

RAPPORT DU TP SUR LA MISE EN PLACE DE SAUVEGARDES AUTOMATISEES

Rédigé par : Le Groupe 6

- DJONTU Etienne Sorel
- LEFODEM ZANGUE ASSOUNA Aristhole
- NGOKO Angeline
- KOMBANG MEBANGA Sosthene

Supervisé par :

Dr. ADAMOU Hamza && M. NGOUANFO Gildas

INTRODUCTION

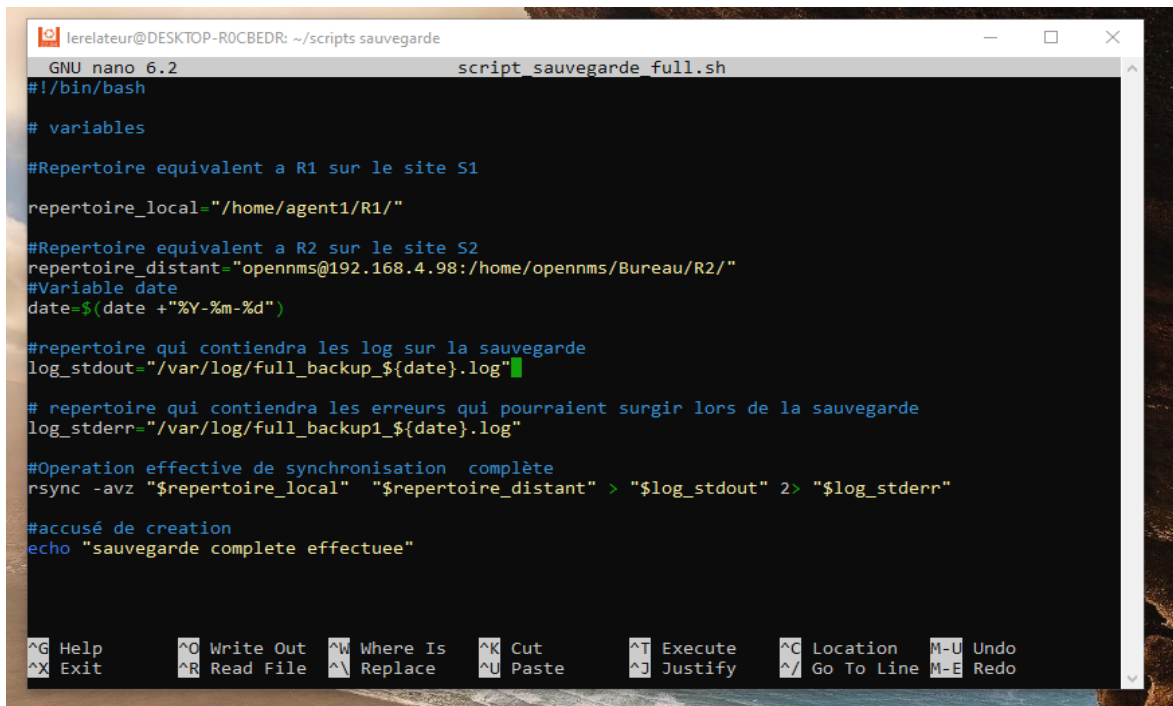
Au cours des présentations des rapports concernant les différents thèmes pour le cours de **INF 361.1 (Administration Systèmes)**, il a été chargé à tous les étudiants, et ceux du **groupe 6** en particulier, de corriger le TD laissé par M. NGOUANFO par rapport aux concepts de sauvegardes et restauration. Dans ce rapport, nous allons présenter comment s'est fait les scripts demandés, ainsi que les réponses aux questions de fin. Nous allons notamment :

- Expliquer les différents scripts
- Démontrer le contenu de la crontab avec les explications
- Justifier les heures choisis pour les différentes opérations.

I. Explications des scripts

˘ Attacher ce rapport, sont les trois fichiers de sauvegardes et nettoyage des données. Le script de sauvegarde complète, **le script de sauvegarde différentielle, et le script de nettoyage des sauvegardes**. Il était question de créer des scripts qui permettent de faire des sauvegardes complètes, puis différentielles d'un répertoire **R1** sur un serveur **S1** dans un répertoire **R2** sur un serveur distant **S2**.

