ZFS Grundlagen am Beispiel Proxmox VE(Stand Dezember 2025)

ZFS Grundlagen mit Proxmox VE – Praxiswissen für sichere Server, Backups und Storage-Systeme

Erfahrung aus rund 150 produktiven Proxmox- und ZFS-Systemen

Der Verfasser dieser Dokumentation, **Christian-Peter Zengel**, verfügt über rund **18 Jahre Praxiserfahrung mit ZFS und Proxmox** und betreibt aktuell etwa **150 Systeme mit Proxmox VE und ZFS**.

Die hier beschriebenen Erfahrungen stammen aus realen Installationen – von einzelnen Standalone-Servern bis zu Proxmox-Clustern mit bis zu zehn Hosts.

Der Fokus liegt auf **ZFS als robuste, sichere und wirtschaftliche Storage-Grundlage** für kleine und mittlere IT-Umgebungen.

Warum ZFS für Proxmox, TrueNAS und Linux-Server?

ZFS ist eine ideale Grundlage für Systeme, bei denen Datenintegrität, Snapshots, Replikation und schnelle Wiederherstellung entscheidend sind. Gerade bei Proxmox VE, TrueNAS, Backup-Servern und Fileservern bietet ZFS erhebliche Vorteile gegenüber klassischen RAID-Controllern oder einfachen Dateisystemen.

- Hohe Datensicherheit durch Prüfsummen
- Snapshots in Sekunden
- Rollbacks bei Fehlern oder Updates
- Replikation auf andere Systeme
- Schutz vor Bedienfehlern
- Bessere Wiederherstellung nach Ausfällen
- Sehr gute Grundlage gegen Ransomware-Schäden
- Flexible Storage-Konzepte ohne klassischen RAID-Controller

In der Praxis zeigt sich ZFS für viele kleine und mittlere Systeme einfacher, vielseitiger, schneller und günstiger als Ceph. Eine produktive Ceph-Expertise wird hier bewusst nicht behauptet; intern wird jedoch intensiv an passenden Hauslösungen gearbeitet.

ZFS Onlinekurs und Schulung

Diese Dokumentation basiert auf einem Onlinekurs von **cloudistboese.de**. Die ZFS-Grundlagenkurse werden regelmäßig angeboten und vermitteln die wichtigsten Konzepte für den sicheren Einsatz von ZFS mit Proxmox VE, TrueNAS und vergleichbaren Systemen.

Aktuelle ZFS-Kurse und weitere Open-Source-Schulungen auf cloudistboese.de ansehen

Die Wiederholung der Onlinekurse sowie der Zugriff auf Aufzeichnungen sind für Kursteilnehmer kostenlos.

Inhalte des ZFS-Grundlagenkurses

ZFS ist die perfekte Grundlage, um kleine und mittlere Systeme gegen Ausfälle, Fehlbedienung und Verschlüsselungstrojaner besser abzusichern.

Der Kurs vermittelt Grundkenntnisse für den praktischen Einsatz mit:

- Proxmox VE
- TrueNAS
- Linux-Servern

- Backup-Systemen
- Fileservern
- kleinen und mittleren Virtualisierungsumgebungen

Themen im Überblick

- Grundbegriffe und Überblick über ZFS
- Aufbau von RAID-Konzepten mit `zpool`
- ZFS-Basisfunktionen am Beispiel von Proxmox VE
- Snapshots und Rollbacks
- Replikation in der Praxis
- Prüfung von Replikationen
- Datasets, ZVOLs und Storage-Struktur
- Cache, Logdevice und Special Device
- Typische Fehlentscheidungen vermeiden

Nach diesem Kurs sind Sie bei Ausfällen, Bedienfehlern und Datenverlust deutlich besser vorbereitet und können viele Systeme innerhalb weniger Minuten wieder produktiv machen.

ZFS RAID mit zpool erstellen

So erstellt Proxmox VE einen ZFS Mirror

```
zpool create -f -o cachefile=none -o ashift=12 rpool mirror disk1 disk2
```

Die wichtigsten Parameter:

- `-f` erzwingt das destruktive Anlegen des Pools
- `-o cachefile=none` verhindert automatischen Import beim Start
- `ashift=12` ist die übliche Wahl für moderne 4K-Sektoren
- `rpool` ist der Name des Pools
- `mirror` definiert den RAID-Level
- Disks sollten eindeutig über `/dev/disk/by-id` angesprochen werden

Mögliche RAID-Varianten sind unter anderem:

- Stripe

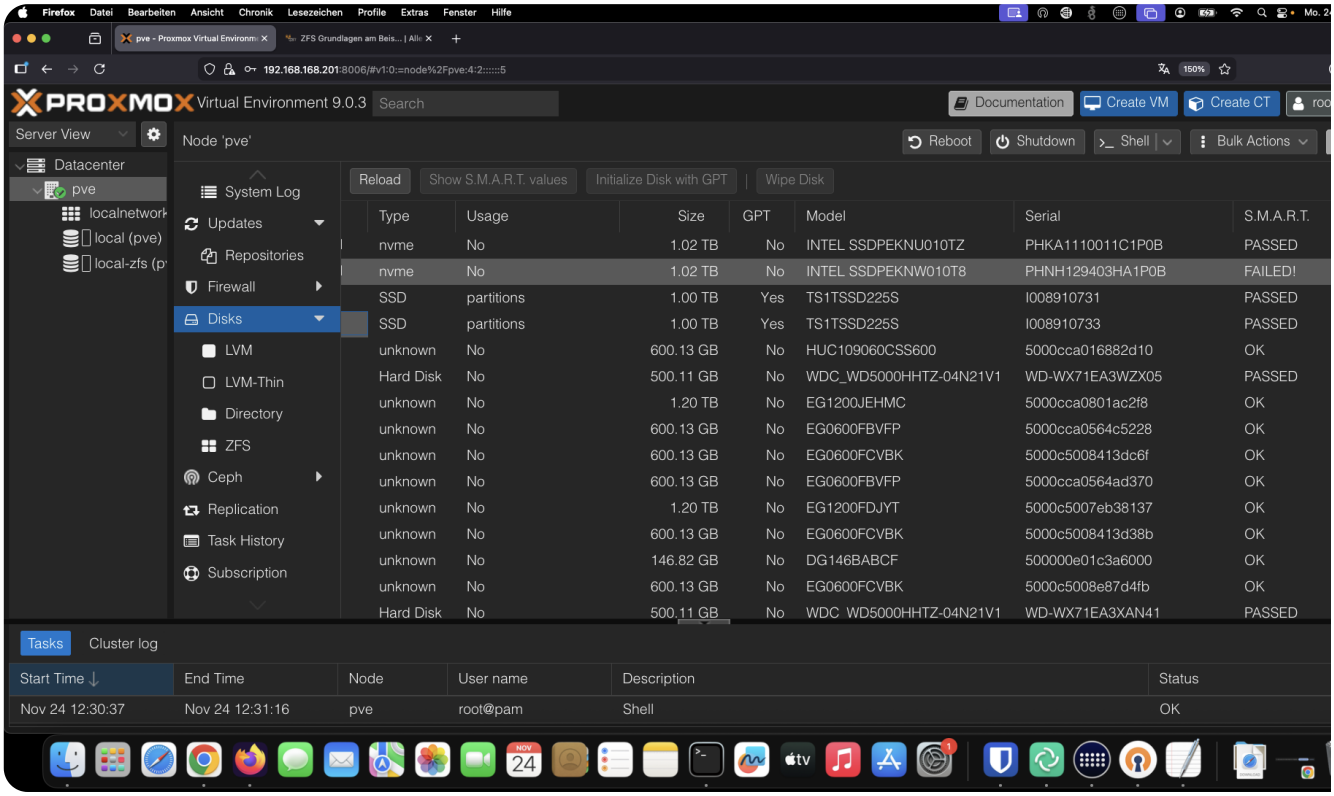
- Mirror
- Striped Mirror / RAID10
- RAIDZ1
- RAIDZ2
- RAIDZ3

