

Support de formation

Administration Système Linux1

Préparation à l'examen LPI 101

version 0.2 – française v 0.1

mis à disposition sous licence GFDL par LinuxIT



Table des matières

INTRODUCTION.....	6
À propos.....	6
Financement.....	6
Marques déposées.....	6
Historique.....	7
Dramatis Personae.....	7
Objectifs.....	7
Plan de formation.....	8
Pré-requis et profil des élèves.....	8
Le programme de certification LPI.....	8
À l'attention du formateur.....	8
Pas de garantie.....	8
Sources d'information.....	9
Conventions de notation.....	9
LES OBJECTIFS DÉTAILLÉS POUR L'EXAMEN DE CERTIFICATION LPI 101.....	10
Sujet 101 : Architecture système.....	10
Sujet 102 : Installation de Linux et gestion de paquetages.....	12
Sujet 103 : Commandes GNU et Unix.....	14
Sujet 104 : Disques, Système de fichiers Linux , Arborescence de fichiers standard.....	19
INSTALLATION.....	24
Le CD d'installation.....	26
Installations locales.....	27
Installation réseau.....	27
Disque de récupération.....	27
Plan de partitionnement.....	28
Plusieurs systèmes d'exploitation sur une machine.....	30
Exercices et résumé.....	30
CONFIGURATION MATÉRIELLE.....	33
Allocation de ressources.....	35
Cartes d'extension.....	36
USB.....	36
SCSI.....	37
Cartes réseau.....	39
Configuration des modems.....	40
Configuration des imprimantes.....	44
Cartes son.....	45
EXERCICES ET RÉSUMÉ.....	47
GESTION DES DISQUES.....	50
Les disques.....	50
Partitions.....	50
Les programmes de partitionnement.....	51
Chargeurs de démarrage ("Master Boot Record").....	52
Gestion des partitions.....	54
Gestion des quotas.....	55

EXERCICES ET RÉSUMÉ.....	56
Questions de révision.....	56
Glossaire.....	56
Fichiers.....	57
Commandes.....	57
Travaux pratiques.....	58
Réponses aux questions.....	59
LE SYSTÈME DE FICHIERS LINUX.....	60
La structure du système de fichiers.....	60
Formatage et cohérence des systèmes de fichier.....	61
Contrôle de l'utilisation des disques.....	63
Droits d'accès et attributs des fichiers.....	64
Résumé et exercices.....	68

.....Copyright (c) 2005 LinuxIT.

Vous avez la permission de copier, distribuer et / ou modifier ce document suivant les termes de la Licence GNU Free Documentation Licence, version 1.2 ou une version plus récente publiée par la Free Software Foundation ; les parties qui ne peuvent pas être modifiées sont : Historique et À propos. Enfin, la première page doit contenir le texte : "mis à disposition sous licence GFDL par LinuxIT."

Retrouvez la licence complète page.Error: Reference source not found

Introduction

À propos

Ce document a été mis à disposition par le centre de formation LinuxIT : www.linuxit.com.

Nous remercions particulièrement Andrew Meredith qui est à l'origine de ce document. Nous remercions également nos élèves qui nous ont aidé à faire passer les aspects les plus techniques de Linux par leurs nombreuses questions. Cela nous a amené à ajouter des illustrations pour présenter des concepts de façon plus pédagogique. Enfin, un grand merci à Paul McEnery pour ses conseils techniques et pour avoir commencé certains des chapîtres les plus difficiles comme ceux qui traitent du serveur X (101), des modems (102) et du noyau Linux (102).

Ce guide est disponible en ligne sur <http://savannah.nongnu.org/projects/lpi-manuals/>. Nous remercions les volontaires de Savannah d'avoir accepté le projet et de nous fournir l'espace Web.

Financement

La réalisation de ces documents a été financé en partie par UNDP-APDIP International Open Source Network and International Development Research Centre Canada.

Marques déposées

Linux est une marque déposée par Linus Torvalds aux États Unis et dans les autres pays.

Red Hat Linux et Red Hat Entreprise Linux sont des marques déposées par RedHat Inc.

Mandriva® Linux® est une marque déposée par Mandriva Inc.

SUSETM (SUSE une marque de SUSE LINUX Products GmbH, une filiale de Novell)

UNIX® est une marque déposée de the Open Group.

Introduction

Historique

Première version (v 0.0) : octobre 2003, révision par Adrian Thommasset.

Mis à jour en janvier 2004 après révision par Andrew Meredith.

Novembre 2004 : ajout de la partie sur les cartes d'extension dans la Configuration du matériel par Adrian Thommasset.

Décembre 2004 : ajout de la table des matières et des objectifs visés par Adrian Thommasset.

Janvier 2005 : ajout du glossaire et d'un résumé sur les commandes et fichiers à voir à la fin des chapitres par Adrian Thommasset.

Juin 2005 : Adaptation du document à partir des recommandations de Sernet en vue d'un agrément par LPI, par Andrew Meredith. Ajout d'informations par Andrew D Marshall, mise à jour par Adrian Thomasset. Ajout de la partie sur les outils Debian par Duncan Thomson.

Août 2005 : ce support pédagogique dans la version anglaise 0.2 a reçu l'agrément LATM (LPI Approved Training Materials – support pédagogique agréé LPI) par Sernet.

Mai 2008 : début de la traduction par Éric Deschamps

Notes de version (version française) :

- 30/06/08 v 0.02 : mise en forme, suppression des tableaux pour les commandes, utilisation du style terminal
-

Dramatis Personae

Adrian Thommasset	< adriant<CHEZ>linuxit<POINT>com >	http://www.linuxit.com/
Andrew Meredith	< andrew<CHEZ>anvil<POINT>org >	http://www.anvil.org/
Andrew D Marshall	< admarshall<CHEZ>gmail<POINT>com >	http://h0lug.sourceforge.net/
Duncan Thomson	< thom-ci0<CHEZ>paisley<POINT>ac<POINT>uk >	http://www.paisley.ac.uk/
Eric Deschamps	< erdesc<CHEZ>free<POINT>fr >	http://traduc.org

Objectifs

Le but principal de ce guide est d'apporter des explications, exemples et exercices aux personnes préparant l'examen LPI 101 pour la certification LPI 1 (LPIC-1).

Trois sources sont utilisées pour la réalisation de ce guide :

- les objectifs de l'examen LPI 101 ;
- les critères pour les supports de formations agréés par LPI ;
- Le guide de l'auteur du projet de documentation Linux (LDP ou TLDP).

Les objectifs pour l'examen LPI 101 et les critères d'agrément LATM pour les documents de formation sont résumés ci-dessous. Vous pouvez également les trouver en ligne à l'adresse :

http://www.lpi.org/en/obj_101.html

Le Guide de l'auteur LDP [<http://www.tldp.org/LDP/LDP-Author-Guide/>] donne un ensemble de directives à ceux qui souhaitent publier des guides pratiques (HOWTO) ou des manuels via le plus grand système de documentation de GNU/Linux, le Linux Documentation Project.

Introduction

Le deuxième objectif de ce guide, à égalité avec le premier est tiré du guide de l'auteur du LDP [LDP-AG, 4.1. Writing the Text] : “de créer un ensemble lisible, divertissant, et compréhensible à partir d'informations brutes”.

Plan de formation

Ce cours est prévu pour accompagner des cours pratiques de préparation de l'examen LPI 101. Généralement, la formation prend entre 24 et 32 heures, coupées en sessions consécutives de 8 h mais le cours est adaptable et vous pouvez l'utiliser dans d'autres situations.

Pré-requis et profil des élèves

Nous partons du principe que les élèves ont :

- une expérience de quelques années d'utilisation d'ordinateurs basés sur une architecture Intel x86, y compris une bonne connaissance des composants matériels et de leur interaction avec le système d'exploitation ;
- une culture générale de l'informatique et des réseaux, comme les systèmes binaires et hexadécimal, les unités de mesure les plus courantes (octets, Ko vs Kbit, Mhz, etc.) les structures de système de fichiers, des bases sur les opérations réseau Ethernet et Internet, le matériel, etc.
- une expérience pratique de plus de 3 mois cumulés d'utilisation de GNU/Linux, BSD ou un autre système Unix, en travaillant en ligne de commande (sur un terminal texte ou dans une console), localement ou à distance.

Les personnes disposant de moins d'expérience mais motivées et prêtes à prendre plus de temps pour rattraper leurs lacunes peuvent également suivre ce manuel. C'est une tâche ardue mais pas impossible.

Les commandes et leur utilisation sont présentées avec leur référence et des exemples d'utilisation ainsi que des exercices et leurs réponses vous mettant dans les conditions de l'examen. Tous les exercices sont facultatifs, les plus recommandés étant présentés ou référencés dans le cours.

Le programme de certification LPI

Il y a actuellement (NdT : au moment de l'écriture du document, pas de sa traduction) deux niveaux de certification LPI. Le premier niveau LPIC-1 s'obtient en passant les examens LPI 101 et 102. De la même façon, vous obtiendrez le niveau 2 (LPIC-2) en passant les examens LPI 201 et 202.

Il n'y a aucun pré-requis pour passer les examens LPI 101 et 102. Vous devrez cependant les avoir obtenus pour passer les examens du LPIC-2.

À l'attention du formateur

Ce manuel ne contient pas de notes pour le formateur. Vous noterez les points suivants :

- les exercices des parties **Gestion du matériel** et **Le système de fichiers Linux** partent sur le principe qu'il est possible de créer une nouvelle partition sur les postes. Prévoyez à l'installation une partition étendue assez grande avec au minimum 100 Mo d'espace libre une fois toutes les autres partitions créées.
- Les paquets suivants sont nécessaires pour les exercices :
 - rpm-build
 - shareutils

Pas de garantie

Ce manuel vous est délivré sans aucune garantie.

Introduction

Sources d'information

www.lpi.org

www.linux-praxis.de

www.lpiforums.com

www.tldp.org

www.fsf.org

www.linuxit.com

Conventions de notation

Les commandes et noms de fichiers apparaissent dans le texte en **gras**.

Les symboles <> indiquent un argument obligatoire.

Les symboles [] indiquent un argument optionnel.

Les commandes pouvant être tapées directement sont mises en valeur comme suit :

commande

Les objectifs détaillés pour l'examen de certification LPI 101

INFORMATION IMPORTANTE : Ceci est la liste des 101 objectifs actuels, qui sont en vigueur **depuis** le 1er avril 2009.

Un poids qui varie de 1 à 10 est affecté à chaque objectif, il représente son importance par rapport aux autres. Les objectifs de poids le plus important seront sujets à davantage de questions durant l'examen.

Sujet 101 : Architecture système

101.1 Déterminer et configurer les paramètres du matériel

Poids	2
Description	Les candidats doivent être capables de déterminer et de configurer le matériel et les périphériques sous GNU/Linux.

Domaines de connaissance les plus importants :

- Activer et désactiver les périphériques intégrés.
- Configurer les systèmes avec ou sans périphérique externe tels que les claviers.
- Savoir différencier les types de périphériques de stockage de masse.
- Paramétrer correctement les identifiants matériels pour les différents périphériques, en particulier le périphérique de démarrage.
- Connaître les différences entre les périphériques qui peuvent être connectés à froid ou à chaud.
- Déterminer les ressources matérielles des périphériques.
- Outils et commandes permettant d'obtenir des informations sur les périphériques (par exemple : `lsusb`, `lspci`, etc.)
- Outils permettant de manipuler les périphériques USB.
- Compréhension des concepts `sysfs`, `udev`, `hald` et `dbus`.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- `/sys`
- `/proc`
- `/dev`
- `modprobe`
- `lsmod`
- `lspci`
- `lsusb`

101.2 Démarrage du système

Poids	3
--------------	---

Les objectifs détaillés pour l'examen de certification LPI 101

Description

Les candidats doivent pouvoir se repérer dans les phases de démarrage d'un système Linux.

Domaines de connaissance les plus importants

- Fournir des commandes au chargeur de démarrage et passer des paramètres d'amorçage au noyau.
- Démontrer sa connaissance des séquences d'amorçage depuis le lancement du BIOS jusqu'à l'achèvement des séquences de démarrage.
- Consulter les événements de la phase de démarrage dans les journaux (logs).

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- /var/log/messages
- dmesg
- BIOS
- bootloader
- kernel
- init

101.3 Changer de niveau d'exécution et arrêter ou redémarrer le système

Poids

3

Description

Les candidats doivent être capables de gérer le niveau d'exécution du système. Cet objectif inclut le passage en mode mono-utilisateur (single user), l'arrêt et le redémarrage du système. Les candidats doivent être capables de prévenir les utilisateurs avant de changer de niveau d'exécution et de terminer les processus correctement. Cet objectif comprend également le paramétrage du niveau d'exécution par défaut.

Domaines de connaissance les plus importants

- Paramétrer le niveau d'exécution par défaut.
- Passer d'un niveau d'exécution à un autre, y compris en mode mono-utilisateur.
- Arrêter et redémarrer le système en ligne de commande.
- Prévenir les utilisateurs avant un changement de niveau d'exécution ou pour d'autres événements systèmes importants.
- Terminer les processus correctement.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- /etc/inittab
 - shutdown
 - init
-

Les objectifs détaillés pour l'examen de certification LPI 101

- /etc/init.d
- telinit

Sujet 102 : Installation de Linux et gestion de paquetages

102.1 Concevoir un schéma de partitionnement du disque dur

Poids	2
Description	Les candidats doivent être capables de concevoir un schéma de partitionnement du disque dur pour un système Linux.

Domaines de connaissance les plus importants

- Allouer les systèmes de fichiers et l'espace d'échange (swap) sur des partitions ou des disques séparés.
- Ajuster le schéma de partitionnement à l'usage prévu du système.
- S'assurer du bon emplacement de la partition /boot qui contient le chargeur de démarrage.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- / : le système de fichiers racine (root)
- /var
- /home
- espace d'échange (swap)
- points de montage
- partitions

102.2 Installation d'un gestionnaire de démarrage

Poids	2
Description	Les candidats doivent être capables de sélectionner, d'installer et de configurer un gestionnaire de démarrage.

Domaines de connaissance les plus importants

- Fournir des lieux d'amorçage alternatifs et sauvegarder les options de démarrage.
- Installer et configurer un chargeur de démarrage tel que GRUB.
- Interagir avec le chargeur de démarrage en ligne de commande.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- /boot/grub/menu.lst
 - grub-install
 - MBR
-

Les objectifs détaillés pour l'examen de certification LPI 101

- superbloc
- /etc/lilo.conf
- lilo

102.3 Gestion des bibliothèques partagées

Poids	1
Description	Les candidats doivent être capables de déterminer les bibliothèques dont dépendent les programmes et les installer en cas de besoin.

Domaines de connaissance les plus importants

- Identifier les bibliothèques partagées.
- Identifier l'emplacement typique des bibliothèques systèmes.
- Charger des bibliothèques partagées.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- ldd
- ldconfig
- /etc/ld.so.conf
- LD_LIBRARY_PATH

102.4 Utilisation du gestionnaire de paquetage Debian

Poids	3
Description	Les candidats doivent être capables de gérer les paquetages en utilisant les outils de gestion de paquetage Debian.

Domaines de connaissance les plus importants

- Installer, mettre à jour et désinstaller des paquetages binaires Debian.
- Rechercher des paquetages contenant des fichiers ou des bibliothèques spécifiques installés ou non.
- Obtenir des informations sur un paquetage Debian comme la version, le contenu, les dépendances, l'intégrité du paquetage, et l'état d'installation (que le paquetage soit installé ou non).

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- /etc/apt/sources.list
 - dpkg
 - dpkg-reconfigure
 - apt-get
 - apt-cache
-

Les objectifs détaillés pour l'examen de certification LPI 101

- aptitude

102.5 Utilisation de gestionnaire de paquetage RPM et YUM

Poids

3

Description

Les candidats doivent être capables de gérer les paquetages avec les outils RPM et YUM.

Domaines de connaissance les plus importants

- Installer, ré-installer, mettre à jour et supprimer les paquetages avec RPM et YUM.
- Obtenir des informations sur un paquetage RPM comme la version, le contenu, les dépendances, l'intégrité du paquetage, la signature et l'état d'installation.
- Déterminer les fichiers relatifs à un paquetage donné, et rechercher à quel paquetage appartient un fichier donné.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- rpm
- rpm2cpio
- /etc/yum.conf
- /etc/yum.repos.d/
- yum
- yumdownloader

Sujet 103 : Commandes GNU et Unix

103.1 Travail en ligne de commande

Poids

4

Description

Les candidats doivent être capables de travailler en ligne de commande. C'est l'utilisation du shell bash qui sera traitée dans cet objectif.

Domaines de connaissance les plus importants

- Utiliser des commandes ou des séquences de commandes pour réaliser des tâches simples en ligne de commande.
- Utiliser et modifier l'environnement du shell, en particulier savoir définir, exporter et référencer des variables d'environnement.
- Utiliser et éditer l'historique des commandes.
- Lancer des commandes comprises ou non dans le chemin par défaut.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- .
-

Les objectifs détaillés pour l'examen de certification LPI 101

- bash
- echo
- env
- exec
- export
- pwd
- set
- unset
- man
- uname
- history

103.2 Traitement de flux de type texte par des filtres

Poids 3

Description Les candidats doivent être capables d'appliquer des filtres à un flux de type texte.

