

T.P. 1

## Familiarisation avec Linux

L'objectif de ce premier TP est de vous familiariser avec un système d'exploitation de type UNIX : Linux. Bien que des différences minimales peuvent apparaître, les opérations réalisées dans ce TP peuvent être appliquées sur d'autres UNIX comme Solaris ou d'autres distributions Linux.

Vous lirez les descriptions et vous effectuerez les manipulations au fur et à mesure.

Une feuille complémentaire vous sera distribuée pour approfondir vos connaissances sous UNIX

### 1 Connexion et déconnexion

Afin de restreindre l'accès d'une machine UNIX aux seuls utilisateurs autorisés, la connexion nécessite de disposer d'un nom d'utilisateur (*login*) et d'un mot de passe associé (on parle couramment de compte utilisateur). Ces informations permettent de s'authentifier et d'accéder à son répertoire personnel.

Votre **login** est unique. Votre **mot de passe** est personnel et ne doit être communiqué à personne.

**Vous êtes responsable de opérations effectuées avec votre login.**

Sur les machines que vous utiliserez plusieurs modes de connexion sont disponibles : texte et graphique.

**Connexion** Dans les deux modes, vous devez indiquer votre login et votre mot de passe. Vous ouvrez alors une session sur la machine (plusieurs sessions peuvent être ouvertes sur la même machine).

Par exemple :

```
login : Votre Nom de Connexion Enter  
Password : Votre Mot de Passe Enter
```

**Procédure de changement de mot de passe** Pour la sécurité du réseau et de vos données, votre mot de passe doit être changé régulièrement.

Celui-ci doit contenir entre 6 et 8 caractères alphanumériques (lettres, chiffres et ponctuation), et ne doit apparaître dans aucun dictionnaire (quelle que soit la langue).

Pour changer votre mot de passe, vous utiliserez la commande `yppasswd`. Votre ancien mot de passe vous sera demandé, puis vous devrez entrer votre mot de passe et le confirmer.

**Spécificité de l'Institut Galilée** Les machines des salles de TPs sont pour la plupart (salles G207, G208, G209, G210, G211, G212, G215, F204), en double boot Windows/Linux. Vous pouvez donc démarrer chaque machine soit sous Windows, soit sous Linux.

Vous travaillerez en TP sous Linux.

La connexion s'effectue en mode graphique. Il est nécessaire de choisir le gestionnaire de fenêtre : KDE, Gnome, ICEWM, etc.

Votre login correspond à votre numéro de carte d'étudiant. Lorsque vous êtes inscrit, votre mot de passe est le numéro INE se trouvant sur votre carte d'étudiant. **Vous devez impérativement le changer lors de la première connexion, en suivant les règles définies ci-dessus.**

N.B. : Si votre mot de passe n'est pas changé dans les quinze jours qui suivent le premier TP, votre compte sera bloqué. Le déblocage des comptes n'est effectué que le lundi et le mercredi matin. La procédure de déblocage du mot de passe en cas de perte est la même.

En cas de blocage de votre compte (perte du mot de passe, etc.), vous devez déposer une photocopie de votre carte d'étudiant dans la boîte aux lettres se trouvant devant le bureau F208, avec mention du système d'exploitation (Windows ou UNIX). Les administrateurs débloqueront votre compte en indiquant comme mot de passe votre numéro INE, le mercredi ou le lundi suivant.

—> **Manipulation 1** : *Connectez vous en mode graphique sur votre machine et changez votre mot de passe*

**Déconnexion** Lorsque vous avez terminé de travailler sur votre machine, vous devez vous déconnecter (dans le cas contraire, n'importe qui peut utiliser votre login et accéder à vos données).

Les procédures de déconnexion varient un peu selon le mode de connexion.

**En mode texte** :

Votre Nom@Nom de la Machine :1 :> logout

ou

Votre Nom@Nom de la Machine :1 :> exit

ou

Votre Nom@Nom de la Machine :1 :> CTRL-D

**En mode graphique** La procédure de déconnexion peut varier d'un gestionnaire de fenêtres à un autre. Dans les menus, demandez la déconnexion, et confirmez la demande.

