

Proxmox Debian als ZFS-Over-ISCSI Server verwenden

Systemannahme: Debian9 mit ZFS / pve-manager/5.3-11/d4907f84 (running kernel: 4.15.18-11-pve) mit ZFS

Seit kurzem funktioniert nun ZFS over ISCSI auch auf Debian Proxmox. Für die Installation ist lediglich ein Paket am Server erforderlich.

```
apt install targetcli-fb -y
```

Vorbereiten der Proxmox Clientnodes

Damit die Nodes auf das ZFS Dataset dynamisch drauf zugreifen können, müssen diese autorisiert werden. Hierfür werden ACL's in der Targetcli und SSH-Keys verwendet.

Erstellen der speziellen SSH-Key's

Die IP des Targets hier im Keynamen ist wichtig, dieser wird von PVE als Default verwendet. IPV6 ist gleichermaßen möglich.

```
mkdir /etc/pve/priv/zfs
ssh-keygen -f /etc/pve/priv/zfs/10.1.1.100_id_rsa
ssh-copy-id -i /etc/pve/priv/zfs/10.1.1.100_id_rsa.pub root@10.1.1.100
```

Nun muss man von jedem Server einmal eine Verbindung aufbauen. Ist alles ok geht die Verbindung ohne Passwort und Fingerprintanfrage durch.

```
ssh -i /etc/pve/priv/zfs/10.1.1.100_id_rsa root@10.1.1.100
```

Auslesen des ISCSI Initiatorsname

Dieser Name wird für die ACL's verwendet. Hierfür geht man auf jede Node die Zugriff haben soll und liest diesen aus:

```
cat /etc/iscsi/initiatorname.iscsi
```

Hier sind die 3 Nodes als Beispiel:

```
iqn.2019-03.org.debian:01:3eba667c8c
iqn.2019-03.org.debian:02:a0bf71a32216
iqn.2019-03.org.debian:03:5eba5ac67c8
```

Erstellen des Targets

Für die Freigabe legt man zuerst ein ganz normales ZFS Dataset auf einem Pool seiner Wahl an. Darunter werden dann die Zvols ganz normal von Proxmox abgelegt.

```
zfs create testpool/iscsi
```

Hat man das Dataset geht es an unsere Cli. Um nun mit der Konfiguration zu starten, geben wir den Befehl `targetcli` im Terminal ein. Mittels `ls` sieht man die Baumstruktur. Diese ist auch als solche zu behandeln. Man kann mit den default Linuxbefehlen darin navigieren. `help` zeigt die Hilfe, mittels `cd` kann man mit Pfeiltasten navigieren, und mit `saveconfig` wird gespeichert.

Das ganze sieht nun so aus:

```
/> ls
o- /
.....
..... [ ... ]
  o- backstores
.....
..... [ ... ]
    | o- block
.....
.. [Storage Objects: 0]
  | o- fileio
.....
. [Storage Objects: 0]
  | o- pscsi
.....
.. [Storage Objects: 0]
  | o- ramdisk
.....
[Storage Objects: 0]
  o- iscsi
.....
..... [Targets: 0]
    o- loopback
.....
..... [Targets: 0]
      o- sbp
.....
..... [Targets: 0]
        o- vhost
.....
..... [Targets: 0]
```

Nun erstellen wir unser Target. Hierfür wechseln wir nach `iscsi` und sagen `create`. Wenn wir nichts angeben wird der Name automatisch gewählt. Danach wechseln wir in das gerade erstellte neue Target, bis zu den ACL's. In den ACL's kommen nun alle zugreifenden Hosts mit ihren Initiatornamen

hinein. Erstellt wird wieder mit `create <initiatorname>`. Nun wechseln wir eine Ebene tiefer auf `tpgl` und dann auf `portals`. Dort werden nun die IP-Adressen der Proxmoxserver eingetragen die zugreifen dürfen. IPV4 oder IPV6 ist hier möglich. Nun einfach mit `exit` verlassen. Gespeichert wird automatisch. Können wir auch noch vorher mit `saveconfig` bestätigen.

