

# Introduction à l'administration *d'un système GNU/Linux*

Ndimby Andriantsoavina

Agence universitaire de la Francophonie

15 juin 2011

- 1 Généralités
- 2 Système de fichiers
- 3 Les commandes de bases
- 4 Gestion des périphériques
- 5 Installation de logiciels
- 6 Administrer son système
- 7 Gestion de processus
- 8 Documentation

## Rôles de l'administrateur

- Créer, modifier, supprimer un utilisateur
- Gérer les fichiers et les disques
- Surveiller l'espace disque
- Organiser les sauvegardes
- Ajouter des périphériques
- Améliorer les performances du système
- Intégrer de nouveaux services
- Veiller à la sécurité du système
- Automatiser les tâches systèmes répétitives
- Suivre les logs du système

## LINUX :

- développé par Linus Torvalds en 1991
- au sens strict c'est le noyau ou kernel linux
- interface entre le matériel et les applications, les droits d'accès
- gère la mémoire, les périphériques

## Le système d'exploitation **GNU/Linux** :

- **Multi-tâches** : execution 'simultanée' de plusieurs programmes
- **Multi-utilisateurs** : chaque utilisateur à un login/mdp
- **libre** : liberté d'exécuter, de copier, de redistribuer, d'étudier, de modifier et d'améliorer
- **multi-processeurs** : plusieurs processeurs
- **multi-plateformes** : PC, SPARC, système embarqué, téléphones mobiles

## Distributions Linux

ensemble cohérent de logiciels assemblé autour du système d'exploitation GNU et du noyau Linux

- Mandriva
- RedHat
- Debian
- Ubuntu
- Fedora

## Pourquoi "*on*" utilise **Debian** ?

- noyau Linux et outils GNU : Debian GNU/Linux
- système de gestion de paquets avec plus de **20000** paquets...  
avec suivi de sécurité
- **stabilité** : plus de 1000 développeurs officiels (et 2000 contributeurs)
- disponible pour **11 architectures**
- Pour plus d'infos <http://www.debian.org/>

## Les versions de Debian GNU/Linux

- **stable** contient la dernière distribution officiellement sortie de Debian (squeeze)
- **testing** contient les paquets qui n'ont pas encore été acceptés dans la distribution stable, mais qui sont en attente de l'être (wheezy)
- **unstable** est celle sur laquelle les activités de développement se déroulent (Sid)



Un peu de lecture avant l'installation...

- Manuel d'installation : <http://www.debian.org/releases/stable/installmanual>
- Installation via le réseau :  
<http://www.debian.org/distrib/netinst>

**TP** : Installation de Débian par le réseau

- 1 Généralités
- 2 Système de fichiers**
- 3 Les commandes de bases
- 4 Gestion des périphériques
- 5 Installation de logiciels
- 6 Administrer son système
- 7 Gestion de processus
- 8 Documentation

## Organisation des partitions sous Linux

- assurer une meilleure résistance du système aux crash disques
- face aux attaques de tout types (ex : *saturation de /home/abdou*)
- failles de sécurité (ex : */tmp accessible en écriture à tous les utilisateurs et programmes* )

L'arborescence système : <http://www.pathname.com/fhs/>

- **/** : répertoire racine doit contenir tout ce qui permet au système de démarrer (1Go)
- **/usr** (*unix system resources*) : ressources non nécessaires au démarrage contenant des programmes, documentation (2Go)
- **/var** : répertoires des données variables (ex : logs, mail, spool, squid) (2go en fonction du serveur)
- **/home** : répertoires des utilisateurs ( si nfs 500 Go)
- **/tmp** : données temporaires (512Mo à 1Go)
- **swap** : 1 Go

L'arborescence système : <http://www.pathname.com/fhs/>

- **/etc** : contient les fichiers de configuration du système
- **/bin**, **/sbin** : contenant les exécutables de base (ex : cp, mv, ls)
- **/lib** : bibliothèques associées
- **/mnt** et **/media** : points de montages
- **/proc** et **/sys** : systèmes de fichiers virtuels
- **/dev** : fichiers spéciaux nommés device correspondants à des périphériques

- 1 Généralités
- 2 Système de fichiers
- 3 Les commandes de bases**
- 4 Gestion des périphériques
- 5 Installation de logiciels
- 6 Administrer son système
- 7 Gestion de processus
- 8 Documentation

