

Atelier n°4 : Solution de Sauvegarde rsync/cron

L'objectif est de mettre en place une sauvegarde automatique sans intervention humaine et pour cela, nous avons deux méthodes possibles qui sont ci-dessous et la différence entre les deux sont:

Pull : Les données sont extraites des sources par le système de sauvegarde.

Push : Les données sont envoyées activement depuis les sources vers le système de sauvegarde

La sauvegarde doit être réalisée toutes les 5 minutes (pour les tests) et toutes les nuits lorsque la solution sera fonctionnelle il faut premièrement créer un nouveau conteneur nommé backup, la commande est

```
lxc-copy -n <container> -N <nouveau_container>
```

lorsque le nouveau conteneur backup a été créé pour la sauvegarde, il faut modifier son adresse ip dans le répertoire /etc/network avec la commande :

```
nano interfaces
```

et le redémarrer avec **systemctl restart networking**

creation du nouvelle machine de sauvegarde Adresse de mon backup1:10.31.96.98 Pour mettre en place l'authentification SSH par clés sans passphrase : On a Généré une paire de clés SSH sur notre machine locale à l'aide de la commande

```
ssh-keygen ssh-keygen -t rsa -b 4098
```

on a Copié la clé publique générée (habituellement située dans ~/.ssh/id_rsa.pub) sur le serveur distant dans le fichier ~/.ssh/authorized_keys. **l'outil rsync** Rsync est un outil de synchronisation de fichiers, généralement utilisé pour copier et synchroniser des fichiers locaux ou distants de manière efficace. Il minimise la quantité de données copiées en ne transférant que les parties des fichiers qui ont été modifiées. Rsync peut être utilisé pour effectuer des sauvegardes locales ou distantes (mirroring) on a installé rsync sur tous les contener (dns, backup, web) avec la commande

```
apt-get update
apt-get install rsync
```

il est possible d'utiliser Rsync de deux manières différentes 1. Mode démon (client/serveur) : non sécurisé par défaut 2. Via SSH : sécurisé (les données ne sont pas transférées en clair)

Dans le cas de la méthode **PUSH**, c'est la machine à sauvegarder qui va se connecter à la machine de backup et qui va initier le transfert.

```
rsync -azv -e ssh /etc/apache2 backup@backup1.m2l.org:/home/backup/web1
```

sert à synchroniser le répertoire /etc/apache2 de la machine locale vers un répertoire distant sur une machine nommée backup1.m2l.org

Dans le cas de la méthode **PULL**, c'est la machine de sauvegarde qui va se connecter aux machines à sauvegarder et qui va initier le transfert.

```
rsync -avz -e ssh backup@www.m2l.org:/etc/apache2 /home/backup/web1
```

En résumé, la principale différence entre les deux commandes réside dans le sens de la synchronisation : dans le cas du PUSH, les fichiers sont envoyés depuis la machine locale vers la machine distante, tandis que dans le cas du PULL, les fichiers sont récupérés depuis la machine distante vers la machine locale.

Script de lancement de la synchronisation

```
#!/bin/bash

# Définir une variable pour le nom de fichier avec la date et l'heure
actuelle
logfile="/home/backup1/backup_$(date +%Y-%m-%d_%H-%M-%S').log"

# Répertoire de sauvegarde pour les backups
backup_dir="/home/backup1"

# Créer les répertoires de sauvegarde s'ils n'existent pas
mkdir -p "$backup_dir/web1"
mkdir -p "$backup_dir/ns1"
mkdir -p "$backup_dir/ns2"

# Enregistrer l'heure de début de sauvegarde dans le fichier journal
echo "Début de la sauvegarde le $(date +%Y-%m-%d %H:%M:%S)" >> "$logfile"

# Effectuer les sauvegardes avec rsync et journaliser les sorties dans un
fichier de log
{
    # Sauvegarder les fichiers de configuration d'Apache2 du conteneur web
    rsync -avz root@10.31.96.81:/etc/apache2/
"$backup_dir/web1/apache2_config"

    # Sauvegarder les sites web du serveur web
    rsync -avz root@10.31.96.81:/var/www/html/ "$backup_dir/web1/sites"

    # Sauvegarder les fichiers de logs d'Apache
    rsync -avz root@10.31.96.81:/var/log/apache2/
"$backup_dir/web1/apache2_logs"

    # Sauvegarder les fichiers de configuration des deux serveurs DNS
    rsync -avz root@10.31.96.54:/etc/bind/ "$backup_dir/ns1/bind_config"
    rsync -avz root@10.31.96.64:/etc/bind/ "$backup_dir/ns2/bind_config"

    # Sauvegarder les fichiers de logs de Bind
    rsync -avz root@10.31.96.54:/var/log/syslog "$backup_dir/ns1/bind_logs"
    rsync -avz root@10.31.96.64:/var/log/syslog "$backup_dir/ns2/bind_logs"

    # Sauvegarder les bases de données du serveur web avec mysqldump
    # Remplacez les options de mysqldump appropriées pour votre
configuration
```