Praxisumgebung im Kurs

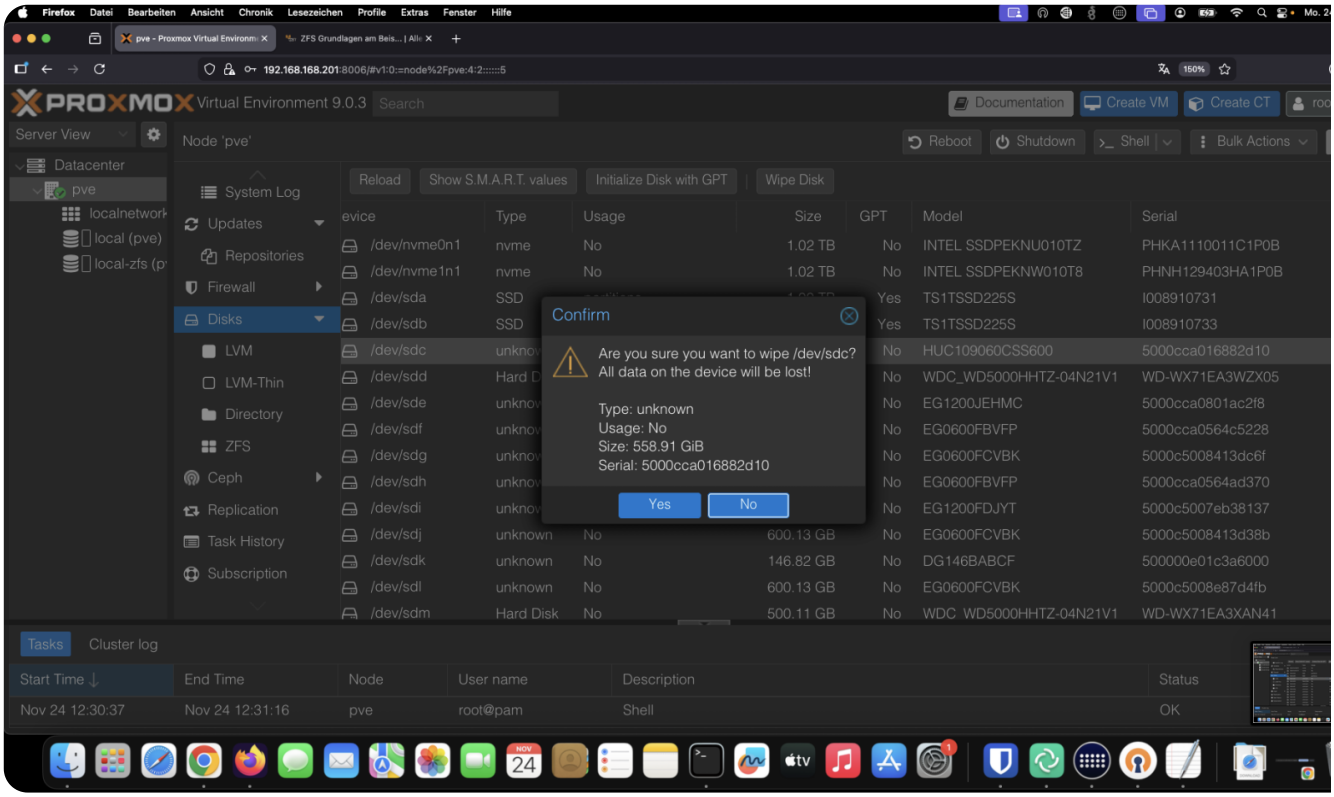
Im Kurs wird ein älterer Supermicro-Server mit 24 Einschüben sowie zwei NVMe-Datenträgern verwendet. Damit lassen sich verschiedene ZFS-Szenarien realistisch abbilden.

Die Verwendung gebrauchter und teilweise ungetesteter Datenträger zeigt sehr schnell, warum ZFS in der Praxis so wertvoll ist.

Empfehlung: Für produktive Systeme immer Enterprise-NVMe und ECC-RAM verwenden.



Bereits verwendete Datenträger können in Proxmox VE über die Wipe-Funktion bereinigt werden. Bei alten LVM- oder Ceph-Installationen empfiehlt sich jedoch häufig eine vorherige vollständige Löschung, beispielsweise mit `nwipe`.



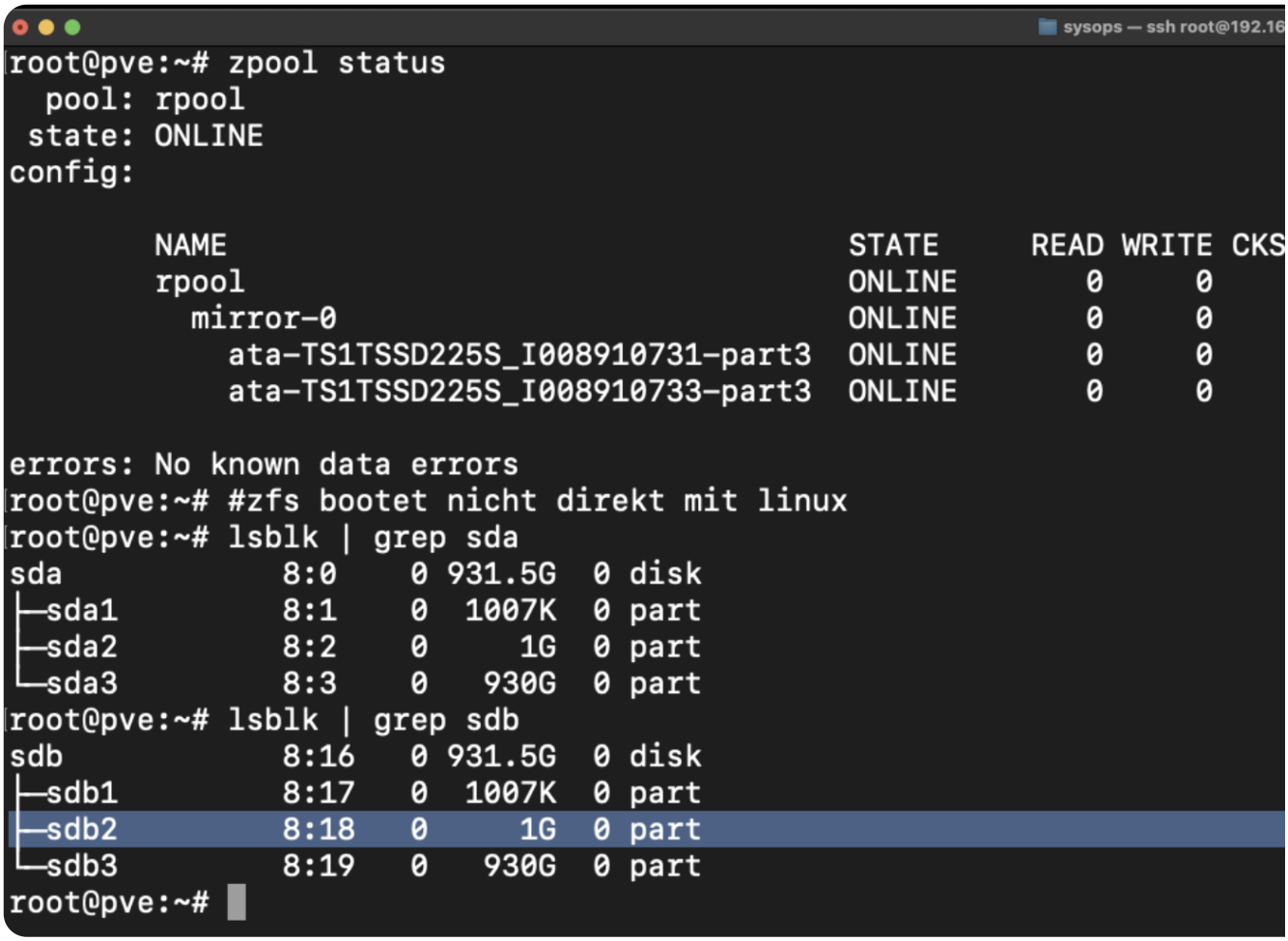
Proxmox VE Bootlayout mit ZFS verstehen

Proxmox VE nutzt bei ZFS-Installationen ein spezielles Bootlayout. Wichtig ist, dass immer ausreichend bootfähige Datenträger mit passendem Partitionslayout vorhanden sind.

