

Praxisbeispiel LVM

Prinzipien von Linux

CNAM Sommersemester 2014

1 Partitionen als Physical Volumes einrichten

Mit fdisk, cfdisk oder sfdisk werden drei Partitionen mit Typ 8e (Linux LVM) angelegt.

Gerät	boot.	Anfang	Ende	Blöcke	Id	System
/dev/sdb1		2048	1230847	614400	8e	Linux LVM
/dev/sdb2		1230848	2459647	614400	8e	Linux LVM
/dev/sdb3		2459648	3688447	614400	8e	Linux LVM

Partitionen als Physical Volume einrichten:

```
[root@localhost ~]# pvcreate /dev/sdb1
...
[root@localhost ~]# pvcreate /dev/sdb3
```

Anzeige des Ergebnisses:

```
[root@localhost ~]# pvscan

PV /dev/sdb1                lvm2 [600,00 MiB]
PV /dev/sdb2                lvm2 [600,00 MiB]
PV /dev/sdb3                lvm2 [600,00 MiB]
```

2 Volume Group erzeugen

```
[root@localhost ~]# vgcreate vg1 /dev/sdb1 /dev/sdb2
```

```
[root@localhost ~]# pvdisplay /dev/sdb1
--- Physical volume ---
PV Name                /dev/sdb1
VG Name                vg1
```

```

PV Size          600,00 MiB / not usable 4,00 MiB
Allocatable      yes
PE Size          4,00 MiB
Total PE         149
Free PE          149
Allocated PE     0

```

```

[root@localhost ~]# vdisplay vg1
--- Volume group ---
VG Name          vg1
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          1,16 GiB
PE Size          4,00 MiB
Total PE         298
Alloc PE / Size  0 / 0
Free PE / Size   298 / 1,16 GiB

```

3 Anlegen der Logical Volumes

```

[root@localhost ~]# lvcreate -L 400M -n lv1 vg1
Logical volume "lv1" created

```

```

[root@localhost ~]# lvcreate -l 100%FREE -n lv2 vg1
Logical volume "lv2" created

```

```

[root@localhost ~]# ldisplay /dev/vg1/lv1
--- Logical volume ---
LV Path          /dev/vg1/lv1
LV Name          lv1
VG Name          vg1
LV UUID          jhe7Tc-GNip-ZTbP-QgGb-yZpz-zQPW-Svtq22
LV Write Access  read/write
LV Creation host, time localhost.localdomain, 2012-11-13 10:40:05 +0100
LV Status        available

```

```

# open                0
LV Size               400,00 MiB
Current LE            100
Segments              1
Allocation             inherit
Read ahead sectors    auto
- currently set to    256
Block device          253:3

```

```

[root@localhost ~]# lvsdisplay /dev/vg1/lv2
--- Logical volume ---
LV Path                /dev/vg1/lv2
LV Name                lv2
VG Name                vg1
LV UUID                W0JoSM-BEc0-lNOL-WGie-dsGk-ah0N-4YF5PO
LV Write Access        read/write
LV Creation host, time localhost.localdomain, 2012-11-13 10:40:29 +0100
LV Status              available
# open                 0
LV Size                792,00 MiB
Current LE             198
Segments               2
Allocation             inherit
Read ahead sectors    auto
- currently set to    256
Block device          253:4

```

4 Einrichten der Dateisysteme

```

[root@localhost ~]# mkfs -t ext4 /dev/vg1/lv1
mke2fs 1.42 (29-Nov-2011)
Dateisystem-Label=
OS-Typ: Linux
Blockgröße=1024 (log=0)
Fragmentgröße=1024 (log=0)
Stride=0 Blöcke, Stripebreite=0 Blöcke
102400 Inodes, 409600 Blöcke
20480 Blöcke (5.00%) reserviert für den Superuser
Erster Datenblock=1
Maximale Dateisystem-Blöcke=67633152
50 Blockgruppen
8192 Blöcke pro Gruppe, 8192 Fragmente pro Gruppe
2048 Inodes pro Gruppe
Superblock-Sicherungskopien gespeichert in den Blöcken:
8193, 24577, 40961, 57345, 73729, 204801, 221185, 401409

