TD - Découverte de Linux

1. Préliminaires
* Installer WSL.

Dans la barre de recherche de windows, taper « panneau de configuration » et sélectionner « Programmes et fonctionnalités ».

Dans la partie gauche de la fenêtre : « Activer ou désactiver des fonctionnalités Windows ». Cocher « Sous-système Windows pour Linux ». Il est possible qu’il vous soit demandé une confirmation

* Télécharger Ubuntu sur le Windows Store.
* Puis initialiser la distribution. Pour cela, lancer Ubuntu, le premier lancement prend un peu de temps. Créer un compte en donnant nom et mot de passe.

Je vous conseille de mettre un mot de passe très simple, du style « linux », car vous risquez de ne pas vous en servir très souvent.

*Remarque 0*: en cas d’erreur, voir <https://learn.microsoft.com/fr-fr/windows/wsl/troubleshooting> (détails ci après pour le problème particulier du passage de Windows 10 à Windows 11)

*Remarque 1*: si vous êtes passés de Windows 10 à à Windows 11, il peut y avoir une erreur lors de l’installation (**erreur 0x80070003 ou l’erreur 0x80370102), il faut activer la virtualisation dans le BIOS. Vous pouvez essayer ce lien :** <https://support.hp.com/fr-fr/document/ish_5637148-5698278-16> **pour les ordinateurs HP. Sinon sur un moteur de recherche, tapez « activer virtualisation bios marque\_ordinateur » ou mieux, en anglais « activer virtualisation bios ». Vous pouvez aussi consulter** <https://support.microsoft.com/fr-fr/windows/activer-la-virtualisation-sur-windows-11-pc-c5578302-6e43-4b4b-a449-8ced115f58e1>**.**

*Remarque 2*: WSL donne souvent lieu à des erreurs difficiles à corriger lors du lancement, qui devient impossible. Dans ce cas, vous pouvez faire une bonne partie de ce TD (partie 2 ci-dessous jusqu’à « Supprimer avec la commande : rm *mon\_fichier*3.txt », questions 3a et 3b, et tester les solutions de vos exercices) de ce TD sur un simulateur en ligne. par contre vous ne pourrez pas faire les trucs rigolos de ce TD !

Trois possibilités parmi d’autres :

* <https://bellard.org/jslinux/> puis Fedora 33 Console (startup link)
* <http://gervaisprof.free.fr/weblinux/> (démarrage un peu lent)
* <https://www.webminal.org/> gratuit le 1er mois. L’inscription avec un compte permet de conserver son travail.



1. Premières commandes

*Remarque préliminaire importante*: chaque fois qu’une commande est refusée/ne semble rien faire, rajouter sudo devant et rentrer éventuellement son mot de passe. Cela signifie substitute user do, et non super utilisateur do, ce dont on a l’impression ceci dit.

Exemple de l’efficacité de sudo (tiré de XKCD) :

*Remarque bis* : si vous êtes bloqués sur un écran, essayez ctrl-c ou q. Si ça ne marche pas, fermez la fenêtre…

*Remarque ter* : toutes les commandes sont des abréviations faciles à retenir, cf. le cours.

**Exécuter les commandes suivantes**

Tester la commande « list », qui donne le contenu d’un répertoire : ls puis ls – a et ls –l et enfin ls -a –l ls -al

Créer un dossier avec la commande : mkdir *mon\_dossier*

Tester à nouveau ls, ls – a, ls –l, ls -a –l

Lire la partie du cours sur les permissions et le détail de ls –l.

Aller dans le dossier avec la commande : cd *mon\_dossier*

Créer un fichier avec la commande : touch *mon\_fichier*.txt

Editer le fichier avec avec la commande : vim *mon\_fichier*.txt (vim est un éditeur de texte)

 Aide de vim :help (les : font partie de la commande)

 Rentrer du texte touche insertion puis écrire

 Quitter avec escape puis pour sauver :x

Afficher le contenu du fichier avec la commande : more *mon\_fichier*.txt

Changer les autorisations de *mon\_fichier*.txt :

* relire la partie du cours sur les autorisations/permissions !
* puis exécuter, après l’avoir comprise, la commande :

sudo chmod u-w mon\_fichier.txt

Éditer à nouveau le fichier avec vim *mon\_fichier*.txt

 Rentrer du texte touche insertion puis écrire, que constatez-vous ?