—> **Manipulation 2** : *Déconnectez vous de la machine. Puis reconnectez vous pour la suite du TP.*

## 2 Arrêt de la machine

Sauf en cas d'extrême nécessité, une machine ne doit **jamais** être arrêtée brutalement en appuyant sur l'interrupteur ou en la débranchant (cela peut endommager le système de fichiers et le système d'exploitation). Il faut utiliser la procédure d'arrêt suivante.

**ATTENTION : Vous ne devez pas éteindre les machines des salles de TP. Les explications ci-dessous ne sont fournies qu'à titre indicatif.**

- Après s'être déconnecté, dans un menu, cliquer sur **shutdown** ou **arrêter**.
- Pour redémarrer la machine, cliquez sur **reboot** ou **redémarrer**. De cette manière, vous pouvez redémarrer votre machine sous Windows XP ou sous Linux.

### 3 Utilisation de l'environnement graphique

Plusieurs environnements graphiques sont à votre disposition. L'agencement des menus et des icônes peuvent varier de l'un à l'autre.

L'environnement graphique vous permet d'accéder, grâce à la souris et au travers du menu (en bas à gauche), à un certain nombre d'application (Terminal, Editeur de texte, Navigateur, Lecteur de messagerie, etc.). Lorsque que vous lancez une application, celle-ci apparaît dans le bureau.

→ **Manipulation 3** : *Identifier les icônes correspondant aux applications suivantes (que ce soit dans les menus, sur le bureau, ou dans la barre de tâche) :*

- Terminal
- Lecteur de disquette
- Cle USB
- Editeur de texte
- Navigateur Internet (Mozilla-Firefox)

Si les icônes ne sont pas disponibles, vous devez utiliser les commandes UNIX dans un terminal (voir ci-dessous).

### 4 Terminal

Le terminal permet d'interagir avec le système d'exploitation à l'aide de commandes (voir ci-dessous). Vous pouvez également taper les noms des applications que vous voulez utiliser (par exemple, `mozilla-firefox`, `kwrite`, etc.)

Un mécanisme de complétion existe dans le terminal que vous utilisez. Ainsi, dans la plupart des cas, vous n'avez qu'à taper les premiers caractères d'une commande et à utiliser la touche de tabulation pour compléter automatiquement le nom de la commande. En cas d'ambigüité, la liste des commandes potentielles sera affichée, à vous de taper quelques caractères supplémentaires.

Pour fermer un terminal, taper la commande `exit` ou cliquez sur la croix en haut à droite.

→ **Manipulation 4** : *Lancez un terminal puis lancer mozilla-firefox.*

### 5 Editeur de texte

Plusieurs éditeurs de texte sont à votre disposition pour écrire des programmes ou des documents : `kwrite`, `nedit`, `emacs`, `kate`, etc.

→ **Manipulation 5** : *Dans un terminal, lancez l'éditeur de texte kwrite.*

### 6 Compilation

De nombreux compilateurs sont disponibles sous UNIX. Le compilateur C correspond à la commande `gcc` et doit être utilisé dans un terminal..

L'utilisation la plus courante est la suivante :

```
gcc NomduFichierSource -o NomDuFichierExécutable
```

L'exécution du programme s'effectue en tapant le nom de l'exécutable (ici `NomDuFichierExécutable`) dans un terminal.

—> **Manipulation 6** : Dans l'éditeur que vous avez ouvert lors de la manipulation précédente, écrivez un programme en C, affichant le message "bonjour tout le monde". Dans un terminal compilez votre programme puis exécutez le

## 7 Impression

L'impression de fichier s'effectue sous Linux à l'aide de la commande `lpr`. Sous d'autres UNIX, il peut s'agir de la commande `lp`.

