

Proxmox - LVM in VM vergrößern

- PV Group / VGroup und Logical Volume inklusive Dateisystem Vergrößern
- Proxmox Host LVM Vergrößern

PV Group / VGroup und Logical Volume inklusive Dateisystem Vergrößern

Laufwerk der VM vergrößern

Im Proxmox einloggen, die Virtuelle Maschine auswählen und auf Hardware gehen. Dort die Festplatte auswählen und auf resize disk klicken.

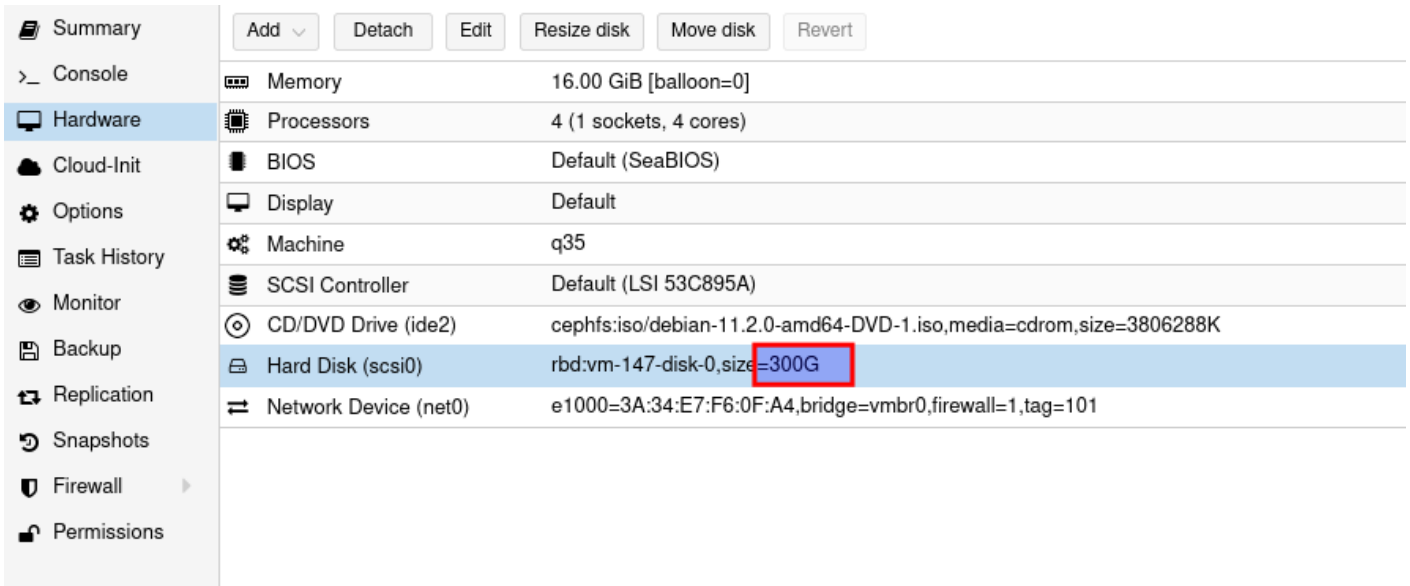
Virtual Machine 147 (SimbaCloud) on node 'vserv0001' 2

Component	Configuration
Memory	16.00 GiB [balloon=0]
Processors	4 (1 sockets, 4 cores)
BIOS	Default (SeaBIOS)
Display	Default
Machine	q35
SCSI Controller	Default (LSI 53C895A)
CD/DVD Drive (ide2)	cephfs:iso/debian-11.2.0-amd64-DVD-1.iso,media=cdrom,size=3806288K
Hard Disk (scsi0)	rbd:vm-147-disk-0,size=201G 1
Network Device (net0)	e1000=3A:34:E7:F6:0F:A4,bridge=vibr0,firewall=1,tag=101

Nun angeben um wie viel Gigabyte das Volume größer werden soll. In unserem Beispiel 99 GB
Somit hat das Laufwerk dann, 300 GB gesamt $201 \text{ GB} + 99 \text{ GB} = 300 \text{ GB}$.
Nachdem der gewünschte Wert angegeben ist auf resize disk klicken.



Nun hat die Disk eine Größe von 300 GB



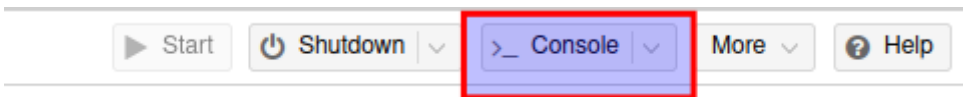
In der VM Gruppen vergrößern

Partition vergrößern mit parted

Anmelden per noNC Konsole oder über ssh wenn möglich.