Domaines de connaissance les plus importants

- Envoyer des fichiers textes ou des sorties de commandes à des filtres textuels pour les modifier en utilisant des commandes UNIX appartenant au paquetage GNU "textutils".

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- cat
 - cut
 - expand
 - fmt
 - head
 - od
 - join
 - nl
 - paste
 - pr
 - sed
 - sort
 - split
 - tail
 - tr
-

Les objectifs détaillés pour l'examen de certification LPI 101

- unexpand
- uniq
- wc

103.3 Effectuer une gestion de base sur les fichiers

Poids	4
Description	Les candidats doivent être capables d'utiliser les commandes Linux de base pour gérer les fichiers et les répertoires.

Domaines de connaissance les plus importants

- Copier, déplacer et détruire des fichiers ou des répertoires.
- Copier plusieurs fichiers et répertoires récursivement.
- Supprimer des fichiers et répertoires récursivement.
- Utiliser de manière simple et avancée les caractères de joker (*,?,[.]).
- Utiliser la commande pour rechercher des fichiers sur la base de leurs types , de leurs tailles ou de leurs dates (de création, de modification et d'accès).
- Utiliser les commandes tar, cpio et dd.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- cp
 - find
 - mkdir
 - mv
 - ls
 - rm
 - rmdir
 - touch
 - tar
 - cpio
 - dd
 - file
 - gzip
 - gunzip
 - bzip2
 - file globing (transformation d'un motif générique en une liste de noms de fichiers correspondants)
-

Les objectifs détaillés pour l'examen de certification LPI 101

103.4 Utilisation des flux, des tubes (pipes) et des redirections

Poids	4
Description	Les candidats doivent être capables de rediriger des flux et de les connecter dans le but de traiter efficacement ces données textuelles. Les tâches à effectuer comprennent les redirections de l'entrée standard, de la sortie standard et de la sortie standard des erreurs, connecter la sortie d'une commande à l'entrée d'une autre, utiliser la sortie d'une commande comme paramètre pour une autre commande et envoyer le résultat en même temps sur la sortie standard et dans un fichier.

Domaines de connaissance les plus importants

- Redirection de l'entrée standard, de la sortie standard et de la sortie standard des erreurs.
- Connecter la sortie d'une commande à l'entrée d'une autre commande.
- Utiliser la sortie d'une commande comme paramètre pour une autre commande.
- Envoyer simultanément le résultat d'une commande vers la sortie standard et vers un fichier.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- tee
- xargs

103.5 Création, surveillance et destruction de processus

Poids	4
Description	Les candidats doivent être capables d'effectuer une gestion de base des processus.

Domaines de connaissance les plus importants

- Exécuter un processus en avant-plan et en arrière plan.
- Indiquer qu'un programme doit continuer son exécution après la déconnexion.
- Contrôler les processus actifs.
- Sélectionner et trier les processus à afficher.
- Envoyer des signaux aux processus.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- &
 - bg
 - fg
 - jobs
-

Les objectifs détaillés pour l'examen de certification LPI 101

- kill
- nohup
- ps
- top
- free
- uptime
- killall

103.6 Modification des priorités des processus

Poids 2

Description Les candidats doivent être capables de gérer les priorités des processus.

Domaines de connaissance les plus importants

- Savoir la priorité par défaut affectée à un processus créé.
- Exécuter un programme avec une priorité plus haute ou plus basse que celle par défaut.
- Changer la priorité d'un processus en cours d'exécution.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- nice
- ps
- renice
- top

103.7 Recherche dans des fichiers texte avec les expressions rationnelles

Poids 2

Description Les candidats doivent être capables de manipuler des fichiers et des données de type texte en utilisant des expressions rationnelles. Cet objectif comprend la création d'expressions rationnelles contenant plusieurs caractères spéciaux. Il comprend également l'utilisation d'outils à base d'expressions rationnelles pour effectuer des recherches dans un système de fichiers ou dans le contenu d'un fichier.

Domaines de connaissance les plus importants

- Créer des expressions rationnelles simples contenant plusieurs éléments de notation.
 - Utiliser des outils à base d'expressions rationnelles pour effectuer des recherches dans un système de fichiers ou dans le contenu d'un fichier.
-

Les objectifs détaillés pour l'examen de certification LPI 101

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- grep
- egrep
- fgrep
- sed
- regex

103.8 Édition de fichiers texte avec "vi"

Poids	3
Description	Les candidats doivent être capables d'éditer le contenu de fichiers texte en utilisant "vi". Cet objectif comprend le déplacement dans "vi", les modes de "vi", l'insertion, la modification, la destruction, la copie et la recherche de texte.

Domaines de connaissance les plus importants

- Se déplacer dans un document édité avec "vi".
- Utiliser les modes "vi" de base à savoir le mode commande, le mode insertion et le mode remplacement.
- Insérer, modifier, détruire, copier et rechercher du texte.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- vi
- /, ?
- h,j,k,l
- i, o, a
- c, d, p, y, dd, yy
- ZZ, :w!, :q!, :e!

Sujet 104 : Disques, Système de fichiers Linux , Arborescence de fichiers standard.

104.1 Création de partitions et systèmes de fichiers.

Poids	2
Description	Les candidats doivent être capables de créer des partitions et des systèmes de fichiers. Ceci inclut la prise en charge des partitions d'échange (swap).

Domaines de connaissance les plus importants

- Utiliser la commande "mkfs" pour configurer une partition et créer différents types de système de fichiers comme :
-

Les objectifs détaillés pour l'examen de certification LPI 101

- ext2
- ext3
- xfs
- reiserfs v3
- vfat

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- fdisk
- mkfs
- mkswap

104.2 Maintenir l'intégrité des systèmes de fichiers

Poids	2
Description	Les candidats doivent être capables de maintenir l'intégrité d'un système de fichiers, ainsi que les données supplémentaires associées à la journalisation.

Domaines de connaissance les plus importants

- Vérifier l'intégrité du système de fichiers.
- Contrôler l'espace libre et les inodes.
- Réparer les problèmes simples du système de fichiers.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- du
- df
- fsck
- e2fsck
- mke2fs
- debugfs
- dumpe2fs
- tune2fs
- xfs tools (tels que xfs_metadump et xfs_info)

104.3 : Contrôle du montage et du démontage des systèmes de fichiers

Poids	3
Description	Les candidats doivent être capables de configurer le montage, le démontage et les options de montage des systèmes de fichiers.

Les objectifs détaillés pour l'examen de certification LPI 101

Domaines de connaissance les plus importants

- Monter et démonter manuellement les systèmes de fichiers.
- Configurer le montage des systèmes de fichiers au démarrage du système.
- Configurer les options de montage des systèmes de fichiers. Par exemple, permettre aux utilisateurs de monter et démonter un système de fichiers amovible.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- /etc/fstab
- /media
- mount
- umount

104.4 Gestion des quotas de disque

Poids	1
Description	Les candidats doivent être capables de gérer les quotas disque des utilisateurs.

Domaines de connaissance les plus importants

- Configurer un quota de disque pour un système de fichiers.
- Éditer, vérifier et générer des rapports d'utilisation de quotas des utilisateurs.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- quota
- edquota
- repquota
- quotaon

104.5 Gérer les permissions et les propriétaires des fichiers

Poids	3
Description	Les candidats doivent être capables de gérer les droits d'accès et les propriétaires des fichiers.

Domaines de connaissance les plus importants

- Gérer les permissions d'accès sur des fichiers classiques, des fichiers spéciaux comme sur des répertoires.
 - Utiliser des modes comme suid, sgid et sticky bit pour garantir la sécurité.
 - Savoir changer le mode de création de fichiers par défaut.
 - Utiliser le groupe propriétaire pour donner des permissions aux membres d'un groupe.
-

Les objectifs détaillés pour l'examen de certification LPI 101

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- chmod
- umask
- chown
- chgrp

104.6 Créer et changer les liens symboliques et physiques sur les fichiers

Poids 2

Description Les candidats doivent être capables de gérer des liens symboliques et physiques sur un fichier.

Domaines de connaissance les plus importants

- Créer les liens.
- Identifier les liens symboliques des liens physiques.
- Copier versus lier les fichiers
- Utiliser des liens pour les tâches d'administration système.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- ln

104.7 Recherche de fichiers et placement des fichiers aux endroits adéquats

Poids 2

Description Les candidats doivent être familiarisés avec l'arborescence standard de fichiers FHS (Filesystem Hierarchy Standard), y compris la place adéquate d'un fichier et les classifications des répertoires.

Domaines de connaissance les plus importants

- Comprendre l'emplacement correct d'un fichier dans le FHS.
- Rechercher les fichiers et les commandes dans l'arborescence Linux
- Connaître l'emplacement des fichiers et des répertoires importants dans l'arborescence Linux.

Liste non-exhaustive de termes, fichiers et utilitaires utilisés pour cet objectif

- find
 - locate
 - updatedb
 - whereis
 - which
 - type
-

Les objectifs détaillés pour l'examen de certification LPI 101

- /etc/updatedb.conf

Page consultée fois

Installation

Installation

Pré-requis

Aucun

Objectifs

- Comprendre l'organisation d'un CD d'installation Linux type
 - Effectuer différents types d'installation
 - Organiser un partitionnement simple (voir également p .Error: Reference source not found)
-

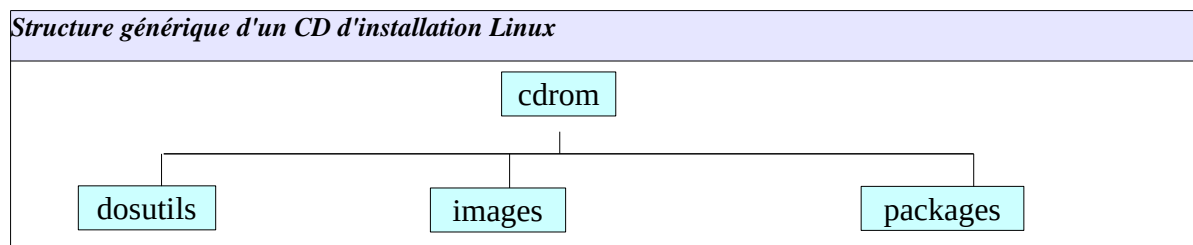
Contenu

INSTALLATION.....	1
1. Le CD d'installation.....	2
2. Installations locales.....	2
3. Installation réseau.....	3
4. Disque de récupération.....	3
5. Plan de partitionnement.....	4
6. Plusieurs systèmes d'exploitation sur une machine.....	5
7. Exercices et résumé.....	6

Installation

Le CD d'installation

Les différentes distributions Linux utilisent des noms de répertoires différents sur les CDs d'installation. La structure générique est la suivante :



packages: ce répertoire contient les paquets au format binaire. Voici les noms correspondants pour les distributions les plus courantes :

debian: **dist**
 mandrake: **Mandrake**
 redhat: **RedHat**
 suse: **suse**

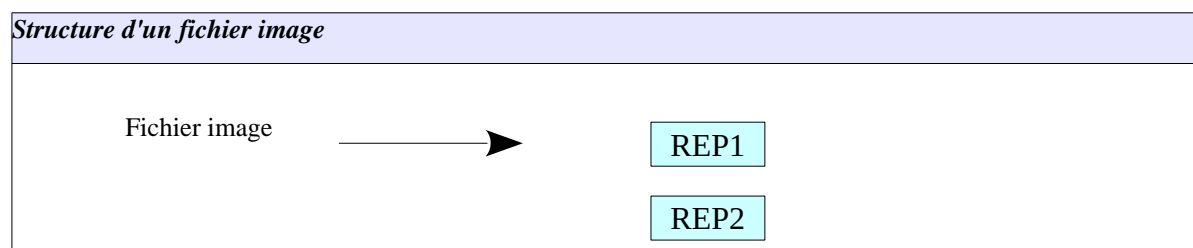
Tous les logiciels installés sur le système à l'installation viennent de ces paquets logiciels. Consultez la partie traitant des gestionnaires de paquets page [Error: Reference source not found](#) pour plus de détails.

images: ce répertoire contient plusieurs “images”. Ce sont des fichiers spéciaux contenant des systèmes de fichiers. Un exemple d'image est un disque virtuel contenant des modules du noyau nécessaires pour démarrer votre système (initrd). Il y a différents types d'images qui permettent de :

- démarrer le programme d'installation ;
- gérer un matériel ou un besoin spécifique via des modules du noyau complémentaires ;
- dépanner le système (image rescue).

Pour installer à partir d'une disquette, vous devez copier certaines images sur la disquette. Sous Linux, on utilise le programme **dd**, **rawrite** permet de le faire sous DOS.

Une image est un fichier spécial qui peut contenir des sous-répertoires (comme un fichier archive).



Une image peut être montée sur un pseudo-périphérique. Si l'image est nommée *Image*, la commande suivante vous permettra de voir le contenu de ce fichier dans le répertoire **/mnt/floppy** :

```
mount -o loop /path/to/Image /mnt/floppy
```

dosutils: ce répertoire contient les programmes DOS qui pourront être utilisés pour préparer l'installation de Linux, comme le programme **rawrite.exe** dont nous venons de parler. Vous y trouverez également **fips** qui vous permet de partitionner un disque C:\ en deux sans perte de données et à la condition que le système de fichiers soit FAT et pas NTFS.

Installation

Installations locales

L'installation la plus courante et la plus facile est l'installation locale. La plupart des distributions sont disponibles sous la forme d'une image ISO d'un Cdrom avec un script d'installation automatique. Sur les machines sans lecteur de Cdrom, il est également possible d'installer à partir d'une disquette.

Installation à partir d'un CD-ROM

Modifiez le paramétrage du BIOS de votre machine pour démarrer à partir du CD. Le menu du programme d'installation vous permet de faire une installation avancée ou basique.

Installation à partir d'une disquette

Si vous ne pouvez pas démarrer à partir d'un CD-ROM vous allez devoir créer la disquette d'installation. Cela peut arriver lorsque le CD n'est pas amorçable ou si vous avez téléchargé une image de votre distribution à un format autre qu'ISO.

Création de la disquette d'installation

<code>dd if=/path/to/<image_name> of=/dev/fd0</code>	Sous Linux
<code>rawrite.exe</code>	sous Windows (pas NT)

Pour les distributions RedHat, les images d'installation sont dans le répertoire **images**. L'image de base est **boot.img**. Vous en trouverez de plus spécifiques comme **bootnet.img** ou **pcmcia.img**.

Sur une distribution Suse, l'image de la disquette est dans le répertoire **disks** et l'image est appelée **bootdisk**.

Installation réseau

Pour une installation de RedHat vous aurez besoin d'une disquette d'installation spécifique. Faites une disquette de démarrage en utilisant l'image **bootnet.img**.

```
dd /mnt/cdrom/images/bootnet.img of=/dev/fd0
```

Pendant la première partie de l'installation en mode texte, vous devrez paramétrer le clavier et réseau. La suite de l'installation se fait par FTP, NFS ou HTTP. Vous pouvez faire une installation en mode graphique en utilisant un protocole qui permet de monter une source réseau sur un point de montage local (NFS) mais si vous utilisez les protocoles de retrait de fichier (FTP ou HTTP), l'installation se fera en mode texte. Ceci n'est plus forcément le cas pour la plupart des distributions modernes.

Notez également que l'installation réseau est possible directement à partir du CD pour la plupart des distributions modernes (pour Mandrake à partir du disque 2 et en introduisant le paramètre **askmethod** au démarrage du CD de Fedora).

Disque de récupération

Si vous ne pouvez plus démarrer votre système Linux, vous pouvez démarrer en utilisant un disque de récupération. C'est une petite version de Linux qui montera un système de fichier virtuel minimal en mémoire.

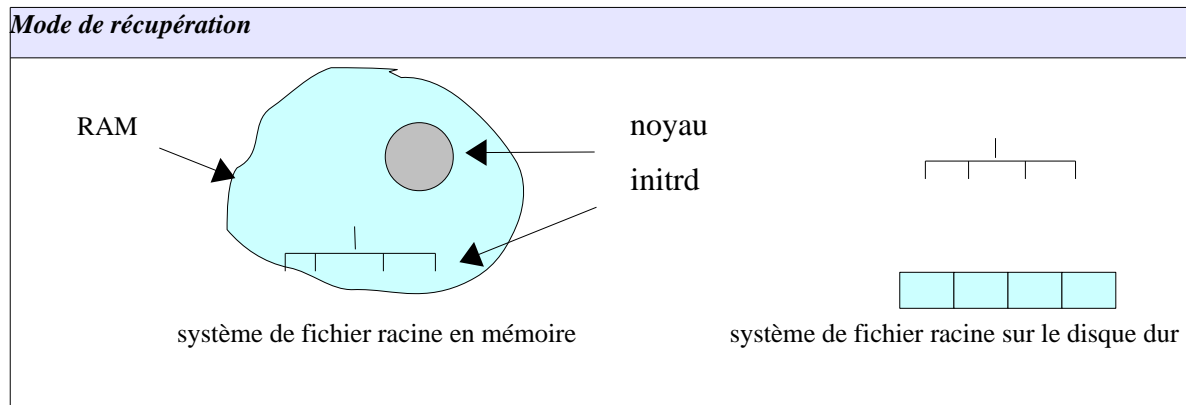
Le système d'exploitation fonctionne entièrement en mémoire vive (RAM). Il permet d'accéder à votre système de fichier racine sur le disque dur. La plupart des disques de récupération le trouvent automatiquement. Par exemple si la racine se trouve sur la première partition logique du premier disque IDE (/dev/hda5), le disque de récupération peut la monter sur un sous-répertoire du système de fichier en mémoire, par exemple /mnt/system.

