	Pris en charge	éditeur de partitions pour disque GPT
kpartx	V:13, I:28	75	Pris en charge	programme pour créer des mappages de périphériques pour les partitions

Table 9.18 – 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.msdo(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 9.7.6](#)) 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](#).

9.5.3 Accès à une partition en utilisant l'UUID

Bien que la reconfiguration de votre partition ou l'ordre d'activation des supports d'enregistrement amovibles puisse conduire à des noms de partitions différents, vous pouvez y accéder de manière homogène. Cela vous aidera aussi si vous avez plusieurs disques et que votre BIOS ne leur donne pas toujours le même nom de périphérique.

- `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)`.

ASTUCE

Les nœuds de périphériques comme les supports d'enregistrement amovibles peuvent être rendus statiques en utilisant, si besoin, des [règles udev](#). Consultez Section [3.3](#).

9.5.4 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.

9.5.5 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 `unde1` 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 `ete2fsck(8)`).

Des commandes semblables sont disponibles pour chaque système de fichiers pris en charge par Linux.

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.

ASTUCE

Le système de fichiers [Btrfs](#) est disponible dans le noyau Linux 3.2 (Debian wheezy). Il devrait devenir le prochain système de fichiers par défaut après `ext4`.

paquet	popcon	taille	description
e2fsprogs	V:418, I:999	4022	utilitaires pour les systèmes de fichiers ext2/ext3/ext4
reiserfsprogs	V:5, I:23	893	utilitaire pour le système de fichiers Reiserfs
dosfstools	V:83, I:570	235	utilitaire pour le système de fichiers FAT (Microsoft :MS-DOS, Windows)
xfsprogs	V:18, I:80	4338	utilitaire pour le système de fichiers XFS (SGI : IRIX)
ntfs-3g	V:170, I:564	1340	utilitaire pour le système de fichiers NTFS (Microsoft : Windows NT, ...)
jfsutils	V:1, I:13	1533	utilitaire pour le système de fichiers JFS (IBM : AIX, OS/2)
reiser4progs	V:0, I:5	1657	utilitaire pour le système de fichiers Reiser4
hfsprogs	V:0, I:9	303	utilitaire pour les systèmes de fichiers HFS et HFS Plus (Apple : Mac OS)
btrfs-tools	V:11, I:40	26	utilitaire pour le système de fichiers Btrfs
zerofree	V:2, I:53	25	programme pour mettre à zéro les blocs libres des systèmes de fichiers ext2, ext3 et ext4

Table 9.19 – Liste des paquets de gestion des systèmes de fichiers

**AVERTISSEMENT**

Vous ne devriez pas encore utiliser le système de fichiers Btrfs pour vos données essentielles avant que la fonctionnalité fsck(8) n'atteigne l'espace dynamique du noyau et qu'il soit pris en charge par le chargeur d'amorçage.

ASTUCE

Certains outils permettent l'accès au système de fichiers sans prise en charge par le noyau Linux (consultez Section 9.7.2).

9.5.6 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

You can run the fsck(8) command safely on all filesystems including root filesystem on reboot by setting "enable_periodic_fsck" in "/etc/mke2fs.conf" and the max mount count to 0 using "tune2fs -c0 /dev/<partition_name>". See mke2fs.conf(5) and tune2fs(8).

ASTUCE

Vous trouverez les résultats de la commande fsck(8) lancée depuis le script de démarrage dans « /var/log/fsck/ ».

9.5.7 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,

```
# <système fic> <point montage> <type> <options> <dump> <pass>
proc /proc proc defaults 0 0
UUID=709cbe4c-80c1-56db-8ab1-dbce3146d2f7 / ext4 noatime,errors=remount-ro 0 1
UUID=817bae6b-45d2-5aca-4d2a-1267ab46ac23 none swap sw 0 0
/dev/scd0 /media/cdrom0 udf,iso9660 user,noauto 0 0
```

ASTUCE

Un [UUID](#) (consultez Section 9.5.3) peut être utilisé pour identifier un périphérique bloc au lieu des noms de périphériques blocs normaux comme « /dev/sda1 » ou « /dev/sda2 » par exemple.

Les performances et les caractéristiques d'un système de fichiers peuvent être optimisées à l'aide des options de montage utilisées (consultez `fstab(5)` et `mount(8)`). Les principales options sont les suivantes :

- l'option « `defaults` » implique les options par défaut : « `rw,suid,dev,exec,auto,nouser,async` » (général) ;
- les options « `noatime` » ou « `relatime` » sont très efficace pour accélérer la vitesse de lecture (général) ;
- l'option « `user` » permet à un utilisateur normal de monter le système de fichiers. Cette option implique la combinaison d'options « `noexec,nosuid,nodev` » (général, utilisée pour les CD et les périphériques de stockage USB).
- la combinaison d'options « `noexec,nodev,nosuid` » est utilisée pour améliorer la sécurité (général) ;
- l'option « `noauto` » limite le montage uniquement à une opération explicite (général) ;
- l'option « `data=journal` » pour `ext3fs` peut améliorer l'intégrité des données lors d'une panne de courant, elle s'accompagne d'une légère baisse de la vitesse d'écriture.

ASTUCE

Vous devez donner au noyau le paramètre de démarrage (consultez Section 3.1.2), par exemple « `rootflags=data=journal` » afin de mettre en œuvre pour le système de fichiers racine un mode de journalisation autre que celui par défaut. Pour Lenny, le mode de journalisation par défaut est « `rootflags=data=ordered` ». Pour squeeze, il s'agit de « `rootflags=data=writeback` ».

9.5.8 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](#).

9.5.9 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` ».

9.5.10 Optimisation du SSD

Les performances et l'usure du [solid-state drive \(SSD\)](#) peuvent être optimisées comme suit.

- Utiliser le dernier noyau (≥ 3.2).
- Réduire les écritures sur le disque pour les accès en lecture.
 - Définir les options de montage « `noatime` » ou « `relatime` » dans `/etc/fstab`.
- Activer la commande [TRIM](#).
 - Définir l'option de montage « `discard` » dans `/etc/fstab` pour les systèmes de fichiers ext4 et Btrfs, les partitions de swap, etc. Consultez `fstab(5)`.
 - Définir l'option « `discard` » dans `/etc/lvm/lvm.conf` pour [LVM](#). Consultez `lvm.conf(5)`.
 - Définir l'option « `discard` » dans `/etc/crypttab` pour [dm-crypt](#). Consultez `crypttab(5)`.
- Activer le schéma d'allocation d'espace disque optimisé pour SSD.
 - Définir l'option de montage « `ssd` » dans `/etc/fstab` pour Btrfs.
- Vider les données sur le disque toutes les dix minutes pour les ordinateurs portables.
 - Définir l'option de montage « `commit=600` » dans `/etc/fstab`. Consultez `fstab(5)`.
 - Configurer `pm-utils` pour utiliser `laptop-mode` même sous tension. Consultez le [bogue Debian n° 659260](#).



AVERTISSEMENT

Modifier l'intervalle de vidage de données à dix minutes, au lieu de la valeur par défaut de cinq secondes, peut rendre les données vulnérables aux coupures de courant.

9.5.11 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` » ;
 - relancez le démon `smartd`(8) avec « `sudo /etc/init.d/smartmontools restart` ».

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.

9.5.12 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.

9.5.13 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.

9.5.14 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 repertoire-travail ancien-repertoire
$ sudo mkfs.ext4 /dev/sdx
$ sudo mount -t ext4 /dev/sdx repertoire-travail
$ sudo cp -a ancien-repertoire/* repertoire-travail
$ sudo rm -rf ancien-repertoire
```

ASTUCE

Vous pouvez aussi monter un fichier image de disque vide (consultez Section [9.6.5](#)) en tant que périphérique de rebouclage (consultez Section [9.6.3](#)). L’utilisation réelle du disque croît avec les données réellement enregistrées.

9.5.15 Extension de l’espace de stockage en remontant un autre répertoire

Si vous avez un répertoire vide (par exemple, « `/chemin/vers/repertoire-temporaire` ») sur une autre partition avec de l’espace disponible, vous pouvez le remonter avec l’option « `--bind` » vers un répertoire (par exemple « `repertoire-travail` ») où vous avez besoin de place.

```
$ sudo mount --bind /chemin/vers/repertoire-temporaire repertoire-travail
```

9.5.16 Extension de l'espace utilisable à l'aide de liens symboliques

ASTUCE

Cette méthode est obsolète. Utilisez plutôt Section [9.5.15](#) si possible.

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 répertoire-travail ancien-répertoire
$ sudo mkdir -p /chemin/vers/répertoire-temporaire
$ sudo ln -sf /chemin/vers/répertoire-temporaire répertoire-travail
$ sudo cp -a ancien-répertoire/* répertoire-travail
$ sudo rm -rf ancien-répertoire
```

**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.

**Attention**

Certains logiciels peuvent ne pas bien fonctionner avec des « liens symboliques vers un répertoire ».

9.5.17 Extension de l'espace de stockage utilisable à l'aide d'overlayfs

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 with [OverlayFS](#) with 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 Le fichier image du disque

Nous discutons ici des manipulations sur l'image disque.

9.6.1 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 disque.img
# dd if=/dev/sdb of=disque.img
```

L'image disque du [master boot record \(MBR\)](#) (secteur principal d'amorçage) (consultez Section [9.5.2](#)) 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.

9.6.2 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
```

9.6.3 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.

9.6.4 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
```

9.6.5 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 9.6.3, 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.

9.6.6 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 repertoire_source
```

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 repertoire_source
```

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.

9.6.7 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).

9.6.8 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](#)).

9.7 Les données binaires

Nous allons ici discuter de la manipulation directe des données binaires sur le support d'enregistrement.

9.7.1 Afficher et éditer des données binaires

La méthode la plus basique pour visualiser des données binaires est d'utiliser la commande « `od -t x1` ».

paquet	popcon	taille	description
coreutils	V:881, I:999	15103	paquet de base utilisant <code>od(1)</code> pour vider des fichiers (HEX, ASCII, OCTAL, ...)
bsdmainutils	V:866, I:998	566	paquets utilitaires qui utilisent <code>hd(1)</code> pour vider les fichiers (HEX, ASCII, OCTAL, ...)
hexedit	V:1, I:11	61	éditeurs et visualisateurs binaires (HEX, ASCII)
bless	V:0, I:5	973	éditeur hexadécimal complet (GNOME)
okteta	V:2, I:22	382	éditeur hexadécimal complet (KDE4)
ncurses-hexedit	V:0, I:2	126	éditeur et visualisateur binaire (HEX, ASCII, EBCDIC)
beav	V:0, I:1	133	éditeur et visualisateur binaire (HEX, ASCII, EBCDIC, OCTAL, ...)

Table 9.20 – Liste des paquets permettant de visualiser et d'éditer des données binaires

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](#).

9.7.2 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:13, I:114	361	utilitaires pour les fichiers MSDOS sans les monter
hfsutils	V:0, I:9	1842	utilitaires pour les fichiers HFS et HFS+ sans les monter

Table 9.21 – Liste des paquets pour manipuler les fichiers sans monter le disque

9.7.3 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.

paquet	popcon	taille	description
par2	V:2, I:11	238	Parity Archive Volume Set, pour vérifier et réparer des fichiers
dvdisaster	V:0, I:2	1743	protection des supports CD et DVD contre les pertes de données, les rayures et le vieillissement
dvbackup	V:0, I:0	412	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 9.22 – Liste d'outils pour ajouter des données de redondance aux fichiers

paquet	popcon	taille	description
testdisk	V:4, I:39	1344	utilitaires pour l'examen de partitions et la récupération de disque
magicrescue	V:0, I:3	220	utilitaire pour la récupération de fichiers et de recherche des octets magiques
scalpel	V:0, I:4	82	récupérateur de fichiers sobre de haute performance
myrescue	V:0, I:2	38	récupérer des données depuis des disques endommagés
extundelete	V:0, I:11	148	utilitaire pour récupérer des fichiers effacés d'un système de fichiers ext3/4
ext4magic	V:0, I:2	232	utilitaire pour récupérer des fichiers effacés d'un système de fichiers ext3/4
ext3grep	V:0, I:3	278	outil pour aider à la récupération de fichiers effacés sur un système de fichiers ext3
scrounge-ntfs	V:0, I:3	45	programme de récupération de données pour les systèmes de fichiers NTFS
gzrt	V:0, I:0	57	boîte à outils de récupération gzip
sleuthkit	V:1, I:8	1114	outil pour autopsie (« forensics analysis »). (Sleuthkit)
autopsy	V:0, I:2	1021	interface graphique à SleuthKit
foremost	V:0, I:7	123	application d'autopsie pour la récupération de données
guymager	V:0, I:0	1104	outil de création d'image d'autopsie basée sur Qt
dcfldd	V:0, I:5	94	version améliorée de dd pour les autopsies et la sécurité

Table 9.23 – Liste de paquets pour la récupération de données et l'analyse par autopsie

9.7.4 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_d`, `eleted_inodes` et `unde1` de `debugfs(8)` dans le paquet `e2fsprogs`.

9.7.5 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 gros_fichier
$ cat x* >gros_fichier
```



Attention

Assurez-vous ne pas avoir de nom de fichier commençant par « x » afin d'éviter des plantages de nom.

9.7.6 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 :

```
$ :>fichier_a_effacer
```

9.7.7 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.
-

9.7.8 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,

9.7.9 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.