1. Script de sauvegarde complète



```
GNU nano 6.2 script_sauvegarde_full.sh
#!/bin/bash

# variables

#Repertoire equivalent a R1 sur le site S1
repertoire_local="/home/agent1/R1/"

#Repertoire equivalent a R2 sur le site S2
repertoire_distant="opennms@192.168.4.98:/home/opennms/Bureau/R2/"
#Variable date
date=$(date +"%Y-%m-%d")

#repertoire qui contiendra les log sur la sauvegarde
log_stdout="/var/log/full_backup_${date}.log"

# repertoire qui contiendra les erreurs qui pourraient surgir lors de la sauvegarde
log_stderr="/var/log/full_backup1_${date}.log"

#Operation effective de synchronisation complète
rsync -avz "$repertoire_local" "$repertoire_distant" > "$log_stdout" 2> "$log_stderr"

#accusé de creation
echo "sauvegarde complete effectuee"
```

Fig. 1 : Script de sauvegarde complète

Dans ce Script :

- Etant donné que le terminal utilisé a déjà une arborescence, nous avons décidé de l'utiliser pour les tâches. Mais les chemins d'accès se remplacent tout à fait correctement par ceux demandés dans l'exercice.
- La sauvegarde peut se faire en local, mais dans notre cas, elle se fait à distance vers le serveur distant S2. Pour cela :
 - Il faudrait avoir au préalable une connexion **SSH entre les deux serveurs** et connaître également leurs adresse IP.
 - Faire la sauvegarde vers **[username]@[IP] : [chemin destination]**

NB :

- ✚ Dans notre cas, le serveur local S1 est nommé **agent1** et chemin vers le répertoire local R1 **/home/agent1/R1/** avec une adresse IP de **192.168.4.129/24**.
- ✚ Le serveur distant S2 est nommé **opennms** et chemin vers le répertoire distant R2 **/home/opennms/Bureau/R2/** avec une adresse IP de **192.168.4.98/24**.

2. Script de sauvegardes différentielles

```
root@agent1-opennms: /home/agent1
GNU nano 6.2 script_sauvegarde_diff.sh
#!/bin/bash

#variables

#la variable suivante contient le chemin absolu du repertoire R1 dans le serveur locale S1
source="/home/agent1/R1/"

#la variable suivante contient le chemin absolu du repertoire R2 dans le serveur distant S2
#l'accès à S2 se fait via une connexion ssh

dest="opennms@192.168.1.101:/home/opennms/Bureau/R2/"

#la variable suivante contient le chemin vers le repertoire du serveur S2 distant
#contenant les sauvegardes différentielles

dest_backup="opennms@192.168.1.101:/home/opennms/Bureau/sauvegarde"

#le repertoire ci-dessous est celui contenant les sauvegardes différentielles en locales

backup_diff="/home/agent1/sauvegarde_diff/"

#date des sauvegardes différentielles

date=$(date +"%d-%m-%Y")

#logs sur les sauvegardes différentielles qui seront contenues dans des répertoires différents
#selon la date à laquelle la sauvegarde différentielle a été effectuée

log_file_diff="/var/log/sauvegarde_diff_${date}.log"

#repertoire qui contiendra les log sur la sauvegarde

log_stdout_diff="/var/log/backup_diff_${date}.log"

# repertoire qui contiendra les erreurs qui pourraient surgir lors de la sauvegarde

^G Aide      ^O Écrire   ^W Chercher ^K Couper   ^T Exécuter ^C Emplacement M-U Annuler  M-A Marquer
^X Quitter   ^R Lire fich. ^\ Remplacer ^U Coller   ^J Justifier ^_ Aller ligne M-E Refaire  M-6 Copier
```

```
root@agent1-opennms: /home/agent1
GNU nano 6.2 script_sauvegarde_diff.sh
log_stdout_diff="/var/log/backup_diff_${date}.log"

# repertoire qui contiendra les erreurs qui pourraient surgir lors de la sauvegarde

log_stderr_diff="/var/log/error_backup_diff_${date}.log"

#la commande suivante creera une variable nommee modifications dans la quelle sera stockee
#les fichiers et/repertoires qui ont ete modifiés en effectuant une comparaison entre le contenu
#du repertoire distant et celui du repertoire source

modifications=$(rsync -av --dry-run --compare-dest="$dest" "$source" "$backup_diff" | grep -v "sending incremental file list" | gr>

#ici on verifi si le contenu de la variable modifications a change: si oui, on copie le fichier ou repertoire
#concerner dans le repertoire contenant les sauvegardes differentielle dans le serveur distant S2
#sinon, on envoie juste le message pour informe qu'aucune sauvegarde na ete faite.

if [ -z "$modifications" ]; then

echo "Aucune modification detectee. Sauvegarde non necessaire. "
exit 0

fi

#sauvegarde differentielle du repertoire R1 en local avant transfert

rsync -av --compare-dest="$dest" "$source" "$backup_diff" > "$log_stdout_diff" 2> "$log_stderr_diff"

#transfert du repertoire contenant la sauvegarde differentielle effectuee

rsync -av "$backup_diff" "$dest_backup" > "$log_stdout_diff" 2> "$log_stderr_diff"

echo "sauvegarde differentielle effectuee avec succes"
```