Changement de perspective

Dans cette situation, nous avons **deux systèmes de fichiers racines**. Pour utiliser celui présent sur le disque dur comme racine de votre système, vous devez changer de perspective (changer de racine). C'est ce que fait la commande **chroot** :

Installation

```
chroot /mnt/sysimage
```



C'est parti

Ancienne méthode :

1. créer une disquette amorçable en utilisant l'image **boot.img** et en la copiant avec : `dd if=boot.img of=/dev/fd0`
2. copier l'image `rescue?img` sur une deuxième disquette : `dd if=rescue.img of=/dev/fd0`
3. démarrer le système avec la disquette l'image **boot.img**
4. à l'invite de LILO, tapez « linux rescue ». Vous devriez voir quelque chose comme :
Insert root file system disk:
5. insérer la disquette avec `rescue.img` et appuyer sur entrée
6. attendre l'invite du shell qui indique que le système a fini de démarrer
7. vous aurez peut-être encore à déterminer quel système de fichier est la racine (non présenté ici)

Nouvelle méthode :

1. insérer le CD d'installation de votre distribution (Suse, RedHat, Mandriva, Debian, ...)
2. tapez « linux rescue » à l'invite
3. suivez les instructions
4. le système de fichier racine devrait être détecté
5. si votre système de fichier racine est monté sur **/mnt/sysimage**, tapez la commande suivante :

```
chroot /mnt/sysimage
```

Plan de partitionnement

Le système d'exploitation utilise un mécanisme appelé montage pour accéder aux différentes ressources sur le disque dur. Sur les systèmes d'exploitation de type UNIX, cela se fait en attachant une partition à un répertoire, appelé point de montage.

L'illustration un peu plus bas vous montre un plan de partitionnement envisageable. Des ressources différentes, partitions, partages réseau, CD-ROMs, etc. sont attachés sur différents points de montage.

Du point de vue de l'utilisateur, tout est vu comme une arborescence simple composée de répertoires et sous-répertoires.

Arborescence du système de fichiers

La racine de l'arborescence est appelée root et est représentée par un slash : « / ». Le point de montage / est également le premier répertoire attaché à une ressource (périphérique racine) par le système d'exploitation.

Installation

Une fois la racine montée, les répertoires et sous-répertoires présents sur ce périphérique peuvent également être utilisés comme points de montage pour d'autres périphériques pour compléter l'arborescence des répertoires.

Ce processus se produit de la façon suivante :

1. le chargeur de démarrage charge le noyau et lui indique où se trouve la racine (cf « Démarrage de Linux » dans le cours pour la préparation de l'examen LPI 102).
2. Les autres répertoires sont montés à partir des informations du fichier `/etc/fstab` (voir p.Error: Reference source not found)

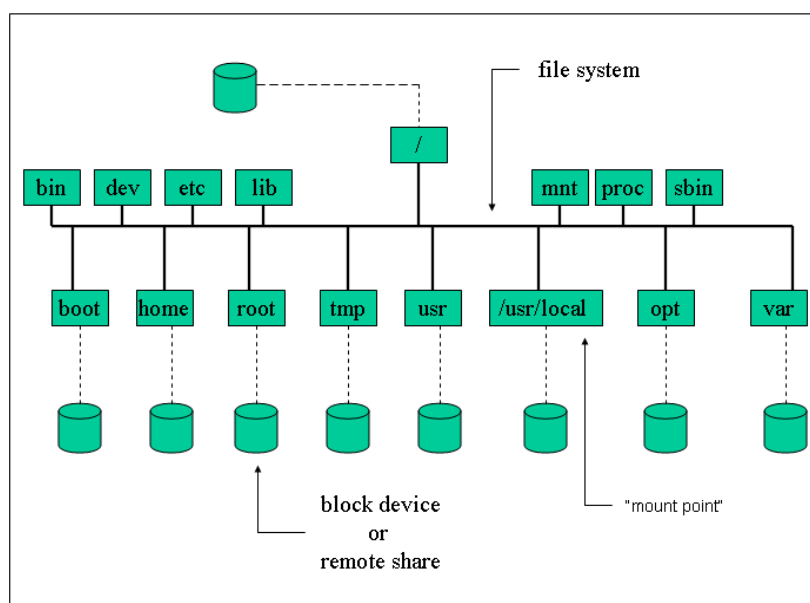


Illustration 1: Points de montage sur le système de fichiers

Partitionnement du disque

Lorsque vous installez Linux vous devez créer votre partitionnement. Cette étape de l'installation se fait généralement en passant par un outil graphique comme Yast ou DiskDruid. Ces programmes ont trois fonctions :

- créer des partitions d'une taille donnée ;
- sélectionner le type de système de fichiers (voir p.Error: Reference source not found) ;
- définir le point de montage pour chaque partition

Si vous utilisez le mode expert pour une installation, vous utiliserez sans doute `fdisk` (voir p.Error: Reference source not found) pour uniquement créer les partitions.

Au minimum, vous aurez besoin de deux partitions : une partition racine et une partition d'échange (pour la mémoire paginée). Il n'y a pas véritablement de règle le partitionnement, il se choisit en général là partir de a fonction de l'ordinateur (poste de travail, serveur de mail etc.).

La partition SWAP

Vous devez choisir la taille de la partition d'échange à sa création. Encore un fois, il n'y a pas de règle. La taille nécessaire dépend du type d'application qui seront lancées sur le PC (bureautique, serveur, applications 3D, etc.). En général cependant, pour un noyau 2.4 avec une quantité mémoire vive assez moyenne (c'est à dire moins de 256 Mo), on crée une partition d'échange du double de la taille de la RAM. Sur un vieux noyau 2.2, nous aurions donné à cette partition la même taille que la quantité de RAM.

En général, la pagination se fait à partir d'une partition. Dans la table des partitions, la valeur hexadécimale

Installation

correspondant à la partition SWAP est 82.

REMARQUE

Contrairement aux partitions utilisées pour le stockage de données, une partition swap n'est jamais montée. Il n'y a donc pas de point de montage pour ce type de partition. Pour créer une partition SWAP à l'installation, il suffit de sélectionner le système de fichier SWAP (ou partition d'échange).

Une fois le système démarré, les informations concernant les partitions SWAP sont dans le fichier `/proc/swaps`.

Vous pouvez également créer des fichiers d'échange au lieu des partitions (voir LPI 201). C'est souvent utilisé en situation d'urgence sur un système en marche mais rarement à l'installation.

Plusieurs systèmes d'exploitation sur une machine

(Cette partie n'est pas nécessaire à la préparation de l'examen).

Si Windows est déjà installé sur le système, le programme d'installation configurera automatiquement LILO ou GRUB pour permettre de démarrer les deux systèmes.

Pré-installation

Avant toute chose vous devriez lancer un programme de défragmentation du disque. Vous aurez ainsi l'assurance que les données utilisées par Windows se trouvent au début du disque.

Ensuite, utilisez un programme comme PartitionMagic ou fips pour partitionner le disque en deux. Les programmes Windows se trouvent sur la première partition. La seconde partition devra être suffisamment grande pour contenir Linux.

Remarque : l'espace moyen nécessaire pour une distribution Linux est de 4 Go.

Démarrer l'installation à partir de DOS

Pour les Windows 9x, redémarrez votre ordinateur en commande DOS. Si vous installez une RedHat, tapez la commande `E:\DOSUTILS\AUTOBOOT.BAT` qui lancera le programme d'installation. Pour installer une Suse, tapez la commande `E:\setup.exe` sous DOS.

Le disque dur vu par Windows

Sous Windows, le système ne voit que les partitions FAT et NTFS. Les partitions Linux sont inaccessibles.

Le disque dur vu par Linux

Sous Linux, la partition Windows devrait être nommée `/dev/hda1` (pour la première partition du premier disque dur). Par défaut, cette partition n'est pas montée. Vous pouvez créer un répertoire `/dos` ou `/mnt/dos` et la monter et y accéder.

Exercices et résumé

Questions (réponses p.Error: Reference source not found)

Oui ou non

1. Le programme rawrite fonctionne sous Linux et est utilisé pour copier un fichier image sur une disquette _____
 2. À la préparation d'un nouveau plan de partitionnement sur un disque vierge, n'importe quelle partition peut être choisie comme partition racine _____
-

Glossaire

<i>Terme</i>	<i>Description</i>
Système de fichiers virtuel	Un système de fichiers est une structure de données qui permet d'organiser les données sur un disque. Pour l'utilisateur, les données se trouvent dans une arborescence de répertoires dont la racine est appelée « root » et est notée « / ». Cette structure est également appelée système de fichiers virtuel puisque l'arborescence ne change pas selon soit l'organisation du disque ou le partitionnement. Sous un système DOS, la situation est radicalement différente : par exemple si votre disque est découpé en 4 partitions, les utilisateurs devront savoir que les données peuvent se trouver sur C:\, D:\, E:\, ou F:\ et dans ce cas, le premier lecteur de CD-Rom sera nommé G:\.
Point de montage	Répertoire sur lequel est lié une partition pour la rendre accessible au système.
Plan de partitionnement	Le partitionnement est réalisé à l'installation. Il comprend le découpage du disque en partitions et le choix des points de montage pour les relier au système de fichier virtuel sur lequel le système d'exploitation est installé. La FHS (FileSystem Hierarchy Standard – Norme de hiérarchie du système de fichiers) indique où sont installés les composants logiciels, où se trouvent les répertoires personnels des utilisateurs etc. Vous devez en tenir compte pour le découpage de votre système en fonction de son rôle. Par exemple la plupart des logiciels sont installés dans le répertoire /usr. Par conséquent, assurez-vous que cette partition est suffisamment grande, au minimum 2
Mode de récupération	Il s'agit d'un système Linux minimal avec juste assez d'outils pour accéder au disque. On utilise en général ce mode en démarrant à partir d'un CD-Rom d'installation, le système fonctionne entièrement en mémoire vive.
root (/)	Répertoire le plus haut de l'arborescence sur lequel est attaché la partition racine. L'ensemble des répertoires et sous-répertoires peut se trouver sur cette partition ou d'autres sous-répertoires peuvent servir de point de montage pour d'autres partitions, cela dépend du plan de partitionnement choisi à l'installation.

Commandes

<i>Commande</i>	<i>Description</i>
chroot	Va dans un répertoire (comme avec cd) et considère ce répertoire comme la racine (/). Par défaut, chroot tentera de lancer le shell Bash /bin/bash , mais vous pouvez spécifier d'autres commandes à lancer (voir « serveurs chrootés » dans LPI202).
dd	Outil qui permet de copier des fichiers tout comme des parties d'un périphérique (disque dur, CD-ROM ou disquette). Les CD d'installation contiennent en général des fichiers que l'on appelle des fichiers image qui peuvent être recopiés sur disquette en utilisant cette commande.
fips.exe	Programme disponible sur la plupart des CD d'installation utilisé pour retailer une

Installation

	partition FAT, afin de récupérer de la place pour installer Windows et Linux en double amorçage.
rawrite	Equivalent à dd sous DOS

Exercices

- Procédez à une installation locale à partir d'un CD-ROM en suivant la stratégie indiquée ci-dessous. Seuls les utilisateurs avancés familiers de l'éditeur de texte vi et de la gestion de paquets devraient tenter les étapes FACULTATIVES.
 - Type d'installation : choisissez « personnalisée »
 - Partitionnement : partitionnez le disque manuellement avec Disk Druid

Voici une proposition pour un espace disque de 3 Go. Si vous disposez de plus d'espace, faites une partition /usr plus grande et installez plus de paquets que ceux indiqués à l'étape (iv). ATTENTION : laissez une partition libre d'au moins 100 Mo, nous en aurons besoin plus tard.

/boot 20M

/ 250M

/usr 2300M

/home 50M

/tmp 100M

/var 150M

SWAP 128M Remarquez que SWAP est le type de système de fichier et que cette partition n'est pas montée (voir p.29)

- (FACULTATIF) Installez LILO sur /dev/hda2 ou ne l'installez pas, en tout cas ne l'installez pas sur le MBR (/dev/hda) nous souhaitons délibérément que le système n'arrive pas à démarrer. Le chargeur de démarrage sera remis à l'étape 2 (i)
 - Paquets à installer (les noms diffèrent suivant la distribution) :
 - « Interface graphique X » + « Environnement de bureau GNOME » OU « Environnement de bureau KDE »
 - « Editeurs »
 - « Outils graphiques pour Internet »
 - « Développement logiciel » (Ceci a son importance puisque nous compilerons des paquets plus tard)
 - Ne créez pas la disquette de démarrage
- (FACULTATIF) Dépanner le système
 - Redémarrez à partir du CD-ROM d'installation. À l'invite, tapez :
linux rescue
 - Lisez les informations à l'écran. À l'invite, utilisez la commande chroot comme nous l'avons vu plus tôt.
 - Le paquet lilo doit être installé. Éditez le fichier /etc/lilo.conf avec vi. Vous devriez avoir :


```
boot=/dev/fd0
prompt
linear
timeout=50
image=/boot/vmlinuz-<kernel-version>
label=linux
read-only
root=/dev/<root-partition>
```
- (iv) lancez /sbin/lilo. En cas d'erreur, vous aurez peut-être à remplacer linear par lba32

Configuration matérielle

Pré-requis

Aucun

Objectifs

- Comprendre les affectations de ressources matérielles (interruptions – IRQ, ports d'entrée / sortie et accès direct à la mémoire DMA)
 - Avoir une vue d'ensemble des matériels comme les cartes d'extension et les périphériques USB et SCSI
 - Détection des cartes réseau et des imprimantes (sans configuration)
 - Comprendre les étapes de base pour la configuration des modems et des cartes son
-

Contenu

1. Allocation de ressources.....	2
Liste des ressources allouées.....	2
2. Cartes d'extension.....	3
3. USB.....	3
Les contrôleurs USB.....	3
Hotplug (branchement à chaud).....	3
usbmgr.....	4
4. SCSI.....	4
Types d'interfaces SCSI.....	4
Le canal SCSI.....	4
Le numéro identifiant du périphérique.....	4
Le numéro logique LUN.....	4
Détection du matériel.....	5
Démarrer à partir d'un disque SCSI.....	5
5. Cartes réseau.....	5
6. Configuration des modems.....	6
Le modem.....	6
Le port série.....	6
7. Configuration des imprimantes.....	9
Utilisation de printtool.....	10
Utilisation de cups.....	10
8. Cartes son.....	10
Détection de la carte son.....	11
Utilisation de sndconfig (objectif LPI101).....	11
Glossaire.....	12
Ressources.....	12
Fichiers.....	12
Commandes.....	13
Exercices.....	13

Configuration matérielle

Allocation de ressources

Pour que les périphériques d'un PC puissent communiquer directement avec les ressources du système, essentiellement le processeur, le système alloue des ressources (canaux, lignes) à chaque périphérique. Ces ressources sont les requêtes d'interruption matérielle (IRQ), les adresses d'entrée sortie (I/O) et les canaux d'accès directs à la mémoire (DMA).

Interruptions (IRQs) : Les périphériques utilisent les interruptions matérielles pour demander du temps processeur. Le processeur arrête son activité et procède aux instructions demandées par le matériel. Sur PC il y a 16 interruptions directement connectées sur les périphériques, elles vont de **0** à **15**.

Entrées/sorties (I/O address) : Les entrées / sorties correspondent à des adresses spécifiques dans la mémoire système. Le processeur et le périphérique communiquent en écrivant et en lisant à ces adresses.

Accès direct à la mémoire (DMA) : certains périphériques accèdent à la mémoire système au travers d'un canal d'accès direct à la mémoire. Ainsi ils accèdent aux données sans passer par le processeur, ce qui peut améliorer les performances.

Liste des ressources allouées

Le noyau inscrit dans les fichiers suivants sur /proc les ressources utilisées par le matériel :

```
/proc/dma
/proc/interrupts
/proc/ioports
/proc/pci
```

Les commandes **lspci** et **dmesg** permettent également d'avoir la liste des ressources allouées.

lspci : Liste les informations du chipset pour tous les périphériques PCI. Vous obtiendrez les IRQs et les ports d'entrées / sorties avec l'option **-v**. L'option **-b** affiche les ressources affectées par le BIOS et non plus par le noyau.

dmesg : Affiche les messages du noyau en continu. Cette commande affiche également les messages affichés au démarrage du système à l'étape de chargement du noyau (kernel stage). À cette étape le noyau recherche les périphériques et charge les pilotes (modules ou inclus au noyau) automatiquement. Ces informations se trouvent dans /var/log/dmesg.