9.7.10 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 toto > titi
$ less titi
$ ps aux | grep ' less[ ]'
bozo    4775  0.0  0.0  92200   884 pts/8    S+   00:18   0:00 less titi
$ rm titi
$ ls -l /proc/4775/fd | grep titi
lr-x----- 1 bozo bozo 64 2008-05-09 00:19 4 -> /home/bozo/titi (deleted)
$ cat /proc/4775/fd/4 >titi
$ ls -l
-rw-r--r-- 1 bozo bozo 4 2008-05-09 00:25 titi
$ cat titir
toto
```

Exécutez sur un autre terminal (lorsque vous avez le paquet `lsOf` installé) comme suit

```
$ ls -li titi
2228329 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:02 titi
$ lsof |grep titi|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/titi
$ rm titi
$ lsof |grep titi|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/titi (deleted)
$ cat /proc/4775/fd/4 >titi
$ ls -li titi
2228302 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:05 titi
$ cat titi
toto
```

9.7.11 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 titi
2738404 -rw-r--r-- 2 root root 0 2008-09-15 20:21 tutu
2738404 -rw-r--r-- 2 root root 0 2008-09-15 20:21 toto
```

« `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'[inœud](#) 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 [inœud](#), par exemple « 2738404 », de la manière suivante :

```
# find /chemin/vers/point/de/montage -xdev -inum 2738404
```

9.7.12 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
```

9.8 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 4.7.4). 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](#) et [eCryptfs](#) facilitent de manière native le chiffrement automatique des données par l'intermédiaire de modules du noyau de Linux avec un minimum d'efforts de la part de l'utilisateur.

paquet	popcon	taille	description
cryptsetup	V:25, I:73	355	utilitaires pour chiffrer un périphérique en mode bloc (dm-crypt / LUKS)
cryptmount	V:2, I:4	224	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
ecryptfs-utils	V:5, I:8	396	utilitaires pour systèmes de fichiers empilés (« stackable ») chiffrés (eCryptfs)

Table 9.24 – Liste d'utilitaires de chiffrement des données

[Dm-crypt](#) est un système de fichiers chiffré qui utilise [device-mapper](#). [Device-mapper](#) effectue le « mapping » d'un périphérique en mode bloc sur un autre.

[eCryptfs](#) est un autre système de fichiers chiffré utilisant un système de fichiers empilé. Un système de fichiers empilé s'empile lui-même par dessus un répertoire existant d'un système de fichiers monté.



Attention

Le chiffrement des données a un coût en matière de temps processeur, etc. Veuillez comparer ses avantages à son coût.

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](#).

9.8.1 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.

9.8.2 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
# /etc/init.d/cryptdisks restart
...
# swapon -a
```

9.8.3 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
```

9.8.4 Chiffrer les fichiers automatiquement avec eCryptfs

Vous pouvez chiffrer automatiquement les fichiers se trouvant sous « `~/Private/` » en utilisant [eCryptfs](#) et le paquet `ecryptfs-utils`.

- Lancez `ecryptfs-setup-private(1)` et configurez « `~/Private/` » en suivant les invites.
- Activez « `~/Private/` » en lançant `ecryptfs-mount-private(1)`.
- Déplacez les fichiers de données sensibles vers « `~/Private/` » et faites les liens symboliques à la demande.
 - Candidats : « `~/.fetchmailrc` », « `~/.ssh/identity` », « `~/.ssh/id_rsa` », « `~/.ssh/id_dsa` » et d'autres fichiers ayant « `go-rwx` »
- Déplacez les répertoires ayant des données sensibles vers un sous-répertoire de « `~/Private/` » et faites les liens symboliques nécessaires.
 - Candidats : « `~/gnupg` » et d'autres répertoires avec « `go-rwx` »
- Créez le lien symbolique de « `~/Desktop/Private/` » vers « `~/Private/` » pour faciliter les opérations du bureau.
- Désactivez « `~/Private/` » en lançant `ecryptfs-umount-private(1)`.
- Activez « `~/Private/` » à l'aide de « `ecryptfs-mount-private` » lorsque vous avez besoin de vos données chiffrées.

ASTUCE

Comme [eCryptfs](#) ne chiffre que sélectivement les fichiers sensibles, son coût en matière de ressources système est bien moindre que l'utilisation de [dm-crypt](#) sur l'ensemble du système de fichiers racine ou sur le périphérique « `/home` ». Il ne demande aucun effort particulier sur l'allocation de l'espace disque mais ne peut conserver la confidentialité de toutes les métadonnées du système de fichiers.

9.8.5 Montage automatique de eCryptfs

Si vous utilisez votre mot de passe de connexion pour envelopper les clés de chiffrement, vous pouvez automatiser le montage de eCryptfs par [PAM \(Pluggable Authentication Modules\)](#).

Insérez la ligne suivante juste avant « `pam_permit.so` » dans « `/etc/pam.d/common-auth` » :

```
auth required pam_ecryptfs.so unwrap
```

Insérez la ligne suivante en tant que toute dernière ligne de « `/etc/pam.d/common-session` » :

```
session optional pam_ecryptfs.so unwrap
```

Insérez la ligne suivante comme première ligne active de « `/etc/pam.d/common-password` » :

```
password required pam_ecryptfs.so
```

C'est assez pratique.



AVERTISSEMENT

Les erreurs de configuration de [PAM](#) peuvent vous interdire l'accès à votre propre système. Consultez Chapitre 4.



Attention

Si vous utilisez votre mot de passe de connexion pour encapsuler les clés de chiffrement, vos données chiffrées sont aussi sûres que l'est votre mot de passe de connexion (consultez Section 4.3). À moins que vous ne preniez le soin de définir un [mot de passe fort](#), vos données encourent un risque lorsque quelqu'un lance un logiciel de [casse de mots de passe](#) après avoir volé votre ordinateur (consultez Section 4.7.4).

9.9 Le noyau

Debian distribue des [noyaux Linux](#) modulaires sous forme de paquets pour les architectures prises en compte.

9.9.1 Noyau Linux 2.6/3.x

Les noyaux Linux 2.6/3.x possèdent quelques fonctionnalités remarquables comparés à 2.4.

- Les périphériques sont créés par le système udev (consultez Section 3.3).
- Les accès en lecture et écriture aux périphériques CD et DVD IDE CD/DVD n'utilisent plus le module `ide-scsi`.
- Les fonctions de filtrage des paquets réseau utilisent les modules `iptables` du noyau.

Le saut de version de Linux passant de 2.6.39 à 3.0 n'est pas lié à des changements technologiques majeurs, mais à la célébration de son 20ème anniversaire.

9.9.2 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 9.6.3)
- Consultez « `kernel-parameters.txt(.gz)` » et d'autres documents en rapport dans la documentation du noyau de Linux (« `/usr/share/doc/linux-doc-3.*/Documentation/filesystems/*` ») fournie par le paquet `linux-doc-3.*`.

9.9.3 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

Pour compiler certains programmes spécifiques au noyau, comme les modules du noyau, à partir de sources externes et le démon de montage automatique (`amd`), vous devez inclure dans votre ligne de commandes un chemin vers les en-têtes du noyau correspondants, par exemple « `-I/usr/src/linux-version-particulière/include/` ». `module-assistant(8)` (ou sa forme abrégée `m-a`) aide les utilisateurs à construire et à installer facilement des paquets de modules pour un ou plusieurs noyaux personnalisés.

9.9.4 Compiler le noyau et les modules associés

Debian a sa propre manière de compiler le noyau et les modules associés.

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).

paquet	popcon	taille	description
build-essential	I:451	20	paquets essentiels pour la construction de paquets Debian : make, gcc, ...
bzip2	V:238, I:933	184	utilitaires de compression et de décompression des fichiers bz2
libncurses5-dev	V:14, I:150	1027	bibliothèques de développement et documentations pour ncurses
git	V:259, I:420	28761	git :: système distribué de contrôle de version utilisé par le noyau de Linux
fakeroot	V:32, I:496	215	fournit l'environnement fakeroot pour construire le paquet sans être administrateur (« root »)
initramfs-tools	V:151, I:990	105	outil pour construire une image mémoire initiale (« initramfs ») (spécifique à Debian)
dkms	V:70, I:207	282	prise en charge dynamique des modules du noyau (dynamic kernel module support : DKMS) (générique)
devscripts	V:10, I:66	2070	scripts d'assistance pour un responsable de paquet Debian (spécifique à Debian)

Table 9.25 – Liste des paquets-clés à installer pour la compilation du noyau sur un système Debian

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.

Note

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.

9.9.5 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_<architecture>_<jeu_de_fonctionnalités>_<savoir> » 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](#) ;
- Manuel du noyau Linux Debian : <http://kernel-handbook.alioth.debian.org>.

9.9.6 Pilotes de matériel et microprogramme

Le pilote de matériel est le code s'exécutant sur le système cible. La plupart des pilotes de matériel sont maintenant disponibles sous forme de logiciels libres et font partie des paquets Debian du noyau dans la section `main`.

- 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`)

Le microprogramme est le code ou les données chargées sur le périphérique (par exemple le [microcode](#) de processeur, le code de rendu exécuté sur les processeurs graphiques, ou les données de [FPGA](#) ou de [CPLD](#), etc.) Certains paquets de microprogramme sont disponibles sous forme de logiciel libre, mais beaucoup de paquets de microprogramme ne le sont pas car ils contiennent des données binaires sans source.

- `firmware-linux-free` (`main`)
- `firmware-linux-nonfree` (`non libre`)
- `firmware-linux-*` (`non libre`)
- `*-firmware` (`non libre`)
- `intel-microcode` (`non libre`)
- `amd64-microcode` (`non libre`)

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](#).

9.10 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> .

9.10.1 Outils de virtualisation

Il y a plusieurs paquets associés à la [virtualisation](#) et à l'[émulation](#) du système sous Debian au-delà d'un simple [chroot](#). Certains paquets vous facilitent la mise en place d'un tel système.

Consultez l'article de Wikipedia [Comparaison de machines pour plateforme virtuelle](#) pour une comparaison détaillée entre les différentes solutions de plateformes de virtualisation.

paquet	popcon	taille	description
schroot	V:6, I:10	2653	outil spécialisé pour l'exécution d'un paquet binaire de Debian dans un chroot
sbuild	V:1, I:3	283	outil pour construire des paquets binaires de Debian depuis les sources Debian
pbuilder	V:2, I:18	962	constructeur personnel de paquets pour des paquets Debian
debootstrap	V:6, I:68	259	amorcer un système Debian de base (écrit en sh)
cdebootstrap	V:0, I:3	112	amorcer un système Debian (écrit en C)
virt-manager	V:8, I:31	7426	Gestionnaire de machine virtuelle : application du gestionnaire de bureau pour la gestion des machines virtuelles
libvirt-clients	V:25, I:47	1907	programmes pour la bibliothèque libvirt
bochs	V:0, I:1	4086	Bochs : émulateur PC IA-32
qemu	I:41	518	QEMU : émulateur de processeur générique rapide
qemu-system	I:43	91	QEMU : binaires pour l'émulation d'un système complet
qemu-user	V:2, I:42	70592	QEMU : binaires pour l'émulation en mode utilisateur
qemu-utils	V:8, I:94	6657	QEMU : utilitaires
qemu-kvm	V:18, I:68	100	KVM : virtualisation complète sur les plateformes x86 ayant une virtualisation assistée par le matériel
virtualbox	V:51, I:64	72786	VirtualBox : solution de virtualisation x86 sur i386 et amd64
xen-tools	V:0, I:7	666	outils pour gérer le serveur virtuel XEN de Debian
wine	V:25, I:106	174	Wine : implémentation de l'API Windows (suite standard)
dosbox	V:3, I:21	2755	DOSBox : émulateur x86 avec graphisme Tandy/Herc/CGA/EGA/VGA/SVGA, son et DOS
dosemu	V:1, I:4	4891	DOSEMU : l'émulateur DOS de Linux
vzctl	V:2, I:4	1070	OpenVZ solution de virtualisation de serveur - outils de contrôle
vzquota	V:2, I:4	222	OpenVZ solution de virtualisation de serveur - outils de quota
lxc	V:8, I:13	1297	Conteneurs Linux outils de l'espace utilisateur

Table 9.26 – Liste des outils de virtualisation

9.10.2 Étapes de la virtualisation

Note

Certaines fonctionnalités décrites ici ne sont disponibles que dans squeeze ou versions supérieures.

Note

Les noyaux par défaut de Debian prennent en charge [KVM](#) depuis Lenny.

La [virtualisation](#) met en œuvre plusieurs étapes :

- 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 9.6.1](#) et [Section 9.6.5](#)).
 - `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 9.6.3](#)).
 - Les images disques prises en charge par [QEMU](#) seront montées en tant que [périphériques réseau en mode bloc](#) (consultez [Section 9.10.3](#)).
- Peupler le système de fichiers cible avec les données requises.
 - L'utilisation de programmes tels que `debootstrap` et `cdebootstrap` facilite ce processus (consultez [Section 9.10.4](#)).
 - 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](#).

9.10.3 Monter le fichier image du disque virtuel

Pour le fichier image disque brut, consultez [Section 8.3](#).

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 9.6.3](#)). 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)`.

9.10.4 Système protégé (chroot)

chroot(8) propose la plupart des moyens simples pour faire tourner simultanément plusieurs instances de l'environnement GNU/Linux sur un même système sans redémarrer.

**Attention**

Les exemple ci-dessous supposent que le système père et le système chroot partagent tous les deux la même architecture de processeur.

Vous pouvez apprendre comment configurer et utiliser chroot(8) en lançant le programme pbuilder(8) sous script(1) comme suit :

```
$ sudo mkdir /sid-root
$ sudo pbuilder --create --no-targz --debug --buildplace /sid-root
```

Vous pouvez voir comment debootstrap(8) ou cdebootstrap(1) installent les données du système pour l'environnement sid sous « /sid-root ».

ASTUCE

debootstrap(8) ou cdebootstrap(1) sont utilisés pour [installer Debian](#) avec l'installateur Debian. Ils peuvent aussi être utilisés pour installer Debian sur un système sans utiliser de disque d'installation de Debian, mais en provenance d'une autre distribution GNU/Linux.

