

Premiers pas avec le Terminal

Terminal et les lignes de commandes vous filent les chocottes ! Et puis vous êtes aussi nombreux à considérer qu'entrer des lignes de commandes est pure bêtise, qu'il s'agit d'un retour à la pré-histoire, que tout ceci est bien inutile quand tout existe sous Linux de manière simple, graphique et visuelle.

Vous n'avez pas tort.

Mais ignorer Terminal et ses facultés, c'est se mettre la tête dans le sable. Car Terminal, s'il n'est pas obligatoire pour bien servir de son Mac, est un outil qui peut s'avérer tout à fait utile, voire indispensable. Mais avant d'aborder ses usages plus ou moins complexes, nous vous invitons à en découvrir le fonctionnement de base.

Pour résumer la situation, Terminal (autrement dit, la Console) est un outil grâce auquel vous pouvez utiliser une kyrielle de logiciels qui sont non seulement invisibles à vos yeux d'humain, mais en plus, dépourvus le plus souvent d'interface graphique, c'est à dire de fenêtres et de panneaux permettant de cliquer ici ou là.

• Où se cachent les fichiers exécutables que Terminal sait lancer ?

En plusieurs endroits du poste. Le principal, est le répertoire `/bin`

Pour vous en convaincre, lancez Terminal et dans la fenêtre qui s'est ouverte, entrez la ligne de commande :

`ls /bin`

(par convention, le gros point noir `•` devra être remplacé par un espace)

Une fois cette commande validée (par la touche Entrée), vous constatez que plusieurs lignes se sont affichées.

Vous venez de demander au Terminal d'afficher la liste (`ls`) des éléments présents dans le répertoire `bin` qui se trouve à la racine du disque dur, c'est à dire au premier niveau du disque.

Les éléments indiqués sont des petits logiciels. Par exemple, vous notez la présence de l'élément «`date`».

Entrez la commande :

`date`

et s'affichent en français : le jour, la date, le mois, l'année, l'heure.

A présent si vous écrivez :

`date •>>+Aujourd'hui nous sommes le %d %m %y`

vous constaterez que vous pouvez écrire des choses intelligibles.

Jusqu'ici, avouez que ça n'est pas très compliqué. Outre le dossier `/bin`, il existe des exécutables un peu partout, dans le répertoire `/usr/bin`, dans le répertoire `/usr/local/bin`, etc.

Les principales commandes s'installent dans le répertoire `/usr/bin`.

Si vous tapez :

`ls •/usr/bin`

vous constaterez que la liste peut être longue. Et ingérable.

Pour qu'elle le devienne, tapez plutôt :

`ls •/usr/bin •>>•liste.txt`

Rien ne se passe dans la fenêtre de Terminal, et pourtant, vous venez de créer en un rien de temps, un fichier texte dans votre répertoire personnel. Ouvrez-le donc pour constater que nous disons vrai. Cet exploit a simplement été rendu possible grâce au signe d'échappement «`>>`» qui signifie de diriger le résultat de la commande «`ls`» vers le fichier «`liste.txt`». S'il avait déjà existé, tout son contenu aurait été remplacé par le nouveau résultat.

Si après cette commande vous tapez :

`gedit •liste.txt`

le fichier créé s'ouvre avec le traitement de texte Gedit. Si à présent vous souhaitez compter le nombre de mots que contient un fichier texte, entrez la commande (`-w` pour les mots, `-l` pour les lignes, `-c` pour les caractères):

`wc •w •liste.txt`

si vous entrez :

`wc •lw •liste.txt`

vous obtiendrez les trois données d'un coup.

(Notez que plutôt que d'indiquer le chemin complet de l'endroit où se trouve votre fichier, il vous suffit de le glisser sur la fenêtre de Terminal à la suite de la commande en veillant à ménager l'espace)

Par ces quelques exemples, vous constatez qu'il n'est pas très compliqué de prendre en main le Terminal. Son usage n'est pas dangereux tant que l'on n'utilise pas les droits de l'administrateur invoqués par la commande : `sudo`.

En adoptant le Terminal vous pourrez aborder des champs inexplorés et sauvages du monde Unix et vous sentir maître de votre machine.

```

opensource@ubuntu: ~
Fichier  Édition  Affichage  Terminal  Onglets  Aide
opensource@ubuntu:~$ ls /bin
arch      chown    grep     mknod    rbash    umount
bash      cp       gunzip   mktemp   readlink uname
bunzip2   cpio    gzexe    more     rm       uncompress
bzcat     dash    gzip     mount    rmdir    vdir
bzcmp     date    hostname mountpoint rnano    which
bzdiff    dd      ip       mt       run-parts zcat
bzegrep   df      kill     mt-gnu   sed      zcmp
bzexe     dir     ld_static mv        setpci   zdiff
bzfgrep   dmesg   ln       nano     sh       zegrep
bzgrep    dnsdomainname loadkeys nc        sleep    zfgrep
bzzip2    echo    login    netcat   stty     zforce
bzless    bzip2recover egrep    false   ls       zgrep
bzmore    cat     fgconsole lsmodule ping     zless
cat       chgrp   fgrep    lspci   ps       zmore
chmod     fuser  fuser    mkdir   pwd      znew

```

La maîtrise du Terminal est indispensable pour bien contrôler sa machine.