

备份、恢复数据分区

1, 查看磁盘挂载状态:

```
[root@post data1]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2       4.8G  2.1G  2.5G  46% /
/dev/sda6       3.0G   69M  2.7G   3% /data1
/dev/sda5       5.8G  141M  5.4G   3% /data2
/dev/sda1      122M   11M  105M  10% /boot
tmpfs           252M     0  252M   0% /dev/shm
```

2, 开始备份磁盘分区, 命令格式为:

dd if=“设备或文件名” of=“备份文件的路径”, 下文示例的磁盘分区为/dev/sda1, 挂载点为/boot, 备份文件设定为/data2/boot.disk;

```
[root@post data1]# dd if=/dev/sda1 of=/data2/boot.disk
256977+0 records in
256977+0 records out
131572224 bytes (132 MB) copied, 1.83272 seconds, 71.8 MB/s
```

备份出来的文件甚至可以用 mount 挂载。

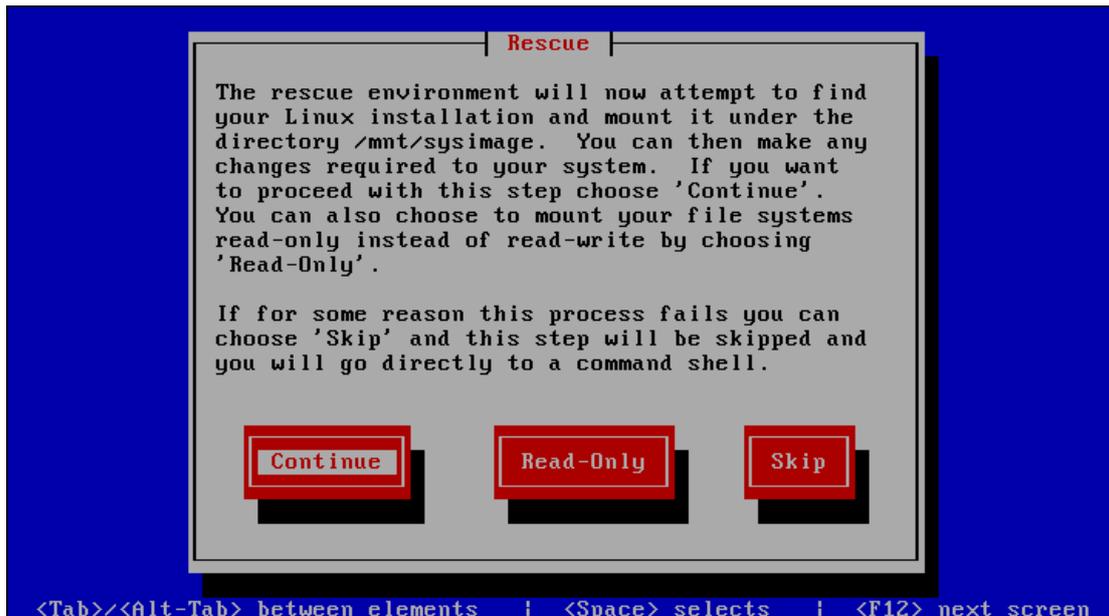
```
[root@post data2]# mount -o loop boot.disk /mnt
[root@post data2]# ll /mnt
total 5195
-rw-r--r-- 1 root root 64551 Oct 11 2007 config-2.6.18-53.e15
drwxr-xr-x 2 root root 1024 Nov 24 15:21 grub
-rw----- 1 root root 2423446 Nov 24 15:21 initrd-2.6.18-53.e15.img
drwx----- 2 root root 12288 Nov 24 23:06 lost+found
-rw-r--r-- 1 root root 87586 Oct 11 2007 symvers-2.6.18-53.e15.gz
-rw-r--r-- 1 root root 903969 Oct 11 2007 System.map-2.6.18-53.e15
-rw-r--r-- 1 root root 1791572 Oct 11 2007 vmlinuz-2.6.18-53.e15
```