 Quitter avec escape puis, sans sauver avec :q , en sauvant :x , et finalement en force avec « :qa! »

Copier le fichier avec la commande : cp *mon\_fichier*2.txt *mon\_fichier*3.txt

Changer le nom avec la commande : mv *mon\_fichier*2.txt *mon\_fichier*3.txt

Supprimer avec la commande : rm *mon\_fichier*3.txt (que constatez-vous ?)

Aller dans son dossier Documents étape par étape, vous pouvez faire des ls – a pour explorer le contenu des dossiers. Commandes :

cd / cd mnt/ cd c/ puis pour aller plus vite cd Users/fmandon/Documents/

Vous aurez bien sûr habilement remplacé fmandon par votre nom d’utilisateur Windows.

Essayer cd .. (remonter au répertoire parent « .. = papa »)

Essayer cd~

Retourner directement dans votre dossier Documents de Windows

Utiliser la commande pwd pour savoir où on est (print working directory)

Recopier un fichier du dossier documents vers *mon\_dossier*, commande : cp *mon\_fichier* /*mon\_dossier*/ *mon\_fichier*

cd / revient à la racine du disque puis ls

Revenir au dossier racine linux cd~

Essayer rmdir *mon\_dossier* pour supprimer ce qu’on vient de faire. Comme ça ne marche pas, alors

rm –r *mon\_dossier*

1. Compléments
	1. *Les opérateurs de redirection et le « pipe »*

Les opérateurs > et >> permettent d’écrire les sorties d’une commande non pas à l’écran, mais dans un fichier. Placez-vous dans un répertoire non vide et testez ls –al > liste\_fichier.log. Vous pouvez ensuite ouvrir le fichier crée avec vim, ou simplement l’afficher avec less. L’opérateur > crée le fichier de sortie, éventuellement en écrasant un fichier de même nom préalablement existant. L’opérateur >> écrit à la fin du fichier sans l’écraser.

L’opérateur | permet d’enchaîner les commandes. Par exemple, ps –e (ou aux) permet de lister tous les processus (cf. cours de terminale… on va dire que ce sont les programmes) tournant sur le noyau Linux. La commande wc permet de compter (word count). wc –l compte les lignes (-c les octets, -w les mots…). Pour compter le nombre de processus, on peut faire ps aux | wc –l

* 1. *Aller sur internet.*

On n’ouvre pas un navigateur, seuls de rares adresses peuvent être lisibles/accessibles, comme celle-ci pour la météo : curl wttr.in/montpellier. La commande curl (client url) est puissante, elle permet notamment de récupérer l’en-tête, le code html etc. d’une page (cf. challenges root-me).

* 1. *Installer des programmes*

Exemple : installation de cowsay

sudo apt-get update

sudo apt-get install cowsay

Puis test du programme en tapant

 cowsay hello beau(belle) gosse

Options à tester après cowsay et avant le texte :

–bdgsty (une seule lettre)

-e 88

-T U

-f *type\_de\_vache* (par exemple dragon)

Exemple : installation de fortunes

 sudo apt-get install fortunes fortunes-fr

Puis test de fortunes avec la commande fortune (parfois très geek et incompréhensible, souvent drôle)

Linux permet d’enchaîner des instructions, en envoyant le résultat d’une commande dans une autre. On appelle cela un tuyau (pipe en anglais). Essayer :

 fortune /usr/share/games/fortunes/fr | cowsay –n

Pour avoir des informations sur son système, le programme inxi est très complet (sudo apt-get install inxi). Plein d’options possibles.

* 1. *Écrire un script de démarrage.*

Éditer son fichier de configuration avec vim. Si vous avez installé Ubuntu, il est dans votre répertoire de base et s’appelle .bashrc.

