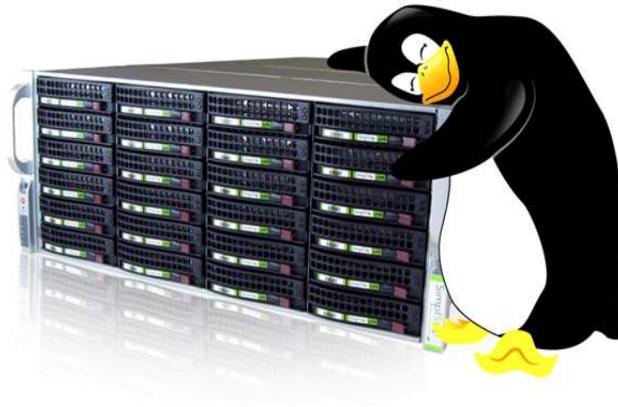


Linux Volume Manager (LVM)



Linux Volume Manager (LVM)

● Définition :

LVM pour « logical volume manager » est une technique qui permet de modifier la taille des partitions sur les disques durs sans avoir besoin de tout reformater, voire de créer des partitions s'étalant sur plusieurs disques.

L'objectif est ainsi d'éviter arrêt et redémarrage d'une machine en production. Cette technique est disponible sur linux depuis la version 2.4 du noyau.

Actuellement , nous sommes à la version 2 du LVM (noté LVM2)

En plus de ces fonctionnalités, LVM permet la création de snapshots.

Avantages :

- Pas de limitations de partitions (comme primaire, étendue, etc.).
- On ne se préoccupe plus de l'emplacement exact des données.
- On peut conserver quelques giga-octets de libres pour pouvoir les ajouter n'importe où et n'importe quand.

- Les opérations de redimensionnement deviennent quasiment sans risques, et peuvent être effectué à chaud « hot swap » contrairement au redimensionnement des partitions. Il n' ya pas besoins de redémarrer et /ou d'arrêter le serveur, on parle alors de haute disponibilité.

Linux Volume Manager (LVM)

Les Snapshots (instancées) :

Un instantané permet de pouvoir figer l'état du système de fichier à un instant t pour des besoins de sauvegarde.

Une sauvegarde peut parfois durer longtemps (plusieurs minutes ou heures) et il est important que pendant le temps que dure la sauvegarde les données ne soient pas modifiées, ceci pour assurer la cohérence des données. Ainsi il est souvent nécessaire de stopper toute activité logicielle qui utilise le système de fichier pour le temps d'une sauvegarde.

Grâce à un snapshot on sauvegarde en fait l'état des données à un instant donné. L'intérêt est que la création d'un snapshot est quelque chose de beaucoup plus rapide qu'une sauvegarde, cela permet de stopper les applications uniquement pour le temps de la création du snapshot.

Le principe du snapshot est en fait de sauvegarder les metadata (pointeurs sur les blocs de fichiers) et d'assurer que pendant la durée de vie du snapshot, aucune modification ne sera faite sur ses blocs. Ainsi, une fois le snapshot créé on peut effectuer la sauvegarde du système de fichier sans risque de corruption.

Linux Volume Manager (LVM)

Organisation de LVM:

Le principe de LVM est de pouvoir regrouper *un ou plusieurs disques dur* dans un *groupe*. Ce groupe peut ensuite être redécoupé en *disques logiques*. Ce système très flexible permet beaucoup de souplesse dans la configuration de ses disques durs.

Attention toutefois, LVM ne doit pas être utilisé avec la partition /boot. Les distributions récentes permettent pour la plupart de créer des volumes LVM dès l'installation, ce qui facilite son utilisation. Il est bien sûr possible de configurer LVM sur de nouveaux disques durs que l'on souhaite installer.

Lorsqu'on installe LVM sur un disque, toutes les données présentes sont effacées!

Si LVM est utilisé après l'installation du système d'exploitation, il est nécessaire de charger au démarrage le module qui prend en charge LVM. Ceci se fait grâce à la commande `lvcreate_initrd`. Il faudra également modifier en conséquence le fichier `/etc/lilo.conf`.

