AGUILAR_Flavien_Backup_TP

I - Basique:

1) Pour réaliser la partie 1) du tp, je vais utiliser deux scripts qui vont réaliser toutes les commandes nécessaire au backup et à la restoration des données :

Backup:

```
1 #!/bin/bash
2
3 mkdir /mnt/backup/
4 cp -ar /home/stallman /mnt/backup
5 cp -ar /home/kernighan/ /mnt/backup/
6 cp -ar /home/murdor/ /mnt/backup/
7 cp -ar /var/ /mnt/backup/
8 cp -ar /etc/fstab /mnt/backup/
```

Restore:

```
1 #!/bin/bash
2
3 cp -a /mnt/backup/murdor/debian /home/murdor
4 cp -a /mnt/fstab /etc/fstab
```

2) Nous allons commencer par mettre en place la backup en exécutant le script :

```
1 root@debian:~# ./backup.sh
```

3) Nous allons ensuite supprimer le fichier /home/murdor/debian et le fichier /etc/fstab

```
1 root@debian:~# rm -rf /home/murdor/debian /etc/fstab
```

4) Nous allons ensuite réaliser la restauration des données :

```
1 root@debian:~# ./restore.sh
```

5) Nous pouvons voir que la restauration des données c'est bien passé car le fichier /home/murdor/debian est bien présent ainsi que le fichier /etc/fstab

```
1 root@debian:~# ls /home/murdor/
2 debian
```

```
1 root@debian:~# cat /etc/fstab
2 # /etc/fstab: static file system information.
4 # Use 'blkid' to print the universally unique identifier for a
5 # device; this may be used with UUID= as a more robust way to name
      devices
6 # that works even if disks are added and removed. See fstab(5).
7 #
8 # <file system> <mount point>
                                  <type> <options>
                                                          <dump> <pass>
9 /dev/mapper/system-rootfs /
                                            ext4
                                                   errors=remount-ro 0
            1
10 # /boot was on /dev/sda1 during installation
11 UUID=e266859f-d5b4-4bb6-b7aa-d5fe4919397e /boot
                                                           ext4
      defaults
                     0
12 /dev/sr0
                  /media/cdrom0 udf,iso9660 user,noauto
                                                             0
                                                                     0
```

6) Lorsque nous essayons d'écrire RIP dans le fichier toto, il n'y a pas de problème car lors de ma backup j'ai précisé qu'il fallait conserver les droits du fichier. En revanche, si je n'avais pas fais cela, les droits du fichier serait ceux de l'utilisateur ayant rétabli les fichiers et l'écriture n'aurait pas été possible.

II - Industriel:

- 1) Nous allons commencer par installer docker et docker-compose en suivant la procédure indiquée sur le site officiel de docker.
- 2) Une fois docker et docker-compose installé, nous allons installaer Bareos :

```
1 root@debian:/home/test# cat install.sh
2 #!/bin/sh
3
4 DIST=Debian_10
5 RELEASE=release/19.2
6 URL=http://download.bareos.org/bareos/$RELEASE/$DIST
7 echo "deb $URL /\n" > /etc/apt/sources.list.d/bareos.list
8 wget -q $URL/Release.key -0- | apt-key add -
```

```
9 apt-get update
10 apt-get install bareos bareos-database-postgresql
```

Nous allons maintenant créer notre client :

```
1 root@debian:/home/test# cat /etc/hosts
2 127.0.0.1 localhost
3 127.0.1.1 debian
4 
5 # The following lines are desirable for IPv6 capable hosts
6 ::1 localhost ip6-localhost ip6-loopback
7 ff02::1 ip6-allnodes
8 ff02::2 ip6-allrouters
9 10.203.0.78 bareos-sd
```

```
1 root@debian:/home/test# cat /data/bareos/config/director/bareos-dir.d/
      pool/Home.conf
2 Pool {
3 Name = Home
  Pool Type = Backup
4
  Recycle = yes
                                        # Bareos can automatically
       recycle Volumes
6
    AutoPrune = yes
                                         # Prune expired volumes
7 Volume Retention = 365 days
                                        # How long should the Full
      Backups be kept? (#06)
8 Maximum Volume Bytes = 50G
                                        # Limit Volume size to something
        reasonable
                                 # Limit number of Volumes in Pool
# Volumes will be labeled "Full-<</pre>
9 Maximum Volumes = 100
     Label Format = "Home-"
        volume-id>"
11 }
```

Il va falloir maintenant créer le mode de sauvegarde LinuxHOME :

```
FileSet {
2
     Name = "LinuxHOME"
     Description = "Backup all regular filesystems, determined by
         filesystem type."
4
     Include {
5
      Options {
6
         Signature = MD5 # calculate md5 checksum per file
         One FS = No # change into other filessytems
7
8
         FS Type = btrfs
         FS Type = ext2 # filesystems of given types will be backed up
9
10
         FS Type = ext3 # others will be ignored
11
         FS Type = ext4
         FS Type = reiserfs
12
         FS Type = jfs
13
14
         FS Type = xfs
         FS Type = zfs
16
       }
       File = /home
18
     }
     # Things that usually have to be excluded
19
     # You have to exclude /var/lib/bareos/archive
20
21
     # on your bareos server
22
    Exclude {
       File = /var/lib/bareos
23
       File = /var/lib/bareos/archive
24
       File = /proc
25
26
       File = /tmp
       File = /var/tmp
27
       File = /.journal
28
29
       File = /.fsck
     }
31 }
```

