**实验报告四**

姓名：叶倩琳

班级：软工1706班

学号：201706061330

提交日期：2019年12月22日

[实验环境：Ubuntu]

**Lab 10**

1.

yql@DESKTOP-TRMDDFQ:~$

2.

yql@DESKTOP-TRMDDFQ:~$ cat>test

yql@DESKTOP-TRMDDFQ:~$ vi test

i

test

Enter esc

:wq

yql@DESKTOP-TRMDDFQ:~$ cat test

test

3.

yql@DESKTOP-TRMDDFQ:~$ ls -l test

-rw-rw-rw- 1 yql yql 0 Dec 21 10:12 test

ls命令用于显示目前工作目录所含之文件及子目录。

语法：

ls [-alrtAFR] [name...]

参数：

-a 显示所有文件及目录 (将名称开头为"."的视为隐藏档，不会列出)

-l 除文件名称外，亦将文件型态、权限、拥有者、文件大小等资讯详细列出

-r 将文件以相反次序显示(原定依英文字母次序)

-t 将文件依建立时间之先后次序列出

-A 同 -a ，但不列出 "." (目前目录) 及 ".." (父目录)

-F 在列出的文件名称后加一符号；例如可执行档则加 "\*", 目录则加 "/"

-R 若目录下有文件，则以下之文件亦皆依序列出

4.

root@DESKTOP-TRMDDFQ:~# chown user1 test

chown 将指定文件的拥有者改为指定的用户或组，用户可以是用户名或者用户ID；组可以是组名或者组ID；文件是以空格分开的要改变权限的文件列表，支持通配符\*。

使用权限：root

语法：

chown [-cfhvR] [--help] [--version] user[:group] file...

参数：

user 新的文件拥有者的使用者 ID

group 新的文件拥有者的使用者组(group)

-c 显示更改的部分的信息

-f 忽略错误信息

-h 修复符号链接

-v 显示详细的处理信息

-R 处理指定目录以及其子目录下的所有文件

--help 显示辅助说明

--version 显示版本

5.

yql@DESKTOP-TRMDDFQ:~$ umask # 获得当前权限掩码

0000

yql@DESKTOP-TRMDDFQ:~$ umask -S

u=rwx,g=rwx,o=rwx

umask命令指定在建立文件时预设的权限掩码。[权限掩码]是由3个八进制的数字所组成，将现有的存取权限减掉权限掩码后，即可产生建立文件时预设的权限。

语法：

umask [-S][权限掩码]

参数：

-S 　以文字的方式来表示权限掩码

若权限掩码0022四组数字，第一组代表的是特殊权限，先不做讨论。进入目录等操作都是需要目录具有执行权限，而对于文件来说，一般不需要执行权限，所以：如果用户创建的是目录，则默认所有权限都开放，为777，默认为：drwxrwxrwx；如果创建的是文件，默认没有x权限，那么就只有r、w两项，最大值为666，默认为：-rw-rw-rw-。

拿走的权限就是这里这个默认值要减掉的权限，r、w、x分别是4、2、1，要拿掉读权限就输入4，拿掉写权限就输入2，以此类推。umask为022，也就是说，对于当前用户没有拿掉权限，group用户和other用户都被拿走了w权限，所以此时如果用户进行创建目录和文件的时候，默认权限是会进行如下的减法操作：新建文件：666-022=644；新建目录：777-022=755。

6.

yql@DESKTOP-TRMDDFQ:~$ umask # 获得当前权限掩码

0000

yql@DESKTOP-TRMDDFQ:~$ umask

0002

umask的设置很简单，只需要在umask命令后加想要拿掉的权限数字。

7.

yql@DESKTOP-TRMDDFQ:~$ cat>test1

yql@DESKTOP-TRMDDFQ:~$ vi test1

i

This is test1.

Enter esc

:wq

yql@DESKTOP-TRMDDFQ:~$ cat test1

This is test1.

yql@DESKTOP-TRMDDFQ:~$ cat>test2

This is test2.

^C

yql@DESKTOP-TRMDDFQ:~$ cat test2

This is test2.

8.

yql@DESKTOP-TRMDDFQ:~$ cmp test1 test2

test1 test2 differ: byte 13, line 1

yql@DESKTOP-TRMDDFQ:~$ diff test1 test2

1c1 # 1c1表示两个文件都在第一行内容有所不同

< This is test1.