Ressources types

Périphérique	Port d'entrée/sortie	IRQ
/dev/ttyS0	0x03f8	4
/dev/ttyS1	0x02f8	3
/dev/lp0	0x378	7
/dev/lp1	0x278	5
soundcard	0x220	

Affectation de ressources manuelle

REMARQUE :

L'exemple suivant est très courant, mais comme les modules du noyau ne seront vus que pour les LPI 102, certains auront des difficultés à le comprendre. Vous pouvez sans problème passer cet exemple et vous rendre à la partie 2.

Exemple : configuration de deux cartes réseau

- Vous pouvez passer des paramètres au noyau au démarrage pour les pilotes inclus au noyau. Dans cet exemple typique, vous disposez de deux cartes réseau, mais une seule est détectée. La commande suivante indique au noyau :
 - qu'une carte utilise l'IRQ 10 et le port d'entrée sortie 0x300 ;
 - que l'autre carte utilise l'IRQ 9 et le port d'entrée sortie 0x340

Configuration matérielle

```
ether=10,0x300,eth0    ether=9,0x340,eth1
```

Vous pouvez taper cette commande en option au prompt de LILO ou GRUB. Sinon, comme pour les paramètres de mémoire vus précédemment, ajoutez ces informations au champ `append=` dans `/etc/lilo.conf` ou `/etc/grub.conf` (ou `/boot/grub/grub.conf`).

Vous pouvez constater que l'instruction `ether=` est une commande générique pour le noyau comme `root=`, `mem=` ou `init=`. Notez également que vous n'avez pas besoin de donner d'autres informations sur les cartes réseau (Intel, netgear, etc.).

- Vous pouvez passer les mêmes paramètres aux modules (pilotes de périphériques chargés dynamiquement) en utilisant `/etc/modules.conf` ou `/etc/conf.modules`. Si l'on part du principe que dans l'exemple précédent les deux cartes utilisent le module `e100.o`, alors le fichier `/etc/modules.conf` contiendrait :

```
alias eth0 e100
alias eth1 e100

options eth0 io=0x300 irq=10
options eth1 io=0x340 irq=9
```

Cartes d'extension

Les cartes d'extension les plus courantes sont les cartes ISA et PCI. Il n'y a presque plus rien à faire pour les configurer depuis les noyaux 2.4. Cependant, pour les bus ISA et avec des noyaux plus anciens, il était nécessaire de scanner le bus pour détecter les cartes présentes (son, ethernet, etc.).

Le programme **pnpdump** fournit par le paquet `isapnptools` permet de scanner le bus ISA pour détecter les périphériques « Plug and Play ». Il renvoie en sortie le chipset de la carte ainsi que les paramètres d'adresses d'entrée sortie, DMA et IRQ. Si la sortie du programme est sauvegardée dans le fichier `/etc/isapnp.conf`, les périphériques ISA sont configurés au démarrage par **isapnp**.

Depuis les noyaux 2.4, l'initialisation des périphériques PnP est géré par le module `isapnp.o`

USB

L'Universal Serial Bus (USB) est un bus informatique plug-and-play à transmission série servant à brancher des périphériques informatiques à un ordinateur. Il existe quatre types de périphériques USB :

- Les périphériques d'affichage (écrans, etc.)
- Les périphériques de communication (modems, etc.)
- Les périphériques audio (cartes son)
- Les périphériques de stockage de masse (disques, etc.)
- Les interfaces Homme / Machine (HID)

Ces périphériques sont branchés sur un port USB qui est géré par un contrôleur USB. Linux gère les contrôleurs USB depuis les noyaux 2.2.7 (voir le Linux USB sub-system HOWTO)

Les contrôleurs USB

Il y a 3 types de contrôleurs USB :

Contrôleur USB	Module du noyau
OHCI (Compaq)	<code>usb-ohci.o</code>
UHCI (Intel)	<code>usb-uhci.o</code>
EHCI (USB v 2.0)	<code>ehci-hdc.o</code>

La commande **lsusb** donne la liste des périphériques USB connectés sur la machine :

Configuration matérielle

```
lsusb
Bus 001 Device 001: ID 0000:0000
Bus 001 Device 002: ID 04a9:1055 Canon, Inc.
```

Hotplug (branchement à chaud)

Hotplug est un système utilisé pour tenir à jour l'état du système lorsque vous branchez ou débranchez des périphériques amovibles. Dans la plupart des cas, le noyau signale les événements en envoyant des paramètres au script `/sbin/hotplug`.

Le script **hotplug** lance les scripts situés dans le répertoire `/etc/hotplug.d` (le script par défaut est **default.hotplug**) qui lance à son tour l'agent approprié défini dans `/etc/hotplug`. Les noms des agents correspondent aux type de connexions : `ieee1394`, `net`, `pci` et `usb`.

Le journal ci-dessous montre ce qu'il se passe lorsqu'on branche une caméra USB :

1^{ère} étape : Les modules du noyau identifient l'évènement et les identifiants de produit et de vendeur :

```
13:26:19 kernel: hub.c: new USB device 00:07.2-1, assigned address 5
13:26:19 kernel: usb.c: USB device 5 (vend/prod 0x4a9/0x3058) is not claimed by any
active driver.
```

2^{ème} étape : les arguments sont passés au script `default.hotplug` :

```
13:26:19 default.hotplug[10507]: arguments (usb) env (DEVFS=/proc/bus/usb OLDPWD=/
PATH=/bin:/sbin:/usr/sbin:/usr/bin ACTION=add PWD=/etc/hotplug HOME=/ SHLVL=2
DEVICE=/proc/bus/usb/001/005 PRODUCT=4a9/3058/1 TYPE=255/255/255 DEBUG=yes
_=/bin/env)
```

3^{ème} étape : l'agent `usb` (`usb.agent`) associe le matériel à une caméra usb (en utilisant `usb.usermap`)

```
13:26:19 default.hotplug[10507]: invoke /etc/hotplug/usb.agent ()
13:26:23 usb.agent[10507]: Setup usbcam for USB product 4a9/3058/1
13:26:23 usb.agent[10507]: Module setup usbcam for USB product 4a9/3058/1
13:26:38 devlabel: devlabel service started/restarted
```

À partir de cet exemple, vous pouvez voir que la première étape utilise les modules du noyau et que les étapes suivantes utilisent hotplug. Il faut également avoir une bonne carte des matériels USB (`usb.usermap`) pour pouvoir utiliser le matériel.

usbmgr

Il existe sur les systèmes Debian une alternative à hotplug : `usbmgr`. Voici les fichiers principaux à connaître :

<code>/usr/sbin/usbmgr</code>	Le service à l'écoute des événements en rapport avec l'USB
<code>/usr/sbin/dump_usbdev</code>	Un outil permettant de lister les périphériques USB connectés (comme <code>lsusb</code>)
<code>/etc/usbmgr/usbmgr.conf</code>	Fichier de configuration contenant les identifiants de matériel

SCSI

Types d'interfaces SCSI

Il y a 2 (NdT : maintenant 3) types d'interfaces SCSI

Configuration matérielle

- SCSI-1 : interface 8 bits avec un bus supportant 8 périphériques, contrôleur inclus, donc 7 périphériques réels ;
- SCSI-2 : interface 16 bits (WIDE). Le bus peut gérer 16 périphériques, contrôleur inclus (donc 15).

Les périphériques SCSI sont identifiés de façon unique à partir du triplet que l'on nomme SCSI_ID :

- le canal SCSI
- l'identifiant du périphérique
- le numéro logique LUN (Logical Unit Number)

Le canal SCSI

Chaque carte SCSI utilise un canal de données sur lequel s'attachent les périphériques (disque, CDROM, etc.). Les canaux sont numérotés à partir de 0.

Le numéro identifiant du périphérique

Un identifiant unique (Device ID) est attribué à chaque périphérique manuellement à l'aide de cavaliers directement sur le périphérique. Sur une carte SCSI-1, les identifiants vont de 0 à 7 et donc de 0 à 15 pour une carte SCSI-2.

Le numéro logique LUN

Les numéros logiques sont utilisés pour faire la différence entre les périphériques à l'intérieur d'un périphérique identifié : partitions d'un disque ou lecteur DAT pour un robot qui en comprend plusieurs. C'est devenu rare de nos jours avec la baisse des coûts des cartes SCSI et leur plus grande capacité.

Détection du matériel

La liste des périphériques détectés se trouve dans le fichier `/proc/scsi/scsi`. L'exemple suivant est tiré du SCSI-2.4-HOWTO :

```
Attached devices:

Host: scsi0 Channel: 00 Id: 02 Lun: 00

Vendor: PIONEER Model: DVD-ROM DVD-303 Rev: 1.10

Type: CD-ROM ANSI SCSI revision: 02

Host: scsi1 Channel: 00 Id: 00 Lun: 00

Vendor: IBM Model: DNES-309170W Rev: SA30

Type: Direct-Access ANSI SCSI revision: 03
```

La commande `scsi_info` affiche le SCSI_ID et le modèle du périphérique à partir des informations de `/proc/scsi/scsi`. À partir de l'exemple précédent, voici la sortie de la commande :

```
scsi_info /dev/sda

SCSI_ID="0,0,0"

MODEL="IBM DNES-309170W"
```

Configuration matérielle

```
FW_REV="SA30"
```

Démarrer à partir d'un disque SCSI

Par défaut le système démarre à partir du périphérique ayant le SCSI_ID 0. Vous pouvez modifier ce comportement dans le BIOS de la carte qui est accessible au démarrage.

Si le système est composé de disques IDE et SCSI, vous devrez commencer par paramétrer le BIOS de la machine pour démarrer sur un périphérique SCSI.

Cartes réseau

Le pilote de la carte réseau (NIC) doit être compilé avec le noyau (ou en module). Vous trouverez des informations sur la ou les cartes utilisées sur votre machine avec :

dmesg, lspci, scanpci, /proc/interrupts, /sbin/lsmodule ou /etc/modules.conf:

- dmesg

```
dmesg

Linux Tulip driver version 0.9.14 (February 20, 2001)
PCI: Enabled device 00:0f.0 (0004 ->0007)
PCI: Found IRQ 10 for device 00:0f.0
eth0: Lite-On 82c168 PNIC rev 32 at 0xf800, 00:0A:CC:D3:6E:0F,
IRQ 10
eth0: MII transceiver #1 config 3000 status 7829 advertising
```

- /proc/interrupts

```
cat /proc/interrupts

0:      8729602          XT-PIC  timer
1:         4           XT-PIC  keyboard
2:         0           XT-PIC  cascade
7:         0           XT-PIC  parport0
8:         1           XT-PIC  rtc
10:     622417          XT-PIC  eth0
11:         0           XT-PIC  usb-uhci
14:     143040          XT-PIC  ide0
15:     180            XT-PIC  ide1
```

- lsmod

```
/sbin/lsmodule
```

Configuration matérielle

Module	Size	Used by
tulip	37360	1 (autoclean)

Dans l'exemple ci-dessus la carte est basée sur un chipset Tulip, que l'adresse d'entrées / sorties est 0xf800 et que l'IRQ est 10. Ces informations peuvent être utilisées pour insérer le module avec une adresse d'entrées/sorties différente en utilisant **modprobe** ou **insmod** ou en inscrivant les valeurs dans **/etc/modules.conf** (dans ce cas les paramètres seront utilisés au prochain démarrage).

Configuration des modems

Il nous faut d'abord détecter le modem. S'il s'agit d'un modem externe, il vous suffit de connaître le port série utilisé. Par contre, s'il s'agit d'un modem PCI, nous devons connaître l'adresse d'entrées sorties et l'interruption qu'il utilise.

Le modem

Pour les modems externes, allez à la partie suivante : « le port série ».

Vous pouvez détecter les modems PCI à l'aide de la commande **lspci** (l'exemple qui suit est tiré du PCI-Modem micro HOWTO).

```
lspci -v
----- snip -----
00:0c.0 Serial controller:US Robotics/3Com 56K FaxModem Model 5610 (rev 01)
(prog-if 02 [16550])
Subsystem: US Robotics/3Com USR 56k Internal FAX Modem (Model 2977)
Flags: medium devsel, IRQ 11
```

Notez que l'adresse d'entrées sorties est 0xe800 et l'IRQ 11. Nous pouvons utiliser ces informations et les attribuer à un port série.

Le port série

Le modem communique à travers une interface série. Les informations sont transmises à travers le réseau téléphonique sous la forme de séquences de bits (série) sur deux fils (entrant et sortant). Les informations entrantes sont repassées de série en parallèle pour le bus du PC et vice versa pour les données qui partent de l'ordinateur. La traduction est effectuée par une puce **UART** située au niveau port série sur la carte mère ou directement sur la carte PCI pour un modem interne.

Pour connaître les ports série détectés au démarrage, entrez la commande suivante :

```
dmesg | grep ttyS

/dev/ttyS0, UART: 16550A, Port: 0x03f8, IRQ: 4
/dev/ttyS1, UART: 16550A, Port: 0x02f8, IRQ: 3
```

Pour le moment, ces ports utilisent les adresses couramment utilisées par des ports série.

REMARQUE

Configuration matérielle

Pour les modems externes, ne considérez que les ports série avec les IRQ 3 ou 4. Les adresses d'entrée sorties ci-dessus sont également standard.

Le tableau ci-dessous présente les équivalents entre les ports COM de DOS et les ports série sur Linux :

DOS	Linux
COM1	/dev/ttyS0
COM2	/dev/ttyS1
COM3	/dev/ttyS2

Vous pouvez aussi utiliser la commande **setserial** pour rechercher les périphériques série. L'option -g affiche les ports série utilisés.

```
setserial -g /dev/ttyS[01]

/dev/ttyS0, UART: 16550A, Port: 0x03f8, IRQ: 4
/dev/ttyS1, UART: 16550A, Port: 0x02f8, IRQ: 3
```

Si vous avez un modem interne détecté par lspci page 40, vous devez garder en mémoire l'IRQ et l'adresse d'entrées sorties utilisées :

<i>Adresses utilisées p 40 par le modem interne</i>	
Adresse d'entrées sortie	0xe800
IRQ	11

Cette fois-ci nous allons utiliser **setserial** pour attribuer ces adresses à un port série virtuel :

```
setserial /dev/ttyS4 port 0xe800 irq 11 autoconfig
```

L'option *autoconfig* permet de paramétrer automatiquement l'UART. Vous pouvez sauver cette commande dans un script que vous appellerez *serial.rc* qui se chargera de configurer le port série à chaque démarrage.

On utilise généralement un lien symbolique **/dev/modem** pointant vers le port série utilisé par le modem.

Création manuelle du lien symbolique :

```
ln -s /dev/ttyS1 /dev/modem
```

On utilise aussi **setserial** pour configurer la vitesse du port série.

<i>Option de setserial</i>	<i>Description</i>
spd_hi	Utiliser 56 kbaud au lieu de 38,4 kbaud
spd_vhi	Utiliser 115kbaud au lieu de 38,4 kbaud
spd_shi	Utiliser 230 kbaud au lieu de 38,4 kbaud
spd_warp	Utiliser 460kbaud au lieu de 38.4 kbaud
spd_cust	Paramètre la vitesse à partir du diviseur (vitesse = vitesse de base en baud / diviseur)
spd_normal	Utiliser 38,4 kbaud

Par exemple, si vous voulez paramétrer la vitesse du port série /dev/ttyS4 à 115 kb :

```
setserial /dev/ttyS4 spd_vhi
```

Configuration matérielle

Configuration des connexions

Le script de configuration `wvdialconf` du paquet `wvdial` recherche les modems série ou USB sur la machine et génère un fichier de configuration de base.

Exemple de fichier `/etc/wvdial.conf` :

```
[Dialer Defaults]
Modem = /dev/ttyS1
Baud = 115200
Init1 = ATZ
Init2 = ATQ0 V1 E1 S0=0 &C1 &D2 S11=55 +FCLASS=0
; Phone = <Target Phone Number>
; Username = <Your Login Name>
; Password = <Your Password>
```

Pour commencer, remplacez *Defaults* par le nom de votre fournisseur d'accès, par exemple *MonFAI*. Entrez également votre nom d'utilisateur, mot de passe et numéro de téléphone, puis tapez la commande suivante :

NdT : décommentez ces lignes en retirant les « ; ».

```
wvdial WorldISP
```

Vous pouvez également utiliser **minicom** pour votre connexion. Il se configure en utilisant l'option `-s` :

```
minicom -s

                                     [configuration]

    Filenames and paths
    File transfer protocols
    Serial port setup
    Modem and dialing
    Screen and keyboard
    Save setup as dfl
    Save setup as..
    Exit
    Exit from Minicom
```

Tout ce qui concerne le port série se trouve dans « *Serial port setup* » et les informations de connexion communiquées par votre fournisseur s'inscrivent dans « **Modem and dialing** ».

Vous pouvez vous connecter à d'autres machines une fois que le modem est paramétré et fonctionnel, mais il faudra lancer **pppd** pour établir une connexion réseau. **pppd** va créer l'interface réseau `ppp0` et assigner une adresse IP (fixe ou via DHCP). Des programmes comme **wvdial** s'en chargent tous seuls.

