ZFS

Basics & Advanced Themen zu ZFS

- Daily Business...
  - Festplatte im Pool ersetzen (durch eine größere)
- Grundsätzliches
  - Pools
  - Scrubbing
  - Datasets
  - Snapshots

Daily Business...

Kurze Anleitungen welche im Alltag anfallen können

Daily Business...

Festplatte im Pool ersetzen (durch eine größere)

Im Vorwege muss geprüft werden ob das Auto-Expand Feature im Ziel-Pool aktiviert ist

```
root@microserver-gen8:~# zpool get autoexpand lager
NAME   PROPERTY    VALUE   SOURCE
lager  autoexpand  off     local

root@microserver-gen8:~# zpool set autoexpand=on lager
NAME   PROPERTY    VALUE   SOURCE
lager  autoexpand  on      local
```

Die zu ersetzende HDD identifizieren mit `zpool status` und `ls -la /dev/disk/by-id`

```
root@microserver-gen8:~# zpool
status

    pool:
lager

    state:
ONLINE


    [...]


config:


        NAME                                            STATE     READ WRITE CKSUM
        lager                                           ONLINE       0     0     0
```

```
            raidz2-0                                ONLINE     0     0     0
                sde2 <-- wird ersetzt               ONLINE     0     0     0
                sdc2                                ONLINE     0     0     0
                sdd2                                ONLINE     0     0     0
                sdf2                                ONLINE     0     0     0

root@microserver-gen8:~# ls -la /dev/disk/by-id

[....]

lrwxrwxrwx 1 root root    9 Aug 30 22:40 scsi-SATA_ST8000DM004-2CX1_XYZXYZXY -> ../../sde
□□□□□□□□□□□□□□□□□□□^- Dieser Teil ist in aller Regel die Serien-Nr. der Festplatte!
                                                                    Diese Merken / notieren
```

Anschließend die identifizierte Festplatte aus dem Pool nehmen

```
sudo zpool offline lager /dev/sde2
```

Bei einem erneuten Blick auf `zpool status` wird nun der Pool als **DEGRADED** angezeigt. Nun kann der Server heruntergefahren werden und die Festplatte gegen die neue getauscht werden.

> Vor dem Einbau der neuen Festplatte die Serien-Nr. der neuen Platte notieren oder ein Foto des Labels machen!

Server wieder starten und anschließend die neue Festplatte mit einem erneuten `ls -la /dev/disk/by-id` identifizieren

```
ls -la /dev/disk/by-id

[...]

lrwxrwxrwx 1 root root    9 Aug 30 22:40 scsi-SATA_WDC_WD40EFRX-68N_WD-XYZXYZXYZXYZ ->
../../sde
```

Die neue Festplatte wurde vom System erkannt. Zur Vorbereitung der Aufnahme in den Pool muss noch die Partitionstabelle neu geschrieben werden, dazu verwenden wir das Tool `parted` und schreiben die Tabelle im GPT Format neu auf die Festplatte. Eine eventuell auftauchende Warnung welche vor Datenverlust bei Ausführen dieses Befehls warnt wird mit ***yes*** bestätigt

```
root@microserver-gen8:~# sudo parted /dev/disk/by-id/scsi-SATA_WDC_WD40EFRX-68N_WD-
XYZXYZXYZXYZ
```

```
GNU Parted 2.3
Using /dev/sde
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) mklabel GPT
(parted) q
Information: You may need to update /etc/fstab.
```

Nun kann die neue Festplatte in den Pool aufgenommen werden bzw. der Austausch der Platten angewiesen werden

```
root@microserver-gen8:~# sudo zpool replace lager /dev/sde2 /dev/disk/by-id/scsi-
SATA_WDC_WD40EFRX-68N_WD-XYZXYZXYZXYZ
```

Nach einem kurzen Moment wurde der Befehl ausgeführt und man kann den Erfolg mit `zpool status` überprüfen. Der Pool wird nun im Hintergrund resilverd und steht während dieser Zeit nur eingeschränkt zur Verfügung

```
root@microserver-gen8:~# zpool status
  pool: lager
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function, possibly in a degraded state.
action: Wait for the resilver to complete.
  scan: resilver in progress since Wed Aug 30 22:45:48 2023
        1.72T scanned at 1.01G/s, 145G issued at 85.3M/s, 14.0T total
        35.3G resilvered, 1.02% done, 1 days 23:12:44 to go
config:

        NAME                                              STATE     READ WRITE CKSUM
        lager                                             DEGRADED     0     0     0
          raidz2-0                                        DEGRADED     0     0     0
            replacing-0                                   DEGRADED     0     0     0
              sde2                                        OFFLINE      0     0     0
              scsi-SATA_WDC_WD40EFRX-68N_WD-XYZXYZXYZXYZ  ONLINE       0     0     0
(resilvering)
            sdc2                                          ONLINE       0     0     0
            sdd2                                          ONLINE       0     0     0
            sdf2                                          ONLINE       0     0     0

errors: No known data errors
```