---

> This is test2.

yql@DESKTOP-TRMDDFQ:~$ diff test1 test2 -y -W 50

This is test1. | This is test2.

（1）cmp命令用于比较两个文件是否有差异。当相互比较的两个文件完全一样时，则该指令不会显示任何信息。若发现有所差异，预设会标示出第一个不同之处的字符和列数编号。若不指定任何文件名称或是所给予的文件名为"-"，则cmp指令会从标准输入设备读取数据。

语法：

cmp [-clsv][-i <字符数目>][--help][第一个文件][第二个文件]

参数：

-c或--print-chars 　除了标明差异处的十进制字码之外，一并显示该字符所对应字符。

-i<字符数目>或--ignore-initial=<字符数目> 　指定一个数目。

-l或--verbose 　标示出所有不一样的地方。

-s或--quiet或--silent 　不显示错误信息。

（2）diff命令用于比较文件的差异。diff以逐行的方式，比较文本文件的异同处。如果指定要比较目录，则diff会比较目录中相同文件名的文件，但不会比较其中子目录。

语法：

diff [-abBcdefHilnNpPqrstTuvwy][-<行数>][-C <行数>][-D <巨集名称>][-I <字符或字符串>][-S <文件>][-W <宽度>][-x <文件或目录>][-X <文件>][--help][--left-column][--suppress-common-line][文件或目录1][文件或目录2]

参数：

-<行数> 　指定要显示多少行的文本。此参数必须与-c或-u参数一并使用。

-a或--text 　diff预设只会逐行比较文本文件。

-b或--ignore-space-change 　不检查空格字符的不同。

-B或--ignore-blank-lines 　不检查空白行。

-c 　显示全部内文，并标出不同之处。

-C<行数>或--context<行数> 　与执行"-c-<行数>"指令相同。

-d或--minimal 　使用不同的演算法，以较小的单位来做比较。

-D<巨集名称>或ifdef<巨集名称> 　此参数的输出格式可用于前置处理器巨集。

-e或--ed 　此参数的输出格式可用于ed的script文件。

-f或-forward-ed 　输出的格式类似ed的script文件，但按照原来文件的顺序来显示不同处。

-H或--speed-large-files 　比较大文件时，可加快速度。

-l<字符或字符串>或--ignore-matching-lines<字符或字符串> 　若两个文件在某几行有所不同，而这几行同时都包含了选项中指定的字符或字符串，则不显示这两个文件的差异。

-i或--ignore-case 　不检查大小写的不同。

-l或--paginate 　将结果交由pr程序来分页。

-n或--rcs 　将比较结果以RCS的格式来显示。

-N或--new-file 　在比较目录时，若文件A仅出现在某个目录中，预设会显示：

Only in目录：文件A若使用-N参数，则diff会将文件A与一个空白的文件比较。

-p 　若比较的文件为C语言的程序码文件时，显示差异所在的函数名称。

-P或--unidirectional-new-file 　与-N类似，但只有当第二个目录包含了一个第一个目录所没有的文件时，才会将这个文件与空白的文件做比较。

-q或--brief 　仅显示有无差异，不显示详细的信息。

-r或--recursive 　比较子目录中的文件。

-s或--report-identical-files 　若没有发现任何差异，仍然显示信息。

-S<文件>或--starting-file<文件> 　在比较目录时，从指定的文件开始比较。

-t或--expand-tabs 　在输出时，将tab字符展开。

-T或--initial-tab 　在每行前面加上tab字符以便对齐。

-u,-U<列数>或--unified=<列数> 　以合并的方式来显示文件内容的不同。

-v或--version 　显示版本信息。

-w或--ignore-all-space 　忽略全部的空格字符。

-W<宽度>或--width<宽度> 　在使用-y参数时，指定栏宽。

-x<文件名或目录>或--exclude<文件名或目录> 　不比较选项中所指定的文件或目录。

-X<文件>或--exclude-from<文件> 　您可以将文件或目录类型存成文本文件，然后在=<文件>中指定此文本文件。

-y或--side-by-side 　以并列的方式显示文件的异同之处。

--help 　显示帮助。

--left-column 　在使用-y参数时，若两个文件某一行内容相同，则仅在左侧的栏位显示该行内容。

--suppress-common-lines 　在使用-y参数时，仅显示不同之处。

9.