Typischerweise gibt es:

- eine kleine Partition für BIOS/GRUB
- eine EFI-Systempartition für `proxmox-boot-tool`
- eine ZFS-Partition für den eigentlichen Pool

Wird später unbedacht mit ganzen Disks gearbeitet, kann es passieren, dass keine geeignete Bootstruktur mehr vorhanden ist.



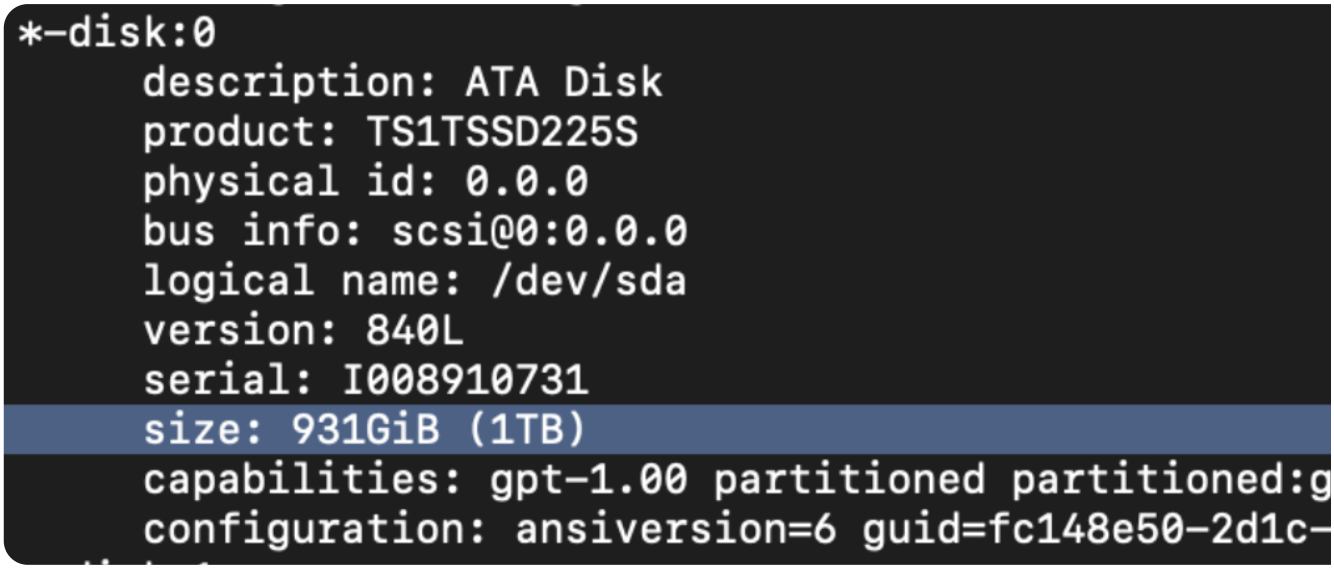
Laufwerke unter Linux eindeutig ermitteln

Vor dem Erstellen eines ZFS-Pools müssen die Laufwerke eindeutig identifiziert werden. Dafür eignen sich unter Linux mehrere Werkzeuge.

```
apt install lshw iotop

# Detaillierte Übersicht aller Laufwerke
lshw -class disk

# Liveansicht beim Einschieben von Festplatten
dmesg -Tw
```

```
# Eindeutige Laufwerksnamen über by-id anzeigen
ls -althr /dev/disk/by-id
```

Beispiel: lshw



Beispiel: /dev/disk/by-id

Wichtige Entscheidungen beim Anlegen eines ZFS-Pools

Die wichtigsten Entscheidungen beim Anlegen eines ZFS-Pools sind:

- RAID-Level
- `ashift`
- Kompression
- Volblocksize
- Dataset-Struktur
- Cache- und Log-Konzept

Fehler an dieser Stelle lassen sich später häufig nur mit erheblichem Aufwand korrigieren.

RAID-Level: Mirror, RAID10 oder RAIDZ?

Mirror und Striped Mirror

Mirror und Striped Mirror sind für Proxmox VE meist die unproblematischste Wahl. Sie kosten zwar 50 % der Rohkapazität, bieten aber hohe Performance, einfache Erweiterbarkeit und sehr gute Wiederherstellbarkeit.

- **Vorteile:** schnell, einfach, robust
- **Nachteile:** 50 % Kapazitätsverlust
- **Ideal für:** VMs, LXCs, produktive Proxmox-Hosts

RAIDZ1, RAIDZ2 und RAIDZ3

RAIDZ bietet mehr nutzbare Kapazität, ist aber für VMs und LXCs auf drehenden Festplatten nur eingeschränkt geeignet.

- **Vorteile:** mehr nutzbarer Speicherplatz
- **Nachteile:** schlechter für viele kleine synchrone Schreibzugriffe
- **Ideal für:** Fileserver, Archivspeicher, Backup-Ziele

Virtuelle Maschinen sollten im Normalfall nicht auf drehenden Festplatten mit RAIDZ betrieben werden. Als Kompromiss kann RAIDZ mit NVMe Cache und Logdevice eingesetzt werden.

Ashift richtig wählen

Bei ZFS bezeichnet `ashift` die interne Sektorgröße eines VDEVs. Früher wurde häufig mit `ashift=9` für 512-Byte-Sektoren gearbeitet. Für heutige Datenträger ist `ashift=12` praktisch immer die richtige Wahl.

Empfehlung: Aktuell immer `ashift=12` verwenden.

Eine falsche Einstellung kann zu massiven Leistungseinbußen und erheblicher Platzverschwendung führen.

Technischer Hintergrund zu ZFS ashift

Kompression mit ZFS

ZFS-Kompression kann erheblich Speicherplatz sparen, insbesondere bei Datenbanken, Logs, Textdateien, VM-Images und vielen Serverdaten.

Unsere heutige Empfehlung:

```
zfs set compression=on poolname/dataset
```

Früher wurde häufig explizit `compression=lz4` gesetzt. In vielen aktuellen Umgebungen ist `compression=on` sinnvoll und ausreichend.