然后进入/boot 分区肆无忌惮的乱删一气……, 重启机器, grub 找到 vmlinuz 了……

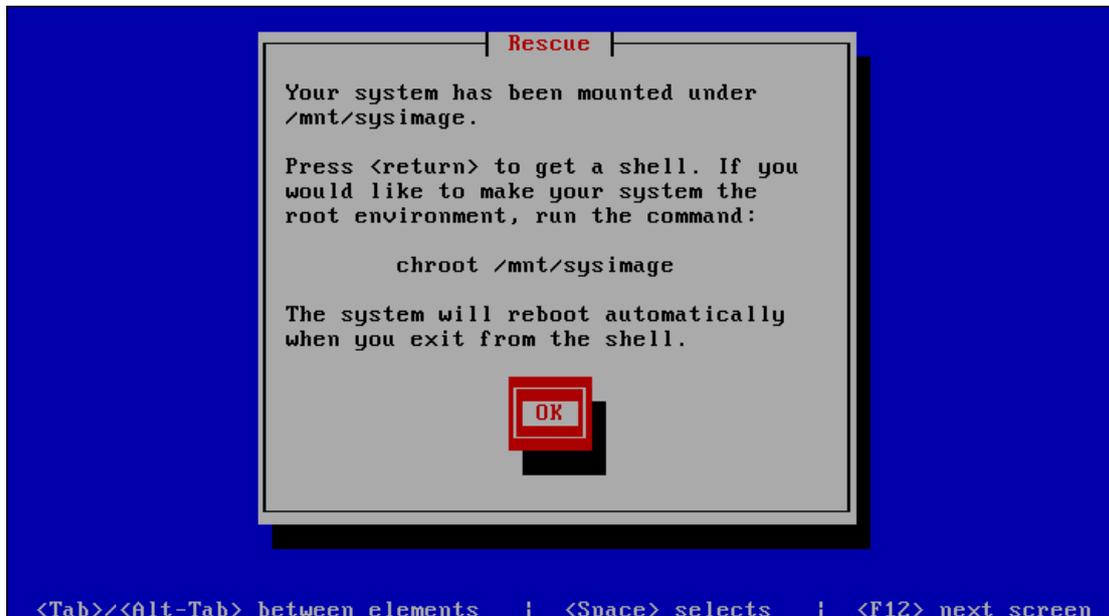
如果是恢复和系统无关的分区, 可以在工作模式下进行恢复, 如果无法启动了, 使用 Linux 启动盘进入救援模式:

```
- To install or upgrade in graphical mode, press the <ENTER> key.
- To install or upgrade in text mode, type: linux text <ENTER>.
- Use the function keys listed below for more information.
[F1-Main] [F2-Options] [F3-General] [F4-Kernel] [F5-Rescue]
boot: linux rescue_
```

切记, 要选择挂载的硬盘为可读写。(如不想在救援模式挂载磁盘请看文尾的小 top)



救援模式下，默认会将硬盘数据挂载到 live CD 的/mnt/sysimage 目录。



这是当前的挂载点信息。

```
sh-3.1# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev            252M   0 252M   0% /dev
/tmp/loop0      82M   82M   0 100% /mnt/runtime
/dev/sda2       4.8G  2.2G  2.4G  48% /mnt/sysimage
/dev/sda6       3.0G  173M  2.6G   7% /mnt/sysimage/data1
/dev/sda5       5.8G  266M  5.3G   5% /mnt/sysimage/data2
/dev/sda1       122M   11M  105M  10% /mnt/sysimage/boot
/dev            252M   0 252M   0% /mnt/sysimage/dev
```

我们可以考虑 chroot，也可以不做。

```
sh-3.1# chroot /mnt/sysimage/
```

现在看看挂载信息。

```
sh-3.1# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2       4.8G  2.2G  2.4G  48% /
/dev/sda6       3.0G  173M  2.6G   7% /data1
/dev/sda5       5.8G  266M  5.3G   5% /data2
/dev/sda1       122M   11M  105M  10% /boot
```

恢复的时候用同样的命令格式，只是源和目标变了：

严重警告！如果你的恢复路径指定错误，比如说指定到/dev/sda2 或者/dev/sda，那一场救援就会变成一场新的灾难，其他磁盘分区的数据会被抹掉!!!

```
sh-3.1# dd if=/data2/boot.disk of=/dev/sda1
dd: writing to '/dev/sda1': Input/output error
256977+0 records in
256976+0 records out
131571712 bytes (132 MB) copied, 13.8728 seconds, 9.5 MB/s
```

磁盘 MBR/分区表的备份恢复。

首先了解下磁盘的结构，前 512K 的内容是 MBR+分区表。

然后再执行 dd 命令做备份时备份的对象是/dev/sda，而非具体分区，另需要加上参数 bs=512 count=1。

bs=BYTES force ibs=BYTES and obs=BYTES

count=BLOCKS copy only BLOCKS input blocks

```
[root@post ~]# dd if=/dev/sda of=/data2/mbrinfo.disk bs=512 count=1
1+0 records in
1+0 records out
512 bytes (512 B) copied, 0.000397585 seconds, 1.3 MB/s
```

