Logical Volume Manager (LVM)

This is a quick and dirty cheat sheet on LVM using Linux, I have highlighted many of the common attributes for each command however this is not an extensive list, make sure you look up the command.

With the pvs, vgs and lvs commands, the number of verboses added the more verbose information for example pvs -vvvvv

|  |  |
| --- | --- |
| **Directory and Files** | |
| **Directories and Files** | ## Directories /etc/lvm                    - default lvm directory location  /etc/lvm/backup         - where the automatic backups go  /etc/lvm/cache          - persistent filter cache  /etc/lvm/archive        - where automatic archives go after a volume group change  /var/lock/lvm             - lock files to prevent metadata corruption   # Files  /etc/lvm/lvm.conf       - main lvm configuration file  $HOME/.lvm               - lvm history |
| **Tools** | |
| **diagnostic** | lvmdump lvmdump -d <dir> dmsetup [info|ls|status]  Note: by default the lvmdump command creates a tar ball |
| **Physical Volumes** | |
| **display** | pvdisplay -v  pvs -v pvs -a pvs --segments (see the disk segments used)   **pvs attributes are:**  1. (a)llocatable  2. e(x)ported |
| **scanning** | pvscan -v  Note: scans for disks for non-LVM and LVM disks |
| **adding** | pvcreate /dev/sdb1  ## Create physical volume with specific UUID, used to recover volume groups (see miscellaneous section)  pvcreate --uuid <UUID> /dev/sdb1   **Common Attributes that you may want to use:**  -M2 create a LVM2 physical volume |
| **removing** | pvremove /dev/sdb1 |
| **checking** | pvck -v /dev/sdb1   Note: check the consistency of the LVM metadata |
| **change physical attributes** | ## do not allow allocation of extents on this drive  pvchange -x n /dev/sdb1  **Common Attributes that you may want to use:**  --addtag add a tag -x allowed to allocate extents -u change the uuid |
| **moving** | pvmove -v /dev/sdb2 /dev/sdb3   Note: moves any used extents from this volume to another volume, in readiness to remove that volume. However you cannot use this on mirrored volumes, you must convert back to non-mirror using "lvconvert -m 0" |
| **Volume Groups** | |
| **display** | vgdisplay -v  vgs -v vgs -a -o +devices   **vgs flags:** #PV - number of physical devices #LV - number of configured volumes          **vgs attributes are:**  1. permissions (r)|(w) 2. resi(z)eable 3. e(x)ported 4. (p)artial 5. allocation policy - (c)ontiguous, c(l)ing, (n)ormal, (a)nywhere, (i)nherited 6. (c)luster |
| **scanning** | vgscan -v |
| **creating** | vgcreate VolData00 /dev/sdb1 /dev/sdb2 /dev/sdb3 vgcreate VolData00 /dev/sdb[123]  ## Use 32MB extent size  vgcreate VolData00 -s 32 /dev/sdb1   **Common Attributes that you may want to use:**  -l  maximum logical volumes -p maximum physical volumes -s physical extent size (default is 4MB) -A autobackup |
| **extending** | vgextend VolData00 /dev/sdb3 |
| **reducing** | vgreduce VolData00 /dev/sdb3  vgreduce --removemissing --force VolData00 |
| **removing** | vgremove VolData00  **Common Attributes that you may want to use:**  -f force the removal of any logical volumes |
| **checking** | vgck VolData00   Note: check the consistency of the LVM metadata |
| **change volume attributes** | vgchange -a n VolData00  **Common Attributes that you may want to use:**   -a control availability of volumes within the group -l  maximum logical volumes -p maximum physical volumes -s physical extent size (default is 4MB) -x resizable yes or no (see VG status in vxdisplay) |
| **renaming** | vgrename VolData00 Data\_Vol\_01  note: the volume group must not have any active logical volumes |
| **converting metadata type** | vgconvert -M2 VolData00  Note: vgconvert allows you to convert from one type of metadata format to another for example from LVM1 to LVM2 |
| **merging** | vgmerge New\_Vol\_Group Old\_Vol\_Group  Note: the old volumes group will be merged into the new volume group |
| **spliting** | vgsplit Old\_Vol\_Group New\_Vol\_Group [physical volumes] [-n logical volume name] |
| **importing** | vgimport VolData00  **Common Attributes that you may want to use:**   -a import all exported volume groups |
| **exporting** | ## to see if a volume has already been export use "vgs" and look at the third attribute should be a x  vgexport VolData00  **Common Attributes that you may want to use:**   -a export all inactive volume groups |
| **backing up** | ## Backup to default location (/etc/lvm/backup)  vgcfgbackup VolData00  # Backup to specific location  vgcfgbackup -f /var/backup/VolData00\_bkup VolData00  Note: the backup is written in plain text and are by default located in /etc/lvm/backup |
| **restoring** | vgcfgrestore -f /var/backup/VolData00\_bkup VolData00  **Common Attributes that you may want to use:**   -l list backups of file -f backup file -M metadataype 1 or 2 |
| **cloning** | vgimportclone /dev/sdb1  Note: used to import and rename duplicated volume group |
| **special files** | vgmknodes VolData00  Note: recreates volume group directory and logical volume special files in /dev |
