

1、通过 tar 将/var/log 目录进行归档并分别实现最低和最高级别的压缩比，归档文件保存在用户目录/home/user 下，记录、观察过程和结果；

将/var/log 目录归档，归档文件并保存在用户目录/home/user 下，如图 1-1

```
[root@server user]# tar -cvf whl.tar /var/log
tar: 从成员名中删除开头的 "/"
/var/log/
/var/log/spooler-20180910
/var/log/yum.log
/var/log/tallylog
/var/log/secure-20180827
/var/log/messages-20180907
/var/log/secure
/var/log/cron-20180910
/var/log/maillog-20180910
/var/log/btmp
/var/log/boot.log-20180903
/var/log/maillog
/var/log/secure-20180907
/var/log/messages
/var/log/cron
/var/log/maillog-20180907
/var/log/boot.log
/var/log/cron-20180827
/var/log/dmesg.old
/var/log/rhsm/
/var/log/firewalld
/var/log/boot.log-20180913
/var/log/secure-20180910
/var/log/cron-20180907
/var/log/boot.log-20180908
/var/log/tuned/
/var/log/tuned/tuned.log
/var/log/spooler
/var/log/grubby_prune_debug
/var/log/maillog-20180827
/var/log/audit/
/var/log/audit/audit.log
```

图 1-1

查看打包文件是否存在，如图 1-2

```
[root@server user]# ls
text4add.txt  text4all.txt  user2.txt  user4.txt  user6.txt
text4add.txt~ user1.txt  user3.txt  user5.txt  whl.tar
[root@server user]#
```

图 1-2

实现最低级别的压缩比，如图 1-3。

```
[root@server user]# gzip -1 whl.tar
[root@server user]# ls
text4add.txt  text4all.txt  user2.txt  user4.txt  user6.txt
text4add.txt~ user1.txt  user3.txt  user5.txt  whl.tar.gz
[root@server user]# ls -l
总用量 812
-rw-rw---- 1 root user 4 9月 11 10:34 text4add.txt
-rw-rw---- 1 user1 user 0 9月 11 10:20 text4add.txt~
-rw-rw---- 1 root root 0 9月 10 18:55 text4all.txt
-rw-rw-r-- 1 user1 user1 4 9月 10 18:45 user1.txt
-rw-rw-r-- 1 user2 user2 0 9月 10 18:40 user2.txt
-rw-rw-r-- 1 user3 user3 4 9月 10 18:50 user3.txt
-rw-rw-r-- 1 user4 user4 0 9月 10 18:27 user4.txt
-rw-rw-r-- 1 user5 user5 0 9月 10 18:28 user5.txt
-rw-rw-r-- 1 user6 user6 0 9月 10 18:28 user6.txt
-rw-r--r-- 1 root root 817001 9月 13 09:02 whl.tar.gz
[root@server user]#
```

图 1-3

实现最高级别的压缩比，如图 1-4。

```
[root@server user]# gzip -9 whl.tar
[root@server user]# ls -l
总用量 644
-rw-rw---- 1 root user      4 9月 11 10:34 text4add.txt
-rw-rw---- 1 user1 user      0 9月 11 10:20 text4add.txt~
-rw-rw---- 1 root root      0 9月 10 18:55 text4all.txt
-rw-rw-r-- 1 user1 user1     4 9月 10 18:45 user1.txt
-rw-rw-r-- 1 user2 user2     0 9月 10 18:40 user2.txt
-rw-rw-r-- 1 user3 user3     4 9月 10 18:50 user3.txt
-rw-rw-r-- 1 user4 user4     0 9月 10 18:27 user4.txt
-rw-rw-r-- 1 user5 user5     0 9月 10 18:28 user5.txt
-rw-rw-r-- 1 user6 user6     0 9月 10 18:28 user6.txt
-rw-r--r-- 1 root root 645601 9月 13 09:02 whl.tar.gz
```

图 1-4

2、通过 cpio 解开/boot/initramfs-x.xxx-xxx.img 文件，并将揭开内容与当前根目录下的结构进行一级子目录数量对比，记录过程和结果。