Die fertige Config sieht nun so aus:

```

/> ls
o- /
.....
..... [ ... ]
  o- backstores
.....
..... [ ... ]
    | o- block
.....
..... [Storage Objects: 0]
    | o- fileio
.....
..... [Storage Objects: 0]
    | o- pscsi
.....
..... [Storage Objects: 0]
    | o- ramdisk
.....
..... [Storage Objects: 0]
  o- iscsi
.....
..... [Targets: 1]
    | o- iqn.2019-03.org.linux-iscsi.backup.tux.at:sn.baef28cdfaff
..... [TPGs: 1]
    |   o- tpg1
.....
..... [no-gen-acls, no-auth]
    |     o- acls
.....
..... [ACLs: 3]
    |       | o- iqn.2019-03.org.debian:01:3eba667c8c
..... [Mapped
LUNs: 0]
    |       | o- iqn.2019-03.org.debian:02:a0bf71a32216
..... [Mapped
LUNs: 0]
    |       | o- iqn.2019-03.org.debian:03:5eba5ac67c8
..... [Mapped
LUNs: 0]
    |     o- luns
.....
..... [LUNs: 0]
    |     o- portals
.....

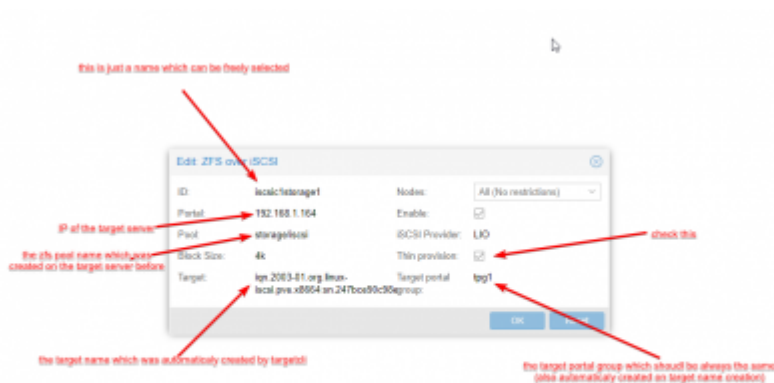
```

```

..... [Portals: 3]
|      o- [2001:470:1e0f:567:4305:eaff:ff9e:43ac]:3260
..... [OK]
|      o- [2001:470:1e0f:567:8c4:eafb:ae33:a7bf]:3260
..... [OK]
|      o- [2001:470:1e0f:567:ef4:6afb:fefe:a874]:3260
..... [OK]
o- loopback
.....
..... [Targets: 0]
o- sbp
.....
..... [Targets: 0]
o- vhost
.....
..... [Targets: 0]

```

Anbinden des Proxmoxclusters



Dies erfolgt ganz einfach über die GUI.

Auf der CMD kann die Config so aussehen:

```

zfs: iscsi-zfs
    blocksize 4k
    iscsiprovider LIO
    pool testpool/iscsi
    portal backup.tux.at
    target iqn.2019-03.org.linux-iscsi.backup.tux.at:sn.baef28cdfaff
    content images
    lio_tpg tpg1
    nowritecache 1
    sparse 1

```

Ab nun kann man ZFS für Proxmox HA verwenden.

Bugs

Derzeit 03.2019, kann man noch keine HDD's auf das Target verschieben wenn die KVM nicht läuft. Hier ist der [Bugreport](#) dazu.

Links

- Diskussion im Thread des [Proxmoxforum](#)

From:
<https://deepdoc.at/dokuwiki/> - DEEPDOC.AT - enjoy your brain

Permanent link:
https://deepdoc.at/dokuwiki/doku.php?id=virtualisierung:proxmox_kvm_und_lxc:proxmox_debian_als_zfs-over-iscsi_server_verwenden&rev=1616495543

Last update: 2021/03/23 11:32

