Extension of the role STORE (LVM not redundant)

Advanced Topic

Warning - this is an advanced topic. Always back up your data first.

Expanding the store with LVM

Most of our Linux / CentOS installations work with LVM. This can be extended - provided the system recognizes a plate enlargement or new plate - as follows. Here is the procedure for a disk expansion.

Step 1 - Determining the current config

Display of discs

```
# If the machine has not yet been rebooted, the SCSI bus can be scanned to detect new disks with the following command
# Ggf - falls mehrere SCSI Adapter vorhanden sind host0 mit host1 oder host2 ... nochmals probieren bis die Platte gefunden wird
echo "- - -" > /sys/class/scsi_host/host0/scan
# Show free space
df -h
# Show partitions
fdisk -l
# Show hard drives
ls /dev/sd*
# View physical volumes managed by LVM
lvm pvs
# Display logical volumes managed by LVM
lvm lvs
# Display logical volume groups managed by LVM
lvm vgs
# Where is what mounted
mount
```

The information above is now needed below.

Troubleshooting (already):

Sometimes, you might see output from frisk -l which looks strange.

For example, you have several disks reported in /dev like this:

```
/dev/sda
/dev/sda1
/dev/sda2
/dev/sda3
```

However, fdisk -l reports something strange like this:

```
Device Boot Start End Blocks Id System
/dev/sda1 1 134217727 67108863+ ee GPT
Partition 1 does not start on physical sector boundary.
```

Therefore, not all of the /dev/sda* disks are visible. This means the disk has been resized in the background, but the OS cannot read the partitions correctly.

The best way to fix this is using **parted**.

Run parted on the disk, and then input **print**.

You will be then asked for a couple of fixes. Input **Fix** at each point, then then **quit**.

Now running fdisk -l should give the correct output.

Here is an example run:

```
[root@con-jtel-dbm-2 ~]# parted /dev/sda
GNU Parted 3.1
Using /dev/sda
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) print
Error: The backup GPT table is not at the end of the disk, as it should be.  This might mean that another operating system believes the disk is smaller.  Fix, by
moving the backup to the end (and removing the old backup)?
Fix/Ignore/Cancel? Fix
Warning: Not all of the space available to /dev/sda appears to be used, you can fix the GPT to use all of the space (an extra 134217728 blocks) or continue with
the current setting?
Fix/Ignore? Fix
Model: Msft Virtual Disk (scsi)
Disk /dev/sda: 137GB
Sector size (logical/physical): 512B/4096B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name                  Flags
 1      1049kB  211MB   210MB   fat16        EFI System Partition  boot
 2      211MB   1285MB  1074MB  xfs
 3      1285MB  68.7GB  67.4GB                                     lvm
(parted) quit
```

Step 2 - Creating a new partition

Here it is necessary to know where the additional space can be found. There are 2 variants - either a new plate, or an extended plate.

In both cases, a new partition is created. Here in this example, the existing plate was extended. If a new disk was added, it can be found on /dev/sdb or /dev/sdc etc.

Adjust the commands below and the partition number (for a new disk, the partition will be 1) accordingly.

Partitioning with fdisk

```
fdisk /dev/sda

# --> Edit the partitions on /dev/sda

n
# --> Create new partition
p
# --> New primary partition
3
# --> Create new partition 3 (view output at fdisk -1 above)
Enter
# --> Confirmation that the first available cylinder should be used
Enter
# --> Confirmation that the last available cylinder is to be used (gives the maximum size in total)
t
# --> Change partition type
3
# --> Edit partition 3
8e
# --> Linux LVM
w
# --> Write
fdisk -l
# --> Check result
```

Step 3 - Adoption into LVM - Create Device

Create device for LVM

```
# Here is the previous output of /dev/sd* --> this is the new disk (the 3rd partition on /dev/sda, the first disk)
lvm pvcreate /dev/sda3
```

Step 4 - Expand Volume Group

Extend LVM

```
# Here we need the output of lvm vgs - the name of the volume group
lvm vgextend "vg_testdb5" /dev/sda3
```

Step 5 - Extend Logical Volume

Extend LVM

```
# Here the output of lvm lvs is needed - the name of the logical volume - this is combined with the vgs to create the path to the device
lvm lvresize -l +100%FREE /dev/vg_testdb5/lv_root
```

Step 6 - Extend File System

Extend LVM - ext filesystems

```
# Same path as previous command
resize2fs /dev/vg_testdb5/lv_root
```

Extend LVM - xfs filesystems

```
# Mount Point from fstab
xfs_growfs /srv/jtel/shared
```

Step 7 - Check final result!

Check final result

```
df -h
```