Pve-Zsync über zwei Standorte - Resync nach Standortausfall

Du möchtest dich gerne für unsere Hilfe erkenntlich zeigen . Gerne. Wir bedanken uns bei dir für deine Spende!

Spenden

Zum frei verfügbaren Apt-Repository

GITLAB:

Pve-Zsync ein tolles Werkzeug um zwei Standorte miteinander zu synchronisieren, auch wenn keine Clusterverbindung besteht. Meist sind hier für die Latenzen viel zu schlecht. Als Annahme haben wir **Standort A** (Hauptstandort) und **Standort B** (Backup) wo der zweite Proxmoxserver steht, der die Daten von Standort A abholt.

- Erster Server am Standort A hat die IP 172.15.3.1
- Zweiter Server am Standort B hat die IP 172.16.3.1
- VMID 114 mit zwei Festplatten

Einrichtung am Standort B

Am zweiten Standort wird der Sync eingerichtet. Sprich der Backupserver holt die Daten von Standort A ab. Dies geschiet mit einem Befehl. Zuerst öffen wir aber einen Screen, falls die Verbindun abbrechen sollte.

```
screen -S admin_peter
```

```
pve-zsync create --source 172.15.3.1:114 --dest mein-zfs-pool/ein-dataset --
verbose --maxsnap 5 --name buchhaltung
```

Mit der Option `–verbose` sagen wir dem Programm das wir den Fortschritt aktiv mit schauen möchten. Wollen wir das nicht, lassen wir die Option weg, und verwenden stattdessen `–skip`. Damit wird der Job erstellt und startet erst nach der ersten Synczeit, sprich 15 Minuten was der Default ist. Mit `pve-zsync list` sieht man die Jobs

Nachdem der Sync das erste mal angelaufen ist, kann man die Zeiten bequem im Cron ändern.

```
nano /etc/cron.d/pve-zsync
```

Dort kann man diesen auch für die Deaktivierung auskommentieren. Durch einen Bug ist die Deaktivierung mit `–disable` der Zeit nicht möglich. Wobei den Cron zu editieren sich auch als einfacher darstellt.

Wenn der Sync dann gelaufen ist, könnte das ganze so aussehen. `zfs list -t snapshot` Für den Test wurde der Sync auf 2 Minuten gesetzt.

```
...
mein-zfs-pool/ein-dataset/vm-114-disk-0@rep_buchhaltung_2019-07-07_22:12:01
1,16M      -  4,92G  -
mein-zfs-pool/ein-dataset/vm-114-disk-0@rep_buchhaltung_2019-07-07_22:14:01
692K      -  4,92G  -
mein-zfs-pool/ein-dataset/vm-114-disk-0@rep_buchhaltung_2019-07-07_22:16:01
482K      -  4,92G  -
mein-zfs-pool/ein-dataset/vm-114-disk-0@rep_buchhaltung_2019-07-07_22:45:00
6,72M      -  4,94G  -
mein-zfs-pool/ein-dataset/vm-114-disk-0@rep_buchhaltung_2019-07-07_22:48:01
0B      -  4,94G  -
mein-zfs-pool/ein-dataset/vm-114-disk-1@rep_buchhaltung_2019-07-07_22:12:01
0B      -  81,4K  -
mein-zfs-pool/ein-dataset/vm-114-disk-1@rep_buchhaltung_2019-07-07_22:14:01
0B      -  81,4K  -
mein-zfs-pool/ein-dataset/vm-114-disk-1@rep_buchhaltung_2019-07-07_22:16:01
0B      -  81,4K  -
mein-zfs-pool/ein-dataset/vm-114-disk-1@rep_buchhaltung_2019-07-07_22:45:00
0B      -  81,4K  -
mein-zfs-pool/ein-dataset/vm-114-disk-1@rep_buchhaltung_2019-07-07_22:48:01
0B      -  81,4K  -
...
```

Annahme Ausfall am Standort A

Durch z.B. eine Brand, Blitzschlag, Spitzenspannung, Sabotage, was auch immer. Die Daten sind nicht mehr zu Retten, also wird die VM am zweiten Standort scharf gemacht und hoch gefahren.

VM am Standort B hochfahren

Als erstes wird der Syncjob deaktiviert. Dies tut man in dem man die Zeile im Cron auskommentiert.

```
nano /etc/cron.d/pve-zsync

#*/2 * * * * root pve-zsync sync --source 172.15.3.1:114 --dest mein-zfs-
pool/ein-dataset --name buchhaltung --maxsnap 5 --method ssh --source-user
root --dest-user root
```

Als nächstes kopiert man die VM-Config seiner Wahl an die richtige Stelle.

```
cp /var/lib/pve-zsync/ein-
dataset/114.conf.qemu.rep_buchhaltung_2019-07-07_22:08:01 /etc/pve/qemu-
server/114.conf
```

Möglicherweise muss man diese je nach Hardware noch anpassen. z.B. CPU Kerne oder RAM. Hier ist auch mögliche Lizenzthemen in Windowssystemen zu achten. Optimal ist wenn beide Server gleich dimensioniert sind. Hat man das erledigt, kann die VM schon starten. Selbstverständlich muss man den IPkreis dem, den Standort B anpassen.