yql@DESKTOP-TRMDDFQ:~$ whoami

yql

yql@DESKTOP-TRMDDFQ:~$ pwd

/home/yql

yql@DESKTOP-TRMDDFQ:~$ sudo -i

[sudo] password for yql:

root@DESKTOP-TRMDDFQ:~# whoami

root

root@DESKTOP-TRMDDFQ:~# pwd

/root

yql@DESKTOP-TRMDDFQ:~$ su root

Password:

root@DESKTOP-TRMDDFQ:/home/yql# cd

root@DESKTOP-TRMDDFQ:~# whoami

root

root@DESKTOP-TRMDDFQ:~# pwd

/root

（1）sudo命令以系统管理者的身份执行指令，也就是说，经由 sudo 所执行的指令就好像是 root 亲自执行。

使用权限：在 /etc/sudoers 中有出现的使用者。

（2）su命令用于变更为其他使用者的身份，除 root 外，需要键入该使用者的密码。

使用权限：所有使用者。

语法

su [-fmp] [-c command] [-s shell] [--help] [--version] [-] [USER [ARG]]

参数说明：

-f 或 --fast 不必读启动档（如 csh.cshrc 等），仅用于 csh 或 tcsh

-m -p 或 --preserve-environment 执行 su 时不改变环境变数

-c command 或 --command=command 变更为帐号为 USER 的使用者并执行指令（command）后再变回原来使用者

-s shell 或 --shell=shell 指定要执行的 shell （bash csh tcsh 等），预设值为 /etc/passwd 内的该使用者（USER） shell

--help 显示说明文件

--version 显示版本资讯

- -l 或 --login 这个参数加了之后，就好像是重新 login 为该使用者一样，大部份环境变数（HOME SHELL USER等等）都是以该使用者（USER）为主，并且工作目录也会改变，如果没有指定 USER ，内定是 root

USER 欲变更的使用者帐号

ARG 传入新的 shell 参数

10.

root@DESKTOP-TRMDDFQ:~# ll

total 20

drwx------ 1 root root 512 Dec 21 13:39 ./

drwxr-xr-x 1 root root 512 Nov 13 13:34 ../

-rw------- 1 root root 1272 Dec 20 23:45 .bash\_history

-rw-r--r-- 1 root root 3106 Apr 9 2018 .bashrc

-rw-r--r-- 1 root root 148 Aug 17 2015 .profile

-rw------- 1 root root 8251 Dec 20 20:46 .viminfo

-rw-r--r-- 1 root root 0 Dec 21 13:39 a.txt

root@DESKTOP-TRMDDFQ:~# chgrp -v yql a.txt

changed group of 'a.txt' from root to yql

root@DESKTOP-TRMDDFQ:~# ll

total 20

drwx------ 1 root root 512 Dec 21 13:39 ./

drwxr-xr-x 1 root root 512 Nov 13 13:34 ../

-rw------- 1 root root 1272 Dec 20 23:45 .bash\_history

-rw-r--r-- 1 root root 3106 Apr 9 2018 .bashrc

-rw-r--r-- 1 root root 148 Aug 17 2015 .profile

-rw------- 1 root root 8251 Dec 20 20:46 .viminfo

-rw-r--r-- 1 root yql 0 Dec 21 13:39 a.txt

chgrp命令用于变更文件或目录的所属群组。

语法：

chgrp [-cfhRv][--help][--version][所属群组][文件或目录...] 或 chgrp [-cfhRv][--help][--reference=<参考文件或目录>][--version][文件或目录...]