Winmodems

Si jamais la configuration précédente ne fonctionne pas, vous avez de fortes chances (façon de parler) d'avoir un

Configuration matérielle

winmodem, qui va avoir besoin de pilotes supplémentaires. Pour avoir une définition fleurie des *winmodem*, jetez un oeil au [winmodems-and-Linux-HOWTO](#).

Un winmodem qui peut fonctionner sous Linux est aussi appelé un linmodem (allez voir le linmodem-HOWTO si vous souhaitez en savoir un peu plus).

RNIS

RNIS c'est le réseau téléphonique classique numérisé jusqu'à la prise du client. C'est le Réseau Numérique à Intégration de Services, commercialisé sous le nom de Numéris en France par France Telecom. Au lieu d'une ligne analogique à 56kbits/s, le RNIS offre un certain nombre de canaux (suivant les contrats) pour les données, les canaux B à 64 kbit/s, et un canal de plus faible débit, le canal D qui gère les informations de liaison. Le service de base dans un certain nombre de pays est RNIS2 : 2 canaux B et un canal D.

Vous pouvez vous connecter en RNIS de différentes façons sous Linux. La plus simple est d'utiliser un routeur RNIS qui se charge de la connexion à votre place et qui se branche directement à votre PC ou votre réseau en Ethernet.

Vous aurez besoin d'une carte adaptateur si votre poste doit être directement connecté sur le réseau RNIS. Les détails concernant les différentes interfaces sortent du programme de ce cours mais vous trouverez en général :

- Interface de commandes compatible AT

Avec les adaptateurs série et certains adaptateurs USB vous pouvez utiliser les commandes `t` exactement comme si vous utilisiez un modem. Dans ce cas, vous configurez l'adaptateur comme un modem. C'est plus simple, mais moins performant que d'autres méthodes et les données sont traitées comme celles d'un modem. Certains caractères doivent être échappés avec les modems parce qu'ils sont utilisés. Sur une ligne RNIS et en utilisant une autre méthode, les données ne sont pas à filtrer.

- Cartes PCI/ISA/PCCARD et `isdn4linux`

Le moyen le plus efficace pour vous connecter à votre réseau RNIS est d'utiliser un adaptateur branché directement sur un bus de votre PC. L'objectif du projet `isdn4linux` est de masquer les détails de la connexion et de se présenter à l'utilisateur comme une interface réseau classique. Le paquet **`isdn4k-utils`** contient tous les outils nécessaires. Sur une distribution basée sur RedHat, le programme **`system-config-network`** gère la configuration.

ADSL

L'ADSL ou liaison numérique à débit asymétrique a très largement remplacé les connexions RNIS et les lignes louées (T1, etc.) pour les connexions haut débit. Tout comme pour le RNIS, l'utilisateur de Linux a plusieurs choix stratégiques. Le plus simple est d'utiliser un routeur ADSL autonome, auquel vous vous branchez directement en Ethernet. Dans ce cas, le routeur est traité comme n'importe quel routeur. De nombreux routeurs ADSL vendus actuellement sont des machines fonctionnant sous Linux. Si pour une raison ou une autre vous n'utilisez pas cette méthode, vous devrez acheter un modem ADSL et choisir un accès pouvant utiliser le protocole PPOE (Protocole Point à Point sur Ethernet). Vous trouverez différents projets de gestion de modems ADSL basés sur différents chipsets. Mais chacun utilise ses outils de configuration et son environnement.

- PPPoe

L'ADSL n'est pas un protocole : il fonctionne plutôt comme un panier interconnectant des protocoles réseau sur PPP (Protocole Point à Point). Si l'équipement et le compte ADSL sont compatibles, les utilisateurs peuvent utiliser PPPoE pour réaliser la connexion. L'équipement ADSL se charge des couches en dessous de la session PPP et transmet les paquets PPP à travers ces couches. Les outils et les informations de connexion nécessaires se trouvent dans le paquet `pppoe`.

Configuration matérielle

Configuration des imprimantes

Le processus d'impression est vu en détail dans le cours de préparation LPI102. Du point de vue matériel, les imprimantes sont automatiquement détectées au démarrage, ce que vous pouvez voir sur la sortie de **dmesg**.

L'impression sous Linux s'effectue en deux étapes. D'abord, les données brutes sont filtrées et mises à un format postscript. Ensuite, l'impression est prise en charge par le programme ghostscript ou **gs**.

Utilisation de printtool

printtool crée une entrée dans `/etc/printcap`. Les informations les plus importantes sont le chemin d'accès au filtre d'entrée (`if`), le répertoire de spool (`sd`) et le périphérique d'imprimante (`lp`).

Si **printtool** n'arrive pas à détecter le port parallèle correspondant à votre imprimante, utilisez **dmesg** pour voir ce que le noyau avait trouvé au démarrage.

L'exemple suivant présente une imprimante connectée sur le premier port parallèle `/dev/lp0` :

```
parport0: PC-style at 0x378 (0x778) [SPP,ECP,ECPEPP,ECPPS2]
parport0: detected irq 7; use procfs to enable interrupt-driven operation.
parport_probe: succeeded
parport0: Printer, HEWLETT-PACKARD DESKJET 610C
lp0: using parport0 (polling)
```

Un fichier `/etc/printcap` :

```
# This file can be edited with the printtool in the control-panel.
##PRINTTOOL3## LOCAL cdj550 300x300 a4 {} DeskJet550 3 {}
lp:\
    :sd=/var/spool/lpd/lp:\
    :mx#0:\
    :sh:\
    :lp=/dev/lp0:\
    :if=/var/spool/lpd/lp/filter:
```

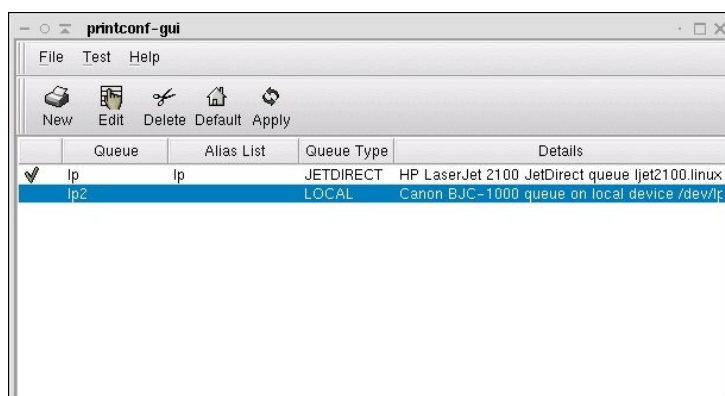


Illustration 2: **printtool**, l'interface GTK de configuration des imprimantes

Utilisation de cups

Cups est un nouvel outil d'administration et de configuration des imprimantes. Les fichiers de configuration sont situés dans `/etc/cups`. Le processus d'impression est le même sauf que **cups** utilise ses propres filtres que

Configuration matérielle

l'on trouve dans `/usr/lib/cups`.

Cups se configure à partir d'une interface WEB, service qui utilise le port TCP 631.

Cups remplace également le service lpd.

REMARQUE

Les imprimantes locales sont détectées au démarrage du système qu'elles sont parallèles ou USB. Vous pouvez afficher ces informations avec la commande `dmesg`.

Cartes son

Il y a deux projets pour la gestion du son sous Linux : OSS (Open Sound System) et ALSA (Advanced Linux Sound Architecture). OSS est un projet commercial de gestion du son sur d'autres systèmes UNIX. Des pilotes modifiés ont été introduits au noyau à partir de la version 2.0.

ALSA est un projet beaucoup plus récent intégré seulement à partir du noyau 2.6. Pour les noyaux plus anciens, il faut recompiler le noyau pour intégrer la gestion d'ALSA, sauf pour certaines distributions qui ont très rapidement adopté ALSA, comme Suse.

Dans la plupart des cas, le son fonctionne à l'installation du système. La plupart des distributions Linux incluent des outils de configuration du son graphiques.

Détection de la carte son

Pour changer on va utiliser `dmesg` pour voir si le noyau a détecté une carte son :

```
dmesg | grep -i audio
NeoMagic 256AV/256ZX audio driver, version 1.1p
Initialized NeoMagic 256ZX audio in PCI native mode
```

REMARQUE

Il est possible que cette commande ne vous renvoie rien. Dans ce cas, regardez plus en détail la sortie de `dmesg` pour tenter de déterminer quel périphérique correspond à votre carte son.

Utilisation de `sndconfig` (objectif LPI101)

Pour configurer la carte précédente, nous devons trouver le module correspondant aux informations de `dmesg`. Avec OSS, ce module est associé à un fichier périphérique qui sera utilisé par les programmes, `sound-slot-0` pour le premier périphérique de son.

Ce travail est fait automatiquement par des programmes de configuration. Nous allons voir `sundconfig`, qui est au programme du LPI101.

`sndconfig` est un programme RedHat de configuration des périphériques de son avec les modules OSS. Vous aurez sans doute besoin de l'installer car la plupart des distributions ne l'incluent plus à l'installation de base. Tapez :

```
sndconfig
```

Au lancement, un menu graphique (ncurses) apparaît et `sndconfig` vous demande s'il doit lancer la recherche de cartes sons sur votre machine. Allez sur « OK ».

Sur notre ordinateur, la carte a été détectée :

Configuration matérielle

Neomagic Corporation | NM2360 [MagicMedia 256ZX Audio]

Si `sndconfig` n'a rien trouvé, il vous présente la liste de fabricants et des cartes gérées par OSS. Dans ce cas, utilisez `lspci` pour obtenir le modèle et vérifiez qu'il est compatible à la page

<http://www.opensound.com/osshw.html>

Une fois la carte sélectionnée, **`sndconfig`** essaie de charger le module du noyau correspondant et de jouer un son (surprise !). Si tout a fonctionné, **`sndconfig`** met à jour `/etc/modules.conf` (vu dans la préparation de l'examen LPI102). Voici dans notre cas, la ligne ajoutée par `sndconfig` :

```
alias sound-slot-0 nm256_audio
```

Exercices et résumé

Révision (réponse p .Error: Reference source not found)

Oui ou non :

1. La partition racine d'un système Linux doit toujours se trouver sur un disque IDE : _____
2. Un système Linux peut utiliser n'importe quel périphérique USB du moment où le noyau a été compilé avec le support USB : _____

Glossaire

Terme	Définition
DMA	L'accès direct à la mémoire est utilisé par certains composants matériels pour effectuer des opérations de lecture / écriture directement en mémoire, sans utiliser le processeur.
Adresses d'entrées / sorties	Zones mémoires prédéfinies et réservées utilisées par les périphériques et le processeur pour des opérations de lecture / écriture.
IRQ	Signal envoyé par un périphérique au processeur pour exécuter une tâche en priorité : arrêt de l'exécution actuelle, exécution de la tâche prioritaire (par exemple saisie clavier) et reprise de la tâche précédente.
Allocation de ressources	Ensemble des ressources : DMA, adresses d'entrées/sorties et IRQ attribuées à un matériel
SCSI	Interface de transfert de données entre un matériel et le bus du PC. Par exemple, les périphériques peuvent être un disque dur, un lecteur de bandes, un lecteur Cédérom, un graveur de CD, un scanner...
USB	Connexion haute vitesse standard permettant de connecter des périphériques externes à l'ordinateur sans avoir besoin de redémarrer. Le principe est un contrôleur sur l'ordinateur avec un hub (concentrateur) initial. On branche ensuite les périphériques sur ce concentrateur ou d'autres concentrateurs, avec un maximum de 127 périphériques connectés (concentrateurs compris) par contrôleur.

Ressources

The Winmodems-and-Linux HOWTO

The Serial HOWTO

The Modem HOWTO

The Linux USB sub-system (<http://www.linux-usb.org/>)

SCSI terminology (<http://www.scsita.org/terms/scsiterms.html>)

The Linux 2.4 SCSI subsystem HOWTO

The Ethernet HOWTO

The Printing HOWTO

The Sound HOWTO

The isdn4linux project (<http://www.isdn4linux.de/>)

The Roaring Penguin PPPoE project (http://www.roaringpenguin.com/penguin/open_source_rp-pppoe.php)

Exercices et résumé

Fichiers

<i>Fichier</i>	<i>Description</i>
/etc/isapnp.conf	Fichier de configuration de isapnp – voir isapnp.conf(5)
/proc/dma	Liste des canaux DMA utilisés
/proc/interrupts	Liste des interruptions utilisées
/proc/ioports	Liste de adresses d'entrées / sorties utilisées
/proc/pci	Informations sur le bus PCI
/etc/hotplug/usb.usermap	Liste des périphériques USB reconnus
/var/log/dmesg	Fichier journal des messages du noyau depuis le démarrage
/proc/scsi/scsi	Informations sur les périphériques SCSI – voir scsi_info(8)

Commandes

<i>Commande</i>	<i>Description</i>
dmesg	Affiche les messages du noyau depuis le démarrage
hotplug	Programme utilisé par le noyau pour traiter les évènements relatifs au matériel – voir hotplug(8)
isapnp	Programme utilisé pour initialiser les cartes ISA avant les noyaux 2.4 – voir isapnp(8)
lspci	Liste des périphériques PCI – voir lspci(8)
lsusb	Liste les périphériques USB – voir lsusb(8)
pnpdump	pnpdump(8) – affiche les informations sur les cartes ISA Plug-and-Play
scsi_info	scsi_info(8) – affiche la description des périphériques SCSI
setserial	setserial(8) – affiche ou règle les paramètres des ports série
usbmgr	Service utilisateur qui charge ou décharge les modules USB. C'est l'alternative à hotplug qui est généralement utilisée sur Debian.
usb.agent	Agent hotplug traitant les évènements relatifs à l'USB
usbmodules	usbmodules(8) – liste les pilotes du noyau qui devraient pouvoir gérer les périphériques USB connectés. usbmodules peut être utilisé par /sbin/hotplug ou un de ses agents (en général /etc/hotplug/usb/agent) quand les périphériques USB sont connectés à chaud.
wvdial	Outil utilisé pour les connexions PPP, voir wvdial(1)

Exercices et résumé

--	--

Exercices

1. Utilisez **dmesg** pour voir les messages de **/var/log/dmesg**. Recherchez les mots clés USB, tty, ou ETH0
 - quels sont les noms des contrôleurs USB utilisés ?
 - Quelles interruptions sont utilisées par les deux premiers ports série ?
2. Consultez les fichiers suivants
 - /proc/ioports
 - /proc/interrupts
 - /proc/pci
 - /proc/dma
3. Le bus PCI
 - À partir des commandes **lspci -v** et **scanpci -v**, déterminez le type de votre carte ethernet
 - Vérifiez qu'il y a autant d'entrées de « bus » avec la commande **lspci** et **/proc/pci**
4. Outils USB
 - Déterminez le type de contrôleur USB sur votre système (xHCI) à partir des commandes **lsmod** et **lsusb** ;
 - Utilisez **usbmodules** pour afficher les modules du noyau pouvant utiliser l'interface
5. Périphériques SCSI
 - À partir du contenu du fichier **/proc/scsi/scsi** suivant, déduisez la sortie de la commande **scsi_info** (voir p.Error: Reference source not found):

```
Attached devices:
Host: scsi0 Channel: 00 Id: 00 Lun: 00
Vendor: PHILIPS Model: CDRW48A Rev: P1.3
Type: CD-ROM ANSI SCSI revision: 02
```

Gestion des disques

Pré-requis

- Expérience de la procédure d'installation d'un système sous Linux (voir [Installation](#))

Objectifs

- Comprendre la différence entre les partitions primaires, étendue et logiques
- Utilisation des outils de partitionnement avant et après l'installation
- Installation et configuration des chargeurs de démarrage LILO et GRUB
- Comprendre les points de montage et le rôle du fichier `/etc/fstab`

Les disques

Sur un système Linux, les disques sont représentés par des fichiers dans le répertoire `/dev`. Le noyau communique avec les périphériques en utilisant une paire de numéros majeur et mineur. Les numéros majeurs sont listés dans `/proc/devices`. Par exemple, le numéro majeur pour le premier disque dur est **3** :

Block devices:

```
1 ramdisk
2 fd
3 ide0
```

Les descripteurs de périphériques de `/dev` commencent par **hd** (IDE) ou **sd** (SCSI). Un lecteur de bande SCSI commencerait par **st**, etc. Les systèmes contiennent généralement plus d'un périphérique bloc, donc une lettre est ajoutée pour préciser le périphérique utilisé.

descripteur

Périphérique bloc physique

hda	Maître sur le premier contrôleur IDE
hdb	Esclave sur le premier contrôleur IDE
hdc	Maître sur le second contrôleur IDE
hdd	Esclave sur le second contrôleur IDE
sda	Premier disque SCSI
sdb	Second disque SCSI

Remarque : si vous insérez un disque SCSI avec un ID SCSI (par exemple 4) compris entre ceux de deux disques existants (par exemple 3 et 7), le descripteur du disque d'ID le plus haut se trouvera déplacé (par exemple il passera de *sdb* à *sdc*), ce qui peut causer quelques inconvénients non négligeables.

