

Dernière mise-à-jour : 2020/01/30 03:27

102.1 - Concevoir l'aménagement du disque (2/60)

Le système de fichiers de Linux est organisé autour d'une arborescence unique ayant un point de départ appelé la **racine**, représenté par le caractère **/**. En dessous de cette racine se trouvent des répertoires contenant fichiers et sous-répertoires. L'organisation des répertoires est conforme à un standard, appelé le **Linux File Hierarchy System**.

RHEL 5

```
[trainee@centos5 ~]$ ls -l
total 138
drwxr-xr-x  2 root root  4096 août 25 13:27 bin
drwxr-xr-x  4 root root  1024 août 25 14:03 boot
drwxr-xr-x 11 root root  3800 août 25 14:06 dev
drwxr-xr-x 102 root root 12288 août 25 14:06 etc
drwxr-xr-x  3 root root  4096 août 25 13:36 home
drwxr-xr-x 14 root root  4096 août 25 13:42 lib
drwx----- 2 root root 16384 août 25 13:16 lost+found
drwxr-xr-x  3 root root  4096 août 25 14:06 media
drwxr-xr-x  2 root root    0 août 25 14:06 misc
drwxr-xr-x  2 root root  4096 mai 11  2011 mnt
drwxr-xr-x  2 root root    0 août 25 14:06 net
drwxr-xr-x  4 root root  4096 août 25 13:44 opt
dr-xr-xr-x 143 root root    0 août 25 14:05 proc
drwxr-xr-x  4 root root  4096 août 25 14:01 root
drwxr-xr-x  2 root root 12288 août 25 14:03 sbin
drwxr-xr-x  4 root root    0 août 25 14:05 selinux
drwxr-xr-x  2 root root  4096 mai 11  2011 srv
drwxr-xr-x 11 root root    0 août 25 14:05 sys
```

```
drwxrwxrwt 14 root root 4096 août 25 14:09 tmp
drwxr-xr-x 14 root root 4096 août 25 13:23 usr
drwxr-xr-x 21 root root 4096 août 25 13:31 var
```

RHEL 6

```
[trainee@centos6 /]$ ls -l
total 98
dr-xr-xr-x.  2 root root  4096  9 août  12:52 bin
dr-xr-xr-x.  5 root root 1024  7 déc.  2014 boot
drwxr-xr-x. 19 root root 3820 25 août  11:29 dev
drwxr-xr-x. 119 root root 12288 25 août  11:28 etc
drwxr-xr-x.  3 root root  4096  3 mai   2013 home
dr-xr-xr-x. 20 root root 12288  9 août  12:52 lib
drwx-----. 2 root root 16384  3 mai   2013 lost+found
drwxr-xr-x.  2 root root  4096  7 déc.  2014 media
drwxr-xr-x.  2 root root    0 25 août  11:28 misc
drwxr-xr-x.  3 root root  4096  5 juil. 12:22 mnt
drwxr-xr-x.  2 root root    0 25 août  11:28 net
drwxr-xr-x.  6 root root  4096  7 déc.  2014 opt
dr-xr-xr-x. 154 root root    0 25 août  11:27 proc
dr-xr-x---. 10 root root  4096  9 août  12:58 root
dr-xr-xr-x.  2 root root 12288  9 août  12:52 sbin
drwxr-xr-x.  7 root root    0 25 août  11:27 selinux
drwxr-xr-x.  2 root root  4096 23 sept. 2011 srv
drwxr-xr-x. 13 root root    0 25 août  11:27 sys
drwxrwxrwt. 16 root root  4096 25 août  11:30 tmp
drwxr-xr-x. 13 root root  4096  3 mai   2013 usr
drwxr-xr-x. 22 root root  4096  9 août  12:50 var
```

RHEL 7

```
[trainee@centos7 ~]$ ls -l
total 32
lrwxrwxrwx.  1 root root    7 Mar  8 13:41 bin -> usr/bin
dr-xr-xr-x.  4 root root 4096 Jun  4 15:00 boot
drwxr-xr-x. 19 root root 3280 Jul  7 15:55 dev
drwxr-xr-x. 131 root root 8192 Jul 23 17:05 etc
drwxr-xr-x.  4 root root   47 Jul  5 14:11 home
lrwxrwxrwx.  1 root root    7 Mar  8 13:41 lib -> usr/lib
lrwxrwxrwx.  1 root root    9 Mar  8 13:41 lib64 -> usr/lib64
drwxr-xr-x.  2 root root    6 Jun 10 2014 media
drwxr-xr-x.  3 root root   18 Jul  5 13:57 mnt
drwxr-xr-x.  4 root root   47 Jun  4 09:36 opt
dr-xr-xr-x. 177 root root    0 Jul  7 15:53 proc
dr-xr-x---.  5 root root 4096 Aug 25 11:31 root
drwxr-xr-x. 35 root root 1100 Jul 23 15:40 run
lrwxrwxrwx.  1 root root    8 Mar  8 13:41 sbin -> usr/sbin
drwxr-xr-x.  2 root root    6 Jun 10 2014 srv
dr-xr-xr-x. 13 root root    0 Jul  7 15:53 sys
drwxrwxrwt. 25 root root 4096 Jul 23 15:40 tmp
drwxr-xr-x. 13 root root 4096 Mar  8 13:41 usr
drwxr-xr-x. 22 root root 4096 Jul  7 15:53 var
```

Linux File Hierarchy System

RHEL/CentOS 5 et 6

- **/bin** : est une abréviation de **binary** ou binaires. Il contient des programmes tels ls.
- **/boot** : contient les fichiers nécessaires au démarrage du système.
- **/cgroup** : utilisé par le nouveau système de *Control Groups*.

- **/dev** : contient les nœuds utilisés pour accéder à tout type de matériel tel `/dev/fd0` pour le lecteur de disquette. C'est le binaire `udev` qui se charge de créer et supprimer d'une manière dynamique les nœuds.
- **/etc** : contient des fichiers de configuration tels `passwd` pour les mots de passe et `fstab` qui est la liste des systèmes de fichiers à monter lors du démarrage du système.
- **/home** : contient les répertoires de chaque utilisateur sauf l'utilisateur `root`.
- **/lib** : contient les bibliothèques 32 bits communes utilisées par les programmes ainsi que les modules.
- **/lib64** : contient les bibliothèques 64 bits communes utilisées par les programmes ainsi que les modules.
- **/lost+found** : contient des fragments de fichiers endommagés et retrouvés par la commande `fsck`.
- **/media** : contient des répertoires pour chaque système de fichiers monté (accessible au système linux) tels `floppy`, `cdrom` etc.
- **/misc** : contient des points de montage pour chaque répertoire local monté par l'automounter.
- **/mnt** : contient des répertoires pour chaque système de fichiers monté temporairement par `root`.
- **/net** : contient des points de montage pour chaque répertoire réseau monté par l'automounter.
- **/opt** : contient des applications optionnelles.
- **/proc** : contient un système de fichiers virtuel qui extrait de la mémoire les informations en cours de traitement. Le contenu des fichiers est créé dynamiquement lors de la consultation. Seul `root` peut consulter la totalité des informations dans le répertoire `/proc`.
- **/root** : le home de `root`, l'administrateur système
- **/sbin** : contient des binaires, donc programmes, pour l'administration du système local.
- **/selinux** : contient des fichiers propres à l'implémentation de SELINUX.
- **/srv** : contient des données pour les **services** hébergés par le système tels `ftp`, bases de données, `web` etc.
- **/sys** : contient un système de fichiers virtuel dont le rôle est de décrire le matériel pour `udev`.
- **/tmp** : stocke des fichiers temporaires créés par des programmes.
- **/usr** : contient des commandes des utilisateurs dans `/usr/bin`, les HOWTO dans `/usr/share/doc`, les manuels dans `/usr/share/man` ainsi que d'autres entrées majeures.
- **/var** : contient des fichiers de taille variable.

RHEL/CentOS 7

- **/bin** : est une abréviation de **binary** ou binaires. Il contient des programmes tels `ls`. Sous RHEL/CentOS 7 il s'agit d'un lien symbolique qui pointe vers `/usr/bin`.
- **/boot** : contient les fichiers nécessaires au démarrage du système.
- **/dev** : contient les nœuds utilisés pour accéder à tout type de matériel tel `/dev/fd0` pour le lecteur de disquette. C'est le binaire `udev` qui se charge de créer et supprimer d'une manière dynamique les nœuds.
- **/etc** : contient des fichiers de configuration tels `passwd` pour les mots de passe et `fstab` qui est la liste des systèmes de fichiers à monter lors du

démarrage du système.

- **/home** : contient les répertoires de chaque utilisateur sauf l'utilisateur root.
- **/lib** : contient les bibliothèques 32 bits communes utilisées par les programmes ainsi que les modules. Sous RHEL/CentOS 7 il s'agit d'un lien symbolique qui pointe vers /usr/lib.
- **/lib64** : contient les bibliothèques 64 bits communes utilisées par les programmes ainsi que les modules. Sous RHEL/CentOS 7 il s'agit d'un lien symbolique qui pointe vers /usr/lib64.
- **/lost+found** : contient des fragments de fichiers endommagés et retrouvés par la commande *fsck*.
- **/media** : contient des répertoires pour chaque système de fichiers monté (accessible au système linux) tels floppy, cdrom etc.
- **/mnt** : contient des répertoires pour chaque système de fichiers monté temporairement par root.
- **/opt** : contient des applications optionnelles.
- **/proc** : contient un système de fichiers virtuel qui extrait de la mémoire les informations en cours de traitement. Le contenu des fichiers est créé dynamiquement lors de la consultation. Seul root peut consulter la totalité des informations dans le répertoire /proc.
- **/root** : le home de root, l'administrateur système.
- **/run** : remplace le répertoire /var/run. Sous RHEL/CentOS 7 /var/run est un lien symbolique qui pointe vers /run.
- **/sbin** : contient des binaires, donc programmes, pour l'administration du système local. Sous RHEL/CentOS 7 il s'agit d'un lien symbolique qui pointe vers /usr/sbin.
- **/srv** : contient des données pour les **services** hébergés par le système tels ftp, bases de données, web etc.
- **/sys** : contient un système de fichiers virtuel dont le rôle est de décrire le matériel pour udev.
- **/tmp** : stocke des fichiers temporaires créés par des programmes.
- **/usr** : contient des commandes des utilisateurs dans /usr/bin, les HOWTO dans /usr/share/doc, les manuels dans /usr/share/man ainsi que d'autres entrées majeures.
- **/var** : contient des fichiers de taille variable.