参数：

-c或--changes 效果类似"-v"参数，但仅回报更改的部分。

-f或--quiet或--silent 　不显示错误信息。

-h或--no-dereference 　只对符号连接的文件作修改。

-R或--recursive 　递归处理，将指定目录下的所有文件及子目录一并处理。

-v或--verbose 　显示指令执行过程。

--help 　在线帮助。

--reference=<参考文件或目录> 　把指定文件或目录的所属群组全部设成和参考文件或目录的所属群组相同。

--version 　显示版本信息。

11.

root@DESKTOP-TRMDDFQ:~# ll

total 20

drwx------ 1 root root 512 Dec 21 13:39 ./

drwxr-xr-x 1 root root 512 Nov 13 13:34 ../

-rw------- 1 root root 1272 Dec 20 23:45 .bash\_history

-rw-r--r-- 1 root root 3106 Apr 9 2018 .bashrc

-rw-r--r-- 1 root root 148 Aug 17 2015 .profile

-rw------- 1 root root 8251 Dec 20 20:46 .viminfo

-rw-r--r-- 1 root root 0 Dec 21 13:39 a.txt

root@DESKTOP-TRMDDFQ:~# chown -v yql:yql a.txt

changed ownership of 'a.txt' from root:root to yql:yql

root@DESKTOP-TRMDDFQ:~# ll

total 20

drwx------ 1 root root 512 Dec 21 13:39 ./

drwxr-xr-x 1 root root 512 Nov 13 13:34 ../

-rw------- 1 root root 1272 Dec 20 23:45 .bash\_history

-rw-r--r-- 1 root root 3106 Apr 9 2018 .bashrc

-rw-r--r-- 1 root root 148 Aug 17 2015 .profile

-rw------- 1 root root 8251 Dec 20 20:46 .viminfo

-rw-r--r-- 1 yql yql 0 Dec 21 13:39 a.txt

chown 将指定文件的拥有者改为指定的用户或组，用户可以是用户名或者用户ID；组可以是组名或者组ID；文件是以空格分开的要改变权限的文件列表，支持通配符\*。

使用权限： root

语法：

chown [-cfhvR] [--help] [--version] user[:group] fileanme

参数：

user 新的文件拥有者的使用者 ID

group 新的文件拥有者的使用者组(group)

-c 显示更改的部分的信息

-f 忽略错误信息

-h 修复符号链接

-v 显示详细的处理信息

-R 处理指定目录以及其子目录下的所有文件

12.

root@DESKTOP-TRMDDFQ:~# chown -R yql:yql a.txt

13.

root@DESKTOP-TRMDDFQ:~# cat>a.txt

root@DESKTOP-TRMDDFQ:~# chmod 111 a.txt

root@DESKTOP-TRMDDFQ:~# ll

---x--x--x 1 yql yql 0 Dec 21 13:39 a.txt\*

14.

root@DESKTOP-TRMDDFQ:~# chmod 777 a.txt

root@DESKTOP-TRMDDFQ:~# ll

-rwxrwxrwx 1 yql yql 0 Dec 21 13:39 a.txt\*

语法：

chmod abc file

其中a,b,c各为一个数字，分别表示User、Group、及Other的权限。

r=4，w=2，x=1

若要rwx属性则4+2+1=7；

若要rw-属性则4+2=6；

若要r-x属性则4+1=5。

chmod a=rwx file <==> chmod 777 file

chmod ug=rwx,o=x file <==> chmod 771 file

15.

root@DESKTOP-TRMDDFQ:~# chmod 4777 a.txt

root@DESKTOP-TRMDDFQ:~# ls -l

total 0

-rwxrwxrwx 1 yql yql 0 Dec 21 13:39 a.txt

若用chmod 4755 filename可使此程序具有root的权限

16.

root@DESKTOP-TRMDDFQ:~# mkdir testdir

root@DESKTOP-TRMDDFQ:~# ls

txt testdir

17.

root@DESKTOP-TRMDDFQ:~# chmod 777 testdir

root@DESKTOP-TRMDDFQ:~# ls -l

total 0

-rwsrwxrwx 1 yql yql 0 Dec 21 13:39 a.txt

drwxrwxrwx 1 root root 512 Dec 21 15:16 testdir

root@DESKTOP-TRMDDFQ:~# chmod g+s testdir

18.

root@DESKTOP-TRMDDFQ:~# logout

yql@DESKTOP-TRMDDFQ:~$ su user1

Password:

user1@DESKTOP-TRMDDFQ:/home/yql$ cd

user1@DESKTOP-TRMDDFQ:~$

19.

