Spécial Débutant sous Linux

Adrien Anselme

Atelier du Club LinUT du 28 mars 2002

Introduction

Aujourd'hui les distributions (le noyau Linux plus un ensemble de logiciels et d'autres spécificités) de Linux sont de plus en plus faciles à utiliser et à installer. L'un des avantages de ces distributions est d'avoir le choix entre des systèmes très faciles à utiliser et automatisés dont on ne peut controler la configuration à 100% et d'autres très sobres et dépouillés qui nécessitent une plus grande connaissance du système. Dans tous les cas il arrive que les outils fournis ne suffisent pas à résoudre certains problèmes ou à configurer un point précis. Il est donc plus que conseillé de connaître les bases d'un système GNU/Linux pour pouvoir faire face à toutes les situations.

Cet atelier se propose de survoler tous les points importants à connaitre pour mieux comprendre le système et mieux "survivre" en cas de panne.

1 Organisation d'un système Linux

Tout d'abord, la première chose à savoir est que ce qu'on appelle Linux n'est que le "noyau" du système, c'est à dire le programme qui dialoque directement avec le matériel. Autour de ce noyau viennent se greffer d'autres programmes dont le shell qui permet de donner des commandes au système. On peut également utiliser une interface graphique commandée par un programme particulier (la plupart du temps il s'agit de X). Souvent, Linux désigne par abus de langage l'ensemble de tous ces programmes.

Après démarrage du système, la première chose que l'on aperçoit est le login. C'est un programme qui vous invite à vous identifier auprès du système avec un nom et un mot de passe. Celui-ci permet de distinguer les utilisateurs entre eux et même si vous êtes la seule personne à utiliser l'ordinateur, vous devez posséder deux identifications : l'une comme utilisateur commun du système aux possibilités limités, l'autre comme administrateur du système ayant la possibilité de tout effacer s'il le désire (voir le point sur les permissions). Il existe donc un utilisateur particulier : l'administrateur appelé `root`.

Après identification apparait l'invite du shell, c'est à dire la ligne sur laquelle on peut taper toutes les commandes désirées. Ces commandes peuvent être soit des commandes spécifiques au shell soit d'autres programmes. Notez qu'à tout moment vous pouvez obtenir une aide sur une commande en tapant à l'invite du shell : `man nom_de_la_commande`, et, pour certaines commandes uniquement : `info nom_de_la_commande`. De plus, la très grande majorité des commandes affichent un rappel de leur syntaxe et des options disponibles lorsque vous leur associez l'option `--help` (ex : `ls --help` affiche la syntaxe de la commande `ls`).

2 Les dossiers, fichiers, permissions et l'arborescence du système

Un système Linux possède une arborescence de répertoires organisée de façon précise.

La base de toute l'arborescence est le répertoire racine `"/"`, c'est le niveau le plus haut de l'arborescence. Ensuite sur le même niveau se trouvent systématiquement les dossiers :

`/bin`	les programmes principaux indispensables au système
`/boot`	les fichiers de démarrage du système
`/dev`	comme *devices* : fait le lien avec le matériel installé
`/etc`	les fichiers de configuration globaux
`/home`	les fichiers des utilisateurs du système
`/lib`	les librairies partagées
`/mnt`	contient les périphériques montés
`/proc`	les informations du noyau
`/root`	les fichiers de l'administrateur système
`/sbin`	programmes supplémentaires
`/tmp`	fichiers temporaires
`/usr`	contient tous les programmes de l'utilisateur
`/var`	comprend les logs du système

Par convention, on essaie de respecter le role de chaque dossier. Ainsi, la plupart des modifications que vous effectuerez en tant que simple utilisateur se feront au niveau de votre répertoire personnel dans `/home/votre_login`. Ce répertoire personnel est aussi parfois symbolisé par le tilde (`~`). Un raccourci pour `cd /home/votre_login` est donc `cd ~`.

Pour naviquer dans l'arborescence, on dispose des commandes suivantes :

- `cd nom_du_dossier` : pour se rendre dans ce dossier. Notez que la chaine de caratère `..` représente le dossier situé au dessus dans l'arborescence. Par exemple si vous vous trouvez dans `/home` et que vous tapez `cd toto` vous vous retrouvez alors dans `/home/toto` si ce dossier existe. Si ensuite vous tapez `cd ..` vous retournez dans `/home`. De même, `.` désigne le répertoire courant : `cd .` n'aura aucun effet.
- `pwd` : permet de savoir à tout moment dans quel dossier on se trouve
- `mkdir nom_du_répertoire` : créé le répertoire `nom_du_répertoire`
- `rmdir nom_du_répertoire` : supprime ce répertoire (s'il est toujours vide)
- `ls` : liste les fichiers contenus dans le répertoire courant. On peut préciser à ls quel dossier il doit lister.
  On peut également lui préciser des options que l'on place à la suite de la commande avec un tiret `-`. Ainsi `ls -al` listera le contenu du répertoire courant sans masquer les fichiers cachés et donnera les détails sur chaque fichier (taille, propriétaire,...).

