

TP 1 : Introduction à Linux

NFP136 : Valeur d'Accueil et de Reconversion en Informatique 2

Les TP seront réalisés sous GNU/Linux. Pour commencer, démarrez une machine sous GNU/Linux et connectez-vous en utilisant vos login et mot de passe habituels ou en utilisant les login/mot de passe génériques : `licencep / 7002n*`

1 Introduction

1.1 Le système de fichiers

Un système de fichier est organisé sous forme *arborescente* où les fichiers sont contenus dans des *répertoires* (ou *dossiers*).

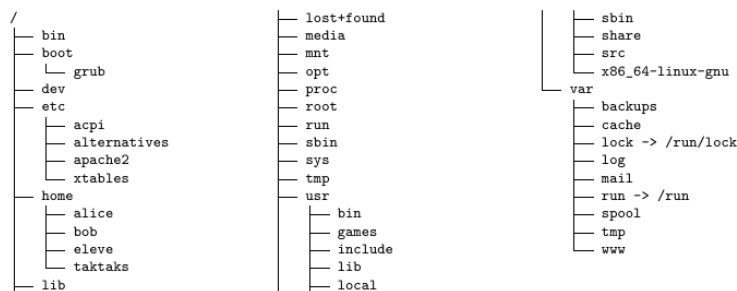


FIGURE 1 – Linux Root File System

Sous UNIX, il existe *une seule* arborescence pour accéder à tous les fichiers indépendamment du nombre d'unité de stockage (partitions, Disques dur, Clés USB, ...).

Sous MS Windows, il existe une arborescence par unité de stockage. Chaque arborescence est désignés par une lettre suivis de ':' (C:, D:, ...).

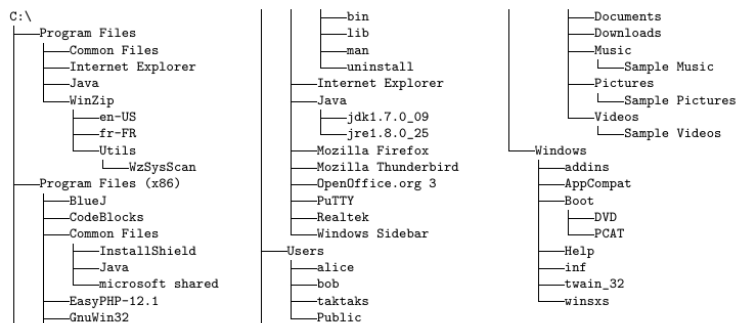


FIGURE 2 – Windows C Root File System

Pour la suite du TP, nous allons considérer les systèmes de fichiers de type UNIX.

1.2 Les Fichiers

Le système de fichiers est donc organisé en une seule structure arborescente contenant des fichiers. Les noeuds sont appelés *répertoires* et les feuilles *fichiers*.

On distingue trois types de fichiers :

- Les fichiers ordinaires : ils contiennent les données, les programmes et les données des utilisateurs et du système ;
- Les fichiers répertoires ou simplement répertoires : un répertoire contient une liste de fichiers incluse dans le répertoire ;
- Les fichiers spéciaux : ils désignent des moyens de communication soit avec les périphériques, soit avec le système, soit entre processus.

La syntaxe d'un nom de fichier n'est pas très stricte, mais certains caractères sont à éviter :

- caractères ayant une signification particulière : \ > < | \$? & [] * ! " ' () @ ~ `
- caractères peu pratiques à l'usage : caractères spéciaux et accentués.

remarque : Le point (".") joue un rôle particulier lorsqu'il se trouve en première position dans le nom d'un fichier. En effet, un fichier dont le nom commence par un point est un fichier caché (c'est-à-dire qu'il n'apparaît pas par défaut lorsque l'on liste le contenu d'un répertoire).

1.3 Répertoires Spécifiques

Le système de fichier étant organisé de façon arborescente, il possède un répertoire racine noté "/" et appelé *root*. C'est ce répertoire qui contient tous les répertoires et tous les fichiers du système.