Le nom de l'imprimante est composé de la manière suivante :

`nom_de_la_salle-imp`

Par exemple : `G207-imp`

pour l'année, vous disposez d'un crédit de 100 feuilles.

## 8 Utilisation d'une clé USB

Sous la plupart des machines de TP (notamment celles sous Linux Mandrake 10.2), les clés USB sont reconnues et montées automatiquement. Le répertoire permettant d'accéder aux données se trouvant sur la clé est `/mnt/removable` dans la plupart des cas.

—> **Manipulation 7** : si vous disposez d'une clé USB, vérifiez que vous pouvez accéder aux données se trouvant dessus.

## 9 Utilisation du Web

La navigation sur Internet s'effectue à l'aide des navigateurs mozilla-firefox, ou konqueror (commande `mozilla-firefox`, ou `konqueror` dans un terminal).

—> **Manipulation 8** : En utilisant le navigateur que vous avez déjà lancé, connectez vous aux sites de l'université `www.univ-paris13.fr`, puis de l'institut Galilée `www-galilee.univ-paris13.fr`.

## 10 Utilisation de la messagerie

Votre adresse électronique est sous la forme `prenom.nom@ig-edu.univ-paris13fr`.

Vous pouvez consulter votre messagerie dans un navigateur Web à l'adresse suivante : `https://webmail.ig-edu.univ-paris13.fr`. Vous pouvez ainsi consulter votre courrier électronique à la fois sous UNIX et sous Windows, sur les machines de l'Institut Galilée mais aussi depuis l'extérieur du réseau de l'université.

**ATTENTION** : il est important de ne pas oublier le **S** de `https`.

—> **Manipulation 9** : En utilisant le navigateur que vous avez déjà lancé, consultez votre messagerie et envoyez un message à votre voisin.

D'autres logiciels de messagerie sont également disponibles, notamment kmail.

Vous disposez également d'une adresse électronique sur le site de l'Université à l'adresse `https://webmail.univ-paris13.fr`. Votre adresse électronique est alors `<Numero Carte d'\`etudiant>@univ-paris13.fr` (par exemple, `1234567890@univ-paris13.fr`).

## 11 Présentation de la structure arborescente sur UNIX

Un système de fichiers est une structure logique permettant de gérer l'espace disque. Chaque disque logique (partition) doit avoir un système de fichiers pour pouvoir être utilisable.

Les systèmes de fichiers sous UNIX sont organisés sous la forme d'une arborescence. Il y a une hiérarchisation des systèmes de fichiers. On rattache au système de fichiers principal, les autres systèmes de fichiers. On parle de montage (commande `mount`) et de démontage (commande `umount`) de systèmes de fichiers. De cette manière, on accède aux autres disques (par exemple, une disquette, une clé USB ou un CDRom) comme à n'importe quel répertoire.

Les noms de fichiers peuvent contenir jusqu'à 255 caractères alphanumériques. Le système UNIX considère les lettres en majuscule et en minuscule comme des caractères différents. Les fichiers cachés commencent tous par un point (`.cshrc`).

**Remarque** : Chaque répertoire contient deux fichiers particuliers :

- `.` : le répertoire courant ;
- `..` : le répertoire père.

### 11.1 Les fichiers ordinaires

Bien qu'il n'y ait pas de distinction au niveau système, il est important de distinguer, pour les commandes que nous utiliserons :

- Les fichiers binaires
- Les fichiers textes : structurés en ligne (suite de caractères se terminant par un caractère `<fin de ligne>`, non visible à l'écran).

La plupart des commandes que nous utiliserons considèrent la ligne comme l'unité élémentaire. Les opérations sur les fichiers textes se feront la plupart du temps ligne par ligne.

Un fichier peut être créé directement par un programme (par exemple, `gcc`) ou par un utilisateur à l'aide d'un éditeur (`nedit`, `kedit`, `emacs`).

