

TP2 - Système de Fichiers

Laurent Tichit

12 septembre 2018

Plan

1 Commandes

2 Chemins

3 Find

Commandes du système de fichiers

Les répertoires . et ..

Tapez les commandes suivantes :

- `tichit@iml230:~$ echo $HOME`
- `tichit@iml230:~$ echo ~`

Lister les fichiers

- Lister le nom des fichiers du répertoire courant : **ls** (pour *list*)
tichit@iml230:~/LaserBeam\$ **ls**
Diffusion_LaserBeam.class Diffusion_LaserBeam.java
- **ls -a** permet de visualiser les fichiers cachés par défaut. Tout fichier dont le nom commence par "." est caché (., .., fichiers de configuration)
- Tapez **ls**, puis **ls -a** dans votre répertoire d'accueil.

Supprimer des fichiers

- Détruire un fichier : `tichit@iml230:~/LaserBeam$ rm [fichier]`
(pour *remove*)
Attention, sous Unix, par défaut, un fichier détruit ne peut pas être récupéré.

Informations associées aux fichiers

Des informations spécifiques sont attachées aux fichiers :

```
tichit@iml230:~/Test$ ls -l
```

```
total 36
```

```
-rw-r--r-- 1 tichit tichit 9145 2009-04-03 11:44 Test.txt  
-rw-r--r-- 1 tichit tichit 21702 2009-04-03 11:50 Titi.pl
```

Les informations sont dans l'ordre :

- Type du fichier et permissions
- Nombre de liens sur le fichier
- Le nom du propriétaire
- Le nom du groupe auquel appartient le propriétaire
- Taille exprimée en octets
- Date et heure de la dernière modification
- Nom du fichier

Exercice

On peut conjuguer les options **-a** et **-l**

Tapez :

- tichit@iml230:~/Test\$ **ls -la**

- tichit@iml230:~/Test\$ **ls -al**

et comparez.

Copie, déplacement, renommage

- Afficher le contenu d'un fichier sur la sortie standard (le terminal actif) : **cat fichier** (pour *catalog*).

```
tichit@iml230:~ cat HelloWorld.pl
```

```
#!/usr/bin/perl  
print("Hello World!\n");
```

- Dupliquer un fichier : **cp** (pour *copy*).

```
tichit@iml230:~/Test$ ls
```

```
Test.txt
```

```
tichit@iml230:~/Test$ cp Test.txt Toto.txt
```

```
Test.txt Toto.txt
```

- Renommer/Déplacer d'un fichier : **mv** (pour *move*).

```
tichit@iml230:~/Test$ mv Toto.txt Titi.pl
```

```
Test.txt Titi.pl
```

Création de fichier, type du fichier

- **touch fichier :**

- ▶ Modifie la date de dernier accès au fichier
- ▶ Crée le fichier s'il n'existe pas.

```
tichit@iml230:~$ ls -l
```

```
-rw-r--r-- 1 tichit tichit 21702 2009-04-03 11:50 Titi.pl
```

```
tichit@iml230:~$ touch Titi.pl
```

```
tichit@iml230:~$ ls -l
```

```
-rw-r--r-- 1 tichit tichit 21702 2009-04-03 12:41 Titi.pl
```

- **file [-f] fichier :** Décrit le contenu du fichier.

```
tichit@iml230:~$ file script.sh
```

```
script.sh: Bourne-Again shell script text executable
```

```
tichit@iml230:~$ file script.py
```

```
script.py: Python script, UTF-8 Unicode text executable
```

Création/destruction d'un répertoire

- **mkdir répertoire...** : crée des répertoires.

```
tichit@iml230:~/Test$ mkdir rep1
```

```
tichit@iml230:~/Test$ ls
```

```
rep1/ Test.txt Titi.pl
```

- **rmdir répertoire...** : supprime des répertoires (vides!).

```
tichit@iml230:~/Test$ rmdir rep1
```

```
tichit@iml230:~/Test$ ls
```

```
Test.txt Titi.pl
```

- Pour supprimer un répertoire et son contenu :

rm -fr répertoire (r = recursive ; f=force).

ATTENTION quand vous utilisez **rm** !

Chemins

Chemin des fichiers

- Plutôt que de recourir tout le temps à la commande **cd**, il est plus rapide de se servir des chemins.

- Ex : Préférer :

```
tichit@iml230:~/Test$ cat /etc/fstab
```

à

```
tichit@iml230:~/Test$ cd /etc
```

```
tichit@iml230:~/Test$ cat fstab
```

On a aussi :

- `tichit@iml230:~/Test$ mv ~/Desktop/image.jpg ~/Images/`
- `tichit@iml230:~/Test$ mv ~/Desktop/image.jpg`
`~/Images/image1.jpg`
- ...

Exercice

En utilisant le manuel :

- Que fait la commande **mkdir A/B/C**?
- Que fait la commande **mkdir -p A/B/C (man mkdir)**?