```
$ sudo pbuilder --login --no-targz --debug --buildplace /sid-root
```

Vous pouvez voir ci-après comment un shell système tournant sous l'environnement sid est créé :

1. Copier la configuration locale (« /etc/hosts », « /etc/hostname », « /etc/resolv.conf »)
2. Monter le système de fichiers « /proc »
3. Monter le système de fichiers « /dev/pts »
4. Créer « /usr/sbin/policy-rc.d » qui existe toujours avec 101
5. Lancer « chroot /sid-root bin/bash -c 'exec -a -bash bin/bash' »

Note

Certains programmes sous chroot peuvent demander l'accès à davantage de fichiers du système parent pour fonctionner que ceux que fournit pbuilder. Par exemple, on peut avoir besoin de monter liés (« bind-mount ») ou de copier « /sys », « /etc/passwd », « /etc/group », « /var/run/utmp », « /var/log/wtmp », etc.

Note

Le fichier « /usr/sbin/policy-rc.d » évite que des programmes démons ne soient démarrés automatiquement sur le système Debian. Consultez « /usr/share/doc/sysv-rc/README.policy-rc.d.gz ».

ASTUCE

Le but d'origine du paquet spécialisé d'environnement protégé (chroot) pbuilder est de construire une système protégé et de construire un paquet depuis cet environnement. C'est le système idéal à utiliser pour vérifier que les dépendances de construction d'un paquet sont correctes et pour s'assurer que les dépendances de construction erronées ou non nécessaires n'existent pas dans le paquet résultant.

ASTUCE

De même, le paquet schroot peut vous donner l'idée de faire tourner un système chroot i386 sous le système parent amd64.

9.10.5 Systèmes de bureaux multiples

Je vous recommande d'utiliser [QEMU](#) ou [VirtualBox](#) sur un système Debian stable afin de faire tourner des systèmes de bureau multiples en utilisant la [virtualisation](#). Cela vous permet de lancer des applications de bureau de Debian unstable et testing sans les risques qui leur sont habituellement associés.

Comme [QEMU](#) pur est très lent, il est recommandé de l'accélérer avec [KVM](#) lorsque le système hôte le prend en charge.

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:908, I:999	2770	.tar	tar(1)	archivageur standard (standard de fait)
cpio	V:305, I:998	720	.cpio	cpio(1)	archivageur de style UNIX System V, utiliser avec find(1)
binutils	V:177, I:715	23348	.ar	ar(1)	archivageur pour la création de bibliothèques statiques
fastjar	V:5, I:56	172	.jar	fastjar(1)	archivageur pour Java (semblable à zip)
pax	V:17, I:54	170	.pax	pax(1)	nouvel archivageur standard POSIX, compromis entre tar et cpio
gzip	V:877, I:999	231	.gz	gzip(1) , zcat(1) , ...	utilitaire de compression GNU LZ77 (standard de fait)
bzip2	V:238, I:933	184	.bz2	bzip2(1) , bzip2cat(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:4, I:60	126	.lzma	lzma(1)	utilitaire de compression LZMA avec un plus haut taux de compression que gzip(1) (obsolète)
xz-utils	V:322, I:952	516	.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)
p7zip	V:38, I:175	934	.7z	7zr(1) , p7zip(1)	archivageur de fichiers 7-Zip avec un haut taux de compression (compression LZMA)
p7zip-full	V:175, I:527	4407	.7z	7z(1) , 7za(1)	archivageur de fichiers 7-Zip avec un haut taux de compression (compression LZMA et autres)
lzop	V:5, I:44	97	.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:47, I:395	608	.zip	zip(1)	InfoZIP : outil d'archive et de compression DOS
unzip	V:271, I:791	534	.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).

Note

L'archive tar(1) utilise l'extension de fichier « .tgz » ou « .tar.gz ».

Note

L'archive tar(1) avec compression xz utilise l'extension de fichier « .txz » ou « .tar.xz ».

Note

La méthode de compression habituelle des outils FOSS tels que tar(1) a été modifiée de la manière suivante :
gzip → bzip2 → xz

Note

cp(1), scp(1) et tar(1) peuvent avoir certaines limitations pour des fichiers spéciaux. cpio(1) est plus souple.

Note

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.

Note

La structure interne des fichiers de données de Libreoffice est composée de fichiers « .jar ».

10.1.2 Outils de copie et de synchronisation

Voici une liste d'outils simples de copie et de sauvegarde sur le système Debian :

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

L'exécution du script bkup mentionné dans Section 10.2.3 avec l'option « -gl » sous cron(8) fournira une fonctionnalité très semblable à dumpfs de Plan9 pour l'archivage des données statiques.

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.

paquet	popcon	taille	outil	fonction
coreutils	V:881, I:999	15103	GNU cp	copier localement des fichiers et répertoires (« -a » pour récursif)
openssh-client	V:818, I:996	4106	scp	copier à distance des fichiers et des répertoires (client, « -r » pour récursif)
openssh-server	V:677, I:807	883	sshd	copier à distance des fichiers et des répertoires (serveur distant)
rsync	V:234, I:595	691	-	synchronisation et sauvegarde distantes unidirectionnelles
unison	V:4, I:20	2213	-	synchronisation et sauvegarde distantes bidirectionnelles
obnam	V:1, I:2	1061	-	sauvegarde incrémentale (distante)
rdiff-backup	V:9, I:18	704	-	sauvegarde incrémentale (distante)

Table 10.2 – Liste des outils de copie et de synchronisation

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

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 9.3.9). 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 /chemin/vers/ \
  -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 /chemin/vers/horodatage -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 « `uhelp=` » 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 [8.3.6](#)).

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 :

système de fichiers	description d'un scénario typique d'utilisation
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)
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)
système de fichiers MINIX	enregistrement efficace en matière d'espace disque de fichiers de données unix sur disquette
système de fichiers ext2	partage de données sur disque dur avec les anciens systèmes Linux
système de fichiers ext3	partage de données sur disque dur avec les anciens systèmes Linux
système de fichiers ext4	partage de données sur disque dur avec les systèmes Linux actuels

Table 10.3 – Liste de choix de systèmes de fichiers pour des périphériques amovibles avec des scénarios typiques d'utilisation

ASTUCE

Consultez [Section 9.8.1](#) 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 :

- les partitionner avec `fdisk(8)`, `cfdisk(8)` ou `parted(8)` (consultez [Section 9.5.2](#)) 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 $= (4\text{GiB} - 1 \text{ octet})$. Pour certaines applications sur le système 32 bits plus ancien, la taille maximum était même plus faible $(2^{31} - 1)$ octets $= (2\text{Gio} - 1 \text{ octet})$. 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 smb.conf(5) et le HOWTO et guide de référence officiel de Samba 3.x.x ou le paquet samba-doc
système de fichiers monté au travers du réseau NFS avec le noyau Linux	partager des fichiers par « UNIX/Linux Network », consultez exports(5) 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 6.10](#) et [Section 6.11](#).

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 9.7 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 6.3) 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/fetchmailrc », « 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 9.3.14 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 :

paquet	popcon	taille	description
dump	V:1, I:7	341	BSD 4.4 dump(8) et restore(8) pour les systèmes de fichiers ext2/ext3/ext4 .
xfsdump	V:0, I:11	838	sauvegarder et restaurer avec xfsdump(8) et xfsrestore(8) pour le système de fichiers XFS sous GNU/Linux et IRIX
backupninja	V:4, I:4	277	système de sauvegarde meta-backup léger et extensible
sbackup	V:0, I:0	488	suite de sauvegarde simple pour le bureau GNOME
bacula-common	V:8, I:19	1291	Bacula : sauvegarde, restauration et vérification par le réseau - fichiers communs
bacula-client	I:5	157	Bacula : sauvegarde, restauration et vérification par le réseau - métapaquet du client
bacula-console	V:1, I:7	64	Bacula : sauvegarde, restauration et vérification par le réseau - console en mode texte
bacula-server	I:2	160	Bacula : sauvegarde, restauration et vérification par le réseau - métapaquet du serveur
amanda-common	V:1, I:3	7418	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	998	Amanda : Advanced Maryland Automatic Network Disk Archiver (Client)
amanda-server	V:0, I:0	1046	Amanda : Advanced Maryland Automatic Network Disk Archiver (Serveur)
backuppc	V:4, I:5	2232	BackupPC est un système de hautes performances pour effectuer la sauvegarde de PC au niveau de l'entreprise (basé sur disques)
backup-manager	V:1, I:2	543	outil de sauvegarde en ligne de commandes
backup2l	V:1, I:1	113	outil de sauvegarde et restauration de faible maintenance pour des supports pouvant être montés (basé sur disque)

Table 10.5 – Liste de suites d'utilitaires de sauvegarde

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.
- [sbackup](#) et [keep](#) fournissent une interface graphique facile pour les utilisateurs de bureau afin d'effectuer des sauvegardes régulières des données de l'utilisateur. Une fonction équivalente peut être réalisée par un simple script (Section 10.2.2) et [cron\(8\)](#).
- [Bacula](#), [Amanda](#) et [BackupPC](#) sont des suites de sauvegardes ayant des fonctionnalités avancées qui sont orientées vers les sauvegardes fréquentes au travers du réseau.

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 [obnam](#) 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 Script d'exemple pour la sauvegarde du système

Pour un système de bureau personnel Debian sur lequel tourne la version `unstable`, je n'ai besoin de protéger que les données personnelles et les données critiques. Je réinstalle cependant le système une fois par an. Je ne vois donc aucune raison de sauvegarder l'ensemble du système ou d'installer un utilitaire de sauvegarde ayant des fonctionnalités avancées.

J'utilise un simple script pour réaliser des archives de sauvegarde et les graver sur CD/DVD en utilisant une interface graphique. Voici un exemple de script pour ce faire :

```
#!/bin/sh -e
# Copyright (C) 2007-2008 Osamu Aoki <osamu@debian.org>, Public Domain
BUUID=1000; USER=osamu # UID and name of a user who accesses backup files
BUDIR="/var/backups"
XDIR0=".+/Mail|.+/Desktop"
XDIR1=".+/.thumbnails|.+/.?Trash|.+/.?[cC]ache|.+/.gvfs|.+/.sessions"
XDIR2=".+/CVS|.+/.git|.+/.svn|.+/.Downloads|.+/.Archive|.+/.Checkout|.+/.tmp"
XSFX=".+\.iso|.+\.tgz|.+\.tar\.gz|.+\.tar\.bz2|.+\.cpio|.+\.tmp|.+\.swp|.+~"
SIZE="+99M"
DATE=$(date --utc +"%Y%m%d-%H%M")
[ -d "$BUDIR" ] || mkdir -p "$BUDIR"
umask 077
dpkg --get-selections \* > /var/lib/dpkg/dpkg-selections.list
debconf-get-selections > /var/cache/debconf/debconf-selections

{
find /etc /usr/local /opt /var/lib/dpkg/dpkg-selections.list \
    /var/cache/debconf/debconf-selections -xdev -print0
find /home/$USER /root -xdev -regextype posix-extended \
    -type d -regex "$XDIR0|$XDIR1" -prune -o -type f -regex "$XSFX" -prune -o \
    -type f -size "$SIZE" -prune -o -print0
find /home/$USER/Mail/Inbox /home/$USER/Mail/Outbox -print0
find /home/$USER/Desktop -xdev -regextype posix-extended \
    -type d -regex "$XDIR2" -prune -o -type f -regex "$XSFX" -prune -o \
    -type f -size "$SIZE" -prune -o -print0
} | cpio -ov --null -O $BUDIR/BU$DATE.cpio
chown $BUUID $BUDIR/BU$DATE.cpio
touch $BUDIR/backup.stamp
```

C'est destiné à être un script d'exemple exécuté avec le compte de l'administrateur.

Vous devrez le modifier et l'exécuter comme suit :

- éditez ce script afin de couvrir l'ensemble de vos données importantes (consultez Section 10.1.5 et Section 10.2) ;
- remplacez « `find ...-print0` » par « `find ...-newer $BUDIR/backup.stamp -print0` » afin d'effectuer des sauvegardes incrémentales ;
- transférez les sauvegardes sur la machine distante en utilisant `scp(1)` ou `rsync(1)` ou gravez-les sur CD/DVD pour plus de sécurité (j'utilise l'interface graphique du bureau GNOME pour graver les CD/DVD. Consultez Section 12.1.8 pour une redondance supplémentaire).

Faites simple !

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` ».

10.2.3 Un script de copie pour la sauvegarde des données

Pour les données se trouvant sous l'arborescence d'un répertoire, une copie avec « `cp -a` » permet une sauvegarde normale.

Pour un gros ensemble de données statiques ne devant pas être réécrites se trouvant dans une arborescence de répertoires telle que celle se trouvant sous le répertoire « `/var/cache/apt/packages/` », les liens physiques avec « `cp -al` » fournissent une alternative à la sauvegarde normale avec une utilisation efficace de l'espace disque.

Voici un script de copie, que j'ai appelé `bkup`, destiné à la sauvegarde de données. Ce script copie tous les fichiers (non-VCS) du répertoire actuel vers le répertoire daté du répertoire parent ou sur une machine distante.

```
#!/bin/sh -e
# Copyright (C) 2007-2008 Osamu Aoki <osamu@debian.org>, Public Domain
fdot(){ find . -type d \( -iname ".?*" -o -iname "CVS" \) -prune -o -print0;}
fall(){ find . -print0;}
mkdircd(){ mkdir -p "$1";chmod 700 "$1";cd "$1">/dev/null;}
FIND="fdot";OPT="-a";MODE="CPIO";HOST="localhost";EXTP="$(hostname -f)"
BKUP="$(basename $(pwd)).bkup";TIME="$(date +%Y%m%d-%H%M%S)";BU="$BKUP/$TIME"
while getopts gcCsStrlLaAxe:h:T f; do case $f in
g)  MODE="GNUCP";; # cp (GNU)
c)  MODE="CPIO";; # cpio -p
C)  MODE="CPIOI";; # cpio -i
s)  MODE="CPIOSSH";; # cpio/ssh
t)  MODE="TARSSH";; # tar/ssh
r)  MODE="RSYNCSSH";; # rsync/ssh
l)  OPT="-alv";; # lien physique (GNU cp)
L)  OPT="-av";; # copier (GNU cp)
a)  FIND="fall";; # tout rechercher
A)  FIND="fdot";; # rechercher ce qui n'est pas CVS/ .???/
x)  set -x;; # trace
e)  EXTP="{OPTARG}";; # hostname -f
h)  HOST="{OPTARG}";; # utilisateur@remotehost.example.com
T)  MODE="TEST";; # tester le mode de recherche
\?) echo "utiliser -x pour une trace."
esac; done
shift $(expr $OPTIND - 1)
if [ $# -gt 0 ]; then
  for x in $@; do cp $OPT $x $x.$TIME; done
elif [ $MODE = GNUCP ]; then
  mkdir -p "../$BU";chmod 700 "../$BU";cp $OPT . "../$BU/"
elif [ $MODE = CPIO ]; then
  mkdir -p "../$BU";chmod 700 "../$BU"
  $FIND|cpio --null --sparse -pvd ../$BU
elif [ $MODE = CPIOI ]; then
  $FIND|cpio -ov --null | ( mkdircd "../$BU"&&cpio -i )
elif [ $MODE = CPIOSSH ]; then
  $FIND|cpio -ov --null|ssh -C $HOST "( mkdircd \"$EXTP/$BU\"&&cpio -i )"
elif [ $MODE = TARSSH ]; then
  (tar cvf - . )|ssh -C $HOST "( mkdircd \"$EXTP/$BU\"&&tar xvpf - )"
elif [ $MODE = RSYNCSSH ]; then
  rsync -aHAXsv ./ "${HOST}:${EXTP}-${BKUP}-${TIME}"
else
  echo "Une autre idée à sauvegarder ?"
  $FIND |xargs -0 -n 1 echo
fi
```

C'est destiné à être un exemple de commandes. Veuillez lire le script et le modifier vous-même avant de l'utiliser.

ASTUCE

J'ai installé ce `bkup` dans mon répertoire « `/usr/local/bin/` ». J'utilise cette commande `bkup`, sans aucune option, depuis mon répertoire de travail lorsque j'ai besoin d'un instantané temporaire de sauvegarde.

ASTUCE

Pour faire un instantané historique d'une arborescence de fichiers sources ou d'une arborescence de fichiers de configuration, il est plus facile et efficace d'utiliser `git(7)` (consultez Section [10.6.5](#)).

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:652, I:999	2088	<code>gpg(1)</code>	GNU Privacy Guard - outil de signature et de chiffrement OpenPGP
gpgv	V:874, I:999	721	<code>gpgv(1)</code>	GNU Privacy Guard - outil de vérification de signature
paperkey	V:0, I:1	54	<code>paperkey(1)</code>	extraire seulement l'information secrète de clés secrètes OpenPGP
cryptsetup	V:25, I:73	355	<code>cryptsetup(8)</code> , ...	utilitaires pour périphérique en mode bloc dm-crypto prenant en charge le chiffrement LUKS
ecryptfs-utils	V:5, I:8	396	<code>ecryptfs(7)</code> , ...	utilitaires pour le chiffrement de systèmes de fichiers empilés ecryptfs
coreutils	V:881, I:999	15103	<code>md5sum(1)</code>	calculer et vérifier un condensé MD5 de message
coreutils	V:881, I:999	15103	<code>sha1sum(1)</code>	calculer et vérifier un condensé SHA1 de message
openssl	V:785, I:988	1255	<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

Consultez la Section [9.8](#) sur [dm-crypto](#) et [ecryptfs](#) qui implémente l'infrastructure de chiffrement automatique des données par l'intermédiaire de modules du noyau de Linux.

10.3.1 Gestion de clés pour GnuPG

Voici les commandes de [GNU Privacy Guard](#) pour la gestion de base des clés :

commande	description
<code>gpg --gen-key</code>	générer une nouvelle clé
<code>gpg --gen-revoke ID_de_mon_utilisateur</code>	générer une clé de révocation pour <code>ID_de_mon_utilisateur</code>
<code>gpg --edit-key ID_utilisateur</code>	éditer la clé de manière interactive, « <code>help</code> » pour obtenir de l'aide
<code>gpg -o fichier --export</code>	exporter toutes les clés vers fichier
<code>gpg --import fichier</code>	importer toutes les clés depuis fichier
<code>gpg --send-keys ID_utilisateur</code>	envoyer la clé de <code>ID_utilisateur</code> vers le serveur de clés
<code>gpg --recv-keys ID_utilisateur</code>	recevoir la clé de <code>ID_utilisateur</code> du serveur de clés
<code>gpg --list-keys ID_utilisateur</code>	afficher la liste des clés de <code>ID_utilisateur</code>
<code>gpg --list-sigs ID_utilisateur</code>	afficher la liste des signatures de <code>ID_utilisateur</code>
<code>gpg --check-sigs ID_utilisateur</code>	vérifier la signature de <code>ID_utilisateur</code>
<code>gpg --fingerprint ID_utilisateur</code>	vérifier l'empreinte de <code>ID_utilisateur</code>
<code>gpg --refresh-keys</code>	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

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

Le greffon gnupg vous permet de lancer GnuPG de manière transparente pour les fichiers ayant l'extension « .gpg », « .asc » et « .pgp ».

