



GSB_Mission_1

Documentation, machine ap22-backup

Cette page contient l'ensemble des scripts et configurations effectuées sur la machine VirtualBox de sauvegarde *ap22-backup*.

Scripts et configurations

La machine était un .ova de store2 [<http://store2.sio.lan/sionas/Public/ova/debian/>] donc aucune configuration n'a été requise initialement. Deux clés SSH ont dû être manuellement exportées vers la machine, la clé de la machine physique pour du travail en SSH, et la clé de la machine *ap22-prod* pour permettre une connexion via paire de clés et un gain de temps substantiel, tout en permettant d'utiliser le compte *root* sans failles de sécurités.

Avant de créer les scripts pour la base de données finale, elle a été clonée et renommée en “TEST”, puis des scripts identiques aux scripts finaux ont été élaborés pour tester les fonctionnalités sans risquer d'endommager la vraie base de données. Une fois ces scripts de test fonctionnels, ceux-ci ont été adaptés pour la base “bdgsb”.

Pour le script de sauvegardes de la base de données avec rotation, le script récupère le jour actuel, puis exporte la table *bdgsb* en local avant de la compresser au format .gz:

Script de sauvegarde

```
#!/bin/bash
DAT=$(date "+%Y")
ssh root@ap22-prod "mysqldump -u root -pYElMxUCgy -B bdgsb |gzip" > /root/bdgsb-$DAT.sql.gz
```

Ensuite, le *crontab* a dû être modifié pour permettre une exécution quotidienne de Lundi à Dimanche, pour effectuer des sauvegardes quotidiennes qui seront remplacées chaque début de semaine suivante:

crontab modifié

```
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
0 11 * * * 1-7 bash /root/bd-dump.sh
```

Une fois le *crontab* modifié pour s'exécuter chaque jour à 11 heures, le script *bd-dump.sh* peut être lancé pour des raisons de test, ce qui nous donne le résultat suivant:

Résultat de l'exécution du script

```
root@ap22-backup:~# ls
bd-dump.sh  bdgsb-4.sql.gz
```

Concernant le script de restauration des sauvegardes, celui-ci demande la version qu'on veut restaurer, puis se charge de restaurer la version spécifiée en lançant une commande spécifique via SSH:

Script de restauration

```
#!/bin/bash
echo "Quel est le jour précédent ? (Lundi = 1, Mardi = 2, ... , Dimanche = 7)"
read version
zcat /root/bdgsb-$version.sql.gz | ssh root@ap22-prod "mysql -u root -pYE1MxUCgy bdgsb"
```

Une fois que ce script a tourné, même si une table entière a été supprimée, la base de données reste intact, si aucune modification néfaste n'a été effectuée avant la sauvegarde.

Contenu des scripts

bd-dump.sh

```
#!/bin/bash
DAT=$(date "+%u")
ssh root@ap22-prod "mysqldump -u root -pYE1MxUCgy -B bdgsb |gzip" > /root/bdgsb-$DAT.sql.gz
```

bd-restore.sh

```
#!/bin/bash
echo "Quel est le jour précédent ? (Lundi = 1, Mardi = 2, ... , Dimanche = 7)"
read version
zcat /root/bdgsb-$version.sql.gz | ssh root@ap22-prod "mysql -u root -pYE1MxUCgy bdgsb"
```