Fig. 2 : Script détaillé de sauvegarde Différentielle

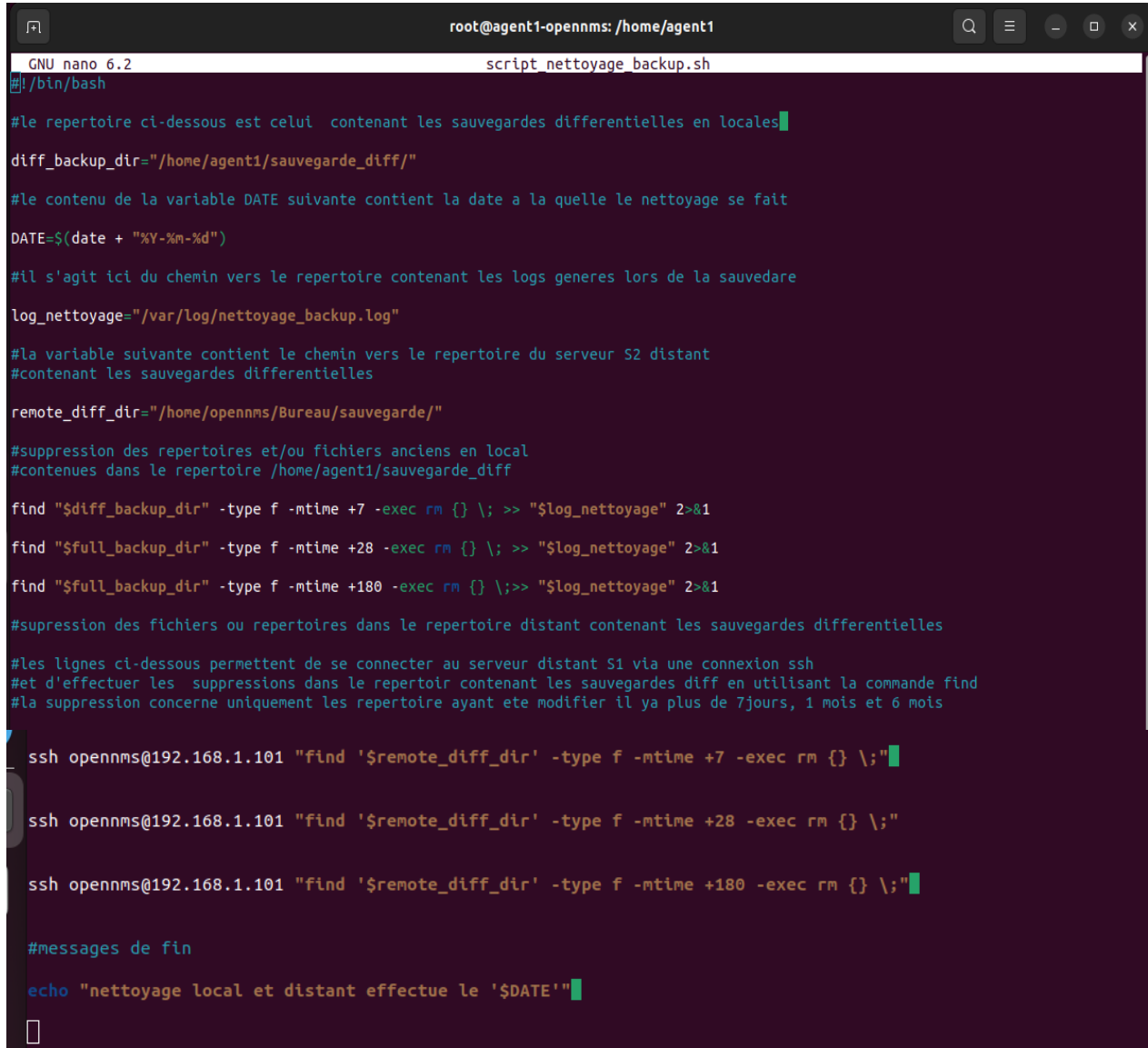
Dans ce script, on vérifie d'abord s'il y a des modifications qui ont été faite sur le répertoire à sauvegarder. On simule d'abord une sauvegarde différentielle, **c'est-à-dire sauvegarde basée sur les modifications constatées** telle que s'il y a effectivement des changements, le répertoire et/ou fichier concerner est stocké dans la **variables modifications**.