```
# aptitude install vim-scripts vim-addon-manager
$ vim-addons install gnupg
```

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

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 toto titi >tutu.md5
$ cat tutu.md5
d3b07384d113edec49eaa6238ad5ff00  toto
c157a79031e1c40f85931829bc5fc552  titi
$ md5sum -c tutu.md5
toto: OK
titi: OK
```

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 :

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 fichier.ancien fichier.nouveau > fichier.patch0
$ diff -u ancien/fichier nouveau/fichier > fichier.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 fichier < fichier.patch0
$ patch -p1 fichier < fichier.patch1
```

10.4.3 Mise à jour par fusion de 3 sources

Si vous avez trois versions d'un code source, vous pouvez effectuer de manière efficace une fusion des trois en utilisant `diff3(1)` de la manière suivante :

```
$ diff3 -m fichier.mien fichier.ancien fichier.votre > fichier
```

paquet	popcon	taille	commande	description
diffutils	V:856, I:978	1327	<code>diff(1)</code>	comparer des fichiers ligne à ligne
diffutils	V:856, I:978	1327	<code>diff3(1)</code>	comparer et fusionner trois fichiers ligne par ligne
vim	V:118, I:393	2374	<code>vimdiff(1)</code>	comparer deux fichiers côte à côte dans vim
patch	V:100, I:928	216	<code>patch(1)</code>	appliquer un fichier de différences (« diff ») à un original
dpatch	V:1, I:17	191	<code>dpatch(1)</code>	gérer une série de rustines (patch) pour les paquets Debian
diffstat	V:20, I:188	70	<code>diffstat(1)</code>	afficher un histogramme des modifications apportées par le fichier de différences
patchutils	V:19, I:180	223	<code>combinediff(1)</code>	créer une rustine (patch) cumulative à partir de deux rustines incrémentales
patchutils	V:19, I:180	223	<code>dehtmldiff(1)</code>	extraire un fichier de différences d'une page HTML
patchutils	V:19, I:180	223	<code>filterdiff(1)</code>	extraire ou exclure des différences d'un fichier de différences
patchutils	V:19, I:180	223	<code>fixcvsdiff(1)</code>	corriger les fichiers de différences créés par CVS que <code>patch(1)</code> interprète mal
patchutils	V:19, I:180	223	<code>flipdiff(1)</code>	échanger l'ordre de deux rustines
patchutils	V:19, I:180	223	<code>grepdiff(1)</code>	afficher quels sont les fichiers modifiés par une rustine correspondant une expression rationnelle
patchutils	V:19, I:180	223	<code>interdiff(1)</code>	afficher les différences entre deux fichiers de différence unifiés
patchutils	V:19, I:180	223	<code>lsdiff(1)</code>	afficher quels sont les fichiers modifiés par une rustine
patchutils	V:19, I:180	223	<code>recountdiff(1)</code>	recalculer les nombres et les décalages dans un contexte unifié de fichiers de différences
patchutils	V:19, I:180	223	<code>rediff(1)</code>	corriger les décalages et les nombres d'un fichier de différences édité manuellement
patchutils	V:19, I:180	223	<code>splitdiff(1)</code>	séparer les rustines incrémentales
patchutils	V:19, I:180	223	<code>unwrapdiff(1)</code>	réparer les correctifs dont les mots ont été coupés
wiggle	V:0, I:0	166	<code>wiggle(1)</code>	appliquer les rustines rejetées
quilt	V:4, I:44	711	<code>quilt(1)</code>	gérer une série de rustines
meld	V:13, I:42	3049	<code>meld(1)</code>	comparer et fusionner des fichiers (GTK)
dirdiff	V:0, I:3	144	<code>dirdiff(1)</code>	afficher les différences et fusionner les modifications entre deux arbres de répertoires
docdiff	V:0, I:0	573	<code>docdiff(1)</code>	comparer deux fichiers mot par mot ou caractère par caractère
imediff2	V:0, I:0	34	<code>imediff2(1)</code>	outil en plein écran pour fusionner dans les deux sens
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
wdiff	V:5, I:85	643	<code>wdiff(1)</code>	afficher les différences de mots entre deux fichiers texte

Table 10.10 – Liste d'outils destinés à fusionner du code source

10.5 Systèmes de contrôle de version

Voici un résumé des [systèmes de contrôle de version \(VCS\)](#) sur le système Debian.

Note

Si vous débutez avec les systèmes VCS, vous devriez commencer votre apprentissage avec **Git**, dont la popularité croît rapidement.

paquet	popcon	taille	outil	type du VCS	commentaire
cssc	V:0, I:2	1979	CSSC	local	clone de SCCS UNIX (obsolète)
rcs	V:4, I:24	547	RCS	local	« SCCS UNIX en mieux »
cvs	V:7, I:60	4574	CVS	distant	standard précédent de VCS distant
subversion	V:40, I:161	4743	Subversion	distant	« CVS en mieux », de fait le nouveau standard de VCS distant
git	V:259, I:420	28761	Git	distribué	DVCS rapide en C (utilisé par le noyau de Linux et d'autres)
mercurial	V:13, I:64	205	Mercurial	distribué	DVCS en Python avec un peu de C
bzd	V:4, I:23	72	Bazaar	distribué	DVCS influencé par tla écrit en Python (utilisé par Ubuntu)
darcs	V:0, I:7	36373	Darcs	distribué	DVCS avec une algèbre intelligente des rustines (lent)
tla	V:0, I:8	1011	GNU arch	distribué	DVCS principalement de Tom Lord (Historique)
monotone	V:0, I:0	5815	Monotone	distribué	DVCS en C++
tkcvs	V:0, I:1	1498	CVS, ...	distant	Interface graphique d'affichage d'une arborescence de dépôt VCS (CVS, Subversion, RCS)
gitk	V:9, I:51	1391	Git	distribué	Interface graphique d'affichage d'une arborescence de dépôt VCS (Git)

Table 10.11 – Liste d'outils pour les systèmes de contrôle de version

VCS est parfois appelé système de contrôle de révision (RCS), ou gestion de configuration logicielle (SCM).

Un VCS distribué comme Git est, de nos jours, l'outil de choix. CVS et Subversion peuvent continuer à être utiles pour se joindre à certaines activités existantes sur les logiciels libres.

Debian fournit des services de VCS libres avec le [service Debian Alioth](#). Il prend en charge pratiquement tous les VCS. Sa documentation se trouve sur <http://wiki.debian.org/fr/Alioth>.

Il existe quelques bases pour la création d'un accès à une archive VCS.

- Utilisez « `umask 002` » (consultez Section [1.2.4](#))
- Rattacher tous les fichiers d'archives VCS à un groupe pertinent
- Activer l'identifiant de groupe de tous les répertoires d'archive VCS (schéma de création de fichiers semblable à BSD), consultez Section [1.2.3](#))
- Rattacher au groupe l'utilisateur partageant l'archive VCS

10.5.1 Comparaison de commandes VCS

Voici une comparaison très simplifiée des commandes natives de VCS destinées à donner une vue d'ensemble. La séquence de commandes typique peut demander des options et des paramètres.

Git	CVS	Subversion	Fonction
git init	cvcs init	svn create	créer le dépôt (local)
-	cvcs login	-	se connecter au dépôt distant
git clone	cvcs co	svn co	vérifiez le dépôt distant comme arborescence de travail
git pull	cvcs up	svn up	mettre à jour l'arborescence de travail depuis le dépôt distant
git add .	cvcs add	svn add	ajouter des fichiers de l'arborescence de travail vers le VCS
git rm	cvcs rm	svn rm	supprimer du VCS des fichiers de l'arborescence de travail
-	cvcs ci	svn ci	déposer des modifications au dépôt distant
git commit -a	-	-	déposer des modifications sur le dépôt local
git push	-	-	mettre à jour le dépôt distant à l'aide du dépôt local
git status	cvcs status	svn status	afficher l'état de l'arborescence de travail d'après le VCS
git diff	cvcs diff	svn diff	diff <dépôt_de_référence> <arborescence_de_travail>
git repack -a -d;git prune	-	-	réempaqueter le dépôt local en un seul paquet
gitk	tkcvs	tkcvs	Interface graphique d'affichage de l'arborescence d'un dépôt VCS

Table 10.12 – Comparaison des commandes natives de VCS

**Attention**

L'appel d'une sous-commande git directement avec « git-xyz » depuis la ligne de commandes est devenu obsolète depuis début 2006.

ASTUCE

S'il existe un fichier exécutable « git-foo » dans le chemin spécifié par « \$PATH » alors entrer la commande « git foo » sans guillemets appellera ce « git-foo ». C'est une fonctionnalité de la commande « git ».

ASTUCE

Des outils avec interface graphique comme tkcvs(1) et gitk(1) vous seront d'une aide appréciable pour le suivi de l'historique de révision des fichiers. L'interface Web fournie par de nombreuses archives publiques pour parcourir leurs dépôts est assez utile aussi.

ASTUCE

Git peut travailler directement avec différents dépôts VCS tels que ceux proposés par CVS et Subversion et il fournit un dépôt local pour les modifications locales à l'aide des paquets git-cvs et git-svn. Consultez [git pour les utilisateurs de CVS](#) et Section 10.6.4.

ASTUCE

Git possède des commandes qui n'ont pas d'équivalent dans CVS ni Subversion : « fetch », « rebase », « cherry-pick », ...

10.6 Git

Git peut tout faire pour la gestion du code source soit local ou distant. Cela signifie que vous pouvez enregistrer les modifications de code source sans avoir besoin d'une connexion réseau avec le dépôt distant.

10.6.1 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 nom.utilisateur "Prénom Nom"
$ git config --global nom-utilisateur.email votre-nom@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
```

10.6.2 Références de Git

Consultez ce qui suit :

- [page de manuel : git\(1\)](/usr/share/doc/git-doc/git.html) (`/usr/share/doc/git-doc/git.html`)
- [Manuel de l'utilisateur de Git](/usr/share/doc/git-doc/user-manual.html) (`/usr/share/doc/git-doc/user-manual.html`)
- [Un tutoriel d'introduction à git](/usr/share/doc/git-doc/gittutorial.html) « A tutorial introduction to git » (`/usr/share/doc/git-doc/gittutorial.html`)
- [Un tutoriel d'introduction à git : deuxième partie](/usr/share/doc/git-doc/gittutorial-2.html) « A tutorial introduction to git: part two » (`/usr/share/doc/git-doc/gittutorial-2.html`)
- [Utilisation de tous les jours de GIT en 20 commandes](/usr/share/doc/git-doc/everyday.html) « Everyday GIT With 20 Commands Or So » (`/usr/share/doc/git-doc/everyday.html`)
- [git pour les utilisateurs de CVS](/usr/share/doc/git-doc/gitcvms-migration.html) « git for CVS users » (`/usr/share/doc/git-doc/gitcvms-migration.html`)
 - Cette documentation décrit aussi comment configurer un serveur tel que CVS et extraire d'anciennes données de CVS vers Git.
- [Autres ressources git disponibles sur le web](#)
 - [Git - Cours pour les familiers de Subversion](#) « SVN Crash Course »
 - [La magie de Git](/usr/share/doc/gitmagic/html/index.html) « 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`.

paquet	popcon	taille	commande	description
git-doc	I:20	10381	N/A	Documentation officielle de git
gitmagic	I:2	719	N/A	« Git Magic », le guide le plus facile à comprendre pour Git
git	V:259, I:420	28761	git(7)	Git, système de contrôle de version rapide, évolutif et distribué
gitk	V:9, I:51	1391	gitk(1)	Interface graphique de navigateur de dépôt Git avec historique
git-gui	V:3, I:29	2149	git-gui(1)	Interface graphique pour Git (pas d'historique)
git-svn	V:2, I:29	931	git-svnimport(1)	importer des données venant de Subversion dans Git
git-svn	V:2, I:29	931	git-svn(1)	fournit un fonctionnement bidirectionnel entre Subversion et Git
git-cvs	V:0, I:13	1053	git-cvimport(1)	importer des données venant de CVS dans Git
git-cvs	V:0, I:13	1053	git-cvsexportcommit(1)	exporter une proposition (« commit ») récupéré de Git vers un CVS
git-cvs	V:0, I:13	1053	git-cvsserver(1)	émulateur de serveur CVS pour Git
git-email	V:0, I:13	733	git-send-email(1)	envoyer une série de rustines sous forme de courriel à partir de Git
stgit	V:0, I:1	1692	stg(1)	quilt par-dessus git (Python)
git-buildpackage	V:2, I:11	3850	git-buildpackage(1)	automatise la mise en paquet Debian avec Git
guilt	V:0, I:0	147	guilt(7)	quilt par-dessus git (SH/AWK/SED/...)

Table 10.13 – Liste des paquets et des commandes relatifs à git

10.6.3 Commandes de Git

Même si votre source amont utilise un VCS différent, c'est une bonne idée d'utiliser [git\(1\)](#) pour l'activité locale parce qu'il vous permet de gérer votre copie locale de l'arborescence des sources sans connexion réseau amont. Voici quelques paquets et commandes utilisés avec [git\(1\)](#).

ASTUCE

Avec [git\(1\)](#), vous travaillez sur une branche locale avec de nombreuses commit et vous utilisez quelque chose comme « [git rebase -i master](#) » pour réorganiser plus tard l'historique des modifications. Cela vous permet de faire des modifications propres de l'historique. Consultez [git-rebase\(1\)](#) et [git-cherry-pick\(1\)](#).

ASTUCE

Si vous désirez retrouver un répertoire propre sans perdre l'état actuel du répertoire de travail, vous pouvez utiliser « [git stash](#) ». Consultez [git-stash\(1\)](#).

10.6.4 Git pour le dépôt Subversion

Un dépôt Subversion en « [svn+ssh://svn.example.org/project/module/trunk](#) » peut être extrait vers un dépôt Git local en « [./dest](#) » puis renvoyé vers le dépôt subversion. Par exemple :

```
$ git svn clone -s -rHEAD svn+ssh://svn.example.org/project dest
$ cd dest
... effectuer des modifications
$ git commit -a
... travailler encore avec git
```

```
$ git svn dcommit
```

ASTUCE

L'utilisation de « rHEAD » nous permet d'éviter de devoir cloner l'ensemble du contenu de l'historique du dépôt Subversion.

