

用語

- ・ PV (Physical Volume)

物理ボリュームのこと。 /dev/hda7 や /dev/hdc と示すディスクそのものを意味する。

- ・ VG (Volume Group)

上記の PV を束ねて作成されるボリュームのこと。この中から実際に使う「論理的な」パーティションを切り出すことになる。

- ・ LV (Logical Volume)

LV とは、ここまで「論理的な」ボリュームと称してきたものであり、VG から切り出して実際に利用することになるパーティションを示す。

LVM の作成

Linux において LVM を利用する手順は下のとおりです。

- 1.LVM パーティションの作成
- 2.PV (Physical Volume) の作成
- 3.VG (Volume Group) の作成
- 4.LV (Logical Volume) の作成

パーティション

LVM 用のパーティションは

```
fdisk /dev/hda
```

で fdisk を起動して

```
t  
で  
8e Linux LVM
```

を選ぶ。

PV (Physical Volume) の作成

```
pvccreate /dev/hda6
```

pv 情報の表示

```
pvdisplay /dev/hda6
```

pv の一覧

```
pvs
```

VG (Volume Group) の作成

```
vgcreate myvg /dev/hda6
```

vg の情報、一覧

```
vgdisplay  
vgs
```

VG の拡張

```
vgextend myvg /dev/sda7
```

論理ボリューム (LogicalVolume) の割り当て

```
lvcreate -L 100m -n lvtest myvg
```

オプション `-L` で論理ボリュームの大きさを指定。
オプション `-n` で論理ボリュームの名前を指定。

未使用領域を全て割り当てて作成する場合

```
lvcreate -l 100%FREE -n lvtest myvg
```

lv の情報、一覧

```
lvdisplay  
lvs
```

論理ボリューム (LogicalVolume) の拡張

`/dev/VolGroup00/LogVol00` を 20G にする

```
lvextend -L 20G /dev/VolGroup00/LogVol00
```

`/dev/VolGroup00/LogVol00` に 20G を足す

```
lvextend -L +20G /dev/VolGroup00/LogVol00
```

`/dev/VolGroup00/LogVol00` に未使用領域を全て割り当てる

```
lvextend -l +100%FREE /dev/VolGroup00/LogVol00
```

論理ボリュームのフォーマット

普通の Linux パーティションと同様にフォーマットしたり mount したり出来ます

```
mke2fs -j /dev/myvg/lvtest
```

ボリュームグループの情報のバックアップとリストア

バックアップ

```
pvdisplay
```

または

```
vgdisplay -v
```

PV UUID を確認する。

```
vgcfgbackup VolGroup00 -f /nfs/VolGroup00.txt
```

ボリュームグループ情報をバックアップ

リストア

```
lvm pvcreate -ff --restorefile /nfs/VolGroup00.txt --uuid バックアップ時に確認した PV UUID /dev/hda6  
lvm vgcfgrestore -f /nfs/VolGroup00.txt VolGroup00
```

アクティブにする

```
lvm lvchange -a y VolGroup00
```

PV から物理ディスクを取り外す

1. 取り外す PV の内容を別の PV に移す
 1. pvmove で取り外すデバイスの内容を他のデバイスへ移動する
 1. pvmove /dev/sdb1
 2. pvmove は時間がかかる。Ctrl + C で中断してもバックグラウンドで動作し続ける
 3. pvmove --abort 止めることができるが、pvmove はもう一度最初からやり直しになる。
 2. PV を VG から外す
 1. vgreduce で VG から PV を外す
 1. vgreduce data /dev/sdb1

スナップショット機能

<http://www.usupi.org/sysad/166.html>

もとのファイルシステムは、ボリュームグループが Volume00 という名前で、論理ボリュームが LogVol00 という名前だとしましょう。

そして、以下のように /mnt/test に mount されているとします。

```
# df /mnt/test  
Filesystem          1K-blocks      Used Available Use% Mounted on  
/dev/mapper/Volume00-LogVol00  
                    142799    107433     29471   79% /mnt/test
```

