

LVM

Logical Volume Management

LVM allows an abstraction from
Disk drives and partitions
To storage units by use

LVM formed of
Physical volumes
Volume groups
Logical volumes

Physical volumes use drives or partitions

Find using pvscan:

```
pvscan
```

```
PV /dev/mapper/luks-1c14505a-fc80-41fe-b389-c927195cc287  
VG fedora lvm2 [237.98 GiB / 4.00 MiB free]  
Total: 1 [237.98 GiB] / in use: 1 [237.98 GiB] / in no VG: 0 [0 ]
```

Use multiple PVs to make a large virtual drive

X10 1TB drives

1 partition each

Each partition is a PV

Join 4 PVs into a Volume Group called vg_root

Join 4 into another called vg_home

Leave 2 spares for later.

Volume Groups

Used to aggregate PVs into a named collection

Find using vgscan

vgscan

Reading all physical volumes. This may take a while...

Found volume group "fedora" using metadata type lvm2

Volume groups

Use names that make sense for large installations

Logical volumes

Carved out of volume groups

Lvs can to find

lvscan

ACTIVE	'/dev/fedora/root' [50.48 GiB] inherit
ACTIVE	'/dev/fedora/swap' [7.50 GiB] inherit
ACTIVE	'/dev/fedora/home' [180.00 GiB] inherit

Summary:

Physical Volumes (PV) from hard drives are grouped into Volume Groups (VG) that are broken up into Logical Volumes (LV) that are mounted and hold a filesystem.

easy, huh?

Tons 'O' Fun

lvm is a “entry command” with a zillion capabilities
[pv|vg|lv]display for lots of details

Plus every command can be run individually
outside of the lvm environment

Let's add a new disk to an existing system

New 1TB disk is sdb

Use fdisk or gparted to create sdb1 as type LVM

pvcreate /dev/sdb1

Using "fedora" VG:

vgextend fedora /dev/sdb1

So now the fedora VG is 1TB larger and a
vgdisplay will show it.

Add more space to the home logical volume
lvextend -t -r -L +500G /dev/fedora/home /dev/sdb1

Note the -t for “test” BIG BUTT SAVER

The -r means to “resize” the underlying filesystem

-L +<size> will add <size>

-L -<size> will shrink by <size>

-L <size> will make the final result of size <size>

<size> units are default M but can be

bBsSkKmMgGtTpPeE

An Exabyte size filesystem doing a mandatory fsck would stink.

Note: XFS filesystem can't shrink. EXT2,3,4 can

(why are you still using ext2???)

So once the liveness check runs smoothly with the -t
Take it out and run it for real

Are you feeling lunky, punk?

Well?

Are you?

Similar process to shrink an existing LV
to make space for growing another LV.

lvreduce -L -<size> /dev/fedora/home

BIG NOTE: The filesystem must be unmountable
to do this. So resizing / is not going to work
without a live CD

The ALL IMPORTANT one

Backups using LVM snapshots

WHY?!?!?

By taking a snapshot, the files in the snapshot are frozen in time. Very useful for databases and log files and anything else on an active system.

Background action: a snapshot is a special logical volume that doesn't actually have anything in it initially. It's really a bunch of logical links to the original physical extents (PEs) on the original logical volume.

As files change on the live file system, the original PEs are copied to the snapshot LV while the original PEs get new data.

Need space to store the snapshot LV. It MUST be large enough to accommodate anticipated growth of the original filesystem during the backup process. If it fills up, it is dumped.

XXX

:-(

Backing up my database with minimal downtime

This would be scripted to be as fast as possible

1) Stop the database

1) Flush all pending commits

2) Stop listener

3) Halt the database

2) Create the snapshot of the database LV

1) Scripted to make the quick LV

3) Start the database

1) Normal startup

4) Run the backup of the snapshot

5) Remove the snapshot LV

6) Celebrate a successful evening with a tasty adult beverage – This step is mandatory.

You are on your own for how to do the database part. This is a talk about LVM not DBA!

Assume the database is stopped and the partition is on VG db and is LV big_data. A lvdisplay says is 10GB so you will need a snapshot space of that size.

How much space you will need for the changes in the snapshot is an estimate based on DBA help. The VG db will be filling up at TWICE the rate the data is changing – once for the data and once more for the original data!

Do the deed (on a need for 10G space for snapshot changes):

lvcreate -L 10G -s -n dbsnapshot /dev/db/big_data

-L is the size

-s is the snapshot type of LV being set

-n names the snapshot

So now you have a snapshot logical volume called dbsnapshot on the volume group db. It's filling up so work faster!

If you mount the snapshot with
mount /dev/db/dbsnapshot <mountpoint>
you can do an ls, du, etc. on the filesystem.

Or you can do a
dd if=/dev/db/dbsnapshot of=<backup space>

If you mounted the dbsnapshot to look at it,
unmount it now.

It must be removed to prevent the snapshot LV
from running out of space

lvremove /dev/db/dbsnapshot

To do the restore:

Stop the database

dd if=<backup space> of=/dev/db/big_data

Start the database

Note that this will be MUCH slower and the database will be offline for all of it.