

Linux下硬件管理(下)

- 本章学习目标
 - 高级磁盘管理
 - 外挂设备的使用



交换分区的查看

- **free** 查看系统内存及交换分区的使用率
- 用法: **free -m** 以兆为单位查看
 - 例如

```

root@RHEL5:~
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[root@RHEL5 ~]# free -m
              total        used        free      shared    buffers
   cached
Mem:          464          459           4           0           7
    269
-/+ buffers/cache:      182        282
Swap:          1027           0        1027
[root@RHEL5 ~]#
    
```

交换分区的增加 3-1

■ 使用虚拟设备生成空文件

➢ **dd if=/dev/zero of=目录/文件名 bs=容量 count=次数**

• 例如

```
root@RHEL5:~  
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
[root@RHEL5 ~]# dd if=/dev/zero of=/tmp/swf1 bs=10M count=10  
10+0 records in  
10+0 records out  
104857600 bytes (105 MB) copied, 5.45828 seconds, 19.2 MB/s
```

■ 生成交换分区文件

➢ **mkswap 目录/文件名**

• 例如

```
root@RHEL5:~  
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
[root@RHEL5 ~]# mkswap /tmp/swf1  
Setting up swapspace version 1, size = 104853 kB  
[root@RHEL5 ~]#
```

交换分区的增加 3-2

■ 激活交换分区

➢ swapon 目录/文件名

● 例如

```
root@RHEL5:~
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[root@RHEL5 ~]# swapon /tmp/swf1
[root@RHEL5 ~]#
```

■ 查看

➢ free -m

```
root@RHEL5:~
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[root@RHEL5 ~]# free -m
```

	total	used	free	shared	buffers
Mem:	464	455	9	0	8
cached	272				
-/+ buffers/cache:		174	289		
Swap:	1127	0	1127		

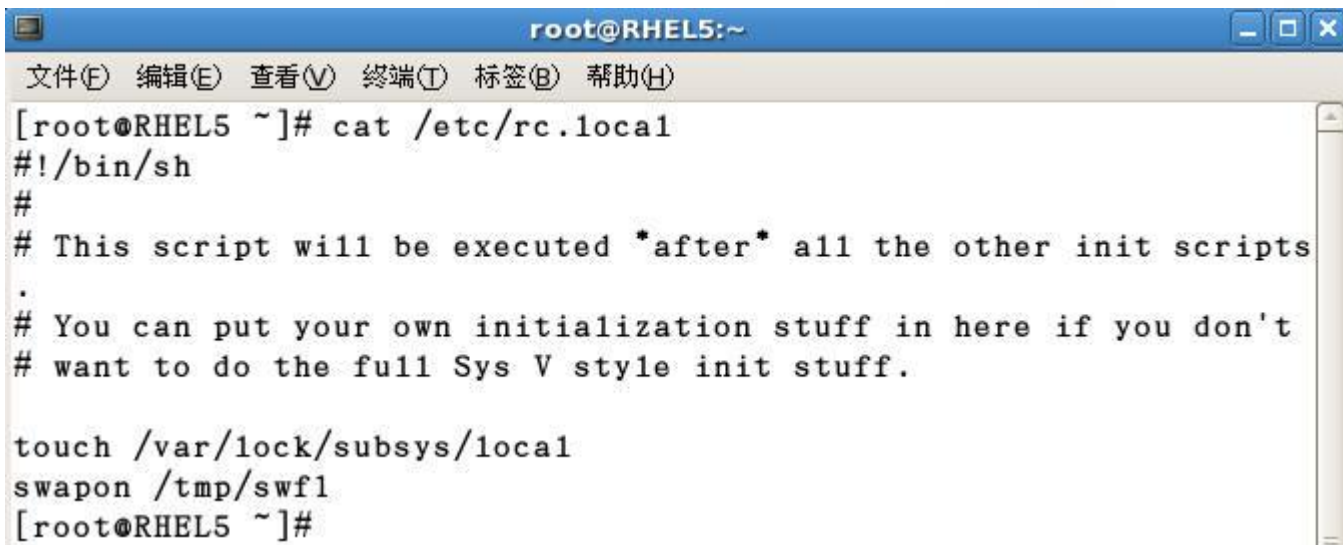
```
[root@RHEL5 ~]#
```

交换分区的增加 3-3

■ 交换分区永久生效

- 在文件`/etc/rc.local`中添加一行 **swapon** 目录/文件名

- 例如



```
root@RHEL5:~  
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
[root@RHEL5 ~]# cat /etc/rc.local  
#!/bin/sh  
#  
# This script will be executed "after" all the other init scripts  
.  
# You can put your own initialization stuff in here if you don't  
# want to do the full Sys V style init stuff.  
  
touch /var/lock/subsys/local  
swapon /tmp/swfl  
[root@RHEL5 ~]#
```