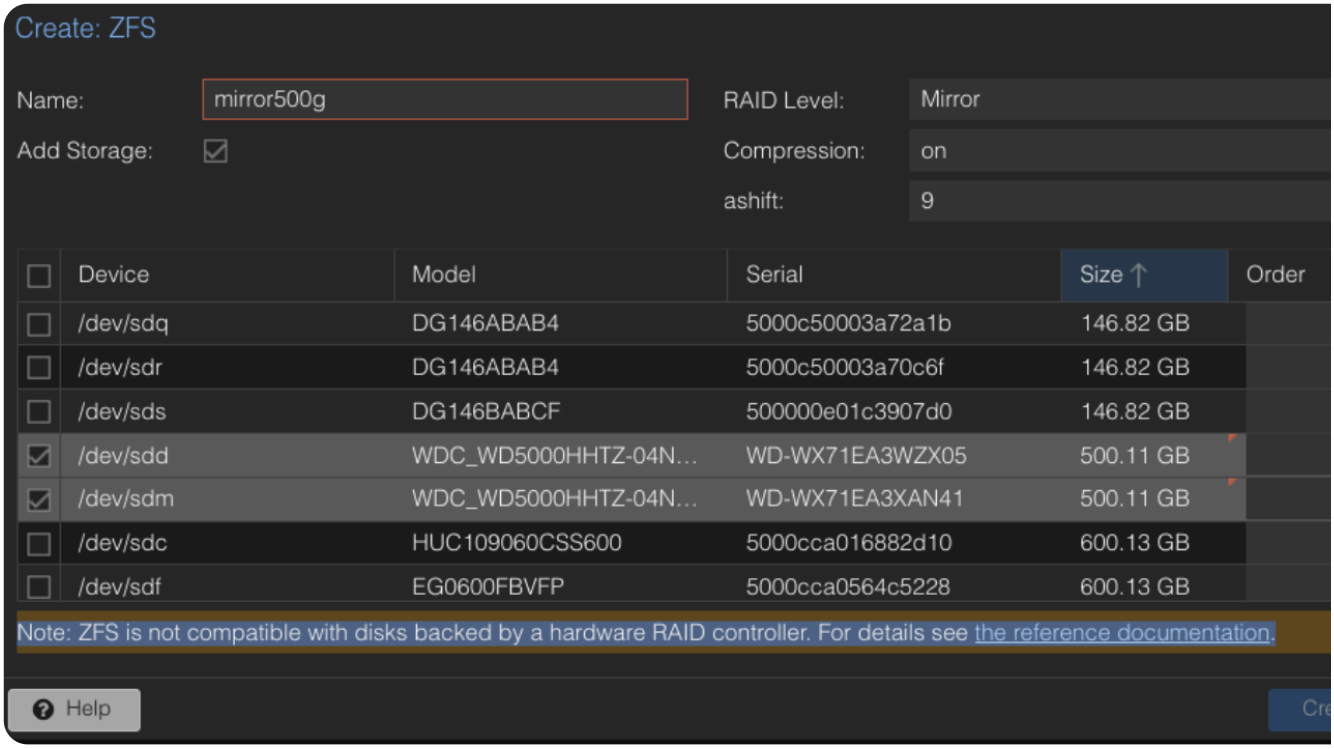
In der Praxis sparen wir damit häufig ein Drittel oder mehr Speicherplatz – ohne spürbare Geschwindigkeitseinbußen.

ZFS Dataset-Struktur für Proxmox VE

Nach dem Erstellen eines Pools sollte nicht einfach der Pool selbst als Proxmox Storage verwendet werden. Besser ist eine saubere Struktur mit getrennten Datasets.

```
zfs create mirror500g/data
zfs create mirror500g/clone
zfs create -o com.sun:auto-snapshot=false mirror500g/repl
```

Danach kann der Storage in Proxmox VE sauber als beispielsweise `mirror500g-data` eingebunden werden.



Praxisbeispiele: zpool create

Stripe / RAID0

Nur für Testsysteme oder Daten ohne Schutz sinnvoll.

```
zpool create -f test \
wwn-0x5000cca01a72ded4 \
wwn-0x5000cca01a7b1e2c \
wwn-0x5000cca01a83a61c
```

Mirror / RAID1

```
zpool create -f test mirror \
wwn-0x5000cca01a72ded4 \
wwn-0x5000cca01a7b1e2c
```

Striped Mirror / RAID10

```
zpool create -f test \
mirror wwn-0x5000cca01a72ded4 wwn-0x5000cca01a7b1e2c \
mirror wwn-0x5000cca01a83a61c wwn-0x5000cca01a832d24 \
mirror wwn-0x5000cca01a83331c wwn-0x5000cca01a7495b8
```

RAIDZ1

```
zpool create test raidz \
wwn-0x5000cca01a72ded4 \
wwn-0x5000cca01a7b1e2c \
wwn-0x5000cca01a83a61c \
wwn-0x5000cca01a832d24 \
wwn-0x5000cca01a83331c \
wwn-0x5000cca01a7495b8
```

RAIDZ2

```
zpool create test raidz2 \
wwn-0x5000cca01a72ded4 \
wwn-0x5000cca01a7b1e2c \
wwn-0x5000cca01a83a61c \
wwn-0x5000cca01a832d24 \
wwn-0x5000cca01a83331c \
wwn-0x5000cca01a7495b8
```

RAIDZ mit mehreren VDEVs

```
zpool create test \
raidz wwn-0x5000cca01a72ded4 wwn-0x5000cca01a7b1e2c wwn-0x5000cca01a83a61c \
```

```
raidz wwn-0x5000cca01a832d24 wwn-0x5000cca01a83331c wwn-0x5000cca01a7495b8
```

Dry-Run vor dem Anlegen

Vor destruktiven Aktionen immer prüfen:

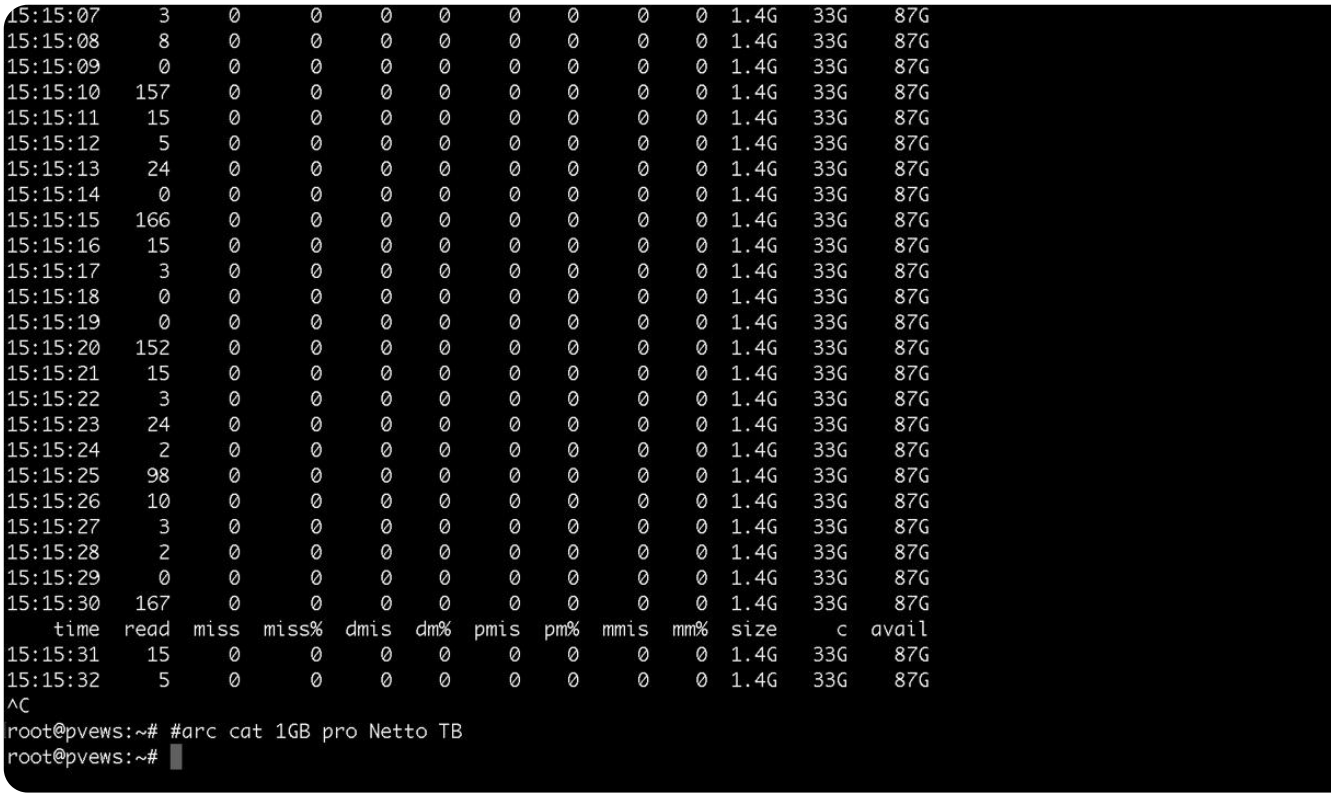
```
zpool create -n test mirror disk1 disk2
```