Zur Erinnerung noVNC Console:

VM anklicken und im Menü oben auf Console klicken



Nun im Terminal erstmal die Partitionen und die Größen rausbekommen mit dem Befehl

```
lsblk
```

Ausgabe:

```

NAME                                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sr0                                  11:0    1   3,6G  0 rom
sda                                  8:0     0  300G  0 disk
├─sda2                               8:2     0  99,5G  0 part
│  ├─nvistaVG-swapVolume             254:1   0   8,3G  0 lvm  [SWAP]
│  └─nvistaVG-ndcVolume              254:0   0  91,3G  0 lvm  /
└─sda1                               8:1     0   476M  0 part /boot
root@ndc:~# _

```

Wie in diesem Beispiel zu sehen, liegen die LVs auf dem Laufwerk /dev/sda2.

Die Festplatte sda hat 300 GB erreicht.

Nun müssen wir die Partiton auch auf 300 GB bringen.

Das machen wir mit dem Programm parted.

Sollte parted nicht installiert sein, dieses nachinstallieren.

```

apt-get install parted      [On Debian/Ubuntu systems]
yum install parted          [On RHEL/CentOS and Fedora]
dnf install parted         [On Fedora 22+ versions]

```

Nun parted starten mit dem Befehl

```
parted
```

Nun dort den Befehl eingeben:

```
select /dev/sda
```

Ausgabe:

```

(parted) select /dev/sda
Using /dev/sda
(parted)

```

Nun die Einheiten vom MB auf Sektoren umstellen damit wir den startsektor bekommen. Dazu den Befehl

```
unit s
```

verwenden.

Nun lassen wir uns die Partitionen inklusive freien Bereich durch den Paramter free anzeigen, mit dem Befehl:

```
print free
```

Ausgabe:

```
(parted) print free
Model: QEMU QEMU HARDDISK (scsi)
Disk /dev/sda: 629145600s
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start      End          Size         Type         File system  Flags
  1      63s       2047s       1985s        primary      Free Space   boot
  2     2048s    976895s    974848s      primary      ext2         lvm
  3    976896s 209713151s 208736256s   primary      Free Space

(parted)
```

Jetzt sehen wir die Infos. Uns interessiert das ende des Laufwerks.

Also Sektor : 629145599s

Diesen müssen wir später angeben.

Nun die Partiton vergrößern mit angabe der Partion, hier Partition 2, da liegt unser lvm drauf.

Befehl:

```
resizepart 2
```

Nun wird nach dem Ende gefragt. Vorgeschlagen wird uns das aktuelle ende der Partition.

Wir wollen die Partition ja auf das ende des Datenträgers legen.

Somit die Volle Kapazität des Datenträgers.

```
(parted) resizepart 2
End? [209713151s]? _
```

Also geben wir als Endsektor 629145599s

```
(parted) resizepart 2
End? [209713151s]? 629145599s
(parted) _
```

Nun nochmals ausgeben mit dem

Befehl:

```
print free
```

nun sieht man die 2 Partition geht bis ans ende. Voilla

```
(parted) print free
Model: QEMU QEMU HARDDISK (scsi)
Disk /dev/sda: 629145600s
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start      End          Size         Type        File system  Flags
  1      63s       2047s       1985s        primary    Free Space
  2     2048s    976895s    974848s      primary    ext2         boot
  3     976896s  629145599s 628168704s  primary    lvm
```

Nun mit dem Befeh:

```
quit
```

raus.

PV Volume vergrößern

die PV Group rausfinden.

Befehl:

```
pvs

Ausgabe:

PV[]VG[]Fmt[]Attr[]PSize[]Free
/dev/sda2[]nvistaVG[]lvm2[]a--[]200,53[]0
```

Nun wissen wir das PV = /dev/sda2 ist und das die VG nvistaVG heißt.

Als erstes müssen wir die PV vergrößern.

Befehl:

```
pvresize <partition>

in userem Beispiel

pvresize /dev/sda2
```

Ausgabe:

```
Physical volume "/dev/sda2" changed
1 physical volume(s) resized / 0 physical volume(s) not resized
```

Nun überprüfen ob das Volume vergrößert wurde

```
pvs
```

Ausgabe:

```
root@ndc:~# pvresize /dev/sda2
Physical volume "/dev/sda2" changed
1 physical volume(s) resized / 0 physical volume(s) not resized
root@ndc:~# pvs
PV          VG          Fmt Attr PSize  PFree
/dev/sda2  nvistaVG    lvm2 a-- 299,53g 200,00g
root@ndc:~#
```

Nun mit Befeh:

```
vgs
```

sehen wir, das auch gleich die volume group mit vergrößert wurde.