スナップショットを作成するには、いつものように、lvcreate コマンドを使用します。簡単な書式は、以下の通りです。

```
# lvcreate -s -L サイズ -n 論理ボリューム名 もとのデバイス
```

スナップショットの論理ボリューム名を -n オプションで指定します。

「もとのデバイス」には、もとのファイルシステムのデバイスファイルを指定します。

-L オプションで指定するサイズは、もとのデータを保持するための領域です。もとのファイルシステムと同じサイズである必要はありません。

というわけで、ここでは LogVol00_snap という名前のスナップショットを作成してみました。サイズは、もとが 144MB ですので、32MB くらいにしてみました。（使用頻度などによりますので、えいやーで決めました。）

```
# lvcreate -s -L 32M -n LogVol00_snap /dev/Volume00/LogVol00
```

lvscan や lvsdisplay で確認しますと、以下のようにスナップショットであることがわかります。

```
# lvscan
ACTIVE Original '/dev/Volume00/LogVol00' [144.00 MB] inherit
ACTIVE Snapshot '/dev/Volume00/LogVol00_snap' [32.00 MB] inherit
# lvsdisplay /dev/Volume00/LogVol00_snap
--- Logical volume ---
LV Name                /dev/Volume00/LogVol00_snap
... 中略 ...
LV Size                144.00 MB
Current LE             36
COW-table size        32.00 MB
COW-table LE          8
Allocated to snapshot 0.07%
Snapshot chunk size   8.00 KB
... 後略 ...
```

では、mount してみましょう。

もちろん、通常の論理ボリュームとして使用できます。

ここでは、/mnt/snap に mount してみました。

```
# mount -o ro /dev/mapper/Volume00-LogVol00_snap /mnt/snap
# df /mnt/test /mnt/snap
Filesystem      1K-blocks      Used Available Use% Mounted on
/dev/mapper/Volume00-LogVol00
142799 107433    29471  79% /mnt/test
/dev/mapper/Volume00-LogVol00_snap
142799 107433    29471  79% /mnt/snap
```

当たり前ですが、もとのファイルシステムとまったく同じですね。

オンラインリサイズ

<http://blog.usoinfo.info/article/193585529.html>

<http://pantora.net/pages/lvm/3/>

--	ext2	ext3/ext4	ReiserFS	XFS	JFS
拡大	オフライン (resize2fs)	オンライン / オフライン (resize2fs)	オンライン / オフライン (resize_reiserfs)	オンライン (xfs_growfs)	オンライン (mount -o remount,resize)
縮小	オフライン	オフライン	オフライン	なし	なし

lvresize で LogicalVolume のサイズを変更

```
# lvresize -L +58G /dev/VolGroup00/LogVol00
```

resize2fs でオンラインのままリサイズ実行 (resize2fs は結構時間が掛かる)

```
# resize2fs -p /dev/mapper/VolGroup00-LogVol00
```

CentOS7(XFS) の場合

```
# xfs_growfs /dev/centos/root
```

resize2fs の -p オプションは作業完了分のパーセントを表すバーを表示するオプション

Volume Group に不良な Logical Volume ができてしまった場合

<http://koozme.blogspot.jp/2012/10/contos.html>

<http://hotaru.4030.jp/category/68853-1.html>

LVM から所定の手続きを踏まずにディスク交換やネットワークの不具合で iSCSI の LVM がおかしくなった場合。

```
PV unknown device VG VolGropu00 ...
```

などが出る場合は、

```
# vgreduce --removemissing VolGroup00  
# vgscan
```

で不正な Physical Volume を外してくれる。ただし、iSCSI とかを使っている場合は扱いに注意。

VolumeGroup の残り容量を確認する

```
vgs
```

で使用領域、未使用領域が確認できる。