



Grundlagen und Hinweise auf mögliche Stolperfallen

# LVM2



Eine Festplatte wächst nicht – die Daten darauf aber schon. Einen Ausweg aus diesem Dilemma bietet der Logical Volume Manager (LVM2), der es erlaubt, den physischen Speicher, der einem Filesystem zur Verfügung steht, dynamisch zu verwalten. Charly Kühnast, Ludger Köhler

Der **Logical Volume Manager** bildet eine Abstraktionsschicht zwischen dem physischen Datenträger und dem Dateisystem. Ganze Festplatten oder auf ihnen angelegte Partitionen fasst er zu logischen Datenträgern (Logical Volumes, LVs) zusammen (**Abbildung 1**).

Wird der Plattenplatz knapp, lässt sich dann sehr einfach eine weitere Festplatte (Physical Volume, PV) oder Partition hinzufügen. Das LV wächst dadurch entsprechend, und die folgende Vergrößerung des Dateisystems stellt den so gewonnenen Platz dem System zur Verfügung.

Für die Administration des LVM stehen gelegentlich distributionsspezifische, teils grafische Werkzeuge bereit; Suse und Open Suse integrieren beispielsweise eine LVM-Verwaltung in ihr Setup-Tool YaST. Die Administration von LVM2 kann daneben auch über EVMS erfolgen. Dieser Beitrag erläutert jedoch die notwendigen Schritte auf der Kommandozeile, die distributionsunabhängig sind.

## Voraussetzungen

Um den LVM benutzen zu können, muss im Kernel die Devicemapper-Unterstützung aktiviert sein. Das Kernel-Modul heißt »dm\_mod«. Mit »lsof | grep dm\_mod« lässt sich überprüfen, ob es geladen ist. Außerdem müssen die LVM-Tools installiert sein. Das Paket heißt bei den meisten Distributionen »lvm2« oder »lvm-tools«.

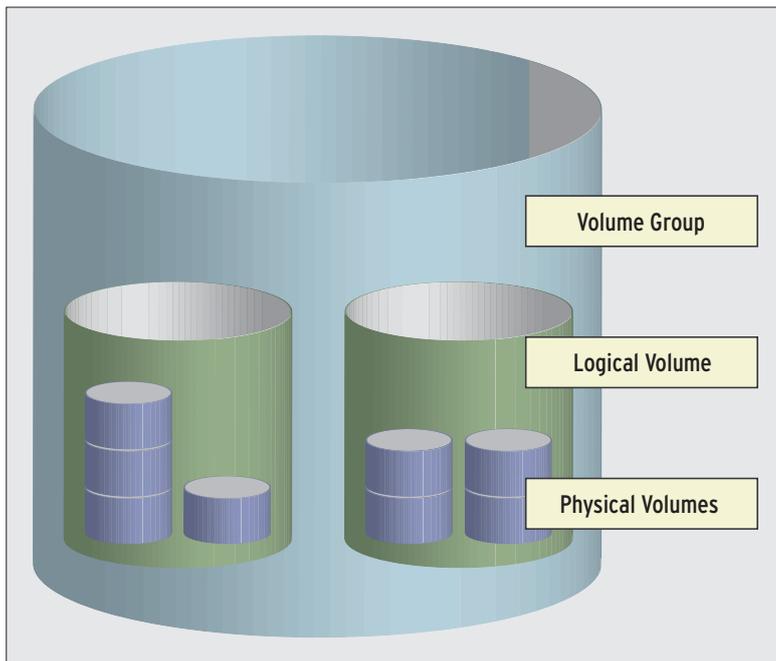
Sind die Voraussetzungen erfüllt, kann man ein LVM-Volume einrichten. Das passiert in folgenden Schritten:

- **Physical Volumes vorbereiten:** Zunächst legt man mit »fdisk« die gewünschten Partitionen an. Als Partitionstyp wählt man dabei Linux LVM (»8E«). Danach kann der Admin dieses PV dem Logical Volume Manager bekannt machen. Erst wenn das geschehen ist, sieht der LVM diese Platten beziehungsweise Partitionen und kann mit ihnen arbeiten:

```
pvcreate /dev/sdb      #ganze Platte
pvcreate /dev/sdc1    # Partition
```

Das Kommando »pvcreate« akzeptiert auch mehrere Platten- oder Partitionsnamen als Parameter. Sie müssen durch ein Leerzeichen getrennt sein.

