

创建 LVM 系统

内容提要

1. 理解在新硬盘上创建 LVM 系统的过程。
2. 掌握常用的物理卷操作命令：`pvcreate`、`pvscan`、`pvdisplay`
3. 掌握常用的卷组操作命令：`vgcreate`、`vgdisplay`
4. 掌握常用的逻辑卷操作命令：`lvcreate`、`lvdisplay`

概述

创建 LVM 系统可以在安装 Linux 过程中进行，参见 [使用 LVM 安装 CentOS](#)。本节讲述在安装好系统后创建 LVM 系统的过程。

本节的操作目的：

1. 在系统中添加一块新硬盘
2. 在新硬盘上创建物理卷
3. 将新创建的物理卷添加到卷组
4. 在卷组中创建逻辑卷
5. 在逻辑卷中创建 `ext3` 文件系统
6. 将创建的文件系统挂装到 `/srv/www`

在磁盘或磁盘分区上创建物理卷PV

分配磁盘用于 LVM2 有如下三种方式：

1. 在未分区的磁盘上创建单个物理卷
2. 在有一个分区的磁盘上创建单个的物理卷
3. 在有多个分区的磁盘上创建多个的物理卷

由于现在的磁盘容量很大，建议使用第三种方式避免因磁盘局部损坏造成整盘数据丢失。每个磁盘分区的大小建议为 40G 左右。

创建 LVM 类型的分区

可以使用 `fdisk` 或 `parted` 命令创建 LVM 类型的分区，LVM 分区的类型 ID 为 `8e`。下面给出一个分区过程示例。

```
# fdisk /dev/sdb
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel. Changes will remain in memory only,
until you decide to write them. After that, of course, the previous
content won't be recoverable.
The number of cylinders for this disk is set to 1044.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
```

```
2) booting and partitioning software from other OSs
(e.g., DOS FDISK, OS/2 FDISK)
Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)
Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-1044, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-1044, default 1044): +4G
Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
e
Partition number (1-4): 2
First cylinder (488-1044, default 488):
Using default value 488
Last cylinder or +size or +sizeM or +sizeK (488-1044, default 1044):
Using default value 1044
Command (m for help): n
Command action
  l   logical (5 or over)
  p   primary partition (1-4)
l
First cylinder (488-1044, default 488):
Using default value 488
Last cylinder or +size or +sizeM or +sizeK (488-1044, default 1044):
Using default value 1044
Command (m for help): p
Disk /dev/sdb: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            1         487      3911796    83   Linux
/dev/sdb2          488        1044     4474102+     5   Extended
/dev/sdb5          488        1044     4474071    83   Linux
Command (m for help): t
Partition number (1-5): 1
Hex code (type L to list codes): 8e
Changed system type of partition 1 to 8e (Linux LVM)
Command (m for help): t
Partition number (1-5): 5
Hex code (type L to list codes): 8e
Changed system type of partition 5 to 8e (Linux LVM)
Command (m for help): p
Disk /dev/sdb: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            1         487      3911796    8e   Linux LVM
/dev/sdb2          488        1044     4474102+     5   Extended
/dev/sdb5          488        1044     4474071    8e   Linux LVM
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
# partprobe
```

1. 作为演示，本例创建了一个主分区，大小为**4G**，和一个基于扩展分区的逻辑分区，其大小也为**4G**。在实际环境中每个分区建议使用 **40G** 左右的大小。
2. 磁盘划分为主分区还是逻辑分区对 **LVM** 来说无关紧要。

创建物理卷

要将分区分配给物理卷可以使用 **pvcreate** [<http://www.proxyserve.net/index.php?q=aHR0cDovL21hbi5jeC9wdmNyZWZ0ZQ%3D%3D>] 命令，其格式为：

```
pvcreate <磁盘或分区设备名>
```

下面将在刚刚创建的两个 **LVM** 分区上创建物理卷。

```
# pvcreate /dev/sdb1
Physical volume "/dev/sdb1" successfully created
# pvcreate /dev/sdb5
Physical volume "/dev/sdb5" successfully created
```

