

## **TD 1 : Introduction à UNIX :**

Ces exercices ont été effectués sous une version unix : **Mandriva Linux 2006**. Vous pouvez la commander en CD, ou la télécharger gratuitement à cette adresse :

<http://www.mandriva.com/fr/download> .

D'une version à une autre, il peut y avoir quelques différences au niveau du shell, mais la logique reste la même dans tous les exercices.

Vous êtes libre de modifier le document pour votre usage personnel. Pour toute fautes, erreurs, ou questions, s'adresser à notre prof de TD : Mr Thomas Tang, [thomas.tang@cgm.cnrs-gif.fr](mailto:thomas.tang@cgm.cnrs-gif.fr) et/ou me le signaler par mail.

L'invite de commande des lignes à taper est représenté par le symbole \$

### **Exercice 1 : Effectuer des actions**

– *Placez-vous dans votre répertoire personnel.*

Pour se placer dans le répertoire personnel il faut taper : \$ cd

– *Créer un sous répertoire nommé test.*

\$ mkdir test

– *Copier dans ce répertoire le fichier passwd du répertoire /etc.*

En étant placé dans notre home/ on tape : \$ cp /etc/passwd

– *Faites que le répertoire test devienne le répertoire courant*

On se place dans le répertoire test : \$ cd test

– *Afficher le contenu détaillé du répertoire courant.*

\$ ls -l

– *Copier le fichier passwd en CopieMotDePasse.*

\$ mv passwd CopieMotDePasse

– *Renommer le fichier passwd en fichier .passwd.*

\$ mv passwd .passwd

– *Créer un répertoire Sauvegarde.*

\$ mkdir Sauvegarde

– *Revenir au répertoire parent.*

\$ cd

– *Afficher le contenu non-détaillé exhaustif du répertoire test en discriminant fichiers et répertoires.*

\$ ls -F

– *Afficher le contenu du fichier CopieMotDePasse à l'écran.*

\$ more CopieMotDePasse

– *Supprimer le répertoire test.*

\$ rm -R ou \$ rm -Rf (pour forcer à effacer le répertoire même s'il contient des fichiers ou autres dossiers). La commande \$ rmdir test permet d'effacer le répertoire s'il ne contient aucuns fichiers.

## **Exercice 2 : Compléter les blancs**

1 : corbeau.txt  
2 : corbeau.txt pascal/ textes/  
3 : mv : cannot stat `pascal/renard.txt' : No such file or directory  
4 : pascal/corbeau.txt  
5 : mv corbeau.txt ../textes  
6 : textes

Pour tester les réponses sur la machine, il faut créer les répertoires et fichiers mentionnés dans l'énoncé :

\$ mkdir home2 (dans le répertoire home/)  
\$ mkdir enseign (dans le répertoire home2/)  
\$ mkdir pascal (dans le répertoire enseign/)  
\$ touch corbeau.txt (dans le répertoire enseign/)  
\$ touch bonjour.pas (dans le répertoire pascal/)  
\$ touch toto.pas (dans le répertoire pascal/)

## **Exercice 3 :**

1 :  
Déterminer le fichier le plus petit : \$ ls -S /sbin | tail -n 1  
Déterminer le fichier le plus grand : \$ ls -Sr /sbin | tail -n 1 ou \$ ls -S /sbin | head -n 1  
Déterminer le fichier le plus vieux : \$ ls -St /sbin | tail -n 1  
Déterminer le fichier le plus récent : \$ ls -St /sbin | head -n 1

2 :  
Pour connaître les droits accordés au fichier /etc/passwd, il faut taper : \$ ls -l /etc  
Réponse > -rw-r- -r- - 1 root root ... ..  
La première série de 3 droits donne les droits du User  
La deuxième série de 3 droits donne les droits du Group  
La troisième série de 3 droits donne les droits des Others (reste du monde)  
On ne peut l'effacer comme il n'y a pas de d en première position des droits du User.

3 :  
Copier le fichier /etc/passwd dans notre répertoire principale : \$ cp /etc/passwd ~/  
Ce fichier est désormais dans notre répertoire et nous appartient (il change de group). Il y a les mêmes droits dessus. Pour modifier le fichier tel que les Others ne puissent plus le lire, il faut taper : \$ chmod o-r passwd

#### **Exercice 4 :**

1 :

> cd ~

Cette commande sert à se placer dans le répertoire personnel.

> ls -dF /etc/r\*

Afficher tous les répertoires uniquement (-d) du répertoire etc/ contenant en première lettre « r » dans leurs noms, avec un / (F).

> ls /bin/?sh

Affiche le chemin relatif des fichiers du répertoire bin/ contenant les 2 dernières lettres « sh ».

> cp \*/passwd .