Types de Fichiers

Il existe trois types majeurs de fichier sous le système Linux :

- les fichiers normaux (ordinary files)
 - les répertoires (directories)
 - les fichiers spéciaux (special files ou Devices)
-

Le fichiers normaux sont des fichiers textes, des tableaux ou des exécutables.

La longueur du nom de fichier est limité à 255 caractères.

Il y a une distinction entre les majuscules et les minuscules.

Si le nom d'un fichier commence par un `.`, le fichier devient caché.

Périphériques de stockage

Les unités de stockage sous Linux sont référencées par un des fichiers se trouvant dans le répertoire **/dev** :

- `hd[a-d]`
 - Les disques IDE et les lecteurs ATAPI
- `sd[a-z]`
 - Les disques SCSI et SATA
- `scd[0-7]`
 - Les CDRoms SCSI
- `xd[a-d]`
 - Les premiers disques sur IBM XT
- `fd[0-7]`
 - Les lecteurs de disquettes
- `st[0-7]`
 - Les lecteurs de bandes SCSI qui **supportent** le rembobinage
- `nst[0-7]`
 - Les lecteurs de bandes SCSI qui ne supportent **pas** le rembobinage
- `rmt8`, `rmt16`, `tape-d`, `tape-reset`
 - Les lecteurs QIC-80
- `ram[0-15]`
 - Les disques virtuels. Ils sont supprimés à l'extinction de la machine. Un de ces disques est utilisé par le système pour monter l'image d'un disque racine défini par le fichier **initrd** au démarrage de la machine
- Périphériques **loop**
 - Il existe 16 unités loop qui sont utilisés pour accéder en mode bloc à un système de fichiers contenu dans un fichier, par exemple, une

image **iso**

- md[x]
 - Un volume **RAID** logiciel
- vg[x]
 - Un groupe de volumes
- lv[x]
 - Un volume logique

Partitions

Un PC comportent en règle générale 2 **contrôleurs** de disque, chacun capable de gérer 2 disques, un **maître** et un **esclave**. Les disques attachés à ces contrôleurs comportent des noms différents pour pouvoir les distinguer :

- Contrôleur 0
 - Maître
 - **hda** - disque IDE
 - **sda** - disque SATA ou SCSI
 - Esclave
 - **hdb** - disque IDE
 - **sdb** - disque SATA ou SCSI
- Contrôleur 1
 - Maître
 - **hdc** - disque IDE
 - **sd** - disque SATA ou SCSI
 - Esclave
 - **hdd** - disque IDE
 - **sdd** - disque SATA ou SCSI

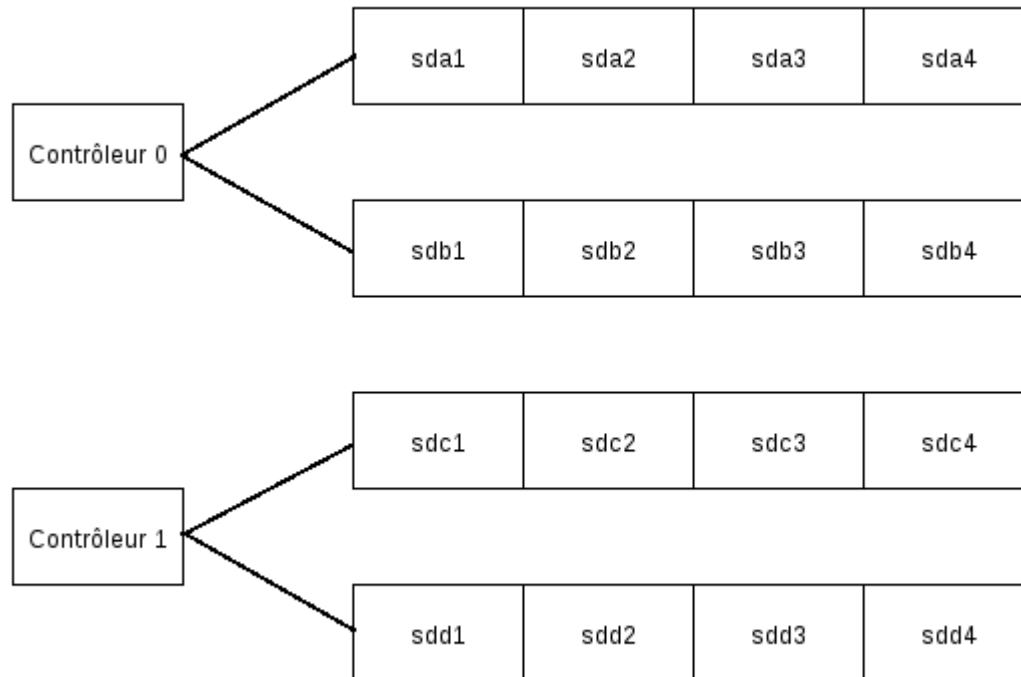
Un disque peut comporter trois types de partitions :

- **Partitions primaires**,
 - Maximum de **4**. En effet, la Table des Partitions est grande de 64 octets. Il faut 16 octets pour coder une partition.
- **Partitions Etendues**,

- Généralement une seule partition étendue par disque. Elle contient des **Lecteurs Logiques** aussi appelés des partitions,
 - **Lecteurs Logiques**.

Les 4 partitions primaires sont numérotées de 1 à 4. Par exemple :

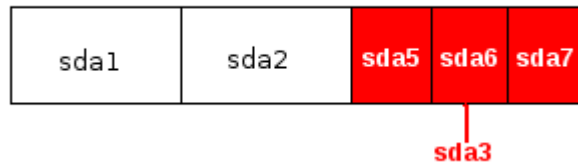
- **hda1, hda2, hda3** et **hda4** pour le premier disque **IDE** sur le premier contrôleur de disque,
- **sda1, sda2, sda3** et **sda4** pour le premier disque **SCSI** ou **SATA** sur le premier contrôleur de disque.



Une partition étendue prend la place d'une partition primaire et les lecteurs logiques qui s'y trouvent commencent à partir de **hda5** ou de **sda5**.

Pour clarifier ceci, considérons un disque **SATA** contenant deux partitions primaires, une seule partition étendue et 3 lecteurs logiques. Dans ce cas, les deux premières partitions sont **sda1** et **sda2**, la partition étendue prend la place de la troisième partition primaire, la **sda3** et s'appelle ainsi tandis que la quatrième partition primaire est inexistante.

Les lecteurs logiques commençant à **sda5**, nous obtenons la liste de partitions suivante : sda1, sda2, sda5, sda6, sda7. Notez que la sda3 ne peut pas être utilisée en tant que partition car elle est cachée par les lecteurs sda5, sda6 et sda7.



Le nombre de partitions sur un disque est limité :

- **IDE**,
 - Jusqu'à **63**,
- **SCSI**,
 - Jusqu'à **15**,
- **Disques utilisant l'API libata**,
 - Jusqu'à **15**.



Important : Ces limites peuvent être dépassées en utilisant la gestion **LVM** (*Logical Volume Management*).

Partitionnement

LAB #1 - Partitionnement de votre Disque sous RHEL/CentOS 7 avec fdisk

Pour procéder au partitionnement de votre disque ou de vos disques, RHEL/CentOS 7 possède l'outil dénommé **fdisk**.

Lancez fdisk en fournissant en argument le fichier de référence de votre premier disque dur, par exemple :

```
[root@centos7 ~]# fdisk /dev/sda
```

```
Welcome to fdisk (util-linux 2.23.2).
```

```
Changes will remain in memory only, until you decide to write them.  
Be careful before using the write command.
```

```
Command (m for help):
```

Tapez ensuite la lettre **m** puis pour obtenir le menu :

```
Command (m for help): m
```

```
Command action
```

```
  a  toggle a bootable flag  
  b  edit bsd disklabel  
  c  toggle the dos compatibility flag  
  d  delete a partition  
  g  create a new empty GPT partition table  
  G  create an IRIX (SGI) partition table  
  l  list known partition types  
  m  print this menu  
  n  add a new partition  
  o  create a new empty DOS partition table  
  p  print the partition table  
  q  quit without saving changes  
  s  create a new empty Sun disklabel  
  t  change a partition's system id  
  u  change display/entry units  
  v  verify the partition table  
  w  write table to disk and exit  
  x  extra functionality (experts only)
```

```
Command (m for help):
```

Pour créer une nouvelle partition, vous devez utiliser la commande **n**.