**Shell** : interpréteur de commandes, interface utilisateur et système d'exploitation

- accès à la machine : login/motdepasse
- lancement de commandes à partir d'un interpréteur *Shell*
- syntaxe : *prompt# [commande] [option(s)] [arguments]*
- exemple : *ls -l /home*

## Opération sur les répertoires

- **ls** : lister les répertoires (syntax : `ls [-options] [répertoires]`)
- **mkdir** : création de répertoire (syntax : `mkdir [-options] [nom du répertoire]`)
- **rmdir** : suppression de répertoire (syntax : `rmdir [-options] [nom du répertoire]`)
- **pwd** : afficher le répertoire courant
- **cd** : changer de répertoire (syntax : `cd [nom du répertoire]`)
- **du** : espace disque utilisé par le répertoire (syntax : `du [option] [nom du répertoire]`)
- **df** : espace disque occupé (ou libre) sur chaque partition (syntax : `df [-options] [fichier]`)



## Opération sur les fichiers

- **cp** : copier les fichiers ou les répertoires (syntax : `cp [-options] <source> <destination>`)
- **mv** : déplace ou renommer les fichiers ou les répertoires (syntax : `mv [-options] <source> <destination>`)
- **rm** : effacer un fichier (syntax : `rm [-options] [fichier]`)
- **find** : chercher un fichier (syntax : `find [-répertoire de départ] [option] [action]`)
- **which** : Montrer le chemin d'accès à une commande (syntax : `which [nom de pgm]`)
- **less ou more** : afficher à l'écran le contenu d'un fichier text (syntax : `less [-options] <nom de fichier>`)
- **whereis et locate** : permet de retrouver les emplacements des fichiers

**Les filtres** : commandes qui, à partir d'un flux d'entrées donné, effectuent des traitements avant d'afficher un résultat en sortie.

- **cat** : concaténation d'un fichier et afficher sur la sortie standard.
- **cut** : afficher suivant les options certains champs d'un fichier  
*cut -d : -f1 /etc/passwd*
- **grep** : cherche une expression dans un fichier  
*grep false /etc/passwd*  
*grep -v root /etc/passwd*
- **sort** : tri ordre croissant
- **join** : effectue des jointures des lignes de deux fichiers différents dans un même champs.
- **paste** : pour fusionner deux fichiers différents en prenant chaque ligne de chaque fichier pour former une nouvelle ligne

## Redirections

- Entrée standard : descripteur **0**
- Sortie standard : descripteur **1**
- Sortie d'erreur standard : descripteur **2**
- `echo 'Bonjour Dakar' > hello.txt` Sortie standard et sortie d'erreur redirigé vers un fichier
- `tail /etc/passwd > fichiersortie 2>&1`
- `tail /etc/passwd >& fichiersortie 2>1`

## Tubes

- **pipe** permet que la sortie d'une commande devient l'entrée d'une autre  
ex : `sort /etc/passwd | head -6`
- **tee** : recopie la sortie dans un fichier  
et affiche en même temps le résultat sur la sortie standard  
ex : `ls -l /etc | tee fichier | wc -l`
- **xargs** passe en argument d'une commande les flux reçus en entrée  
ex : `cat lefichier | xargs ls`

- 1 Généralités
- 2 Système de fichiers
- 3 Les commandes de bases
- 4 Gestion des périphériques**
- 5 Installation de logiciels
- 6 Administrer son système
- 7 Gestion de processus
- 8 Documentation

## Gestion des périphériques

- Périphériques, disques durs, et partitions gérés sous forme de fichier dans `/dev`
- Chaque périphérique correspond un point de montage
- **`/dev/hda`** : disque dur maître du canal IDE 1
- `/dev/hda1` : Première partition du DD IDE 1
- `/dev/hda2` : Deuxième partition du DD IDE 1
- **`/dev/hdb`** : disque dur esclave du canal IDE1
- `/dev/sda` : disque dur maître SATA1
- **Partitions primaires sont numérotées de 1-4**
- **Partitions logiques sont numérotées à partir de 5-20**

## Les périphériques spéciaux

- **/dev/null** : tout ce qui est envoyé à ce périphérique est détruit
- **/dev/zero** : on peut lire une infinité de zéros depuis ce périphérique
- **/dev/random** : on peut lire des nombre aléatoires depuis ce périphérique