Remarque : Attention à la casse ! Linux fait la distinction entre les majuscules et les minuscules. Vous pourrez très bien créer deux fichiers, l'un s'appelant "Toto" et l'autre "toto".

On peut ensuite effectuer plusieurs opérations sur les fichiers et dossiers :

- copier un fichier :
  `cp nom_du_fichier destination`
- déplacer un fichier :
  `mv nom_du_fichier destination`
- copier un dossier avec tout ce qu'il contient :
  `cp -R nom_du_dossier destination`
- déplacer un dossier avec tout ce qu'il contient :
  `mv nom_du_dossier destination`
- lister le contenu d'un fichier : `cat nom_du_fichier`
  Dans ce cas, le contenu du fichier défile en une fois sur l'écran. Si l'on veut faire défiler page par page, il faut combiner `cat` avec `more` ou `less` (si less est installé) de la façon suivante : `cat nom_du_fichier | less` ou `cat nom_du_fichier | more` ou alors directement `more nom_du_fichier` ou `less nom_du_fichier`.

Lorsque vous faites un `ls -l` dans un dossier, vous remarquez que la colonne de gauche contient un ensemble de lettres et de tirets : il s'agit des permissions concernant chaque fichier ou dossier. Par exemple :

```
-rw-r--r-- 1 rene users 324102 Mar 15 2002 Index.tmp
-rwxr-xr-x 1 rene users   4096 Jan 10 2002 documents
```

Le premier caractère symbolise le type de fichier. Lorsque l'on voit `d` c'est qu'il s'agit d'un répertoire. Les 3 suivants indiquent les permissions pour le propriétaire du fichier, les 3 autres indiquent celles du groupe d'utilisateurs et les 3 dernières, celles des autres utilisateurs.

En effet, pour que les gens puissent travailler ensemble, UNIX découpe la population des utilisateurs en 3 catégories : Le propriétaire (souvent celui qui a créé le fichier), le groupe, et les autres. Les `autres` désignent tous ceux qui ont accès au système mais qui ne sont ni le propriétaire lui-même, ni un membre du groupe auquel appartient le fichier.

Les groupes permettent de créer des ensembles d'utilisateurs, comme par exemple une équipe de programmeurs, qui auront un accès commun à des données. Par exemple, un développeur créant du code source peut ne laisser la permission d'écriture que pour lui-même, mais laisser les autres programmeurs lire son travail, et interdire complètement l'accès aux autres utilisateurs. (Les groupes présents sur l'ordinateur sont contenus dans le fichier `/etc/group`, la liste des utilisateurs du système est, elle, stockée dans `/etc/passwd`.)

Pour les fichiers :

- `r` indique que le fichier est lisible pour l'entité (propriétaire,groupe, ou autres).
- `w` donne les droits en écriture.
- `x` donne les droits d'exécution du fichier s'il s'agit d'un fichier exécutable.

Lorsqu'il s'agit d'un répertoire, les choses sont un peu différentes :

- `r` autorise à consulter le contenu du répertoire.
- `w` autorise à ajouter/supprimer des fichiers dans le répertoire.
- `x` autorise à se déplacer dans ce répertoire (commande cd).

En pratique, les droits de lecture et d'exécution des répertoires vont ensemble, attribuez les en même temps pour ne pas avoir de problèmes.

Pour ajouter un utilisateur, utilisez la commande `adduser` en tant que *root*, de même avec `addgroup` pour ajouter un groupe.

Vous allez sûrement être amené un jour ou l'autre à devoir modifier les droits d'accès à un fichier ou d'un répertoire, pour cela, il faut utiliser la commande `chmod`. La commande `chown` quant à elle, permet de changer le propriétaire et le groupe d'un fichier. Notez que sous Linux, seul *root* a le droit de changer le propriétaire et le groupe d'un fichier.

La commande `chmod` s'utilise ainsi :

– chmod +x nom_du_fichier ajoute la permission d'exécution à tous.

On peut préciser à gauche du `+` à qui s'applique le changement (`'u'` pour utilisateur, `'g'` pour groupe, `'o'` pour les autres). Ainsi la commande `chmod ug+rw nom_du_fichier` ajoute les droits d'accès en lecture et écriture à l'utilisateur et au groupe pour le fichier. De façon similaire, il suffit de remplacer "+" par "-" pour supprimer des droits.

3 Le multitache

Linux est un système multitâches. De ce fait il dispose d'un système de processes. Tout programme lancé par l'utilisateur ou de façon automatique se retrouve sous forme de process.

On peut lister les processes avec la commande `ps` tous les processes avec le nom de l'utilisateur les ayant lancés. `ps aux` permet de lister tous les process du système.

Lorsque vous lancez un programme, tapez `CTRL+Z` pour le suspendre ou `CTRL+C` pour l'arrêter. Taper `bg` relance le programme en arriere plan. `fg` le remet en avant plan.

L'un des autres avantages d'un système multitâche comme Linux est la possibilité de lancer plusieurs consoles en-même temps, c'est-à-dire d'avoir plusieurs écrans avec plusieurs shells comme si vous aviez plusieurs ordinateurs en un seul.