创建的物理卷之后可以使用 **pvscan** [<http://www.proxyserve.net/index.php?q=aHR0cDovL21hbi5jeC9wdmNjYW4%3D>] 命令查看物理卷对应的设备名称。

```
# pvscan
PV /dev/sda2   VG VolGroup00   lvm2 [7.88 GB / 0    free]   # 此为安装系统时创建的物理卷
PV /dev/sdb1   VG               lvm2 [3.73 GB]
PV /dev/sdb5   VG               lvm2 [4.27 GB]
Total: 3 [15.87 GB] / in use: 1 [7.88 GB] / in no VG: 2 [8.00 GB]
```

还可以使用 **pvdisplay** [<http://www.proxyserve.net/index.php?q=aHR0cDovL21hbi5jeC9wdmRpc3BsYXk%3D>] 命令查看新创建的物理卷的状态和大小，其命令格式为：

```
pvdisplay [<物理卷设备名>]
```

```
# pvdisplay
--- Physical volume ---                                # 此为安装系统时创建的物理卷
PV Name                /dev/sda2
VG Name                VolGroup00
PV Size                7.90 GB / not usable 23.41 MB
Allocatable            yes (but full)
PE Size (KByte)        32768
Total PE               252
Free PE                0
Allocated PE           252
PV UUID                wy2NBg-eLiF-0QC6-DX1a-20MA-NE3I-pK3J4f
--- NEW Physical volume ---
PV Name                /dev/sdb1
VG Name
PV Size                3.73 GB
Allocatable            NO
PE Size (KByte)        0
Total PE               0
```

```

Free PE                0
Allocated PE           0
PV UUID                JxaDHJ-P4YC-0WeS-pEe0-rEG9-Pq5S-tVNtrE
--- NEW Physical volume ---
PV Name                /dev/sdb5
VG Name
PV Size                4.27 GB
Allocatable            NO
PE Size (KByte)        0
Total PE               0
Free PE                0
Allocated PE           0
PV UUID                E0ui8b-J5qS-3aMd-hwlH-roAu-GY8r-BORHIm

```

使用物理卷创建卷组VG

要使用物理卷创建卷组，可以使用 `vgcreate` [<http://www.proxyserve.net/index.php?q=aHR0cDovL21hbi5jeC92Z2NyZWFOZQ%3D%3D>] 命令，其格式为：

```
vgcreate <卷组名> <物理卷设备名> [<物理卷设备名> ...]
```

要使用上面创建的2个物理卷创建名为 `wwwVG` 的卷组，使用如下的命令：

```
# vgcreate wwwVG /dev/sdb1 /dev/sdb5
Volume group "wwwVG" successfully created
```

卷组创建之后，可以使用 `vgdisplay` [<http://www.proxyserve.net/index.php?q=aHR0cDovL21hbi5jeC92Z2Rpc3BsYXk%3D>] 命令查看卷组的状态，命令格式为：

```
vgdisplay [<卷组名>]
```

```
# vgdisplay
--- Volume group ---
VG Name                wwwVG
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                 7.99 GB
PE Size                  4.00 MB
Total PE                 2046
Alloc PE / Size          0 / 0
Free PE / Size           2046 / 7.99 GB
VG UUID                 RZI9pk-r0ix-u8p8-ieu2-W67A-Ujfo-Z1jNq0
--- Volume group ---
VG Name                VolGroup00          # 此为安装系统时创建卷组

```

```
System ID
Format                lvm2
Metadata Areas        1
Metadata Sequence No  4
VG Access              read/write
VG Status              resizable
MAX LV                0
Cur LV                3
Open LV                3
Max PV                0
Cur PV                1
Act PV                1
VG Size                7.88 GB
PE Size                32.00 MB
Total PE              252
Alloc PE / Size        252 / 7.88 GB
Free PE / Size          0 / 0
VG UUID                p3q80x-Gxrf-qGH1-G0kq-91Gu-ZefK-F8r0tb
```

在卷组VG中创建逻辑卷LV

要在卷组中创建逻辑卷需要使用 `lvcreate` [<http://www.proxyservice.net/index.php?q=aHR0cDovL21hbi5jeC9sdmNyZWZlZDQ%3D%3D>] 命令，格式为：

```
格式1: lvcreate <-l/--size 逻辑卷大小[kKmMgGtT]> <-n/--name 逻辑卷名称> <卷组名称>
格式2: lvcreate <-l/--extents LogicalExtentsNumber> <-n/--name 逻辑卷名称> <卷组名称>
```