Ausgabe:

```
root@ndc:~# vgs
VG          #PV #LV #SN Attr   VSize  VFree
nvistaVG    1   2   0 wz--n- 299,53g 200,00g
root@ndc:~#
```

Logical Volume aufs Maximale Vergrößern

Mite dem Befehlen anzeigen lassen welche LogicalVolumes es gibt

```
ls /dev/mapper
```

Ausgabe:

```
root@ndc:~# ls /dev/mapper/
control nvistaVG-ndcVolume nvistaVG-swapVolume
root@ndc:~#
```

Nun das Volume nvistaVG-ndcVolume vergrößern.

mit dem Befehl

```
lvextend --resizefs -l +100%FREE /dev/mapper/<volumegroup>--<logicalvolume>
```

In unserem Beispiel

```
lvextend --resizefs -l +100%FREE /dev/mapper/nvistaVG-ndcVolume
```

Ausgabe:

```
root@ndc:~# lvresize --resizefs -l +100%FREE /dev/mapper/nvistaVG-ndcVolume
  Size of logical volume nvistaVG/ndcVolume changed from 91,27 GiB (23365 extents) to 291,27 GiB (74565 extents).
  Logical volume ndcVolume successfully resized
resize2fs 1.42.12 (29-Aug-2014)
Dateisystem bei /dev/mapper/nvistaVG-ndcVolume ist auf / eingehängt; Online-Größenänderung ist
erforderlich
old_desc_blocks = 6, new_desc_blocks = 19
The filesystem on /dev/mapper/nvistaVG-ndcVolume is now 76354560 (4k) blocks long.
root@ndc:~#
```

Speicher überprüfen mit dem Befehl:

```
df -h
```

in der Ausgabe zu sehen, das root laufwerk ist auf 287 GB gewachsen, denn in der Volume Group liegt ja noch das swap Volume und in der ersten Partition die Boot Partition, so das das ndcVolume nicht die Gesamtkapazität von 300 GB haben kann:

```
root@ndc:~# df -h
Dateisystem      Größe Benutzt Verf. Verw% Eingehängt auf
udev             7,8G          0  7,8G   0% /dev
tmpfs            3,2G      8,6M  3,2G   1% /run
/dev/dm-0        287G      60G  215G  22% /
tmpfs            7,9G          0  7,9G   0% /dev/shm
tmpfs            5,0M          0  5,0M   0% /run/lock
tmpfs            7,9G          0  7,9G   0% /sys/fs/cgroup
/dev/sda1        461M      67M  371M  16% /boot
root@ndc:~#
```

Fertig

Proxmox Host LVM

Vergrößern

Laufwerk vergrößern und LVM Volumes vergrößern

Partition vergrößern mit parted

Nun im Terminal erstmal die Partitionen und die Größen rausbekommen mit dem Befehl

```
lsblk
```

Ausgabe:

```
NAME                                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sr0                                  11:0    1   3,6G  0 rom
sda                                  8:0     0  300G  0 disk
├─sda2                                8:2     0  99,5G  0 part
│   ├─nvistaVG-swapVolume            254:1     0   8,3G  0 lvm  [SWAP]
│   └─nvistaVG-ndcVolume             254:0     0  91,3G  0 lvm  /
└─sda1                                8:1     0   476M  0 part  /boot
root@ndc:~# _
```

Wie in diesem Beispiel zu sehen, liegen die LVs auf dem Laufwerk /dev/sda2.

Die Festplatte sda hat 300 GB erreicht.

Nun müssen wir die Partiton auch auf 300 GB bringen.

Das machen wir mit dem Programm parted.

Sollte parted nicht installiert sein, dieses nachinstallieren.

```
apt-get install parted    [On Debian/Ubuntu systems]
yum install parted        [On RHEL/CentOS and Fedora]
dnf install parted        [On Fedora 22+ versions]
```

Nun parted starten mit dem Befehl

```
parted
```

Nun dort den Befehl eingeben:

```
select /dev/sda
```

Ausgabe:

```
(parted) select /dev/sda
Using /dev/sda
(parted)
```

Nun die Einheiten vom MB auf Sektoren umstellen damit wir den startsektor bekommen. Dazu den Befehl

```
unit s
```

verwenden.