Linux Volume Manager (LVM)

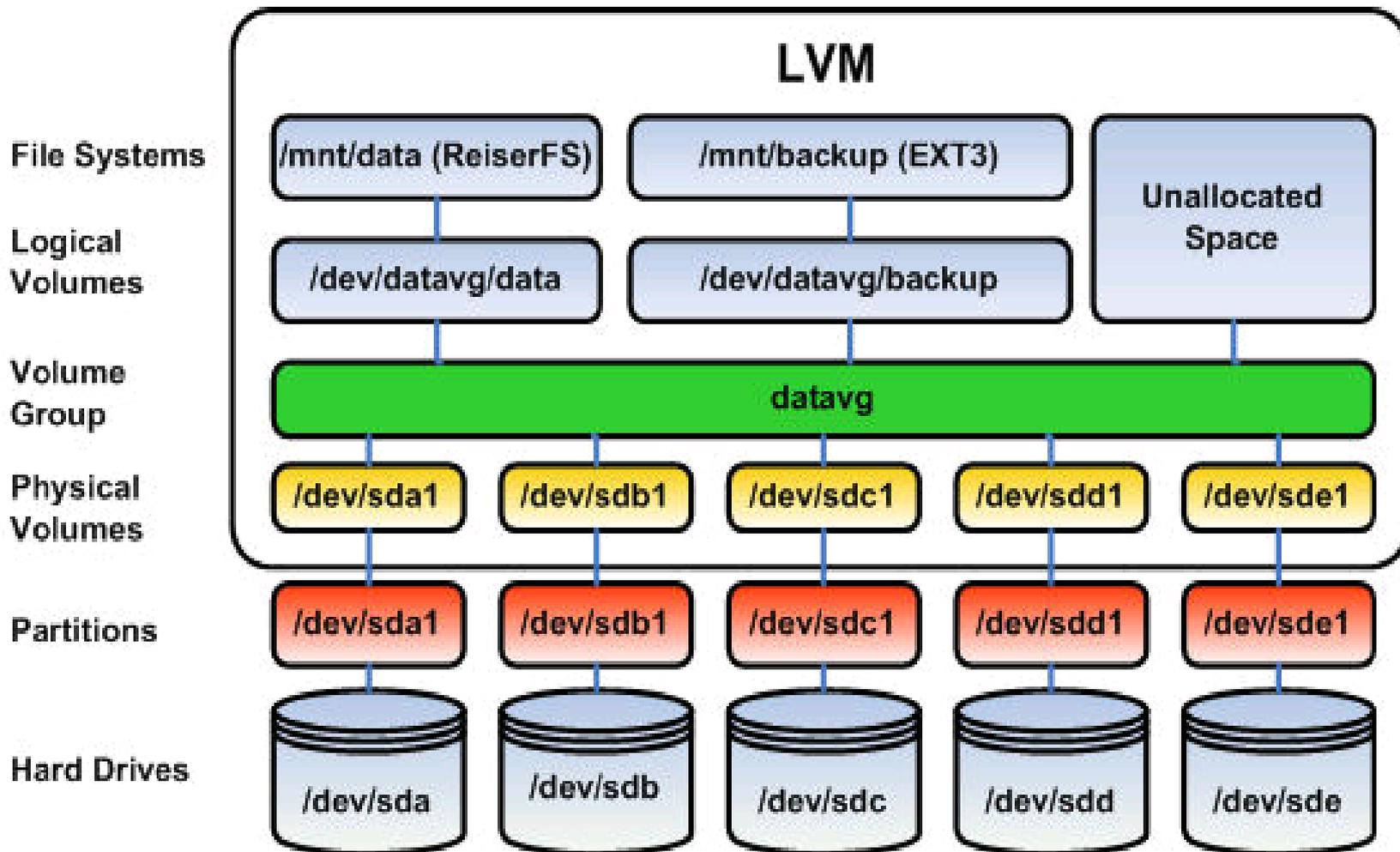
Organisation de LVM:

Les termes utilisés dans LVM pour décrire son organisation sont :

- **Volume physique** (physical volume) : chaque disque dur deviendra un volume physique au moment de la création.
- **Groupe de volumes** (volume group) : les volumes physiques sont regroupés dans un groupe de Volumes.
- **Volumes logiques** (logical volume) : un groupe de volumes est découpé en un ou plusieurs volumes logiques.

Linux Volume Manager (LVM)

Organisation de LVM:



Linux Volume Manager (LVM)

1-Création d'un volume physique:

On crée un volume physique grâce à la commande **pvcreate** (physical volume creation).

```
# pvcreate [-f] /dev/sdxx
```

-f : force la création du volume. A utiliser si le disque avait déjà été transformé en volume physique.

