

Les commandes de bases Linux

1. Commande **ls**: cette commande liste les fichiers et répertoires dans le répertoire courant
Ici on a deux dossiers : **elzo** et **etudiant**

Options les plus fréquentes :

-l : Permet un affichage détaillé du répertoire (permissions d'accès, le nombre de liens physiques, le nom du propriétaire et du groupe, la taille en octets, et l'horodatage)

-h : Associé avec **-l** affiche la taille des fichiers avec un suffixe correspondant à l'unité (K, M, G)

-a : Permet l'affichage des fichiers et répertoires cachés (ceux qui commencent par un **.** (point))

-lct : Permet de trier les fichiers et répertoires par date de modification décroissante

```
elzo@EG-HP:/home$ ls
elzo  etudiant
```

2. Commande **cd**: (change direction)

Exemple: **étudiant** (cette commande vous place dans le répertoire **document**)

cd etudiant permet de se placer dans le répertoire **etudiant**

Exemples d'utilisation :

cd

permet de revenir au répertoire **/home/utilisateur** (identique à **cd ~**)

cd -

permet de revenir au répertoire précédent

cd ..

permet de remonter au répertoire parent (ne pas oublier l'espace contrairement à windows)

cd /

permet de remonter à la racine de l'ensemble du système de fichiers

cd /usr/bin/ ou **usr/bin**

se place dans le répertoire **/usr/bin/**

```
elzo@EG-HP:/home$ cd etudiant
elzo@EG-HP:/home/etudiant$
```

3. Commande **mkdir**: crée un nouveau répertoire(nouveau dossier)

Exemples d'utilisation :

- **mkdir photos**
Crée le répertoire *photos*
- **mkdir -p photos/2005/noel**
Crée le répertoire *noel* et s'ils n'existent pas les répertoires *2005* et *photos*

```
elzo@EG-HP:/home/etudiant$ mkdir  
nouveauDossier  
elzo@EG-HP:/home/etudiant$ ls  
nouveauDossier  
elzo@EG-HP:/home/etudiant$
```

4. Commande **touch**: crée un nouveau fichier vide

`touch nouveuaFichier.txt` crée un nouveau fichier nommé `nouveuaFichier.txt`

Exemple d'utilisation

- a, modifiez uniquement l'heure d'accès
- c, si le fichier n'existe pas, ne le créez pas
- d, mettre à jour les horaires d'accès et de modification
- m, change l'heure de modification uniquement
- r, utiliser les horaires d'accès et de modification du fichier
- t, crée un fichier en utilisant une heure spécifiée

```
elzo@EG-HP:/home/etudiant$ touch nouveauFichier.txt  
elzo@EG-HP:/home/etudiant$ ls  
nouveauDossier  nouveauFichier.txt  
elzo@EG-HP:/home/etudiant$
```

5. Commande **cp**: copie des fichier ou répertoirs

La commande `cp nouveauFichier.txt doss1` copie le fichier `nouveauFichier.txt` dans le dossier `doss1`

Options les plus fréquentes :

- a : Archive. Copie en gardant les droits, dates, propriétaires, groupes, etc.
- i : Demande une confirmation avant d'écraser

-f : Si le fichier de destination existe et ne peut être ouvert alors le détruire et essayer à nouveau

-R ou **-r** : Copie un répertoire et tout son contenu, y compris les éventuels sous-répertoires

-u : Ne copie que les fichiers plus récents ou qui n'existent pas

-v : permet de suivre les copies réalisées en temps réel

- **Exemples d'utilisation :**

cp monFichier sousrep/

Copie *monFichier* dans *sousrep*

cp -r monRep/ ailleurs/

Copie le répertoire *monRep* (et ses éventuels sous-répertoires) vers *ailleurs* en créant le répertoire *ailleurs/monRep* s'il n'existe pas.

cp monRep/{*.cpp,*.h,MakeFile,Session.vim} ailleurs/

Copie les fichiers spécifiés dans {} contenus dans le répertoire *monRep* vers *ailleurs*. Notez bien qu'il n'y a pas d'espace entre ces noms de fichiers.

```
elzo@EG-HP:/home/etudiant$ ls
doss1  nouveauDossier  nouveauFichier.txt
elzo@EG-HP:/home/etudiant$ cp nouveauFichier.txt doss1/
elzo@EG-HP:/home/etudiant$ cd doss1
elzo@EG-HP:/home/etudiant/doss1$ ls
nouveauFichier.txt
elzo@EG-HP:/home/etudiant/doss1$
elzo@EG-HP:/home/etudiant/doss1$
```

Commande **mv**: déplace des fichiers ou répertoires

Dans le dossier étudiant on a deux dossiers **doss1** et **nouveauDossier** et fichier **nouveauFichier.txt**

Le fichier **nouveauFichier.txt** est déplacé dans le dossier **nouveauDossier**.

Options les plus fréquentes :

- **-f** : Écrase les fichiers de destination sans confirmation

- **-i** : Demande confirmation avant d'écraser

- **-u** : N'écrase pas le fichier de destination si celui-ci est plus récent

- **Exemples d'utilisation :**

mv monFichier unRep/

Déplace *monFichier* dans le répertoire *unRep*

mv unRep/monFichier .

