

Rapport projet unix

L'application client/serveur développé dans le cadre ce projet est composé de 3 scripts qui sont respectivement : `surveille`, `serveurSurveille` et `surevents`.

Le script `surveille` est chargé de lancer l'exécution du script `serveurSurveille` qui contient le code serveur et le script `surevents` correspond au client et permet d'afficher les résultats de l'inspection effectué par le serveur.

Les trois scripts doivent se situer dans le même répertoire de travail.

Les syntaxes d'utilisation sont les suivantes

- `surveille start [-p pid] [-u user] [-d x]` → démarre le serveur
- `surveille stop` → stoppe le serveur
- `surveille restart` → stoppe le serveur et le redémarre avec les mêmes paramètres initiaux.
- `surevents` → affiche les résultats de l'activité du serveur.

Le script `surveille`

Ce script contient la fonction `startServeur` qui permet de lancer l'exécution du serveur en fonction des options de la syntaxe d'utilisation. Cette fonction `startServeur` lance l'exécution du script `serveurSurveille` en arrière-plan qui contient l'implémentation du serveur

Le script `surveille` crée le fichier `journal.txt` dans lequel seront écrits les événements rencontrés, il crée le fichier `restart.txt` pour la sauvegarde des options nécessaires pour relancer le serveur avec les mêmes paramètres initiaux. Le script continue avec la commande `switch case` qui exécute différentes commandes selon que l'on lance, arrête ou relance le serveur.

Le Script `serveurSurveille`

Ce script contient les fonctions `displaySuspectProcess`, `displaySuspectOpenFiles` et `surveilleProcessAndFiles`.

La fonction `surveilleProcessAndFiles` est juste des combinaisons de lancements des deux autres en fonction de la syntaxe d'utilisation, des options et arguments.

La fonction `displaySuspectProcess` reçoit en entrée une liste de processus et renvoie en sortie les processus suspects selon les critères spécifiés le cahier des charges

La fonction `displaySuspectOpenFiles` reçoit en entrée une liste de fichiers et renvoie en sortie les fichiers ouverts suspects selon les critères spécifiés dans le cahier des charges.

Le programme principal serveur dans ce script exécute la fonction `surveilleProcessAndFiles` dans une boucle infinie.

Le script surevent est le client de l'application. Une fois lancé il va lire les informations dans le fichier journal.txt et l'affiche à l'écran.

NB : ce qui est très important à noter est qu'avant de lancer surevent il faut impérativement ouvrir le fichier journal.txt en lecture avec le descripteur de fichier 3 et le maintenir ouvert en lecture. Ceci se fait en exécutant la commande suivante dans le Bash dans lequel on souhaite exécuter le script surevent :

exec 3 < journal.txt

ainsi à chaque exécution du script surevent, le script affichera uniquement les nouvelles lignes écrites dans le fichier journal.txt par le serveur.

Il est aussi important de noter qu'il faut fermer la lecture du fichier journal.txt juste avant ou juste après avoir arrêté le serveur en exécutant la commande suivante :

exec 3<&-

toutes les commandes utilisées ont été vues en cours.