LVM Logical volumes

CONCEPTS

Disks become PVs, PVs pool into a VG, and you carve LVs out of the VG. An LV is a block device: put a filesystem on it.

Disk / partition /dev/sdb1
↓ pvcreate
Physical volume PV
↓ vgcreate
Volume group VG, pool of extents
↓ lvcreate
Logical volume /dev/<vg>/<lv>

Space is handed out in extents (4 MiB by default). Sizes are rounded to whole extents.

Name an LV as <vg>/<lv> or /dev/<vg>/<lv>. Use those paths, not /dev/mapper, in scripts.

INSPECT

pvs
List physical volumes

vgs
List volume groups with free space

lvs
List logical volumes

lvs -a -o +devices
Show LVs and the PVs they use

pvdisplay *<pv>*
Detailed PV info

vgdisplay *<vg>*
Detailed VG info (extent size, free PE)

lvdisplay *<vg>*/*<lv>*
Detailed LV info and status

lsblk
Show the block device tree

PHYSICAL VOLUMES

pvcreate *<dev>*...
Initialise a disk or partition

pvremove *<dev>*
Remove the LVM label from a PV

pvresize *<dev>*
Grow a PV after the disk got bigger

pvmove *<old-pv>* [*<new-pv>*]
Move all extents off a PV

VOLUME GROUPS

vgcreate *<vg>* *<pv>*...
Create a VG from PVs

vgextend *<vg>* *<pv>*...
Add PVs to a VG

vgreduce *<vg>* *<pv>*
Remove an empty PV from a VG

vgrename *<old>* *<new>*
Rename a VG

vgchange -ay *<vg>*
Activate all LVs in a VG

vgchange -an *<vg>*
Deactivate all LVs in a VG

LOGICAL VOLUMES

lvcreate -L *<size>* **-n** *<lv>* *<vg>*
Create an LV of a fixed size

lvcreate -l 100%FREE -n *<lv>* *<vg>*
Create an LV from all free space

lvcreate -l 50%VG -n *<lv>* *<vg>*
Use a share of the whole VG

mkfs.ext4 /dev/<vg>/<lv>
Put a filesystem on the LV

mount /dev/<vg>/<lv> *<dir>*
Mount the LV

lvrename *<vg>* *<old>* *<new>*
Rename an LV

lvremove *<vg>*/*<lv>*
Delete an LV (data is lost)

lvchange -ay *<vg>*/*<lv>*
Activate an LV

lvchange -an *<vg>*/*<lv>*
Deactivate an LV

RESIZE

lvextend -L +<size> <vg>/<lv>
Grow an LV by an amount

lvextend -L *<size>* *<vg>*/*<lv>*
Grow an LV to an absolute size

lvextend -l +100%FREE *<vg>*/*<lv>*
Grow an LV into all free space

lvextend -r -L +<size> <vg>/<lv>
Grow the LV and its filesystem

RESIZE (cont.)

lvresize -r -L -<size> <vg>/<lv>
Shrink LV and filesystem together

resize2fs /dev/<vg>/<lv>
Grow an ext4 fs to fill its LV

xfs_growfs *<mountpoint>*
Grow a mounted XFS fs

Shrinking order: unmount, fsck, shrink the filesystem, then reduce the LV, never the other way round. Prefer lvresize -r and take a backup first.

Growing works online for ext4 and XFS. Extend the LV first, then the filesystem (or let -r do it).

SNAPSHOTS

lvcreate -s -L *<size>* **-n** *<snap>* *<vg>*/*<lv>*
Snapshot an LV

lvconvert --merge *<vg>*/*<snap>*
Roll the origin back to a snapshot

lvremove *<vg>*/*<snap>*
Drop a snapshot

lvs -o +snap_percent
Show how full a snapshot is

BACKUP AND REPAIR

vgcfgbackup *<vg>*
Back up VG metadata

vgcfgrestore -f *<file>* *<vg>*
Restore VG metadata

RECIPES

New data disk to mounted volume

```
pvcreate /dev/sdb
vgcreate data /dev/sdb
lvcreate -l 100%FREE -n files
data
mkfs.ext4 /dev/data/files
mount /dev/data/files /srv/files
```

Add a disk and grow a mounted volume

```
pvcreate /dev/sdc
vgextend data /dev/sdc
lvextend -r -l +100%FREE data/
files
df -h /srv/files
```