/dev/sdxx : fichier spécial du disque ou de la partition à transformer en volume physique
Par exemple, création d'un volume physique à partir du disque /dev/sda1

```
# pvcreate [-f] /dev/sda1  
pvcreate -- physical volume "/dev/hdb1" successfully created
```

Linux Volume Manager (LVM)

2- Création d'un groupe de volumes et ajout de volumes physiques:

Une fois qu'un volume physique est créé, il faut l'insérer dans un groupe de volumes. Cela se fait grâce à la commande **vgcreate**.

```
# vgcreate nom_du_volume /dev/sdxx
```

Où *nom_du_volume* : nom du groupe de volume - l'opération crée alors le répertoire /dev/nom_du_volume
/dev/sdxx : fichier spécial du volume physique

Par exemple on crée le groupe de volumes volume1 à partir du volume physique /dev/sda1

```
# pvcreate volume1 /dev/sda1
```

```
vgcreate -- INFO: using default physical extent size 4 MB
vgcreate -- INFO: maximum logical volume size is 255.99 Gigabyte
vgcreate -- doing automatic backup of volume group "volume1"
vgcreate -- volume group "volume1" successfully created and
activated
```

Linux Volume Manager (LVM)

3- Ajouter d'autres volumes physiques au groupe de volume:

Pour maintenant ajouter d'autres volumes physiques au groupe de volume, on utilise la commande **vgextend**

```
# vgextend nom_volume /dev/sdxx
```

nom_volume : nom du groupe de volume à redimensionner

/dev/sdxx : volume physique à ajouter ou retirer du groupe de volumes

Ajout de sda2 au groupe de volumes volume1

```
# vgextend volume1 /dev/sda2
```

```
vgextend -- INFO:maximum logical volume size is 255.99 Gigabyte  
Vgextend --doing automatic backup of volume group "volume1" vgextend  
Vgextend --volume group "volume1" successfully extended
```

Linux Volume Manager (LVM)

3- Création d'un volume logique:

Il s'agit de la dernière étape de la configuration de volumes LVM. Lorsqu'un groupe de volumes a été créé, il faut le séparer en un ou plusieurs volumes logiques.

La commande **lvcreate** permet de créer un volume logique de la taille qu'on désire avec le nom qu'on désire dans tel groupe de volumes.

Une fois cette étape passée, un « disque » LVM est créé dans
/dev/nom_du_groupe_de_volume/nom_du_volume_logique

```
# lvcreate -L tailleK|M|G [-n nom] <nom_volume>
```

Où :

- **L tailleK|M|G** : taille du volume logique exprimable en Ko, Mo ou Go
- **n nom** : nom du volume logique - l'opération crée un fichier spécial portant ce nom pour le volume logique et sera placé dans le répertoire /dev/nom_volume
- **<nom_volume>** : nom du groupe de volumes dans lequel sera créé le volume logique.

Linux Volume Manager (LVM)

3- Création d'un volume logique:

Exemple :

On crée un volume logique de 600 Mo nommé part1 dans le groupe de volume « volume1 »

```
# lvcreate -L 600 -n part1 volume1
lvcreate -- doing automatic backup of "volume1"
lvcreate -- logical volume "/dev/volume1/part1" successfully created
```

4- Informations sur les composants de LVM

On peut obtenir des informations sur chaque composant :

volume physique, groupe de volumes et volume logique grâce aux commandes :

```
# pvdisplay/dev/sdxx
#vgdisplay volume1
#lvdisplay /dev/volume1/part1
```

Linux Volume Manager (LVM)

3- Création de snapshot:

La création de snapshot avec LVM se fait de façon simple : on crée un nouveau volume logique qui contient le snapshot, on monte ce volume, on effectue sa sauvegarde, on démonte le volume puis on le détruit.

Exemple :

On crée ici le snapshot qui apparaîtra comme un duplicata du volume logique qu'on souhaite sauvegarder.