ZFS Cache, ARC, L2ARC und Logdevice

ARC – Cache im RAM

Der ARC ist der First-Level-Cache von ZFS und nutzt RAM. Als grobe Praxisregel kann man etwa 1 GB RAM pro 1 TB Netto-Daten ansetzen. Entscheidend ist jedoch immer der konkrete Einsatzzweck.

```
arcstat 1
```



L2ARC – zusätzlicher Lesecache

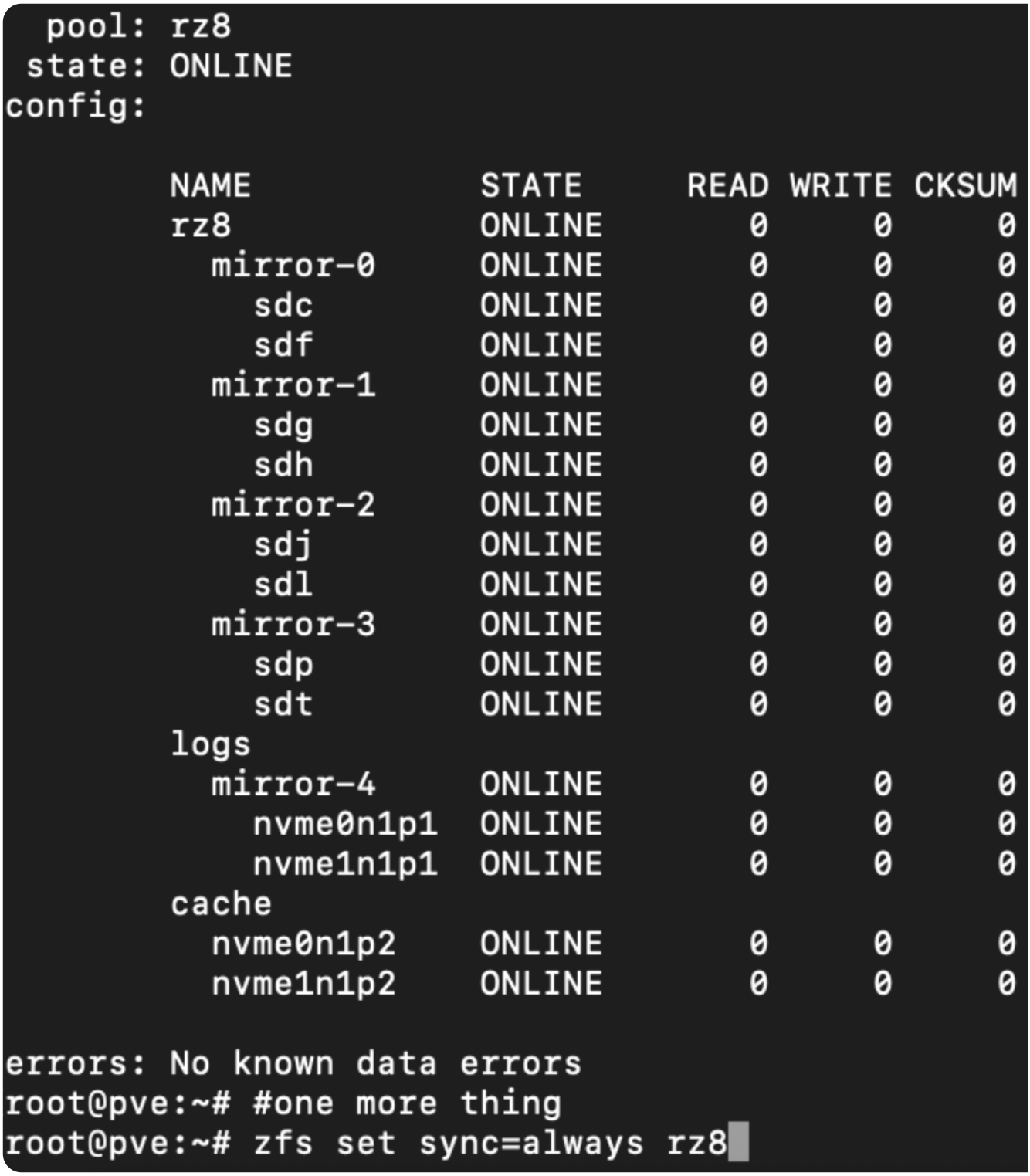
Ein L2ARC kann als schneller Lesecache auf SSD oder NVMe eingesetzt werden. Er ist besonders interessant, wenn der Arbeitsspeicher nicht ausreicht und viele wiederkehrende Lesezugriffe auftreten.

```
zpool add test cache /dev/disk/by-id/nvme-cache-device
```

SLOG / Logdevice

Ein Logdevice kann synchrone Schreibzugriffe beschleunigen, muss aber zwingend zuverlässig und im produktiven Einsatz gespiegelt sein.

```
zpool add test log mirror /dev/disk/by-id/nvme-log-1 /dev/disk/by-id/nvme-log-2
zfs set sync=always test
```



ZFS Special Device

Ein Special Device kann Metadaten und kleine Blöcke auf schnelle SSDs oder NVMes auslagern. Es muss zwingend redundant, also mindestens gespiegelt, aufgebaut werden.

Sinnvoll wird ein Special Device erst mit einer passenden Definition der Blockgröße, beispielsweise für kleine Blöcke bis etwa 64K.

In unseren Tests zeigte ein Special Device vor allem bei Fileservern und LXC-Workloads Vorteile. Im klassischen VM-Betrieb war der Nutzen deutlich geringer.

```
root@pve:~# zpool add rz8 -n special mirror nvme0n1p2 nvme1n1
would update 'rz8' to the following configuration:

        rz8
          mirror-0
            sdc
            sdf
          mirror-1
            sdg
            sdh
          mirror-2
            sdj
            sdl
          mirror-3
            sdp
            sdt
        special
          mirror
            nvme0n1p2
            nvme1n1p2
root@pve:~# zpool add rz8 special mirror nvme0n1p2 nvme1n1p2
```

```
root@pve:~# zfs set special_small_blocks=64k r
```

ZFS Pool importieren und entfernen

ZFS-Pools können importiert werden, ohne sie zu löschen. Besonders sinnvoll ist der Import über `/dev/disk/by-id`, damit später im Status eindeutige Laufwerksnamen angezeigt werden.

```
zpool import
```

```
zpool import poolname


zpool import -d /dev/disk/by-id poolname
```

```
root@pvews:~# zpool import
   pool: test
     id: 14075249358432111533
  state: ONLINE
 action: The pool can be imported using its name or numeric identifier.
 config:

        test                        ONLINE
          mirror-0                  ONLINE
            wwn-0x5000cca01a72ded4  ONLINE
            wwn-0x5000cca01a7b1e2c  ONLINE
          mirror-1                  ONLINE
            wwn-0x5000cca01a83a61c  ONLINE
            wwn-0x5000cca01a832d24  ONLINE
          mirror-2                  ONLINE
            wwn-0x5000cca01a83331c  ONLINE
            wwn-0x5000cca01a7495b8  ONLINE
root@pvews:~#
```