Partitions

Les disques peuvent être divisés en partitions. Celles-ci sont représentées par un numéro qui est ajouté au descripteur du disque :

Descripteur	Partition
hda1	Première partition sur le premier disque
hda2	Seconde partition sur le premier disque
sdc3	Troisième partition sur le troisième disque SCSI

Il est possible de créer **4 partitions principales** sur les disques IDE. Une de ces partitions peut être **étendue**. Les **partitions étendues** permettent de créer autant de **partitions logiques** que désiré. On peut aller jusqu'à 64

Gestion des disques

partitions sur un disque IDE, 16 sur un disque SCSI.

Exemple 1 : 4 partitions principales

```
+-----+
| M |   |   |   |   |
| B | 1 | 2 | 3 | 4 |
| R |   |   |   |   |
+-----+
```

Exemple 2 : 2 partitions principales dont une partition étendue (no 2) contenant 4 partitions logiques (no 5 à 8)

```
+-----+
| M |   |   | [-----] | | | |
| B | 1 | 2 | [ 5 | 6 | 7 | 8 ] |
| R |   |   | [-----] |
+-----+
```

Sortie type de fdisk -l :

```
Disque /dev/sda: 163.9 Go, 163928604672 octets
255 heads, 63 sectors/track, 19929 cylinders
Units = cylindres of 16065 * 512 = 8225280 bytes
Identifiant disque: 0x0009e33d
```

Périphérique	Amorce	Début	Fin	Blocs	Id	Système
/dev/sda1		1	122	979933+	83	Linux
/dev/sda2		123	3769	29294527+	83	Linux
/dev/sda3		3770	7416	29294527+	83	Linux
/dev/sda4		7417	19929	100510672+	f	W95 Etendu (LBA)
/dev/sda5		7417	9848	19535008+	83	Linux
/dev/sda6		9849	10456	4883728+	83	Linux
/dev/sda7		10457	11064	4883728+	83	Linux
/dev/sda8		11065	12280	9767488+	83	Linux
/dev/sda9		12281	12523	1951866	83	Linux
/dev/sda10		12524	12766	1951866	82	Linux swap / Solaris
/dev/sda11		12767	19929	57536766	83	Linux

Sur ce disque, il y a 4 partitions principales. La quatrième partition (/dev/sda4) est étendue et n'est pas utilisée directement, elle contient seulement les 7 partitions logiques sur lesquelles se trouvent les systèmes de fichiers.

Remarque : faites bien la distinction entre les différents types de partition. De plus, il est important de comprendre la convention de nommage pour les disques IDE.

Les programmes de partitionnement

Avant l'installation (non nécessaire à l'examen)

- [PartitionMagic](#)
- Fips

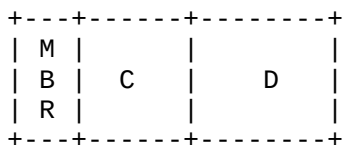
Fips ne sait gérer que les partitions en *FAT16* et *FAT32*. [PartitionMagic](#) est beaucoup plus puissant, il sait traiter les partitions UNIX.

NdT : vous pouvez également utiliser *parted* et ses interfaces *Qparted* ou *Gparted* qui feront très bien le travail.

Il n'est pas nécessaire de partitionner si vous avez déjà deux partitions DOS/Windows C:\ et D:\ et que la partition D:\ est vide (et assez grande).

```
+-----+
| M |   |   |   |   |
| B |   |   |   | C |
| R |   |   |   |   |
+-----+
```

Gestion des disques



Pendant l'installation (non nécessaire pour la préparation de l'examen)

Pendant l'installation de Linux, vous pouvez créer vos partitions et les associer à des points de montage. Les utilisateurs expérimentés peuvent le faire en deux étapes :

- Utiliser *fdisk* pour créer les partitions
- Associer un point de montage à chaque partition

La plupart des distributions utilisent un programme facile d'accès vous permettant de réaliser ces deux étapes en une seule :

- **diskdrake**(Mandrake)
- **DiskDruid** (RedHat)

Enfin, pour les débutants et les administrateurs systèmes pressés, les dernières distributions réaliseront automatiquement le plan de partitionnement.

Sur un système en cours de fonctionnement

Sur un système installé, utilisez **fdisk** pour faire vos nouvelles partitions. Voyons la syntaxe de *fdisk* sur un exemple :

Partitionner le premier disque dur

- `fdisk /dev/hda`
- tapez **m** pour l'aide, puis créez une nouvelle partition avec **n**.
- tapez **w** pour sauver les modifications sur le disque
- **redémarrez**

Voici les 4 points à suivre pour créer de nouvelles partitions. Le quatrième point est généralement oublié. Cela force le système à relire la table des partitions dans le **MBR**.

Remarque : vous devez formater la nouvelle partition avec **mkfs** ou **mke2fs** avant de pouvoir l'utiliser.

C'est la fin de notre petit tour sur les outils de partitionnement. Attaquons-nous maintenant aux chargeurs de démarrage.

Chargeurs de démarrage ("Master Boot Record")

Le MBR est placé sur le premier secteur du disque (512 octets) et contient la table des partitions ainsi qu'un chargeur d'amorçage. Au démarrage, le chargeur (bootloader) cherche la partition active dans la table de partitions et charge le premier secteur de cette partition.

Le chargeur de démarrage pour Linux : LILO

Grosso-modo, il y a trois choses à connaître sur LILO :

- **LILO** : installé dans le MBR, il charge la seconde étape de démarrage (second stage loader), en général **/boot/boot.b**
 - **/etc/lilo.conf** : c'est le fichier de configuration de LILO. En voici les options :
-

Gestion des disques

- **boot** `<nowiki>*</nowiki>` où installer LILO (Le MBR du premier disque est /dev/hda)
- **install** deuxième chargeur (par défaut **boot.b**)
- **prompt** invite utilisateur (présente éventuellement un choix de système à démarrer)
- **default** nom de l'image lancée par défaut
- **timeout** utilisé conjointement avec prompt, pause avant le démarrage par défaut (en dixièmes de secondes)
- **image** `<nowiki>*</nowiki>` chemin d'accès au noyau à lancer (il est possible d'utiliser other pour lancer un autre chargeur)
- **label** `<nowiki>*</nowiki>` Nom de l'image, c'est le nom que l'utilisateur peut entrer au démarrage
- **root** `<nowiki>*</nowiki>` nom du périphérique contenant le système de fichiers racine /
- **read-only** `<nowiki>*</nowiki>` monter le système de fichiers en lecture seule pour que **fsck** fonctionne normalement
- **append** paramètres pour le chargement des pilotes inclus au noyau (statiques)
- **linear/lba32** ces options s'excluent mutuellement. Les deux options indiquent à LILO de lire le disque en utilisant l'adressage par bloc logique (LBA). linear est typiquement utilisé sur les disques très grands. lba32 est utilisé pour accéder à des données au delà des premiers 1024 cylindres au démarrage (voir aussi p 10(UNDEFINED REFERENCE: "1024 cylinder (gloss)")).
- **/sbin/lilo** : c'est le programme binaire qui installe LILO dans le MBR en fonction de la configuration dans **/etc/lilo.conf**.

Lancez **/sbin/lilo** à chaque modification de **/etc/lilo.conf**.

GRUB (Grand Unified Bootloader)

GRUB s'installe également dans le MBR. On peut modifier le contenu du MBR en utilisant le shell de grub **/sbin/grub** ou en utilisant le fichier de configuration de grub **/boot/grub/grub.conf** qui est lu par **/sbin/grub-install**.

Des informations détaillées sur GRUB sont disponibles dans les pages **info**.

Principales parties de **/boot/grub/grub.conf**

- **General** / Global
 - **default** : image qui démarre par défaut (la première entrée est 0)
 - **timeout** : temps d'affichage de l'invite en secondes
- **Image**
 - **title** : nom de l'image
 - **root** : racine du système de fichiers
 - **kernel** : chemin du noyau à partir du root défini précédemment (par ex. /vmlinuz)
 - **ro** : read-only – lecture seule
 - **initrd** : chemin d'accès au disque virtuel en mémoire (ramdisk)

Exemple de fichier **grub.conf**

```
default=0
```

Gestion des disques

```
timeout=10
splashimage=(hd0,0)/grub/splash.xpm.gz
title Linux (2.4.18-14)
    root (hd0,0)
    kernel /vmlinuz-2.4.18-14 ro root=/dev/hda5
    initrd /initrd-2.4.18-14.img
```

Gestion des partitions

Au démarrage, le fichier **/etc/fstab** attribue les points de montage aux périphériques de type bloc.

Format de **/etc/fstab**

- Périphérique **./.** point-de-montage **./.** système-de-fichier **./.** options nombre-dump **./.** nombre-fsck

Exemple de **/etc/fstab**

LABEL=/	/	ext2	defaults	1 1
LABEL=/boot	/boot	ext2	defaults	1 2
LABEL=/home	/home	ext3	defaults	1 2
/dev/fd0	/mnt/floppy	auto	noauto,owner	0 0
LABEL=/usr	/usr	ext2	defaults	1 2
LABEL=/var	/var	ext3	defaults	1 2
none	/proc	proc	defaults	0 0
none	/dev/shm	tmpfs	defaults	0 0
none	/dev/pts	devpts	gid=5,mode=620	0 0
/dev/hdc9	swap,pri=-1	swap	defaults	0 0
/dev/cdrom	/mnt/cdrom	iso9660	noauto,owner,kudzu,ro	0 0

La commande **mount** est utilisée pour rendre un périphérique accessible sur un répertoire (point de montage).

La syntaxe de **mount** est :

```
mount -t [SYSTÈME-FICHIERS] -o [OPTIONS] PÉRIPHÉRIQUE RÉPERTOIRE
```

Par exemple on montera un CD-ROM sur le point de montage **/mnt/cdrom** avec :

```
mount -t iso9660 /dev/cdrom /mnt/cdrom
```

Sur un système en fonctionnement, le fichier **etc/fstab** sert aussi de raccourci pour assigner un périphérique à un répertoire :

```
mount /dev/cdrom
```

La commande **mount** lit **/etc/fstab** et déduit le système de fichier et le point de montage. Vous noterez que dans l'exemple de **fstab** précédent, nous accédons à certains périphériques via une étiquette (LABEL). Pour affecter une étiquette à un périphérique, utilisez la commande **tune2fs** :

```
tune2fs -L /usr/local /dev/hdb12
```

Quelques options de **mount**

- **rw, ro** : lecture/écriture et lecture seule
- **noauto** : le périphérique n'est pas monté au démarrage
- **users** : le périphérique est lisible et démontable par tous les utilisateurs
- **user** : seul l'utilisateur spécifié peut démonter le périphérique
- **owner** : le propriétaire du périphérique devient celui qui l'a monté
- **usrquota** : active les quotas utilisateurs sur le périphérique
- **grpquota** : active les quotas de groupe sur le périphérique

Remarque : pour l'examen, souvenez-vous que **mount -a** montera tous les systèmes de fichier de **/etc/fstab** qui ne sont pas déjà montés et qui n'ont pas l'option **noauto**.

La commande **umount** démonte un périphérique. Sa syntaxe est :

Gestion des disques

umount PÉRIPHÉRIQUE ou POINT-DE-MONTAGE

Aviez-vous remarqué que cette commande comporte une faute d'orthographe anglaise !

Par exemple, les deux commandes suivantes démontent le CD-ROM :

```
umount /dev/cdrom
```

ou

```
umount /mnt/cdrom
```

Gestion des quotas

Les outils de gestion des quotas sous Linux permettent aux administrateurs de mettre en place les quotas sans devoir redémarrer le système. Voici les étapes à suivre :

- Éditez **/etc/fstab** et ajoutez l'option **usrquota**
- Remontez la partition :

```
mount -o remount <nowiki><device> </nowiki>
```

- Initialisez les données sur les quotas : un fichier préliminaire **aquota.user** est créé à la base du répertoire

```
quotacheck -ca
```

- Éditez les quotas pour chaque utilisateur :

```
edquota -u <user>
```

Vous devez fixer une limite dépassable (soft) et une limite fixe (hard) pour l'espace disque utilisé (en blocs) et pour le nombre d'inodes pour chaque utilisateur. Le système autorisera l'utilisateur à dépasser la limite basse durant un certain temps (période de grâce). A l'expiration de cette période, la limite basse sera considérée comme limite haute.

- Appliquez les quotas

```
quotaon -a
```

Les utilisateurs peuvent connaître leur statut de quota avec la commande **quota**. L'administrateur système peut générer des rapports avec **repquota** ou **quotastats**.

Exercices et résumé

Questions de révision

Oui ou non :

1. Lorsque LILO est installé et que vous avez modifié le fichier lilo.conf, il n'est pas nécessaire de relancer /sbin/lilo :
2. Lorsque GRUB est installé et que vous avez modifié le fichier grub.conf, il n'est pas nécessaire de relancer grub-install :
3. Si la racine est au-delà de la limite du cylindre 1024, il est possible que votre système Linux ne démarre pas :
4. On ne peut utiliser les quotas que sur une partition dans son ensemble :

[Voir les réponses](#)

Glossaire

Terme

partition

Description

partie indépendante d'un disque dur qui peut soit être utilisée directement pour stocker des données, soit être divisée en partitions logiques. Toutes les partitions montées et manipulées par les systèmes d'exploitation ont des systèmes de fichiers séparés. De plus, elles sont considérées comme des périphériques indépendants. C'est pourquoi une partition sera parfois nommée "périphérique" parfois "système de fichiers".

partition principale

partition décrite dans l'un des enregistrements originaux de la table des partitions. Il n'y en a que 4 de disponible, donc les disques ne peuvent contenir que 4 partitions principales.

partition étendue

partition principale dont la description contient une liste de partitions, ce qui permet de créer des partitions nommées partitions logiques au delà des 4 primaires.

partition logique

partition contenue dans une partition étendue.

MBR

1er secteur du disque dur (512 octets) contenant le chargeur de démarrage et la table des partitions.

limite des 1024 cylindres

les anciens BIOS utilisent l'adressage CHS (pour Cylinder/Head/Sector = Cylindres / Têtes / Secteurs) : 10 bits pour le nombre de cylindres, 8 pour les têtes et 6 pour les secteurs. Ceci permet d'accéder à des disques d'une taille maximale de $(2^{10})^{(28)} \cdot (2^6 - 1) \cdot 512$ octets ce qui correspond à 8,5 Go. Lorsque vous lancez **/sbin/lilo**, les adresses sont données sous la forme CHS pour que le BIOS puisse les lire, à moins que vous n'utilisiez les modes linear ou lba32. Si la deuxième

Exercices et résumé

chargeur d'amorçage (ou de démarrage)	partie du chargeur d'amorçage /boot/boot.b est plus loin que le 1024ème cylindre après le MBR, le système ne pourra pas démarrer.
GRUB	code stocké sur les 512 premiers octets du disque, lu par le BIOS et utilisé pour démarrer un système d'exploitation.
LILO	GRand Unified Bootloader : chargeur d'amorçage permettant, au même titre que LILO, de choisir au démarrage de son ordinateur entre plusieurs systèmes d'exploitation.
quotas	LInux boot LOader.
grâce	restrictions définies sur le nombre d'inodes et la taille de disque pour l'utilisation d'un périphérique par un groupe ou un utilisateur.
	délai de tolérance pour les dossiers ou les fichiers qui ont dépassé la limite basse des quotas fixés.

Fichiers

Fichier

	Description
/boot/boot.b	deuxième partie (second stage) du chargeur d'amorçage LILO.
grub.conf	fichier de configuration de grub. NdT : vous trouverez également menu.lst .
/etc/fstab	fstab(5) – Le fichier fstab contient des informations décrivant les différents systèmes de fichiers. fstab est uniquement lu par les programmes, l'écriture est laissée à l'administrateur, qui doit créer et maintenir lui-même ce fichier.
/etc/lilo.conf	fichier de configuration lu par le programme d'installation du chargeur de démarrage /sbin/lilo
/proc/devices	périphériques vus sur le système et numéros majeurs associés
aquota.user	fichier de description des quotas se trouvant à la racine d'un périphérique sur lequel les quotas sont activés

Commandes

Commande

	Description ou apropos
/sbin/lilo	cf lilo(8) – installe le chargeur de démarrage
edquota	cf edquota(8) – éditeur des quotas utilisateurs
fdisk	fdisk(8) – gestionnaire de tables de partitions pour

Exercices et résumé

	Linux
grub-install	cf grub-install(8) - installe GRUB sur votre disque
mount	cf mount(8) – monte un système de fichiers
quotas	cf quota(1) – affiche l'utilisation et les limites pour un périphérique
quotacheck	cf quotacheck(8) – balaye un système de fichiers, crée, vérifie et répare les fichiers de description des quotas
quotaon / quotaoff	cf quotaon(8) / quotaoff – active ou désactive les quotas
quotastats	cf quotastats(8) – programme permettant de consulter les statistiques sur les quotas
repquota	cf repquota(8) – rapport résumé sur les quotas pour un système de fichiers
tune2fs	cf tune2fs(8) – Ajuster les paramètres des systèmes de fichiers ext2...
usrquota,grpquota	n'est pas une commande mais une option dans /etc/fstab qui permet d'activer les quotas sur un périphérique
umount	umount(8) – démonter des systèmes de fichiers

Travaux pratiques

1. créez une nouvelle partition en utilisant **fdisk** sur **/dev/hda**

```
fdisk /dev/hda
```

Conseils : Pour créer une nouvelle partition, tapez **n**. Le type de la partition est par défaut **83** (Linux). Pour écrire la table de partition, tapez **w**. Le système doit relire la table de partition : REDÉMARRER !