Il faut prévoir suffisamment de place pour ce volume logique (environ 500Mos) et spécifier à la création qu'il s'agit d'un volume logique

```
# lvcreate -L500M -s -n snapshot /dev/volume1/part1
lvcreate -- WARNING: the snapshot must be disabled if it gets full
lvcreate -- INFO: using default snapshot chunk size of 64 KB for
"/dev/volume1/snapshot"
lvcreate -- doing automatic backup of "volume1"
lvcreate -- logical volume "/dev/volume1/snapshot" successfully
created
```

Linux Volume Manager (LVM)

3- Création de snapshot:

Si il n'y a pas assez de place pour le snapshot, celui ci est désactivé.
Si le **système de fichier utilisé est XFS**, il faut utiliser la commande :

```
# xfs_freeze -f
```

Pour verrouiller le système de fichier avant la création du snapshot

```
# xfs_freeze -u
```

puis le déverrouiller avec

4- Montage du snapshot:

On crée le point de montage et on monte le snapshot.

```
# mkdir /mnt/snapshot  
# mount /dev/volume1/snapshot /mnt/snapshot  
mount: block device /dev/volume1/snapshot is write-protected,  
mounting read-only
```

Avec un système de fichier XFS il faut spécifier l'option `-onouuid,ro` à mount.

Linux Volume Manager (LVM)

5- Sauvegarde:

On utilise le mode de sauvegarde qu'on veut : tar, rsync, etc. :

```
# tar -cvzf /backup.tar.gz /mnt/snapshot
```

6- démontage du snapshot:

Une fois que la sauvegarde est finie, on a plus besoin du snapshot, on le détruit donc.

```
# umount /mnt/snapshot
# lvremove /dev/volume1/snapshot
lvremove -- do you really want to remove "/dev/volume1/snapshot"?
[y/n]: y
lvremove -- doing automatic backup of volume group "volume1"
lvremove -- logical volume "/dev/volume1/snapshot" successfully
removed
```

freeBSD et UFS

UFS2 est le système de fichier utilisé par défaut par FreeBSD à partir de sa version 5. Ce système de fichier permet la création de snapshot.

Ici le principe est différent de celui de LVM, il existe deux types d'utilisation :

- la sauvegarde du système de fichier avec l'outil dump
- la création d'un snapshot pour naviguer dans l'image figée à la date de création du snapshot

7- Sauvegarde avec kdump:

Le principe est de faire une sauvegarde complète d'un système de fichier grâce à l'outil dump.

On pourra plus tard restaurer les données avec la commande **restore**.

Attention, un système de fichier qu'on souhaite sauvegarder avec dump est un de ceux définis dans le fichier **/etc/fstab**.

Avant de faire le dump, créer un dossier « .snap » à la racine du système de fichier à sauvegarder.

Le propriétaire doit être « root », le groupe « operator » et le mode « 0770 »

Exemple : sauvegarde du système de fichier /home

```
# cd /home  
#mkdir .snap
```

freeBSD et UFS

Si le dossier n'a pas les bonnes propriétés de propriétaire, groupe, mode énoncées au dessus : les changer. Enfin, effectuer la sauvegarde effective :

dump de **/home** dans **fichier_de_sauvegarde**:

```
# dump -L -f fichier-de-sauvegarde /home
```

L'option **-L** permet de créer un **dump** basé sur un snapshot.

Pour plus d'informations, consulter les pages de manuel de **dump(8)** et **restore(8)**.

8- Exploration du snapshot :

Au départ on crée un snapshot du système de fichier que l'on souhaite sauvegarder. Le snapshot sera en fait un fichier.

Exemple : création du snapshot de **/home**

```
# mount -u -o snapshot /home/fichier_snapshot /home
```

Cela crée le snapshot « **fichier_snapshot** » du système de fichier **/home** grâce à l'option « **-o snapshot** » de la commande **mount**.

Attention, le fichier du snapshot doit être créé dans le système de fichier dont on crée le snapshot.

freeBSD et UFS

Il faut maintenant monter le snapshot pour pouvoir naviguer dans l'image figée.:

```
# mdconfig -a -t vnode -f /home/fichier_snapshot -u 1
# mkdir /mnt/image_figee
# mount -o ro /dev/md1 /mnt/image_figee
```

Avant de monter l'image, on crée un « disque dur virtuel » **/dev/md1** à partir de **fichier_snapshot** grâce à la commande **mdconfig**.

On monte ensuite ce disque virtuel en lecture seule (**option « -o ro »**) dans le dossier **image_figee**.

Une fois qu'on n'a plus besoin d'utiliser l'image figée du système de fichier, on démonte l'image, on désactive le disque virtuel et enfin on supprime le fichier du snapshot.

```
# umount /mnt/image_figee
# mkdir /mnt/image_figee
# rm -f /home/fichier_snapshot
```