## Utiliser fdisk

- **fdisk** permet de créer, effacer, et changer les partitions sur un disque
- ex : `fdisk /dev/hda`
- **p** : afficher la table de partition
- **n** : crée une nouvelle partition (primaire ou logique)
- spécifier la taille de la partition
- **w** : changer la table de partition



## Créer un système de fichier avec mkfs

- **mkfs** initialise un fichier système sur une nouvelle partition
- ex : `mkfs -t ext3 -c /dev/hda2`
- **t** : applique le système de fichier
- **c** : vérifie les erreurs blocs sur le disque

## Montage et démontage

- **mount** monte un système de fichier et **umount** pour démonter
- ex : `mount /dev/sdb3 /mnt/extra` (monte le périphérique /dev/sdb3 sur le point de montage /mnt/extra)
- ex : `mount -t vfat /dev/hdd1 /mnt/windows` (spécifie le point de montage)
- `man mount`
- ex : `umount /dev/sdb3` ou `umount /mnt/extra` pour démonter le périphérique
- il est impossible de démonter un système de fichier occupé

## Montage automatique

- **/etc/fstab** contient les informations des systèmes de fichiers contenu de l'administrateur
- et qui sont montés au démarrage

device	mount point	type	options	dump	pass-no
/dev/hda3	/home	/ext4	defaults	1	2

## fstab

- **device** : périphériques à monter
- **mount point** : point de montage
- **type** : système de fichier (ext2,vfat,auto,nfs)
- **options** : noauto(non chargé au démarrage),ro(read-only),users
- **dump** : 1 (système de fichier normale) ou 0 (système de fichier removeable)
- **pass-no** : 1 (système de fichier racine), 0 système de fichier non monté au démarrage, 2 pour les autres systèmes de fichier.

## Maintenir un système de fichier

- **df** : affiche le résumé des espaces libre sur la partition
- **du** : affiche les informations de l'espace disque utilisé sur la branche d'un répertoire
- **fsck** est un utilitaire qui permet le contrôle de l'intégrité des partitions
- ex: `fsck /dev/sdb3` de préférence sur un système de fichier non monté

- 1 Généralités
- 2 Système de fichiers
- 3 Les commandes de bases
- 4 Gestion des périphériques
- 5 Installation de logiciels**
- 6 Administrer son système
- 7 Gestion de processus
- 8 Documentation

## Ajout/suppression de logiciels depuis **les sources**

### Compilation du programme

```
tar zxvf logiciel.tar.gz  
cd logiciel  
./configure  
make  
make install
```

### Avantages

- Dernières versions
- Logiciel adapté à la plateforme, plus performant
- Options supplémentaire

## Ajout/suppression de logiciels par **APT : Advanced Package Tool**

- Dépôts de paquets `/etc/apt/sources.list`
- *install* : installation d'un paquet = téléchargement + installation + configuration
- *upgrade* : mise à jour de tous les paquets installés (suivi de sécurité)
- Notion de dépendances entre les paquets
- APT ne fait pas lui-même l'installation et la configuration, il passe par **dpkg**



## aptitude : le centre de commande APT

- **/etc/apt/sources.list** : la configuration des dépôts
- **aptitude update** : lecture des paquets disponibles dans les dépôts
- **aptitude search** *motclef*
- **aptitude show** *paquet*
- **aptitude install** *paquet*
- **aptitude safe-upgrade** et **aptitude full-upgrade**
- et aussi : *dpkg -l*

## Les dépôts conseillés :

- dépôts officiels, depuis le miroir le plus proche :  
deb `http://ftp.fr.debian.org/debian squeeze main`
- dépôt de sécurité (obligatoire!) :  
deb `http://security.debian.org/ squeeze/updates main`
- dépôt updates :  
deb `http://ftp.fr.debian.org/debian/ squeeze-updates main`
- dépôts backports (uniquement en cas de nécessité impérieuse) : `http://www.backports.org`

NB : les autres dépôts ne sont pas toujours dignes de confiance !