■ 重新启动计算机即生效

配置raid磁盘

■ 查看磁盘

➤ fdisk -l

```
root@RHEL5:~
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[root@RHEL5 ~]# fdisk -l /dev/sdb

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           13       104391    83  Linux
/dev/sdb2            14           26       104422    83  Linux
/dev/sdb3            27           39       104422    83  Linux
/dev/sdb4            40          1044      8072662     5  Extended
/dev/sdb5            40           52       104391    83  Linux
/dev/sdb6            53           65       104391    83  Linux
/dev/sdb7            66           78       104391    83  Linux
/dev/sdb8            79           91       104391    83  Linux
/dev/sdb9            92          104       104391    83  Linux
/dev/sdb10          105          117       104391    83  Linux
/dev/sdb11          118          130       104391    83  Linux
/dev/sdb12          131          143       104391    83  Linux
/dev/sdb13          144          156       104391    83  Linux
/dev/sdb14          157          169       104391    83  Linux
```

配置raid磁盘

■ 创建

- **mdadm -C** raid设备名 **-l**raid级别 **-n**raid磁盘数 **-x**raid备份磁盘数 磁盘1 磁盘2 磁盘3 磁盘4

- 例如

```
#mdadm -C /dev/md_d0 -l5 -n3 -x1 /dev/sdb1 /dev/sdc1 /dev/sdd1 /dev/sde1
```

■ 查看状态

```
#mdadm --detail /dev/md_d0
```

使用raid设备

■ 格式化

➤ mkfs.ext3 raid设备名

```
root@RHEL5:~  
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
[root@RHEL5 ~]# mkfs.ext3 /dev/md0  
mke2fs 1.39 (29-May-2006)  
Filesystem label=  
OS type: Linux  
Block size=1024 (log=0)  
Fragment size=1024 (log=0)  
52208 inodes, 208640 blocks  
10432 blocks (5.00%) reserved for the super user  
First data block=1  
Maximum filesystem blocks=67371008  
26 block groups  
8192 blocks per group, 8192 fragments per group  
2008 inodes per group  
Superblock backups stored on blocks:  
    8193, 24577, 40961, 57345, 73729, 204801  
  
Writing inode tables: done  
Creating journal (4096 blocks): done  
Writing superblocks and filesystem accounting information: done  
  
This filesystem will be automatically checked every 24 mounts or  
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```


使用raid设备

■ 挂载

➤ mount raid设备名 挂载点

```

root@RHEL5:~
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[root@RHEL5 ~]# mount /dev/md0 /mulu1/
[root@RHEL5 ~]# df
文件系统              1K-块          已用        可用  已用%  挂载点
/dev/sda2             12892796     3925424     8301880    33%  /
/dev/sda1             124427       11284      106719    10%  /boot
tmpfs                 237656         0      237656     0%  /dev/shm
/dev/md0              202047        5920      185695     4%  /mulu1
[root@RHEL5 ~]# mount
/dev/sda2 on / type ext3 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/sda1 on /boot type ext3 (rw)
tmpfs on /dev/shm type tmpfs (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw)
/dev/md0 on /mulu1 type ext3 (rw)
    
```

raid设备的修复

■ 标记损坏设备

➢ mdadm raid设备 -f 损坏设备

```
root@RHEL5:~  
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
[root@RHEL5 ~]# mdadm /dev/md0 -f /dev/sdb1  
mdadm: set /dev/sdb1 faulty in /dev/md0  
[root@RHEL5 ~]#
```

■ 查看状态

➢ more /proc/mdstat

```
root@RHEL5:~  
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
[root@RHEL5 ~]# more /proc/mdstat  
Personalities : [raid6] [raid5] [raid4]  
md0 : active raid5 sdb3[2] sdb2[1] sdb1[3] (F)  
      208640 blocks level 5, 64k chunk, algorithm 2 [3/2] [_UU]  
  
unused devices: <none>  
[root@RHEL5 ~]#
```

raid设备的修复

- 添加新设备到阵列中并查看状态
 - mdadm raid设备 -a 新设备