复制/boot/ initramfs-x.xxx-xxx.img 文件到/root/666,如图 2-1。

```
[root@server 666]# cp /boot/initramfs-3.10.0-693.el7.x86_64.img /root/666
[root@server 666]# ls
initramfs-3.10.0-693.el7.x86_64.img
[root@server 666]# mv initramfs-3.10.0-693.el7.x86_64.img initramfs-3.10.0-693.el7.x86_64.gz
[root@server 666]# ls
initramfs-3.10.0-693.el7.x86_64.gz
```

图 2-1

将揭开内容与当前根目录下的内容对比，如图 2-2。

```
[root@server 666]# file initramfs-3.10.0-693.el7.x86_64.gz
initramfs-3.10.0-693.el7.x86_64.gz: ASCII cpio archive (SVR4 with no CRC)
[root@server 666]# mv initramfs-3.10.0-693.el7.x86_64.gz initramfs-3.10.0-693.el7.x86_64.img
[root@server 666]# cpio -i < initramfs-3.10.0-693.el7.x86_64.img
75536 块
[root@server 666]# ls
bin      initramfs-3.10.0-693.el7.x86_64.img  root      sys      var
dev      lib                                  run      sysroot
etc      lib64                              sbin     tmp
init     proc                              shutdown usr
[root@server 666]# ls /
aquota.group  boot  home  lost+found  opt  run  sys  var
aquota.user   dev  lib  media      proc sbin tmp
bin           etc  lib64 mnt        root srv  usr
[root@server 666]#
```

图 2-2

3、为用户目录/home/user 挂载镜像 lv，并记录操作过程，破坏主卷并实施救援，记录操作观察结果；

添加四块磁盘，vd[e-h]如图 3-1。

```
[root@server ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sr0         11:0    1 1024M  0 rom
vda         253:0    0  100G  0 disk
├─vda1      253:1    0    1G  0 part /boot
├─vda2      253:2    0  95.1G  0 part /
└─vda3      253:3    0   3.9G  0 part [SWAP]
vdb         253:16   0   20G  0 disk
└─vdb1      253:17   0   20G  0 part
vde         253:64   0   20G  0 disk
vdf         253:80   0   20G  0 disk
vdg         253:96   0   20G  0 disk
vdh         253:112  0   20G  0 disk
```

图 3-1

创建 pv，如图 3-2

```
[root@server ~]# pvcreate /dev/vd[e-h]
Physical volume "/dev/vde" successfully created.
Physical volume "/dev/vdf" successfully created.
Physical volume "/dev/vdg" successfully created.
Physical volume "/dev/vdh" successfully created.
```

图 3-2

创建带有镜像能力的 vg，如图 3-3

```
[root@server ~]# vgcreate vg_mirror /dev/vd[e-f]
Volume group "vg_mirror" successfully created
```

图 3-3

创建带有镜像能力的 lv，如图 3-4。

```
[root@server ~]# lvcreate -L 4G -m1 -n lv_mirror vg_mirror
Logical volume "lv_mirror" created.
```

图 3-4

格式化成带有镜像能力的 ext4，如图 3-5。

```
tmpfs tmpfs 550M 0 550M 0% /run/user/70
[root@server ~]# mkfs.ext4 /dev/vg_mirror/lv_mirror
mke2fs 1.42.9 (28-Dec-2013)
文件系统标签=
OS type: Linux
块大小=4096 (log=2)
分块大小=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
262144 inodes, 1048576 blocks
52428 blocks (5.00%) reserved for the super user
第一个数据块=0
Maximum filesystem blocks=1073741824
32 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: 完成
正在写入inode表: 完成
Creating journal (32768 blocks): 完成
Writing superblocks and filesystem accounting information: 完成
```

图 3-5

为用户目录/home/user 挂载镜像 lv，如图 3-6。

```
[root@server ~]# mount /dev/vg_mirror/lv_mirror /home/user
[root@server ~]# df -hT
```

文件系统	类型	容量	已用	可用	已用%	挂载点
/dev/vda2	ext4	94G	1.3G	88G	2%	/
devtmpfs	devtmpfs	2.0G	0	2.0G	0%	/dev
tmpfs	tmpfs	2.0G	0	2.0G	0%	/dev/shm
tmpfs	tmpfs	2.0G	8.5M	2.0G	1%	/run
tmpfs	tmpfs	2.0G	0	2.0G	0%	/sys/fs/cgroup
/dev/vda1	xfs	1014M	179M	836M	18%	/boot
tmpfs	tmpfs	396M	0	396M	0%	/run/user/0
/dev/mapper/vg_mirror-lv_mirror	ext4	3.9G	16M	3.6G	1%	/home/user