备份完成后用 fdisk 删掉/dev/sda1 分区，重启后服务器肯定起不来了。同样用救援模式从光盘引导系统，新引导的系统已经没有/dev/sda1 了。

```
sh-3.1# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev            252M   0  252M   0% /dev
/tmp/loop0      82M   82M   0 100% /mnt/runtime
/dev/sda2       4.8G  2.1G  2.5G  46% /mnt/sysimage
/dev/sda6       3.0G   69M  2.7G   3% /mnt/sysimage/data1
/dev/sda5       5.8G  266M  5.3G   5% /mnt/sysimage/data2
/dev            252M   0  252M   0% /mnt/sysimage/dev
```

可选的环节，chroot

```
sh-3.1# chroot /mnt/sysimage/
```

再次检查磁盘挂载情况；

```
sh-3.1# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2       4.8G  2.1G  2.5G  46% /
/dev/sda6       3.0G   69M  2.7G   3% /data1
/dev/sda5       5.8G  266M  5.3G   5% /data2
```

恢复的时候是同样的命令返回来用，如果是单独恢复分区表，请将后缀参数改成 bs=512 skip=446 count=66；

```
sh-3.1# dd if=/data2/mbrinfo.disk of=/dev/sda bs=512 count=1
1+0 records in
1+0 records out
512 bytes (512 B) copied, 0.0380975 seconds, 13.4 kB/s
```

重启服务器，就会加载新的分区表并正常启动计算机，被我通过 fdisk 删除的/boot 分区又出

现出现了。如果还有磁盘的损坏，请重启后继续进入救援模式继续修复。

生产应用中的小技巧

1, 利用 jobs 队列将 dd 扔到后台执行;

如果我们早有打算，在执行 dd 命令的结尾加上 “&” 就直接将 dd 扔到了后台执行，但很多情况是当前窗口开始 dd 一段时间以后，我们突然想断开 shell 了。这种情况重新 dd 是挺可惜的，可以用下面的方法来将运行中的 dd 扔到后台执行：

A. 首先按 ctrl+z，暂停 dd 的执行，回到 shell 里。

```
[root@caotest-1 ~]# dd if=/dev/xvda2 of=/root/testdd.disk
[1]+  Stopped                  dd if=/dev/xvda2 of=/root/testdd.disk
[root@caotest-1 ~]#
```

B. 执行 jobs，看看被你暂停掉的 dd 命令的 job id 是多少。如果之前你没有 ctrl+z 暂停过其他任务，则其 job id 为 1。

```
[root@caotest-1 ~]# jobs
[1]+  Stopped                  dd if=/dev/xvda2 of=/root/testdd.disk
```

C. 使用 bg + % {job id} 可以让暂停的进程在后台执行，使用 fg + % {job id} 可以让暂停的进程在前台执行。下面的实验中我们肯定是用后台执行了，执行完成后再看下 jobs，状态已经变成 running 了。

```
[root@caotest-1 ~]# bg %1
[1]+ dd if=/dev/xvda2 of=/root/testdd.disk &
[root@caotest-1 ~]# jobs
[1]+  Running                  dd if=/dev/xvda2 of=/root/testdd.disk &
```

请注意上图中的 “1” 是 job id，dd 进程的运行状态是 running 了。

2, 如果在后台运行的 dd，我们想看看它的执行进度该怎么看哪？其实 dd 的 help 里已经提到了这个功能了：

```
[root@caotest-1 ~]# dd --help
.....
Sending a USR1 signal to a running `dd' process makes it
print I/O statistics to standard error and then resume copying.
```

英文我们看不太懂，那就看看实际执行的方法和结果吧。

首先我们用 ps 命令找到 dd 的 pid，然后将 pid 写到这个命令里就可以了

命令是：kill -USR1 {dd-pid}。

从输出内容里我们可以看到，现在已经复制了多少数据，以用了多少时间，当前的速度是多少。

```
[root@caotest-1 ~]# ps -ef|grep -v grep|grep dd
root      4728   4114  7 00:12 pts/0    00:00:02 dd if /dev/xvda2 of /root/testdd.disk
[root@caotest-1 ~]# kill -USR1 4728
[root@caotest-1 ~]# ps -ef|grep -v grep|grep dd
root      4728   4114  7 00:12 pts/0    00:00:02 dd if /dev/xvda2 of /root/testdd.disk
3459296+0 records out
1771159552 bytes (1.8 GB) copied, 48.9213 seconds, 36.2 MB/s
```

3, 救援模式默认会挂载磁盘内系统的根分区，而且无法强制卸载，这个时候需要在引导 linux 的时候就加上一个参数，进入救援模式的命令变成：

```
linux rescue nomount
```