Truenas-Installation mit Partitionierung in boot und data

Am Besten nur die Festplatten anschließen, die als boot verwendet werden sollen.

Bootfähigen USB-Stick mit Truenas Scale-Image/Iso erstellen.

PC neu starten und vom USB-Stick booten.

Im ersten Auswahlmenü 'Truenas Installer' auswählen

Im nächsten Menü mit 4 Unterpunkten 'shell' auswählen

1. Truenas-Installationsskript anpassen. Folgende Befehle in der shell:

```
# Pfeiltasten aktivieren
bash

# truenas-install Skript finden
find / -name truenas-install

# vi editor nutzen :(
vi /usr/sbin/truenas-install

# Folgende Code-Zeile finden (ca. line 386).
# In vi mit ':.=" Current Line unten anzeigen
...
create_partition()
{
...
  # Create boot pool
  if ! sgdisk -n3:0:0 -t3:BF01 /dev/${_disk}; then
    return 1
  fi
```

```
...

#Partitionierung in 64GiB disk.
# Nachfolgende Befehle fürs editieren nutzen:
# Vi-Tastenbefehle: 'x' löschen an cursor-Stelle
#                   'i' edit mode
#                   'u' undo, falls irgendwas verkackt mit der Zeile
#                   'ESC' exit edit mode
#                   ':wq:' eingeben zum Änderung zu speichern
...
  # Create boot pool
  if ! sgdisk -n3:0:+64GiB -t3:BF01 /dev/${_disk}; then
    return 1
  fi
...

# Skript ausführen
/usr/sbin/truenas-install
```

2. Install/Upgrade auswählen und Festplatten für Installation auswählen (Ich hatte 2 Festplatten fürs mirroring). Installer folgen.

3. Restlichen Speicher der Festplatte partitionieren und zu einem Pool hinzufügen. Nach der Installation von Truenas muss in der GUI SSH unter Einstellungen -> Services aktiviert werden. Ich habe den Root-Account genutzt für die nachfolgenden Befehle (ggfls. Password unter Credentials -> Local users -> root setzen).

```
# Boot-Pool ausfindig machen
zpool status boot-pool
# +
fdisk -l

# Bei den boot-Laufwerken sollten 3 Unterpartitionen (4 wenn Swap eingerichtet wurde)
zu sehen sein z.b. /dev/sda1, /dev/sda2,  /dev/sda3 ...). Bei mir waren die beiden
Festplatten als sda und sdb bezeichnet.

# no swap
sgdisk -n4:0:0 -t4:BF01 /dev/sda
sgdisk -n4:0:0 -t4:BF01 /dev/sdb
# swap
sgdisk -n5:0:0 -t5:BF01 /dev/sdX
```

```
sgdisk -n5:0:0 -t5:BF01 /dev/sdY

# update the linux kernel table
partprobe

# UUID der neuen Partition herausfinden
fdisk -lx /dev/sda
fdisk -lx /dev/sdb

# Beispiel der Konsolenausgabe
Device         Start       End   Sectors Type-UUID
UUID                                  Name Attrs
/dev/sdb1       4096      6143      2048 21686148-6449-6E6F-744E-656564454649
3EDDE409-829A-4CFD-AD28-E76FADAD9D40      LegacyBIOSBootable
/dev/sdb2       6144   1054719   1048576 C12A7328-F81F-11D2-BA4B-00A0C93EC93B
EEDB9E96-BF0B-4CF3-9D86-7032C6BC8C09
/dev/sdb3    1054720 134217728 133163009 6A898CC3-1DD2-11B2-99A6-080020736631
14F8F26A-EA8E-484A-9AEF-C82DAEC27D58
/dev/sdb4  134219776 976773134 842553359 6A898CC3-1DD2-11B2-99A6-080020736631
AE77275A-6B71-4FC1-917B-B3C6C995AC8F

# zpool erstellen
# [uuid_from fdisk -lx disk1] von sda4 bzw. sdb4 = Spalte UUID = AE77275A-6B71-4FC1-
917B-B3C6C995AC8F
# Achtung: UUID ist case-sensitive. Ich musste alle Buchstaben klein schreiben, weil
die Partitionsordner im Verzeichnis '/dev/dis/by-partuuid/' ebenfalls
kleingeschrieben waren.
# Kann mit 'cd /dev/dis/by-partuuid/' und 'ls' gecheckt werden
zpool create -f ssd-storage mirror /dev/disk/by-partuuid/[uuid_from fdisk -lx disk1]
/dev/disk/by-partuuid/[uuid_from fdisk -lx disk2]

# zpool exportieren
zpool export ssd-storage
```

4. ssd-storage in der webGUI importieren. Dazu unter 'Storage -> Import-Pool' oben anklicken und ssd-storage sollte da zu sehen sein!

Revision #2
Created 2023-07-27 20:43:37 UTC by Dung
Updated 2023-07-28 17:23:58 UTC by Dung