Datasets und ZVOLs verstehen

Datasets

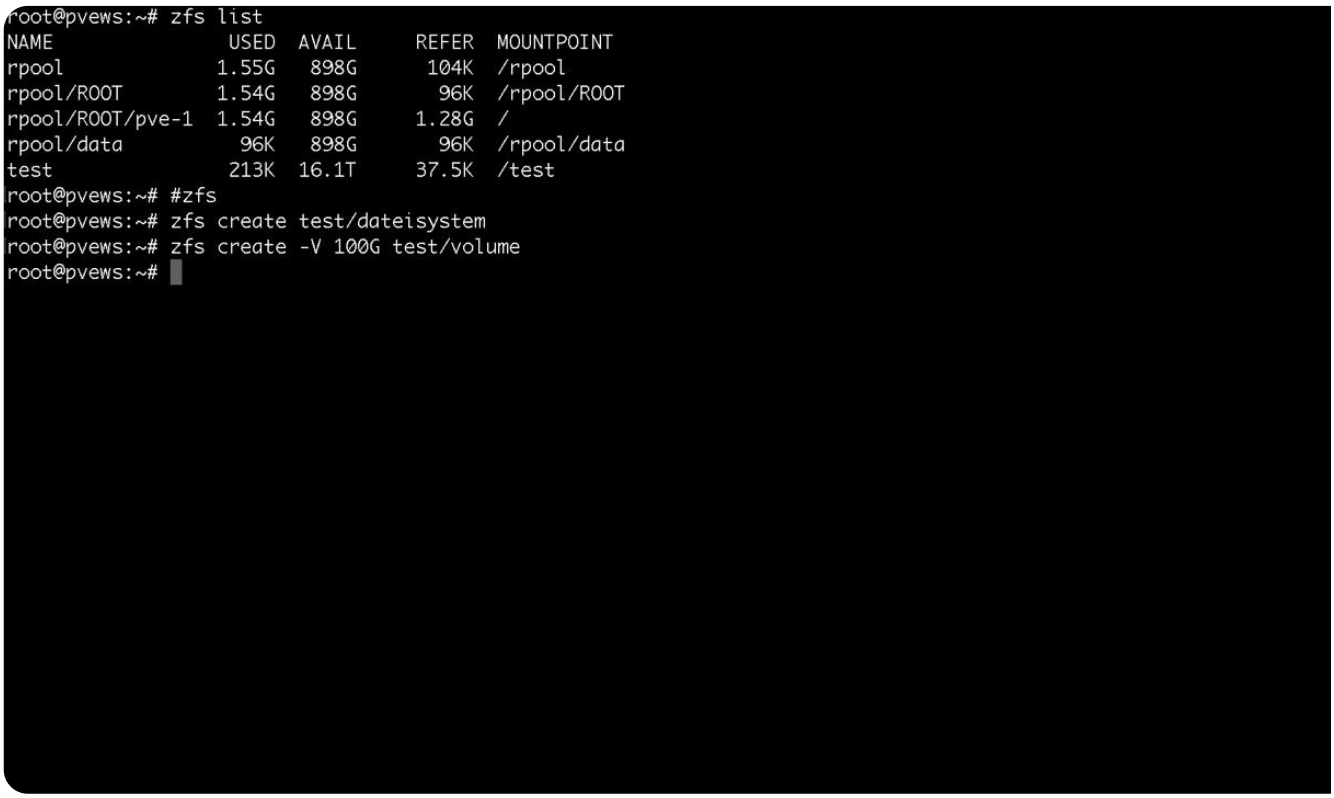
Datasets werden wie Dateisysteme gemountet und eignen sich für:

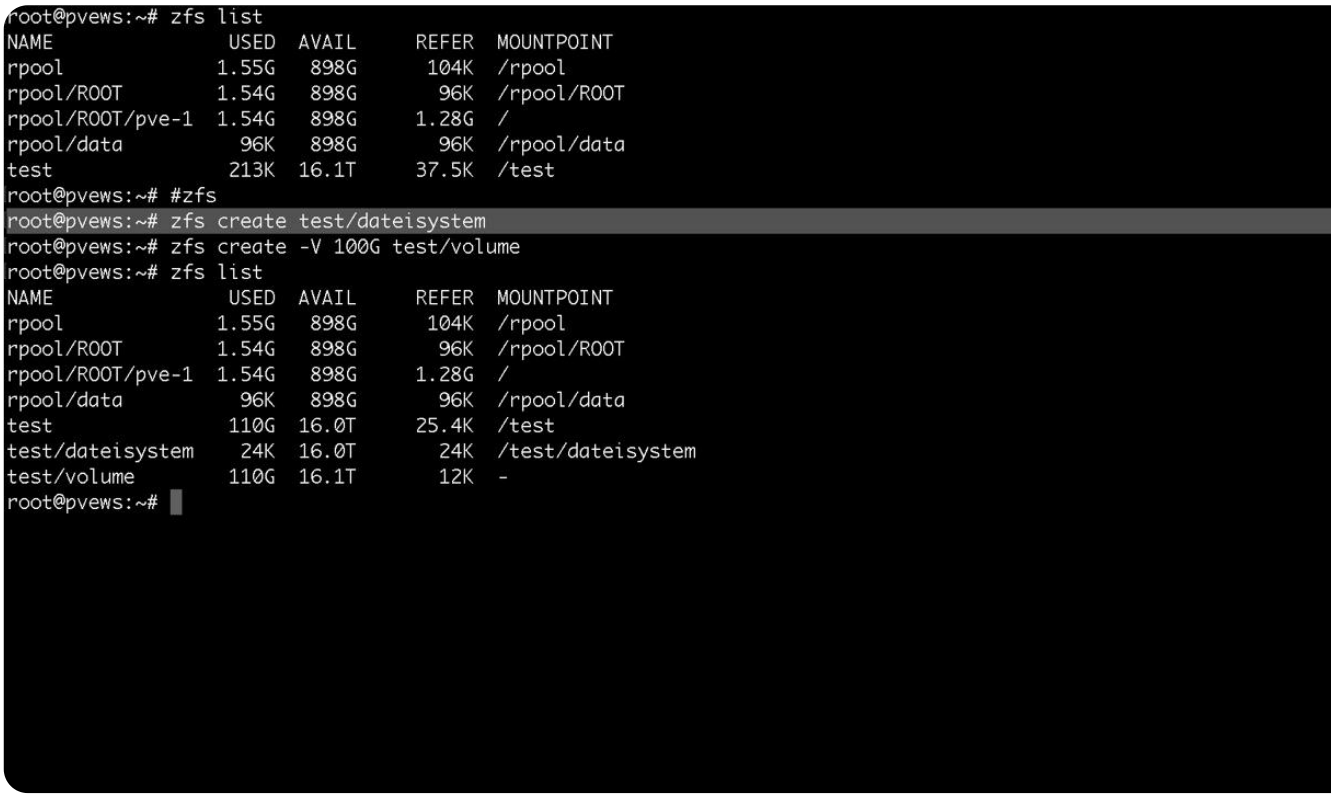
- LXC-Container
- Backupdateien
- Vorlagen
- Serverdateien
- Fileserver-Daten

ZVOLs

ZVOLs verhalten sich wie virtuelle Blockgeräte. Sie werden unter `/dev/zd...` und mit sprechenden Namen unter `/dev/zvol/poolname/...` bereitgestellt.

