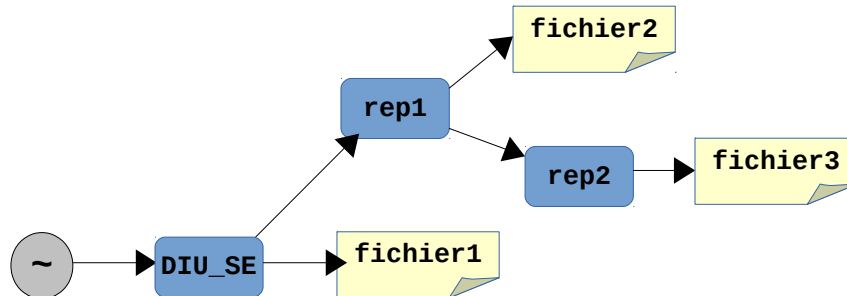


Commandes UNIX de base

Objectif : Se familiariser avec les commandes UNIX de base

Exercice 1 – Création d'arborescence

On envisage de construire l'arborescence suivante :



1) A quel répertoire correspond ~ ?

2) Dans le terminal, identifier les commandes successives à taper pour créer l'ensemble des répertoires et fichiers de l'arborescence précédente.

Exercice 2 – Gestion des droits d'accès

Dans le répertoire DIU_SE de l'exercice précédent, remarquez que les droits du répertoire rep1 sont par défaut à 755 (rwxr-xr-x). Quelles sont les commandes (en notation symbolique et en base 8) pour lui donner les droits suivants (on suppose qu'après chaque commande on remet le répertoire à 755) :

	<i>Propriétaire</i>			<i>Groupe</i>			<i>Les autres</i>		
	Lecture	Ecriture	Accès	Lecture	Ecriture	Accès	Lecture	Ecriture	Accès
Commande 1	oui	oui	oui	oui	non	oui	non	non	oui
Commande 2	oui	non	oui	non	oui	non	non	non	oui
Commande 3	non	oui	non	non	non	oui	oui	non	non
Commande 4	non	non	oui	oui	non	oui	non	non	non

Exercice 3 – La commande ls

Placez-vous dans votre répertoire d'accueil et précisez les options à utiliser pour la commande `ls` dans les cas listés ci-dessous. Pour connaître la liste des options, consultez le manuel en ligne en tapant « `man ls` ».

- 1) Listage incluant les fichiers cachés ou ceux qui commencent par ".". On remarquera la présence des 2 fichiers "." et "..".
- 2) Listage avec descriptif complet de chaque référence : droits, nombres de liens, dates, taille user group, etc.
- 3) Créez des sous-répertoires et faites un listage récursif du contenu de tous ces sous-répertoires.
- 4) Listage par ordre chronologique, et inverse.
- 5) Listage simple du contenu du répertoire, avec spécification du type de fichier : répertoire /, lien symbolique @, exécutable * .
- 6) Quelles informations complémentaires vous fournit la commande `file` ?

Exercice 4 – Visualisation de processus

Pour voir quels processus tournent sur une machine à un moment donné, il faut utiliser la commande `ps`.

- 1) Ouvrir deux terminaux. Dans le premier terminal, lancer 2 applications, par exemple *firefox* et *gedit* à l'aide des commandes `firefox &` puis `gedit &`. Dans le deuxième terminal, tapez la commande `ps`.

Que se passe-t-il ? Pourquoi *firefox* et *gedit* n'apparaissent-ils pas dans la liste ?
Quelle option utiliser avec `ps` pour les voir ?

- 2) Utilisez la commande `ps` pour déterminer le PID (*Process ID*) du *firefox* que vous avez lancé. Tapez `kill -9 lepiddefirefox`.

Que se passe-t-il ? Déterminez le PID de l'un des processus *bash* et arrêtez-le à l'aide de la commande `kill -9 lepiddebash`. Pourquoi la fenêtre du terminal disparaît-elle ?

- 3) Tapez `firefox` dans le premier terminal.

Pouvez-vous exécuter d'autres commandes dans ce terminal ? Pourquoi ? Faites un `Ctrl-C`.
Quel processus a été tué ?