```

Platz für Gruppentabellen wird angefordert: erledigt
Inode-Tabellen werden geschrieben: erledigt
Erstelle Journal (8192 Blöcke): erledigt
Schreibe Superblöcke und Dateisystem-Accountinginformationen: erledigt

```
[root@localhost ~]# mkfs -t ext4 /dev/vg1/lv2
mke2fs 1.42 (29-Nov-2011)
Dateisystem-Label=
OS-Typ: Linux
Blockgröße=4096 (log=2)
Fragmentgröße=4096 (log=2)
Stride=0 Blöcke, Stripebreite=0 Blöcke
50736 Inodes, 202752 Blöcke
10137 Blöcke (5.00%) reserviert für den Superuser
Erster Datenblock=0
Maximale Dateisystem-Blöcke=209715200
7 Blockgruppen
32768 Blöcke pro Gruppe, 32768 Fragmente pro Gruppe
7248 Inodes pro Gruppe
Superblock-Sicherungskopien gespeichert in den Blöcken:
32768, 98304, 163840
```

Platz für Gruppentabellen wird angefordert: erledigt
Inode-Tabellen werden geschrieben: erledigt
Erstelle Journal (4096 Blöcke): erledigt
Schreibe Superblöcke und Dateisystem-Accountinginformationen: erledigt

5 Änderung 1: Erweitern

Bisheriger Stand:

```
[root@localhost ~]# vgdisplay vg1
--- Volume group ---
VG Name                vg1
System ID
Format                 lvm2
Metadata Areas        2
Metadata Sequence No  3
VG Access              read/write
VG Status              resizable
MAX LV                 0
Cur LV                2
Open LV                0
Max PV                 0
```

```

Cur PV          2
Act PV          2
VG Size         1,16 GiB
PE Size         4,00 MiB
Total PE        298
Alloc PE / Size 298 / 1,16 GiB
Free PE / Size  0 / 0

```

Nun wird ein neues Physical Volume hinzugefügt:

```

[root@localhost ~]# vgextend vg1 /dev/sdb3
Volume group "vg1" successfully extended

```

```

[root@localhost ~]# vgsdisplay vg1
--- Volume group ---
VG Name          vg1
System ID
Format           lvm2
Metadata Areas   3
Metadata Sequence No 4
VG Access        read/write
VG Status        resizable
MAX LV           0
Cur LV          2
Open LV          0
Max PV           0
Cur PV          3
Act PV           3
VG Size          1,75 GiB
PE Size          4,00 MiB
Total PE         447
Alloc PE / Size 298 / 1,16 GiB
Free PE / Size   149 / 596,00 MiB

```

Nun kann z. B. eines der Logical Volumes vergrößert werden:

```

[root@localhost ~]# lvextend -L +200M -r /dev/vg1/lv1
fsck von util-linux 2.21.1
/dev/mapper/vg1-lv1: sauber, 11/102400 Dateien, 23438/409600 Blöcke
  Extending logical volume lv1 to 600,00 MiB
  Logical volume lv1 successfully resized
resize2fs 1.42 (29-Nov-2011)
Die Grösse des Dateisystems auf /dev/mapper/vg1-lv1 wird auf 614400 (1k) Blöcke
Das Dateisystem auf /dev/mapper/vg1-lv1 ist nun 614400 Blöcke groß.

```

```
[root@localhost ~]# vgdisplay vg1
--- Volume group ---
VG Name                vg1
System ID
Format                 lvm2
Metadata Areas         3
Metadata Sequence No   5
VG Access              read/write
VG Status              resizable
MAX LV                 0
Cur LV                2
Open LV                0
Max PV                 0
Cur PV                3
Act PV                 3
VG Size                1,75 GiB
PE Size                4,00 MiB
Total PE               447
Alloc PE / Size        348 / 1,36 GiB
Free PE / Size         99 / 396,00 MiB
```

6 Änderung 2: Verkleinern und Entfernen eines PV

Nun soll `/dev/sdb1` entfernt werden. Da dieses PV aus 149 PE besteht, in der Volume Group aber nur 99 PE frei sind, soll ein LV entsprechend verkleinert werden:

```
[root@localhost ~]# lvreduce -l -50 -r /dev/vg1/lv2
fsck von util-linux 2.21.1
/dev/mapper/vg1-lv2: sauber, 11/50736 Dateien, 7491/202752 Blöcke
resize2fs 1.42 (29-Nov-2011)
Die Grösse des Dateisystems auf /dev/mapper/vg1-lv2 wird auf 151552 (4k) Blöcke
```