Un répertoire peut donc contenir des sous-répertoires et/ou des fichiers ordinaires. Un fichier est repéré par son nom et sa position dans l'arborescence, son chemin d'accès :

`/repertoire/sous-repertoire/sous-sous-repertoire/nom-du-fichier`

Nom de répertoires spéciaux :

- Le répertoire *home* d'un utilisateur est le répertoire contenant les fichiers de cet utilisateur ; ce répertoire est désigné par `~`
- Le répertoire courant est désigné par `.`
- Le répertoire parent du répertoire courant est désigné par `..`

1.4 Principales Commandes de Manipulation de Fichiers et Répertoires

1.4.1 Manipulation des répertoires :

commande	argument(s)	Description	origine du nom
<code>cd</code>	<code>nom_rep</code>	Changer de répertoire courant	call directory
<code>ls</code>	<code>nom_rep</code>	Lister le contenu du répertoire	list
<code>mkdir</code>	<code>nom_rep</code>	Créer un nouveau répertoire	make directory
<code>rmdir</code>	<code>nom_rep</code>	Supprimer un répertoire	remove directory
<code>pwd</code>	<i>(aucun)</i>	Afficher le nom du répertoire courant	print current/working directory

Exemple :

- Obtenir le nom du répertoire courant : `amelie@toto: > pwd /home/eleve`
- Lister le contenu du répertoire courant : `amelie@toto: > ls`

Archives	Documents	public_html
bin	Downloads	Téléchargements
bonjour.txt	mbox	test.txt
Bureau	Music	tmp
Desktop	Pictures	Videos
- Lister le contenu d'un répertoire : `amelie@toto: > ls /usr/bin`

```

7z                ab                adb2mhc
7za               aclocal          addftinfo
a2p              aconnect          addpart
a2ping           acpi              add-patch
a2x              acpi_listen       adhocfilelist
a5booklet        acyclic           aj
a5toa4           adb               aj5
— Créer un nouveau répertoire : amelie@toto: > mkdir NSY103
— Supprimer un répertoire (le répertoire doit être vide) : amelie@toto: > rmdir tmp

```

1.4.2 Manipulation des fichiers :

commande	argument(s)	Description	origine du nom
cat	nom_fichier	Afficher le contenu d'un fichier	Concatenate files
less	nom_fichier	Afficher le contenu d'un fichier page par page	pager
cp	nom_src nom_dest	Copie d'un fichier	Copy
cp	-r nom_src rep_dest	Copie d'un répertoire	Copy (recursive)
mv	nom_src rep_dest	renommer/déplacer un fichier	<i>move</i>
rm	nom_fichier	Suppression d'un fichier	<i>remove</i>

Exemples :

```

— Afficher le contenu du fichier bonjour.txt
amelie@toto: > cat bonjour.txt
Bonjour le monde!

— Copie d'un fichier :
amelie@toto: > ls bonjour.txt
amelie@toto: > cp bonjour.txt bonjour2.txt
amelie@toto: > ls
bonjour2.txt bonjour.txt

— Copie d'un répertoire :
amelie@toto: > ls rep1
bonjour2.txt bonjour.txt
amelie@toto: > cp -r rep1 rep2
amelie@toto: > ls
rep1/ rep2/
amelie@toto: > ls rep2
bonjour.txt bonjour2.txt

— Renommer un fichier
amelie@toto: > mv rep2 rep_bonjour
amelie@toto: > ls
rep1/ rep_bonjour/

— Suppression d'un fichier bonjour2.txt dans le repertoire rep1 :
amelie@toto: > rm rep1/bonjour2.txt
amelie@toto: > ls rep1
bonjour.txt

```

1.5 Obtenir de l'aide : les pages de manuels

Remarque : Si votre terminal n'est pas en français, vous pouvez le mettre en français en entrant la commande suivante :

```
amelie@toto: > export LANG="fr_FR.UTF-8"
```

Les différentes commandes présentées jusqu'à présent possèdent un certain nombre d'options. Afin d'obtenir de l'aide sur les différentes options d'une commande, il existe plusieurs possibilités :

- l'aide en ligne de commande, souvent obtenue grâce à l'option `--help` ou `-h`.

Exemple : pour afficher l'aide en ligne de la commande `ls`

```
amelie@toto: > ls --help
```

- en consultant les pages de manuel grâce à l'utilitaire `man`.

Exemple : pour afficher la page de manuel de la commande `ls`

```
amelie@toto: > man ls
```

Les pages de manuel sont organisées en sections, la section 1 concernant les commandes utilisateur. Pour obtenir de l'aide sur l'utilitaire `man`, il suffit de taper la commande

```
amelie@toto: > man man
```

- en consultant les pages d'information grâce à l'utilitaire `info`.