| **Logical Volumes** | |
| **display** | lvdisplay -v  lvdisplay --maps      display mirror volumes  lvs -v lvs -a -o +devices   ## lvs commands for mirror volumes  lvs -a -o +devices lvs -a -o +seg\_pe\_ranges --segments   **lvs attributes are:**  1. volume type: (m)irrored, (M)irrored without initail sync, (o)rigin, (p)vmove, (s)napshot, invalid (S)napshot, (v)irtual, mirror (i)mage                       mirror (I)mage out-of-sync, under (c)onversion 2. permissions: (w)rite, (r)ead-only 3. allocation policy - (c)ontiguous, c(l)ing, (n)ormal, (a)nywhere, (i)nherited 4. fixed (m)inor  5. state: (a)ctive, (s)uspended, (I)nvalid snapshot, invalid (S)uspended snapshot, mapped (d)evice present with-out tables,              mapped device present with (i)nactive table  6. device (o)pen (mounted in other words) |
| **scanning** | lvscan -v  lvmdiskscan |
| **creating** | ## plain old volume  lvcreate -L 10M VolData00  ## plain old volume but with a specific name web01 lvcreate -L 10M -n web01 VolData00   ## plain old volume but on a specific disk  lvcreate -L 10M VolData00 /dev/sdb1  ## a striped volume called lvol1 (note the captial i for the stripe size)  lvcreate -i 3 -I 32 -L 24M -n lvol1 vg01  ## Mirrored volume lvcreate -L 10M -m1 -n data01 vg01  ## Mirrored volume without a mirror log file lvcreate -L 10M -m1 --mirrorlog core -n data01 vg01  **Common Attributes that you may want to use:**   -L size of the volume [kKmMgGtT] -l number of extents -C contiguous [y|n] -i stripes -I stripe size -m mirrors --mirrorlog  -n volume name |
| **extending** | lvextend -L 20M /dev/VolData00/vol01  **Common Attributes that you may want to use:**  -L size of the volume [kKmMgGtT] -l number of extents -C contiguous [y|n] -i stripes -I stripe size  Note: you can extend a ext2/ext3 filesystem using the "resize2fs" or "fsadm" command   fsadm resize /dev/VolData01/data01 resize2fs -p /dev/mapper/VolData01-data01 [size]   The -p option displays bars of progress while extendingthe filesystem |
| **reducing/resizing** | lvreduce -L 5M /dev/VolData00/vol01 lvresize -L 5M /dev/VolData00/vol01  Note: rounding will occur when extending and reducing volumes to the next extent (4MB by default), you can use resize2fs or fsadm to shrink the filesystem  fsadm resize /dev/VolData01/data01 [size]  resize2fs -p /dev/mapper/VolData01-data01 [size] |
| **removing** | lvremove /dev/VolData00/vol01 |
| **adding a mirror to a non-mirrored volume** | lvconvert -m1 --mirrorlog core /dev/VolData00/vol01 /dev/sdb2 |
| **change volume attributes** | lvchange -a n /dev/VolData00/vol01  **Common Attributes that you may want to use:**  -a availability -C contiguous [y|n] |
| **renaming** | lvrename /dev/VolData00/vol\_old /dev/VolData00/vol\_new |
| **snapshotting** | lvcreate --size 100M --snapshot -name snap /dev/vg01/data01 |
| **Miscellaneous** | |
| **Simulating a disk failure** | dd if=/dev/zero of=/dev/sdb2 count=10 |
| **reparing a failed mirror no LVM corruption** | ## check volume, persume /dev/sdb2 has failed  lvs -a -o +devices  # remove the failed disk from the volume (if not already done so) , this will convert volume into a non-mirrored volume  vgreduce --removemissing --force VolData00  ## replace the disk physically, remember to partion it with type 8e  fdisk /dev/sdb ........   ## add new disk to LVM pvcreate /dev/sdb2  ## add the disk back into volume group vgextend VolData00 /dev/sdb2  ## mirror up the volume lvconvert -m1 --mirrorlog core /dev/VolData00/vol02 /dev/sdb2 |
| **corrupt LVM metadata without replacing drive** | # attempt to bring the volume group online vgchange -a y VolData00  # Restore the LVM configation vgcfgrestore VolData00  # attempt to bring the volume grou online vgchange -a y VolData00  # file system check e2fsck /dev/VolData00/data01 |
| **corrupt LVM metadata but replacing the faulty disk** | # attempt to bring the volume group online but you get UUID conflict errors make note of the UUID number  vgchange -a y VolData00 vgchange -a n VolData00   ## sometimes it my only be a logical volume problem lvchange -a y /dev/VolData00/web02 lvchange -a n /dev/Voldata00/web02   ## replace the disk physically, remember to partion it with type 8e  fdisk /dev/sdb ........  # after replacing the faulty drive the disk must have the previuos UUID number or you can get it from /etc/lvm directory  pvcreate --uuid <**previous UUID number taken from above command**> /dev/sdb2  # Restore the LVM configation vgcfgrestore VolData00  # attempt to bring the volume group online or logical volume  vgchange -a y VolData00 lvchange -a y /dev/VolData00/web02   # file system check e2fsck /dev/VolData00/data01  Note: if you have backed the volume group configuration you can obtain the UUID number in the backup file by default located in /etc/lvm/backup or running "pvs -v" |

For other LVM's and Array utilities see my [LVM central](http://www.datadisk.co.uk/main/lvm.htm) page

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*