10.6.5 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 [9.2.10](#).

10.7 CVS

Consultez ce qui suit :

- `cvs(1)`
 - « `/usr/share/doc/cvs/html-cvsclient` »
 - « `/usr/share/doc/cvs/html-info` »
 - « `/usr/share/doc/cvsbook` »
 - « `info cvs` »
-

10.7.1 Configuration du dépôt CVS

La configuration suivante permet de ne faire des dépôts (« commit ») vers le dépôt du CVS que par les membres du groupe « SRC » et l'administration du CVS que par un membre du groupe « staff », réduisant ainsi la chance de se détruire soi-même.

```
# cd /var/lib; umask 002; mkdir cvs
# export CVSR00T=/srv/cvs/projet
# cd $CVSR00T
# chown root:src .
# chmod 2775 .
# cvs -d $CVSR00T init
# cd CVSR00T
# chown -R root:staff .
# chmod 2775 .
# touch val-tags
# chmod 664 history val-tags
# chown root:src history val-tags
```

ASTUCE

Vous pouvez restreindre la création d'un nouveau projet en modifiant le propriétaire du répertoire « \$CVSR00T » pour « root:staff » et ses permissions à « 3775 ».

10.7.2 Accès local à CVS

La variable « \$CVSR00T » pointe vers le dépôt par défaut. Ce qui suit définit « \$CVSR00T » pour l'accès local :

```
$ export CVSR00T=/srv/cvs/project
```

10.7.3 Accès distant à CVS avec pserver

De nombreux serveurs CVS d'accès public fournissent un accès distant en lecture avec le nom de compte « anonymous » par l'intermédiaire du service pserver. Par exemple, le contenu du site web Debian est maintenu par l'intermédiaire du [projet webwml](#) sous le CVS du service alioth de Debian. Ce qui suit permet de configurer « \$CVSR00T » pour un accès distant à ce dépôt CVS :

```
$ export CVSR00T=:pserver:anonymous@anonscm.debian.org:/cvs/webwml
$ cvs login
```

Note

Comme pserver est sujet à des attaques d'espionnage et peu sûr, l'accès en écriture est habituellement désactivé par les administrateurs du serveur.

10.7.4 Accès distant à CVS avec ssh

Ce qui suit définit « \$CVS_RSH » et « \$CVSR00T » pour l'accès distant avec SSH au dépôt CVS du [projet webwml](#) :

```
$ export CVS_RSH=ssh
$ export CVSR00T=:ext:account@cvs.alioth.debian.org:/cvs/webwml
```

Vous pouvez aussi utiliser une clé publique d'authentification pour SSH, ce qui élimine l'invite à distance pour l'entrée du mot de passe.

10.7.5 Importer une nouvelle source dans CVS

Créez un nouvel emplacement d'arborescence locale des sources dans « ~/chemin/vers/module1 » par la commande suivante :

```
$ mkdir -p ~/chemin/vers/module1; cd ~/chemin/vers/module1
```

Mettez des fichiers dans la nouvelles arborescence locale de sources se trouvant en « ~/chemin/vers/module1 ».

Importez-les vers CVS avec les paramètres suivants :

— Module name : « module1 »

— Vendor tag: « Branche-principale » (étiquette pour toute la branche)

— Release tag: « Version-initiale » (étiquette pour une version spécifique)

```
$ cd ~/chemin/vers/module1
$ cvs import -m "Start module1" module1 Branche-principale Version-initiale
$ rm -Rf . # optionnel
```

10.7.6 Permissions des fichiers du dépôt CVS

CVS n'écrase pas un fichier du dépôt actuel mais le remplace par un autre. La permission en écriture vers le répertoire du dépôt est donc critique. Lancez ce qui suit, pour chaque nouveau module de « module1 » dans le dépôt à « /srv/cvs/projet », afin de vous assurer, si nécessaire, que cette condition est remplie :

```
# cd /srv/cvs/projet
# chown -R root:src module1
# chmod -R ug+rwX module1
# chmod 2775 module1
```

10.7.7 Flux de travail avec CVS

Voici un exemple de flux de travail typique utilisant CVS.

Vérifiez tous les modules disponibles du projet CVS pointé par « \$CVSR00T » comme suit :

```
$ cvs rls
CVSR00T
module1
module2
...
```

Récupérez le « module1 » depuis son répertoire par défaut « ./module1 » de la manière suivante :

```
$ cd ~/path/to
$ cvs co module1
$ cd module1
```

Effectuez les modifications de contenu au besoin.

Vérifiez les modifications par un équivalent de « diff -u [repository] [local] » de la manière suivante :

```
$ cvs diff -u
```

Vous-vous apercevez que vous avez cassé sévèrement un fichier appelé « fichier_à_restaurer » mais que les autres fichiers sont corrects.

Écrasez le fichier « fichier_à_restaurer » avec une copie propre venant du CVS de la manière suivante :


```
$ cvs up -C fichier_à_restaurer
```

Enregistrez l'arborescence source locale vers le CVS comme suit :

```
$ cvs ci -m "Décrire les modifications"
```

Créez et ajoutez le fichier « `fichier_à_ajouter` » au CVS comme suit :

```
$ vi fichier_à_ajouter
$ cvs add fichier_à_ajouter
$ cvs ci -m "Ajout de fichier_à_ajouter"
```

Fusionnez la dernière version depuis le CVS comme suit :

```
$ cvs up -d
```

Recherchez les lignes commençant par « `C filename` » qui indiquent des modifications en conflit.

Recherchez du code non modifié dans « `.#nom_fichier.version` ».

Recherchez « `<<<<<<<` » et « `>>>>>>>` » dans les fichiers qui indiquent des conflits de modifications.

Au besoin, éditez les fichiers pour résoudre les conflits.

Ajoutez l'étiquette de version (« `release tag` ») « `Version-1` » comme suit :

```
$ cvs ci -m "dernière proposition pour Version-1"
$ cvs tag Version-1
```

Poursuivez l'édition.

Supprimez l'étiquette de version « `Version-1` » comme suit :

```
$ cvs tag -d Version-1
```

Vérifiez les modifications du CVS comme suit :

```
$ cvs ci -m "vraiment la dernière proposition pour Version-1"
```

Ajoutez de nouveau, de la manière suivante, l'étiquette « `Version-1` » à HEAD de « `main` » du CVS mis à jour :

```
$ cvs tag Version-1
```

Créez une branche avec une étiquette de branche « collante » « `Version-initiale-corrections-de-bogues` » depuis la version d'origine pointée par l'étiquette « `Version-initiale` » et exportez-la vers le répertoire `~/chemin/vers/ancien` comme suit :

```
$ cvs rtag -b -r Version-initiale Version-initiale-corrections-de-bogues module1
$ cd ~/chemin/vers
$ cvs co -r Version-initiale-corrections-de-bogues -d ancien module1
$ cd ancien
```

ASTUCE

Utilisez « `-D 2005-12-20` » (format de date [ISO 8601](#)) plutôt que « `-r Version-initiale` » pour indiquer une date particulière comme point de branche.

Travaillez sur cette arborescence locale possédant l'étiquette « collante » « `Version-initiale-corrections-de-bogues` » qui est basée sur la version d'origine.

Travaillez vous-même sur cette branche... jusqu'à ce que quelqu'un d'autre rejoigne cette branche « `Version-initiale-corrections-de-bogues` ».

Synchronisez les fichiers modifiés par d'autres sur cette branche, en créant de nouveaux répertoires si nécessaire, de la manière suivante :

```
$ cvs up -d
```

Au besoin, éditez les fichiers pour résoudre les conflits.

Vérifiez les modifications du CVS comme suit :

```
$ cvs ci -m "checked into this branch"
```

Mettez à jour l'arborescence locale depuis HEAD de main en supprimant l'étiquette collante (« sticky tag ») (« -A ») et sans expansion de mots-clés (« -kk ») comme suit :

```
$ cvs up -d -kk -A
```

Mettez à jour l'arborescence locale (contenu = HEAD de main) en fusionnant depuis la branche « Version-initiale-corrections-de-bogues » et sans expansion des mots-clés en faisant ce qui suit :

```
$ cvs up -d -kk -j Version-initiale-corrections-de-bogues
```

Corrigez les conflits avec l'éditeur.

Vérifiez les modifications du CVS comme suit :

```
$ cvs ci -m "Version-initiale-corrections-de-bogues fusionnée"
```

Réalisez une archive de la manière suivante :

```
$ cd ..
$ mv ancien ancien-module1-corrections_bogues
$ tar -cvzf ancien-module1-corrections_bogues.tar.gz ancien-module1-corrections_bogues
$ rm -rf ancien-module1-corrections_bogues
```

ASTUCE

La commande « cvs up » peut prendre l'option « -d » pour créer de nouveaux répertoires et l'option « -P » pour supprimer les répertoires vides.

ASTUCE

Vous pouvez ne vérifier qu'un sous-répertoire de « module1 » en indiquant son nom comme, par exemple, « cvs co module1/sous-répertoire ».

option	signification
-n	simulation, pas d'effet
-t	afficher les messages montrant les étapes des actions de cvs

Table 10.14 – Options importantes des commandes CVS (à utiliser comme premier(s) paramètre(s) à cvs(1))

10.7.8 Derniers fichiers venant du CVS

Pour obtenir les derniers fichiers du CVS, utilisez « tomorrow » comme suit :

```
$ cvs ex -D tomorrow nom_module
```

10.7.9 Administration de CVS

Ajoutez l'alias « mx » à un projet CVS (serveur local) de la manière suivante :

```
$ export CVSR00T=/srv/cvs/projet
$ cvs co CVSR00T/modules
$ cd CVSR00T
$ echo "mx -a module1" >>modules
$ cvs ci -m "mx est maintenant un alias de module1"
$ cvs release -d .
```

Vous pouvez maintenant vérifier « module1 » (alias : « mx ») depuis le CVS vers le répertoire « new » de la manière suivante :

```
$ cvs co -d nouveau mx
$ cd nouveau
```

Note

Afin de pouvoir effectuer la procédure précédente, vous devrez avoir les permissions d'accès aux fichiers appropriées.

10.7.10 Bit d'exécution pour la vérification (« checkout ») du CVS

Lorsque vous vérifiez des fichiers du CVS, leur bit d'autorisation en exécution est conservé.

Si vous rencontrez des problèmes de droits d'exécution avec un fichier récupéré, par exemple « nom_fichier », modifiez-en les permissions dans le répertoire correspondant du dépôt CVS par ce qui suit afin de corriger ce problème :

```
# chmod ugo-x nom_de_fichier
```

10.8 Subversion

Subversion est un système de contrôle de version de **génération récente** qui remplace l'ancien CVS. Il en possède la plupart des fonctionnalités à l'exception des étiquettes et des branches.

Vous devrez installer les paquets `subversion`, `libapache2-svn` et `subversion-tools` pour mettre en place un serveur Subversion.

10.8.1 Configuration du dépôt Subversion

Actuellement, le paquet `subversion` ne configure pas le dépôt, on doit donc le faire manuellement. Un emplacement possible pour le dépôt se trouve en « /srv/svn/projet ».

Créez un répertoire de la manière suivante :

```
# mkdir -p /srv/svn/projet
```

Créez la base de données du dépôt comme suit :

```
# svnadmin create /srv/svn/projet
```

10.8.2 Accès à Subversion par l'intermédiaire d'un serveur Apache2

Si vous n'avez accès au dépôt de Subversion que par un serveur Apache2, il vous suffira de ne rendre le dépôt accessible en écriture que par le serveur WWW de la manière suivante :

```
# chown -R www-data:www-data /srv/svn/projet
```

Ajouter (ou décommenter) ce qui suit dans « /etc/apache2/mods-available/dav_svn.conf » afin de permettre l'accès au dépôt avec une authentification de l'utilisateur.

```
<Location /projet>
  DAV svn
  SVNPath /srv/svn/projet
  AuthType Basic
  AuthName "Dépôt subversion"
  AuthUserFile /etc/subversion/passwd
<LimitExcept GET PROPFIND OPTIONS REPORT>
  Require valid-user
</LimitExcept>
</Location>
```

Créer un fichier d'authentification des utilisateurs avec la commande suivante :

```
# htpasswd2 -c /etc/subversion/passwd nom_utilisateur
```

Redémarrez Apache2.

Votre nouveau dépôt Subversion est accessible à l'URL « <http://localhost/projet> » et « <http://example.com/projet> » depuis svn(1) (en supposant que l'URL du serveur web est « <http://example.com/> »).

10.8.3 Accès local à Subversion par un groupe

Ce qui suit configure le dépôt Subversion pour un accès local par un groupe, `projet`, par exemple :

```
# chmod 2775 /srv/svn/projet
# chown -R root:src /srv/svn/projet
# chmod -R ug+rwX /srv/svn/projet
```

Votre nouveau dépôt Subversion est accessible aux utilisateurs locaux appartenant au groupe `projet` à l'URL « `file:///localhost/srv/svn/projet` » ou « `file:///srv/svn/projet` » depuis svn(1). Vous devrez lancer des commandes comme `svn`, `svnserve`, `svnlook` et `svnadmin` avec « `umask 002` » afin de permettre l'accès au groupe.

10.8.4 Accès distant à Subversion avec SSH

Un dépôt Subversion accessible à un groupe se trouve pour SSH à l'URL « `example.com:/srv/svn/projet` », vous pouvez y accéder avec svn(1) à l'URL « `svn+ssh://example.com:/srv/svn/projet` ».

10.8.5 Structure de répertoires de Subversion

Afin de compenser le manque de branches et d'étiquettes, de nombreux projets utilisent une arborescence de répertoires semblable à la suivante pour Subversion :

```
----- module1
|   |-- branches
|   |-- tags
|   |   |-- version-1.0
|   |   '-- version-2.0
|   |
|   |
```

```
|   '-- trunk
|       |-- fichier1
|       |-- fichier2
|       '-- fichier3
|
|-- module2
```

ASTUCE

Vous devez utiliser la commande « `svn copy ...` » pour marquer les branches et les étiquettes. Cela permet de s'assurer que Subversion enregistre correctement l'historique de modification des fichiers et économise l'espace disque.

10.8.6 Importer une nouvelle source vers Subversion

Créez un nouvel emplacement d'arborescence locale des sources dans « `~/chemin/vers/module1` » par la commande suivante :

```
$ mkdir -p ~/chemin/vers/module1; cd ~/chemin/vers/module1
```

Mettez des fichiers dans la nouvelles arborescence locale de sources se trouvant en « `~/chemin/vers/module1` ».

