–LVM USING MULTIPATH : LABS

Erfan Hooman
November 14, 2024

CREATE PHYSICAL PARTITIONS

- **add two virtual hard drive of 10 GiB to VM**
- **use fdisk to partition the disks**

$ sudo fdisk /dev/sdb
n
p

w

- do the same thing for the sdc/ sdf / sdg
-
- **ensure that partitioning work as expected**

$ lsblk

CREATE THE RAID 6 ARRAY

- **Create**

$ sudo mdadm --create --verbose /dev/md0 --level=6 --raid-devices=4 /dev/sdb /dev/sdc /dev/sdf /dev/sdg
y

- **Verify RAID Array Creation**

$ cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4]
md0 : active raid6 sdg[3] sdf[2] sdc[1] sdb[0]
    20953088 blocks super 1.2 level 6, 512k chunk, algorithm 2 [4/4] [UUUU]

unused devices: <none>

- **Create a Filesystem on the RAID Array**

$ sudo mkfs.ext4 /dev/md0

ADD THE RAID ARRAYS TO LVM

- **Create Physical Volume**

$ sudo pvcreate /dev/md0
$ sudo pvcreate /dev/md1

- **Create a Volume Group**

$ sudo vgcreate erfan_raid /dev/md0 /dev/md1

- **Create Logical Volumes**

$ sudo lvcreate -L 20G -n raid_1 erfan_raid
$ sudo lvcreate -l 100%FREE -n raid_2 erfan_raid

- **Format the Logical Volumes**

$ sudo mkfs.ext4 /dev/vg_raid/lv_data
$ sudo mkfs.ext4 /dev/vg_raid/lv_backup

- **now you can mount and use it** Z

THE FINAL $LSBLK LOOK LIKE THIS

```
root@localhost-live:/shared# lsblk
NAME                  MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINTS
loop0                   7:0    0  2.1G  1 loop  /run/rootfsbase
sda                     8:0    0    5G  0 disk
└─mpatha              253:0    0    5G  0 mpath
  ├─mpatha1           253:1    0    1M  0 part
  ├─mpatha2           253:2    0    1G  0 part
  └─mpatha3           253:3    0    4G  0 part  /run/media/liveuser/fedora
sdb                     8:16   0   10G  0 disk
└─md0                   9:0    0   20G  0 raid6
  └─erfan_raid-raid_1 253:4    0   20G  0 lvm   /run/media/liveuser/9a3543c8-71dd-4a9d-816b-8e021d58ca57
sdc                     8:32   0   10G  0 disk
└─md0                   9:0    0   20G  0 raid6
  └─erfan_raid-raid_1 253:4    0   20G  0 lvm   /run/media/liveuser/9a3543c8-71dd-4a9d-816b-8e021d58ca57
sdd                     8:48   0    5G  0 disk
├─sdd1                  8:49   0    5G  0 part
└─md1                   9:1    0   10G  0 raid6
  ├─erfan_raid-raid_1 253:4    0   20G  0 lvm   /run/media/liveuser/9a3543c8-71dd-4a9d-816b-8e021d58ca57
  └─erfan_raid-raid_2 253:5    0   10G  0 lvm
sde                     8:64   0    5G  0 disk
├─sde1                  8:65   0    5G  0 part
└─md1                   9:1    0   10G  0 raid6
  ├─erfan_raid-raid_1 253:4    0   20G  0 lvm   /run/media/liveuser/9a3543c8-71dd-4a9d-816b-8e021d58ca57
  └─erfan_raid-raid_2 253:5    0   10G  0 lvm
sdf                     8:80   0   10G  0 disk
└─md0                   9:0    0   20G  0 raid6
  └─erfan_raid-raid_1 253:4    0   20G  0 lvm   /run/media/liveuser/9a3543c8-71dd-4a9d-816b-8e021d58ca57
sdg                     8:96   0   10G  0 disk
└─md0                   9:0    0   20G  0 raid6
  └─erfan_raid-raid_1 253:4    0   20G  0 lvm   /run/media/liveuser/9a3543c8-71dd-4a9d-816b-8e021d58ca57
sdh                     8:112  0    5G  0 disk
├─sdh1                  8:113  0    5G  0 part
└─md1                   9:1    0   10G  0 raid6
  ├─erfan_raid-raid_1 253:4    0   20G  0 lvm   /run/media/liveuser/9a3543c8-71dd-4a9d-816b-8e021d58ca57
  └─erfan_raid-raid_2 253:5    0   10G  0 lvm
sdi                     8:128  0    5G  0 disk
├─sdi1                  8:129  0    5G  0 part
└─md1                   9:1    0   10G  0 raid6
  ├─erfan_raid-raid_1 253:4    0   20G  0 lvm   /run/media/liveuser/9a3543c8-71dd-4a9d-816b-8e021d58ca57
  └─erfan_raid-raid_2 253:5    0   10G  0 lvm
sr0                    11:0    1  2.3G  0 rom   /run/initramfs/live
zram0                 252:0    0  1.9G  0 disk  [SWAP]
```

LVM-HOMEWORKS

ASYMMETRY IN RAID 4

- In RAID 4, all parity information is written to a dedicated disk, while data is striped across the other disks. This dedicated parity disk creates an asymmetric workload, as it becomes a bottleneck for write operations:
- Every write operation, even if small, must update the parity on the dedicated parity disk.
- This dedicated parity disk tends to wear out faster and becomes a performance bottleneck, as all write operations require access to it.

WRITE-BACK CACHING IN RAID 5

- Write-back caching can significantly help mitigate the write bottleneck and improve the asymmetry issues in RAID 5. In write-back caching:
- Write operations are temporarily stored in a cache (typically in the RAID controller's memory) and acknowledged to the system as completed.
- This allows the system to proceed without waiting for the actual disk write operation to finish, reducing latency and improving overall performance