Créez donc les partitions suivantes sur votre disque :

Partition	Type	Taille de la Partition
/dev/sda4	Extended	Du premier cylindre disponible au dernier cylindre du disque
/dev/sda5	Logique	500 Mo
/dev/sda6	Logique	200 Mo
/dev/sda7	Logique	300 Mo
/dev/sda8	Logique	500 Mo
/dev/sda9	Logique	400 Mo
/dev/sda10	Logique	500 Mo
/dev/sda11	Logique	500 Mo
/dev/sda12	Logique	200 Mo

Créez d'abord la partition étendue :

```
Command (m for help): n
Partition type:
   p   primary (3 primary, 0 extended, 1 free)
   e   extended
Select (default e): e
Selected partition 4
First sector (20891648-41943039, default 20891648):
Using default value 20891648
Last sector, +sectors or +size{K,M,G} (20891648-41943039, default 41943039):
Using default value 41943039
Partition 4 of type Extended and of size 10 GiB is set

Command (m for help):
```

Créez ensuite les autres partitions l'une après l'autre :

```
Command (m for help): n
All primary partitions are in use
```

Adding logical partition 5

First sector (20893696-41943039, default 20893696):

Using default value 20893696

Last sector, +sectors or +size{K,M,G} (20893696-41943039, default 41943039): +500M

Partition 5 of type Linux and of size 500 MiB is set

Command (m for help): n

All primary partitions are in use

Adding logical partition 6

First sector (21919744-41943039, default 21919744):

Using default value 21919744

Last sector, +sectors or +size{K,M,G} (21919744-41943039, default 41943039): +200M

Partition 6 of type Linux and of size 200 MiB is set

Command (m for help): n

All primary partitions are in use

Adding logical partition 7

First sector (22331392-41943039, default 22331392):

Using default value 22331392

Last sector, +sectors or +size{K,M,G} (22331392-41943039, default 41943039): +300M

Partition 7 of type Linux and of size 300 MiB is set

Command (m for help): n

All primary partitions are in use

Adding logical partition 8

First sector (22947840-41943039, default 22947840):

Using default value 22947840

Last sector, +sectors or +size{K,M,G} (22947840-41943039, default 41943039): +500M

Partition 8 of type Linux and of size 500 MiB is set

Command (m for help): n

All primary partitions are in use

Adding logical partition 9

First sector (23973888-41943039, default 23973888):

```
Using default value 23973888
Last sector, +sectors or +size{K,M,G} (23973888-41943039, default 41943039): +400M
Partition 9 of type Linux and of size 400 MiB is set

Command (m for help): n
All primary partitions are in use
Adding logical partition 10
First sector (24795136-41943039, default 24795136):
Using default value 24795136
Last sector, +sectors or +size{K,M,G} (24795136-41943039, default 41943039): +500M
Partition 10 of type Linux and of size 500 MiB is set

Command (m for help): n
All primary partitions are in use
Adding logical partition 11
First sector (25821184-41943039, default 25821184):
Using default value 25821184
Last sector, +sectors or +size{K,M,G} (25821184-41943039, default 41943039): +500M
Partition 11 of type Linux and of size 500 MiB is set

Command (m for help): n
All primary partitions are in use
Adding logical partition 12
First sector (26847232-41943039, default 26847232):
Using default value 26847232
Last sector, +sectors or +size{K,M,G} (26847232-41943039, default 41943039): +200M
Partition 12 of type Linux and of size 200 MiB is set

Command (m for help):
```

Tapez ensuite la lettre **p** puis pour visualiser la nouvelle table des partitions. Vous obtiendrez un résultat similaire à celui-ci :

```
Command (m for help): p
```

```
Disk /dev/sda: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000f2006
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	411647	204800	83	Linux
/dev/sda2		411648	14747647	7168000	83	Linux
/dev/sda3		14747648	20891647	3072000	82	Linux swap / Solaris
/dev/sda4		20891648	41943039	10525696	5	Extended
/dev/sda5		20893696	21917695	512000	83	Linux
/dev/sda6		21919744	22329343	204800	83	Linux
/dev/sda7		22331392	22945791	307200	83	Linux
/dev/sda8		22947840	23971839	512000	83	Linux
/dev/sda9		23973888	24793087	409600	83	Linux
/dev/sda10		24795136	25819135	512000	83	Linux
/dev/sda11		25821184	26845183	512000	83	Linux
/dev/sda12		26847232	27256831	204800	83	Linux



Important : Chaque bloc fait 1 024 octets. Chaque secteur fait 512 octets. Quand la partition contient un nombre impair de secteurs, celle-ci est marquée avec un +. Ceci implique que le dernier secteur de 512 octets est effectivement perdu.

Ecrivez la table des partitions sur disque et exécutez la commande **partprobe** :

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
```

```
WARNING: Re-reading the partition table failed with error 16: Device or resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
[root@centos7 ~]# partprobe
```

Lancez fdisk puis tapez ensuite la lettre **p** puis pour visualiser la table des partitions actuelle :

```
[root@centos7 ~]# fdisk /dev/sda
Welcome to fdisk (util-linux 2.23.2).
```

```
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
```

```
Command (m for help): p
```

```
Disk /dev/sda: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000f2006
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	411647	204800	83	Linux
/dev/sda2		411648	14747647	7168000	83	Linux
/dev/sda3		14747648	20891647	3072000	82	Linux swap / Solaris
/dev/sda4		20891648	41943039	10525696	5	Extended
/dev/sda5		20893696	21917695	512000	83	Linux
/dev/sda6		21919744	22329343	204800	83	Linux
/dev/sda7		22331392	22945791	307200	83	Linux
/dev/sda8		22947840	23971839	512000	83	Linux
/dev/sda9		23973888	24793087	409600	83	Linux

/dev/sda10	24795136	25819135	512000	83	Linux
/dev/sda11	25821184	26845183	512000	83	Linux
/dev/sda12	26847232	27256831	204800	83	Linux

Command (m for help):

Pour supprimer une partition, utilisez la commande **d** puis ↵ Entrée. fdisk vous demandera le numéro de la partition à supprimer, par exemple :

Command (m for help): d

Partition number (1-12, default 12): 12

Partition 12 is deleted

Command (m for help): p

Disk /dev/sda: 21.5 GB, 21474836480 bytes, 41943040 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk label type: dos

Disk identifier: 0x000f2006

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	411647	204800	83	Linux
/dev/sda2		411648	14747647	7168000	83	Linux
/dev/sda3		14747648	20891647	3072000	82	Linux swap / Solaris
/dev/sda4		20891648	41943039	10525696	5	Extended
/dev/sda5		20893696	21917695	512000	83	Linux
/dev/sda6		21919744	22329343	204800	83	Linux
/dev/sda7		22331392	22945791	307200	83	Linux
/dev/sda8		22947840	23971839	512000	83	Linux
/dev/sda9		23973888	24793087	409600	83	Linux
/dev/sda10		24795136	25819135	512000	83	Linux
/dev/sda11		25821184	26845183	512000	83	Linux

Command (m for help):

A ce stade, la partition n'a **pas** été réellement supprimée. En effet, vous avez la possibilité de sortir de fdisk en utilisant la commande **q**.

Tapez donc q pour sortir de fdisk puis relancez fdisk. Vous obtiendrez un résultat similaire à celui-ci :

Command (m for help): q

```
[root@centos7 ~]# fdisk /dev/sda
Welcome to fdisk (util-linux 2.23.2).
```

```
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
```

Command (m for help): p

```
Disk /dev/sda: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000f2006
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	411647	204800	83	Linux
/dev/sda2		411648	14747647	7168000	83	Linux
/dev/sda3		14747648	20891647	3072000	82	Linux swap / Solaris
/dev/sda4		20891648	41943039	10525696	5	Extended
/dev/sda5		20893696	21917695	512000	83	Linux
/dev/sda6		21919744	22329343	204800	83	Linux
/dev/sda7		22331392	22945791	307200	83	Linux
/dev/sda8		22947840	23971839	512000	83	Linux
/dev/sda9		23973888	24793087	409600	83	Linux

```
/dev/sda10    24795136    25819135    512000    83    Linux
/dev/sda11    25821184    26845183    512000    83    Linux
/dev/sda12    26847232    27256831    204800    83    Linux
```

```
Command (m for help):
```

LAB #2 - Modifier les Drapeaux des Partitions avec fdisk

Afin de mettre en place un RAID logiciel ou un volume logique, il est nécessaire de modifier les types de systèmes de fichiers sur les partitions créées.

Modifiez donc les nouvelles partitions à l'aide de la commande **t** de **fdisk** selon le tableau ci-dessous :

Taille de la Partition	Système de Fichiers
500 Mo	RAID (fd)
200 Mo	Linux LVM (8e)
300 Mo	Linux LVM (8e)
500 Mo	RAID (fd)
400 Mo	Linux LVM (8e)
500 Mo	RAID (fd)
500 Mo	RAID (fd)
200 Mo	Inchangé

Vous obtiendrez un résultat similaire à celui-ci :

```
[root@centos7 ~]# fdisk /dev/sda
Welcome to fdisk (util-linux 2.23.2).
```

```
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
```

```
Command (m for help): p
```

```
Disk /dev/sda: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000f2006
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	411647	204800	83	Linux
/dev/sda2		411648	14747647	7168000	83	Linux
/dev/sda3		14747648	20891647	3072000	82	Linux swap / Solaris
/dev/sda4		20891648	41943039	10525696	5	Extended
/dev/sda5		20893696	21917695	512000	83	Linux
/dev/sda6		21919744	22329343	204800	83	Linux
/dev/sda7		22331392	22945791	307200	83	Linux
/dev/sda8		22947840	23971839	512000	83	Linux
/dev/sda9		23973888	24793087	409600	83	Linux
/dev/sda10		24795136	25819135	512000	83	Linux
/dev/sda11		25821184	26845183	512000	83	Linux
/dev/sda12		26847232	27256831	204800	83	Linux

```
Command (m for help): t
Partition number (1-12, default 12): 5
Hex code (type L to list all codes): fd
Changed type of partition 'Linux' to 'Linux raid autodetect'
```

```
Command (m for help): t
Partition number (1-12, default 12): 6
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'
```

```
Command (m for help): t
Partition number (1-12, default 12): 7
Hex code (type L to list all codes): 8e
```

Changed type of partition 'Linux' to 'Linux LVM'

```
Command (m for help): t
Partition number (1-12, default 12): 8
Hex code (type L to list all codes): fd
Changed type of partition 'Linux' to 'Linux raid autodetect'
```