### 11.2 Les droits

Les fichiers sous UNIX possèdent des droits décrivant leurs modes d'utilisation et d'accès : lisible, modifiable, ou exécutable (pouvant être ouvert, dans le cas d'un répertoire). Ces droits sont répartis sur trois niveaux correspondant à trois classes d'utilisateurs :

- Le propriétaire du fichier (*user*) ;
- Le groupe auquel appartient le propriétaire (fixé par l'administrateur du système) (*group*) ;
- Les autres utilisateurs (*other*).

Il existe aussi une autre classe : Le *super utilisateur*. Il possède tous les droits sur tous les fichiers.

Chaque fichier appartient à un utilisateur (**user**) et à un groupe (**group**). Ils possèdent des droits en lecture (**r**) et écriture (**w**) et à l'exécution (**x**) pour l'utilisateur, le groupe et les autres (**others**).

On peut visualiser les droits à l'aide de la commande `ls` et de l'option `-l` :

```
-rwxr--r-- 1 sadki enseignant 1365 sep 17 12 :13 fichier1
```

Les dix premiers caractères indiquent :

- La nature du fichier sur le premier caractère (ordinaire : -, répertoire : d);
- Les droits pour le propriétaire sur les trois caractères suivants;
- Les droits pour le groupe sur les trois caractères suivants;
- Les droits pour les autres sur les trois derniers caractères.

Pour l'exemple ci-dessus :

Droit en lecture pour l'utilisateur, le groupe et les autres ;

Droit en écriture pour l'utilisateur ;

Droit en 'exécution pour l'utilisateur ;

Exemple pour un répertoire :

```
drwxr-xr-x 2 sadki enseignant 4096 sep 17 12 :12 Repertoire1
```

Droit en lecture pour l'utilisateur le groupe et les autres ;

Droit en écriture pour l'utilisateur ;

Droit à l'ouverture pour l'utilisateur le groupe et les autres ;

→ **Manipulation 10** : *Effectuez les opérations ci-dessous.*

N.B. : Plusieurs commandes utiles à la manipulation de fichiers (création, déplacement, suppression, etc) sont présentées en annexe. Vous êtes invité à la consulter.

- à l'aide de la commande `mkdir`, créer un répertoire **AS**, puis créer un sous-répertoire **TP1** de **AS**.
- Visualiser les droits sur les répertoires créés à l'aide de la commande `ls`.
- Déplacez vous dans le répertoire **TP1** à l'aide de la commande `cd`.
- Assurez vous que vous êtes bien dans le répertoire **TP1** que vous venez de créer (les opérations de cette manipulation y seront effectuées) à l'aide de la commande `pwd`.
- Déplacer le programme **C** que vous avez écrit et l'exécutable correspondant dans le répertoire courant (c'est-à-dire **AS/TP1**) à l'aide de la commande `mv`
- Visualiser les droits des deux fichiers.
- Lancer l'exécutable, que se passe-t-il ?
- Modifiez les droits de l'exécutable de la manière suivante : `chmod -x nom_de_l'ex\ecutable`
- Visualiser les droits de l'exécutable.
- Lancer l'exécutable, que se passe-t-il ?

## 12 Les shells : lignes de commandes

Les systèmes d'exploitation de type UNIX offrent plusieurs interfaces utilisateur interactives (on parle de shell ou interpréteur de commandes). Elles sont

constituées d'un programme séparé, permettant à l'utilisateur de dialoguer avec le cœur du système d'exploitation (le noyau). Elle intègre un langage de commandes très sophistiqué (script). L'utilisateur pourra notamment avoir accès au shell é travers un terminal.

Contrairement à des environnements uniquement graphiques, UNIX facilite la communication d'informations entre les différents traitements.

Il existe plusieurs mécanismes de communication au niveau du shell. Chaque programme se voit associer trois fichiers standards :

- Le fichier d'entrée contenant les données utilisées par le programme (par défaut, le clavier)
- Le fichier de sortie contenant les résultats du programme (par défaut l'écran)
- Le fichier de sortie des erreurs lors de l'exécution du programme (par défaut, l'écran).

