

## TP N° 5 : SHELL ET PROCESSUS SOUS UNIX

### 1- La commande ps

La commande `ps` permet d'obtenir la liste des processus qui tournent sur la machine. Elle dispose de beaucoup d'options qui permettent d'obtenir plus ou moins de renseignements sur ces processus. Pour tout savoir sur la commande, n'oubliez pas que vous pouvez taper `man ps`.

1. Visualisez grâce à la commande `ps`, la liste des processus concernant le shell dans lequel vous la tapez.
2. Visualisez, grâce à la commande `ps -u votrenom`, la liste des processus vous appartenant.
3. Visualisez avec la commande `ps -ef`, tous les processus tournant dans la machine.
4. Il existe une commande `wc` qui compte le nombre des éléments qu'on lui passe. En redirigeant la commande `ps -ef` vers la commande `wc` (au moyen des tubes), obtenez le nombre total de processus présents dans la machine.
5. Utilisez `ps` avec l'option `-l` pour obtenir encore plus de renseignements sur les processus. Remarquez la colonne S qui vous donne l'état du processus. Un 0 indique que le processus est actif, un S que le processus est bloqué (Sleeping), et un R qu'il est prêt (Ready).
6. Lancez la commande `xclock` en faisant Alt+F2. En utilisant la commande `ps`, établissez la hiérarchie des processus entre `xclock` et le processus `init`.

### 2- La commande top

Tout comme la commande `ps`, la commande `top` permet d'observer les processus de la machine, mais en temps réel. On peut ainsi suivre l'évolution du cycle de vie d'un processus (actif, prêt, bloqué) ainsi que son degré d'utilisation du processeur. Par défaut la commande affiche les quinze processus les plus actifs pendant la seconde écoulée, et remet à jour cette liste toutes les secondes. Elle donne aussi le taux d'occupation moyen du processeur pendant la dernière seconde, ainsi que la mémoire utilisée dans la machine.

1. grâce à la commande `top` observez l'évolution de l'activité des processus. En particulier, essayez de manipuler très rapidement les fenêtres et donnez le nom des processus qui s'activent le plus.
2. Parmi les processus présents dans la mémoire de votre machine, combien sont respectivement bloqués, prêts et actifs ?
3. Quel est le taux moyen d'utilisation du processeur ? Expliquez le pourquoi de ce taux.
4. Quand `top` fonctionne, appuyez sur la touche `h` pour obtenir de l'aide.
5. Classez les processus affichés en fonction de la priorité des processus.
6. Faites afficher à `top`, uniquement les processus dont vous êtes le propriétaire.

7. Lancez en parallèle de la commande `top` des applications supposées gourmandes en temps de calcul. Observez l'activité du processeur.

### 3- Manipulation des processus

Le shell UNIX propose une commande qui permet d'envoyer des signaux aux processus. Cette commande s'appelle `kill`. Utilisée sans option elle envoie le signal `SIGINT` au processus spécifié, ce qui a pour effet par défaut de lui demander de se terminer. Par exemple `kill 1234` demande au processus numéro 1234 de se terminer. Si on veut lui faire envoyer un autre signal, on utilise la syntaxe `kill -SIGNAL NUM_PROCESS` en remplaçant `SIGNAL` soit par le nom du signal à envoyer, soit par son numéro et en remplaçant `NUM_PROCESS` par le pid du processus cible.

1. La commande `xclock` permet d'afficher l'heure de façon graphique. Lisez la documentation de cette commande pour trouver quelle est l'option qui permet d'afficher l'aiguille des secondes.
2. Tapez six fois la commande `xclock &` (avec ajoutant l'option pour afficher les secondes) de façon à lancer plusieurs fois l'application. Le symbole `&` que vous avez mis à la fin de votre commande permet aux tâches de se lancer en arrière plan (sans bloquer le shell). Essayez de lancer un dernier `xclock` sans ce caractère pour voir la différence.
3. Grâce à la commande `ps` repérez le numéro des processus correspondant à tous les processus `xclock` que vous avez lancés.
4. Avec la commande `kill` et en utilisant les numéros repérés lors de la question précédente, tuez quatre de ces pendules.
5. Essayer de tuer un processus qui ne vous appartient pas et voyez si c'est possible.
6. En utilisant la commande `kill -l`, trouvez la liste des signaux que vous pouvez envoyer aux processus.
7. Interrompez (mais ne tuez pas) la dernière horloge active à l'écran. Pour cela utilisez un des signaux de la commande `kill` (son nom contient `STOP`).
8. Faites redémarrer l'horloge que vous venez d'interrompre. Pour cela utilisez un autre signal de la commande `kill` (son nom contient `CONT`).
9. On peut aussi suspendre un programme lancé dans le shell (sans le symbole `&`) en tapant `Ctrl-Z`. Lancer une commande `xclock` et tapez `Ctrl-Z`. Pour le relancer comme précédemment : tapez `fg`. Pour le relancer en tâche de fond : tapez `bg`.
10. Essayez de tuer une horloge en lui envoyant le signal `SIGKILL` (numéro 9). Ce signal ne demande pas à la tâche de se terminer mais demande au SE de terminer la tâche. Dans quel cas cela peut-il être utile ?
11. Essayez de tuer le shell où vous tapez vos commandes et voyez le résultat.

## 4- Droits et processus

Lorsque l'on lance la commande `chsh`, le fichier `/etc/passwd` est modifié. Cela est dû au SetUID du fichier `chsh`

1. Allez voir les droits du fichier de la commande `chsh`.
2. Dans un terminal, lancez la commande `chsh` et ne mettez pas votre mot de passe. Pendant ce temps, ouvrez un autre terminal et affichez la liste des processus qui vous appartiennent (`ps -u votrenom`). Comment expliquer le résultat ?
3. Tapez la commande `ps -U votrenom`. Le processus `chsh` apparaît. Comment l'expliquer ?
4. Copiez le fichier `/usr/bin/chsh` dans votre répertoire utilisateur. Quels sont les droits du fichier ? Exécutez-le (`./chsh`). Modifiez ses permissions afin qu'il fonctionne.
5. Créez un fichier `test` dans lequel vous mettez la commande `cat /etc/shadow`. Mettez les droits SetUID à ce fichier et exécutez-le. A votre avis pourquoi cela ne marche-t-il pas ?