```
Command (m for help): t
Partition number (1-12, default 12): 9
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'
```

```
Command (m for help): t
Partition number (1-12, default 12): 10
Hex code (type L to list all codes): fd
Changed type of partition 'Linux' to 'Linux raid autodetect'
```

```
Command (m for help): t
Partition number (1-12, default 12): 11
Hex code (type L to list all codes): fd
Changed type of partition 'Linux' to 'Linux raid autodetect'
```

```
Command (m for help):
```

A l'issu des modifications, vous obtiendrez un résultat similaire à celui-ci :

```
Command (m for help): p

Disk /dev/sda: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000f2006
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	2048	411647	204800	83	Linux
/dev/sda2		411648	14747647	7168000	83	Linux
/dev/sda3		14747648	20891647	3072000	82	Linux swap / Solaris
/dev/sda4		20891648	41943039	10525696	5	Extended
/dev/sda5		20893696	21917695	512000	fd	Linux raid autodetect
/dev/sda6		21919744	22329343	204800	8e	Linux LVM
/dev/sda7		22331392	22945791	307200	8e	Linux LVM
/dev/sda8		22947840	23971839	512000	fd	Linux raid autodetect
/dev/sda9		23973888	24793087	409600	8e	Linux LVM
/dev/sda10		24795136	25819135	512000	fd	Linux raid autodetect
/dev/sda11		25821184	26845183	512000	fd	Linux raid autodetect
/dev/sda12		26847232	27256831	204800	83	Linux

Command (m for help):

Pour écrire la nouvelle table des partitions sur disque, vous devez utiliser la commande **w** puis la commande **partprobe** :

Command (m for help): w

The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.

The kernel still uses the old table. The new table will be used at

the next reboot or after you run partprobe(8) or kpartx(8)

Syncing disks.

[root@centos7 ~]# partprobe

Options de la Commande fdisk

Les options de cette commande sont :

```
[root@centos7 ~]# fdisk --help
fdisk: invalid option -- '-'
Usage:
  fdisk [options] <disk>      change partition table
  fdisk [options] -l <disk>   list partition table(s)
  fdisk -s <partition>       give partition size(s) in blocks

Options:
  -b <size>                  sector size (512, 1024, 2048 or 4096)
  -c[=<mode>]                compatible mode: 'dos' or 'nondos' (default)
  -h                          print this help text
  -u[=<unit>]                display units: 'cylinders' or 'sectors' (default)
  -v                          print program version
  -C <number>                specify the number of cylinders
  -H <number>                specify the number of heads
  -S <number>                specify the number of sectors per track
```

Logical Volume Manager (LVM)

LAB #3 - Volumes Logiques Linéaires

Afin de mettre en place le LVM, vous avez besoin du paquet **lvm2** et du paquet **device-mapper**.

Nous allons travailler sous RHEL/CentOS 7 avec les partitions suivantes :

/dev/sda6	21919744	22329343	204800	8e	Linux LVM
/dev/sda7	22331392	22945791	307200	8e	Linux LVM
/dev/sda9	23973888	24793087	409600	8e	Linux LVM

Pour initialiser le LVM saisissez la commande suivante :

```
[root@centos7 ~]# vgscan
Reading all physical volumes. This may take a while...
```

Les options de la commande **vgscan** sont :

```
[root@centos7 ~]# vgscan --help
vgscan: Search for all volume groups
```

```
vgscan  [--cache]
        [--commandprofile ProfileName]
        [-d|--debug]
        [-h|--help]
        [--ignorelockingfailure]
        [--mknodes]
        [-P|--partial]
        [-v|--verbose]
        [--version]
```

Physical Volume (PV)

Pour créer le **PV** il convient d'utiliser la commande **pvcreate** :

```
[root@centos7 ~]# pvcreate /dev/sda6 /dev/sda7 /dev/sda9
Physical volume "/dev/sda6" successfully created
Physical volume "/dev/sda7" successfully created
Physical volume "/dev/sda9" successfully created
```

Les options de la commande **pvcreate** sont :

```
[root@centos7 ~]# pvcreate --help
pvcreate: Initialize physical volume(s) for use by LVM
```

```
pvccreate
  [--norestorefile]
  [--restorefile file]
  [--commandprofile ProfileName]
  [-d|--debug]
  [-f[f]|--force [--force]]
  [-h|-?|--help]
  [--labelsector sector]
  [-M|--metadatatype 1|2]
  [--pvmetadatacopies #copies]
  [--bootloaderareaseize BootLoaderAreaSize[bBsSkKmMgGtTpPeE]]
  [--metadatasize MetadataSize[bBsSkKmMgGtTpPeE]]
  [--dataalignment Alignment[bBsSkKmMgGtTpPeE]]
  [--dataalignmentoffset AlignmentOffset[bBsSkKmMgGtTpPeE]]
  [--setphysicalvolumesize PhysicalVolumeSize[bBsSkKmMgGtTpPeE]]
  [-t|--test]
  [-u|--uuid uuid]
  [-v|--verbose]
  [-y|--yes]
  [-Z|--zero {y|n}]
  [--version]
  PhysicalVolume [PhysicalVolume...]
```

Pour visualiser le PV il convient d'utiliser la commande **pvdiskdisplay** :

```
[root@centos7 ~]# pvdiskdisplay /dev/sda6 /dev/sda7 /dev/sda9
"/dev/sda6" is a new physical volume of "200.00 MiB"
--- NEW Physical volume ---
PV Name                /dev/sda6
VG Name
PV Size                200.00 MiB
Allocatable            NO
PE Size                0
Total PE               0
```

```
Free PE          0
Allocated PE      0
PV UUID          9o7SOM-NU2B-dKzi-crvR-rJej-kw20-QtY0t5
"/dev/sda9" is a new physical volume of "400.00 MiB"
--- NEW Physical volume ---
PV Name          /dev/sda9
VG Name
PV Size          400.00 MiB
Allocatable      NO
PE Size          0
Total PE         0
Free PE          0
Allocated PE      0
PV UUID          vu7nY2-ac3k-Hp19-gFVQ-ny0I-sZoY-Sykbcbw
"/dev/sda7" is a new physical volume of "300.00 MiB"
--- NEW Physical volume ---
PV Name          /dev/sda7
VG Name
PV Size          300.00 MiB
Allocatable      NO
PE Size          0
Total PE         0
Free PE          0
Allocated PE      0
PV UUID          HpElCF-L3x7-pjnl-IgmA-CeNc-ZBCu-5sk3AZ
```

Les options de la commande **pvdisk** sont :

```
[root@centos7 ~]# pvdisk --help
pvdisk: Display various attributes of physical volume(s)

pvdisk
  [-c|--colon]
  [--commandprofile ProfileName]
```

```
[-d|--debug]
[-h|--help]
[--ignorelockingfailure]
[--ignoreskippedcluster]
[-m|--maps]
[--nosuffix]
[--readonly]
[-s|--short]
[--units hHbBsSkKmMgGtTpPeE]
[-v|--verbose]
[--version]
[PhysicalVolumePath [PhysicalVolumePath...]]
```

```
pvdiskdisplay --columns|-C
  [--aligned]
  [-a|--all]
  [--binary]
  [--commandprofile ProfileName]
  [-d|--debug]
  [-h|--help]
  [--ignorelockingfailure]
  [--ignoreskippedcluster]
  [--noheadings]
  [--nosuffix]
  [-o|--options [+]Field[,Field]]
  [-O|--sort [+|-]key1[,[+|-]key2[,...]]]
  [-S|--select Selection]
  [--readonly]
  [--separator Separator]
  [--unbuffered]
  [--units hHbBsSkKmMgGtTpPeE]
  [-v|--verbose]
  [--version]
  [PhysicalVolumePath [PhysicalVolumePath...]]
```

Volume Group (VG) et Physical Extent (PE)

Pour créer un Volume Group dénommé **vg0**, il convient d'utiliser la commande **vgcreate** :

```
[root@centos7 ~]# vgcreate -s 8M vg0 /dev/sda6 /dev/sda7 /dev/sda9
Volume group "vg0" successfully created
```

Les options de la commande **vgcreate** sont :

```
[root@centos7 ~]# vgcreate --help
vgcreate: Create a volume group

vgcreate
  [-A|--autobackup {y|n}]
  [--addtag Tag]
  [--alloc AllocationPolicy]
  [-c|--clustered {y|n}]
  [--commandprofile ProfileName]
  [-d|--debug]
  [-h|--help]
  [-l|--maxlogicalvolumes MaxLogicalVolumes]
  [--metadataprofile ProfileName]
  [-M|--metadatatype 1|2]
  [--[vg]metadatacopies #copies]
  [-p|--maxphysicalvolumes MaxPhysicalVolumes]
  [-s|--physicalextentsize PhysicalExtentSize[bBsSkKmMgGtTpPeE]]
  [-t|--test]
  [-v|--verbose]
  [--version]
  [-y|--yes]
  [ PHYSICAL DEVICE OPTIONS ]
  VolumeGroupName PhysicalDevicePath [PhysicalDevicePath...]
```

Pour afficher les informations concernant **vg0**, il convient d'utiliser la commande **vgdisplay** :

```
[root@centos7 ~]# vgdisplay vg0
--- Volume group ---
VG Name                vg0
System ID
Format                 lvm2
Metadata Areas         3
Metadata Sequence No   1
VG Access               read/write
VG Status               resizable
MAX LV                 0
Cur LV                 0
Open LV                 0
Max PV                 0
Cur PV                 3
Act PV                 3
VG Size                 880.00 MiB
PE Size                 8.00 MiB
Total PE                110
Alloc PE / Size         0 / 0
Free PE / Size          110 / 880.00 MiB
VG UUID                 0Q3Moh-KXD8-Ca2H-B3ry-imT8-e4SG-NYSHwi
```

