

Linux 文件管理实验

实验报告



姓名

班级

学号

电话

Email

日期

一、 实验目的

熟练掌握 Linux 操作系统的使用，掌握 Linux 的系统的进程管理和文件管理功能。

二、 实验内容

1. 将若干已有用户加入到同一个组 `xjtuse` 中。在 `/home` 下创建一个共享的公用目录 `public`，允许 `xjtuse` 组中的用户对该目录具有读写和执行操作。（给出相关命令及运行结果）
2. 对于 `public` 目录下的文件，只有文件的拥有者才具有删除文件的权限。（给出相关命令及运行结果）
3. 对于 `public` 目录下的文件，也可以通过路径 `/mnt/public` 来访问。（给出相关命令及运行结果）
4. 看 Linux 系统磁盘空间的使用情况（给出显示结果），并为/分区创建磁盘配额,使得用户可用空间的软限制为 **100M**,硬限制为 **150M**，且每个用户可用的 `inodes` 的软限制为 **100**，硬限制为 **120**。并对磁盘配额情况进行验证测试。（给出相关命令及运行结果）

三、 题目分析及基本设计过程分析

1. 可以使用 `useradd` 指令来添加用户、`groupadd` 指令来添加组、`usermod` 指令将用户加入到组。利用 `mkdir` 指令创建目录、`chmod` 指令更改目录权限、`chown` 指令设置目录所有者。
2. 为确保 `/home/public` 目录中的文件只能由文件的拥有者删除，需要在 `public` 目录上设置粘滞位，启用粘滞位后，只有文件的拥有者才能

删除该目录下的文件。调用 `touch` 命令分别让两个用户创建文件，调用 `rm` 指令尝试让其中一个用户删除这两个文件，若粘滞位成功启用，该用户只能删除自己创建的文件，删除其他用户创建的文件时系统会进行提示并无法删除该文件（除非调用 `rf` 强制删除）。

3. 调用 `mount -bind` 指令使该目录在 `/mnt/public` 可用。
4. 调用 `df` 指令来查看磁盘使用情况。配置 `/etc/fstab` 文件中的条目，为用户级别的限额输入关键字 `usrquota`、为组限额输入关键字 `grpquota` 以激活文件系统的限额，再输入 `mount -o remount /` 指令将文件系统重新装入根分区中，使用 `quotacheck` 命令初始化限额系统。调用 `edquota` 指令来设置用户限额，完成后可调用 `repquota` 指令查看报告。

四、 运行截图和相关说明

1. 创建三个新用户 `filetest1`、`filetest2` 和 `filetest3`，将用户 `filetest1` 和 `filetest2` 加入 `xjtuse` 组，并使用 `chmod` 和 `chown` 赋予用户组读写权限。之后切换到 `filetest1` 和 `filetest2`，可以正常访问并添加目录 `filetest1` 和 `filetest2`，而未在组内的 `filetest3` 不能。

创建用户和组（`xjtuse` 组在之前的实验中已创建过）：

```
[xhwang@bogon ~]$ su -  
Password:  
[root@bogon ~]# useradd filetest1  
[root@bogon ~]# useradd filetest2  
[root@bogon ~]# useradd filetest3  
[root@bogon ~]# groupadd xjtuse  
groupadd: group xjtuse exists
```

赋予权限：

```
[root@bogon ~]# usermod -g xjtuse filetest1
[root@bogon ~]# usermod -g xjtuse filetest2
[root@bogon ~]# chmod 770 /home/public
[root@bogon ~]# chown :xjtuse /home/public
```

创建文件并赋予组成员读、写、执行权限：

```
[root@bogon ~]# su filetest1
[filetest1@bogon root]$ cd /home/public
[filetest1@bogon public]$ mkdir filetest1
[filetest1@bogon public]$ ls
filetest1 public-test

[filetest1@bogon public]$ su -
Password:
[root@bogon ~]# su filetest2
[filetest2@bogon root]$ cd /home/public
[filetest2@bogon public]$ mkdir filetest2
[filetest2@bogon public]$ ls
filetest1 filetest2 public-test

[root@bogon ~]# chmod 770 /home/public/filetest1.txt

[root@bogon ~]# chmod 770 /home/public/filetest2
```