Exercice

En utilisant le manuel :

- Que fait la commande **mkdir A/B/C** ?
- Que fait la commande **mkdir -p A/B/C** (man **mkdir**) ?
C'est plus rapide que **mkdir A ; mkdir A/B ; mkdir A/B/C**
ou que **mkdir A ; cd A ; mkdir B ; cd B ; mkdir C**

Chemin des fichiers

Pour obtenir le nom d'un fichier à partir de son chemin complet, on utilise la commande suivante :

- **basename chaîne suffixe** : supprime le plus long préfixe se terminant par '/' de la chaîne et le suffixe (s'il se trouve en fin de chaîne).

```
tichit@iml230:~$ basename /home/tichit/Test/test.txt  
test.txt
```

- **dirname chemin-de-fichier** : donne le chemin du répertoire dans lequel se trouve le fichier. tichit@iml230:~\$ **dirname**

```
/home/tichit/Test  
/home/tichit
```


Commandes relatives au système de fichiers

- **du [option] fichier-ou-répertoire** : place disque utilisée par un fichier/répertoire en kilo-octets.

```
tichit@iml230:~$ du Test
```

```
40 Test/
```

- **df [répertoire]** : place libre et occupée sur le(s) système(s) de fichiers.

```
tichit@iml230:~$ df .
```

Sys. de fich.	1K-blocs	Occupé	Disponible	C
/dev/sda6	14136684	9092060	4326524	68% /h

Note : . représente le répertoire courant.

Exercice

- Que fait la commande **df -h** (**man df**) ?
- Combien d'espace disque reste-t'il sur votre partition principale (/) ?

Find

La commande **find**

La commande **find nom-de-chemin expression** permet de rechercher récursivement dans un répertoire les fichiers qui satisfont à l'expression booléenne.

Les expressions peuvent être écrites grâce aux primitives suivantes :

- **+n** signifie $>n$
- **-n** signifie $<n$
- **n** signifie $=n$
- **!** signifie NOT
- **-a** signifie AND
- **-o** signifie OR

n représente un entier positif quelconque.

La commande **find**

Paramètres.

- -name nom : Vrai si fichier est le nom du fichier courant. Il est possible d'utiliser une expression régulière.
- -perm clés : Vrai si les permissions du fichier correspondent au nombre octal donné
- -type c : Vrai si le fichier est de type c (b,c,d,f)
- -links n : Vrai si le fichier a n liens
- -user nom : Vrai si le fichier appartient à nom
- -group gr : Vrai si le fichier est dans le groupe gr
- -size n : Vrai si la taille en blocs du fichier est n
- -inum n : Vrai si le inode du fichier est n
- -atime n : Vrai si le fichier a été lu depuis n jours
- -mtime n : Vrai si le fichier a été modifié depuis n jours
- -exec comm : Vrai si le code de retour de commande est 0
- -ok comm : Comme exec mais demande de confirmation
- -print : Vrai ; imprime le chemin courant
- -newer fich : Vrai si le fichier courant est plus récent que fich

La commande **find**

Exemple d'utilisation.

```
find /home/tichit -name dico.txt -print
```

```
/home/tichit/dico.txt
```

```
find /home/tichit -name "*nn*" -print
```

```
/home/tichit/.mcp/trader-cache/example_autopanner.mcopclass  
/home/tichit/.mcp/trader-cache/example_autopanner.arts.mcoptype  
/home/tichit/.mcp/trader-cache/example_autopanner.arts.mcopclass  
/home/tichit/repertoire/annonce
```

```
...
```

```
/home/tichit/perl5.10/t/Module_Pluggable/10innerpack_inner.t  
/home/tichit/perl5.10/ext/Encode/ucm/macROMnn.ucm  
/home/tichit/perl5.10/ext/Encode/ucm/macRUMnn.ucm
```

```
find /home/tichit -type d -name "*" -print
```

```
/home/tichit  
/home/tichit/public_html  
/home/tichit/public_html/Perl  
/home/tichit/public_html/Perl/Exos  
/home/tichit/public_html/Perl/Jaquettes
```

```
...
```

La commande **find**

Autre exemple.

```
find /home/tichit -size +20000 -mtime -1 -exec ls -l {} \;
```

```
-rw----- 1 tichit tichit 12913843 2009-04-24 15:02 /home/tichit/...  
-rw-r--r-- 1 tichit tichit 53919744 2009-04-24 15:00 /home/tichit/...
```

- **-size +20000** : fichiers de plus de 20000 Ko = 20 Mo
- **-mtime -1** : fichiers accédés en écriture depuis moins de 1 jour
- **-exec ls -l {} \;** : pour chacun des fichiers trouvés, lancer la commande **ls -l fichier**.
{} représente le nom du fichier, et \; indique que la commande **ls** est terminée.