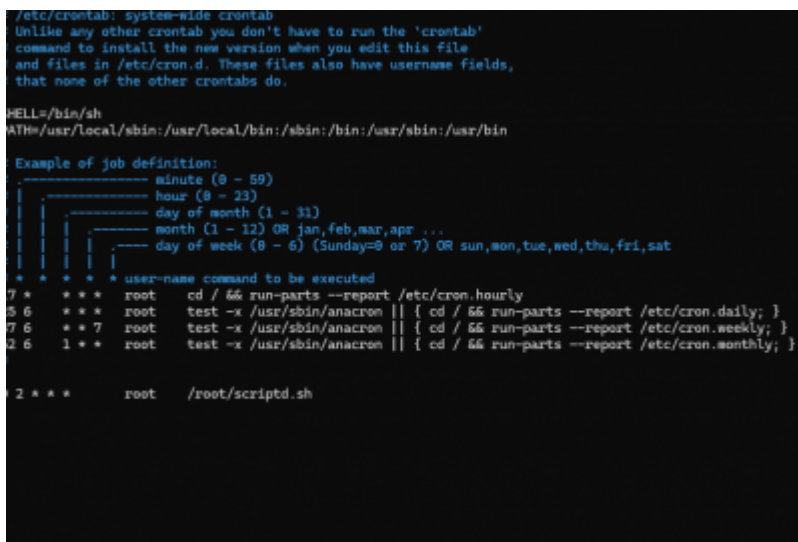
```
mysqldump -h 10.31.96.81 -u DOS -ppassword --all-databases >
"$backup_dir/web1/db_backup.sql"
} >> "$logfile" 2>&1 # Redirige les sorties standards et les erreurs vers
le fichier de log

# Enregistrer l'heure de fin de sauvegarde dans le fichier journal
echo "Fin de la sauvegarde le $(date +%Y-%m-%d %H:%M:%S)" >> "$logfile"

# Notification de fin de sauvegarde
echo "Sauvegarde terminée. Les détails sont dans le fichier de log :
$logfile"
```

cron cron est un service serveur bien utile et assez simple à mettre en œuvre. Il permet de programmer des actions à intervalles réguliers.

structure de ma crontab Vérifier votre crontab : <https://crontab.guru/>



```
/etc/crontab: system-wide crontab
Unlike any other crontab you don't have to run the `crontab`
command to install the new version when you edit this file
and files in /etc/cron.d. These files also have username fields,
that none of the other crontabs do.

HELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# Example of job definition:
# .
# minute (0 - 59)
# hour (0 - 23)
# day of month (1 - 31)
# month (1 - 12) OR jan,feb,mar,apr ...
# day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# * * * * * user-name command to be executed
7 * * * * root cd / && run-parts --report /etc/cron.hourly
6 * * * * root test -x /usr/sbin/anacron || { cd / && run-parts --report /etc/cron.daily; }
5 * * * * root test -x /usr/sbin/anacron || { cd / && run-parts --report /etc/cron.weekly; }
4 * * * * root test -x /usr/sbin/anacron || { cd / && run-parts --report /etc/cron.monthly; }
.
.
2 * * * * root /root/scriptd.sh
```

La commande `crontab -e` est utilisée pour éditer le fichier de configuration des tâches planifiées sur un système Unix ou Linux. nb : **SSH**

Selon la méthode choisie (PUSH ou PULL), les clés sont générées sur la machine de sauvegarde ou les machines à sauvegarder. Des comptes “backup” doivent être créés sur toutes les machines, et via “sudo”, le compte “backup” doit être autorisé à utiliser “rsync” avec des permissions administrateur.

La commande pour lancer un script dépend du langage de programmation utilisé pour écrire le script. `./nom_du_script.sh` La commande **cat /home/backup1/backup_2024-03-13_15-35-4** est utilisée pour afficher le contenu du fichier nommé “backup_2024-03-13_15-35-4” qui se trouve dans le répertoire “/home/backup1”.



```
rsync: command started on 2024-03-13 15:35:41
rsync: incremental file list
sent 20 bytes, received 8.80k bytes, 31.79k bytes/sec
total size is 91.14k, speedup is 10.68
rsync: incremental file list
sent 828 bytes, received 134.567 bytes, 260.998 bytes/sec
total size is 81.949.721, speedup is 509.16
rsync: incremental file list
sent 20 bytes, received 40k bytes, 312.67 bytes/sec
total size is 21.387, speedup is 82.88
rsync: incremental file list
sent 20 bytes, received 333 bytes, 700.80 bytes/sec
total size is 4.807, speedup is 19.42
rsync: connect to host 18.11.96.64 port 22: no route to host
rsync: connection unexpectedly closed (8 bytes received so far) [Receiver]
rsync error: unexplained error (code 255) at io.c(210) [Receiver] 2.71
rsync: incremental file list
rsync
sent 1.19k bytes, received 767 bytes, 3.93k bytes/sec
total size is 119.109, speedup is 69.11
rsync: connect to host 18.11.96.64 port 22: no route to host
rsync: connection unexpectedly closed (8 bytes received so far) [Receiver]
rsync error: unexplained error (code 255) at io.c(210) [Receiver] 2.71
Fin de la sauvegarde le 2024-03-13 15:36:01
root@beaupyat:~#
```

From:

<https://sisr2.beaupeyrat.com/> - **Documentations SIO2 option SISR**

Permanent link:

<https://sisr2.beaupeyrat.com/doku.php?id=sisr1-g6:rsync>

Last update: **2024/03/13 15:46**