图 3-6

对/dev/vde 进行破坏，如图 3-7。

```
[root@server ~]# dd if=/dev/zero of=/dev/vde count=10
记录了10+0 的读入
记录了10+0 的写出
5120字节(5.1 kB)已复制, 0.000314043 秒, 16.3 MB/秒
[root@server ~]# lvs -a -o +devices
WARNING: Device for PV hChNtt-y8u4-mwSp-ka3X-7Hg2-pdFe-zG00jY not found or rejected by a filter.
WARNING: Couldn't find all devices for LV vg_mirror/lv_mirror_rimage_0 while checking used and assumed devices.
WARNING: Couldn't find all devices for LV vg_mirror/lv_mirror_rmeta_0 while checking used and assumed devices.
LV VG Attr LSize Pool Origin Data% Meta% Move Log Cpy%Sync Convert Devices
lv_mirror vg_mirror rwi-aor-p- 4.00g 100.00 lv_mirror_rimage_0(0),lv_mirror_rimage_1(0)
[lv_mirror_rimage_0] vg_mirror iwi-aor-p- 4.00g [unknown](1)
[lv_mirror_rimage_1] vg_mirror iwi-aor--- 4.00g /dev/vdf(1)
[lv_mirror_rmeta_0] vg_mirror ewi-aor-p- 4.00m [unknown](0)
[lv_mirror_rmeta_1] vg_mirror ewi-aor--- 4.00m /dev/vdf(0)
[root@server ~]#
```

图 3-7

恢复数据，如图 3-8。

```
[root@server ~]# pvcreate /dev/vde
WARNING: Not using lvmetad because a repair command was run.
Physical volume "/dev/vde" successfully created.
[root@server ~]# vgextend vg_mirror /dev/vde
WARNING: Not using lvmetad because a repair command was run.
Volume group "vg_mirror" successfully extended
[root@server ~]# lvconvert -m1 /dev/vg_mirror/lv_mirror /dev/vdf
WARNING: Not using lvmetad because a repair command was run.
Are you sure you want to convert linear LV vg_mirror/lv_mirror to raid1 with 2 images enhancing resilience? [y/n]: y
Insufficient suitable allocatable extents for logical volume : 1025 more required
[root@server ~]# lvdisplay
WARNING: Not using lvmetad because a repair command was run.
--- Logical volume ---
LV Path                /dev/vg_mirror/lv_mirror
LV Name                 lv_mirror
VG Name                 vg_mirror
LV UUID                 8FRsut-GCwh-36sU-ap8C-d3u8-4ncx-E9dPkx
LV Write Access         read/write
LV Creation host, time  server, 2018-09-13 14:58:36 +0800
LV Status                available
# open                  1
LV Size                 4.00 GiB
Current LE              1024
Segments                1
Allocation               inherit
Read ahead sectors      auto
- currently set to      8192
Block device            252:4
```

图 3-8

4、为用户目录/home/user 挂载 raid5 卷，记录操作过程，随机破坏阵列中一块硬盘并实施救援，记录操作观察结果；

创建 raid 设备 md5,RAID 级别 5, 可用设备 3 个, 热备设备 1 个. 分别为/dev/vde,/dev/vdg,/dev/vdh,/dev/vdi,如图 4-1。

```
[root@server ~]# mdadm -C /dev/md5 -a yes -l 5 -n 3 -x 1 /dev/vd{e,g,h,i}
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md5 started.
```

图 4-1

格式化 RAID 阵列，如图 4-2。

```
[root@server ~]# mkfs.ext4 /dev/md5mk2fs 1.42.9 (28-Dec-2013)
文件系统标签=
OS type: Linux
块大小=4096 (log=2)
分块大小=4096 (log=2)
Stride=128 blocks, Stripe width=256 blocks
2621440 inodes, 10477568 blocks
523878 blocks (5.00%) reserved for the super user
第一个数据块=0
Maximum filesystem blocks=2157969408
320 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624

Allocating group tables: 完成
正在写入inode表: 完成
Creating journal (32768 blocks): 完成
Writing superblocks and filesystem accounting information: 完成
```