验证 xjtuse 组内的 filetest1 和 filetest2 用户是否对/home/public 具有读写权限（“r”、“w”、“x”分别表示组成员具有读、写和执行权限）：

```
[root@bogon ~]# ls -l /home/public
total 12
drwxrwx--- 2 filetest1 xjtuse 4096 Oct 28 10:04 filetest1
drwxrwx--- 2 filetest2 xjtuse 4096 Oct 28 10:06 filetest2
-rwxrwxrwx 1 root      root      0 Oct 27 19:18 public-test
```

查看不在 xjtuse 组内的 filetest3 用户是否对/home/public 具有读写权限：

```
[filetest2@bogon public]$ su -
Password:
[root@bogon ~]# su filetest3
[filetest3@bogon root]$ cd /home/public
bash: cd: /home/public: Permission denied
```

2. 通过 chmod 指令为 /home/public 目录添加粘滞位:

```
[root@bogon ~]# ls -ld /home/public
drwxrwx--T 4 filetest1 xjtuse 4096 Oct 28 10:06 /home/public
```

调用 touch 指令让 filetest1 用户创建文件 filetest1.txt, 并让 filetest2 用户创建文件 filetest2.txt, 让 filetest2 用户尝试删除文件 filetest1.txt 和文件 filetest2.txt, 发现文件 filetest2.txt 顺利删除, 文件 filetest1.txt 删除失败:

```
[filetest1@bogon root]$ ls /home/public
filetest1 filetest2 public-test
[filetest1@bogon root]$ touch /home/public/filetest1.txt
[filetest1@bogon root]$ ls /home/public
filetest1 filetest1.txt filetest2 public-test
[filetest1@bogon root]$ su -
Password:
[root@bogon ~]# su filetest2
[filetest2@bogon root]$ touch /home/public/filetest2.txt
[filetest2@bogon root]$ rm /home/public/filetest1.txt
rm: remove write-protected regular empty file `/home/public/filetest1.txt'?
[filetest2@bogon root]$ rm /home/public/filetest2.txt
[filetest2@bogon root]$ ls /home/public
filetest1 filetest1.txt filetest2 public-test
```

3. 执行 mount --bind 指令, 执行后可以在 /mnt/public 访问到 xjtuse-test:

```
[root@bogon ~]# ls /home/public
filetest1 filetest1.txt filetest2 public-test
[root@bogon ~]# mkdir /mnt/public
[root@bogon ~]# mount --bind /home/public /mnt/public
[root@bogon ~]# ls /mnt/public
filetest1 filetest1.txt filetest2 public-test
```

4. 使用 df 指令查看磁盘使用情况:

```
[root@bogon ~]# df
Filesystem            1K-blocks    Used Available Use% Mounted on
/dev/sda2              17981340   2079656   14973532  13% /
/dev/sda1              295561     16010     264291    6% /boot
tmpfs                  517536      0       517536    0% /dev/shm
/dev/scd0              2935370   2935370      0 100% /media/RHEL_5.4 i386 DVD
[root@bogon ~]# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/sda2             18G   2.0G   15G   13% /
/dev/sda1             289M   16M   259M    6% /boot
tmpfs                 506M    0   506M    0% /dev/shm
/dev/scd0             2.8G  2.8G    0 100% /media/RHEL_5.4 i386 DVD
```

编辑/etc/fstab 文件:

```
LABEL=/ / ext3 defaults,usrquota,grpquota
ta 1 1
```

调用 mount 指令将文件系统重新装入根分区中，并使用
quotacheck 指令初始化限额系统:

```
[root@bogon ~]# vi /etc/fstab
[root@bogon ~]# mount -o remount /
[root@bogon ~]# quotacheck -avug
quotacheck: Cannot remount filesystem mounted on / re:
ight not be right.
Please stop all programs writing to filesystem or use
[root@bogon ~]# quotacheck -avug -m
quotacheck: Scanning /dev/sda2 [/] done
quotacheck: Checked 8727 directories and 84300 files
```