```
root@RHEL5:~  
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
[root@RHEL5 ~]# mdadm /dev/md0 -a /dev/sdb5  
mdadm: added /dev/sdb5  
[root@RHEL5 ~]# more /proc/mdstat  
Personalities : [raid6] [raid5] [raid4]  
md0 : active raid5 sdb5[0] sdb3[2] sdb2[1] sdb1[3] (F)  
      208640 blocks level 5, 64k chunk, algorithm 2 [3/3] [UUU]  
  
unused devices: <none>  
[root@RHEL5 ~]#
```

raid设备的修复

■ 移出损坏的设备并查看

➢ mdadm raid设备 -r 标记损坏的设备

```
root@RHEL5:~  
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
[root@RHEL5 ~]# mdadm /dev/md0 -r /dev/sdb1  
mdadm: hot removed /dev/sdb1  
[root@RHEL5 ~]# more /proc/mdstat  
Personalities : [raid6] [raid5] [raid4]  
md0 : active raid5 sdb5[0] sdb3[2] sdb2[1]  
      208640 blocks level 5, 64k chunk, algorithm 2 [3/3] [UUU]  
  
unused devices: <none>  
[root@RHEL5 ~]#
```

生成配置文档，否则重启之后需要重新配置

```
# mdadm --detail --scan >/etc/mdadm.conf
```


LVM逻辑卷管理器

■ 更改分区ID值为lvm

➢ fdisk 磁盘

/dev/sdb10	105	117	104391	8e	Linux LVM
/dev/sdb11	118	130	104391	8e	Linux LVM
/dev/sdb12	131	143	104391	8e	Linux LVM
/dev/sdb13	144	156	104391	8e	Linux LVM
/dev/sdb14	157	169	104391	8e	Linux LVM

■ 生成物理卷

➢ pvcreate 磁盘分区

```

root@RHEL5:~
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[root@RHEL5 ~]# pvcreate /dev/sdb10
Physical volume "/dev/sdb10" successfully created
[root@RHEL5 ~]# pvcreate /dev/sdb11 /dev/sdb12 /dev/sdb13 /dev/sdb14
Physical volume "/dev/sdb11" successfully created
Physical volume "/dev/sdb12" successfully created
Physical volume "/dev/sdb13" successfully created
Physical volume "/dev/sdb14" successfully created
[root@RHEL5 ~]#
    
```

LVM逻辑卷管理器

■ 生成卷组并查看

➤ vgcreate 卷组名 物理卷1 物理卷2

```
[root@RHEL5 ~]# vgcreate vg1 /dev/sdb10 /dev/sdb11
```

```
Volume group "vg1" successfully created
```

```
[root@RHEL5 ~]# vgsdisplay 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	200.00 MB
PE Size	4.00 MB
Total PE	50
Alloc PE / Size	0 / 0
Free PE / Size	50 / 200.00 MB
VG UUID	143C xv-wZvm-3kxe-Rz1F-ipdF-vBqu-uwab9w



LVM逻辑卷管理器

■ 生成逻辑卷并查看

➤ **lvcreate -L 容量 -n 逻辑卷名 卷组名**

```
[root@RHEL5 ~]# lvcreate -L 100M -n lv1 vg1
Logical volume "lv1" created
[root@RHEL5 ~]# lvsdisplay /dev/vg1/lv1
--- Logical volume ---
LV Name                /dev/vg1/lv1
VG Name                vg1
LV UUID                SJRefn-8sSM-it6X-CJc9-w0yC-9zZY-7rw8hM
LV Write Access        read/write
LV Status              available
# open                 0
LV Size                100.00 MB
Current LE             25
Segments               1
Allocation             inherit
Read ahead sectors     0
Block device           253:0
```

```
[root@RHEL5 ~]# █
```

➤ 注意：卷组完整名称是/dev/卷组名/逻辑卷名

使用逻辑卷

■ 格式化、挂载并查看

```

root@RHEL5:~
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[root@RHEL5 ~]# mkfs.ext3 /dev/vg1/lv1
[root@RHEL5 ~]# mount /dev/vg1/lv1 /tmp
[root@RHEL5 ~]# mount
/dev/sda2 on / type ext3 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/sda1 on /boot type ext3 (rw)
tmpfs on /dev/shm type tmpfs (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw)
/dev/mapper/vg1-lv1 on /tmp type ext3 (rw)
[root@RHEL5 ~]#
    
```