图 4-2

查看 RAID 的创建过程，如图 4-3。

```
@server:~  
/dev/md5:  
Version : 1.2  
Creation Time : Thu Sep 13 18:41:01 2018  
Raid Level : raid5  
Array Size : 41910272 (39.97 GiB 42.92 GB)  
Used Dev Size : 20955136 (19.98 GiB 21.46 GB)  
Raid Devices : 3  
Total Devices : 4  
Persistence : Superblock is persistent  
  
Update Time : Thu Sep 13 18:41:27 2018  
State : clean, degraded, recovering  
Active Devices : 2  
Working Devices : 4  
Failed Devices : 0  
Spare Devices : 2  
  
Layout : left-symmetric  
Chunk Size : 512K  
Consistency Policy : resync  
Rebuild Status : 16% complete  
  
Name : server:5 (local to host server)  
UUID : a57b7088:a25e8702:19757453:8432c361  
Events : 7  
  
Number Major Minor RaidDevice State  
0 253 64 0 active sync /dev/vde  
1 253 96 1 active sync /dev/vdg  
4 253 112 2 spare rebuilding /dev/vdh  
3 253 128 - spare /dev/vdi
```

图 4-3

为用户目录/home/user 挂载 raid5 卷，如图 4-4

```
[root@server ~]# mount /dev/md5 /home/user/  
[root@server ~]# df -hThTh  
文件系统 类型 容量 已用 可用 已用% 挂载点  
/dev/vda2 ext4 94G 1.3G 88G 2% /  
devtmpfs devtmpfs 2.0G 0 2.0G 0% /dev  
tmpfs tmpfs 2.0G 0 2.0G 0% /dev/shm  
tmpfs tmpfs 2.0G 8.5M 2.0G 1% /run  
tmpfs tmpfs 2.0G 0 2.0G 0% /sys/fs/cgroup  
/dev/vda1 xfs 1014M 179M 836M 18% /boot  
tmpfs tmpfs 396M 0 396M 0% /run/user/0  
/dev/md5 ext4 40G 49M 38G 1% /home/user  
[root@server ~]#  
[root@server ~]#  
[root@server ~]# df -hT  
文件系统 类型 容量 已用 可用 已用% 挂载点  
/dev/vda2 ext4 94G 1.3G 88G 2% /  
devtmpfs devtmpfs 2.0G 0 2.0G 0% /dev  
tmpfs tmpfs 2.0G 0 2.0G 0% /dev/shm  
tmpfs tmpfs 2.0G 8.5M 2.0G 1% /run  
tmpfs tmpfs 2.0G 0 2.0G 0% /sys/fs/cgroup  
/dev/vda1 xfs 1014M 179M 836M 18% /boot  
tmpfs tmpfs 396M 0 396M 0% /run/user/0  
/dev/md5 ext4 40G 49M 38G 1% /home/user
```

图 4-4

破坏阵列中一块硬盘/dev/vdg

```
/dev/md5:
  Version : 1.2
  Creation Time : Thu Sep 13 18:41:01 2018
  Raid Level : raid5
  Array Size : 41910272 (39.97 GiB 42.92 GB)
  Used Dev Size : 20955136 (19.98 GiB 21.46 GB)
  Raid Devices : 3
  Total Devices : 4
  Persistence : Superblock is persistent

  Update Time : Thu Sep 13 19:00:51 2018
  State : clean, degraded, recovering
  Active Devices : 2
  Working Devices : 4
  Failed Devices : 0
  Spare Devices : 2

  Layout : left-symmetric
  Chunk Size : 512K
Consistency Policy : resync

  Rebuild Status : 38% complete

    Name : server:5 (local to host server)
    UUID : a57b7088:a25e8702:19757453:8432c361
    Events : 34

  Number   Major   Minor   RaidDevice State
    0       253     64      0     active sync   /dev/vde
    3       253    128      1     spare rebuilding /dev/vdi
    4       253    112      2     active sync   /dev/vdh
    5       253     96      -     spare   /dev/vdg

[root@server ~]#
```