Ensuite, on enlève les possibles sauts de ligne et espaces blancs, afin de n'avoir que les éventuelles modifications.

Si cette variable contient effectivement quelque chose après avoir fait tout ça, alors il y a effectivement eu modification, et la sauvegarde peut se faire.

3. Script de nettoyage

Le nettoyage se fait en fonction de l'âge (plus de 7 jours, 4 semaines = 28 jours, 6 mois = 180 jours) en locale et à distance.



```
GNU nano 6.2 script_nettoyage_backup.sh
#!/bin/bash

#le repertoire ci-dessous est celui contenant les sauvegardes differentielles en locales
diff_backup_dir="/home/agent1/sauvegarde_diff/"

#le contenu de la variable DATE suivante contient la date a la quelle le nettoyage se fait
DATE=$(date + "%Y-%m-%d")

#il s'agit ici du chemin vers le repertoire contenant les logs generes lors de la sauvedare
log_nettoyage="/var/log/nettoyage_backup.log"

#la variable suivante contient le chemin vers le repertoire du serveur S2 distant
#contenant les sauvegardes differentielles
remote_diff_dir="/home/opennms/Bureau/sauvegarde/"

#suppression des repertoires et/ou fichiers anciens en local
#contenus dans le repertoire /home/agent1/sauvegarde_diff

find "$diff_backup_dir" -type f -mtime +7 -exec rm {} \; >> "$log_nettoyage" 2>&1
find "$full_backup_dir" -type f -mtime +28 -exec rm {} \; >> "$log_nettoyage" 2>&1
find "$full_backup_dir" -type f -mtime +180 -exec rm {} \; >> "$log_nettoyage" 2>&1

#suppression des fichiers ou repertoires dans le repertoire distant contenant les sauvegardes differentielles
#les lignes ci-dessous permettent de se connecter au serveur distant S1 via une connexion ssh
#et d'effectuer les suppressions dans le repertoire contenant les sauvegardes diff en utilisant la commande find
#la suppression concerne uniquement les repertoire ayant ete modifier il ya plus de 7jours, 1 mois et 6 mois

ssh opennms@192.168.1.101 "find '$remote_diff_dir' -type f -mtime +7 -exec rm {} \;"

ssh opennms@192.168.1.101 "find '$remote_diff_dir' -type f -mtime +28 -exec rm {} \;"

ssh opennms@192.168.1.101 "find '$remote_diff_dir' -type f -mtime +180 -exec rm {} \;"

#messages de fin
echo "nettoyage local et distant effectue le '$DATE'"
```

Fig.3 : script de nettoyage détaillé

Ce script est commenté et comporte toutes les explications nécessaires permettant de bien la comprendre.

II. CONTENU DE LA Crontab

Afin de planifier les sauvegardes et le nettoyage, on utilise l'outil **cron** qui est capable de planifier une multitude de taches pouvant s'exécutées simultanément.

