Compte rendu SAE 15 TP 1

HARTMANN Matthias, 3A

Sommaire

Partie 1 : Observation des fichiers

Explication de la consigne :

Dans cette partie nous devions afficher (à l'aide d'un script bash) pour chaque fichier du répertoire Tmp (préalablement créé dans le répertoire ~/SAE15/TP1). Ces fichiers étaient nommés f1, f2, f3 et étaient vides.

Le Script Bash :

```
#! /bin/bash
path="/home/iziram/Documents/GitHub/iutCours/SAE_15/TP_1/Tmp/"
for i in $(ls $path)
      do
            file=$path$i
            if test -f $file;
            then
            depoch=$(date +%s)
            tailleD=$(du $file -k | cut -f 1);
            echo "$i,$tailleD,$depoch"
      fi;
done

exit 0;
```

Explication du script bash :

Premièrement on spécifie au système d'utiliser l'interpréteur bash au lieu du shell par défaut.

Puis on assigne à la variable « path » le chemin du dossier Tmp

Ensuite en fait une boucle pour chaque objet (fichier ou dossier) se trouvant dans Tmp (à l'aide de ls)

Dans cette boucle on vérifie que l'objet avec lequel on travaille soit bien un fichier grâce au test -f.

Si c'est le cas on assigne à la variable depoch le timestamp correspondant (date en seconde), et on assigne la taille du fichier en Kilo octet à la variable tailleD (à l'aide de la commande du et cut)

Enfin on affiche dans la sortie standard le nom du fichier suivit de la taille du fichier suivit du timestamp en suivant ce format : nomFichier,tailleFichier,timeStamp

Pour finir on finit l'execution du script par un exit 0 ; pour signifier la fin du script sans erreur.

Exemple de bon fonctionnement :

Affichage suite au lancement du fichier script depuis un terminal :

```
f1,0,1637158681
f2,0,1637158681
f3,0,1637158681
```

Partie 2 : Génération de données

Explication de la consigne :

Afin de pouvoir observer des changements de la taille des fichiers du répertoire Tmp, nous avions à créer un script bash qui génère des chaines de caractères d'une longeur aléatoire dans les fichiers du répertoire Tmp.

Le script Bash :

```
path="/home/iziram/Documents/GitHub/iutCours/SAE_15/TP_1/Tmp/"
for i in $(ls $path)
do
    if test -f $path$i;
    then
        txt=""
        typeset -i nbT=$RANDOM
        typeset -i j=0
        while test $j -lt $nbT;
        do
            txt="${txt}a";
            j=$j+1;
        done
        echo "$txt" >> $path$i
```

```
    fi;
done
exit 0;
```

Explication du Script :

Premièrement on assigne le chemin du dossier Tmp à une variable path.

Puis pour chaque fichier qui se trouve dans le dossier Tmp on va executer les instructions suivantes :

- On assigne à la variable « txt » une chaine de caractères vide
- On assigne à la variable nbT un entier aléatoire (à l’aide de typeset -it qui spécifie le type de variable (entier) et $RANDOM qui est un générateur de nombre aléatoire)
- On déclare une variable d’itération j de type entier
- Puis tant que « j » est inférieur à « nbT » on ajoute à la variable « txt » la lettre a puis on incrémente de 1 « j ».
- Lorsque la boucle est finie on écrit dans le fichier notre variable « txt »

Et enfin on termine l’execution du script à l’aide du exit 0 ;

Exemple de bon fonctionnement :

Contenu du ficher f1 :

aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

Partie 3 : Répétition des scripts

Explication de la consigne :

Dans cette dernière partie nous devions mettre en place 2 taches répétitives. La première devait executer le script de génération de texte toutes les minutes tandis que la seconde devait executer le script de surveillance des fichiers toutes les deux minutes. Pour cela nous avons utiliser le gestionnaire de répétition de taches « Cron »

Contenu du fichier Cron :

```
* * * * * "/home/iziram/Documents/GitHub/iutCours/SAE_15/TP_1/genereTxt.sh"
*/2 * * * *
"/home/iziram/Documents/GitHub/iutCours/SAE_15/TP_1/surveilleTailleFichiers.sh"
>> "/home/iziram/Documents/GitHub/iutCours/SAE_15/TP_1/rapport"
```

Explication du fichier Cron :

Chaque ligne du fichier cron symbolise une tache à répéter.

De cette manière nous avons donc symbolisé la première tache (la génération de txt aléatoire dans les fichiers du dossier Tmp) . les 5 * du début signifie que le fichier s'executera toutes les minutes. Ensuite nous indiquons le chemin absolu du script à executer.

Pour la seconde tache nous avons la même structure mais avec quelques changements à noter :

- La première * s'est transformée en « */2 » Cela signifie qu'au lieu de s'executer toutes les minutes, la tache s'executera toutes les 2 minutes.
- On a changer le chemin du script pour utiliser le script de surveillance des fichiers
- A l'aide de « >> » nous changeons l'affichage de la sortie standard vers le chemin qui se trouve à droite.
- Enfin on a le chemin du fichier dans lequel l'affichage du script de surveillance sera redirigé.

Exemple de bon fonctionnement :

contenu du fichier rapport dans le dossier TP_1 :

```
f1,4,1637157602
f2,4,1637157602
f3,4,1637157602
f1,24,1637157628
f2,8,1637157628
f3,4,1637157628
f1,24,1637157721
f2,8,1637157721
f3,4,1637157721
```