Importez-le dans Subversion avec les paramètres suivants :

- Module name : « `module1` »
- URL du site Subversion : « `file:///srv/svn/projet` »
- Répertoire de Subversion : « `module1/trunk` »
- Étiquette de Subversion : « `module1/tags/Version-initiale` »

```
$ cd ~/chemin/vers/module1
$ svn import file:///srv/svn/projet/module1/trunk -m "Lancement module1"
$ svn cp file:///srv/svn/projet/module1/trunk file:///srv/svn/projet/module1/tags/Version- ↵
  initiale
```

ou encore, comme suit :

```
$ svn import ~/chemin/vers/module1 file:///srv/svn/projet/module1/trunk -m "Lancement ↵
  module1"
$ svn cp file:///srv/svn/projet/module1/trunk file:///srv/svn/projet/module1/tags/Version- ↵
  initiale
```

ASTUCE

Vous pouvez remplacer des formats d'URL comme « `file:///...` » par d'autres formats d'URL comme « `http://...` » et « `svn+ssh://...` ».

10.8.7 Flux de travail avec Subversion

Voici un exemple typique de flux de travail utilisant Subversion avec son client natif.

ASTUCE

Les commandes du client proposées par le paquet `git-svn` peuvent proposer un mode d'utilisation de remplacement pour Subversion en utilisant la commande `git`. Consultez Section [10.6.4](#).

Vérifiez de la manière suivante tous les modules disponibles depuis le projet pointé par l'URL « `file:///srv/svn/projet` » :

```
$ svn list file:///srv/svn/projet
module1
module2
...
```

Récupérez « `module1/trunk` » dans un répertoire « `module1` » comme suit :

```
$ cd ~/chemin/vers
$ svn co file:///srv/svn/projet/module1/trunk module1
$ cd module1
```

Effectuez les modifications de contenu au besoin.

Vérifiez les modifications par un équivalent de « `diff -u [repository] [local]` » de la manière suivante :

```
$ svn diff
```

Vous-vous apercevez que vous avez cassé sévèrement un fichier appelé « `fichier_à_restaurer` » mais que les autres fichiers sont corrects.

Écrasez le fichier « `fichier_a_annuler` » avec une nouvelle copie propre depuis Subversion comme suit :

```
$ svn revert fichier_a_annuler
```

Enregistrez l'arbre local des sources mis à jour vers Subversion comme suit :

```
$ svn ci -m "Décrire les modifications"
```

Créez et ajoutez le fichier « `fichier_a_ajouter` » dans Subversion comme suit :

```
$ vi fichier_a_ajouter
$ svn add fichier_a_ajouter
$ svn ci -m "fichier_a_ajouter ajouté"
```

Fusionnez la dernière version depuis Subversion comme suit :

```
$ svn up
```

Recherchez les lignes commençant par « `C filename` » qui indiquent des modifications en conflit.

Recherchez le code non modifié, par exemple « `nom_fichier.r6` », « `nom_fichier.r9` » et « `nom_fichier.mien` ».

Recherchez « `<<<<<<<` » et « `>>>>>>>` » dans les fichiers qui indiquent des conflits de modifications.

Au besoin, éditez les fichiers pour résoudre les conflits.

Ajoutez l'étiquette de version (« `release tag` ») « `Version-1` » comme suit :

```
$ svn ci -m "dernier « commit » pour la Release-1"
$ svn cp file:///srv/svn/projet/module1/trunk file:///srv/svn/projet/module1/tags/Release-1
```

Poursuivez l'édition.

Supprimez l'étiquette de version « `Version-1` » comme suit :

```
$ svn rm file:///srv/svn/projet/module1/tags/Release-1
```

Récupérez les modifications depuis Subversion comme suit :

```
$ svn ci -m "vraiment le dernier commit pour Release-1"
```

Ajoutez de nouveau la balise « `Version-1` » depuis la version mise à jour de HEAD de trunk de Subversion comme suit :

```
$ svn cp file:///srv/svn/projet/module1/trunk file:///srv/svn/projet/module1/tags/Version-1
```

Créez une branche dont le chemin est « `module1/branches/Release-initial-bugfixes` » depuis la version initiale pointée par le chemin « `module1/tags/Release-initial` » et récupérez-le vers le répertoire « `~/chemin/vers/ancien` » comme suit :

```
$ svn cp file:///srv/svn/projet/module1/tags/Version-initiale file:///srv/svn/projet/ ↵  
  module1/branches/Version-initiale-corrections-de-bogues  
$ cd ~/chemin/vers  
$ svn co file:///srv/svn/projet/module1/branches/Version-initiale-corrections-de-bogues ↵  
  ancien  
$ cd ancien
```

ASTUCE

Utilisez « `module1/trunk@{2005-12-20}` » (format de date [ISO 8601](#)) plutôt que « `module1/tags/Release-initial` » pour indiquer une date particulière comme point de branche.

Travaillez sur cet arbre de sources local pointant vers la branche « `Version-initiale-corrections-de-bogues` » qui est basée sur la version initiale.

Travaillez vous-même sur cette branche... jusqu'à ce que quelqu'un d'autre rejoigne cette branche « `Version-initiale-corrections-de-bogues` ».

Effectuez la synchronisation avec les fichiers de cette branche modifiés par d'autres comme suit :

```
$ svn up
```

Au besoin, éditez les fichiers pour résoudre les conflits.

Récupérez les modifications depuis Subversion comme suit :

```
$ svn ci -m "récupérés dans cette branche"
```

Mettez à jour l'arborescence local avec HEAD de trunk de la manière suivante :

```
$ svn switch file:///srv/svn/projet/module1/trunk
```

Mettez à jour l'arborescence locale (contenu = HEAD de trunk) en fusionnant depuis la branche « `Version-initiale-corrections-de-bogues` » par ce qui suit :

```
$ svn merge file:///srv/svn/projet/module1/branches/Version-initiale-corrections-de-bogues
```

Corrigez les conflits avec l'éditeur.

Récupérez les modifications depuis Subversion comme suit :

```
$ svn ci -m "Version-initiale-corrections-de-bogues fusionnée"
```

Réalisez une archive de la manière suivante :

```
$ cd ..  
$ mv ancien ancien-module1-corrections_bogues  
$ tar -cvzf ancien-module1-corrections_bogues.tar.gz ancien-module1-corrections_bogues  
$ rm -rf ancien-module1-corrections_bogues
```

ASTUCE

Vous pouvez remplacer des formats d'URL comme « `file:///...` » par d'autres formats d'URL comme « `http://...` » et « `svn+ssh://...` ».

ASTUCE

Vous pouvez ne vérifier (« checkout ») qu'un sous-répertoire de « `module1` » en indiquant son nom comme, par exemple, « `svn co file:///srv/svn/projet/module1/trunk/sous-répertoire module1/sous-répertoire` », etc.

option	signification
<code>--dry-run</code>	simulation, pas d'effet
<code>-v</code>	affichage détaillé des messages d'activité de Subversion

Table 10.15 – Options importantes des commandes de Subversion (à utiliser comme premier(s) paramètre(s) de `svn(1)`)

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:933, I:999	10679	jeu de caractères	convertisseur de codage de texte entre différents paramètres linguistiques à l'aide d'iconv(1) (fondamental)
recode	V:5, I:35	608	charset+eol	convertisseur de codage de texte entre différents paramètres linguistiques (flexible avec plus d'alias et de fonctionnalités)
konwert	V:2, I:58	123	jeu de caractères	convertisseur de codage de texte entre différents paramètres linguistiques (le luxe)
nkf	V:1, I:11	346	jeu de caractères	jeux de caractères pour le japonais
tcs	V:0, I:0	479	jeu de caractères	jeu de caractères du traducteur
unaccent	V:0, I:0	36	jeu de caractères	remplacer les lettres accentuées par leur équivalent accentué
tofromdos	V:2, I:35	50	eol (fin de ligne)	convertisseur de format de texte entre DOS et UNIX : fromdos(1) et todos(1)
macutils	V:0, I:1	298	eol (fin de ligne)	convertisseur de format de texte entre Macintosh et UNIX :frommac(1) et tomac(1)

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 codage1 -t codage2 entrée.txt >sortie.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

Certains codages ne sont pris en compte que pour la conversion de données et ne sont pas utilisés en tant que valeur de paramètre linguistique (Section [8.3.1](#)).

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 "Trouvé non-UTF-8"
```

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 8.3.6](#)) 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	<code>^J</code>	10	0A
MSDOS et Windows	CR-LF	<code>^M^J</code>	13 10	0D 0A
Apple Macintosh	CR	<code>^M</code>	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.site-distant.com/help-info.html >fichier-texte
```

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:212, I:813	2294	html → texte	convertisseur HTML vers texte avec la commande « <code>w3m -dump</code> »
html2text	V:21, I:74	258	html → texte	convertisseur avancé HTML vers texte (ISO 8859-1)
lynx	V:32, I:104	1901	html → texte	convertisseur HTML vers texte avec la commande « <code>lynx -dump</code> »
elinks	V:14, I:32	1587	html → texte	convertisseur HTML vers texte avec la commande « <code>elinks -dump</code> »
links	V:17, I:46	2149	html → texte	convertisseur HTML vers texte avec la commande « <code>links -dump</code> »
links2	V:2, I:17	5392	html → texte	convertisseur HTML vers texte avec la commande « <code>links2 -dump</code> »
antiword	V:6, I:14	623	MSWord → texte	convertir des fichiers MSWord en texte brut ou en ps
catdoc	V:27, I:45	676	MSWord → texte	convertir les fichiers MSWord en texte brut ou en TeX
pstotext	V:4, I:7	126	ps/pdf → texte	extraire du texte depuis des fichiers PostScript et PDF
unhtml	V:0, I:0	42	html → texte	supprimer les balises d'un fichier HTML
odt2txt	V:3, I:6	53	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:20, I:433	27580	mise en évidence	MACRO Vim pour convertir du code source en HTML avec « :source \$VIMRUNTIME/syntax/html.vim »
cxref	V:0, I:0	1174	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:1, I:7	2008	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:16	997	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:2	104	texte → couleur	coloriseur générique pour n'importe quoi (Python)
txt2html	V:0, I:4	296	texte → html	convertisseur de texte vers HTML (Perl)
markdown	V:0, I:7	56	texte → html	formateur de documents textes Markdown vers (X)HTML (Perl)
asciidoc	V:0, I:14	63	texte → tout format	formateur de documents textes AsciiDoc vers XML/HTML (Python)
pandoc	V:4, I:26	69422	texte → tout format	Marqueur de conversion global (Haskell)
python-docutils	V:41, I:569	1725	texte → tout format	formateur de documents ReStructured Text vers XML (Python)
txt2tags	V:0, I:1	951	texte → tout format	conversion de documents de texte vers HTML, SGML, LaTeX, page de manuel, MoinMoin, Magic Point et PageMaker (Python)
udo	V:0, I:0	569	texte → tout format	document universel - utilitaire de traitement de texte (langage C)
stx2any	V:0, I:0	264	texte → tout format	convertisseur de documents depuis un texte brut structuré vers d'autres formats (m4)
rest2web	V:0, I:0	526	texte → html	convertisseur de documents depuis ReStructured Text vers html (Python)
aft	V:0, I:0	235	texte → tout format	système de préparation de document de « forme libre » (Perl)
yodl	V:0, I:0	547	texte → tout format	langage de pré-document et outils pour le manipuler
sdf	V:0, I:0	1445	texte → tout format	analyseur simple de documents (Perl)
sisu	V:0, I:0	5338	texte → tout format	infrastructure de structuration, de publication et de recherche de documents (Ruby)

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">contenu</nom>
```

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
<code>&quot;</code> ;	" : double apostrophe
<code>&apos;</code> ;	' : apostrophe
<code>&lt;</code> ;	< : inférieur à
<code>&gt;</code> ;	> : supérieur à
<code>&amp;</code> ;	& : esperluette

Table 11.7 – Liste des entités XML prédéfinies



Attention

« < » ou « & » ne peuvent pas être utilisés dans des attributs ni des éléments.

Note

Lorsqu'on utilise des entités définies par l'utilisateur de style SGML, par exemple « &some-tag: », la première définition prend le pas sur les suivantes. La définition de l'entité est exprimée par « <!ENTITY une-balise "valeur entité"> ».

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.

paquet	popcon	taille	mot clé	description
docbook-xml	I:530	2131	xml	Définition de type de document XML (DTD) pour DocBook
xsltproc	V:15, I:120	149	xslt	processeur en ligne de commandes XSLT (XML → XML, HTML, texte brut, etc.)
docbook-xsl	V:15, I:231	14998	xml/xslt	feuilles de style XSL pour le traitement de DocBook XML vers divers formats de sortie avec XSLT
xmlto	V:3, I:37	126	xml/xslt	convertisseur XML-vers-tout avec XSLT
dbtoepub	V:0, I:1	71	xml/xslt	Convertisseur Docbook XML vers .epub
dblatex	V:6, I:25	4639	xml/xslt	convertir les fichiers Docbook en documents DVI, PostScript, PDF avec XSLT
fop	V:3, I:53	64	xml/xsl-fo	convertir les fichiers Docbook XML en PDF

Table 11.8 – Liste d'outils XML

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 \(DSSSL\)](#).

ASTUCE

`ye1p` 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 :

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.

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.

paquet	popcon	taille	mot clé	description
openjade	V:3, I:36	988	dsssl	processeur DSSSL à la norme ISO/IEC 10179:1996 DSSSL (le plus récent)
openjade1.3	V:0, I:0	NOT_FOUND	dsssl	processeur DSSSL à la norme ISO/IEC 10179:1996 DSSSL (séries 1.3.x)
jade	V:0, I:11	NOT_FOUND	dsssl	processeur DSSSL original de James Clark (séries 1.2.x)
docbook-dsssl	V:2, I:39	2604	xml/dsssl	feuilles de style DSSSL pour le traitement des DocBook XML vers divers formats de sortie avec DSSSL
docbook-utils	V:1, I:27	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 <code>docbook2*</code>
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

paquet	popcon	taille	mot clé	description
wv	V:6, I:9	717	MSWord → n'importe quoi	convertisseur de document depuis Microsoft Word vers HTML, LaTeX, etc.
texi2html	V:0, I:10	1832	texi → html	convertisseur de Texinfo vers HTML
man2html	V:0, I:3	141	manpage → html	convertisseur de page de manuel en HTML (prise en charge de CGI)
tex4ht	V:1, I:22	36	tex ↔ html	convertisseur entre (La)TeX et HTML
unrtf	V:2, I:4	145	rtf → html	convertisseur de documents de RTF vers HTML, etc
info2www	V:3, I:4	156	info → html	convertisseur depuis GNU info vers HTML (prise en charge de CGI)
ooo2dbk	V:0, I:1	217	sxw → xml	convertisseur de documents OpenOffice.org SXW en DocBook XML
wp2x	V:0, I:0	215	WordPerfect → any	fichiers WordPerfect 5.0 et 5.1 vers TeX, LaTeX, troff, GML et HTML
doclifter	V:0, I:0	457	troff → xml	convertisseur de troff vers DocBook XML

Table 11.10 – Liste d'outils d'extraction de données XML

paquet	popcon	taille	mot clé	description
libxml2-utils	V:24, I:318	177	xml ↔ html ↔ xhtml	outil XML en ligne de commandes xmllint(1) (vérification de la syntaxe, reformatage, décomposition...)
tidy	V:2, I:17	83	xml ↔ html ↔ xhtml	vérificateur de syntaxe et reformateur HTML

Table 11.11 – Liste d'outils d'impression élégante du XML

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:6, I:69	68	(La)TeX	système TeX pour la composition, l'aperçu et l'impression
groff	V:7, I:151	9362	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 méta paquet [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 une_page_de_manuel | lpr
```

```
$ man -Tps une_page_de_manuel | mpage -2 | lpr
```

Le deuxième exemple imprime deux pages par feuille.