Les options de la commande **vgdisplay** sont :

```
[root@centos7 ~]# vgdisplay --help
vgdisplay: Display volume group information

vgdisplay
  [-A|--activevolumegroups]
  [-c|--colon | -s|--short | -v|--verbose]
  [--commandprofile ProfileName]
  [-d|--debug]
```

```
[-h|--help]
[--ignorelockingfailure]
[--ignoreskippedcluster]
[--nosuffix]
[-P|--partial]
[--readonly]
[--units hHbBsSkKmMgGtTpPeE]
[--version]
[VolumeGroupName [VolumeGroupName...]]
```

```
vgdisplay --columns|-C
  [--aligned]
  [--binary]
  [--commandprofile ProfileName]
  [-d|--debug]
  [-h|--help]
  [--ignorelockingfailure]
  [--ignoreskippedcluster]
  [--noheadings]
  [--nosuffix]
  [-o|--options [+]Field[,Field]]
  [-O|--sort [+|-]key1[,[+|-]key2[,...]]]
  [-P|--partial]
  [-S|--select Selection]
  [--readonly]
  [--separator Separator]
  [--unbuffered]
  [--units hHbBsSkKmMgGtTpPeE]
  [--verbose]
  [--version]
  [VolumeGroupName [VolumeGroupName...]]
```

Logical Volumes (LV)

Pour créer un **Logical Volume** dénommé **lv0** dans le **Volume Group vg0**, il convient d'utiliser la commande **lvcreate** :

```
[root@centos7 ~]# lvcreate -L 350 -n lv0 vg0
Rounding up size to full physical extent 352.00 MiB
Logical volume "lv0" created.
```



Notez que la taille du LV est un multiple du PE.

Les options de la commande **lvcreate** sont :

```
[root@centos7 ~]# lvcreate --help
lvcreate: Create a logical volume

lvcreate
  [-A|--autobackup {y|n}]
  [-a|--activate [a|e|l]{y|n}]
  [--addtag Tag]
  [--alloc AllocationPolicy]
  [-H|--cache
    [--cachemode {writeback|writethrough}]]
  [--cachepool CachePoolLogicalVolume{Name|Path}]
  [-c|--chunksize ChunkSize]
  [-C|--contiguous {y|n}]
  [--commandprofile ProfileName]
  [-d|--debug]
  [-h|-?|--help]
  [--errorwhenfull {y|n}]
  [--ignoremonitoring]
```

```
[--monitor {y|n}]
[-i|--stripes Stripes [-I|--stripesize StripeSize]]
[-k|--setactivationskip {y|n}]
[-K|--ignoreactivationskip]
{-l|--extents LogicalExtentsNumber[%{VG|PVS|FREE}] |
-L|--size LogicalVolumeSize[bBsSkKmMgGtTpPeE]}
[-M|--persistent {y|n}] [-j|--major major] [--minor minor]
[--metadataprofile ProfileName]
[-m|--mirrors Mirrors [--nosync]
  [--mirrorlog {disk|core|mirrored}|--corelog]]
[-n|--name LogicalVolumeName]
[--noudevsync]
[-p|--permission {r|rw}]
[--poolmetadatasize MetadataSize[bBsSkKmMgG]]
[--poolmetadataspare {y|n}]
[--[raid]minrecoveryrate Rate]
[--[raid]maxrecoveryrate Rate]
[-r|--readahead {ReadAheadSectors|auto|none}]
[-R|--regionsize MirrorLogRegionSize]
[-T|--thin
  [--discards {ignore|nopassdown|passdown}]]
[--thinpool ThinPoolLogicalVolume{Name|Path}]
[-t|--test]
[--type VolumeType]
[-v|--verbose]
[-W|--wipesignatures {y|n}]
[-Z|--zero {y|n}]
[--version]
VolumeGroupName [PhysicalVolumePath...]
```

lvcreate

```
{ {-s|--snapshot} OriginalLogicalVolume[Path] |
  [-s|--snapshot] VolumeGroupName[Path] -V|--virtualsize VirtualSize}
{-H|--cache} VolumeGroupName[Path][OriginalLogicalVolume]
```

```
{-T|--thin} VolumeGroupName[Path][PoolLogicalVolume]
        -V|--virtualsize VirtualSize}
[-A|--autobackup {y|n}]
[--addtag Tag]
[--alloc AllocationPolicy]
[--cachepolicy Policy] [--cachesettings Key=Value]
[-c|--chunksize]
[-C|--contiguous {y|n}]
[--commandprofile ProfileName]
[-d|--debug]
[--discards {ignore|nopassdown|passdown}]
[-h|-?|--help]
[--ignoremonitoring]
[--monitor {y|n}]
[-i|--stripes Stripes [-I|--stripesize StripeSize]]
[-k|--setactivationsskip {y|n}]
[-K|--ignoreactivationsskip]
{-l|--extents LogicalExtentsNumber[%{VG|FREE|ORIGIN}] |
 -L|--size LogicalVolumeSize[bBsSkKmMgGtTpPeE]}
[--poolmetadatasize MetadataVolumeSize[bBsSkKmMgG]]
[-M|--persistent {y|n}] [-j|--major major] [--minor minor]
[--metadataprofile ProfileName]
[-n|--name LogicalVolumeName]
[--noudevsync]
[-p|--permission {r|rw}]
[-r|--readahead ReadAheadSectors|auto|none]
[-t|--test]
[{--thinpool ThinPoolLogicalVolume[Path] |
  --cachepool CachePoolLogicalVolume[Path]}]
[-v|--verbose]
[--version]
[PhysicalVolumePath...]
```

Créez maintenant un répertoire dans /mnt pour monter lv0 :

```
[root@centos7 ~]# mkdir /mnt/lvm
```

Créez un système de fichiers en **ext3** sur /dev/vg0/lv0 :

```
[root@centos7 ~]# mke2fs -j /dev/vg0/lv0
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
Stride=0 blocks, Stripe width=0 blocks
90112 inodes, 360448 blocks
18022 blocks (5.00%) reserved for the super user
First data block=1
Maximum filesystem blocks=67633152
44 block groups
8192 blocks per group, 8192 fragments per group
2048 inodes per group
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729, 204801, 221185

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
```

Montez votre lv0 :

```
[root@centos7 ~]# mount -t ext3 /dev/vg0/lv0 /mnt/lvm
```

Vous allez maintenant copier le contenu de votre répertoire /home vers /mnt/lvm.

Saisissez donc la commande pour copier le contenu de /home :

```
[root@centos7 ~]# cp -a /home /mnt/lvm
```

Constatez ensuite le contenu de /mnt/lvm :

```
[root@centos7 ~]# ls -l /mnt/lvm
total 14
drwxr-xr-x. 5 root root 1024 Oct 15 18:27 home
drwx----- 2 root root 12288 Oct 20 18:24 lost+found
```

Une particularité du volume logique est la capacité de d'être agrandi ou réduit sans pertes de données. Commencez par constater la taille totale du volume :

```
[root@centos7 ~]# df -h /mnt/lvm
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/vg0-lv0 333M   50M  266M  16% /mnt/lvm
```

Dans la cas de notre exemple, la taille est de **333 Mo** avec **50 Mo** occupé.

LAB #4 - Étendre un Volume Logique à Chaud

Pour agrandir un volume logique, le paquet **lvm2** contient les commandes **lvextend** et **resize2fs** :

```
[root@centos7 ~]# lvextend -L +100M /dev/vg0/lv0
Rounding size to boundary between physical extents: 104.00 MiB
Size of logical volume vg0/lv0 changed from 352.00 MiB (44 extents) to 456.00 MiB (57 extents).
Logical volume lv0 successfully resized
```



Notez que l'agrandissement du volume est un multiple du PE.

Les options de la commande **lvextend** sont :

```
[root@centos7 ~]# lvextend --help
lvextend: Add space to a logical volume

lvextend
  [-A|--autobackup y|n]
  [--alloc AllocationPolicy]
  [--commandprofile ProfileName]
  [-d|--debug]
  [-f|--force]
  [-h|--help]
  [-i|--stripes Stripes [-I|--stripesize StripeSize]]
  {-l|--extents [+]LogicalExtentsNumber[%{VG|LV|PVS|FREE|ORIGIN}]} |
  -L|--size [+]LogicalVolumeSize[bBsSkKmMgGtTpPeE]}
  --poolmetadatasize [+]MetadataVolumeSize[bBsSkKmMgG]}
  [-m|--mirrors Mirrors]
  [--nosync]
  [--use-policies]
  [-n|--nofsck]
  [--noudevsync]
  [-r|--resizefs]
  [-t|--test]
  [--type VolumeType]
  [-v|--verbose]
  [--version]
  LogicalVolume[Path] [ PhysicalVolumePath... ]
```

Le volume ayant été agrandi, il est nécessaire maintenant d'agrandir le filesystem qui s'y trouve :

```
[root@centos7 ~]# resize2fs /dev/vg0/lv0
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/vg0/lv0 is mounted on /mnt/lvm; on-line resizing required
old_desc_blocks = 2, new_desc_blocks = 2
```

The filesystem on /dev/vg0/lv0 is now 466944 blocks long.

Constatez maintenant la modification de la taille du volume :

```
[root@centos7 ~]# df -h /mnt/lvm
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/vg0-lv0 434M   51M  362M  13% /mnt/lvm
```

Vous noterez que la taille a augmentée mais que les données sont toujours présentes.