2. Formatez une de vos partitions

```
mkfs <périphérique>
```

- i. Créez un répertoire **data** à la racine

```
mkdir /data
```

- ii. Modifiez **/etc/fstab** et reliez le point de montage **/data** à la partition que vous avez précédemment formatée

```
<périphérique> /data ext2 defaults 0 2
```

3. Forcez **mount** à lire **/etc/fstab** :

```
mount -a
```

Si ça ne fonctionne pas, vérifiez le fichier **fstab** et assurez-vous que **/data** existe **2(i)**.

4. Suivez les étapes décrites dans le cours pour activer les quotas sur ce périphérique

À la deuxième étape, tapez **mount** et consultez sa sortie. Quelle option de **/etc/fstab** indique que les quotas sont activés sur le périphérique ?

À la troisième étape, quel fichier est créé dans le répertoire **/data** ? *_ Avant de vérifier les quotas pour les utilisateurs normaux, ajoutez les permissions en lecture/écriture sur /data : **chmod o+rw /data** Dans les cas extrêmes, il pourrait être plus simple de redémarrer et de laisser les scripts de démarrage créer*

Exercices et résumé

le fichier **aquota.user** ou **aquota.group**. Si les commandes **quotas**, **repquota** ou **quotastats** n'indiquent rien, assurez-vous que tout le monde a accès en lecture écriture sur **/data** : **chmod a+rw /data**(FACULTATIF) Monter un partage NFSLe formateur a un partage NFS. Déterminez quel répertoire est partagé et modifiez **/etc/fstab** pour monter ce partage sur **/mnt/nfs**. Utilisez l'option **noauto** pour ne pas monter le partage au démarrage.Changement de chargeur de démarrageDésinstallez LILO du MBR ou de la disquette :**lilo** – adaptez l'exemple de **grub.conf** donné plus haut pour qu'il convienne à votre système.Installez GRUB sur votre disquette avec la commande :**grub-install /dev/fd0**

Réponses aux questions

1. **Non** : il faut relancer **/sbin/lilo** à chaque modification de **/etc/lilo.conf**
2. **Oui** : On n'installe qu'une fois GRUB
3. **Non** : la limite du cylindre 1024 affecte la deuxième partie du chargeur d'amorçage (**NdT** : dans **/boot**)
4. **Oui** : On ne peut pas fixer de quota sur un répertoire

Page consultée fois

Le système de fichiers Linux

État :

Pré-requis

- Avoir déjà installé Linux ([voir Installation](#))

Objectifs

- Introduction aux répertoires de base et aux concepts issus de la FHS (Filesystem Hierarchy Standard) ou norme de la hiérarchie des systèmes de fichiers
- Formater une partition en EXT2 ou EXT3
- Contrôler l'espace libre par périphérique ou répertoire
- Comprendre les droits sur les fichiers et les répertoires dans le monde UNIX

La structure du système de fichiers

Un système de fichiers est similaire à une arborescence. La racine de l'arborescence est au sommet et les feuilles en dessous.

Comme nous l'avons déjà vu, vous devez donner un point de montage aux partitions que vous créez. En général, on le fait au moment de l'installation. Pour mieux comprendre où se trouvent les choses, jetons un œil à l'arborescence typique d'un système de fichiers Linux.

Au sommet de l'arborescence, on trouve la racine (*root*) : /. C'est un peu comme le C:\ sous DOS, sauf que C:\ est également le premier périphérique, alors que la racine peut correspondre à n'importe quel périphérique de votre système.

Répertoires de base

```

                                     / <--- racine
                                     |
bin  dev  etc  lib      |      mnt  proc  sbin
|    |    |    |      |      |    |    |
-----
|    |    |    |    |    |    |    |
boot home root tmp  usr  /usr/local  opt  var

```

Répertoires dont le point de montage peut être un périphérique autre que celui de la racine :

```
boot home root tmp usr /usr/local opt var
```

Les répertoires de base sont les premiers sous-répertoires sous la racine. Sur un système de type RedHat, ils sont installés par un paquet *rpm* du nom de *filesystem* :

```
$ rpm -ql filesystem
```

Au démarrage du système, le noyau monte la partition racine (/). Pour pouvoir monter et vérifier les autres partitions et systèmes de fichiers, un certain nombre de programmes comme *fsck*, *insmod* ou *mount* doivent être accessibles.

Remarque : les répertoires /dev, /bin, /sbin, /etc et /lib doivent être sur la partition racine. De plus, la racine doit contenir un répertoire /proc vide. Il est utilisé par le noyau pour informer sur le statut du système d'exploitation (processus, statistiques d'utilisation de la mémoire, etc.).

Rôle des répertoires de base

- /bin et /sbin : contient les binaires nécessaires au démarrage et les commandes essentielles

Le système de fichiers Linux

- /dev : fichiers périphériques ou fichiers spécifiques
- /etc : fichiers et répertoires de configuration spécifiques à la machine
- /lib : bibliothèques partagées pour les binaires de /bin et /sbin. Contient également les modules du noyau.
- /mnt ou /media : points de montage pour les systèmes de fichiers externes
- /proc : informations du noyau. En lecture seule sauf pour /proc/sys.
- /boot : contient le noyau Linux, le System.map (carte des symboles du noyau) et les chargeurs d'amorçage secondaires.
- /home (facultatif) : répertoires utilisateurs, avec, en général, une copie du contenu de /etc/skel.
- /root : répertoire de l'utilisateur *root*.
- /tmp : fichiers temporaires.
- /usr : *User Specific Resource*. Contenu essentiellement statique et partageable. NdT : /usr est composé de sous-répertoires bin, sbin, lib, slib et autres qui contiennent l'essentiel des programmes et bibliothèques de votre système non essentielles ni nécessaires au démarrage.
- /usr/local ou /opt : programmes et bibliothèques supplémentaires. NdT : en général, c'est dans ces répertoires que l'on place les programmes qui ne font pas partie des paquets des distributions.
- /var : données variables comme les spool ou les journaux. Les sous-répertoires peuvent être soit partageables (comme /var/spool/mail) soit non partageables (comme /var/log).

/var/www, /var/ftp ou /srv : pages web ou fichiers ftp anonymes.

Formatage et cohérence des systèmes de fichier

Il est nécessaire de créer un système de fichiers pour organiser les données sur une partition. La procédure d'installation vous demande quel type de système de fichier vous souhaitez utiliser.

Vous avez un choix important de système de fichiers. Le format **ext2**, aussi connu comme le format natif pour Linux, est utilisé par défaut. Pour les programmes d'installation plus récents, ext3 (NdT et même aujourd'hui ext4 !) est utilisé par défaut. Ext3 est en fait un système ext2 journalisé.

Pour la mémoire paginée (swap), vous n'avez pas le choix, vous devez utiliser le système de fichiers *swap*.

Ext2

Regardons ext2 d'un peu plus près. Ext2 est constitué de blocs de 1024 octets (par défaut). Sans aller trop loin dans les détails, il y a 3 types de blocs : * les superblocs : répétés tous les 8193 blocs, les superblocs contiennent des informations sur la taille des blocs, les inodes libres, le dernier montage, etc. * les inodes : contient des pointeurs sur les blocs de données. Les 12 premiers blocs de données sont accédés directement. Si les données vont au-delà des 12 Ko, alors les inodes indirectes font le relais. * les blocs de données : correspondant aux fichiers ou aux répertoires, ils contiennent les données.

Outils de formatage

Les systèmes de fichiers gérés par le noyau nous permettent de lire les disques déjà formatés. Pour créer des systèmes de fichiers sur un système en fonction, nous devons installer et utiliser les outils correspondants. Pour ext2, le programme de formatage est mkfs.ext2 ou mke2fs. De la même façon, pour le système de fichiers de *Silicon Graphics* xfs, vous utiliserez mkfs.xfs que vous devrez sans doute installer.

Le programme mkfs agit comme une interface pour les outils spécifiques à chaque système de fichier. Sa syntaxe est :

Le système de fichiers Linux

```
mkfs -t <type de système> <périphérique>
```

Vous noterez de nouveau qu'ext3 est un système de fichiers ext2 auquel on a ajouté la journalisation (voir plus loin les exercices).

Exemple 1 : formatage en jfs

```
mkfs -t jfs /dev/hda12
```

Exemple 2 : formatage en ext2fs

```
mke2fs /dev/hda11 [or mkfs -t ext2 /dev/hda11]
```

Cohérence des systèmes de fichiers

Vous devez utiliser **fsck** quand le système de fichiers est endommagé ou corrompu, mais cela nécessite au minimum de démonter la partition ou de la monter en lecture seule.

Comme mkfs, fsck se comporte comme une interface qui détecte automatiquement le type de système de fichiers de la partition. Ensuite, les programmes fsck.ext2, fsck.ext3, etc. sont appelés. Comme EXT2 est le système de fichiers principal de Linux, il existe une commande e2fsck qui ne gère que ce système de fichiers.

Vous pouvez préciser le type de système de fichiers en utilisant la syntaxe suivante :

```
fsck -t <type de système> <périphérique>
```

Exemple : vérification d'un système de fichiers reiserfs sur le périphérique /dev/sdb10 :

```
fsck -t reiserfs /dev/sdb10
```

```
fsck.reiserfs /dev/sdb10
```

Commandes de débogage des systèmes de fichiers

Les commandes debugfs et dumpe2fs sont rarement utilisées mais elles peuvent être utiles pour vous donner des informations de bas niveau sur un système de fichiers ext2 ou ext3.

```
debugfs [ -b tailledebloc ] [ -s superbloc ] [ -f cmd_file ] [ -R request ] [ -V ] [ [ -w ] [ -c ] [ -i ] [ périphérique ] ]
```

Le programme debugfs est un débogueur de système de fichiers interactif. Il peut être utilisé pour examiner ou modifier l'état d'un système de fichiers ext2/3.

Une fois dans le shell de debugfs, vous pouvez utiliser les commandes internes pour parcourir l'arborescence, examiner les données d'inode, supprimer des fichiers, créer des liens, récupérer le journal d'ext3, etc. Cette commande est très puissante et doit être utilisée avec précaution, en général seulement pour essayer d'avancer alors que fsck a échoué.

```
dumpe2fs [ -bfhixV ] [ -ob superbloc ] [ -oB tailledebloc ] <périphérique>
```

dumpe2fs affiche les informations des superblocs et des groupes de blocs pour le système de fichiers sur le périphérique.

```
dumpe2fs /dev/hda1
```

```
dumpe2fs 1.35 (28-Feb-2004)
Filesystem volume name:   /boot1
Last mounted on:         <not available>
Filesystem UUID:         d741042c-3eaf-49ee-94c1-7dd8c5459764
Filesystem magic number: 0xEF53
Filesystem revision #:   1 (dynamic)
Filesystem features:     has_journal ext_attr resize_inode dir_index filetype
needs_recovery sparse_super
Default mount options:   (none)
Filesystem state:        clean
Errors behavior:         Continue
Filesystem OS type:      Linux
Inode count:             25584
Block count:             102280
Reserved block count:    5114
Free blocks:             80564
Free inodes:             25537
```

Le système de fichiers Linux

```

First block:          1
Block size:          1024
Fragment size:       1024
Reserved GDT blocks: 256
Blocks per group:    8192
Fragments per group: 8192
Inodes per group:    1968
Inode blocks per group: 246
Filesystem created:  Sat May  7 10:40:51 2005
Last mount time:     Sun May 29 04:08:01 2005
Last write time:     Sun May 29 04:08:01 2005
Mount count:         10
Maximum mount count: -1
Last checked:        Sat May  7 10:40:51 2005
Check interval:      0 (<none>)
Reserved blocks uid: 0 (user root)
Reserved blocks gid: 0 (group root)
First inode:         11
Inode size:          128
Journal inode:       8
Default directory hash: tea
Directory Hash Seed: 50108791-6a0a-41ff-9608-0485c993eaf9
Journal backup:      inode blocks

```

```

Group 0: (Blocks 1-8192)
  Primary superblock at 1, Group descriptors at 2-2
  Block bitmap at 259 (+258), Inode bitmap at 260 (+259)
  Inode table at 261-506 (+260)
  0 free blocks, 1942 free inodes, 2 directories
  Free blocks:
  Free inodes: 27-1968

```

[....]

Contrôle de l'utilisation des disques

Avec mount et df

Ces deux outils fonctionnent au niveau des périphériques et non pas au niveau des répertoires. Les commandes **mount** et **umount** mettent à jour la liste des systèmes de fichiers montés dans **/etc/mstab**.

mount sans option affiche la liste des systèmes de fichiers actuellement montés, qui est équivalente au contenu de **/etc/mstab**. Le noyau garde également une trace des systèmes de fichiers montés dans **/proc/mounts**.

df affiche également les systèmes de fichiers montés mais il affiche également l'espace disque utilisé et disponible. Par défaut, l'espace est donné en **blocs de 1K**.

```
df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/hda9	289M	254M	20M	93%	/
/dev/hda2	23M	7.5M	14M	35%	/boot
none	62M	0	61M	0%	/dev/shm
/dev/hda5	1.4G	181M	1.1G	13%	/share
/dev/hda7	787M	79M	669M	11%	/tmp
/dev/hda3	4.3G	3.4G	813M	81%	/usr
/dev/hda6	787M	121M	627M	17%	/var
//192.168.123.2/share	12G	8.8G	3.7G	71%	/mnt/smb

Le système de fichiers Linux

Utilisation de du

Cette commande affiche l'utilisation du disque, mais cette fois en se basant sur les répertoires, et non plus les périphériques. Vous noterez que *du* ne peut pas afficher l'espace disponible sur le disque puisque cette information n'est accessible qu'au niveau du périphérique.

La commande suivante affiche la taille utilisée pour /etc en utilisant une unité compréhensible par les humains (-h) et n'affiche que le total (-s : summary).

```
du -sh /etc
62M    /etc/
```

Droits d'accès et attributs des fichiers

Changement des permissions et des propriétaires

Exemple :

```
-rw-rw-r-- 1 utilisateur groupe 13281 2007-03-09 11:00 STATS.ods
  ^      ^      ^
  chmod  chown  chown/chgrp
```

Comme vous pouvez le voir dans l'exemple précédent, **chmod** est utilisé pour modifier les permissions. Il y a trois types de propriétaire pour chaque fichier ou répertoire :

- u : un utilisateur existant dans /etc/passwd
- g : un groupe existant valide de /etc/group
- o : les autres

Exemple :

```
-rw-rw-r-- 1 jade sales 24880 Oct 25 17:28 libcgic.a
```

Modification des permission avec chmod :

```
chmod g=r,o-r libcgic.a
chmod g+w libcgic.a
```

Modification de l'utilisateur et du groupe avec chown et chgrp :

```
chown root libcgic.a
chgrp apache libcgic.a
```

Remarque : Les trois outils **chmod**, **chown** et **chgrp** ont l'option très pratique **-R** qui applique les modifications récursivement sur les répertoires et fichiers indiqués.

Notation symbolique et octale

Les permissions sont soit la lecture read=r, l'écriture write=w et l'exécution execute=x. Voici les valeurs octales correspondant à ces permissions :

Symbolique	Octal	Binaire
r	4	'100'
w	2	'010'
x	1	'001'

Ces permissions s'appliquent à l'utilisateur, au groupe et aux autres. Un élément a un ensemble de 3 permissions groupées pour chacune de ces catégories.

Exemple : Interprétation d'une permission 755 ou -rwxr-xr-x :

- utilisateur : rwx, c'est à dire $4+2+1 = 7$
 - groupe r-x, c'est à dire $4+1 = 5$
 - autres r-x, c'est à dire $4+1 = 5$
-

Le système de fichiers Linux

Les permissions standard

Les systèmes UNIX créent des fichiers et répertoires avec des permissions standard comme suit :

Fichiers | 666

-rw-rw-rw-

Répertoires | 777

-rwxrwxrwx

Umask

Chaque utilisateur peut définir un umask qui altère la permission standard. Pour obtenir les permissions d'un fichier, on soustrait (*) la valeur octale du **umask** à la valeur octale de la permission standard. Cette permission n'a pas de nom mais on pourrait l'appeler la permission effective.

(*) Même si la soustraction fonctionne dans la plupart des cas, vous devriez noter que la formule utilisée pour calculer les permissions est :

Permissions finales = Permissions standard ET(logique) NON umask

Sur les systèmes où les utilisateurs appartiennent à différents groupes, la valeur du umask peut être 002. Pour les systèmes où tous les utilisateurs sont dans le groupe users, le umask a toutes les chances d'être fixé à 022.

Le SUID

On peut assigner une permission spéciale à un programme de façon à ce qu'il soit toujours lancé en tant que le propriétaire du fichier. On appelle cette permission **SUID**, ce qui signifie « set user ID » (définition de l'identifiant utilisateur). La valeur symbolique pour cette permission est **s** et la valeur numérique **4000**.

Les outils d'administration peuvent avoir le **SUID** défini de façon à permettre aux utilisateurs de modifier des fichiers système.

Par exemple, la commande **passwd** peut être lancée par n'importe quel utilisateur pour lui permettre de changer son mot de passe, qui sera écrit soit dans **/etc/passwd**, soit dans **/etc/shadow**. Pourtant, ces deux fichiers appartiennent à l'utilisateur root et leurs permissions respectives sont **644** et **600**.

On résout ce problème en plaçant le bit SUID sur la commande passwd, ce qui la force à être lancée en tant que root et lui permet de modifier les fichiers /etc/passwd et /etc/shadow.

Le SUID sur passwd :

```
ls -l $(which passwd)
-r-s--x--x  1 root    root      18992 Jun  6  2003 /usr/bin/passwd
```

Remarque : dans la commande précédente, le SUID est affiché dans sa forme symbolique. Vous pouvez obtenir plus d'information sur un fichier en utilisant **stat** et en regardant la représentation octale des permissions :

```
stat /usr/bin/passwd
File: `/usr/bin/passwd'
Size: 18992    Blocks: 40    IO Block: 4096   regular file
Device: 305h/773d    Inode: 356680    Links: 1
Access: (4511/-r-s--x--x)  Uid: ( 0/ root)   Gid: ( 0/ root)
```

Attention : La permission SUID pose souvent des problèmes de sécurité. En voici un exemple :

1. Un utilisateur cherche à lire les mails du root. D'abord, il change sa variable d'environnement MAIL :

```
export MAIL=/var/spool/mail/root
```

1. Ensuite, il lance la commande mail :

```
mail
/var/spool/mail/root: Permission denied
```

Ça ne fonctionne pas, ça aurait été trop facile ! Mais si un administrateur s'est laissé convaincre de placer le bit SUID sur la commande mail, les commandes précédentes permettraient à n'importe quel utilisateur de lire les mails de n'importe quel autre (root compris) !

Les exemples suivants sont également dangereux, pourquoi ?