Copier tous les fichiers passwd dans notre répertoire personnel se trouvant à distance 1 de la racine (\*).

> rm -f pass\*

Supprime tous les fichiers contenant les lettres « pass » dans leurs noms de fichiers se trouvant dans le répertoire courant.

2 :

– *Afficher la liste des fichiers du répertoire /bin dont la première lettre est un m.*

\$ ls -d /bin /m\*

– *Afficher la liste des fichiers du répertoire /bin dont la deuxième lettre est un z.*

\$ ls -d /bin/ ?z\*

– *Copier tous les fichiers à distance 2 de la racine qui ont pour deuxième lettre un t et pour troisième un i dans un répertoire tmp de votre répertoire personnel.*

\$ cp \*/\*/ ?ti\* ~/tmp

Si on se trouve dans le rép perso : \$ cp ../../ \*/\*/ ?ti\* tmp    ou    \$ ../../ ?ti\* tmp

#### **Exercice 5 :**

1 : \$ echo Longtemps je me suis couche de bonne heure > proust.txt

2 : \$ cat > moliere.txt

belle marquise

vos yeux

me font

mourir

amour

[puis la touche Ctrl + D]

3 : \$ (ls -l /etc) > fichierliste.txt

### **Exercice 6 :**

```
1 : $ wc -w proust.txt
2 :   $ cat moliere.txt | sort -d ou $ sort
   $ cat moliere.txt | sort -dr
3 : $ sed s/couche/leve/ proust.txt | less
4 : $ who
5 : $ /sbin/ifconfig (pour afficher l'ip)
   $ IP=193.55.28.156 (pour affecter l'ip 193.55.28.156 à la variable d'environnement IP)
   $ echo $IP (pour afficher la valeur de la variable d'environnement IP)
6 : $ grep -a firefox | ps -eaf (et vérifier visuellement dans la liste si le processus firefox est lancé).
```

### **Exercice 7 :**

```
> export toto=coucou
Définit la variable toto avec le contenu coucou
> export coucou=toto
Définit la variable coucou avec le contenu toto
> echo toto
Affiche le texte : toto
> echo $toto
Affiche le contenu de la variable toto : coucou
> echo 'echo $toto'
Affiche le texte entre ' ' : echo $toto
> echo "echo $toto"
Affiche le texte : echo et le contenu de la variable toto : coucou => echo coucou
> echo 'echo $toto'
Affiche le contenu de la variable toto uniquement : coucou
> unset $toto
Supprime la variable coucou (supprime le contenu toto)
> echo $toto
Affiche le contenu de la variable toto : coucou
> echo $coucou
Affiche une ligne blanche, car il n'y a plus de variable coucou
> unset toto
Supprime la variable toto
> echo $toto
Affiche une ligne blanche car la variable n'existe plus.
```

### **Exercice 8 :**

```
1 : $ export lz=`ls -l`
2 : $ alias lz=`ls -l`
3 : $ mkdir test ; touch toto ; cd test
4 : $ echo lz
    $ lz
5 : $ ln -s toto tsymb
6 : $ ln toto tdur (par défaut la commande ln crée des liens dures)
7 : $ echo 'hello world' > toto (ou 'hello world')
    $ cat tsymb
    $ cat tdur
```

Taille des fichiers avec \$ ls -l et les fichiers tdur et toto ont la même taille, tsymb étant 3 fois plus petit (lien symbolique est un raccourci contrairement au lien dure tdur qui est une copie du fichier)

```
8 : $ rm toto
    $ cat tsymb renvoie une erreur (plus de lien source du lien symbolique tsymb)
    $ cat dur renvoie la ligne : hello world
    $ cat > toto [retour chariot] tsymb [touche Ctrl + D]
    $ diff tsymb tdur : cette commande renvoie les lignes différentes dans les 2 fichiers.
9 : $ mv tsymb ../ et $ mv tdur ../
    tsymb ne renvoie rien (chemin d'accès modifié) et tdur renvoie : hello world
10 : rm tsymb ; rm tdur ; ...
```

L'avantage d'un lien symbolique est d'avoir un raccourci dans le répertoire que l'on désire, cependant, si le fichier source est déplacé ou effacer, le lien ne sera plus en état de répondre. Un lien en dur permet une copie complète du fichier source mais n'apporte aucun avantage du point de vue de la taille.

### **Exercice 9 :**

```
1 : $ alias ls
    $ which ls (which renvoie le chemin)
2 : $ alias rm='rm -Rf'
3 : $ alias lx='ls -aF | grep "*"'
```

### **Exercice 10 :**

```
1 : $ ps -eaf
2 : $ nice -n 10 firefox &
    $ man firefox
3 : $ fg %1 puis [Ctrl+Z] puis bg
4 : $ bg
```