## 12.1 Redirection d'entrée-sortie

Les shells offrent la possibilité de rediriger les entrées/sorties des programmes.

### Redirection sortante

Redirection de l'écran vers un fichier, pour stocker les résultats.

- Option `-o`
- Avec le signe de redirection sortante : `>`.

*Programme > fichier*

Exemple :

```
tail fichier.txt > 10premiereslignes.txt
```

Attention, `>` écrase le fichier s'il existe. Pour ajouter (concaténer) les résultats dans un fichier, on utilise le signe `>>`.

### Redirection entrante

Les commandes traitent, par défaut, leurs données à partir du clavier, si on ne mentionne pas de nom de fichier en argument.

La commande `tr` est un cas particulier. On ne peut pas mentionner de nom de fichier. La commande lit le clavier.

Pour que les commandes lisent les données dans un fichier, on peut utiliser le symbole de redirection entrante `<`.

Par exemple :

```
tr '\011' '@' < fichier.txt
```

```
tail < fichier.txt
```

Les deux modes de redirection peuvent être combinés :

```
tail < fichier.txt > fin_fichier.txt
```

→ **Manipulation 11** : *Enregistrez dans le fichier `ListeProc.txt` la liste des programmes en cours d'exécution sur votre machine.*

Pour visualiser les résultats d'une commande, on peut les sauvegarder dans un fichier, puis utiliser `cat` ou `more`.

→ **Manipulation 12** : *Visualisez le fichier `ListeProc.txt` avec `cat` puis `more`*

## 12.2 Mécanisme de communication entre processus : les tubes

Il est aussi possible de passer des informations d'une commande à une autre en créant un canal de communication, symbolisés par `|`. On appelle ce mécanisme, un tube.

Les données produites par la première commande sont communiquées à la seconde commande pour qu'elle puisse les traiter.

Exemple :

```
head fichier1.txt | tail -2
```

Affichage de la neuvième et de la dixième ligne du fichier `fichier1.txt`.

→ **Manipulation 13** : *Utiliser la commande `wc -l` pour compter le nombre de programmes en cours d'exécution sur votre machine.*

## Annexe – Commandes UNIX utiles en TP

### A Introduction

Chaque série de caractères tapée et se terminant par un retour chariot (entrée, **Enter**) est interprétée par le shell comme un commande (première série de caractères jusqu'à un séparateur) suivie de ses arguments. Les séparateurs sont l'espace et la tabulation.

Syntaxe générale des commandes UNIX :

**commande** [**argument(s)**]

En général, une commande lit ses données sur le clavier (on parle l'entrée standard) et écrit les données en sortie sur l'écran (la sortie standard). Les erreurs sont écrites sur l'écran (la sortie en erreur standard).

**ATTENTION** : sous UNIX, les minuscules et les majuscules sont différenciées.

Par exemple, **LICENCE** et **licence** sont deux chaînes de caractères différentes.

### B Gestion et déplacement des arborescences

#### Visualisation du contenu d'un répertoire

**ls Options**

La commande peut prendre en argument :

- un ou plusieurs répertoires
- une liste de fichiers

Options :

**-a** : Affiche également les fichiers cachés

**-l** : Affiche toutes les informations sur les fichiers (droits d'accès, propriétaire, taille, dernière date de mise à jour)

**Exemples :**

**ls -l**

Affichage à l'écran :

```
-rwxr--r-- 1 sadki enseignant 1365 sep 17 2009 exo1tp1.c
drwxr-xr-x 2 sadki enseignant 4096 sep 17 2009 Mestp
```

**ls -al**

Affichage à l'écran :

```
drwxr-xr-x 5 sadki enseignant 4096 sep 17 2009 .
drwxrwxrwx 4 sadki enseignant 4096 sep 17 2009 ..
drwxr-xr-x 6 sadki enseignant 4096 sep 17 2009 AS
drwxr-xr-x 3 sadki enseignant 4096 sep 17 2009 TP1
drwxr-xr-x 2 sadki enseignant 4096 sep 17 2009 TP2
-rwxr--r-- 1 sadki enseignant 1365 sep 17 2009 exoTP2.c
```

