

ZFS

Die allgemeine Empfehlung ist ZFS unter Linux nicht zu verwenden, da die Lizenz, unter der ZFS freigegeben wurde, problematisch ist und Oracle Besitzer des eigentlichen ZFS Code, ehemals Sun, ist. Daher müsst Ihr selbst wissen, ob Ihr das Dateisystem einsetzt.

Beschreibung

- Dateisystem entwickelt von Sun für Solaris
- wurde portiert auf FreeBSD, Linux und OpenSolaris(und entsprechenden Derivaten)
- bildet ein Software RAID und Volumenmanagement ab (unter Linux als Alternative BTRFS oder mdadm mit lvm und ext4 oder ähnlichem)

Anleitung

Empfehlung für Hetzner: Standard Debian installieren (boot(400MB), root(10-20GB) und swap(2x Memory) per default). Dann proxmox installieren und zum Schluss den restlich Platz von ZFS als zpool konfigurieren. ¹⁾

single Pool

```
zpool create -f <pool> <device1>
```

mirror aka RAID1 Pool

```
zpool create -fd <pool> mirror <device1> <device2>
```

RAIDZ1

```
zpool create -f -o ashift=12 <pool> raidz1 <device1> <device2> <device3>
```

- braucht mindestens drei Geräte und
- kann auch Partition anstatt komplettes Device sein

create dataset

```
zpool create zpool0/dataset
```

share nfs

```
sudo apt-get install -y nfs-kernel-server  
zfs set sharenfs=on pool-name/dataset-name
```

- muss nach einen Reboot mit `zfs set sharenfs pool-name/dataset-name` wieder exportiert werden

Option

```
zfs set sharenfs="rw=@192.168.11.0/24" pool-name/dataset-name
zfs set share="name=my-
share,path=/path/on/server,prot=nfs,sec=sys,rw=*,public" pool-name/dataset-
name
zfs set sharenfs="rw=@192.168.0.0/24,rw=@10.0.0.0/24" pool-name/dataset-
name
zfs set
sharenfs=10.1.1.0/24:rw,fsid=100,no_subtree_check,async,no_root_squash
zfs/home #Einstellungen für ESX(i)
```

Shares prüfen

```
exportfs
```

- zeigt die exportierten Verzeichnisse an

export nach neustart

[/etc/rc.local](#)

```
# add this to /etc/rc.local
zfs share -a
```

alternative über systemd

[/etc/systemd/system/autostart.service](#)

```
[Unit]
Description=Autostart

[Service]
Type=oneshot
ExecStart=/sbin/zfs share -a
```

2)

Speicherbelegung im RAM

[/etc/modprobe.d/zfs.conf](#)

```
options zfs zfs_arc_max=8589934592
```

```
update-initramfs -u
```

- Speicherbelegung im RAM für ZFS
 - Arbeitsspeicher in MB * 1024 * 1024
 - Empfehlung ist maximal die Hälfte des Arbeitsspeichers
 - das Betriebssystem sollte 4GB RAM haben → mehr ist besser!

Index

C

- [cache](#)

C

- [compression](#)

D

- [Debian and proxmox](#)

S

- [swap](#)

Links

- <http://blog.programster.org/zfs-create-disk-pools>
- https://pve.proxmox.com/wiki/ZFS_on_Linux
- https://pve.proxmox.com/wiki/ZFS:_Tips_and_Tricks
- [zfs](#)
- https://docs.oracle.com/cd/E23824_01/html/821-1448/gayne.html ← Befehlsset von Oracle
- <https://blog.programster.org/sharing-zfs-datasets-via-nfs> ← Beispiel für export als NFS
- <http://codebytez.blogspot.com/2011/06/exporting-zfs-file-system-over-nfs.html> ← Einstellungen zum NFS Export
- <http://www.vmwareadmins.com/using-zfs-file-system-as-an-nfs-datastore-for-esxi/>
- http://pubman.mpg.de/pubman/item/escidoc:2403926/component/escidoc:2403925/Then_Bachelor.pdf ←
then_bachelor.pdf
- <http://www.zfsbuild.com/2010/05/26/zfs-raid-levels/>
- <https://www.hiroom2.com/2016/05/18/ubuntu-16-04-share-zfs-storage-via-nfs-smb/#sec-4> ←

ZFS mit Samba

1)

Stand:01.2018

2)

<https://bbs.archlinux.de/viewtopic.php?id=21655>

From:

<https://wiki.lug-wr.de/wiki/> - **Wiki der Linux User Group Wernigerode**

Permanent link:

<https://wiki.lug-wr.de/wiki/doku.php?id=public:sw:linux:fs:zfs:start&rev=1584350931>

Last update: **2020/03/16 10:28**