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:1, I:18	179	SGML → <code>manpage</code>	macros de conversion de DocBook SGML vers roff <code>man</code>
help2man	V:0, I:9	454	text → <code>manpage</code>	générateur automatique d'une pages de manuel depuis <code>--help</code>
info2man	V:0, I:0	134	info → <code>manpage</code>	convertisseur depuis GNU info vers POD ou page de manuel
txt2man	V:0, I:1	65	text → <code>manpage</code>	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 »).

La licence de la dernière version amont de Ghostscript par Artifex a été modifiée d'AFPL vers GPL et fusionnée avec les dernières modifications de version d'ESP telles que celles relatives à CUPS version 8.60 sous forme d'une version unifiée.

paquet	popcon	taille	description
ghostscript	V:123, I:686	225	L'interpréteur GPL Ghostscript PostScript/PDF
ghostscript-x	V:25, I:76	211	interpréteur GPL Ghostscript PostScript/PDF - prise en charge de l'affichage X
libpoppler64	V:45, I:109	3214	bibliothèque de rendu PDF dérivée du visualisateur PDF xpdf
libpoppler-glib8	V:233, I:525	435	bibliothèque de rendu PDF (bibliothèque partagée basée sur GLib)
poppler-data	V:96, I:669	12123	CMaps pour PDF la bibliothèque de rendu (pour la prise en charge de CJK : Adobe-*)

Table 11.14 – Liste des interpréteurs Ghostscript PostScript

ASTUCE

« `gs -h` » permet d'afficher la configuration de Ghostscript.

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 -sFichierSortie=blabla.ps -f toto1.ps toto2.ps
$ gs -q -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -sFichierSortie=blabla.pdf -f toto1.pdf toto2. ↵
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 :

paquet	popcon	taille	mot clé	description
poppler-utils	V:53, I:488	526	pdf → ps,text, ...	utilitaires PDF :: pdftops, pdfinfo, pdfimages, pdftotext, pdffonts
psutils	V:11, I:199	219	ps → ps	outils de conversion de document PostScript
poster	V:0, I:7	49	ps → ps	créer des posters de grande dimension à partir de pages PostScript
enscript	V:2, I:27	2111	text → ps, html, rtf	convertir du text ASCII en PostScript, HTML, RTF ou Pretty-Print
a2ps	V:3, I:28	3624	text → ps	Convertisseur de « N'importe quoi vers PostScript » et imprimeur élégant
pdftk	V:8, I:57	3873	pdf → pdf	outil de conversion de documents PDF : pdftk
mpage	V:0, I:5	141	text,ps → ps	imprimer plusieurs pages par feuille
html2ps	V:0, I:5	320	html → ps	convertisseur de HTML vers PostScript
gnuhtml2latex	V:0, I:1	53	html → latex	convertisseur de html vers latex
latex2rtf	V:0, I:7	475	latex → rtf	convertir des documents de LaTeX en RTF qui peuvent être lus par MS Word
ps2eps	V:7, I:113	94	ps → eps	convertisseur de PostScript vers EPS (PostScript encapsulé)
e2ps	V:0, I:0	112	text → ps	convertisseur de texte vers PostScript avec la prise en charge du codage japonais
impose+	V:0, I:1	180	ps → ps	Utilitaires PostScript
trueprint	V:0, I:0	138	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:5	27	ps → svg	convertisseur de PDF vers le format Scalable vector graphics (« SVG »)
pdftoipe	V:0, I:0	63	ps → ipe	convertisseur de PDF vers le format IPE d'XML

Table 11.15 – Liste des utilitaires pour les données imprimables

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

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

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
sharutils	V:8, I:117	1385	mail	shar(1), unshar(1), uuencode(1), uudecode(1)
mpack	V:2, I:25	91	MIME	encoder et décoder des messages MIME : mpack(1) et munpack(1)
tnef	V:6, I:11	98	ms-tnef	dépaqueter des attachements MIME de type « application/ms-tnef » qui est un format propre à Microsoft
uudeview	V:0, I:5	97	mail	encodeur et décodeur pour les formats suivants : uuencode , xxencode , BASE64 , quoted printable et BinHex
readpst	I:1	21	PST	convertir les fichiers Outlook PST files de Microsoft vers le format mbox

Table 11.16 – Liste de paquets facilitant la conversion de données de courrier électronique

ASTUCE

Le serveur de [protocole Internet d'accès au messages](#) (« d'Internet Message Access Protocol ») version 4 (IMAP4) (consultez Section 6.7) peut être utilisé pour déplacer des courriels depuis des systèmes de courriels propriétaires si le logiciel de courriel client peut aussi être configuré pour utiliser le serveur IMAP4.

11.5.1 Bases concernant les données de courrier électronique

Les données ([SMTP](#)) de courrier électronique doivent être limitées à des séries de données de 7 bits. Les données binaires et les données textuelles sur 8 bits sont codées dans un format sur 7 bits avec [Multipurpose Internet Mail Extensions \(MIME\)](#) et la sélection du jeu de caractères (consultez Section 8.3.1).

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

Pour les données de courriel non-UNIX accessibles par un logiciel client ne venant pas de Debian, il est possible de dialoguer avec le serveur IMAP4, il vous faudra peut-être les déplacer en lançant votre propre serveur IMAP4 (consultez Section 6.7).

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.

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).

Vous pouvez aussi extraire des données depuis le format RPM avec ce qui suit :

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

paquet	popcon	taille	mot clé	description
gimp	V:91, I:507	16278	image (bitmap)	Programme de manipulation d'images GNU (GNU Image Manipulation Program »)
imagemagick	V:142, I:538	192	image (bitmap)	programmes de manipulation d'images
graphicsmagick	V:7, I:15	4821	image (bitmap)	programmes de manipulation d'images (dérivés d' imagemagick)
xsane	V:22, I:191	909	image (bitmap)	Interface graphique X11 de SANE basée sur GTK+ (Scanner Access Now Easy)
netpbm	V:29, I:541	4302	image (bitmap)	outils de conversion graphique
icoutils	V:8, I:80	191	png ↔ ico (bitmap)	convertit les icônes et curseurs de MS Windows de et vers des formats PNG (favicon.ico)
scribus	V:13, I:29	19136	ps/pdf/SVG/...	éditeur DTP Scribus
libreoffice-draw	V:320, I:478	8970	image (vectorielle)	suite de bureautique LibreOffice - dessin
inkscape	V:141, I:354	129840	image (vectorielle)	éditeur SVG (Scalable Vector Graphics)
dia-gnome	V:5, I:10	20	image (vectorielle)	éditeur de diagrammes (GNOME)
dia	V:25, I:41	3881	image (vectorielle)	éditeur de diagrammes (Gtk)
xfig	V:12, I:19	1795	image (vectorielle)	Outil de génération interactive de figures sous X11
pstoedit	V:13, I:339	667	ps/pdf → image (vectorielle)	convertisseur de fichiers PostScript et PDF en graphiques vectoriels éditables (SVG)
libwmf-bin	V:13, I:359	104	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:16	453	image → image	outil de post-traitement pour des pages numérisées pour OCR
tesseract-ocr	V:4, I:28	558	image → texte	logiciel OCR libre basé sur le moteur commercial OCR de HP
tesseract-ocr-eng	I:29	37486	image → texte	moteur de données OCR : fichier de langue de tesseract-ocr pour le texte en anglais
gocr	V:1, I:23	477	image → texte	logiciel OCR libre
ocrad	V:1, I:7	299	image → texte	logiciel OCR libre
eog	V:95, I:329	10582	image (Exif)	Programme de visualisation d'images « Eye of Gnome » (l'œil de Gnome)
gthumb	V:15, I:26	3269	image (Exif)	visionneuse et gestionnaire de photos (GNOME)
geeqie	V:16, I:24	1588	image (Exif)	Visualiseur d'images utilisant GTK+
shotwell	V:15, I:126	5762	image (Exif)	gestionnaire de photos numériques (GNOME)
gtkam	V:0, I:7	1079	image (Exif)	Application destinée à rapatrier des données numériques depuis les appareils photo numériques (GTK+)
gphoto2	V:1, I:14	969	image (Exif)	Le client en ligne de commande pour appareil photo numérique gphoto2
gwenview	V:29, I:104	4509	image (Exif)	visionneuse d'images (KDE)
kamera	V:3, I:104	230	image (Exif)	gestion des appareils photo numériques dans les applications KDE
digikam	V:4, I:17	1760	image (Exif)	application de gestion photos numériques (KDE)
exiv2	V:5, I:71	247	image (Exif)	Outil pour manipuler les méta-données EXIF/IPTC
exiftran	V:2, I:25	71	image (Exif)	transformer les images JPEG des appareils photo numériques
jhead	V:1, I:13	105	image (Exif)	manipuler la partie non graphique des fichiers JPEG compatibles avec Exif (appareils photo numériques)
exif	V:1, I:10	236	image (Exif)	utilitaire en ligne de commandes pour afficher les informations EXIF contenues dans les fichiers JPEG
exiftags	V:0, I:4	288	image (Exif)	utilitaire pour lire les balises Exif depuis un fichier JPEG d'appareil photo numérique
exifprobe	V:0, I:3	482	image (Exif)	lire les métadonnées des images numériques
			image	décoder les images brutes (« raw ») des appareils

paquet	popcon	taille	mot clé	description
alien	V:4, I:64	166	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:36	49341	tout → EPUB	Convertisseur de livre numérique et gestion de bibliothèque.

Table 11.18 – Liste d’outils divers de conversion de données

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 programmations :

paquet	popcon	taille	documentation
autoconf	V:29, I:226	1868	« info autoconf » fourni par autoconf-doc
automake	V:27, I:220	1707	« info automake » fourni par automake1.10-doc
bash	V:853, I:999	5799	« info bash » fourni par bash-doc
bison	V:10, I:113	2061	« info bison » fourni par bison-doc
cpp	V:394, I:806	41	« info cpp » fourni par cpp-doc
ddd	V:1, I:13	3965	« info ddd » fourni par ddd-doc
exuberant-ctags	V:7, I:38	333	exuberant-ctags(1)
flex	V:10, I:101	1174	« info flex » fourni par flex-doc
gawk	V:355, I:478	2199	« info gawk » fourni par gawk-doc
gcc	V:148, I:606	43	« info gcc » fourni par gcc-doc
gdb	V:21, I:140	7983	« info gdb » fourni par gdb-doc
gettext	V:53, I:367	7076	« info gettext » fourni par gettext-doc
gfortran	V:20, I:63	16	« info gfortran » fourni par gfortran-doc (Fortran 95)
fpc	I:4	113	fpc(1) et les pages html fournies par fp-doc (Pascal)
glade	V:1, I:12	2209	l'aide fournie par le menu (UI Builder)
libc6	V:933, I:999	10679	« info libc » fourni par glibc-doc et glibc-doc-reference
make	V:154, I:622	1211	« info make » fourni par make-doc
xutils-dev	V:2, I:18	1466	imake(1) , xmkmf(1) , etc.
mawk	V:371, I:997	183	mawk(1)
perl	V:610, I:996	651	perl(1) et les pages html fournies par perl-doc et perl-doc-html
python	V:683, I:988	648	python(1) et les pages html fournies par python-doc
tcl8.4	V:3, I:50	NOT_FOUND	tcl(1) et les pages de manuel détaillées fournies par tcl8.4-doc
tk8.4	V:1, I:31	NOT_FOUND	tk(1) et les pages de manuel détaillées fournies par tk8.4-doc
ruby	V:103, I:321	38	ruby(1) et la référence interactive fournir par ri
vim	V:118, I:393	2374	menu d'aide (F1) fourni par vim-doc
susv2	I:0	15	aller chercher « The Single UNIX Specifications v2 »
susv3	I:0	15	aller chercher « The Single UNIX Specifications v3 »

Table 12.1 – Liste de paquets pour aider à 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.

**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
... lignes de commandes
```

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

De nombreux scripts systèmes peuvent être interprétés par n'importe lequel des interpréteurs de commande [POSIX](#) (consultez Tableau 1.13). L'interpréteur de commandes par défaut pour le système est « `/bin/sh` » qui est un lien symbolique pointant vers le programme réel.

- `bash(1)` pour `lenny` ou plus ancien
- `dash(1)` pour `squeeze` ou plus récent

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)`.

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.

Bon : POSIX	À éviter : bashisme
if ["\$toto" = "\$titi"] ; then ...	if ["\$toto" == "\$titi"] ; then ...
diff -u fichier.c.orig fichier.c	diff -u fichier.c{.orig,}
mkdir /tototiti /tototutu	mkdir /toto{titi,tutu}
funcname() { ... }	function funcname() { ... }
format octal : « \377 »	format hexadécimal : « \xff »

Table 12.2 – Liste de bashismes typiques

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
\$0	nom de l'interpréteur ou du script de l'interpréteur
\$1	premier (1er) paramètre de l'interpréteur
\$9	neuvième (9ème) paramètre de l'interpréteur
\$#	nombre de paramètres positionnels
"\$*"	"\$1 \$2 \$3 \$4 ..."
"\$@"	"\$1" "\$2" "\$3" "\$4" ...
\$?	valeur de retour de la commande la plus récente
\$\$	PID de ce script de l'interpréteur de commandes
#!	PID de la tâche de fond la plus récemment lancée

Table 12.3 – Liste des paramètres de l'interpréteur de commandes

Les **expansions de paramètre** les plus courantes à retenir sont mentionnées ci-dessous :

forme de l'expression du paramètre	valeur si <code>var</code> est positionnée	valeur si <code>var</code> n'est pas positionnée
<code>\${var:-chaîne}</code>	« \$var »	« chaîne »
<code>\${var:+chaîne}</code>	« chaîne »	« null »
<code>\${var:=chaîne}</code>	« \$var »	« chaîne » (et lancer « <code>var=chaîne</code> »)
<code>\${var:?chaîne}</code>	« \$var »	echo « chaîne » vers stderr (et quitter avec une erreur)

Table 12.4 – Liste des expansions de paramètre de l'interpréteur

Ici, les deux points « : » dans tous ces opérateurs sont en fait optionnels.

- **avec** « : » = opérateur de test pour **existe** et **différent de null**
- **sans** « : » = opérateur de test pour **existe** uniquement

12.1.3 Opérateurs conditionnels de l'interpréteur

Chaque commande retourne un **état de sortie** qui peut être utilisé pour des expressions conditionnelles.

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 12.5 – Liste des substitutions-clés de paramètres de l’interpréteur

- 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 [ <expression_conditionnelle> ]; then
  <si_succès_lancer_cette_commande>
else
  <si_pas_de_succes_lancer_cette_commande>
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` ».

équation	condition pour retourner une valeur logique vraie
<code>-e <fichier></code>	<code><fichier></code> existe
<code>-d <fichier></code>	<code><fichier></code> existe et est un répertoire
<code>-f <fichier></code>	<code><fichier></code> existe et est un fichier normal (« régulier »)
<code>-w <fichier></code>	<code><fichier></code> existe et peut être écrit
<code>-x <fichier></code>	<code><fichier></code> existe et est exécutable
<code><fichier1> -nt <fichier2></code>	<code><fichier1></code> est plus récent que <code><fichier2></code> (modification)
<code><fichier1> -ot <fichier2></code>	<code><fichier1></code> est plus ancien que <code><fichier2></code> (modification)
<code><fichier1> -ef <fichier2></code>	<code><fichier1></code> et <code><fichier2></code> sont sur le même périphérique et le même numéro d’inœud

Table 12.6 – Liste des opérateurs de comparaison dans les expressions conditionnelles

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
-z <str>	la longueur de <str> est nulle
-n <str>	la longueur de <str> est non nulle
<str1> = <str2>	<str1> et <str2> sont égales
<str1> != <str2>	<str1> et <str2> ne sont pas égales
<str1> < <str2>	<str1> est trié avant <str2> (dépendant des paramètres linguistiques)
<str1> > <str2>	<str1> est trié après <str2> (dépendant des paramètres linguistiques)

Table 12.7 – 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 9.3.9.

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>
 - Métacaractères : < > | ;& ()
 - 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
coreutils	V:881, I:999	15103	utilitaires du cœur de GNU
debianutils	V:934, I:999	213	divers utilitaires spécifiques à Debian
bsdmainutils	V:866, I:998	566	collection d'autres utilitaires provenant de FreeBSD
bsdutils	V:851, I:999	238	utilitaires de base provenant de 4.4BSD-Lite
moreutils	V:4, I:19	201	utilitaires supplémentaires d'UNIX

Table 12.8 – Lites des paquets comportant des petits programmes utilitaires pour les scripts de l'interpréteur de commandes

ASTUCE

Bien que `moreutils` puisse ne pas exister en dehors de Debian, il propose d'intéressants petits programmes. Le plus remarquable est `sponge(8)` qui est bien pratique pour écrire directement sur le fichier d'origine.

12.1.7 Dialogue de l'interpréteur de commandes

L'interface utilisateur d'un simple programme d'interpréteur de commandes peut être améliorée au-delà de l'interaction bête des commandes `echo` et `read` afin de devenir plus interactive en utilisant un des programmes appelés « dialogues », etc.

paquet	popcon	taille	description
x11-utils	V:305, I:635	637	xmessage(1) : afficher un message ou une question dans une fenêtre (X)
whiptail	V:415, I:995	68	afficher des boîtes de dialogues conviviales depuis des scripts de l'interpréteur de commandes (newt)
dialog	V:29, I:151	1111	afficher des boîtes de dialogues conviviales depuis des scripts de l'interpréteur de commandes (ncurses)
zenity	V:112, I:425	362	afficher de boîtes de dialogue graphiques depuis des scripts de l'interpréteur de commandes (gtk2.0)
ssft	V:0, I:0	129	Outil frontal de scripts de l'interpréteur de commandes (enrobeur pour zenity, kdialog et dialog avec gettext)
gettext	V:53, I:367	7076	« /usr/bin/gettext.sh »: traduire des messages

Table 12.9 – Liste d'interfaces utilisateur

12.1.8 Exemple de script avec zenity

Voici un exemple simple qui crée une image ISO avec des données RS02 fournies par dvdaster(1) :

