

I Le shell

Shell signifie coquille en anglais, en informatique, c'est le programme que l'on trouve sur les machines. UNIX et qui permet d'interpréter les commandes.

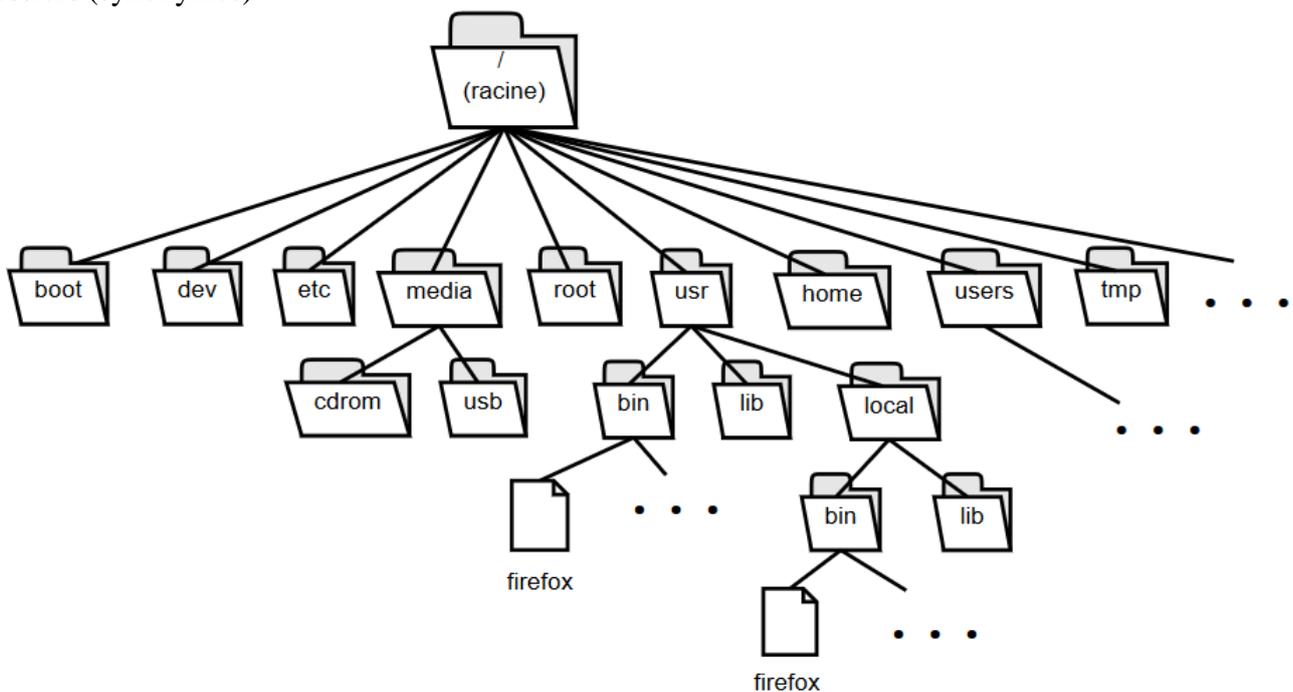
Si vous travaillez sous LINUX, il suffira d'ouvrir un terminal de commande, sous windows, vous allez devoir installer le programme : <https://www.cygwin.com/> qui vous permettra d'émuler un système Unix.

Il permet à l'utilisateur d'interagir avec la machine et de concevoir des scripts, c'est-à-dire des programmes. Nous aborderons principalement le premier point.



II Arborescence de répertoires

Tous les fichiers sont identifiables par leur nom et leur emplacement dans une arborescence de répertoires ou de dossiers (synonymes).



Pour atteindre un fichier, il faudra fournir la liste des répertoires qu'il faut traverser avant de parvenir à celui contenant le fichier. On appelle cette information le chemin (path en anglais) du fichier. Chaque répertoire d'un chemin est séparé du répertoire suivant par le caractère «/».

Il existe deux manières de donner un chemin :

- en spécifiant la liste la plus courte des répertoires à traverser *depuis la racine de la hiérarchie*, on dit alors que c'est un **chemin absolu** .

