Replacing a XCP-NG Raid1 disk and growing the array

Check RAID status

```
cat /proc/mdstat
```

Identify RAID devices and members

```
mdadm --detail /dev/md0
```

If `/dev/md0` doesn't exist, check which md device is being used:

```
lsblk | grep md
```

Check physical disks and partitions

```
lsblk -o NAME,SIZE,TYPE,UUID,MOUNTPOINT
```

Check volume groups and logical volumes (if LVM is used)

```
vgs
lvs
```

Check mount points and filesystems

```
df -h
```

For Lenovo m720q host:

```
cat /proc/mdstat # check everything is synced

mdadm /dev/md127 --fail /dev/nvme0n1 # fail drive
mdadm /dev/md127 --remove /dev/nvme0n1 # remove drive

mdadm --detail /dev/md127 # double check
```

```
shutdown -h now # power off host
```

Swap disk out.

```
sgdisk -R=/dev/nvme0n1 /dev/sda #  clone partiion layout of /dev/sda
sgdisk -G /dev/nvme0n1   # Give it a new unique GPT GUID
```

This will mirror the partitions and metadata exactly. The commands assume that the new drive is still /dev/nvme0n1.

```
mdadm /dev/md127 --add /dev/nvme0n1 # add new disk to the RAID
watch cat /proc/mdstat # check rebuild progress
```

Wait until [UU] is shown again and rebuild completes (could take 15–60+ minutes depending on system load and speed).

```
mdadm --grow /dev/md127 --size=max # grow the raid array to use the full disk
mdadm --detail /dev/md127 # verify that its has been successful
```

Resize the Partition Table on /dev/md127

```
gdisk /dev/md127
```

- In `gdisk`:
  - `p` — print partition table (record start of partition 3)
  - `d` — delete partition 3
  - `n` — create new partition:
    - Partition number: 3
    - First sector: type the exact start sector (e.g., `75497472`)
    - Last sector: just press Enter to accept default (uses remaining space)
    - Hex code: default (`8e00` for LVM)
  - `w` — write and exit

Double-check the start sector of `md127p3` beforehand using `lsblk -o NAME,START,SIZE` or `gdisk`

```
pvresize /dev/md127p3 # resize the LVM PV
```

Expand Your Volume Group or Logical Volumes

```
vgs # verify free space
lvextend -l +100%FREE /dev/VG_XenStorage-xxx/VHD-xxxxx #example of estending the volume
resize2fs /dev/VG_XenStorage-xxx/VHD-xxxxx # grow filesystem if needed (EXT4)
```

```
xfs_growfs /mount/point # (if XFS)
```

You can leave the free space in the VG for XenServer to manage VM disks.

Final checks

```
mdadm --detail --scan >> /etc/mdadm.conf # update mdadm config
```

Double-check everything work;

```
cat /proc/mdstat
vgs
lvs
df -h
```

Other notes

- `VG_XenStorage-xxx`: **The Volume Group (VG) name**

```
vgs
```

- `VHD-xxxxx`: **The Logical Volume (LV) name**

```
lvs # can add VG name
```

Crafted with the assistance of https://chatgpt.com/share/68328b0c-5094-8001-b27a-d7a27a61cdfc

---

Revision #5
Created 2025-05-25 02:47:45 UTC by Admin
Updated 2025-05-28 03:43:24 UTC by Admin