Sie eignen sich insbesondere für virtuelle Festplatten von VMs.





Thin- und Thick-Provisioning mit ZFS

Proxmox VE erzeugt virtuelle Festplatten auf ZFS typischerweise als ZVOLs. Thin Provisioning spart zunächst Speicherplatz, muss aber sorgfältig überwacht werden.

Beispiel: Thin Provisioning für eine VM

```
zfs create -s -b 16k -V 33554432k rpool/data/vm-301-disk-4
```

Beispiel: Dataset für LXC

```
zfs create -o acltype=posixacl -o xattr=sa -o refquota=33554432k rpool/data/subvol-106-disk-1
```

Bei LXC-Containern ist die Dimensionierung besonders wichtig, da Snapshots und Aufbewahrungszeiten den effektiv nutzbaren Platz beeinflussen.

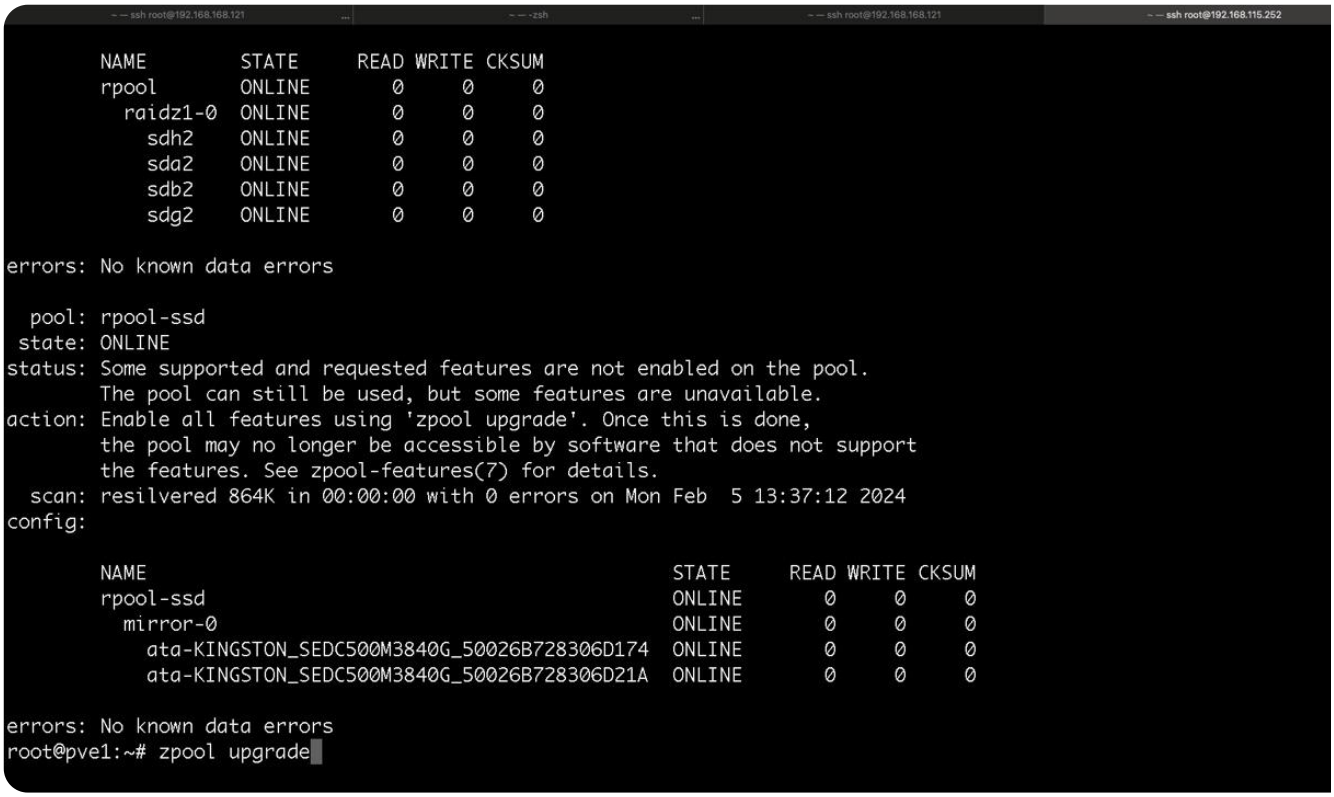
Virtuelle Diskimages wie QCOW2, VMDK oder RAW-Dateien sollten normalerweise nicht innerhalb eines ZFS-Datasets abgelegt werden, außer bei speziellen NFS- oder iSCSI-Szenarien.

ZFS Pool Features aktualisieren

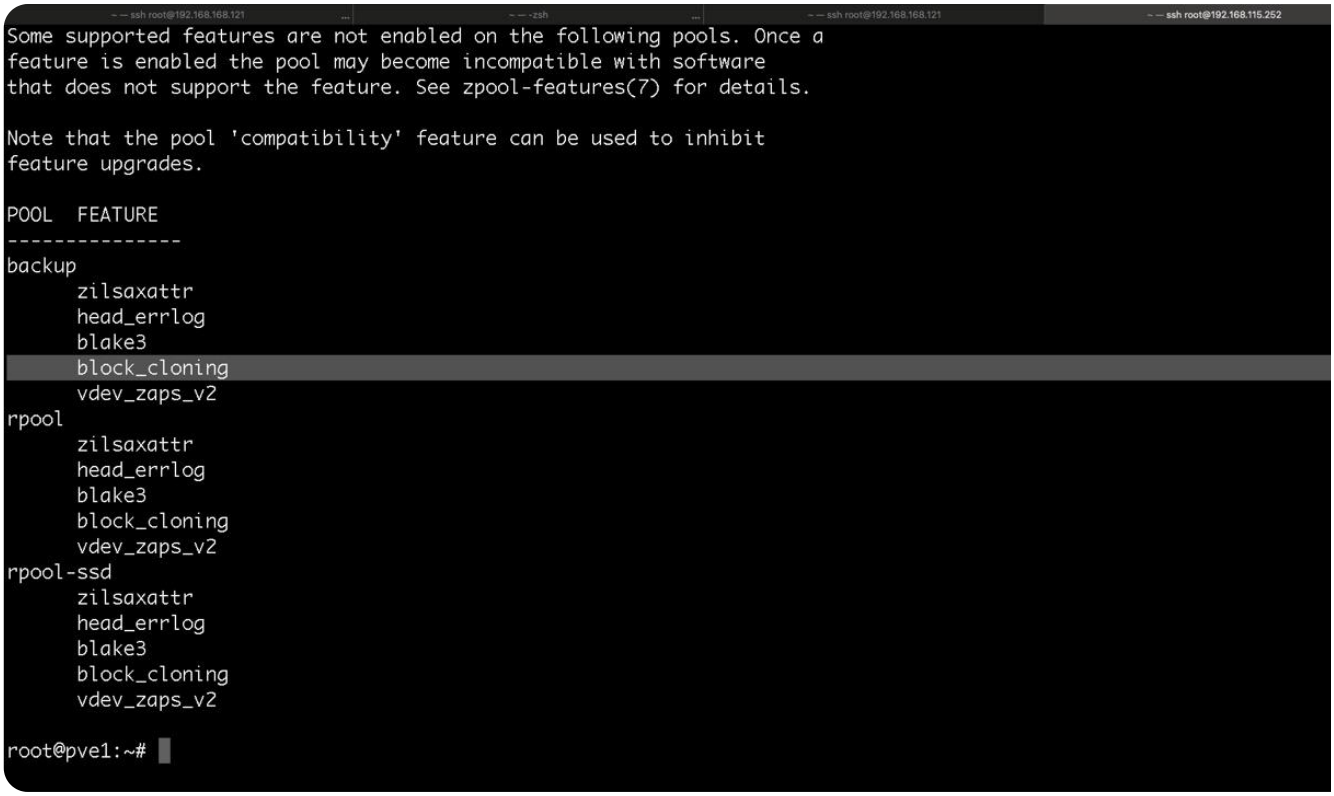
Wenn `zpool status` neue ZFS-Funktionen meldet, können diese grundsätzlich aktualisiert werden:

```
zpool upgrade -a
```