## Déplacement dans un répertoire

`cd nom du répertoire`

**Exemples :**

- `cd Repertoire1` (Déplacement dans le répertoire `Repertoire1`)
- `cd /export/home/users/enseignants/sadki/AS` (Déplacement dans le répertoire `/export/home/users/enseignants/sadki/AS`)
- `cd` (Retour au répertoire de connexion)
- `cd ..` (Retour au répertoire père)

## Affichage du répertoire courant

`pwd`

**Exemple :** `pwd`

Affichage à l'écran :

`/export/home/users/Enseignants/sadki`

## Création d'un répertoire

`mkdir nom du répertoire`

**Exemple :** `mkdir Repertoire1` (création du répertoire `Repertoire1`)

## Destruction d'un répertoire

`rmdir nom du répertoire`

**Exemple :** `rmdir Repertoire1` (destruction du répertoire `Repertoire1`)

# C Gestion et Manipulation des fichiers

## Copie de fichiers

`cp options fichier1 fichier2`

(Copie du contenu de `fichier1` dans `fichier2`)

ou

`cp options fichier1 [fichier2 ...] répertoire`

(Copie du contenu des fichiers `fichier1` et `fichier2` dans `répertoire`)

Options :

- `-i` : mode interactif (confirmation avant écrasement) ;
- `-r` : copie récursive de répertoires ;
- `-p` : conserve la date du fichier source.

Exemples :  
`cp /etc/fstab .`  
`cp -r rep1 rep2`

## Déplacement de fichiers

`mv options fichier1 [fichier2 ...] repertoire`  
Déplace les fichiers *fichier1* et *fichier2* dans *repertoire*.  
Options :  
– *-i* : mode interactif (confirmation avant écrasement).  
– *-f* : force la commande  
Exemple :  
`mv fichier1 rep1`

## Destruction de fichiers

`rm options fichier1 [fichier2 ...]`  
Suppression des fichiers spécifiés.  
Options :  
– *-i* : mode interactif (confirmation avant écrasement).  
– *-f* : force la commande  
– *-r* : supprime récursivement les répertoires.  
Exemples :  
`rm fichier1`  
`rm -r repertoire1`

## Visualisation ou concaténation de fichiers

`cat fichier...`  
Affiche à l'écran (la sortie standard) le fichier donné en argument ou les concatène (ajout les uns à la suite des autres) s'il y en a plusieurs.  
Par défaut, la lecture s'effectue sur l'entrée standard et l'affichage sur la sortie standard.  
Exemple :  
`cat /etc/fstab`  
Affichage à l'écran du contenu du fichier `/etc/fstab`.

## Visualisation du début/de la fin d'un fichier

`head -n fichier` : affichage des *n* premières lignes d'un fichier  
`tail -n fichier` : affichage des *n* dernières lignes d'un fichier  
Par défaut, *n* vaut 10.

## Visualisation d'un fichier page par page

`more fichier...`

Affichage à l'écran, page par page, des fichiers spécifiés.

## Tri des lignes d'un fichier

`sort -ufnr -o fic fichier...`

Par défaut, tri par ordre alphabétique.

Options :

`-u` : n'affiche qu'une fois les lignes multiples.

`-f` : ne différencie pas les minuscules des majuscules

`-n` : effectue un tri numérique

`-r` : ordre décroissant

`-o fic` : spécifie un fichier de sortie.

Autre option :

`-k <premier champ><type>,<dernier champ><type>`

Par exemple, `sort -k 2n,2n` (Tri sur le deuxième champ).

## Décompte des nombres mots/lignes/caractères d'un fichier

`wc -lwc fichier...`

Options :

`-l` : Affiche que le nombre de lignes

`-w` : Affiche que le nombre de mots

`-c` : Affiche que le nombre de colonnes

Si aucune option n'est spécifiée, la commande affiche le nombre de lignes, de mots et de colonnes.

