

# Installation d'un RAID 5 sur une VM Debian 12.8, avec les modules WEBDAV, SAMBA et SSH/SFTP

On commence ce tutoriel par l'installation de `mdadm`, qui est un outil qui permet de configurer et de gérer les matrices RAID sous Linux.

On commence par mettre à jour les packages disponible et télécharger puis installer le package :

```
sudo apt update
sudo apt install mdadm -y
```

## Création d'un RAID 5

On commence par ajouter 3 disque à sa VM, les miens seront par exemple de 5go chacun, veuillez redémarrer la VM pour qu'ils soient pris en compte. On peut effectuer cette commande pour vérifier les disques installé sur la machine :

```
lsblk
```

On peut voir nos disques, les miens sont `sdb` `sdc` et `sdd`.

### Création du tableau

La commande `mdadm --create` permet de créer le tableau de RAID.

```
sudo mdadm --create --verbose /dev/md0 --level=5 --raid-devices=3 /dev/sdb /dev/sbc /dev/sdd
```

La variable `/dev/md0` est le nom du nouveau périphérique qui va être créer. Les variables `/dev/sdb` `/dev/sbc` `/dev/sdd` sont à changer selon le nom de vos disques.

L'outil commencera à configurer la baie Cette opération peut prendre un certain temps, mais la baie reste utilisable pendant ce temps. Vous pouvez suivre la progression de la mise en miroir en consultant le fichier `/proc/mdstat` :

```
cat /proc/mdstat
```

Comme vous pouvez le constater sur la première ligne en surbrillance, le `/dev/md0` périphérique a été créé dans la configuration RAID 5 à l'aide des périphériques `/dev/sdb`, `/dev/sdc` et `/dev/sdd`. La deuxième ligne en surbrillance indique la progression de la création.

### Création et montage du système de fichiers

Ensuite, créez un système de fichiers sur la matrice :

```
sudo mkfs.ext4 -F /dev/md0
```

Créez un point de montage pour attacher le nouveau système de fichiers :

```
sudo mkdir -p /mnt/md0
```

Vérifiez si le nouvel espace est disponible en tapant :

```
df -h -x devtmpfs -x tmpfs
```

Le nouveau système de fichiers est monté et accessible.

### Enregistrement de la disposition du tableau

Pour s'assurer que le tableau soit réassemblé automatiquement au démarrage, nous devons ajuster le fichier `/etc/mdadm/mdadm.conf`.

Comme indiqué précédemment, avant de modifier la configuration, vérifiez à nouveau que l'assemblage de la matrice est terminé. Si vous effectuez cette étape avant la création de la matrice, le système ne pourra pas l'assembler correctement au redémarrage.

```
cat /proc/mdstat
```

Le résultat nous indique que la reconstruction est terminée. Nous pouvons maintenant analyser automatiquement le tableau actif et ajouter le fichier en saisissant :

```
sudo mdadm --detail --scan | sudo tee -a /etc/mdadm/mdadm.conf
```

Ensuite, vous pouvez mettre à jour l'initramfs, ou système de fichiers RAM initial, afin que la matrice soit disponible pendant le processus de démarrage précoce :

```
sudo update-initramfs -u
```

Ajoutez les nouvelles options de montage du système de fichiers au fichier `/etc/fstab` pour le montage automatique au démarrage :

```
echo '/dev/md0 /mnt/md0 ext4 defaults,nofail,discard 0 0' | sudo tee -a /etc/fstab
```

Votre matrice RAID 5 devrait maintenant être automatiquement assemblée et montée à chaque démarrage.

## Configuration de Samba pour le fichier partagé

Mettez à jour le système et installez Samba :

```
sudo apt update
sudo apt install samba -y
```

Vous pouvez ajouter un nouvel utilisateur pour utiliser Samba avec :

```
sudo useradd -m utilisateur1
```

Changez "utilisateur1" par l'utilisateur que vous voulez créer.

Créez un mot de passe pour cet utilisateur :

```
sudo passwd utilisateur1
```

Et assigniez-le au group Samba :

```
sudo smbpasswd -a utilisateur1
```

### Préparation du fichier partagé

Nous allons créer le fichier partagé dans le RAID 5 que nous avons fait juste avant :

```
sudo mkdir /dev/md0/partage
```

Maintenant vous pouvez modifier le fichier de configuration de Samba `smb.conf` :

```
sudo nano /etc/samba/smb.conf
```

Il permet de créer des dossiers partagés, d'en autoriser l'accès et de configurer d'autres paramètres de service importants.

Créez maintenant une nouvelle ressource et définissez ses droits d'accès.

Créez un dossier « partage » et définissez les autorisations pour l'utilisateur 1 :

```
[partage]
path = /dev/md0/partage
read only = no
guest ok = no
valid users = utilisateur1
```

Après avoir créé ces éléments, le répertoire « partage » sera accessible à l'utilisateur 1.

### Redémarrer Samba

Après avoir modifié les paramètres, le service doit être redémarré :

```
sudo systemctl restart smbd.service
```

### Vérification

Pour vérifier le bon fonctionnement, on peut effectuer le test dans la barre d'adresse de l'explorateur windows en tapant :

```
\\Votre_IP_du_serveur\partage
```

Vous devriez avoir accès aux fichiers qui seront stocké dans `/dev/md0/partage`

## Configuration du Webdav

---

On installe Apache 2 :

```
sudo apt install apache2 cadaver -y
```

On active les modules d'Apache :

```
sudo a2enmod dav
sudo a2enmod dav_fs
```

Redémarrez Apache pour sauvegarder :

```
sudo systemctl restart apache2.service
```

### Configuration d'Apache pour le Webdav sur le NAS

On commence par créer un dossier dans le NAS où seront stocker les fichiers :

```
sudo mkdir /mnt/md0/webdav
```

On donne les droit à Apache pour qu'il puisse lire ce dossier :

```
sudo chown www-data:www-data /mnt/md0/webdav
```

Créez un fichier de configuration dans le `/etc/apache2/sites-available/` comme ceci :

```
sudo nano /etc/apache2/sites-available/webdav.conf
```

Et ajoutez-y la configuration suivante :

```
<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    DocumentRoot /mnt/md0/webdav
    DAV On
    <Directory /mnt/md0/webdav>
        Options Indexes
        AllowOverride None
        Require all granted
    </Directory>
</VirtualHost>
```

Activez la configuration et relancer Apache :

```
sudo a2ensite webdav.conf
sudo systemctl reload apache2
```

Avec l'adresse IP vous pouvez vous connecter sur le Webdav.