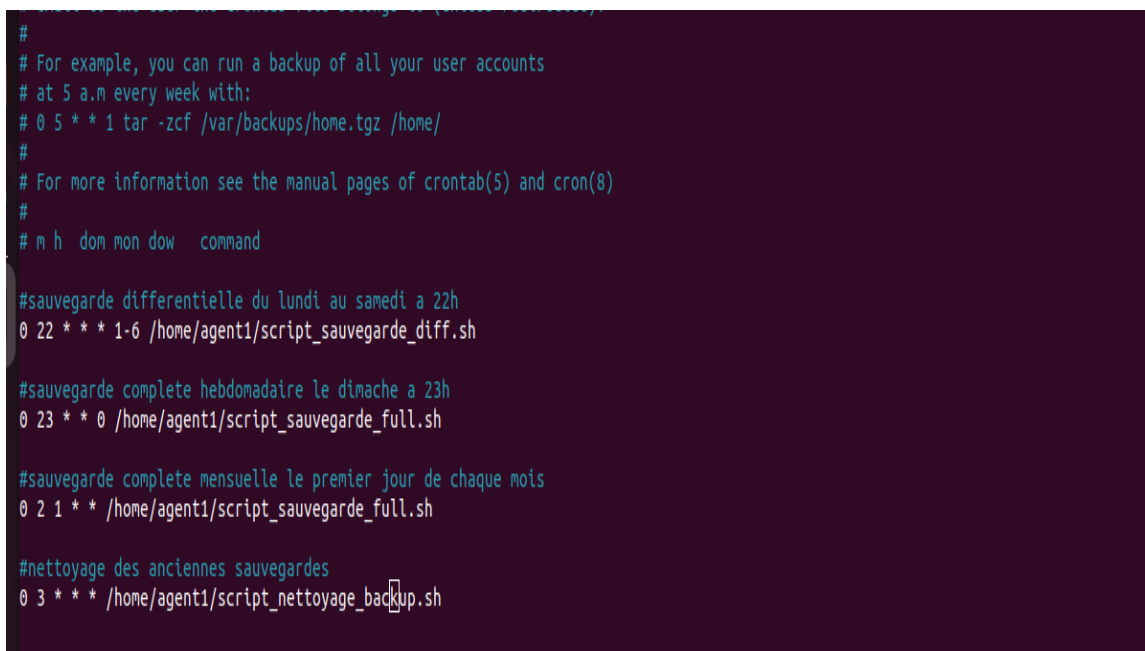
On ajoute les liens suivants pour répondre à la question posée :

0 22 * * 1-6 /chemin/vers/script_sauvegarde_diff.sh

0 23 * * 0 /chemin/vers/script_sauvegarde_full.sh

0 2 1 * * /chemin/vers/script_sauvegarde_full.sh

0 3 * * * /chemin/vers/script_nettoyage_backup.sh



```
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
#sauvegarde differentielle du lundi au samedi a 22h
0 22 * * * 1-6 /home/agent1/script_sauvegarde_diff.sh

#sauvegarde complete hebdomadaire le dimanche a 23h
0 23 * * 0 /home/agent1/script_sauvegarde_full.sh

#sauvegarde complete mensuelle le premier jour de chaque mois
0 2 1 * * /home/agent1/script_sauvegarde_full.sh

#nettoyage des anciennes sauvegardes
0 3 * * * /home/agent1/script_nettoyage_backup.sh
```

Fig. 4 : planification des sauvegardes dans la crontab

Il est à noter que à partir du moment où la liste des taches a été ajoutée dans la crontab, elle prend effet immédiatement et seront exécutées selon le planning dans la crontab, par conséquent, **cron n'est pas une commande qu'il faudra exécutée.**

5. Justification des horaires

Les horaires ont été choisis pour ne pas empêcher l'utilisation normales des ressources et du temps de l'entreprise ;

- **22h** : Sauvegardes différentielles après les heures de bureau, pour presque directement mettre les modifications en lieu sûr.

- **23h** : Sauvegardes complètes hebdomadaires, afin que la sauvegarde ait le temps de se terminer avant que les activités ne reprennent ;

- **2h** : Sauvegardes complètes mensuelles à cette heure pour exploiter au maximum les ressources de l'entreprise ;

- **3h** : Nettoyage effectué après les sauvegardes, pour réduire l'espace utilisé avant le prochain jour.