## Travaux pratiques

- **vim** : l'éditeur de texte
- **rsync** : synchronisation
- **tar** : archivage

- 1 Généralités
- 2 Système de fichiers
- 3 Les commandes de bases
- 4 Gestion des périphériques
- 5 Installation de logiciels
- 6 Administrer son système**
- 7 Gestion de processus
- 8 Documentation

## Utilisateur et groupe

- tout ce qui utilise une machine linux est un utilisateur
- les utilisateurs ont différents privilèges
- les utilisateurs peuvent appartenir à des groupes
- à chaque utilisateur son répertoire de base (ex : /home/toto)
- seul l'utilisateur a les droits sur son répertoire
- chaque utilisateur est identifié par un UID (User IDentificator)
- su : permet de changer d'utilisateur

## Le Superutilisateur : Root

- Chaque système Linux a un utilisateur root
- root est le 'tout-puissant'
- seul **root** peut créer, supprimer, modifier un utilisateur
- le prompt du root se termine souvent par #
- su - : permet de se connecter en tant que root

## Gestion des droits

- **chown** : change l'appartenance du fichier ou du répertoire à un utilisateur  
ex : `chown [toto] [logfile.txt]`
- **chgrp** : change l'appartenance du fichier ou du répertoire à un groupe  
ex : `chgrp [staff] [document.txt]`
- l'option **-R** permet de changer les droits de manière récurssive  
ex : `chown -R root /usr/local/share/music`  
ex : `chown -R toto.www-docs`  
`/www/intranet/user/toto/`

## Permission sur les fichiers

- Les permissions représentent les actions pouvant être affectées aux fichiers
- Trois types de permissions pour les fichiers
- **r** et **x** sont nécessaires à l'exécution d'un script

Permission	Lettre	Description
Read	r	Permission de lire les données dans le fichier
Write	w	Permission d'écrire, trunquer, et d'écraser les données dans le fichier
Execute	x	Permission d'executer le contenu du fichier comme un programme



## Permission sur les répertoires

- r,w,x ont aussi leur signification sur les répertoires
- Les permissions sur les répertoires sont un petit peu différents

Permission	Lettre	Description
Read	r	Permission d'avoir la liste du répertoire
Write	w	Permission de créer, effacer, renommer les fichiers dans le répertoire
Execute	x	Permission de changer le répertoire, ou de l'utiliser comme répertoire intermédiaire

## Concept de bases

- Un fichier a un **propriétaire** et un **groupe propriétaire**
- Les permissions  $r, w, x$  sont spécifiés pour l'utilisateur, le groupe d'utilisateur est le reste (les autres)

## Examen des permissions

- `ls -l`  
`drwxr-x— 9 toto staff 4096 Oct 12 12 :57 accounts`  
`-rw-rw-r— 1 toto staff 11170 Dec 9 14 :11 report.txt`
- `d` répertoire, `-` fichier texte
- 3 caractères `rwX` pour les permissions de **l'utilisateur** propriétaire
- 3 caractères `rwX` pour les permissions du **groupe** propriétaire
- 3 caractères `rwX` pour les permissions des **autres**

Note: `cp -p` ou `-a` préserve les permissions et l'horodotage

## Changer les permissions des fichiers et des répertoires : chmod

- les permissions peuvent être changées seulement par son **propriétaire** ou par le **superutilisateur**
- `chmod a+x un-programme`
- ajoute **(+)** les permissions d'exécution **(x)** pour tous les utilisateurs **(a)** sur le fichier un-programme

## Spécification des permissions

- les formats de spécifications : **[ugoa] [+=-] [rwx]**
- ajout de permissions : u pour l'utilisateur, g pour le groupe, o pour les autres, a pour tout le monde.
- = applique la permission au fichier
- + ajoute la permission à ceux déjà appliquer
- - retire les permissions
- La dernière lettre indique laquelle des permissions utilisés rwx
- ex : `chmod -R g+rwx,o+rx repertoire-public`

## La méthode octale

- Les permissions peuvent être affecté grâce à des nombres

	<b>Utilisateur</b>	<b>Groupe</b>	<b>Autres</b>
Lecture	400	40	4
Ecriture	200	20	2
Execution	100	10	1

- `chmod 664 *.txt`
- `chmod ug=rw,o=r *.txt`

- 1 Généralités
- 2 Système de fichiers
- 3 Les commandes de bases
- 4 Gestion des périphériques
- 5 Installation de logiciels
- 6 Administrer son système
- 7 Gestion de processus**
- 8 Documentation