> Dieser Vorgang kann je nach Größe des Pools mehrer Stunden bis Tage dauern! Während dieser Zeit sollte man den Pool nicht verwenden und den Vorgang in Ruhe arbeiten lassen

Nachdem das Resilvern abgeschlossen ist steht der dazugewonnene Speicherplatz eventuell noch nicht sofort zur Verfügung, dies kann mit `zpool list` geprüft werden

```
root@microserver-gen8:~# zpool list
NAME    SIZE  ALLOC   FREE  CKPOINT  EXPANDSZ   FRAG    CAP  DEDUP    HEALTH  ALTROOT
lager  14.5T  14.0T   598G        -      6,8T    15%    95%  1.00x    ONLINE  -
```

Hier sieht mann dass die Größe sich noch nicht angepasst hat, allerdings stehen dem Pool 14,5TB an Erweiterungsgröße zur Verfügung. Im folgenden Übernehmen wir diesen verfügbaren Speicherplatz in unseren Pool

```
root@microserver-gen8:~# zpool online -e lager scsi-SATA_WDC_WD40EFRX-68N_WD-XYZXYZXYZXYZ

NAME    SIZE  ALLOC   FREE  CKPOINT  EXPANDSZ   FRAG    CAP  DEDUP    HEALTH  ALTROOT
lager  21,3T  14.0T   7,3T        -         -    15%    68%  1.00x    ONLINE  -
```

Grundsätzliches

Kurze Übersichten über die verfügbaren Befehle welche zur Verwaltung von ZFS Arrays benötigt werden

Pools

Pools auflisten

```
sudo zpool list
```

Neuen ZFS Pool auf einer einzelnen Festplatte erstellen:

```
sudo zpool create vol0 /dev/sd[x]
```

> Der neu erstellte Pool wird automatisch unter `/[pool name]` gemounted

Alle Datasets in einem Pool löschen

```
sudo zfs destroy -r [pool name]
```

Pool löschen

```
sudo zpool destroy [pool name]
```

Festplattenstatus einsehen

```
sudo zpool status
```

Poolbalance einsehen

```
zpool list -v
```

Wird einem bestehendem Pool eine weitere Festplatte hinzugefügt, kann es passieren dass die Daten über alle Festplatten ungleichmäßig verteilt werden da ZFS kein automatisches Balancing betreibt.

Der einfachste Weg den Pool zu balancen wäre ein neues Dataset zu erstellen und alle Daten auf dieses zu verschieben und danach wieder zurück.

Dies sollte jedoch nicht mit `mv` getan werden, da hierbei erst alle Daten kopiert und danach gelöscht werden - Der Speicherplatz könnte für dieses Verfahren nicht ausreichen! Besser ist es hierfür `rsync` zu verwenden da hierbei eine Datei nach der anderen verschoben wird und keine doppelte Belegung des Speicherplatzes entsteht

Scrubbing

Pool scrubben (Dateien auf Integrität prüfen)

```
sudo zpool scrub [pool name]
```

> Den Fortschritt des Scrubbings kann man mit `sudo zpool status` einsehen

Datasets

Dataset erstellen

```
sudo zfs create [pool name]/[dataset name]
```

> ZFS wird das neue Dataset automatisch unter `/path/to/pool/[dataset name]` mounten

Kaskadiertes Dataset erstellen

```
sudo zfs create [pool name]/[dataset name]/[descendent filesystem]
```

Alle Datasets und Pools auflisten

```
sudo zfs list
```

Dataset löschen

```
sudo zfs destroy [pool name]/[dataset name]
```

> Ein Dataset kann nicht gelöscht werden solange noch Snapshots oder Klone dieses Datasets bestehen

Dataset Recordsize setzen

```
sudo zfs set recordsize=[size] pool/dataset/name
```

> Die Recordsize muss mit der richtigen Einheit angegeben werden, bspw. 16k, 128k, 1M, 1G usw.

Dataset Recordsize abfragen

```
sudo zfs get recordsize pool/dataset/name
```

Snapshots

Snapshot eines Datasets anlegen

```
sudo zfs snapshot [pool]/[dataset name]@[snapshot name]
```

Snapshots auflisten

```
sudo zfs list -t snapshot
```

Snapshot umbenennen

```
sudo zfs rename [pool]/[dataset]@[old name] [new name]
```

Zu einem Snapshot zurückkehren / wiederherstellen

```
sudo zfs rollback -r [pool]/[dataset]@[snapshot name]
```

> Dadurch werden alle späteren Snapshots gelöscht!

> Das Dateisystem welches durch den Snapshot betroffen ist wird für den Rollback automatisch un- und wieder gemounted. Sollte dies Fehlschlagen kann durch `-f` das Unmounten forciert werden

Snapshot löschen

```
sudo zfs destroy [pool]/[dataset]@[snapshot name]
```