Linux 系统下创建 VG

一【实验目标】

- 学习并掌握 Linux 系统下创建 VG

二【实验环境】

- 实验机环境：Centos 6.6
- 目标机环境： Centos 6.6
- 实验拓扑：如图 1 所示。

图 1 实验拓扑

三【实验原理】

在 Linux 系统下，我们往往会遇到扩充磁盘的情况。普通情况下需要新加一块盘，重分区、格式化、数据复制、卸载就分区、挂载新分区等繁琐的步骤。其实，我们可以在安装系统时使用 LVM 来管理我们的文件系统，这样就可以弹性调整文件系统的容量。

四【实验步骤】

1、新加磁盘启动系统后，查看现有磁盘使用情况

(1)命令：df –h

```
[uroot@localhost ~]$ df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup-lv_root
                       50G  3.3G   44G   7% /
tmpfs                 497M   84K  497M   1% /dev/shm
/dev/sda1             477M   29M  424M   7% /boot
/dev/mapper/VolGroup-lv_home
                       12G   30M   11G   1% /home
```

图 2

(2)命令：fdisk –l

我们发现系统已经有一个 1T 的磁盘 sdb，这个磁盘共有 1305 个柱面，每个柱面大小是 121597，但是还没对其分区。

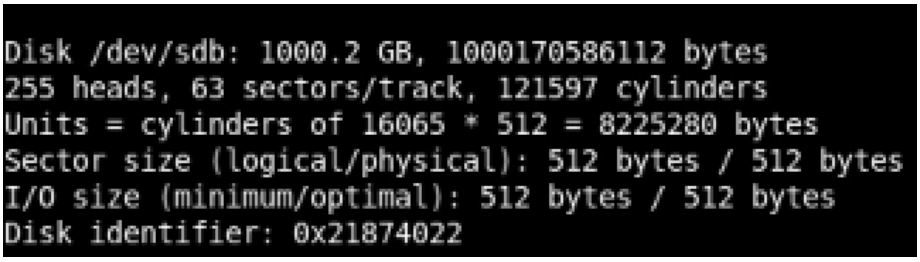

图 3

2、对磁盘 sdb 进行分区

如果不知道 fdisk 里面的具体操作，可输入 m 进行帮助。最常用的是 n（新建）d（删除）p（打印）q（退出）t（修改系统标识符）w（写入并退出）。

命令：fdisk /dev/sdb