- **Eine Volume Group (VG) anlegen:** Eine VG verbindet verschiedene PVs zu einer logischen Einheit. Diese Einheit bekommt einen Namen, den der Administrator frei wählen



**Abbildung 1:** Logische Volumes abstrahieren von den physikalischen Devices, aus denen sie bestehen. Für das Betriebssystem sind es große Festplatten, die zusätzlich wachsen und schrumpfen können.

darf (hier: »my\_vg«). Diese Stelle markiert die eigentliche Abstraktion der logischen Datenträger von den physikalischen Festplatten:

```
vgcreate my_vg /dev/sdb /dev/sdc1
```

Ab jetzt sind die physikalischen Platten hinter »my\_vg« verborgen.

- **Logische Volumes anlegen:** Die Volume Group »my\_vg« kann man sich nun wie eine große, unpartitionierte Festplatte vorstellen. Ihre Partitionierung erfolgt durch das Anlegen der Logischen Volumes (LVs). Auch die LVs bekommen Namen, die im Prinzip frei wählbar sind. Der Übersichtlichkeit halber stimmen die Namen oft mit den Mountpoints überein, an denen die LVs später Dienst tun sollen:

```
lvcreate -L40G -nhome my_vg
lvcreate -L120G -nvar my_vg
```

Diese Befehlsfolge legt als Beispiel zwei logische Volumes von 40 beziehungsweise 120 Gbyte an, die »home« und »var« heißen.

- **Formatieren und mounten:** Der letzte Schritt, bevor die neuen Volumes ihre Arbeit aufnehmen können, besteht im Anlegen der Dateisysteme und dem Einhängen am gewünschten Mountpunkt: ▶

```
mkfs.ext3 /dev/my_vg/home
mkfs.ext3 /dev/my_vg/var
mount -t ext3 /dev/my_vg/home /home
mount -t ext3 /dev/my_vg/var /var
```

Hat alles funktioniert, kann man das Mounten in Zukunft auch der »fstab« überlassen:

```
/dev/my_vg/home /home ext3 defaults
/dev/my_vg/var /var ext3 defaults
```

## Vergrößern eines Volumes

Das logische Volume für die »/home«-Partition ist zu klein. Eine zusätzliche 60 GByte große Partition »/dev/sde1« soll sie vergrößern. Das bewerkstelligt der Befehl:

```
vgextend my_vg /dev/sde1
```

Er fügt die neue Partition der Volume Group »my\_vg« hinzu. Anschließend lässt sich auch das logische Volume »home« um den zusätzlichen Plattenplatz erweitern:

```
lvextend -L+60G /dev/my_vg/home
```

Zuletzt ist noch das Ext3-Dateisystem auf der Partition »/home« entsprechend zu vergrößern, damit der Platz auch nutzbar ist. Auf diesen Punkt geht der Beitrag weiter unten ein.

Um einen schnellen Überblick über die LVs und ihre Größe zu bekommen, genügt das Kommando »lvscan«. Mehr Details zeigt jedoch der Befehl »lvdisplay -a«.

## Stolperfallen

- **Bootpartition auf LVM:** So gut wie jede aktuelle Linux-Distribution benutzt die Initial RAM Disk (»initrd«). Sie liegt üblicherweise auf der Bootpartition. Solange sie nicht geladen ist, bleibt LVM unbenutzbar, ergo sollte »/boot« LVM-frei bleiben.
- **Swap auf LVM:** Es ist technisch ohne weiteres möglich, die Swap-Partition auf einem LV einzurichten. Empfehlenswert ist es aber nicht, weil die Performance darunter leiden kann. Während eine normale Swap-Partition einen zusammenhängenden Bereich auf einer Platte belegt, verteilt sie LVM sehr wahrscheinlich auf verschiedene Zonen des oder der Datenträger, wodurch die Zugriffszeiten sich verlängern.
- **Root-Partition auf LVM:** Ebenso wie die Swap-Partition kann man auch das Root-System (»/«) auf einem logischen Volume ein-

richten. Allerdings lassen sich etwaige Reparatur- und Wartungsarbeiten am LVM nicht bei eingehängtem Filesystem erledigen. Das Rebooten des Servers in das Live-Filesystem einer Rettungs-CD ist notwendig, und man tut gut daran, sich vorher zu vergewissern, dass das Rettungssystem auch die notwendigen Tools und Module für LVM2 mitbringt. Wer Knoppix benutzt, kann fehlende Werkzeuge per »apt-get« nachinstallieren.