LAB #5 - Snapshots

Un snapshot est un instantané d'un système de fichiers. Dans cet exemple, vous allez créer un snapshot de votre lv0 :

Avant de commencer, créez un fichier de 10Mo dans le volume :

```
[root@centos7 ~]# dd if=/dev/zero of=/mnt/lvm/10M bs=1048576 count=10
10+0 records in
10+0 records out
10485760 bytes (10 MB) copied, 0.044034 s, 238 MB/s
```

Créez maintenant un snapshot :

```
[root@centos7 ~]# lvcreate -s -L 65M -n testsnap /dev/vg0/lv0
Rounding up size to full physical extent 72.00 MiB
Logical volume "testsnap" created.
```

Pour avoir une confirmation de la création du snapshot, utilisez la commande **lvs** :

```
[root@centos7 ~]# lvs
LV      VG      Attr      LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
lv0     vg0     owi-aos--- 456.00m
```

testsnap vg0	swi-a-s---	72.00m	lv0	0.02
--------------	------------	--------	-----	------



Notez que le snapshot est créé dans le même VG que le LV d'origine.

Les options de la commande **lvs** sont :

```
[root@centos7 ~]# lvs --help
lvs: Display information about logical volumes
```

```
lvs
  [-a|--all]
  [--aligned]
  [--binary]
  [--commandprofile ProfileName]
  [-d|--debug]
  [-h|--help]
  [--ignorelockingfailure]
  [--ignoreskippedcluster]
  [--nameprefixes]
  [--noheadings]
  [--nosuffix]
  [-o|--options [+]Field[,Field]]
  [-O|--sort [+]-key1[,[+]-key2[,...]]]
  [-P|--partial]
  [--readonly]
  [--rows]
  [--segments]
  [-S|--select Selection]
  [--separator Separator]
  [--trustcache]
  [--unbuffered]
```

```
[--units hHbBsSkKmMgGtTpPeE]
[--unquoted]
[-v|--verbose]
[--version]
[LogicalVolume[Path] [LogicalVolume[Path]...]]
```

Créez maintenant un répertoire pour monter le snapshot :

```
[root@centos7 ~]# mkdir /mnt/testsnap
```

Montez le snapshot :

```
[root@centos7 ~]# mount /dev/vg0/testsnap /mnt/testsnap
```

Comparez le volume d'origine et le snapshot :

```
[root@centos7 ~]# ls -l /mnt/lvm
total 10296
-rw-r--r--. 1 root root 10485760 Oct 20 18:53 10M
drwxr-xr-x. 5 root root    1024 Oct 15 18:27 home
drwx-----. 2 root root    12288 Oct 20 18:24 lost+found

[root@centos7 ~]# ls -l /mnt/testsnap
total 10296
-rw-r--r--. 1 root root 10485760 Oct 20 18:53 10M
drwxr-xr-x. 5 root root    1024 Oct 15 18:27 home
drwx-----. 2 root root    12288 Oct 20 18:24 lost+found
```

Supprimez maintenant le fichier **10M** de votre volume d'origine :

```
[root@centos7 ~]# rm /mnt/lvm/10M
rm: remove regular file '/mnt/lvm/10M'? y
```

Constatez le résultat de cette suppression :

```
[root@centos7 ~]# df -Ph /mnt/lvm
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/vg0-lv0 434M   51M  362M  12% /mnt/lvm
[root@centos7 ~]# df -Ph /mnt/testsnap
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/vg0-testsnap 434M   61M  352M  15% /mnt/testsnap
[root@centos7 ~]# lvs
LV      VG      Attr      LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
lv0     vg0     owi-aos--- 456.00m
testsnap vg0     swi-aos--- 72.00m          lv0     0.04
```



Restaurez le fichier 10M à partir du snapshot.

LAB #6 - Suppression des Volumes

La suppression d'un volume logique se fait grace à la commande **lvremove** :

```
[root@centos7 ~]# umount /mnt/testsnap/
[root@centos7 ~]# lvremove /dev/vg0/testsnap
Do you really want to remove active logical volume testsnap? [y/n]: y
Logical volume "testsnap" successfully removed
[root@centos7 ~]# umount /mnt/lvm
[root@centos7 ~]# lvremove /dev/vg0/lv0
Do you really want to remove active logical volume lv0? [y/n]: y
Logical volume "lv0" successfully removed
```





Notez que cette opération nécessite à ce que le volume logique soit démonté.

Les options de la commande **lvremove** sont :

```
[root@centos7 ~]# lvremove --help
lvremove: Remove logical volume(s) from the system

lvremove
  [-A|--autobackup y|n]
  [--commandprofile ProfileName]
  [-d|--debug]
  [-f|--force]
  [-h|--help]
  [--noudevsync]
  [-t|--test]
  [-v|--verbose]
  [--version]
  LogicalVolume[Path] [LogicalVolume[Path]...]
```

Le Volume Group peut aussi être supprimé :

```
[root@centos7 ~]# vgremove vg0
Volume group "vg0" successfully removed
```

Les options de la commande **vgremove** sont :

```
[root@centos7 ~]# vgremove --help
vgremove: Remove volume group(s)

vgremove
  [--commandprofile ProfileName]
  [-d|--debug]
```

```
[-f|--force]
[-h|--help]
[--noudevsync]
[-t|--test]
[-v|--verbose]
[--version]
VolumeGroupName [VolumeGroupName...]
```

Ainsi que le volume physique :

```
[root@centos7 ~]# pvremove /dev/sda6 /dev/sda7 /dev/sda9
Labels on physical volume "/dev/sda6" successfully wiped
Labels on physical volume "/dev/sda7" successfully wiped
Labels on physical volume "/dev/sda9" successfully wiped
```

Les options de la commande **pvremove** sont :

```
[root@centos7 ~]# pvremove --help
pvremove: Remove LVM label(s) from physical volume(s)

pvremove
  [--commandprofile ProfileName]
  [-d|--debug]
  [-f[f]|--force [--force]]
  [-h|-?|--help]
  [-t|--test]
  [-v|--verbose]
  [--version]
  [-y|--yes]
  PhysicalVolume [PhysicalVolume...]
```

LAB #7 - Volumes Logiques en Miroir

Re-créez maintenant votre Volume Group :

```
[root@centos7 ~]# pvcreate /dev/sda6 /dev/sda7 /dev/sda9
Physical volume "/dev/sda6" successfully created
Physical volume "/dev/sda7" successfully created
Physical volume "/dev/sda9" successfully created
[root@centos7 ~]# vgcreate -s 8M vg0 /dev/sda6 /dev/sda7 /dev/sda9
Volume group "vg0" successfully created
```

Créez maintenant un Logical Volume en miroir grâce à l'option **-m** de la commande **lvcreate**, suivi du nombre de miroirs :

```
[root@centos7 ~]# lvcreate -m 1 -L 100M -n lv1 vg0
Rounding up size to full physical extent 104.00 MiB
Logical volume "lv1" created.
```

Constatez maintenant la présence du miroir :

```
[root@centos7 ~]# lvdisplay -m /dev/vg0/lv1
--- Logical volume ---
LV Path                /dev/vg0/lv1
LV Name                 lv1
VG Name                 vg0
LV UUID                 lJp5jc-4oI2-G2De-W739-Vdcn-uKfe-fU3sS5
LV Write Access         read/write
LV Creation host, time  centos7.fenestros.loc, 2015-10-22 10:49:03 +0200
LV Status                available
# open                  0
LV Size                 104.00 MiB
Current LE              13
Mirrored volumes        2
Segments                1
```

```
Allocation                inherit
Read ahead sectors        auto
- currently set to        8192
Block device              253:4
--- Segments ---
Logical extents 0 to 12:
  Type                    raid1
  Monitoring              monitored
  Raid Data LV 0
    Logical volume        lv1_rimage_0
    Logical extents       0 to 12
  Raid Data LV 1
    Logical volume        lv1_rimage_1
    Logical extents       0 to 12
  Raid Metadata LV 0      lv1_rmeta_0
  Raid Metadata LV 1      lv1_rmeta_1
```

Le miroir s'étend sur plusieurs volumes physiques :

```
[root@centos7 ~]# pvdisplay -m /dev/sda6 /dev/sda7 /dev/sda9
--- Physical volume ---
PV Name                    /dev/sda6
VG Name                    vg0
PV Size                    200.00 MiB / not usable 8.00 MiB
Allocatable                yes
PE Size                    8.00 MiB
Total PE                   24
Free PE                    10
Allocated PE               14
PV UUID                    7MZ7ZS-rdXi-GuS3-K06g-63f9-vXqv-puIfLA
--- Physical Segments ---
Physical extent 0 to 0:
  Logical volume           /dev/vg0/lv1_rmeta_0
  Logical extents          0 to 0
```

```
Physical extent 1 to 13:
  Logical volume  /dev/vg0/lv1_rimage_0
  Logical extents 0 to 12
Physical extent 14 to 23:
  FREE
--- Physical volume ---
PV Name           /dev/sda7
VG Name           vg0
PV Size           300.00 MiB / not usable 4.00 MiB
Allocatable       yes
PE Size           8.00 MiB
Total PE          37
Free PE           23
Allocated PE      14
PV UUID           aID1Yn-BUDT-kUd0-85qI-QhSQ-NatL-R3Rdql
--- Physical Segments ---
Physical extent 0 to 0:
  Logical volume  /dev/vg0/lv1_rmeta_1
  Logical extents 0 to 0
Physical extent 1 to 13:
  Logical volume  /dev/vg0/lv1_rimage_1
  Logical extents 0 to 12
Physical extent 14 to 36:
  FREE
--- Physical volume ---
PV Name           /dev/sda9
VG Name           vg0
PV Size           400.00 MiB / not usable 8.00 MiB
Allocatable       yes
PE Size           8.00 MiB
Total PE          49
Free PE           49
Allocated PE      0
PV UUID           8eb8Do-DdRh-zZrs-AhZ0-quW3-5s8f-XooriL
```

```
--- Physical Segments ---  
Physical extent 0 to 48:  
FREE
```

La suppression du miroir se fait en utilisant la commande **lvconvert** en indiquant quel volume physique doit être vidé de son contenu :

```
[root@centos7 ~]# lvconvert -m 0 /dev/vg0/lv1 /dev/sda7
```

