Basic Proxmox Configuration

- ZFS Storage
- Wake-on-LAN
- Disable Subscription Message
- Switch Repository to non-enterprise
- GPU Passthrough
- Docker in LXC
- Powersaving

ZFS Storage

I use ZFS as filesystem for my backend storage, due to its seamless integration with Snapshots and the Proxmox Backup Server. It is decently performant compared to the likes of ext, but offers great resilience in RAID-like solutions, is flexible and easy to use. I recommend reading through the basic ZFS introduction to clarify terms and usecases before following this guide, as this only reflects a good usecase for me.

1. Create Pool
    1. `Zpool create -o ashift=12 mirror POOL /dev/sdc /dev/sdd mirror /dev/sde /dev/sdf`
    2. Hardening against reboot issues with /dev/sdX nomenclature
        1. `Zpool export POOL`
        2. `Zpool import -d /dev/disk/dev-by-id`
        3. `Zpool import -c /etc/zfs/zpool.cache`
        4. `Zpool status`
2. Set compression: `zfs set compression=lz4 POOL`
3. Create datasets (+tuning)
    1. `zfs create pool/dataset-name`
    2. `sudo zfs set recordsize=[size] data/media/series`
    3. No-snap für datenbanksets d. Set mountpoints for datasets e. ZFS Scrub
4. Set mountpoints for datasets
5. ZFS Scrub
6.

https://docs.oracle.com/cd/E24841_01/html/820-2313/gbbwa.html

https://www.usenix.org/system/files/login/articles/login_winter16_09_jude.pdf

https://shatteredsilicon.net/blog/2020/06/05/mysql-mariadb-innodb-on-zfs/

https://docs.oracle.com/cd/E24841_01/html/820-2313/gbbwa.html

https://unix.stackexchange.com/questions/288599/forcing-zpool-to-use-dev-disk-by-id-in-ubuntu-xenial https://docs.oracle.com/cd/E19253-01/819-5461/gaynr/index.html

Wake-on-LAN

https://www.jm.technology/post/proxmox-wol-mac-set_april-2019/

https://pve.proxmox.com/wiki/Proxmox_Node_Management

Disable Subscription Message

https://johnscs.com/remove-proxmox51-subscription-notice/

```
sed -Ezi.bak "s/(Ext.Msg.show\(\{\s+title: gettext\('No valid sub)/void\(\{ \/\/\1/g" /usr/share/javascript/proxmox-widget-toolkit/proxmoxlib.js && systemctl restart pveproxy.service
```

Switch Repository to non-enterprise

The enterprise repository requires a bought key, hence the usage of Proxmox in a homelab usually lead to the usage of the non-production repository.

1. Comment out everything with "#" in /etc/apt/sources.list/pve-enterprise.list
2. create new file in /etc/apt/sources.list/pve-no-enterprise.list
3. insert:
   `deb http://download.proxmox.com/debian/pve buster pve-no-subscription`
4. replace "buster" with name of debian version corresponding to Proxmox version.

Sources

https://pve.proxmox.com/wiki/Package_Repositories

GPU Passthrough

1. Kernel modules
2. Disable drivers (Intel for iGPU, Nvidia for dedicated)
3. Set up VMs accordingly
    1. https://pve.proxmox.com/wiki/PCI(e)_Passthrough
    2. https://pve.proxmox.com/wiki/Pci_passthrough
    3. https://blog.quindorian.org/2018/03/building-a-2u-amd-ryzen-server-proxmox-gpu-passthrough.html/

       https://cetteup.com/216/how-to-use-an-intel-vgpu-for-plexs-hardware-accelerated-streaming-in-a-proxmox-vm/

Docker in LXC

Docker LXC https://kayomo.de/blog/docker-unter-proxmox-installieren/

Powersaving

Powergoverons

Required: -

Related: -

1. `apt install cpufrequtils`
2. Check available govenors with `cpufreq-info` or `cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_available_governors`
3. apply governor on reboot in crontab with `echo "#NAMEOFGOVENOR" | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor`

See also: -

Sources:

https://wiki.debian.org/CpuFrequencyScaling