## Filesysteme

Den LVM setzt der Admin in erster Linie wegen seiner Flexibilität ein, die er bei der Verwaltung des Plattenplatzes bietet. Um diese Flexibilität auch vollständig nutzen zu können, muss das gewählte Dateisystem ebenfalls zum Anwendungsprofil passen. Das gilt besonders für die Möglichkeit des Vergrößerns und Verkleinerns. Wichtig ist, im Vorfeld zu klären, ob das Volume in Zukunft nur wächst, man die Möglichkeit der nachträglichen Verkleinerung also außer acht lassen kann, und ob die Größenänderung online, also bei gemountetem Filesystem, erfolgen soll, oder ob es möglich ist, dieses vorher auszuhängen.

Die Eignung der gängigen Filesysteme für diese Aufgaben ist höchst unterschiedlich:

- **Ext3** lässt sich offline vergrößern und verkleinern, mit einem Kernel-Patch auch online (ext3online-Patch). Einige Distributionen enthalten bereits so gepatchte Kernel, zum Beispiel Fedora Core.
- **ReiserFS3** lässt sich online vergrößern, aber nur offline verkleinern.
- **XFS** ist online (und nur online!) vergrößerbar, lässt sich aber weder on- noch offline verkleinern.

## Troubleshooting

Obwohl LVM2 wartungsarm zu betreiben ist, sind manuelle Reparaturarbeiten nicht immer auszuschließen. Zwar ist der Ausfall einer physikalischen Platte heutzutage ein zu vernachlässigendes Problem, weil im professionellen Umfeld die Redundanz der Datenträger durch RAIDs sichergestellt ist. Den Ausfall einer Platte bekommt LVM im Regelfall so überhaupt nicht mit. Für andere Probleme jedoch braucht es Handarbeit. Dazu gehört das Sichern und Wiederherstellen der Metadaten.

Die Metadaten, also Informationen über die physikalischen und logischen Geräte, insbeson-

dere deren Größen und eindeutige Kennzeichnungen (UUID), sollte der Admin unbedingt sichern. LVM besitzt selbst einen Mechanismus dafür, der in der »lvm.conf« aktiviert wird:

```
backup {
    backup = 1
    backup_dir = "/etc/lvm/backup"
    archive = 1
    archive_dir = "/etc/lvm/archive"
    retain_min = 10
    retain_days = 30
}
```

Bei Bedarf, zum Beispiel vor geplanten Änderungen an der Plattenkonfiguration, empfiehlt es sich, das Backup vorsichtshalber noch einmal manuell zu aktivieren, um auf jeden Fall eine Sicherung vom aktuellen Status zu besitzen. Das gelingt mit »vgcfgbackup -f /etc/lvm/archive/my\_vg.vg«

Sollten die Metadaten eines Volumes versehentlich überschrieben worden sein, äußert sich dies beim Ausführen von »lvs« mit einer Fehlermeldung wie zum Beispiel:

```
# lvs -a -o +devices
Couldn't find device with uuid 𐀀
'bIrCWE-HIJZ-qgBF-JB9G-sMnr-Ldei-Wrnchw'.
Couldn't find all physical volumes 𐀀
for volume group my_vg.
```

Um das Volume wieder in Betrieb nehmen zu können, sind jetzt drei Schritte notwendig. Zuerst wird das physikalische Device mit »pvcreate« vorbereitet. Als Parameter muss man die UUID übergeben; sie lässt sich aus der »lvs«-Fehlermeldung oder aus der Backup-Datei entnehmen.

```
pvcreate -uuid 𐀀
"bIrCWE-HIJZ-qgBF-JB9G-sMnr-Ldei-Wrnchw" 𐀀
--restorefile /etc/lvm/archive/my_vg.vg 𐀀
/dev/sdc1
```

Danach kann man mit »vgcfgrestore« das eigentliche Wiederherstellen der Metadaten starten, im Beispiel: »vgcfgrestore -f /etc/lvm/archive/my\_vg.vg«. Zum Schluss ist das wiederhergestellte Volume noch zu aktivieren. Das übernimmt das Kommando »lvchange -ay /dev/my\_vg/lv-name«. (jcb) ■■■