Pour naviguer entre les différentes consoles virtuelles, on utilise la combinaison `ALT+[F1 à F5]`. A tout moment, on peut fermer sa session et se déloguer en tappant la commande `exit` ou le raccourci `CTRL+d`.

Note : Lorsque vous êtes dans un environnement graphique, c'est la combinaison `ALT+[F1 à F5]` qu'il faut utiliser pour ouvrir une nouvelle console virtuelle. Pour retourner dans votre environnement graphique, tapez ensuite `ALT+F7`.

4 Survivre avec vi

`vi` est un éditeur de fichiers des plus rudimentaires. C'est souvent la façon la plus rapide de modifier un fichier quant on a un problème technique nous empêchant d'utiliser autre chose. Malgré son austérité, vi est un outil très complet.

Syntaxe : `vi nom_du_fichier`

On se déplace dans le texte avec les curseurs ou les touches `h,j,k,l`.

Vi possède deux modes de fonctionnements : le mode *commande* et le mode *insertion*. Au démarrage, Vi est en mode *commande*, et pour passer en mode *insertion*, il faut presser la touche `i`. Pour repasser en mode *commande*, il faut presser la touche `ESC`.

Quelques commandes utiles : `r` puis une lettre pour remplacer la lettre sous le curseur par celle proposée, `R` pour faire de même avec un texte. `d` détruit le caractère courant, tandis que `dd` détruit toute la ligne. Pour sauvegarder, on utilise la commande `:w`, et `:q` pour quitter. Pour quitter sans sauvegarder, on utilise `:q !`, et pour sauvergarder et quitter `:wq`.

5 Survivre avec Emacs

EMACS est sans conteste l'éditeur le plus complet et le plus multifonctions existant sous Linux. Sa manipulation, bien que peu attrayante au premier abord, s'avère en fait d'une efficacité redoutable. Nous allons uniquement aborder ici les quelques commandes à savoir en cas de besoin.

La lettre C désigne la touche CTRL, la lettre M désigne indifféremment les touches ESC ou ALT.

– Se déplacer dans le fichier : touches flêchées ou C-p, C-n, C-f et C-b.
– Ouvrir un fichier : C-x C-f nom_du_fichier
– Détruire la ligne : C-k C-k
– Sauvegarder : C-x C-s
– Quitter : C-x C-c

6 Monter, démonter un périphérique

Sous linux, il n'existe pas de `a :`, `c :`, tout se trouve dans la même arborescence. Ainsi, lorsque vous voulez accéder à un cdrom ou une disquette, vous devez préalablement le *monter* c'est-à-dire associer le périphérique (situé symboliquement dans `/dev`) à un dossier (habituellement dans `/mnt`).

Exemple de fichier de configuration : *fstab*. fstab est le fichier de configuration concernant

les partitions du système. Il est accessible dans `/etc/fstab` et n'est éditable que par root.

7 Le mode graphique

Sous Linux, le mode graphique n'est qu'une option, il est tout à fait possible de s'en passer !

Néanmoins le système *Xfree* permet d'utiliser les possibilités des cartes graphiques pour utiliser des environnements graphiques.

Au dessus de *Xfree*, on doit utiliser ce que l'on appelle des *Window Managers*, chargés de dessiner le contour des fenêtres, etc... Il existe de nombreux *window managers* : sawmill, window maker, kwm, fvwm, twm.

Au dessus de ces *window managers*, on peut utiliser des *"gestionnaire de bureau"*, comme KDE ou Gnome.

Certains programmes ont été conçus pour fonctionner avec KDE, d'autres pour Gnome, ce qui signifie pas qu'on ne peut pas éxécuter des programmes Gnome en étant sous KDE !

8 Documentations, liens

Pour tout problème concernant l'utilisation d'une commande précise, utilisez le `man`. Ex : `man ls`.

Pour plus d'informations sur un programme complet (gcc, emacs, etc...), on peut utiliser `info`.

Il existe de nombreuses documentations pour Linux, appelés les `HOWTOs`, disponibles sur Internet en de nombreux formats. Ils expliquent la plupart des manipulations courantes (par exemple gravure, impression, réseau, etc...). Ces HOWTOs sont disponibles en français.

D'autre part, il faut se souvenir que tout logiciel sous Linux a une documentation, une recherche avec `http://www.google.fr` donnera surement de bons résultats. Les docs des logiciels sont en général dans `/usr/doc` ou `/usr/share/doc`.

Quelques sites intéressants :

- `http://www.lea-linux.org` : plein de conseils pour les grands débutants sous Linux.
- `http://www.linuxdoc.org` : le site officiel des documentations de Linux.
- `http://www.linux-france.org` : un site général sur Linux
- `http://www.root66.net/linux/HowtoEnFrancais/html/index.html` : les HOWTOs en Français.

Un livre intéressant : *Le système Linux*, de Matt Welsh et Lar Kaufman, disponible en français chez Oreilly.

Et si vous avez des questions, la mailing list du Club LinUT est là ! Plus d'informations sur `http://club-linut.enix.org`.