1. 格式1：在创建逻辑卷时直接指定大小，单位可以是 kK、mM、gG、tT，默认为 M。
2. 格式2：在创建逻辑卷时使用 `LogicalExtentsNumber` 指定大小，`LogicalExtentsNumber` 可以通过如下命令或得：

```
# vgdisplay|grep "Free PE"
Free PE / Size        2046 / 8.00 GB
Free PE / Size          1 / 32.00 MB
```

上面的 2046 就是卷组 `wwwVG` 当前可用的最大 `LogicalExtentsNumber`。

使用 LVM 的最大好处在于“按需分配、动态调整”。通常在创建逻辑卷时只分配卷组中的部分空间给逻辑卷，剩余空间可以创建其他的逻辑卷，或留作日后再分配。

下面的命令用于在 `wwwVG` 卷组中创建大小为 2G，名字为 `www` 的逻辑卷：

```
# lvcreate -L 2G -n www wwwVG
Logical volume "www" created
```

若将卷组中的所有空间都分配给逻辑卷 `www`，可以使用如下命令：

```
# lvcreate -l 2047 -n www wwwVG
Logical volume "www" created
```

然后可以使用 `lvdisplay` [<http://www.proxyservice.net/index.php?q=aHR0cDovL21hbi5jeC9sdmRpc3BsYXk%3D>] 命令查看刚创建的逻辑卷的信息：

```
# lvdisplay
--- Logical volume ---                                # 此为新建的逻辑卷，其他逻辑卷是系统安装过程中创建的
LV Name                /dev/wwwVG/www
VG Name                wwwVG
LV UUID                KL8CNA-e11x-z6bd-QzvU-Gwju-v41b-Nj3YfX
LV Write Access        read/write
LV Status              available
# open                 0
LV Size                2.00 GB
Current LE             512
Segments              1
Allocation             inherit
Read ahead sectors     0
Block device           253:3
--- Logical volume ---
LV Name                /dev/VolGroup00/LogVolRoot
VG Name                VolGroup00
LV UUID                Rzmk2c-KDy1-mjGF-QZKM-d0y7-98c6-o4p8Ig
LV Write Access        read/write
LV Status              available
# open                 1
LV Size                3.97 GB
Current LE             127
Segments              1
Allocation             inherit
Read ahead sectors     0
Block device           253:0
--- Logical volume ---
LV Name                /dev/VolGroup00/LogVolHome
VG Name                VolGroup00
LV UUID                q6j2U1-g5u9-inAj-zWNy-hrK7-T71j-N59hMZ
LV Write Access        read/write
LV Status              available
# open                 1
LV Size                2.91 GB
Current LE             93
Segments              1
Allocation             inherit
Read ahead sectors     0
Block device           253:1
--- Logical volume ---
LV Name                /dev/VolGroup00/LogVolSWAP
VG Name                VolGroup00
LV UUID                oZ01Rm-fh3g-u7Ij-GsMc-PpkS-DugJ-Bku7DF
LV Write Access        read/write
LV Status              available
# open                 1
LV Size                1.00 GB
Current LE             32
Segments              1
Allocation             inherit
Read ahead sectors     0
Block device           253:2
```

从上面的显示可以看出新建的逻辑卷的设备名是 `/dev/wwwVG/www`，它是使用 `lvcreate` 命令创建该逻辑卷时创建的。创建逻辑卷时，LVM 系统还在 `/dev/mapper/` 目录下创建了该逻辑卷的控制设备，将逻辑卷映射到创建它的物理卷。如下所示：

```
# ls -l /dev/wwwVG/
total 0
lrwxrwxrwx 1 root root 21 Dec 15 18:08 www -> /dev/mapper/wwwVG-www
# ls -l /dev/mapper/
total 0
crw----- 1 root root 10, 63 Dec 15 16:25 control
brw-rw---- 1 root disk 253, 1 Dec 15 16:25 VolGroup00-LogVolHome
brw-rw---- 1 root disk 253, 0 Dec 15 16:25 VolGroup00-LogVolRoot
brw-rw---- 1 root disk 253, 2 Dec 15 16:25 VolGroup00-LogVolSWAP
brw-rw---- 1 root disk 253, 3 Dec 15 18:08 wwwVG-www
```

