Exercices de TP sur le shell et la mémoire

Fabien Rico

1 Ressources d'un processus

Pour cet exercice, utilisez le code : `code.py`.

1) Lancez le programme ; pendant qu'il tourne trouvez son pid grâce à la commande (`ps auxf`).

2) Utilisez les commandes `fuser` ou `lsof` pour connaître les fichiers utilisés dans le répertoire courant.

3) Dans quel cas cela peut-il vous servir ?

4) Lorsqu'un processus est lancé vous pouvez retrouver les informations le concernant dans `/proc/< pid >/...`. Où se trouvent les informations sur les fichiers ouverts ? Le répertoire courant ? Les variables d'environnement ?

5) Lancez la commande `ls -R / | less` ; retrouvez leur pid et afficher les répertoires contenant les descripteur de fichiers : `ls -lh /proc/<pid>/fd/`. À quoi correspondent les descripteurs 0 et 1 ?

1.1 Mémoire :

Pour cet exercice, utilisez les codes : `code1.c`, `code2.c`, `code3.c` et `code4.c`. Ce sont des codes en C car python avec son interpreteur ajoute trops de chose pour que l'étude de la mémoire soit clair.

Pour compiler ces codes, vous pouvez utiliser `gcc` grâce à la commande suivante :

```
gcc -g -Wall -O0 nomdufichier.c -o nomexecutable
```

— Grâce à la commande `pmap`, affichez les segments utilisés par les programmes `code1` et `code2`. Quelles sont les différences ? À quoi correspondent-elles ?
— Faites de même pour les codes code3 et code4. Que devient le tableau tab ?
— Donnez pour `code2` le contenu des différents segments.
— Peut-on retrouver ces informations dans le répertoire `/proc/< pid >/` ?

1.2 Variables d'environnement

Nous allons voir les variables d'environnement et leur effet à travers la notion d'environnement virtuel python. Ces environements permettent de définir un ensemble de dossiers contenant tout le necessaire au fonctionnement d'un projet avec les fichiers python, mais aussi les librairie nécessaire et en dehors de celle du système hote.

Pour gérer un envirronement vous disposez des commandes suivantes :

```
# creation de l'envirronement
python3 -m venv ./nomDuRepertoireDeLEnv
# "entrée dans l'environnement (en bash)
source <repEnv>/bin/activate
# "sortie de l'environnement
deactivate
```

1) Créez l'environnement `testTP`

2) Dans ce répertoire, que contient le sous répertoire `./lib/python???/` le répertoire `./bin/`

3) Ouvrez le fichier `./bin/activate`, que fait ce code pour l'activation, la desactivation ?

4) Entrez dans l'environnement et comparez les variables `PATH`, `PYTHONHOME` `VIRTUAL_ENV` et `PS1` avec un shell qui n'y est pas. A quoi servent ces 3 variables ?

5) Installer la librairie `pandas` grâce à `pip`. Quel est le résultat dans le répertoire `./lib/` ?

6) Lancer un processus python et vérifiez que le chemin de recherche des bibliothèque est bien modifié par l'environnement (le contenur de `sys.path`)

7) Comment lancer un processus python dans l'environnement sans entrer dedans ?