## Gestion des applications

- **Application** = ensemble de processus
- Processus :  
fichier exécutable / des bibliothèques partagées (.so)  
un ou plusieurs “threads”
- **Activation** = sh, init, inetd, cron
- **Information** = fichiers de log



## Les processus

- Exécution d'une commande : création d'un processus
- Attribution d'un PID unique
- Différents états (actif, en attente, ...)
- Associés à un compte utilisateur, à un ou plusieurs groupes
- Création d'un processus via la primitive "fork", "exec"
- Référence du père : PPID

## L'environnement

Le fils hérite de son père :

- le répertoire courant
- le umask
- le ulimit
- variables d'environnement
- le répertoire racine (modification via « chroot »)

## Gestion des processus

- ps, top, pstree : lister les processus
- kill, killall : envoyer des signaux a des processus
- strace : lister les appels système
- lsof : lister les fichiers ouverts

## cron

- Exécution périodique de commandes
- Fichier `/etc/crontab`
- Répertoire `/etc/cron.*` (hourly, daily, weekly, montly)

```
17 * * * * root run-parts --report /etc/cron.hourly
```

## cron

- Limitation des utilisateurs via les fichiers  
    /etc/cron.allow  
    /etc/cron.deny  
    (Si non présent : soit root soit tout le monde)
- Commande crontab :  
    crontab fichier soumission d'une table cron  
    crontab -l : visualisation des tables actives  
    crontab -r : arrêt de l'exécution de la table active  
    crontab -e : modification de la table active

## Le syslog

- Démon syslogd
- Reçoit les messages d'erreur émis par le noyau / certains services
- Permet de choisir la destination des log (fichiers, machine distante)
- Différents critères : origine / sévérité
- Fichier de configuration : `etc/syslog.conf`

## /etc/syslog.conf

```
user.notice      -/var/log/user.log
*.               @log.exemple.com
daemon.*         -/var/log/syslog
lpr. !*          /dev/xconsole
*.crit;kern.none *
mail.info        root
*.=debug         /dev/tty8
```

## Les sous systèmes

authpriv	service de sécurité et d'authentification
cron	service cron
daemon	les démons système
kern	noyau
syslog	lui-même
user	processus utilisateurs
lpr, ftp, mail, news, ...	



## Les niveaux de sévérité

emerg	système inutilisable
alert	intervention immédiate
crit	erreur critique
err	erreur
warn	avertissement
notice	événement normal
info	information
debug	test, mise au point
none	ignorer les messages

## Syntaxe des actions

- ajouter le message à un fichier (`/var/log/message`)
- envoyer le message à un serveur syslog (`@log.sn.auf`)
- envoyer le message dans un tube nommé préexistant (`|/dev/xconsole`)
- envoyer le message aux utilisateurs qui sont connecté (`root,roger`)
- écrire sur la console texte (`/dev/tty8`)

- 1 Généralités
- 2 Système de fichiers
- 3 Les commandes de bases
- 4 Gestion des périphériques
- 5 Installation de logiciels
- 6 Administrer son système
- 7 Gestion de processus
- 8 Documentation**

## Sur le système, les pages de man

- le format d'une page de manuel : NAME, SYNOPSIS, DESCRIPTION, OPTIONS, FILES, SEE ALSO, BUGS, ...
- `man man` : les différentes sections. En cas d'ambiguïté, on précise : `man 1 crontab` ou `man 5 crontab`
- `whatis` et `apropos`
- en français : paquet `manpages-fr`
- le système `info`

## Sur le web : à boire et à manger !

- Les grands sites de documentations **validées** : [tldp.org](http://tldp.org) et [traduc.org](http://traduc.org), [linux-france.fr](http://linux-france.fr), [framabook.org](http://framabook.org),
- Les sites «constructeurs» : [www.debian.org](http://www.debian.org), [help.ubuntu.com](http://help.ubuntu.com), [fedoraproject.org](http://fedoraproject.org), [www.redhat.com](http://www.redhat.com)
- Les sites des logiciels : [www.apache.org](http://www.apache.org), [www.php.net](http://www.php.net), ...
- Attention : aux sites *wiki* publics, aux forums (par mail, sur le web ou sur les news), aux blogs et autres sites personnels.

Au secours je suis perdu, aidez moi !

- Comment poser les questions de manière intelligente
- <http://www.linux-france.org/article/these/smart-questions/smart-questions-fr.html>