调用 edquota 指令配置限制:

```
[root@bogon ~]# edquota -u filetest1
```

Disk quotas for user filetest1 (uid 508):					
Filesystem	blocks	soft	hard	inodes	soft
hard					
/dev/sda2	40	100000	150000	11	100
120					

配置后调用 repquota 指令查看报告:

```
[root@bogon ~]# repquota -aug
*** Report for user quotas on device /dev/sda2
Block grace time: 7days; Inode grace time: 7days
```

User	used	Block limits			grace	File limits			grace
		used	soft	hard		used	soft	hard	
root	-- 2425484	0	0		92667	0	0		
daemon	-- 24	0	0		3	0	0		
lp	-- 16	0	0		2	0	0		
ntp	-- 8	0	0		1	0	0		
rpc	-- 4	0	0		1	0	0		
avahi	-- 12	0	0		3	0	0		
smmsp	-- 12	0	0		2	0	0		
rpcuser	-- 8	0	0		1	0	0		
xfs	-- 0	0	0		1	0	0		
xhwang	-- 12356	0	0		254	0	0		
tux	-- 68	0	0		9	0	0		
tux1	-- 60	0	0		8	0	0		
bob	-- 60	0	0		8	0	0		
Alice	-- 60	0	0		8	0	0		
lily	-- 60	0	0		8	0	0		
abc	-- 60	0	0		8	0	0		
tom	-- 68	0	0		9	0	0		
apache	-- 12	0	0		3	0	0		
filetest1	-- 40	100000	150000		11	100	120		
filetest2	-- 32	0	0		9	0	0		
filetest3	-- 28	0	0		8	0	0		

五、 实验中出现的问题和解决

1.在问题 4 中想要调用 mount 指令将 “/” 重新装入根分区时出现下列情况：

```
[root@bogon ~]# mount -o remount
Usage: mount -V          : print version
       mount -h          : print this help
       mount             : list mounted filesystems
       mount -l          : idem, including volume labels
So far the informational part. Next the mounting.
The command is `mount [-t fstype] something somewhere'.
Details found in /etc/fstab may be omitted.
       mount -a [-t|-O] ... : mount all stuff from /etc/fstab
       mount device       : mount device at the known place
       mount directory    : mount known device here
       mount -t type dev dir : ordinary mount command
Note that one does not really mount a device, one mounts
a filesystem (of the given type) found on the device.
One can also mount an already visible directory tree elsewhere:
       mount --bind olddir newdir
or move a subtree:
       mount --move olddir newdir
One can change the type of mount containing the directory dir:
       mount --make-shared dir
       mount --make-slave dir
       mount --make-private dir
       mount --make-unbindable dir
One can change the type of all the mounts in a mount subtree
containing the directory dir:
       mount --make-rshared dir
       mount --make-rslave dir
       mount --make-rprivate dir
       mount --make-runbindable dir
```

解决：发现是指令输入错误，少输入了一个 “/”，正确输入下图所示指令即可成功将 “/” 重新装入根分区：

```
[root@bogon ~]# mount -o remount /
```

2. 在问题 4 中想要使用 quotacheck 指令初始化限额系统时，系统发生以下报错，导致无法进行初始化限额：

```
[root@bogon ~]# quotacheck -avug
quotacheck: Cannot remount filesystem mounted on / read-only so counted values might not be right.
Please stop all programs writing to filesystem or use -m flag to force checking.
```

解决：使用 quotacheck -avug -m 强制进行检查：

```
[root@bogon ~]# quotacheck -avug -m
quotacheck: Scanning /dev/sda2 [/] done
quotacheck: Checked 8727 directories and 84300 files
```

六、 实验体会

本次实验中,我学习并掌握了 Linux 系统中文件管理相关内容的基础知识。在实践过程中,我了解并使用了创建用户、创建组、创建目录、将用户加入组、更改用户权限、设置目录权限、设置目录拥有者、设置粘滞位、mount 指令相关内容、查看磁盘使用情况、创建磁盘配额等内容的相关命令。本次实验课程让我受益匪浅,不仅深化了我对 Linux 系统文件管理相关内容的理解与掌握,还提高了使用 Linux 系统命令行的熟练度,进一步提升我的代码编写能力。