```
chmod 4755 /bin/cat
```

Le système de fichiers Linux

```
chmod u+s /bin/grep
```

Le SGID

Le **SGID** est une permission similaire au SUID mais pour le groupe. Sa valeur symbolique est **s** et sa valeur octale est **2000**.

Si vous placez le SGID sur un répertoire, les fichiers qui y seront ensuite créés appartiendront au groupe propriétaire de ce répertoire. Il n'est donc pas nécessaire d'utiliser **newgrp** pour modifier le groupe du processus utilisé pour la création du fichier (voir l'exercice plus loin).

Exemples :

```
chmod 2755 /home/data  
chmod g+s /bin/wc
```

Le sticky bit

La permission sticky bit avec la valeur **1000** a l'effet suivant :

- appliqué à un répertoire, il empêche les utilisateurs de supprimer les fichiers dont ils ne sont pas propriétaires (idéal pour un répertoire partagé par un groupe) (**NdT** : le sticky bit est typiquement utilisé pour /tmp).
- appliqué à un fichier, il était utilisé pour charger le fichier en mémoire pour en accélérer les futurs accès ou exécutions. La valeur symbolique pour un exécutable est **t**, **T** pour un fichier non exécutable. Aujourd'hui comme le cache des systèmes de fichier est plus courant et plus performant, le sticky bit pour les fichiers tend à ne plus être supporté.

Exemples :

```
chmod 1666 /data/store.txt  
chmod o+t /bin/bash
```

Les attributs sur les fichiers

Au côté des permissions standard, il existe un autre système pour modifier la façon dont on peut accéder à un fichier. Les attributs de fichiers ne s'affichent pas avec la commande **ls** mais avec la commande **lsattr**. On utilise la commande **chattr** pour paramétrer ou supprimer ces attributs.

Les attributs suivants existent. Veuillez noter les cas d'utilisation : * « A » : Quand on accède à un fichier avec l'attribut A, sa valeur *atime* (temps d'accès) n'est pas modifiée. Ceci évite un certain nombre d'accès disques, ce qui est plutôt pratique pour des fichiers temporaires. Attention cependant, certains outils comme *tmpwatch* se basent sur la valeur du *atime* pour déterminer l'utilisation récente d'un fichier, donc si le *atime* n'est pas mis à jour l'état du fichier sera mal interprété.

- « a » : L'accès en écriture aux fichiers avec l'attribut « a » est limité en ajout (append). Seul le super utilisateur ou un processus disposant de la capacité `CAP_LINUX_IMMUTABLE` peut définir ou supprimer cet attribut. L'utilisation la plus évidente est pour les fichiers journaux système, pour empêcher un intrus de retirer les traces de leur passage. Cependant, vous devriez garder à l'esprit qu'un intrus a besoin des droits root pour éditer ces fichiers journaux. Par conséquent, s'il les a, il peut retirer l'attribut « a », modifier les fichiers puis rétablir l'attribut « a ».
- « c » : un fichier avec l'attribut « c » est automatiquement compressé par le noyau. Vous accédez à vos données décompressées en lecture, et l'écriture sur le fichier compresse les données avant de les stocker sur le disque. Note : Même si l'attribut est défini sur un fichier et qu'il est affiché avec la commande **lsattr**, il n'est pas traité par les pilotes du noyau pour Ext2 ou Ext3.
- « D » : les modifications sur un répertoire avec l'attribut « D » sont écrites de façon synchrone sur le disque. C'est l'équivalent de l'option `dirsync` de `mount` mais ici appliquée à un sous-ensemble de fichiers. Avec cet attribut, les opérations suivantes sont synchrones dans le répertoire : `create`, `link`,

Le système de fichiers Linux

unlink, symlink, mkdir, rmdir, mknod et rename.

- « d » : un fichier avec l'attribut « d » ne sera pas sauvegardé quand la commande dump sera lancée.
- « i » : un fichier avec l'attribut « i » (immutable = immuable) ne peut pas être modifié. Il est impossible de le supprimer ou de le renommer, ni de créer un lien vers ce fichier, ni d'écrire dans le fichier. Seul le superutilisateur ou un processus ayant la capacité CAP_LINUX_IMMUTABLE peut définir ou retirer cet attribut.
- 'j' : les données d'un fichier ayant l'attribut « j » sont inscrites dans le journal ext3 avant d'être inscrites dans le fichier si le système de fichiers est monté avec l'option « data=ordered » ou « data=writeback ». Lorsque le système de fichiers est monté avec l'option « data », l'attribut est sans effet puisque les données de tous les fichiers sont écrites dans le journal. Seul le super utilisateur ou un processus ayant la capacité CAP_SYS_RESSOURCE peut définir ou retirer cet attribut.
- « s » : lorsqu'un fichier avec l'attribut « s » est supprimé, ses blocs sont remis à zéro et réinscrits sur le disque. Note : comme pour l'attribut « c », cet attribut n'est pas traité par les pilotes de système de fichiers ext2 et ext3.
- « S » : les données d'un fichier ayant l'attribut « S » sont inscrites sur le disque de façon synchrones. C'est l'équivalent de l'option de « sync » de mount appliquée à un sous-ensemble de fichiers. Cet attribut est généralement utilisé pour les fichiers dits « cooked files », les fichiers dans lesquels les serveurs de bases de données stockent leurs informations. Ainsi, on contourne l'ensemble des deux systèmes de cache des pilotes du noyau et le cache de la base de données, optimisé, écrit directement sur le disque.
- « T » : l'attribut « T » n'est disponible qu'à partir des noyaux 2.6.x. Son but est d'indiquer le sommet de la hiérarchie, fonctionnalité utilisée par l'algorithme « Orlov block allocator ». Les nouvelles règles d'allocation de fichiers sur les systèmes de fichiers ext2 et ext3 rapprochent les sous-répertoires de façon à accélérer les accès sur une arborescence, si l'arborescence a été créée avec un noyau 2.6.
- « t » : le dernier bloc d'un fichier avec l'attribut « t » ne pourra pas être partagé avec d'autres fichiers (pour les systèmes de fichiers qui gèrent le « tail-merging »). C'est nécessaire pour des applications comme LILO qui accèdent directement au système de fichier et qui ne sont pas capables d'interpréter les blocs partagés. Note : au moment de l'écriture de la documentation, ext2 et ext3 ne font pas, en dehors de rustines très expérimentales, de « tail-merging ».
- « u » : lorsqu'on supprime un fichier ayant l'attribut "u", son contenu est sauvegardé. Ainsi, l'utilisateur peut demander sa récupération. Encore un attribut qui est géré par tout sauf le noyau.

Exemple :

```
# lsattr testfile
----- testfile

# chattr +i testfile
# lsattr testfile
----i----- testfile
# rm -f testfile
rm: cannot remove `testfile': Operation not permitted

# chattr -i testfile
# rm -f testfile
# ls testfile
ls: testfile: No such file or directory
```

Résumé et exercices

Questions de révision

Oui ou Non

1. Le répertoire /usr doit toujours se trouver sur la partition racine puisqu'il contient des données essentielles au démarrage
2. Il est possible que certaines opérations ne fonctionnent pas si le répertoire personnel d'un utilisateur se trouve ailleurs que dans /home
3. La création d'un système de fichiers sur une partition détruit toujours toutes les données de cette partition

[Voir les réponses](#)

Glossaire

Terme

Répertoires de base

Bloc de données

Sous-répertoires de la racine essentiels

ext2

ext3

Permissions sur les fichiers

FHS

Inode

Description

Sous-répertoires se trouvant directement sous la racine

Bloc utilisé pour stocker les données sur le système de fichiers. Ils sont référencés par les inodes

Répertoires devant se trouver sur le système de fichiers racine pour le démarrage du système

Second extended file system : système de fichiers par défaut sur Linux (**NdT** : aujourd'hui remplacé par Ext3 puis Ext4)

Third extended filesystem : évolution de ext2 avec les mêmes fonctionnalités plus un système de journalisation

Attributs stockés dans l'inode du fichier pour les droits en lecture, écriture et exécution attribués au propriétaire (u), au groupe (g) ou aux autres (o) ainsi que des permissions plus avancées comme le SUID, le SGID et le "sticky bit"

Filesystem Hierarchy Standard (« norme de la hiérarchie des systèmes de fichiers »). Ce standard consiste en un ensemble de règles et de recommandations pour le placement des fichiers et des répertoires sur les systèmes de la famille UNIX. L'objectif de ces règles est d'améliorer l'interopérabilité des applications, outils d'administration ou de développement, et des scripts, ainsi que d'uniformiser la documentation sur ces systèmes. Voir

Bloc utilisé pour stocker les informations sur les fichiers, répertoires et les liens symboliques. La localisation des données des fichiers, les permissions, l'horodatage et le type de fichier (répertoire, fichier ou

Orlov block allocator

lien symbolique) se trouvent dans l'inode.

Algorithme qui améliore les performances des systèmes de fichiers EXT2/EXT3. Disponible uniquement sur les noyaux 2.6. Cette fonctionnalité a été portée de BSD puis améliorée par Alexander Viro, Andrew Morton, et Ted Ts'o. Son auteur originel est Grigory Orlov. Cette fonctionnalité peut être activée avec la commande `chattr` sous Linux

Superbloc

Le superbloc est lu au montage du système de fichiers. Il contient différentes informations comme la taille des blocs du système de fichiers (1024 par défaut), le nombre d'inodes libres, le nombre de montage et le nombre maximum de montages utilisés pour déterminer si une vérification du système de fichiers doit être lancée

Fichiers

Fichier

/etc/mstab

Description

Fichier utilisé **par mount** pour conserver la trace des périphériques montés

/proc/mounts

Fichier utilisé **par le noyau** pour conserver la trace des périphériques montés

Commandes

Commande

chattr

Description (apropos)

Modifier les attributs des fichiers d'un système de fichiers Linux

chgrp

Changer le groupe propriétaire d'un fichier

chmod

Changer les permissions d'un fichier

chown

Modifier le propriétaire et le groupe d'un fichier

df

Indiquer l'espace occupé par les systèmes de fichiers

du

Évaluer l'espace disque occupé par des fichiers

e2fsck

Vérifier un système de fichiers Linux ext2/ext3/ext4

fsck

Vérifier et réparer un système de fichiers Linux

lsattr

Afficher les attributs des fichiers sur une partition Linux de type ext2

mke2fs

Créer un système de fichiers ext2/ext3/ext4

mkfs

Créer un système de fichiers Linux

mount

Tous les fichiers accessibles sur un système de fichiers

Unix sont rangés sur une arborescence unique, la hiérarchie des fichiers dont la racine est /. Ces fichiers peuvent être répartis sur différents périphériques. La commande `mount` est utilisée pour attacher le système de fichiers d'un périphérique sur l'arborescence. À l'inverse, la commande `umount` détache ce système de fichiers de l'arborescence

`umask`

Outil utilisé pour configurer le masque de création de mode de fichier - voir `help umask`

Travaux pratiques

Systèmes de fichiers

1. Créez deux nouvelles partitions de plus de 50 Mo sur `/dev/hda` (ou `/dev/sda`) en utilisant `fdisk`.
 - Conseils :
 - Pour créer une nouvelle partition, tapez **n**. Le type de partition par défaut est **83** (Linux)
 - Pour écrire la table des partitions, tapez **w**.
 - Le système doit lire votre nouvelle partition : redémarrez votre ordinateur !
2. Formatez la première partition en **ext2** et la seconde en **reiserfs**.
 - Conseil : Le programme `mkfs` est une interface à `mkfs.ext2` ou `mkfs.reiserfs`, etc. Sa syntaxe est :
`mkfs -t <type_de_système> <périphérique>`
3. Créez deux répertoires dans `/mnt` et montez vos deux partitions
 - `mkdir /mnt/ext2`
`mkdir /mnt/reiserfs`
4. Vérifiez l'état de votre système :
 - Utilisez `mount` pour lister les périphériques montés. Les permissions configurés dans `fstab` sont également visibles.
 - Utilisez `df` pour voir le nombre total de blocs utilisés. L'option `-k` converti le nombre de blocs en kilo-octets (taille par défaut pour un bloc sur ext2)
 - Lancez `fsck` sur l'un des deux systèmes de fichiers créés. Le programme `fsck` est une interface à `fsck.ext2`, `fsck.ext3`, `fsck.reiserfs`, etc. Sa syntaxe est :
`fsck <périphérique>`
5. Aller plus loin : passer de **ext2** à **ext3**
 - Passez le périphérique monté sur `/mnt/ext2` en ext3 avec `tune2fs`, qui ajoutera un journal au système de fichiers existant. Assurez-vous d'adapter votre fichier `/etc/fstab` pour prendre en compte cette modification.
`tune2fs -j /dev/hdaN`
À ce stade, le système a ajouté un **journal** à la partition `/dev/hdaN`, qui est devenue une partition **ext3**. Ce procédé est non-destructif et réversible. Si vous montez un système de fichiers **ext3** en **ext2**, le fichier **.journal** est écrasé. Vous pouvez le recréer avec `tune2fs`.

Le système de fichiers Linux

Permissions sur les fichiers

1. Connectez vous en utilisateur normal (pas root). Créez un fichier en utilisant **touch** et vérifiez que les permissions effectives sur ce fichier sont **664**.
2. Changez le **umask** en **027**. Si vous créez un nouveau fichier, quelles seront les permissions effectives sur ce fichier ? Où configure-t-on la valeur du **umask** ? Suivant les systèmes, ce sera dans **/etc/profile** ou **/etc/bashrc**.
3. Créez deux utilisateurs :
useradd util1
useradd util2
Initialisez leurs mots de passe avec **passwd util1** et **passwd util2**
4. Créez le groupe **ventes**
groupadd ventes
5. Ajoutez les utilisateurs au groupe ventes
gpasswd -a util1 ventes
gpasswd -a util2 ventes
6. Créez un répertoire **/news** avec comme groupe propriétaire **ventes** et en lecture/écriture pour ce groupe.
mkdir -m 770 /news ; chown .ventes /news
7. Placez le GID sur le répertoire **/news**
chmod g+s /news
Quelles sont les permissions symboliques (-rwxr-xr-x) sur **/news** ? (Utilisez **ls -ld /news**)
Vérifiez qu'un membre du groupe ventes n'a pas besoin de taper "**newgrp ventes**" pour créer des fichiers avec les bonnes permissions. Les membres du groupe ventes peuvent-ils modifier n'importe quel fichier de ce répertoire ?
8. Ajoutez le **sticky-bit** sur **/news**. Vérifiez que seuls les propriétaires peuvent modifier leurs fichiers dans ce répertoire. Comment se présentent les permissions dans **/news** ?
9. En tant que root, placez le SUID root sur **xeyes**. Connectez-vous en utilisateur normal et vérifiez que le programme est lancé par le root.
chmod u+s `which xeyes`
Connectez-vous en utilisateur normal et lancez **xeyes**, puis :
ps aux | grep xeyes
Le programme devrait être lancé par root.

Réponses aux questions

1. **Non** : aucun programme placé dans **/usr** n'est nécessaire au démarrage du système
2. **Non** : on peut placer le répertoire personnel d'un utilisateur n'importe où sur le système
3. **Oui**