De même, il est possible de créer un miroir pour un volume logique existant :

```
[root@centos7 ~]# lvconvert -m 1 /dev/vg0/lv1
```

Supprimez de nouveau votre miroir :

```
[root@centos7 ~]# lvconvert -m 0 /dev/vg0/lv1 /dev/sda7
```

Les options de la commande **lvconvert** sont :

```
[root@centos7 ~]# lvconvert --help  
lvconvert: Change logical volume layout  
  
lvconvert [-m|--mirrors Mirrors [{--mirrorlog {disk|core|mirrored}|--corelog}]]  
  [--type SegmentType]  
  [--repair [--use-policies]]  
  [--replace PhysicalVolume]  
  [-R|--regionsize MirrorLogRegionSize]  
  [--alloc AllocationPolicy]  
  [-b|--background]  
  [-f|--force]  
  [-i|--interval seconds]  
  [--stripes Stripes [-I|--stripesize StripeSize]]  
  [--commandprofile ProfileName] [-d|--debug] [-h|-?|--help]  
  [--noudevsync] [-t|--test] [-v|--verbose] [--version] [-y|--yes]  
  LogicalVolume[Path] [PhysicalVolume[Path]...]
```

```
lvconvert [--splitmirrors Images --trackchanges]
  [--splitmirrors Images --name SplitLogicalVolumeName]
  [--commandprofile ProfileName] [-d|--debug] [-h|--help]
  [--noudevsync] [-t|--test] [-v|--verbose] [--version] [-y|--yes]
  LogicalVolume[Path] [SplittablePhysicalVolume[Path]...]
```

```
lvconvert --splitsnapshot
  [--commandprofile ProfileName] [-d|--debug] [-h|--help]
  [--noudevsync] [-t|--test] [-v|--verbose] [--version] [-y|--yes]
  SnapshotLogicalVolume[Path]
```

```
lvconvert --splitcache
  [--commandprofile ProfileName] [-d|--debug] [-h|--help]
  [--noudevsync] [-t|--test] [-v|--verbose] [--version] [-y|--yes]
  CacheLogicalVolume[Path]
```

```
lvconvert --split
  [--name SplitLogicalVolumeName]
  [--commandprofile ProfileName] [-d|--debug] [-h|--help]
  [--noudevsync] [-t|--test] [-v|--verbose] [--version] [-y|--yes]
  SplittableLogicalVolume[Path]
```

```
lvconvert --uncache
  [--commandprofile ProfileName] [-d|--debug] [-h|--help]
  [--noudevsync] [-t|--test] [-v|--verbose] [--version] [-y|--yes]
  CacheLogicalVolume[Path]
```

```
lvconvert [--type snapshot|-s|--snapshot]
  [-c|--chunksize]
  [-Z|--zero {y|n}]
  [--commandprofile ProfileName] [-d|--debug] [-h|--help]
  [--noudevsync] [-t|--test] [-v|--verbose] [--version] [-y|--yes]
  OriginalLogicalVolume[Path] SnapshotLogicalVolume[Path]
```

```
lvconvert --merge
  [-b|--background]
  [-i|--interval seconds]
  [--commandprofile ProfileName] [-d|--debug] [-h|-?|--help]
  [--noudevsync] [-t|--test] [-v|--verbose] [--version] [-y|--yes]
  LogicalVolume[Path]

lvconvert [--type thin[-pool]]-T|--thin
  [--thinpool ThinPoolLogicalVolume[Path]]
  [--chunksize size]
  [--discards {ignore|nopassdown|passdown}]
  [--poolmetadataspare {y|n}]
  [{--poolmetadata ThinMetadataLogicalVolume[Path] |
  --poolmetadatasize size}]
  [-r|--readahead ReadAheadSectors|auto|none]
  [--stripes Stripes [-I|--stripesize StripeSize]]]
  [--originname NewExternalOriginVolumeName]]
  [-Z|--zero {y|n}]
  [--commandprofile ProfileName] [-d|--debug] [-h|-?|--help]
  [--noudevsync] [-t|--test] [-v|--verbose] [--version] [-y|--yes]
  [ExternalOrigin|ThinDataPool]LogicalVolume[Path] [PhysicalVolumePath...]

lvconvert [--type cache[-pool]]-H|--cache
  [--cachepool CacheDataLogicalVolume[Path]]
  [--cachemode CacheMode]
  [--chunksize size]
  [--poolmetadataspare {y|n}]]
  [{--poolmetadata CacheMetadataLogicalVolume[Path] |
  --poolmetadatasize size}]
  [--commandprofile ProfileName] [-d|--debug] [-h|-?|--help]
  [--noudevsync] [-t|--test] [-v|--verbose] [--version] [-y|--yes]
  [Cache|CacheDataPool]LogicalVolume[Path] [PhysicalVolumePath...]
```

LAB #8 - Modifier les Attributs LVM

En cas de présence d'un miroir, la commande **lvs** indique la présence du miroir dans la colonne **Attr** avec la lettre **m** :

```
[root@centos7 ~]# lvs
LV   VG   Attr      LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
lv1  vg0   -wi-a----- 104.00m
```



Consultez [cette page](#) pour comprendre les attributs.

La commande équivalente pour les Volume Groups est **vgs** :

```
[root@centos7 ~]# vgs
VG   #PV #LV #SN Attr   VSize   VFree
vg0   3   1   0 wz--n- 880.00m 776.00m
```



Consultez [cette page](#) pour comprendre les attributs.

La commande équivalente pour les Physical Volumes est **pvs** :

```
[root@centos7 ~]# pvs
PV          VG   Fmt  Attr PSize   PFree
/dev/sda6   vg0  lvm2 a--  192.00m 88.00m
/dev/sda7   vg0  lvm2 a--  296.00m 296.00m
/dev/sda9   vg0  lvm2 a--  392.00m 392.00m
```





Consultez [cette page](#) pour comprendre les attributs.

Les commandes **lvchange**, **vgchange** et **pvcchange** permettent de modifier les attributs des Logical Volumes, Volume Groups et Physical Volumes respectivement.

Par exemple, pour rendre inutilisable un Logical Volume, il convient d'enlever l'attribut **a** :

```
[root@centos7 ~]# lvchange -a n /dev/vg0/lv1
```

Pour faire l'inverse il convient de saisir la commande suivante :

```
[root@centos7 ~]# lvchange -a y /dev/vg0/lv1
```

Les options de la commande **lvchange** sont :

```
[root@centos7 ~]# lvchange --help
lvchange: Change the attributes of logical volume(s)

lvchange
  [-A|--autobackup y|n]
  [-a|--activate [a|e|l]{y|n}]
  [--activationmode {complete|degraded|partial}    [--addtag Tag]
  [--alloc AllocationPolicy]
  [-C|--contiguous y|n]
  [--cachepolicy policynome] [--cachesettings parameter=value]
  [--commandprofile ProfileName]
  [-d|--debug]
  [--deltag Tag]
  [--detachprofile]
  [--errorwhenfull {y|n}]
  [-f|--force]
  [-h|--help]
  [--discards {ignore|nopassdown|passdown}]
```

```
[--ignorelockingfailure]
[--ignoremonitoring]
[--ignoreskippedcluster]
[-k|--setactivationsskip {y|n}]
[-K|--ignoreactivationsskip]
[--monitor {y|n}]
[--poll {y|n}]
[--noudevsync]
[-M|--persistent y|n] [-j|--major major] [--minor minor]
[--metadataprofile ProfileName]
[-P|--partial]
[-p|--permission r|rw]
[--[raid]minrecoveryrate Rate]
[--[raid]maxrecoveryrate Rate]
[--[raid]syncaction {check|repair}
[--[raid]writebehind IOCount]
[--[raid]writemostly PhysicalVolume[:{t|n|y}]]
[-r|--readahead ReadAheadSectors|auto|none]
[--refresh]
[--resync]
[--sysinit]
[-t|--test]
[-v|--verbose]
[--version]
[-y|--yes]
[-Z|--zero {y|n}]
LogicalVolume[Path] [LogicalVolume[Path]...]
```

LAB #9 - Volumes Logiques en Bandes

Un volume logique en bandes est créé pour augmenter, comme dans le cas du RAID, les performances des entrées et sorties. Pour créer ce volume, la commande **lvcreate** prend deux options supplémentaires :

- **-i** - indique le nombre de volumes de bandes,
- **-I** - indique la taille en Ko de chaque bande.