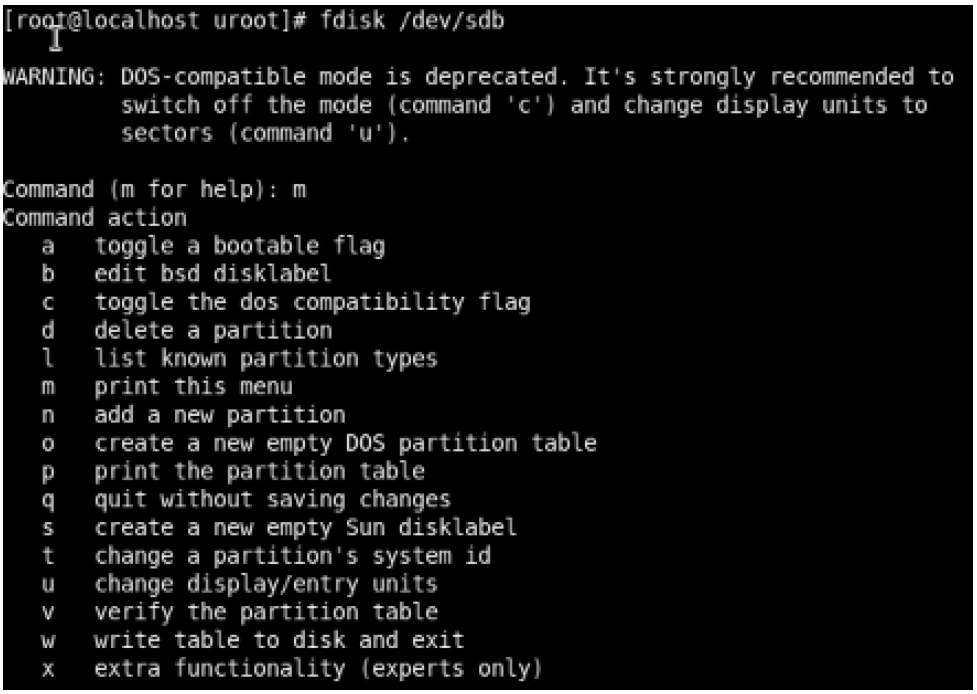

图 4

(1) 建立新的分区

输入 **p** 打印现有分区情况（还没有分区）

输入 **n** 新建分区

输入 **p** 为建立主分区（此时的 p 是在 n 后的，不是打印）

输入 **1** 为建立第一个主分区

分区起始位置可以直接回车，默认是 1

分区最后位置为 **650**（因为每个柱面约 8M，650 柱面约是 5G，本实验只 5G，剩余的做增加 LV 实验用）

输入 **p** 打印分区情况，发现已建立一个分区 **/dev/sdb1**，但是 此分区为 Linux 格式

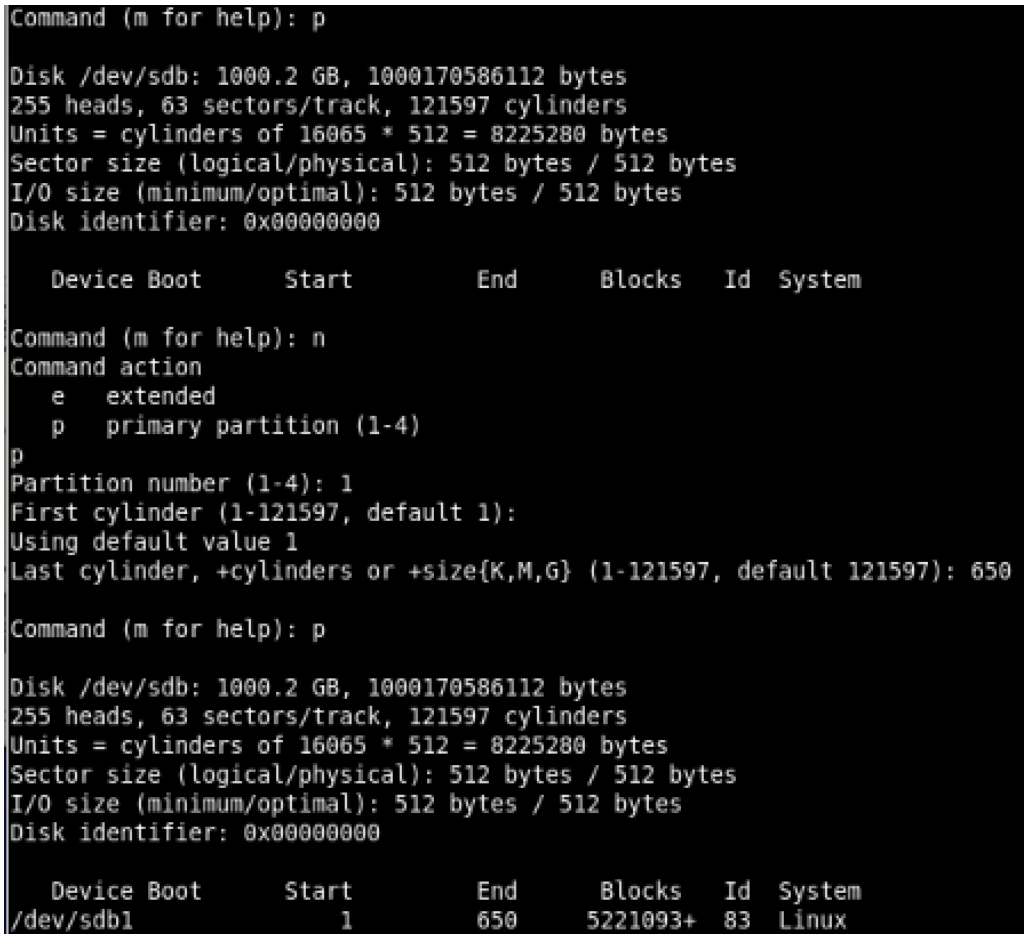

图 5

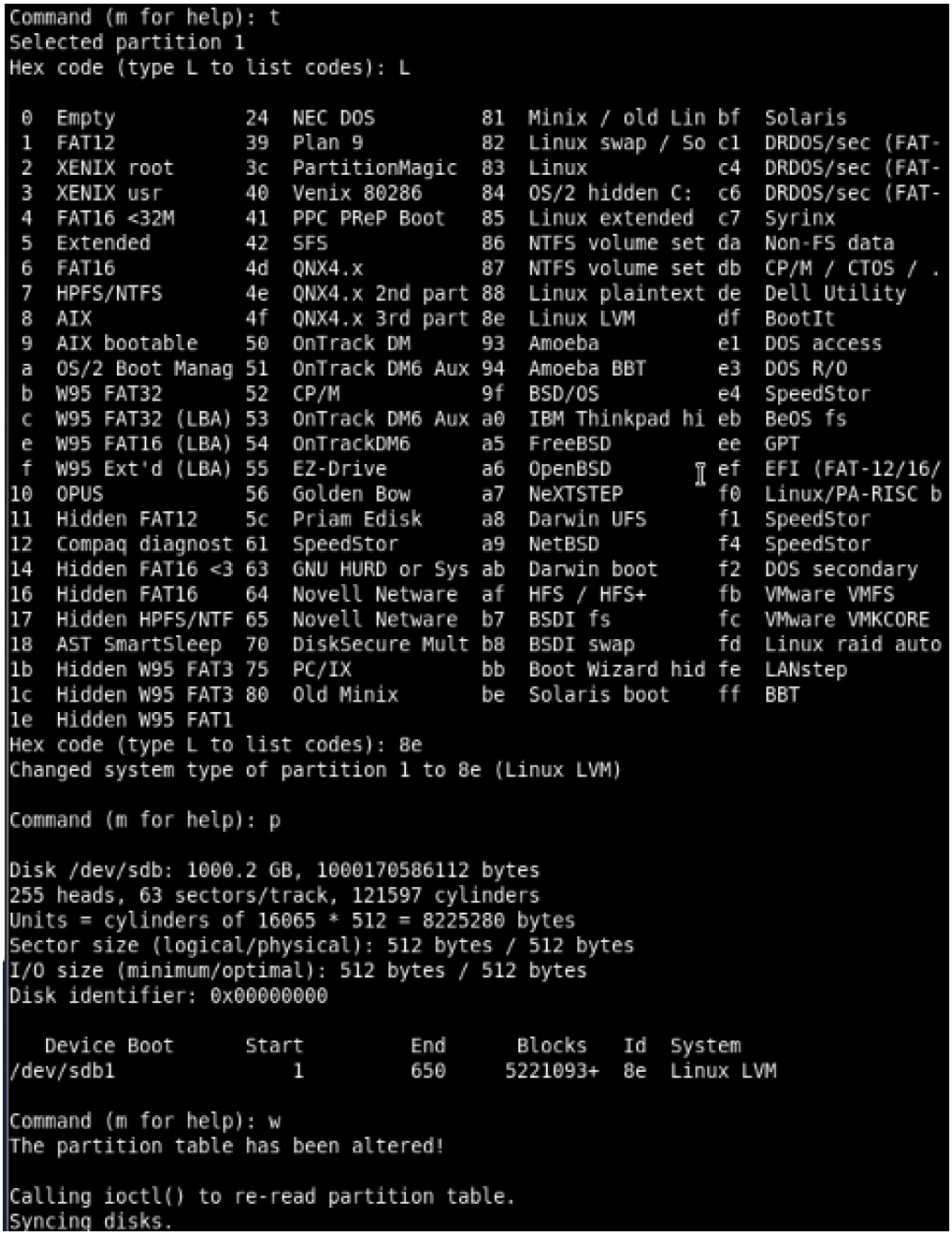

图 6

(2)改变系统标识符：

输入 t 改变分区 1 的属性

输入 L 查看有个属性对应的命令

输入 8e 改变分区 1 为 Linux LVM 格式

输入 p 打印分区情况，发现建立的分区 /dev/sdb1 为 Linux LVM 格式

tip:再次使用 **fdisk -l** 查看系统内磁盘情况发现 /dev/sdb 上已有一个 Linux LVM 格式的 **/dev/sdb1** 分区

```
Disk /dev/sdb: 1000.2 GB, 1000170586112 bytes
255 heads, 63 sectors/track, 121597 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1               1         650     5221093+  8e  Linux LVM
```

图 7

(3)使 kernel 重新读取分区表

命令：**partprobe**

对于 /dev/sda 的警告不予理会

```
[root@centos6 ~]# partprobe