使用Vim编辑器，打开/etc/fstab，在最后一行添加需要挂载的分区LV，指定要挂载的目录（挂载点），ext3分区文件系统格式，其它使用默认即可

卷组的容量的更改

■ 卷组容量增加并查看

➤ vgextend 卷组名 物理卷

```
[root@RHEL5 ~]# vgextend vg1 /dev/sdb12 /dev/sdb13
Volume group "vg1" successfully extended
[root@RHEL5 ~]# vgdisplay vg1
--- Volume group ---
VG Name                vg1
System ID
Format                 lvm2
Metadata Areas         4
Metadata Sequence No   5
VG Access               read/write
VG Status               resizable
MAX LV                 0
Cur LV                 1
Open LV                 0
Max PV                 0
Cur PV                 4
Act PV                 4
VG Size                 400.00 MB
PE Size                 4.00 MB
Total PE                100
Alloc PE / Size         25 / 100.00 MB
Free PE / Size           75 / 300.00 MB
VG UUID                 143Cxx-vZvm-3kxe-Rz1F-ipdF-vBqu-uwab9w
```



卷组容量的更改

■ 卷组容量减少并查看

➤ **vgreduce** 卷组名 物理卷

```
[root@RHEL5 ~]# vgreduce vg1 /dev/sdb13
Removed "/dev/sdb13" from volume group "vg1"
```

```
[root@RHEL5 ~]# 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	1
Open LV	0
Max PV	0
Cur PV	3
Act PV	3
VG Size	300.00 MB
PE Size	4.00 MB
Total PE	75
Alloc PE / Size	25 / 100.00 MB
Free PE / Size	50 / 200.00 MB
VG UUID	143Cxc-wZvm-3kxe-Rz1F-ipdF-vBqu-uwab9w

逻辑卷容量的更改

■ 逻辑卷容量增加

➤ **lvextend -L 容量/+容量 逻辑卷全名**

```
[root@RHEL52 ~]# lvextend -L 320M /dev/vg1/lv1
Extending logical volume lv1 to 320.00 MB
Logical volume lv1 successfully resized
[root@RHEL52 ~]# lvextend -L +20M /dev/vg1/lv1
Extending logical volume lv1 to 340.00 MB
Logical volume lv1 successfully resized
[root@RHEL52 ~]#
```

- **Lvreduce -L 容量 逻辑卷全名**
- 对LV执行容量刷新
- **#resize2fs /dev/VG/LV**
- **# df -h** 查看挂载情况，已经扩容

逻辑卷容量的更改

■ 逻辑卷容量减少

➢ **lvreduce -L 容量/+容量 逻辑卷全名**

```
[root@RHEL52 ~]# lvreduce -L 300M /dev/vg1/lv1
WARNING: Reducing active and open logical volume to 300.00 MB
THIS MAY DESTROY YOUR DATA (filesystem etc.)
Do you really want to reduce lv1? [y/n]: y
Reducing logical volume lv1 to 300.00 MB
Logical volume lv1 successfully resized
[root@RHEL52 ~]# █
```

```
[root@RHEL52 ~]# lvreduce -L -50M /dev/vg1/lv1
Rounding up size to full physical extent 48.00 MB
WARNING: Reducing active and open logical volume to 252.00 MB
THIS MAY DESTROY YOUR DATA (filesystem etc.)
Do you really want to reduce lv1? [y/n]: y
Reducing logical volume lv1 to 252.00 MB
Logical volume lv1 successfully resized
[root@RHEL52 ~]# █
```


使用U盘或者移动硬盘

■ 查看系统是否是别设备（分区表）

➢ fdisk -l

```

Device Boot      Start   End  Blocks  Id System
/dev/sdc1        *          1    125    1006048+  b  W95 FAT32
Partition 1 has different physical/logical endings:
   phys=(1023, _255, 63) logical=(124, 195, 3)
    
```

■ 挂载U盘

➢ mount 磁盘分区 挂载点

```

root@RHEL5:~
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[root@RHEL5 ~]# mount /dev/sdc1 /dir1
[root@RHEL5 ~]#
    
```

■ 卸载U盘

➢ umount 挂载点

```

root@RHEL5:~
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[root@RHEL5 ~]# umount /dir1
[root@RHEL5 ~]#
    
```

总结

- 交换分区的大小如何增加？
- 简述制作Raid1的主要步骤。
- pv, vg, lv是什么关系？



上机实验

- 添加交换分区容量
- 创建raid
- 创建LVM
- 使用u盘



作业

- 增加/删除交换分区
- 创建raid1,raid5
- 添加/删除lvm
- 使用U盘/移动硬盘



课堂提问时间