Déplace le fichier *monFichier* du répertoire *unRep* là où on se trouve

mv unRep monRep

Renomme *unRep* en *monRep*

```
-HP:/home/etudiant$ ls
nouveauDossier  nouveauFichier.txt
HP:/home/etudiant$ mv nouveauFichier.txt nouveauDossier
HP:/home/etudiant$ ls
nouveauDossier
HP:/home/etudiant$ cd nouveauDossier
HP:/home/etudiant/nouveauDossier$ ls
Fichier.txt
HP:/home/etudiant/nouveauDossier$
HP:/home/etudiant/nouveauDossier$
```

7. Commande **rm**; supprime des fichiers ou des répertoires

rm nouveauFichier.txt supprime le fichier **nouveauFichier.txt**

Options les plus fréquentes :

-i : Demande confirmation avant d'effacer

-f : Ne demande pas de confirmation avant d'effacer

-r : Efface *récurivement*. Ce mot signifie "y compris ses sous-répertoires et leur contenu".

- **Exemples d'utilisation :**

rm CeFichier

Efface du répertoire courant le fichier *CeFichier*.

rm -rf /tmp/LeRep

Efface le répertoire */tmp/LeRep* ainsi que tous ses fichiers, liens et sous-répertoires sans demander de confirmation.

rm -rf /*

...La commande qui "tue"... Disparition immédiate de tous vos fichiers.

```
HP:/home/etudiant/nouveauDossier$ ls
Fichier.txt
```

```
P:/home/etudiant/nouveauDossier$ rm nouveauFichier.txt
P:/home/etudiant/nouveauDossier$ ls
P:/home/etudiant/nouveauDossier$
P:/home/etudiant/nouveauDossier$
```

8. Comma,de **cat** : affiche le contenu

Cat nouveauFichier.txt affiche le contenu du fichier **nouveauFichier.txt**

Options les plus fréquentes :

-n : Affiche les numéros de ligne

-v : Affiche les caractères de contrôles

- **Exemple d'utilisation :**

cat -n monFichier

```
HP:/home/etudiant/doss1$ ls
chier.txt
HP:/home/etudiant/doss1$ cat nouveauFichier.txt
je suis novice au système Unix,
utilisateur :
```

9. Commande **grep** : recherche des mots dans un fichier

Par exemple dans cet cas de figure on recherche le Unix dans le fichier **nouveauFichier.txt**

Options les plus fréquentes :

- -c : Retourne le nombre de lignes au lieu des lignes elles-mêmes
- -n : Retourne les lignes préfixées par leur numéro
- -i : Insensible à la casse
- -r : Recherche récursivement dans tous les sous-répertoires ; on peut utiliser la commande rgrep
- -G : Recherche en utilisant une expression rationnelle basique (option par défaut)
- -E : Recherche en utilisant une expression rationnelle étendue ; on peut utiliser la commande egrep
- -F : Recherche en utilisant une chaîne fixe ; on peut utiliser la commande fgrep
- -v toto : Recherche les lignes qui ne contiennent pas le mot *toto*

- **Exemples d'utilisation :**

- **grep -n montexte monfichier**

Retourne toutes les lignes ainsi que leur numéro où *montexte* apparaît dans *monfichier*

```

elzo@EG-HP: /home/etudiant/doss$ grep "Unix" nouveauFichier.txt
Bonjour je suis novice au système Unix.
Multi-utilisateur : Unix est conçu pour être utilisé par plusieurs utilisateurs simultanément. Chaque utilisateur a un compte distinct et des autorisations spécifiques pour accéder aux fichiers et aux res
sources du système.
Interface en ligne de commande : L'interface utilisateur principale d'Unix est en ligne de commande (CLI - Command Line Interface). Les utilisateurs interagissent avec le système en saisissant des command
es textuelles.
Système de fichiers hiérarchique : Les fichiers sur un système Unix sont organisés dans une structure de répertoires (dossiers) hiérarchique. Chaque fichier est situé à un emplacement spécifique dans cette
structure.
Redirection et tubes (pipes) : Unix offre des fonctionnalités puissantes de redirection de flux de données. Les utilisateurs peuvent connecter la sortie d'une commande à l'entrée d'une autre, permettant ainsi des opérations complexes en combinant plusieurs commandes.
Communication interprocessus : Les processus sur un système Unix peuvent communiquer entre eux en utilisant des mécanismes tels que les tubes (pipes) et les signaux.
Stabilité et robustesse : Unix est réputé pour sa stabilité et sa robustesse. Il est souvent utilisé dans des environnements critiques tels que les serveurs, les systèmes embarqués et les supercalculateurs.
Portabilité : La conception de Unix favorise la portabilité. De nombreuses variantes d'Unix existent, adaptées à diverses plates-formes matérielles.
elzo@EG-HP: /home/etudiant/doss$

```

10 commande **chmod** Modifie les permissions d'accès à un fichier ou à un répertoire.

Options les plus fréquentes :

- **-R** : Récursif, modifie les autorisations d'un répertoire et tout ce qu'il contient
- **-c** : Ne montrer que les fichiers ayant été réellement modifiés
- **-f** : Ne pas afficher les messages d'erreur

11 Commande **chown** change le propriétaire et le groupe propriétaire d'un fichier

● **Options les plus fréquentes :**

- **-R** : Modifie récursivement un répertoire et tout ce qu'il contient

12 Commande **sudo** : Permet d'exécuter des commandes en tant qu'un autre utilisateur, donc avec d'autres privilèges que les siens.

● **Options les plus fréquentes :**

- **-s** : Importe les variables d'environnement du shell
- **-k** : Lorsque l'on utilise **sudo**, il garde en mémoire le mot de passe ; cette