Exemple de chemin absolu sous windows:

C:\Users\Tartempion\Documents\COURS 2019-2020\NSI

Exemple de chemin absolu sous Linux : /usr/bin/firefox

- en spécifiant une liste des répertoires à traverser à partir d'un répertoire particulier de la hiérarchie, on dit dans ce cas que c'est un **chemin relatif** à ce répertoire de départ.

Exemple de chemin relatif si l'on est dans le répertoire Documents de windows :
COURS 2019-2020\NSI

Exemple de chemin relatif sous Linux quand on est dans le répertoire usr : /bin/firefox

III Exploration de l'arborescence de répertoires et création de répertoires et fichiers

Ouvrir un terminal de commandes UNIX

L'ouverture d'un terminal de commandes par défaut, se fait à la racine de votre répertoire personnel.

Les commandes de base à connaître par cœur :

Commande	Rôle	Exemple de syntaxe
ls list	Visualiser le contenu d'un répertoire	ls
ls -a ou ls -al list all	Visualiser tous les fichiers d'un répertoire, y compris les fichiers cachés	
cd .. change directory	(change directory) Permet de remonter dans l'arborescence des répertoires	cd ..
cd	Se déplacer dans un répertoire	cd nouveaurep
mkdir Make a diectory	(make directory) Permet de créer un nouveau répertoire dont on donnera nom	mkdir nouveaurep
pwd print working directory	affiche le chemin d'accès vers le répertoire où se situe l'utilisateur qui a entré la commande	pwd
touch	Crée un fichier texte ou met à jour la date et l'heure d'accès d'un fichier	touch nouv
cp copy	Copie un fichier dans un autre répertoire	cp /chemin absolu du fichier chemin absolu du rep2/
	Crée une copie d'un fichier dans son répertoire d'origine	cp fich copiefich créé une copie du fichier fich appelée copiefich
mv move	Renomme un fichier	mv fichier nouvFichier
rm remove	Supprime un fichier	rm fichier
rmdir	Supprime un répertoire vide	rmdir rep2
echo	Permet d'écrire du texte dans un fichier existant	echo « voici le texte »> mon_fichier
cat	Permet de visualiser et lire le contenu d'un fichier	cat mon_fichier
tree	Permet de visualiser le contenu d'un répertoire sous forme d'arborescence (pas natif sous cygwin)	tree

Pour davantage de détails : https://doc.ubuntu-fr.org/tutoriel/console_commandes_de_base

TD A FAIRE

1. Dans l'interpréteur, à l'aide des commandes, créer à la racine un répertoire nommé TDsystemExploit, qui contiendra les répertoires sem1 et sem2. Le répertoire sem1 contiendra les répertoires td1 et td2. Dans td1, créer deux fichiers fich1 et fich2 et dans td2, créer trois répertoires rep1, rep2 et rep3 créer un fichier fich3 dans rep1.
2. Ecrire dans fich1 le texte « Contenu du fichier 1 »
3. Créer un graphe représentant cette arborescence de tous ces répertoires et fichiers.
4. Indiquer le chemin absolu de fich1.
5. Indiquer le chemin relatif de fich3, lorsque l'on se trouve dans td2.
6. Copier dans sem2, le fichier fich3.

IV Gestion des droits

Les droits d'accès précisent pour chaque fichier ou chaque répertoire, qui a le droit de lire, d'exécuter (ou de le traverser dans le cas d'un répertoire) ou de le modifier.

Cette notion de droit, est indispensable dans un système multi-utilisateurs afin de préserver la confidentialité de ses documents à chaque utilisateur.

Lire les droits

Chaque fichier a plusieurs propriétés associées : le *propriétaire*, le *groupe propriétaire*, la date de dernière modification, et les *droits d'accès*. On peut examiner ces propriétés grâce à l'option `-l` de `ls`. Dans cet exemple, nous voyons les permissions standards d'un répertoire et de deux fichiers :

Comment se présentent les informations concernant les droits d'accès ?