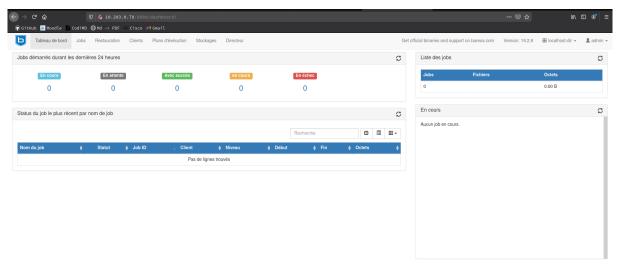
Nous allons maintenant relancer les containers dir et sd :

```
1 root@debian:/home/test# docker ps
2 CONTAINER ID
                                                       COMMAND
                     IMAGE
                     CREATED
                                         STATUS
                                                            PORTS
                       NAMES
                                                       "/docker-
3 6ade845d81a5
                     barcus/bareos-director:19-alpine
     entrypoint..!
                    54 minutes ago
                                       Up 17 minutes
                                                          9101/tcp
                    test_bareos-dir_1
4 72f6c37be68a
                    barcus/bareos-webui:19-alpine
                                                       "/docker-
     entrypoint..." 54 minutes ago Up 54 minutes
     0.0.0.0:8080->9100/tcp test_bareos-webui_1
5 0f56de56617c
                    barcus/php-fpm-alpine
                                                       "/docker-
                    54 minutes ago
     entrypoint.."
                                       Up 54 minutes
                                                           9000/tcp
                    test_php-fpm_1
6 e2daf0da4909
               barcus/bareos-client:19-alpine
                                                       "/docker-
```

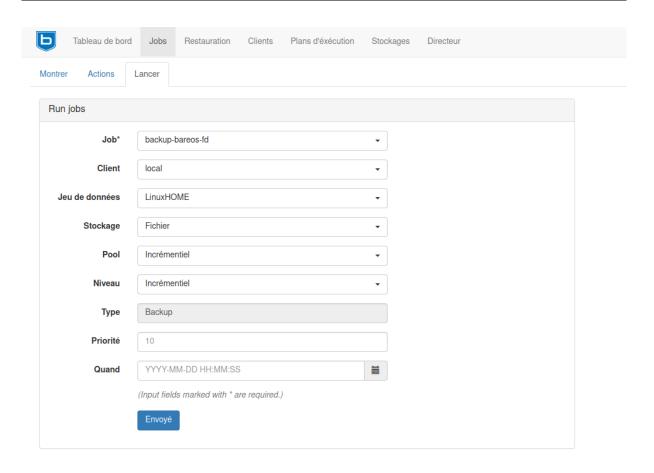
```
entrypoint..." 54 minutes ago Up 54 minutes
                                                        9102/tcp
                   test_bareos-fd_1
7 0998e6781067
                   mysql:5.6
                                                    "docker-
     entrypoint.s.! 54 minutes ago Up 54 minutes
                                                        3306/tcp
                  test_bareos-db_1
8 3508e700328e
                   namshi/smtp
                                                    "/bin/entrypoint
     .sh..." 54 minutes ago
                               Up 54 minutes
                                                 25/tcp
                    test_smtpd_1
                                                    "/docker-
9 873a8265b317
                   barcus/bareos-storage:19-alpine
     entrypoint.." 54 minutes ago
                                 Up 17 minutes
     0.0.0.0:9103->9103/tcp test_bareos-sd_1
```

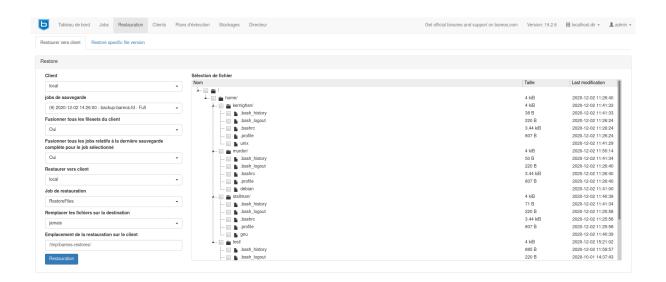
```
1 root@debian:/home/test# docker restart 6ade845d81a5 873a8265b317
```

Bareos est maintenant installé et configuré, nous pouvons y accèder depuis le navigateur web du client :



Nous pouvons désormais lancer la sauvegarde :

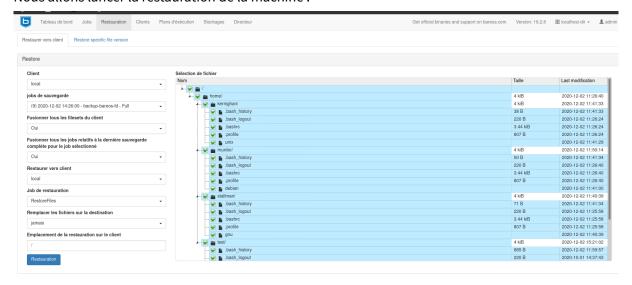




Nous allons maintenant supprimer le répertoire /home et nous allons le restaurer :

1 root@debian:/data/bareos/config/director/bareos-dir.d/fileset# rm -rf /
 home

Nous allons lancer la restauration de la machine :



Nous pouvons enfin retourner à la maison....

- 1 root@debian:/# cd /home/test/
- 2 root@debian:/home/test#