



IUT, département Informatique
2ème année
Année Universitaire 2012-2013

Programmation système et réseaux

TP n° 1 Fork et exec

Durée 1 semaine

Objectifs :

Le but de ce TP est de faire une utilisation de `fork` et `wait`, puis de faire une utilisation combinée de `fork` et `exe`. On observera l'état des processus de l'utilisateur donné par la commande `ps`, y compris lors d'une utilisation du programme `suid`.

Note. Chaque semaine, le listing commenté du TP est à rendre en début de la séance de la semaine suivante.

Exercice 1 a) Écrire un programme avec un processus père qui lance n processus fils, le nombre n étant saisi au clavier. On vérifiera lors de la saisie que le nombre n est inférieur à 10.

Chaque processus fils aura un numéro entre 1 et n . Chaque processus fils doit attendre un nombre aléatoire de secondes, le nombre de secondes étant compris entre 1 et 20. Le processus fils commencera par afficher son numéro, son *PID*, et sa durée d'attente, et se terminera en affichant son numéro, et son *PID*. Le processus père affichera la durée totale de l'opération avant de se terminer.

b) Observez l'état des processus de l'utilisateur lors du déroulement du programme.

c) Modifiez le programme pour que chaque processus fils F se duplique en deux processus qui chacun attend une durée aléatoire entre 1 et 5. Le fils F affiche la durée totale de ses deux fils.

d) Observez l'état des processus de l'utilisateur lors du déroulement du programme.

e) Modifier le programme du *a)* pour qu'il enregistre le login de l'utilisateur et l'état des processus de l'utilisateur dans un fichier (dont le nom est passé en argument du programme) lors de la terminaison de chaque fils. On fera un programme qui puisse être utilisé de manière sûre dans une application *Set-UID*. Pour écrire dans le fichier, on utilisera des descripteurs de fichiers et la fonction `open`, puis on redirigera la sortie `stdout` (de descripteur `STDOUT_FILENO`) vers notre descripteur de fichier en utilisant la fonction `dup2`.

f) Même question qu'au *e)* mais pour le programme du *c)*.

g) Modifiez le programme pour que le processus père ouvre le fichier texte généré dans *emacs* une fois que tous ses fils sont terminés.

h) Lorsque le programme est au point, rajouter les droits d'exécution *Set-UID*, et arrangez-vous avec un autre binôme pour tester l'exécution du programme à partir d'un autre compte

d'utilisateur. Décrivez ce que vous observez sur le fichier des processus que génère le programme dans un commentaire à la fin du listing.