Type	Propriétaire	Groupe auquel appartient le propriétaire	Autres utilisateurs
<ul style="list-style-type: none">• d : répertoire• - : fichier	<ul style="list-style-type: none">• r : droit de le lire le fichier ou d'examiner le répertoire• w : droit d'écrire dans le fichier ou d'ajouter un répertoire dans un répertoire• x : droit d'exécuter le fichier		
d	rwX	r-X	r- -
-	rw-	r- -	r- -

r,w,x n'ont pas la même signification suivant qu'elles s'appliquent à un fichier ou un dossier :

Pour les fichiers

- Lecture (noté **r**) : on peut par exemple lire le fichier avec un logiciel.
- Écriture (noté **w**) : on peut modifier le fichier et le vider de son contenu.
- Exécution (noté **x**) : on peut exécuter le fichier s'il est prévu pour, c'est-à-dire si c'est un fichier exécutable (script, programme).

Pour les dossiers

- Lecture (noté **r**) : il autorise l'affichage du contenu du répertoire (la liste des fichiers présents à la racine de ce répertoire).
- Écriture (noté **w**) : il autorise la création, la suppression et le changement de nom des fichiers qu'il contient, quels que soient les droits d'accès des fichiers de ce répertoire (même s'ils ne possèdent pas eux-mêmes le droit en écriture). Néanmoins le droit spécial *sticky bit* permet de passer outre ce comportement.
- Exécution (noté **x**) : il autorise l'accès (le traverser) au répertoire.

Les commandes de base à connaître pour modifier les droits d'accès

Il existe 4 commandes liées aux droits d'accès :

- `chmod` (*change mode*), la plus utile, qui modifie les permissions
- `chgrp` (*change group*) qui modifie le groupe d'un fichier ou répertoire
- `chown` (*change owner*) qui modifie le propriétaire d'un fichier ou répertoire
- `umask`, qui définit les protections par défaut

Les commandes `chgrp` et `chown` servent surtout à l'administrateur du système, nous ne les examinerons pas ici. En revanche, **`chmod`** et **`ls -l`** seront les deux commandes que vous devrez utiliser et connaître.

Notation symbolique

Cette notation consiste à associer un ou plusieurs degrés de propriété, une action et un ou plusieurs types d'accès. On l'utilise généralement pour ajouter ou retirer des permissions par rapport à la normale

(-rw-r--r-- pour un fichier, drwxr-xr-x pour un répertoire).

Degré de propriété	Action	Type d'accès
u (user-utilisateur)	+ (ajoute le droit)	r (read-lecture)
g (group-groupe)	- (enlève le droit)	w (write-écriture)
o (others-autres)	= (définit le droit)	x (execute-exécution)
a (all-tout le monde)		

Quelques exemples :

```
$ ls -l
total 2
-rw-r--r-- 1 Duquenne Aucun 0 22 déc. 21:38 apt-cyg
drw-rw--w+ 1 Duquenne Aucun 0 2 avr. 19:45 NSI1
```

```
$ chmod a+rx NSI1 Ajout des droits rx à tous les utilisateurs sur NSI1
```

```
$ ls -l
total 2
-rw-r--r-- 1 Duquenne Aucun 0 22 déc. 21:38 apt-cyg
drwxrwxrwx+ 1 Duquenne Aucun 0 2 avr. 19:45 NSI1
```

```
$ chmod o-rwx NSI1 Retrait des droits rwx aux autres utilisateurs sur NSI1
```

```
$ ls -l
total 2
-rw-r--r-- 1 Duquenne Aucun 0 22 déc. 21:38 apt-cyg
drwxrwx---+ 1 Duquenne Aucun 0 2 avr. 19:45 NSI1
```

D'autres ici : <http://www.tuteurs.ens.fr/unix/droits.html>

Pour aller plus loin : notation numérique

La notation numérique permet via le passage à un codage binaire de représenter un bloc de permissions, sous forme d'un triplet de chiffres compris entre 0 et 7.

Exemple : le bloc rwx r - - r - sera codé en binaire 111 100 100 et lui-même par 7 4 4

Chaque lettre du bloc est codée par 1 et chaque - par 0, puis chaque triplet binaire est converti en base 10.

Exercice : compléter ces tableaux de correspondance

1)

Accès	---	--x	-w-					
binaire				011	100	101		
décimal							6	7

2)

Notation symbolique des droits	Notation numérique	signification
rwXrwxrwx		
	755	
rwX--- ---		
rw-rw-rw		
	644	
		lecture et écriture réservées au propriétaire.
		Aucun accès à qui que ce soit