Install randomize-line puis edit .bashrc et taper en dernière ligne la commande suivante, en faisant attention aux différents types d’apostrophes.

cowsay -f `ls /usr/share/cowsay/cows |shuf -n1|cut -d'.' -f1` "`fortune -s`"

* 1. *Splitter (séparer) la fenêtre de Linux en plusieurs sous-fenêtres :*

sudo screen puis :

* ctrl a puis | pour une séparation verticale ;
* ctrl a puis S pour une séparation horizontale ;
* ctrl a puis Q pour une supprimer les séparations ;
* ctrl a puis tab pour passer d’une fenêtre à une autre.
* Pour démarrer une session dans la nouvelle fenêtre, ctrl a puis c.

Utiliser ces commandes pour séparer l’écran en deux, travailler dans une partie et lancer dans l’autre :

telnet towel.blinkenlights.nl

Telnet est un protocole de communication par texte, abandonné sauf pour faire des tests.

* 1. *Autres bêtises et amusements :*
* Ajouter ses propres citations dans fortune : <https://doc.ubuntu-fr.org/fortune>
* Installer un magnifique aquarium : voir <https://doc.ubuntu-fr.org/asciiquarium>

Les commandes (en 2022) sont :

apt-get install libcurses-perl

sudo cpan Term::Animation

cd /tmp

wget <http://www.robobunny.com/projects/asciiquarium/asciiquarium.tar.gz>

tar -zxvf asciiquarium.tar.gz

cd asciiquarium\_1.1/

sudo cp asciiquarium /usr/local/bin

sudo chmod 0755 /usr/local/bin/asciiquarium

Puis simplement exécuter : asciiquarium

* Si vous faites comme de nombreux linuxiens, et que pour lister le contenu d’un répertoire vous tapez souvent sl à la place de ls, alors essayez :

sudo apt-get install sl

exerciceS

Exercice 1.

Créer un dossier *programmes*, et dedans un fichier *bonjour.py*.

Le code du script Python peut être (ce n’est pas une obligation, vous pouvez faire plus sophistiqué si vous le souhaitez) :

from time import sleep

nom = input("Quel est votre nom ? ")

print("Bonjour ", nom)

sleep(5)

Rendre le fichier exécutable pour vous.

Vérifier quelle est la version de Python dans /usr/bin. Vous constaterez qu’un « ls » donne trop de résultats. On utilise alors ls py\* pour trouver tous les fichier dont le nom commence par py (et est suivi de n’importe quoi d’autre).

Lancer le programme avec la commande : *bonne\_version\_de\_python* bonjour.py

Exercice 2.

Créer l’arborescence suivante de dossiers et fichiers dans votre répertoire de travail.



On précisera toutes les commandes utilisées.

A l’aide d’un « pipe », lister à partir du répertoire courant tous les répertoires triés par taille décroissante. On pourra consulter le manuel pour les commandes du et sort.

Exercice 3.

Dans le répertoire ./Unix/TP2, créer deux fichiers public.txt et prive.txt.

Donner les commandes permettant de mettre les permissions demandées, quelles que soient les permissions initiales sur les répertoires ou fichiers.

1. Le répertoire personnel possède tous les droites pour l'utilisateur, et uniquement le droit d’exécution pour le groupe et les autres.
2. Les répertoires Unix et Unix/TP2 possèdent tous les droits pour l'utilisateur et les droits de lecture et d'exécution pour le groupe et les autres
3. Le fichier public.txt du répertoire Unix/TP2 possède les droits de lecture et d'écriture pour l'utilisateur et uniquement les droits de lecture pour le groupe et les autres.
4. Le fichier prive.txt du répertoire Unix/TP2 possède les droits de lecture et d'écriture pour l'utilisateur et aucun droit pour le groupe et les autres.

Exercice 4.

Sans tester ces commandes dans un terminal, dessiner ci-dessous l'arborescence finale des fichiers et des répertoires.

On utilisera TEST comme racine de l'arborescence.

1. mkdir a b c d

2. touch a/t.txt d/foo.txt

3. cd c

4. mkdir ../b/e f g

5. cd ..

6. cp \*/\*.txt c/g # le caractère spécial "\*" remplace n'importe quelle chaîne de caractère

7. rm -r f d

Cours de Frédéric Mandon, licence Creative Commons BY NC SA, <https://creativecommons.org/licenses/by-nc-sa/3.0/fr/>

Merci aux collègues de la liste NSI pour leurs idées et leurs exercices, dont je n’ai pas noté la provenance :-/ .