Nun lassen wir uns die Partitionen inklusive freien Bereich durch den Parameter `free` anzeigen, mit dem Befehl:

```
print free
```

Ausgabe:

```
(parted) print free
Model: QEMU QEMU HARDDISK (scsi)
Disk /dev/sda: 629145600s
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start      End          Size         Type         File system  Flags
   1     63s       2047s       1985s                Free Space
   2    2048s    976895s    974848s    primary    ext2         boot
   2    976896s 209713151s 208736256s primary    lvm
   2   209713152s 629145599s 419432448s                Free Space

(parted)
```

Jetzt sehen wir die Infos. Uns interessiert das Ende des Laufwerks.

Also Sektor : 629145599s

Dieses müssen wir später angeben.

Nun die Partiton vergrößern mit angabe der Partion, hier Partition 2, da liegt unser lvm drauf.

Befehl:

```
resizepart 2
```

Nun wird nach dem Ende gefragt. Vorgeschlagen wird uns das aktuelle ende der Partition.

Wir wollen die Partition ja auf das ende des Datenträgers legen.

Somit die Volle Kapazität des Datenträgers.

```
(parted) resizepart 2
End? [209713151s]? _
```

Also geben wir als Endsektor 629145599s

```
(parted) resizepart 2
End? [209713151s]? 629145599s
(parted) _
```

Nun nochmals ausgeben mit dem Befehl:

```
print free
```

nun sieht man die 2 Partition geht bis ans ende. Voilla

```
(parted) print free
Model: QEMU QEMU HARDDISK (scsi)
Disk /dev/sda: 629145600s
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start      End          Size         Type         File system  Flags
   1     2048s     976895s     974848s     primary     ext2         boot
   2     976896s   629145599s  628168704s  primary                                     lvm

(parted) _
```

Nun mit dem Befehl:

```
quit
```

raus.

PV Volume vergrößern

die PV Group rausfinden.

Befehl:

```
pvs
```

Ausgabe:

```
PV[]VG[]Fmt[]Attr[]PSize[]Free
/dev/sda2[]nvistaVG[]lvm2[]a--[]200,53[]0
```

Nun wissen wir das PV = /dev/sda2 ist und das die VG nvistaVG heißt.

Als erstes müssen wir die PV vergrößern.

Befehl:

```
pvresize <partition>  
in userem Beispiel  
pvresize /dev/sda2
```

Ausgabe:

```
Physical volume "/dev/sda2" changed  
1 physical volume(s) resized / 0 physical volume(s) not resized
```

Nun überprüfen ob das Volume vergrößert wurde

```
pvs
```

Ausgabe:

```
root@ndc:~# pvresize /dev/sda2  
Physical volume "/dev/sda2" changed  
1 physical volume(s) resized / 0 physical volume(s) not resized  
root@ndc:~# pvs  
PV          VG          Fmt Attr PSize  PFree  
/dev/sda2  nvistaVG    lvm2 a--  299,53g 200,00g  
root@ndc:~#
```

Nun mit Befehl:

```
vgs
```

sehen wir, das auch gleich die volume group mit vergrößert wurde.

Ausgabe:

```
root@ndc:~# vgs  
VG          #PV #LV #SN Attr   VSize  VFree  
nvistaVG    1   2   0 wz--n- 299,53g 200,00g  
root@ndc:~#
```

Logical Volume aufs Maximale Vergrößern

Nun das Meta Volume um 32GB erweitern und dann das Daten Volume den rest mit dem Befehl

```
lvextend -L +32G /dev/pve/data_tmeta
```

```
lvextend -l +100%FREE /dev/pve/data
```

Ausgabe:

```
lvextend -L +32G /dev/pve/data_tmeta
```

```
lvextend -l +100%FREE /dev/pve/data
```

Reached maximum pool metadata size <15.88 GiB (4065 extents).

Thin pool will use metadata without cropping.

Size of logical volume pve/data_tmeta changed from 1.44 GiB (369 extents) to <15.88 GiB (4065 extents).

Logical volume pve/data_tmeta successfully resized.

Size of logical volume pve/data_tdata changed from <157.54 GiB (40329 extents) to <1.76 TiB (460266 extents).

Logical volume pve/data successfully resized.

Speicher überprüfen in der Oberfläche

Node 'pve01'

Reboot Shutdown Shell Bulk Actions Help

Search Reload Create: Thinpool Thinpool pve/data: More

Name ↑	Volume Group	Usage	Size	Used	Metadata Usage	Metadata Size	Metadata Used
data	pve	2%	1.93 TB	33.98 GB	0%	17.05 GB	35.80 M
nvme	nvme	48%	980.07 GB	481.90 GB	1%	10.00 GB	143.05 M

Fertig