## Substitution de caractères

`tr caracteres caracteres < fichier`

Substitue un caractère par un autre (Suppression des caractères invisibles).

On peut utiliser le code hexadécimal du caractère.

Exemple :

`tr '\011' '@' < fichier1`

## Affichage de certaines colonnes d'un fichier

```
cut -d delim -f champs fichier
```

Options :

-d *delim* : spécifie le délimiteur (entre double quote, " );

-f *champs* : spécifie les champs à afficher (séparés par une virgule).

Le délimiteur par défaut est la tabulation.

Exemple :

```
cut -f1 /etc/fstab
```

Affichage de la première colonne du fichier */etc/fstab*.

## Modification des droits d'un fichier/répertoire

```
chmod [-R] <classe d'utilisateur>< nature de la modification>< droit fi-  
chiers>
```

- [-R] : Opération récursive sur les fichiers et les sous-répertoires

- *classe d'utilisateur* :

**u** : le propriétaire;

**g** : le groupe;

**o** : les autres;

**a** : les trois classes.

- *nature de la modification* :

+ ajout du droit

- retrait du droit

= affectation du droit

- *droit* :

**r** droit en lecture

**w** droit en écriture

**x** droit en exécution

Exemples :

```
chmod g-rw fichier1
```

```
chmod -x fichier2
```

## Recherche de fichiers ou de répertoires

```
find répertoire options
```

Recherche dans toute l'arborescence à partir du point spécifié.

Options courantes :

**-name** *fichier* : recherche sur le nom *fichier*

**-print** : affiche le résultat de la recherche

**!** : négation de la recherche

Exemple :

```
find / -name "fstab" -print
```

Recherche des fichiers dont le nom est **fstab**, dans tous les répertoires.

## Recherche d'une chaîne de caractères dans un fichier

```
grep options expression fichier
```

## D Archivage et compression

Les systèmes UNIX offrent de nombreux outils d'archivage et de compression des fichiers. Les commandes les plus couramment utilisés sont **gzip** et **tar** :

– **gzip** *fichier*

Cette commande permet la compression de fichiers donnés en argument.

Le fichier produit possède une extension supplémentaire : **gz**

L'utilisation de l'option **-d** permet la décompression du fichier préalablement. La commande **gunzip** *fichier* est équivalente à l'utilisation de cette option.

Exemples :

– Compression du fichier *Fichier1.txt* : **gzip** *Fichier1.txt*

Le fichier compressé s'appelle *Fichier1.txt.gz*

– Décompression du fichier *Fichier1.txt.gz* : **gzip** *Fichier1.txt.gz*

Le fichier décompressé s'appelle *Fichier1.txt*

– **tar** *Options FichierArchive Fichiers*

Cette commande permet l'archive (et la compression) d'un ensemble de fichiers et de répertoires mais aussi le désarchivage (et la décompression) d'archives créées précédemment.

– Archivage (et compression) : **tar czvf** *Archive.tar.gz Fichiers*

– Désarchivage (et décompression) : **tar xzvf** *Archive.tar.gz*

(les fichiers contenus dans l'archive sont créés dans le répertoire courant)

## E Gestion des programmes en cours d'exécution

Les commandes suivantes permettent d'interagir avec les programmes en cours d'exécution sur la machine.

– **ps -aux** : Visualisation de programmes en cours d'exécution avec leur identifiant (PID)

– **kill** *PID* : Terminaison immédiate du processus de pid *PID*

## F Commandes diverses

### Affichage de texte sur la sortie standard

`echo -n 'message'`  
Option : `-n` Pas de retour chariot final

### Effacement de l'écran

`clear`  
(Pas d'option)

## G Le manuel UNIX

La commande `man nom` permet de visualiser l'aide de la commande ou de la fonction spécifiée.

**Exemple** : `man ls` : Affiche l'aide de la commande `ls`.