```
#!/bin/sh -e
# gmkrso2 : Copyright (C) 2007 Osamu Aoki <osamu@debian.org>, Public Domain
#set -x
error_exit()
{
    echo "$1" >&2
    exit 1
}
# Initialiser les variables
DATA_ISO="$HOME/Desktop/iso-$$img"
LABEL=$(date +%Y%m%d-%H%M%S-%Z)
if [ $# != 0 ] && [ -d "$1" ]; then
    DATA_SRC="$1"
else
    # Sélectionner le répertoire pour la création de l'image ISO à partir d'un dossier du bureau
    DATA_SRC=$(zenity --file-selection --directory \
        --title="Sélectionner la racine de l'arborescence de répertoires pour créer l'image ISO" \
        || error_exit "Quitter lors de la sélection du répertoire"
fi
# Vérifier la taille de l'archive
xterm -T "Check size $DATA_SRC" -e du -s $DATA_SRC/*
SIZE=$((du -s $DATA_SRC | awk '{print $1}')/1024)
if [ $SIZE -le 520 ] ; then
    zenity --info --title="Dvdisaster RS02" --width 640 --height 400 \
        --text="La taille des données convient à pour une sauvegarde sur CD :\n $SIZE Mo"
elif [ $SIZE -le 3500 ] ; then
    zenity --info --title="Dvdisaster RS02" --width 640 --height 400 \
        --text="La taille des données convient à pour une sauvegarde sur DVD :\n $SIZE Mo"
else
    zenity --info --title="Dvdisaster RS02" --width 640 --height 400 \
        --text="La taille des données est trop grande pour une sauvegarde : $SIZE Mo"
    error_exit "Taille de données trop importante pour la sauvegarde :\n $SIZE Mo"
fi
# on n'est certain d'avoir une option -e fonctionnelle uniquement avec xterm
# Créer une image ISO brute
rm -f "$DATA_ISO" || true
xterm -T "genisoimage $DATA_ISO" \
    -e genisoimage -r -J -V "$LABEL" -o "$DATA_ISO" "$DATA_SRC"
# Créer la redondance RS02 supplémentaire
```



```
xterm -T "dvdisaster $DATA_ISO" -e dvdisaster -i "$DATA_ISO" -mRS02 -c
zenity --info --title="Dvdisaster RS02" --width 640 --height 400 \
--text="Données ISO/RS02 ($SIZE Mo) \n créées sur : $DATA_ISO"
# EOF
```

Vous désirez peut-être créer un lanceur sur le bureau avec une commande définie comme « `/usr/local/bin/gmkrs02 %d` ».

12.2 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 :

```
cible: [ prérequis... ]
[ TAB ] commande1
[ TAB ] -commande2 # ignorer les erreurs
[ TAB ] @commande3 # supprimer l'écho
```

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
<code>\$@</code>	cible
<code>\$<</code>	première exigence
<code>\$?</code>	toutes les exigences plus récentes
<code>\$^</code>	toutes les exigences
<code>\$*</code>	« <code>%</code> » correspond au radical dans le motif cible

Table 12.10 – 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 12.11 – Liste de l'expansion des variables de **make**

Exécutez « `make -p -f/dev/null` » afin de voir les règles automatiques internes.

12.3 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)

12.3.1 Programme simple en C (gcc)

Un exemple simple « `example.c` » peut être compilé avec la bibliothèque « `libm` » 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); /* évite les débordements de tampon */
    y[10] = '\0'; /* remplir afin d'être certain que la chaîne se termine par '\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..

12.4 Déboguer

Le débogage est une partie de l'activité de programmations. Savoir comment déboguer des programmes fera de vous un bon utilisateur de Debian qui pourra produire des rapports de bogues documentés.

12.4.1 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
```

Un bon didacticiel de `gdb` est proposé par « [info gdb](#) » ou peut être trouvé [ailleurs sur le web](#). Voici un exemple simple d'utilisation de `gdb(1)` sur un « `program` » compilé avec l'option « `-g` » qui produit les informations de débogage.

```
$ gdb program
(gdb) b 1                # définit un point d'arrêt à la ligne 1
(gdb) run args           # lancer les programmes avec des paramètres
(gdb) next               # ligne suivante
...
(gdb) step               # avancer d'un pas
...
(gdb) p parm             # afficher parm
...
(gdb) p parm=12          # définir sa valeur à 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.

12.4.2 Déboguer un paquet Debian

Comme, par défaut, sous Debian, tous les paquets binaires sont « strippés » (dépouillés de leurs symboles de débogage), la plupart des symboles de débogage sont supprimés des paquets normaux. Pour pouvoir déboguer des paquets Debian à l'aide de `gdb(1)`, les paquets `*-dbg` correspondants doivent être installés (par exemple `libc6-dbg` dans le cas de `libc6`).

Si un paquet à déboguer ne possède pas de paquet `*-dbg` correspondant, vous devrez l'installer après l'avoir reconstruit comme suit :

```
$ mkdir /chemin/nouveau ; cd /chemin/nouveau
$ sudo apt-get update
$ sudo apt-get dist-upgrade
$ sudo apt-get install fakeroot devscripts build-essential
$ sudo apt-get build-dep nom_paquet_source
$ apt-get source nom_paquet
$ cd nom_paquet*
```

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 nom_paquet*.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.

12.4.3 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.

La trace peut être obtenue de la manière suivante :

- Lancer le programme sous `gdb(1)` ;
- Reproduire le plantage ;
 - Vous allez être ramené à l'invite de `gdb` ;
- Entrez « `bt` » à l'invite de `gdb` ;

Si le programme se fige, vous pouvez le planter en pressant `Ctrl-C` dans le terminal où tourne `gdb` pour récupérer l'invite de `gdb` ;

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 « `$MALLOC_CHECK_` » à la valeur 2 (`malloc(3)`). Vous pouvez le faire en lançant `gdb` de la manière suivante :

```
$ MALLOC_CHECK_=2 gdb hello
```

12.4.4 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 12.12 – Liste des commandes avancées de `gdb`

12.4.5 Déboguer les erreurs de X

Si un programme `aperçu1` de GNOME a reçu une erreur X, vous devriez obtenir un message comme suit :

```
Le programme 'aperçu1' a reçu une erreur du système X Window.
```

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.

12.4.6 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 [9.3.6](#).

12.4.7 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:269, I:615	15628	mtrace(1) : fonctionnalité de débogage de malloc dans glibc
valgrind	V:8, I:56	70463	débogueur mémoire et optimiseur
kmtrace	V:1, I:18	351	traceur de fuites de mémoire de KDE utilisant mtrace(1) de glibc
allegoop	V:0, I:1	655	frontal pour GNOME du vérificateur mémoire Valgrind
electric-fence	V:0, I:6	70	débogueur malloc(3)
leaktracer	V:0, I:2	56	traceur de fuites de mémoire pour les programmes C++
libdmalloc5	V:0, I:3	360	bibliothèque de débogage de l'allocation mémoire

Table 12.13 – Liste des outils de détection des fuites de mémoire

12.4.8 Outils d'analyse du code statique

Il y a des outils tels que [lint](#) pour l'analyse du code statique.

paquet	popcon	taille	description
splint	V:0, I:5	1889	outil pour vérifier de manière statique les bogues d'un programme en C
flawfinder	V:0, I:1	175	outil pour examiner le code source en C/C++ et rechercher des faiblesses du point de vue de la sécurité
perl	V:610, I:996	651	interpréteur ayant un vérificateur de code statique interne : B::Lint(3perl)
pylint	V:6, I:15	945	vérificateur de code statique Python
weblint-perl	V:0, I:2	34	vérificateur de syntaxe et de style minimal pour HTML
linklint	V:0, I:1	343	vérificateur rapide de liens et outils de maintenance de sites web
libxml2-utils	V:24, I:318	177	utilitaires avec xmllint(1) pour valider les fichiers XML

Table 12.14 – Liste des outils d'analyse du code statique :

12.4.9 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

12.5 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.

12.6 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:10, I:113	2061	générateur d'analyseur GNU LALR
byacc	V:0, I:7	160	générateur d'analyseur Berkeley LALR
btyacc	V:0, I:0	207	générateur d'analyseur avec retour arrière basé sur <code>byacc</code>

Table 12.15 – 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 « `yyle x()` », habituellement créé avec Flex.

```
%%
%%
```

12.7 Autoconf

Autoconf est un outil destiné à produire des scripts en shell qui configurent automatiquement un code source de logiciel pour l'adapter à de nombreux types de systèmes « UNIX-like » en utilisant l'ensemble du système de construction GNU.

`autoconf(1)` produit le script de configuration « `configure` ». « `configure` » crée automatiquement un « `Makefile` » en utilisant le patron « `Makefile.in` ».

12.7.1 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/loca
$ make
$ make install # cela met les fichiers dans le système
```

12.7.2 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 « toutes-les-options-que-vous-lui-avez-passé »
# 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
```

Si vous n'êtes pas certain de l'emplacement où les fichiers ont été installés, vous devriez envisager d'utiliser `checkinstall`(8) du paquet `checkinstall` qui fournit une voie propre pour la désinstallation. Il prend maintenant en charge la création d'un paquet Debian à l'aide de l'option « `-D` ».

12.8 Folie de courts scripts en Perl

Bien que tous les scripts en [AWK](#) puissent être réécrits automatiquement en [Perl](#) en utilisant `a2p`(1), il est plus facile de convertir manuellement les scripts AWK constitués d'une seule ligne en scripts Perl d'une seule ligne.

Regardons le bout de script AWK suivant :

```
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.

Consultez `perlrun`(1) pour les options de la ligne de commandes. Pour des scripts en Perl plus fous, il peut être intéressant de consulter [Perl Golf](#).

12.9 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.

12.10 La conversion du code source

Il existe des programmes pour convertir les codes sources.

paquet	popcon	taille	mot clé	description
perl	V:610, I:996	651	AWK → PERL	convertir le code source de AWK vers PERL : a2p(1)
f2c	V:0, I:7	433	FORTRAN → C	convertir le code source de FORTRAN 77 vers C/C++ : f2c(1)
intel2gas	V:0, I:0	174	intel → gas	convertisseur depuis NASM (format Intel) vers l'Assembleur GNU (GAS)

Table 12.16 – Liste des outils de conversion de code source

12.11 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 [9.10.4](#) pour comprendre les techniques de base d'un environnement isolé (« chroot »)
- [debuid\(1\)](#), [pbuilder\(1\)](#) et [pdebuild\(1\)](#)
- Section [12.4.2](#) pour recompiler avec les informations de débogage
- [Guide du nouveau responsable Debian](#) comme tutoriel (paquet `maint-guide`)
- [Référence du développeur Debian](#) (paquet `developers-reference`)
- [Charte Debian](#) (paquet `debian-policy`)
- [Guide for Debian Maintainers](#) (the `debmake-doc` package)

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.

J'espère que cette « Référence Debian (version 2) » offrira un bon cap à suivre pour les personnes se trouvant dans le labyrinthe de Debian.

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.

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)

— rendue obsolète par cette nouvelle « [Référence Debian](#) (version 2) »

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.

De nombreuses pages de manuel et de pages info du système Debian ont été utilisées comme référence primaire pour écrire ce document. Dans la mesure où Osamu Aoki considérait que c’était compris dans une utilisation équitable (« [fair use](#) »), de nombreuses parties d’entre-elles, particulièrement les définitions des commandes, ont été utilisées comme morceaux de phrases après un effort éditorial soigneux afin de les insérer dans le style et avec les objectifs de ce 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) » 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

La source du document original en anglais est actuellement écrite sous forme de fichiers texte [AsciiDoc](#). [AsciiDoc](#) n’est utilisé que par facilité car il demande moins de frappe que l’XML natif et prend en charge les tableaux dans un format très intuitif. Vous devez avoir à l’esprit que les fichiers XML et PO sont les vrais fichiers sources. Ils sont convertis dans le format XML DocBook par le script de construction et les données créées automatiquement sont insérées afin de former un source final XML Docbook. Ce source final XML Docbook peut être converti en HTML, epub, en text brut, en PostScript et en PDF. (Certains formats peuvent être omis pour la distribution.)