创建了逻辑卷之后使用 **vgdisplay** 命令查看卷组状态的变化。

```
# vgdisplay
--- Volume group ---
VG Name                wwwVG
System ID
Format                 lvm2
Metadata Areas         2
Metadata Sequence No   2
VG Access               read/write
VG Status               resizable
MAX LV                 0
Cur LV                 1
Open LV                0
Max PV                 0
Cur PV                 2
Act PV                 2
VG Size                 7.99 GB
PE Size                 4.00 MB
Total PE                2046
Alloc PE / Size        512 / 2.00 GB
Free PE / Size          1534 / 5.99 GB
VG UUID                RZl9pk-r0ix-u8p8-ieu2-W67A-Ujfo-Z1jNq0
--- Volume group ---
VG Name                VolGroup00
System ID
Format                 lvm2
Metadata Areas         1
Metadata Sequence No   4
VG Access               read/write
VG Status               resizable
MAX LV                 0
Cur LV                 3
Open LV                3
Max PV                 0
Cur PV                 1
Act PV                 1
VG Size                 7.88 GB
PE Size                 32.00 MB
Total PE                252
Alloc PE / Size        252 / 7.88 GB
Free PE / Size          0 / 0
VG UUID                p3q80x-Gxrf-qGH1-G0kq-91Gu-ZefK-F8r0tb
```

从上面的输出可以看出在卷组 **wwwVG** 中使用了 **512** 个物理区域 (PE)，即分配了 **2G** 给 **www** 逻辑卷，并且还有

1534 的 PE (6.0G) 剩余空间。

创建了逻辑卷之后使用 **pvdisplay** 命令查看物理卷的状态变化。

```
# pvdisplay
--- Physical volume ---
PV Name                /dev/sdb1
VG Name                wwwVG
PV Size                3.73 GB / not usable 4.11 MB
Allocatable            yes
PE Size (KByte)        4096
Total PE               954
Free PE                954
Allocated PE           0
PV UUID                JxaDHJ-P4YC-0WeS-pEe0-rEG9-Pq5S-tVNtrE
--- Physical volume ---
PV Name                /dev/sdb5
VG Name                wwwVG
PV Size                4.27 GB / not usable 1.21 MB
Allocatable            yes
PE Size (KByte)        4096
Total PE               1092
Free PE                580
Allocated PE           512
PV UUID                E0ui8b-J5qS-3aMd-hwlH-roAu-GY8r-BORHIm
--- Physical volume ---
PV Name                /dev/sda2
VG Name                VolGroup00
PV Size                7.90 GB / not usable 23.41 MB
Allocatable            yes (but full)
PE Size (KByte)        32768
Total PE               252
Free PE                0
Allocated PE           252
PV UUID                wy2NBg-eLiF-0QC6-DX1a-20MA-NE3I-pK3J4f
```

从上面的输出可以看出，2G 的 **www** 逻辑卷被分配到了物理卷 **/dev/sdb5** 上，**Allocated PE** 的值从原来的 0 变为了现在的 512。

在逻辑卷LV上创建文件系统FS

要在系统中使用逻辑卷，下面的任务就是在其上创建文件系统。有关文件系统的详细内容请参见下一章。

使用如下命令创建 **ext3** 类型的文件系统。

```
# mkfs -t ext3 /dev/wwwVG/www
mke2fs 1.39 (29-May-2006)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
262144 inodes, 524288 blocks
26214 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=536870912
16 block groups
```



```
32768 blocks per group, 32768 fragments per group
16384 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
This filesystem will be automatically checked every 39 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

然后编辑 **/etc/fstab**，在文件末添加如下的行，以便每次启动系统时自动挂装此文件系统。

```
-----
| /dev/wwwVG/www /srv/www ext3 defaults 0 0 |
|-----|
```

为了使其在当前生效，需要执行如下命令：

```
# mkdir /srv/www
# mount -a
```

- 显示源文件
- 登录