IN329 : TP Linux

SI

http://personnel.supaero.fr/garion-christophe/IN329

Ce TP va vous permettre de découvrir rapidement le noyau Linux, en particulier en traçant les appels système sur une commande simple.

---

1 Le pseudo-système de fichier /proc

Ce pseudo-système de fichier contient les données manipulées par le noyau. En particulier, chaque sous-répertoire représente un processus identifié par son PID.

On peut identifier les régions virtuelles d'un processus via le fichier `maps` présent dans la hiérachie. Par exemple :

```
cat /proc/[self]/maps
```

nous donnera une représentation de la mémoire virtuelle utilisée par la commande `cat` lancée.

Explorer la hiérarchie pour découvrir des informations sur le système (CPU, systèmes de fichiers compilés dans le noyau etc).

2 Tracer les appels système sur une commande

Nous allons tracer les appels système qui se produisent lors de l'appel d'une commande simple, `cat`.

Dans un premier temps :

- ouvrir un terminal
- exécuter la commande suivante dans ce terminal : `echo $$`. Cette commande renvoie le PID du shell. Conserver ce PID.
- ouvrir un deuxième terminal
- dans ce deuxième terminal, exécuter la commande suivante : `strace -o trace-cat.txt -f -p PID` où PID est l'identifiant noté précédemment.
- dans le premier terminal, exécuter `cat /etc/resolv.conf`
- dans le second terminal, tuer `strace` avec Control-C

On dispose maintenant d'un fichier `trace-cat.txt` qui contient une trace des appels système lors de l'exécution de la commande `cat`.

Nous allons essayer d'identifier dans ce fichier (on pourra utiliser `grep` pour récupérer les lignes correspondant à l'exécution de `cat` via son PID) :

- les appels correspondant à la saisie de la commande
- la préparation au lancement de `cat` (le PATH etc)
- la création du processus fils (`fork` ou `clone`), la mise en attente du shell
- initialisation de `cat` et le chargement de l'exécutable en mémoire (voir en particulier le rapport avec la méthode `main` de l'exécutable)
- le chargement de la bibliothèque `libc` : ouverture du fichier, mapping de la bibliothèque dans la mémoire
- lecture du fichier `/etc/resolv.conf`, affiche du contenu et fermeture du fichier
- reprise du shell