Saisissez donc la commande suivante :

```
[root@centos7 ~]# lvcreate -i2 -I64 -n lv2 -L 100M vg0 /dev/sda7 /dev/sda9
Rounding up size to full physical extent 104.00 MiB
Rounding size (13 extents) up to stripe boundary size (14 extents).
Logical volume "lv2" created.
```

Constatez la présence de vos bandes sur /dev/sda7 et sur /dev/sda9 :

```
[root@centos7 ~]# lvdisplay -m /dev/vg0/lv2
--- Logical volume ---
LV Path                /dev/vg0/lv2
LV Name                 lv2
VG Name                 vg0
LV UUID                 WlQ0rn-MWr9-3Sgo-fSDR-RBSD-oEuR-0YoRxh
LV Write Access         read/write
LV Creation host, time centos7.fenestros.loc, 2015-10-22 11:13:40 +0200
LV Status               available
# open                  0
LV Size                 112.00 MiB
Current LE              14
Segments                1
Allocation              inherit
Read ahead sectors      auto
- currently set to      512
Block device            253:1
--- Segments ---
Logical extents 0 to 13:
  Type                  striped
  Stripes               2
  Stripe size           64.00 KiB
```

```
Stripe 0:
  Physical volume    /dev/sda7
  Physical extents   0 to 6
Stripe 1:
  Physical volume    /dev/sda9
  Physical extents   0 to 6
```

Utilisez maintenant la commande **lvs** pour visualiser les volumes physiques utilisés par le volume logique :

```
[root@centos7 ~]# lvs -o +devices
LV   VG   Attr      LSize   Pool Origin Data%  Meta%   Move Log Cpy%Sync Convert Devices
lv1  vg0   -wi-a----- 104.00m                               /dev/sda6(1)
lv2  vg0   -wi-a----- 112.00m                               /dev/sda7(0),/dev/sda9(0)
```

LAB #10 - Gérer les Métadonnées

Les métadonnées pour chaque Volume Group sont stockés dans un fichier texte au nom du Volume Group dans le répertoire **/etc/lvm/backup** :

```
[root@centos7 ~]# cat /etc/lvm/backup/vg0
# Generated by LVM2 version 2.02.115(2)-RHEL7 (2015-01-28): Thu Oct 22 11:13:40 2015

contents = "Text Format Volume Group"
version = 1

description = "Created *after* executing 'lvcreate -i2 -I64 -n lv2 -L 100M vg0 /dev/sda7 /dev/sda9'"

creation_host = "centos7.fenestros.loc"   # Linux centos7.fenestros.loc 3.10.0-229.14.1.el7.x86_64 #1 SMP Tue Sep
15 15:05:51 UTC 2015 x86_64
creation_time = 1445505220   # Thu Oct 22 11:13:40 2015

vg0 {
    id = "g9ITer-9KG7-N0mK-Gny3-pU4U-7Pl1-EPIKzj"
    seqno = 10
```

```
format = "lvm2"          # informational
status = ["RESIZEABLE", "READ", "WRITE"]
flags = []
extent_size = 16384      # 8 Megabytes
max_lv = 0
max_pv = 0
metadata_copies = 0

physical_volumes {

    pv0 {
        id = "7MZ7ZS-rdXi-GuS3-K06g-63f9-vXqv-puIf1A"
        device = "/dev/sda6"    # Hint only

        status = ["ALLOCATABLE"]
        flags = []
        dev_size = 409600    # 200 Megabytes
        pe_start = 2048
        pe_count = 24    # 192 Megabytes
    }

    pv1 {
        id = "aID1Yn-BUDT-kUd0-85qI-QhSQ-NatL-R3Rdql"
        device = "/dev/sda7"    # Hint only

        status = ["ALLOCATABLE"]
        flags = []
        dev_size = 614400    # 300 Megabytes
        pe_start = 2048
        pe_count = 37    # 296 Megabytes
    }

    pv2 {
        id = "8eb8Do-DdRh-zZrs-AhZ0-quW3-5s8f-XooriL"
```

```
        device = "/dev/sda9"    # Hint only

        status = ["ALLOCATABLE"]
        flags = []
        dev_size = 819200      # 400 Megabytes
        pe_start = 2048
        pe_count = 49         # 392 Megabytes
    }
}

logical_volumes {

    lv1 {
        id = "lJp5jc-4oI2-G2De-W739-Vdcn-uKfe-fU3sS5"
        status = ["READ", "WRITE", "VISIBLE"]
        flags = []
        creation_host = "centos7.fenestros.loc"
        creation_time = 1445503743    # 2015-10-22 10:49:03 +0200
        segment_count = 1

        segment1 {
            start_extent = 0
            extent_count = 13        # 104 Megabytes

            type = "striped"
            stripe_count = 1        # linear

            stripes = [
                "pv0", 1
            ]
        }
    }

    lv2 {
```

```
id = "WlQ0rn-MWr9-3Sgo-fSDR-RBSD-oEuR-0YoRxh"
status = ["READ", "WRITE", "VISIBLE"]
flags = []
creation_host = "centos7.fenestros.loc"
creation_time = 1445505220    # 2015-10-22 11:13:40 +0200
segment_count = 1

segment1 {
    start_extent = 0
    extent_count = 14    # 112 Megabytes

    type = "striped"
    stripe_count = 2
    stripe_size = 128    # 64 Kilobytes

    stripes = [
        "pv1", 0,
        "pv2", 0
    ]
}
}
```

Des archives sont créées lors de chaque modification d'un groupe de volumes et elles sont placées dans le répertoire **/etc/lvm/archives** :

```
[root@centos7 ~]# ls /etc/lvm/archive/
vg0_00000-1537004428.vg  vg0_00003-1961496268.vg  vg0_00006-1090426669.vg  vg0_00009-1613274821.vg
vg0_00012-346101557.vg
vg0_00001-1563486661.vg  vg0_00004-1701914435.vg  vg0_00007-986450784.vg  vg0_00010-1614712452.vg
vg0_00002-1169707904.vg  vg0_00005-1940504069.vg  vg0_00008-2002482726.vg  vg0_00011-522317302.vg
```

La commande **vgcfgbackup** est utilisée pour sauvegarder les métadonnées manuellement dans le fichier **/etc/lvm/backup/nom_du_volume_group** :

La commande **vgcfgrestore** permet de restaurer une sauvegarde. Sans l'option **-f** pour spécifier la sauvegarde à restaurer, cette commande renvoie la liste des sauvegardes disponibles :

```
[root@centos7 ~]# vgcfgbackup vg0
Volume group "vg0" successfully backed up.
```

Il est aussi possible de modifier l'emplacement de la sauvegarde avec l'option **-f** de la commande :

```
[root@centos7 ~]# vgcfgbackup -f /tmp/vg0_backup vg0
Volume group "vg0" successfully backed up.
```

```
[root@centos7 ~]# vgcfgrestore --list vg0
File:      /etc/lvm/archive/vg0_00000-1537004428.vg
Couldn't find device with uuid 9o7S0M-NU2B-dKzi-crvR-rJej-kw20-QtY0t5.
Couldn't find device with uuid HpElCF-L3x7-pjnl-IgmA-CeNc-ZBCu-5sk3AZ.
Couldn't find device with uuid vu7nY2-ac3k-Hp19-gFVQ-ny0I-sZoY-Sykbw.
VG name:    vg0
Description: Created *before* executing 'vgcreate -s 8M vg0 /dev/sda6 /dev/sda7 /dev/sda9'
Backup Time: Tue Oct 20 18:20:31 2015

File:      /etc/lvm/archive/vg0_00001-1563486661.vg
VG name:    vg0
Description: Created *before* executing 'lvcreate -L 350 -n lv0 vg0'
Backup Time: Tue Oct 20 18:22:54 2015

File:      /etc/lvm/archive/vg0_00002-1169707904.vg
VG name:    vg0
Description: Created *before* executing 'lvextend -L +100M /dev/vg0/lv0'
Backup Time: Tue Oct 20 18:48:17 2015

File:      /etc/lvm/archive/vg0_00003-1961496268.vg
VG name:    vg0
Description: Created *before* executing 'lvcreate -s -L 65M -n testsnap /dev/vg0/lv0'
Backup Time: Tue Oct 20 18:54:41 2015
```

File: /etc/lvm/archive/vg0_00004-1701914435.vg
VG name: vg0
Description: Created *before* executing 'lvremove /dev/vg0/testsnap'
Backup Time: Thu Oct 22 10:41:09 2015

File: /etc/lvm/archive/vg0_00005-1940504069.vg
VG name: vg0
Description: Created *before* executing 'lvremove /dev/vg0/lv0'
Backup Time: Thu Oct 22 10:44:50 2015

File: /etc/lvm/archive/vg0_00006-1090426669.vg
VG name: vg0
Description: Created *before* executing 'vgremove vg0'
Backup Time: Thu Oct 22 10:46:36 2015

File: /etc/lvm/archive/vg0_00007-986450784.vg
Couldn't find device with uuid 7MZ7ZS-rdXi-GuS3-K06g-63f9-vXqv-puIfLA.
Couldn't find device with uuid aID1Yn-BUDT-kUd0-85qI-QhSQ-NatL-R3Rdql.
Couldn't find device with uuid 8eb8Do-DdRh-zZrs-AhZ0-quW3-5s8f-XooriL.
VG name: vg0
Description: Created *before* executing 'vgcreate -s 8M vg0 /dev/sda6 /dev/sda7 /dev/sda9'
Backup Time: Thu Oct 22 10:48:40 2015

File: /etc/lvm/archive/vg0_00008-2002482726.vg
VG name: vg0
Description: Created *before* executing 'lvcreate -m 1 -L 100M -n lv1 vg0'
Backup Time: Thu Oct 22 10:49:03 2015

File: /etc/lvm/archive/vg0_00009-1613274821.vg
VG name: vg0
Description: Created *before* executing 'lvconvert -m 0 /dev/vg0/lv1 /dev/sda7'
Backup Time: Thu Oct 22 10:52:31 2015

File: /etc/lvm/archive/vg0_00010-1614712452.vg

```
VG name:      vg0
Description:   Created *before* executing 'lvconvert -m 1 /dev/vg0/lv1'
Backup Time:  Thu Oct 22 10:53:01 2015

File:         /etc/lvm/archive/vg0_00011-522317302.vg
VG name:      vg0
Description:   Created *before* executing 'lvconvert -m 0 /dev/vg0/lv1 /dev/sda7'
Backup Time:  Thu Oct 22 10:55:54 2015

File:         /etc/lvm/archive/vg0_00012-346101557.vg
VG name:      vg0
Description:   Created *before* executing 'lvcreate -i2 -I64 -n lv2 -L 100M vg0 /dev/sda7 /dev/sda9'
Backup Time:  Thu Oct 22 11:13:40 2015

File:         /etc/lvm/backup/vg0
VG name:      vg0
Description:   Created *after* executing 'vgcfgbackup vg0'
Backup Time:  Thu Oct 22 11:16:44 2015
```

<html>

Copyright © 2004-2017 Hugh Norris.

Ce(tte) oeuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Pas de Modification 3.0 France.

</html>
