

TP commandes Linux

L'ensemble de ce TP va se focaliser sur les commandes Linux à exécuter dans un shell (similaire à la console Python, sauf que les commandes permettent d'effectuer des actions directement sur le système d'exploitation).

Vous allez manipuler les commandes sur le site suivant :

<https://bellard.org/jslinux/vm.html?url=alpine-x86.cfg&mem=192>

Les commandes Linux

Les commandes s'écrivent directement dans le shell et peuvent prendre des paramètres (un fichier, un dossier, etc.), et peuvent aussi avoir des options.

Voici un tableau non exhaustif comportant les commandes principales : ce qu'elles font, leurs arguments, et leurs options.

commande	descriptif	argument(s)	option(s)
pwd	Affiche le chemin du répertoire d'où l'on se trouve		
ls	Affiche les fichiers et dossiers du répertoire courant		-l : affiche le nom, le type, les autorisations d'accès
cd	Permet de changer de répertoire	chemin, .. (répertoire parent), / (répertoire racine)	
touch	Crée des fichiers	touch nomFichier	
cp	Copie un fichier	cp nomFichier nouveauFichier	* : copie tous les fichiers d'un répertoire
mv	Déplace un fichier dans un autre répertoire	mv source destination	* déplace tous les fichiers d'un répertoire
mkdir	Crée un répertoire	mkdir repertoire	
rmdir	Supprime un répertoire	rmdir repertoire	
rm	Supprime un fichier	rm fichier	-i : demande confirmation à l'utilisateur, -f : force sans confirmation, -r : supprime les sous répertoires
echo	Affiche ce qu'on écrit après	echo "bonjour"	
ps	Affiche les processus en cours		
nano	Permet d'éditer un fichier pour le modifier	nano nomFichier	

Exercices

Gestion des répertoires

À partir du menu, ouvrez un terminal.

En utilisant les commandes adéquates, répondez aux questions suivantes (en notant les commandes utilisées et les réponses dans la zone prévue à cet effet).

1. Dans quel répertoire vous trouvez-vous ?
2. Comment s'appellent les fichiers présents dans le répertoire ?
3. Créer un répertoire qui s'appelle `pilier`
4. Se déplacer dans le dossier `pilier`
5. créer les fichiers `Tengen.py` et `Rengoku.py`
6. Supprimer le fichier `Rengoku.py`
7. Éditer le fichier `Tengen.py`
8. Nous éditons un fichier Python. Nous pouvons donc écrire du code Python dedans.
Écrire le code qui permet d'afficher "Je suis Tengen Uzui, pilier du Son de l'Armée des Pourfendeurs de démons".

Pour enregistrer le fichier, il faut appuyer sur CTRL + X. On vous dit ensuite "Save modified buffer ?", appuyer sur la touche Y, puis sur la touche Entrée.
9. On peut exécuter du code Python directement dans le shell. Écrire `python3` suivi du nom de votre fichier Python, et appuyer sur Entrée.
10. Déplacer le fichier `Tengen.py` en dehors du dossier `pilier`
11. Supprimer le répertoire `pilier`
12. Créer un répertoire `pourfendeur`, et dans le répertoire `pourfendeur`, créer le répertoire `eau`
13. Se placer dans le répertoire `eau`, et créer le fichier `tanjiro.py`. Vérifier que le fichier est bien créé.
14. N'utiliser qu'une seule commande pour revenir au répertoire `root` (le tout premier = la racine).
15. Supprimer `pourfendeur` ainsi que tous ses sous-répertoires

Gestion des processus

16. Afficher les processus en cours à l'aide de la commande adéquate (une sorte de table apparaît à l'écran)

Écrire les en-têtes de la table :

17. Les PID représentent l'ordre chronologique de création des processus sur la machine. Plus leur numéro est grand, plus le processus est récent.

Déterminer quel est le tout premier processus et à quoi correspond sa commande.

18. Pourquoi y-a-t-il des "trous" dans la numérotation des PID ?

19. La plupart des processus affichés sont nécessaires au bon fonctionnement de Linux. Dans la liste, repérer le nom de votre terminal et noter son PID.

20. Trouver le dernier processus dans la liste. À quoi correspond-t-il ?