Standort A wieder ok, Resync der Vm's

Nun nach einigen Wochen ist der Standort A wieder ok und man möchte mit seinen Daten wieder umziehen. Die Festplatten konnten noch gerettet werden und laufen schon wieder im neuen Server. Nun synchronisieren wir die Daten zurück. Natürlich nur die in der Zwischenzeit passierten Änderungen.

Hierfür machen wir einen manuellen Sync vom **Standort A** aus und holen uns die Daten **Standort B**. Das ganze sieht dann so aus:

```
pve-zsync sync --source 172.16.3.1:114 --dest mein-zfs-pool/ein-dataset --
verbose --maxsnap 5 --name buchhaltung
send from @rep_buchhaltung_2019-07-07_22:16:01 to mein-zfs-pool/ein-
dataset/vm-114-disk-0@rep_buchhaltung_2019-07-07_22:45:00 estimated size is
117M
total estimated size is 117M
TIME        SENT   SNAPSHOT
22:45:03   88,8M   mein-zfs-pool/ein-dataset/vm-114-
disk-0@rep_buchhaltung_2019-07-07_22:45:00
send from @rep_buchhaltung_2019-07-07_22:16:01 to mein-zfs-pool/ein-
dataset/vm-114-disk-1@rep_buchhaltung_2019-07-07_22:45:00 estimated size is
624B
total estimated size is 624B
TIME        SENT   SNAPSHOT
```

Danach starten wir die VM einfach wieder, die Config liegt ja noch im richtigen Pfad und hat auch noch die ursprüngelichen Werte. Sofern sich natürlich am neuen Server einiges geändert hat, müssen die Werte für CPU, Ram usw. möglicherweise auch wieder angepasst werden.

Syncjob am Standort B wieder reaktivieren

Nun läuft wieder alles wie gewollt. Es ist nun an der Zeit den Syncjob am **Standort B** wieder zu reaktivieren. Dies erledigen wir wieder ganz einfach in dem wir die betreffende Zeile im Cronjob einkommentieren.

```
nano /etc/cron.d/pve-zsync

*/2 * * * * root pve-zsync sync --source 172.15.3.1:114 --dest mein-zfs-
pool/ein-dataset --name buchhaltung --maxsnap 5 --method ssh --source-user
root --dest-user root
```

Und läufts wieder in die andere Richtung.

Einfacher VM-Test am Standort B

Oft kommt es vor das man einfach nur mal sein Backup am zweiten Standort testen möchte. Ob es denn überhaupt funktioniert, ob es hochfährt, wie es performt… Dies kann man einfach erledigen in dem man sich aus seinem ausgesuchten Snapshot eine VM erstellt. Bevor man klont sollte man den Syncjob deaktivieren. Warum? Hier ein Auszug aus der Manmpage:

A clone is a writable volume or file system whose initial contents are the same as another dataset. As with snapshots, creating a clone is nearly instantaneous, and initially consumes no additional space. Clones can only be created from a snapshot. When a snapshot is cloned, it creates an implicit dependency between the parent and child. Even though the clone is created somewhere else in the dataset hierarchy, the original snapshot cannot be destroyed as long as a clone exists. The origin property exposes this dependency, and the destroy command lists any such dependencies, if they exist. The clone parent-child dependency relationship can be reversed by using the promote subcommand. This causes the „origin“ file system to become a clone of the specified file system, which makes it possible to destroy the file system that the clone was created from.

```
zfs clone mein-zfs-pool/ein-dataset/vm-114-
disk-0@rep_buchhaltung_2019-07-07_22:48:01 mein-zfs-pool/ein-dataset/vm-444-
disk-0
zfs clone mein-zfs-pool/ein-dataset/vm-114-
disk-1@rep_buchhaltung_2019-07-07_22:48:01 mein-zfs-pool/ein-dataset/vm-444-
disk-1
cp /var/lib/pve-zsync/ein-
dataset/114.conf.qemu.rep_buchhaltung_2019-07-07_22:08:01 /etc/pve/qemu-
server/444.conf
```

Ist man mit dem Test fertig, kann man einfach die VM löschen.

Möchte man einen vollen Klon haben und nicht mit den anderen Snapshots verknüpft sein, kopiert man zuerst die VM-Config, aber mit der gleichen ID. Danach kann man die VM vom Proxmox Webinterface klonen (Vollklone).

Links

- Linux ZFS
- pve-zsync (Proxmoxdoku)

From:
https://v-source.org/dokuwiki/ - **DEEPDOC.AT - enjoy your brain**

Permanent link:
https://v-source.org/dokuwiki/doku.php?id=virtualisierung:proxmox_kvm_und_lxc:pve-zsync_ueber_zwei_standorte_-_resync_nach_standortausfall

Last update: **2025/05/18 09:01**