Exemple : pour afficher la page de manuel de la commande `ls`

```
amelie@toto: > info ls
```

2 Exercices

Ouvrez un terminal et, en utilisant les informations ci-dessus, répondez aux questions suivantes :

Exercice 1

1. Créez un repertoire `rep1`

```
amelie@toto: > mkdir rep1
```

2. Déplacez vous dans le repertoire `rep1`

```
amelie@toto: > cd rep1
```

3. Créez un fichier `bonjour.txt`

```
amelie@toto: > touch bonjour.txt
```

Utilisez l'éditeur de texte `kwrite` pour l'éditer :

```
amelie@toto: > kwrite bonjour.txt
```

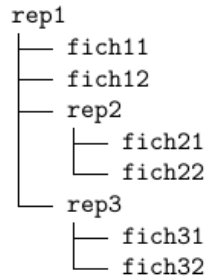
Entrez le texte "Bonjour le monde" et suivez le fichier, puis fermez l'éditeur de texte `kwrite`.

4. Essayez les exemples donnés à la section 1.4 de l'introduction et comparez les résultats obtenus à ceux de l'énoncé.

Exercice 2

1. Déplacez-vous dans le repertoire racine ;
2. Listez les fichiers et repertoires présents à la racine ;
3. Déplacez-vous dans le repertoire `/etc/` ;

4. Ouvrez le fichier `protocols` qui se trouve dans le répertoire `/etc/` avec `less` ;
5. Comment aurait-on pu ouvrir le fichier `/etc/protocols` sans se déplacer dans le répertoire `/etc/` ?
6. Retournez dans votre répertoire personnel ;
7. Affichez le nom du répertoire dans lequel vous vous trouvez ;
8. Tapez la commande `"cd ."` puis affichez le nom du répertoire courant. Que constatez-vous ?
9. Tapez la commande `"cd .."` puis affichez le nom du répertoire courant. Retapez la commande `"cd ."` puis affichez le nom du répertoire courant. Que constatez-vous ?
10. Listez le contenu du répertoire courant. Quelle différence y a-t-il entre le résultat de la commande `"ls"` et le résultat de la commande `"ls ."` ? Que pouvez-vous en conclure sur les noms des répertoires `"."` et `".."` ?
11. Retournez dans votre répertoire personnel et créez un fichier `essai` à l'aide d'un éditeur de texte (par exemple `vi`, `emacs`, `kwrite`, ...);
12. Créez l'arborescence suivante :



13. Déplacez toute l'arborescence `rep3` dans `rep2`.

Exercice 3

1. Trouvez les options de la commande `ls` pour afficher les informations détaillées de toute une arborescence ;
2. Trouvez l'option de la commande `rm` pour supprimer le répertoire `rep1` ainsi que tout son contenu.

Exercice 4 — *la compilation en java*

Java dispose d'un ensemble de classes prédéfinies (une bibliothèque) qui sont rassemblés dans le JDK (**J**ava **D**evelopment **K**it). Pour exécuter des programmes Java, un utilisateur a besoin d'un environnement spécial qui est le JRE (**J**ava **R**untime **E**nvironment). La compilation d'un programme Java se fait par la commande `javac=java compiler`. La compilation d'une classe Java produit du `bytecode` qui est un langage machine pour une machine spéciale : la machine virtuelle Java (JVM). En effet, un programme Java s'exécute au sein d'un processus Java (la machine virtuelle) qui interprète les instructions du `bytecode`. Ce processus lit une instruction, récupère les données la concernant et l'exécute.

1. Ouvrez une invite de commande
2. Déplacez dans votre repertoire personnel.
3. Créez le repertoire `NFP136` en tapant la commande :
`amelie@toto: > mkdir NFP136`
4. Déplacez vous dans le repertoire `NFP136` en tapant la commande :
`amelie@toto: > cd NFP136`
5. Un programme Java est composé de classes qui sont sauvegardées dans des fichiers. Créez un fichier `Hello.java` dans le repertoire `NFP136` avec l'éditeur de texte `kwrite`. Tapez le code suivant de la classe `Hello` :

```
class Hello {  
    public static void main ( String [] a ) {  
        System.out.println("_Hello_world_!");  
    }  
}
```

6. Pour compiler, il faut entrer a commande suivante :

```
amelie@toto: > javac Hello . java
```

Vérifier avec que le fichier `Hello.class` a été créé en listant le contenu du répertoire courant en entrant la commande suivante :

```
amelie@toto: > ls
```

7. Pour exécuter un programme Java, vous devez indiquer la classe qui contient la méthode `main`, dans cet exemple, la classe `Hello`. Vous pouvez lancer le programme en tapant :

```
amelie@toto: > java Hello
```