Warning: WARNING: the kernel failed to re-read the partition table on /dev/sda (
Device or resource busy).  As a result, it may not reflect all of your changes u
ntil after reboot.                                                    SEAGLE
```

图 8

3、创建 PV:

扫面系统 PV：**pvscan**

创建 PV：**pvcreate /dev/sdb1**

查看 PV：**pvdisplay**

这样我们就创建了一个 4.98G 的 PV，注意 Allocatable 为 NO

```
[root@localhost Desktop]# pvscan
  PV /dev/sda2   VG VolGroup   lvm2 [63.51 GiB / 0    free]
  Total: 1 [63.51 GiB] / in use: 1 [63.51 GiB] / in no VG: 0 [0   ]
[root@localhost Desktop]# pvcreate
  Please enter a physical volume path
  Run `pvcreate --help' for more information.
[root@localhost Desktop]# pvcreate /dev/sdb1
  Physical volume "/dev/sdb1" successfully created
[root@localhost Desktop]# pvscan
  PV /dev/sda2   VG VolGroup   lvm2 [63.51 GiB / 0    free]
  PV /dev/sdb1                 lvm2 [4.98 GiB]
  Total: 2 [68.49 GiB] / in use: 1 [63.51 GiB] / in no VG: 1 [4.98 GiB]
[root@localhost Desktop]# pvdisplay
  --- Physical volume ---
  PV Name               /dev/sda2
  VG Name               VolGroup
  PV Size               63.51 GiB / not usable 3.00 MiB
  Allocatable           yes (but full)
  PE Size               4.00 MiB
  Total PE              16258
  Free PE               0
  Allocated PE          16258
  PV UUID               4bR5c3-h4YE-cPlI-Djep-zJnQ-86SD-zRADRy

  "/dev/sdb1" is a new physical volume of "4.98 GiB"
  --- NEW Physical volume ---
  PV Name               /dev/sdb1
  VG Name
  PV Size               4.98 GiB
  Allocatable           NO
  PE Size               0
  Total PE              0
  Free PE               0
  Allocated PE          0
  PV UUID               6UaEvL-8CTs-JFRB-9Rix-EGiH-77mq-dLKZA0
```

图 9

4、创建 VG:

扫面系统 VG：**vgscan**

创建 VG：**vgcreate vg_test /dev/sdb1**

查看 VG：**vgdisplay**

这样我们就创建了一个 4.98G（**1274** 个 PE，要记住这个数字）的 VG（名字为 **vg_test**）

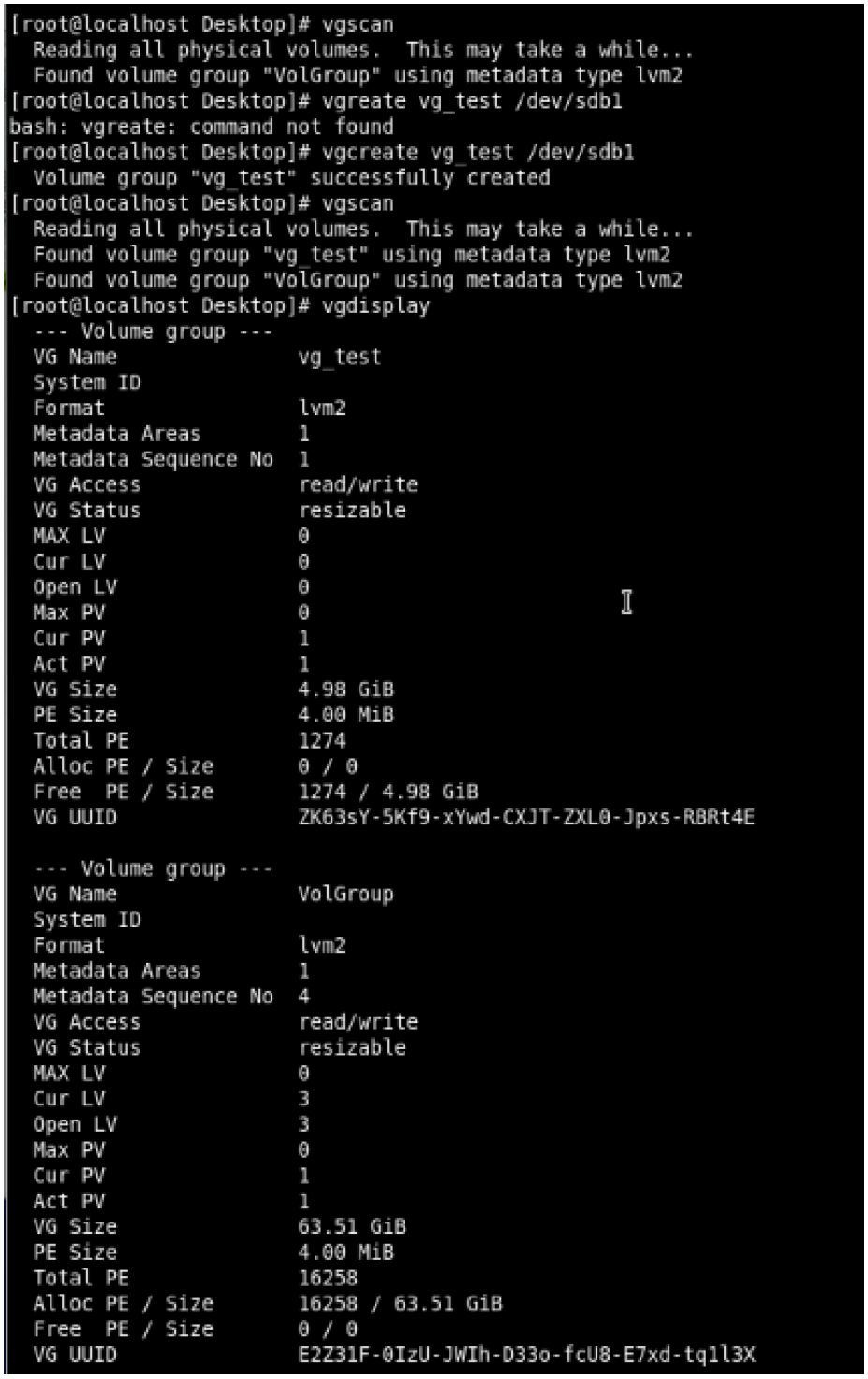

图 10

五【实验思考】

- 使用 fdisk 命令进行分区。
- partprobe, kernel 重新读取分区表。