
TP N°6 : Les processus

Objectifs	: <ol style="list-style-type: none"> 1. Rediriger l'entrée standard, la sortie standard et la sortie standard des erreurs 2. Enchaîner plusieurs commandes avec les pipes 3. Savoir contrôler l'exécution d'un processus 3. Savoir lancer un processus en avant plan, en arrière plan, en détaché du terminal 4. Savoir tuer un processus
Volume horaire	: 3 heures

I. Partie théorique

Répondre aux questions suivantes.

1. Qu'est-ce qu'un processus ?
2. Quels sont les états possibles d'un processus ?
3. Comment peut-on exécuter un processus en arrière plan ?

II. Manipulations pratiques

Exercice 1 :

1. Ecrire une seule ligne de commande permettant d'afficher la date, la liste des utilisateurs. et le nom du répertoire de travail.
2. Reprendre la question 1) en envoyant le résultat dans un fichier Temp
3. Ecrire une commande qui permet de compter le nombre de fichiers et de répertoires dans le répertoire de travail.
4. Ajouter au fichier Temp la liste de fichiers et de répertoires de votre répertoire de travail ainsi que leur nombre.
5. Interpréter les commandes suivantes

```
$ls | wc-l
```

```
$ ls -i | head -5 | tail -1
```

```
$ ls -i | tail -4 | head -2
```

6. Créez deux fichiers activite1 et activite2 contenant chacun votre position dans l'arborescence (pwd), votre nom de connexion (whoami) ainsi que votre identité complète (id) en respectant les consignes ci-après :

Pour créer le fichier activite1, commencez par créer trois fichiers séparés en n'utilisant que des redirections simples (>), puis les concaténer tous les trois avec une nouvelle redirection simple pour créer ce fichier activite1.

Pour créer le fichier activite2, utilisez les redirections en mode ajout (>>) pour créer le fichier et le compléter au fur et à mesure.

Vérifiez à l'aide d'une commande de comparaison (diff et/ou cmp et/ou comm) que ces 2

fichiers sont identiques.

7. Créez, par une redirection, un fichier liste contenant la liste (au format long) des entrées du répertoire courant.
8. Visualisez ce fichier pour constater que le fichier liste apparaît dans cette liste comme un fichier vide alors que maintenant il a une taille non nulle.
9. Quelle est la différence entre tee et cat ?
tee - Copier l'entrée standard sur la sortie standard et dans un fichier.
cat - Concaténer des fichiers et les afficher sur la sortie standard.
10. Que font les commandes suivantes :

```
$ ls | cat
```

```
$ ls -l | cat > liste
```

```
$ ls -l | tee liste
```

```
$ ls -l | tee liste | wc -l
```

Exercice 2 : - Étude des processus Linux

1. Etudiez la commande ps avec les options suivantes : -l, -a, -u, -x
2. À l'aide de la commande **ps**, afficher la liste de tous les processus tournant sur votre machine, avec les informations suivantes :

USER	nom de l'utilisateur propriétaire du processus
PID	numéro d'identification du processus
%CPU	
%MEM	
STAT	Etat
START	Date de début
TIME	
COMMAND	Commande utilisée pour lancer ce processus

(vous vous aiderez du manuel (man ps) et du résumé (ps --help)).

- 1.1 A quoi correspond l'information TIME ?
- 1.2 Quel est le processus ayant le plus utilisé le processeur sur votre machine ?
- 1.3 Quel a été le premier processus lancé après le démarrage du système ?
- 1.4 A quelle heure votre machine a-t-elle démarré ?
- 1.5 Pouvez-vous établir le nombre approximatif de processus créés depuis le démarrage ("boot") de votre machine ?
- 2- Sous Linux, chaque processus (excepté le premier) est créé par un autre processus, son processus père. Le processus père d'un processus est identifié par son PPID (Parent PID).
 - 2.1 Trouver une option de la commande ps permettant d'afficher le PPID d'un processus.
 - 2.2 Donner la liste ordonnée de tous les processus ancêtres de la commande ps en cours d'exécution.
- 3- Reprendre la question précédente avec la commande pstree.
- 4- Essayez la commande top, qui affiche les mêmes informations que ps mais en rafraîchissant périodiquement l'affichage.
 - 4.1 Quel est le plus "gros" processus sur votre machine ? A quoi correspond-il ? (Rappel: vous pouvez utiliser man truc pour découvrir ce que fait truc...).
 - 4.2 Quel est le processus sur votre machine qui utilise le plus de mémoire ?
 - 4.3 Quel est le processus sur votre machine qui utilise le plus le CPU ?

4.4 Quel est le processus sur votre machine qui a utilisé le plus de temps CPU ?

Exercice 3 : - Arrêt d'un processus.

- 1-Créer dans votre répertoire de connexion un nouveau répertoire nommé exo2.
- 2-Lancer l'exécution du programme top et vérifier qu'il fonctionne. L'arrêter en tapant CTRL-C.
- 3-En utilisant les fonctionnalités du shell (&, fg, bg), lancer quatre instances du programme top en même temps. Mettre au premier plan la troisième, l'arrêter (CTRL-Z) puis la relancer en arrière plan.
- 4-A l'aide des commandes jobs et kill %n, arrêter tous les compteurs.
- 5-Même question en utilisant les commandes ps et kill (avec un PID).
- 6-Ouvrez une autre session et lancer une commande dont l'exécution dure longtemps (par exemple cat sans paramètres).
- 7-A partir de la première session, déterminer le numéro du processus (PID) correspondant à la commande lancée. Arrêter ce processus avec la commande Kill -9 PID ps. Interprétez les résultats.
- 8-Réfaire la question 5, en utilisant les commandes : kill -15 PID, puis kill -2 PID.
- 9-Ecrire le script suivant (bonjour) avec un éditeur, puis l'ajouter le droit d'exécution.

```
$Cat > bonjour
While true
Do
    Echo bonjour
Sleep 30
Ctrl-D
$
```

Lancer ce script en arrière plan : `$./bonjour&`

- 10-Tuer ce processus en utilisant son PID ou son numéro de processus.
- 11-Lancer de nouveau ce script en détaché du terminal (insensible à la déconnexion), où écrit-il ses sorties ?
- 12-Déconnectez-vous, et connectez-vous de nouveau. Affichez vos processus en tapant : `$ ps` Le processus bonjour n'apparaît pas, pourquoi ?
- 13-Quelle commande devez-vous exécuter pour afficher le processus qui exécute bonjour.
- 14-Tuez le processus bonjour et détruisez son fichier de sortie.

Rédigez un compte rendu décrivant ce que vous avez fait.