

# GNU/Linux : commandes Bash essentielles

**Rappel :** le *shell* est l'interpréteur de commandes. Sous GNU/Linux, le plus courant est **bash**. Le *prompt* se termine par `$` pour un utilisateur standard, `#` pour `root`.

## 1. Navigation et exploration

Commande	Effet
<code>pwd</code>	Affiche le répertoire courant ( <i>print working directory</i> )
<code>ls</code>	Liste les fichiers du répertoire courant
<code>ls -lh</code>	Liste détaillée avec tailles lisibles
<code>ls -a</code>	Affiche aussi les fichiers cachés (noms commençant par <code>.</code> )
<code>cd DIR</code>	Change de répertoire vers <code>DIR</code>
<code>cd ~</code>	Va au répertoire personnel ( <code>/home/utilisateur</code> )
<code>cd ..</code>	Remonte au répertoire parent
<code>man CMD</code>	Affiche le manuel de la commande <code>CMD</code> (quitter avec <code>Q</code> )

## 2. Gestion des fichiers et répertoires

Commande	Effet
<code>mkdir DIR</code>	Crée le répertoire <code>DIR</code>
<code>mkdir -p A/B/C</code>	Crée les répertoires intermédiaires si nécessaire
<code>rmdir DIR</code>	Supprime <code>DIR</code> s'il est <i>vide</i>
<code>cp src dst</code>	Copie <code>src</code> vers <code>dst</code>
<code>mv src dst</code>	Déplace (ou renomme) <code>src</code> en <code>dst</code>
<code>rm fichier</code>	Supprime <code>fichier</code>
<code>rm -r DIR</code>	Supprime <code>DIR</code> et tout son contenu (⚠️ <b>dangereux!</b> )
<code>touch fichier</code>	Crée un fichier vide (ou met à jour sa date)
<code>cat fichier</code>	Affiche le contenu de <code>fichier</code>

## 3. Droits et permissions

Sous GNU/Linux, **tout est fichier**. Les périphériques eux-mêmes sont représentés comme des fichiers, accessibles depuis `/dev`. Ainsi, le disque dur est `/dev/sda` ou `/dev/nvme0n1` ; le terminal est `/dev/tty`, etc. Cette philosophie unifie l'accès à toutes les ressources du système sous une même interface.

Un **répertoire** est aussi un fichier, un peu particulier.

Pour chaque fichier (ou répertoire) existent des droits, qui concernent trois publics différents :

- l'utilisateur ;
- le groupe associé au fichier ;
- les autres.

## Exemple

```
drwxrwxr-x 3 eleve users 4096 juin  1 18:37 .
drwxrwxr-x 6 root  root  4096 avril 5 18:41 ..
drwxr-xr-- 3 eleve users 4096 oct.  15 09:07 Documents
-rw-r--r-- 1 eleve users 1024 nov.  30 16:30 fichier1.txt
```

On peut voir, de gauche à droite dans l'exemple ci-dessus :

- les droits :
  - 1<sup>er</sup> caractère : `-` pour un fichier, `d` pour un répertoire, `l` pour un lien symbolique (un fichier qui renvoie vers un autre fichier, situé ailleurs) ;
  - dans les blocs suivants, `r` signifie « *read* » (on peut lire), `w` signifie « *write* » (on peut écrire), et `x` signifie « *execute* » (on peut exécuter le fichier, ou *entrer dans* le répertoire) ;
  - on peut voir, dans les blocs suivants pour le répertoire `Documents` :
    - `rw` pour le propriétaire (`u`) ;
    - `r-x` pour le groupe (`g`) ;
    - `r--` pour les autres (`o`) ;
- propriétaire et groupe principal :
  - le dossier courant (symbolisé par `.`) appartient à l'utilisateur `eleve`, son groupe est `users` ;
  - le dossier *parent* (symbolisé par `..`) appartient à l'utilisateur `root` (le super-utilisateur du système), son groupe est `root` ;
- la taille des fichiers, la date et l'heure de création ou de modification ;
- enfin, le nom du fichier.

On peut agir sur les droits des fichiers avec les commandes suivantes.

Commande	Effet
<code>chmod u+x fichier</code>	Donne le droit d'exécution au propriétaire
<code>chmod g-w fichier</code>	Retire le droit d'écriture au groupe
<code>chmod o+r fichier</code>	Donne le droit de lecture aux autres
<code>sudo CMD</code>	Exécute <code>CMD</code> avec les droits administrateur ( <i>root</i> )

## 4. Processus

Commande	Effet
<code>ps</code>	Liste les processus du terminal courant
<code>ps -fau</code>	Liste détaillée de tous les processus
<code>top</code>	Vue dynamique des processus (quitter avec <code>Q</code> )
<code>pstree</code>	Affiche l'arborescence des processus
<code>kill PID</code>	Envoie un signal d'arrêt au processus <code>PID</code>
<code>kill -9 PID</code>	Arrêt forcé du processus <code>PID</code>

### Rappels importants

- Les chemins **absolus** commencent par `/` ; les chemins **relatifs** partent du répertoire courant.
- `.` = répertoire courant ; `..` = répertoire parent ; `~` = répertoire personnel.
- `rm -r` est irréversible : pas de « corbeille » en ligne de commande !