Wichtig: Vor einem Upgrade sollte geprüft werden, ob vorhandene Notfall-ISOs, Rettungssysteme und andere Hosts die neuen ZFS-Features bereits unterstützen.



```
        NAME        STATE     READ WRITE CKSUM
        rpool       ONLINE       0     0     0
          raidz1-0  ONLINE       0     0     0
            sdh2    ONLINE       0     0     0
            sda2    ONLINE       0     0     0
            sdb2    ONLINE       0     0     0
            sdg2    ONLINE       0     0     0

errors: No known data errors

  pool: rpool-ssd
 state: ONLINE
status: Some supported and requested features are not enabled on the pool.
        The pool can still be used, but some features are unavailable.
action: Enable all features using 'zpool upgrade'. Once this is done,
        the pool may no longer be accessible by software that does not support
        the features. See zpool-features(7) for details.
  scan: resilvered 864K in 00:00:00 with 0 errors on Mon Feb  5 13:37:12 2024
config:

        NAME                                            STATE     READ WRITE CKSUM
        rpool-ssd                                       ONLINE       0     0     0
          mirror-0                                      ONLINE       0     0     0
            ata-KINGSTON_SEDC500M3840G_50026B728306D174  ONLINE       0     0     0
            ata-KINGSTON_SEDC500M3840G_50026B728306D21A  ONLINE       0     0     0

errors: No known data errors
root@pve1:~# zpool upgrade
```

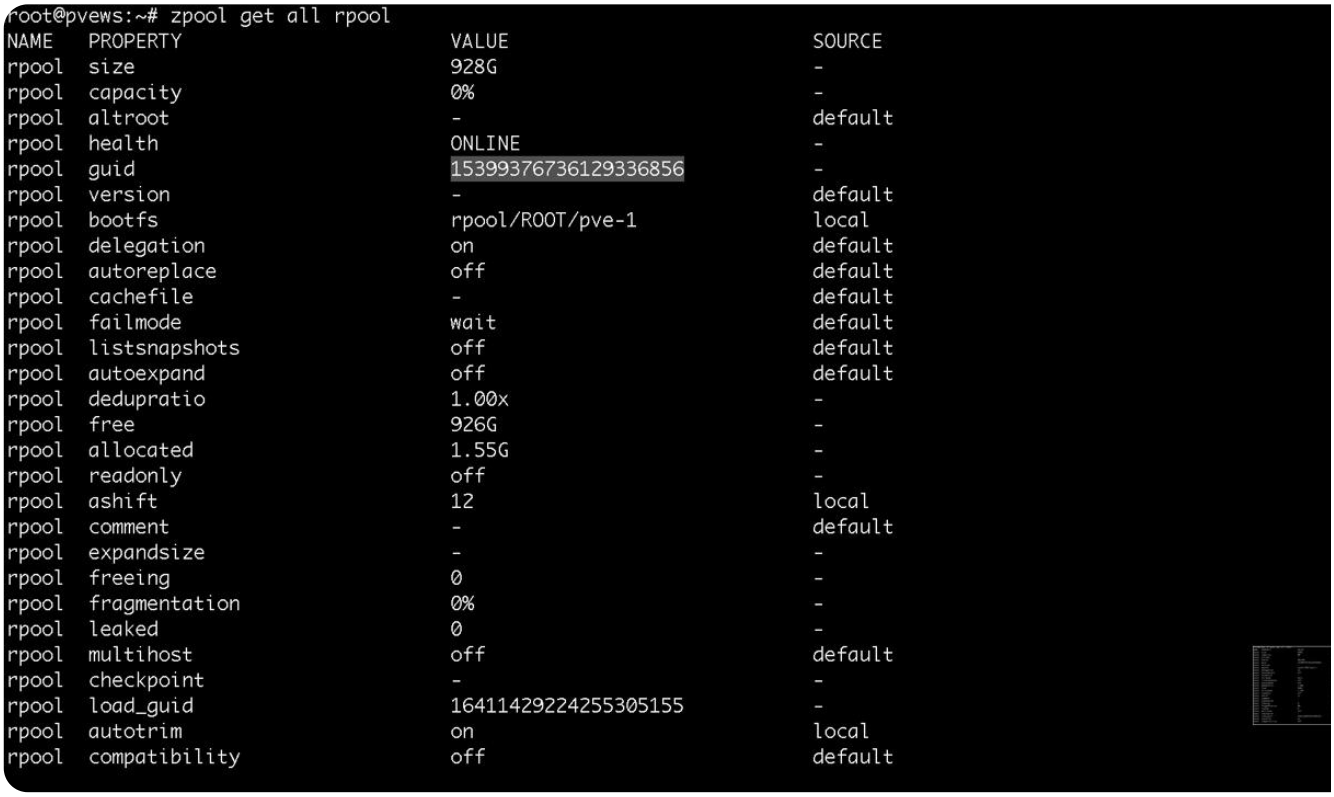


```
Some supported features are not enabled on the following pools. Once a
feature is enabled the pool may become incompatible with software
that does not support the feature. See zpool-features(7) for details.

Note that the pool 'compatibility' feature can be used to inhibit
feature upgrades.

POOL  FEATURE
---------------
backup
      zilsaxattr
      head_errlog
      blake3
      block_cloning
      vdev_zaps_v2
rpool
      zilsaxattr
      head_errlog
      blake3
      block_cloning
      vdev_zaps_v2
rpool-ssd
      zilsaxattr
      head_errlog
      blake3
      block_cloning
      vdev_zaps_v2

root@pve1:~#
```

ZFS Parameter auslesen

Für Pools, Datasets, Volumes und Snapshots können Eigenschaften mit `zfs get` ausgelesen werden.

```
zfs get all poolname
zfs get compression poolname/dataset
zfs get used,available,referenced poolname/dataset
```



Fazit: ZFS als stabile Basis für Proxmox VE

ZFS ist für viele Proxmox-Umgebungen die beste Grundlage, wenn Datensicherheit, Snapshots, Replikation und Wiederherstellbarkeit im Vordergrund stehen.

Für produktive Systeme sind besonders wichtig:

- saubere Planung des RAID-Layouts
- `ashift=12`
- passende Volblocksize
- Kompression aktivieren
- Enterprise-Datenträger verwenden
- ECC-RAM einsetzen
- Snapshots und Replikation regelmäßig prüfen

- Backups und Wiederherstellung testen

Diese Dokumentation wird fortlaufend erweitert.

ZFS Schulung oder Beratung anfragen

Sie möchten ZFS mit Proxmox VE produktiv einsetzen, bestehende Systeme prüfen oder ein sicheres Storage- und Backup-Konzept entwickeln?

Die sysops GmbH unterstützt bei Planung, Migration, Betrieb, Monitoring und Schulung rund um Proxmox VE, ZFS, Backup und Open Source Infrastruktur.

info@sysops.de
+49 (0) 6021 2125-0

Version #4
Erstellt: 2025-12-15 08:16:21 UTC von Christian Zengel (sysops GmbH)
Zuletzt aktualisiert: 2026-07-08 11:07:03 UTC von Christian Zengel (sysops GmbH)