Das Dateisystem auf `/dev/mapper/vg1-lv2` ist nun 151552 Blöcke groß.

```
Reducing logical volume lv2 to 592,00 MiB
Logical volume lv2 successfully resized
```

```
[root@localhost ~]# vgdisplay vg1
--- Volume group ---
VG Name                vg1
System ID
Format                 lvm2
Metadata Areas         3
Metadata Sequence No   6
VG Access              read/write
```

```

VG Status          resizable
MAX LV            0
Cur LV           2
Open LV           0
Max PV            0
Cur PV           3
Act PV           3
VG Size           1,75 GiB
PE Size           4,00 MiB
Total PE          447
Alloc PE / Size   298 / 1,16 GiB
Free PE / Size    149 / 596,00 MiB

```

Nun ist eigentlich genug Platz, aber das PV kann noch nicht entfernt werden, weil darauf PEs allokiert sind:

```

[root@localhost ~]# pvdisplay /dev/sdb1
--- Physical volume ---
PV Name           /dev/sdb1
VG Name           vg1
PV Size           600,00 MiB / not usable 4,00 MiB
Allocatable       yes
PE Size           4,00 MiB
Total PE          149
Free PE           49
Allocated PE      100

```

```

[root@localhost ~]# pvdisplay /dev/sdb2
--- Physical volume ---
PV Name           /dev/sdb2
VG Name           vg1
PV Size           600,00 MiB / not usable 4,00 MiB
Allocatable       yes
PE Size           4,00 MiB
Total PE          149
Free PE           1
Allocated PE      148

```

```

[root@localhost ~]# pvdisplay /dev/sdb3
--- Physical volume ---
PV Name           /dev/sdb3
VG Name           vg1
PV Size           600,00 MiB / not usable 4,00 MiB
Allocatable       yes
PE Size           4,00 MiB
Total PE          149

```

```
Free PE          99
Allocated PE     50
```

Also müssen zunächst die allokierten PE vom Physical Volume auf die anderen verschoben werden:

```
[root@localhost ~]# pvmove /dev/sdb1
/dev/sdb1: Moved: 1,0%
/dev/sdb1: Moved: 18,0%
/dev/sdb1: Moved: 33,0%
/dev/sdb1: Moved: 47,0%
/dev/sdb1: Moved: 62,0%
/dev/sdb1: Moved: 78,0%
/dev/sdb1: Moved: 94,0%
/dev/sdb1: Moved: 99,0%
/dev/sdb1: Moved: 100,0%
```

```
[root@localhost ~]# pvdisplay /dev/sdb?
--- Physical volume ---
PV Name                /dev/sdb1
VG Name                vg1
PV Size                600,00 MiB / not usable 4,00 MiB
Allocatable           yes
PE Size               4,00 MiB
Total PE              149
Free PE               149
Allocated PE          0
PV UUID               K9DdTx-BJR6-1QbC-CqBE-inW7-4sxD-kxyN99

--- Physical volume ---
PV Name                /dev/sdb2
VG Name                vg1
PV Size                600,00 MiB / not usable 4,00 MiB
Allocatable           yes (but full)
PE Size               4,00 MiB
Total PE              149
Free PE               0
Allocated PE          149
PV UUID               JtL5hN-qUw0-lThe-Fe0H-Tz1G-hAPl-L8xhc0

--- Physical volume ---
PV Name                /dev/sdb3
VG Name                vg1
PV Size                600,00 MiB / not usable 4,00 MiB
Allocatable           yes (but full)
PE Size               4,00 MiB
Total PE              149
```



```
Free PE          0
Allocated PE     149
PV UUID          jAzczb-UQ3k-0a00-D1eh-AB23-tROs-2xt9Lp
```

Hiermit ist das Ziel erreicht, das PV sdb1 ist frei und kann aus der Volume Group herausgelöst werden:

```
[root@localhost ~]# vgreduce vg1 /dev/sdb1
Removed "/dev/sdb1" from volume group "vg1"
[root@localhost ~]# vgdisplay vg1
--- Volume group ---
VG Name          vg1
System ID
Format           lvm2
Metadata Areas   2
Metadata Sequence No 11
VG Access        read/write
VG Status        resizable
MAX LV           0
Cur LV          2
Open LV          0
Max PV           0
Cur PV          2
Act PV           2
VG Size          1,16 GiB
PE Size          4,00 MiB
Total PE         298
Alloc PE / Size  298 / 1,16 GiB
Free PE / Size   0 / 0
```