

# LVM Grundkonfiguration

Im folgenden Artikel wird die grundlegende Konfiguration von LVs erläutert. Das verwendete System ist ein Ubuntu Server 10.4 mit dem Kernel 2.6.32-24 und der LVM-Version 2.02.54(1) (2009-10-26). Es wird das Anlegen der Partitionen, der Physical Volumes (PVs), einer Volume Group (VG) und den darauf aufsetzenden Logical Volumes (LVs) an einem Beispiel erklärt.

## Anlegen der Partitionen

Zu Beginn werden die Partitionen für die PVs angelegt. Folgende Dinge müssen dabei beachtet werden:

- Partition Alignment
  - Anzeige umstellen auf Sektoren (Switch „-u“)
  - DOS-compatible Mode ausschalten (Switch „-c“)
- Für späteres LVM-Management
  - System ID der Partition auf „8e“ ändern (Switch „-t“ bei fdisk)

Nach den Änderungen sieht die Partitionstabelle wie folgt aus:

```
root@ubuntu:/home/tktest# fdisk -lu

Disk /dev/sda: 5368 MB, 5368709120 bytes
255 heads, 63 sectors/track, 652 cylinders, total 10485760 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00051afd

      Device Boot      Start        End      Blocks   Id  System
/dev/sda1  *        2048     9920511    4959232    83  Linux
Partition 1 does not end on cylinder boundary.
/dev/sda2            9922558    10483711    280577     5  Extended
Partition 2 does not end on cylinder boundary.
/dev/sda5            9922560    10483711    280576    82  Linux swap / Solaris

Disk /dev/sdb: 2147 MB, 2147483648 bytes
22 heads, 16 sectors/track, 11915 cylinders, total 4194304 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x1673663d

      Device Boot      Start        End      Blocks   Id  System
/dev/sdb1                2048    4194303   2096128    8e  Linux LVM
```

```
Disk /dev/sdc: 2147 MB, 2147483648 bytes
22 heads, 16 sectors/track, 11915 cylinders, total 4194304 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xbd277faf

Device Boot Start End Blocks Id System
/dev/sdc1 2048 4194303 2096128 8e Linux LVM
```

## Erstellung von PVs

PVs beinhalten auch immer Metadaten für die Verwaltung der Volumes (s.a. Artikel zu [LVM Grundlagen](#)). Standardmäßig werden für die Metadaten 255 Sektoren (à 512 Byte) angelegt. Eine zu kleine Metadaten-Area kann unter anderem dazu führen, dass z.B. keine Snapshots mehr erzeugt werden können: [LVM VG vgname metadata too large for circular buffer beheben](#). Daher kann es durchaus sinnvoll sein, dass eine größere Metadaten-Area konfiguriert wird. Will man eine Vergrößerung der Metadaten-Area, so fügt man dem Kommando „pvcreate“ den Parameter „--metadatasize“ und anschließend die gewünschte Größe bei: --metadatasize

Anschließend werden die Partitionen als PV initialisiert.

```
root@ubuntu:~# pvcreate /dev/sdb1
Physical volume "/dev/sdb1" successfully created
root@ubuntu:~# pvcreate /dev/sdc1
Physical volume "/dev/sdc1" successfully created
```

Die Befehle „pvs“ und „pvdisplay“ bieten dabei zahlreiche Möglichkeiten sich den aktuellen Status der PVs anzeigen zu lassen.

```
root@ubuntu:~# pvs
PV VG Fmt Attr PSize PFree
/dev/sdb1 lvm2 -- 2.00g 2.00g
/dev/sdc1 lvm2 -- 2.00g 2.00g
```

## Anlegen einer VG

Die zuvor angelegten PVs werden nun zu einer VG zusammengefasst.

```
root@ubuntu:~# vgcreate vg00 /dev/sdb1 /dev/sdc1
Volume group "vg00" successfully created
```

Der Befehl „pvdisplay“ zeigt nun auch, dass eine VG mit den PVs angelegt wurde:

```
root@ubuntu:~# pvdisplay
--- Physical volume ---
PV Name          /dev/sdb1
VG Name          vg00
PV Size          2.00 GiB / not usable 3.00 MiB
Allocatable      yes
PE Size          4.00 MiB
Total PE         511
Free PE          511
Allocated PE    0
PV UUID          fl9ipM-bhhQ-V46G-2iH3-R3yZ-9DsN-JbRmY9

--- Physical volume ---
PV Name          /dev/sdc1
VG Name          vg00
PV Size          2.00 GiB / not usable 3.00 MiB
Allocatable      yes
PE Size          4.00 MiB
Total PE         511
Free PE          511
Allocated PE    0
PV UUID          d1iY5L-ac3F-W5Sz-zyaE-uaT3-f66r-I3831o
```

vgdisplay zeigt außerdem die Informationen zur VG an:

```
root@ubuntu:~# vgdisplay
--- Volume group ---
VG Name          vg00
System ID
Format          lvm2
Metadata Areas  2
Metadata Sequence No  1
VG Access       read/write
VG Status       resizable
MAX LV          0
Cur LV          0
Open LV          0
Max PV          0
Cur PV          2
Act PV          2
VG Size          3.99 GiB
PE Size          4.00 MiB
Total PE         1022
Alloc PE / Size 0 / 0
Free PE / Size  1022 / 3.99 GiB
VG UUID          YTEj9f-9LCT-E0P5-JBEA-YHSz-c0R1-TMzVmy
```

Hier fällt auf, dass die PE Größe 4.00 MiB beträgt. Seit dem lvm2-Format ist die Anzahl der PEs nicht mehr beschränkt, laut Man-Page von vgcreate kann eine hohe Anzahl von PEs aber die Tools verlangsamen. Auf die I/O-Performanz der Logical Volumes wirkt sich die Anzahl der PEs aber nicht aus! Will man die PE-Größe ändern, so fügt man bei „vgcreate“ den Parameter

-s, --physicalextentsize PhysicalExtentSize

hinzu.

## Anlegen von LVs

From:

<https://wiki.cooltux.net/> - TuxNet DokuWiki

Permanent link:

[https://wiki.cooltux.net/doku.php?id=it-wiki:linux:lvm\\_index:lvm-grundkonfiguration&rev=1676017488](https://wiki.cooltux.net/doku.php?id=it-wiki:linux:lvm_index:lvm-grundkonfiguration&rev=1676017488)

Last update: 2023/02/10 08:24