root@DESKTOP-TRMDDFQ:~# cd testdir

root@DESKTOP-TRMDDFQ:~/testdir# cat>testfile

aaaaaaaaaaa

^C

root@DESKTOP-TRMDDFQ:~/testdir# cat testfile

aaaaaaaaaaa

root@DESKTOP-TRMDDFQ:~/testdir# ls

Testfile

20.

root@DESKTOP-TRMDDFQ:~/testdir# ls -l

total 0

-rw-r--r-- 1 root root 12 Dec 21 15:45 testfile

21.

yql@DESKTOP-TRMDDFQ:~$ su

Password:

root@DESKTOP-TRMDDFQ:/home/yql#

22.

root@DESKTOP-TRMDDFQ:~# mkdir dir1

root@DESKTOP-TRMDDFQ:~# ls -l

total 0

-rwsrwxrwx 1 yql yql 0 Dec 21 13:39 a.txt

drwxr-xr-x 1 root root 512 Dec 21 15:54 dir1

drwxrwsrwx 1 root root 512 Dec 21 15:44 testdir

23.

root@DESKTOP-TRMDDFQ:~# chmod 1777 dir1

root@DESKTOP-TRMDDFQ:~# ls -l

total 0

-rwsrwxrwx 1 yql yql 0 Dec 21 13:39 a.txt

drwxrwxrwt 1 root root 512 Dec 21 15:54 dir1

drwxrwsrwx 1 root root 512 Dec 21 15:44 testdir

设置粘滞位（stickybit）权限，就是在原先的基础上对其他人o加上t粘滞位权限，在查看目录详细信息时可以发现，other其他人后面多了一个t。使其他用户无法删除文件。chmod o+t dir/

24.

root@DESKTOP-TRMDDFQ:~# logout

yql@DESKTOP-TRMDDFQ:~$ su user1

Password:

user1@DESKTOP-TRMDDFQ:/home/yql$ cd

user1@DESKTOP-TRMDDFQ:~$

25.

root@DESKTOP-TRMDDFQ:~# cd dir1

root@DESKTOP-TRMDDFQ:~/dir1# cat>userfile1

26.

root@DESKTOP-TRMDDFQ:~# logout

yql@DESKTOP-TRMDDFQ:~$ su user2

Password:

user2@DESKTOP-TRMDDFQ:/home/yql$ cd

user2@DESKTOP-TRMDDFQ:~$

27.

root@DESKTOP-TRMDDFQ:~# ls -l

total 0

-rwsrwxrwx 1 yql yql 0 Dec 21 13:39 a.txt

drwxrwxrwt 1 root root 512 Dec 21 15:54 dir1

drwxrwsrwx 1 root root 512 Dec 21 15:44 testdir

root@DESKTOP-TRMDDFQ:~# cd dir1

root@DESKTOP-TRMDDFQ:~/dir1# ls

root@DESKTOP-TRMDDFQ:~/dir1# cat>a.txt

Aaaaa

^C

root@DESKTOP-TRMDDFQ:~/dir1# ls

a.txt

root@DESKTOP-TRMDDFQ:~/dir1# su yql

yql@DESKTOP-TRMDDFQ:/root/dir1$ ls

a.txt

yql@DESKTOP-TRMDDFQ:/root/dir1$ vi a.txt

i

Abbb

Enter esc

E45: 'readonly' option is set (add ! to override) # 修改文件失败

yql@DESKTOP-TRMDDFQ:/root/dir1$ rm a.txt

rm: remove write-protected regular file 'a.txt'? Y

rm: cannot remove 'a.txt': Operation not permitted # 删除文件失败

yql@DESKTOP-TRMDDFQ:/root/dir1$ ls

a.txt

“/tmp”、“/var/tmp”目录作为Linux系统的临时文件夹，权限为“rwxrwxrwx”，即允许任意用户、任意程序在该目录中进行创建、删除、移动文件或子目录等操作。但当目录被设置了粘滞位权限以后，即便用户对该目录有写入权限，也不能删除该目录中其他用户的文件数据，而是只有该文件的所有者和root用户才有权将其删除。设置了粘滞位之后，正好可以保持一种动态的平衡：允许各用户在目录中任意写入、删除数据，但是禁止随意删除其他用户的数据。

(NOTE:粘滞位权限只能针对目录设置，对于文件无效)

粘滞位权限都是针对其他用户（other）设置，使用chmod命令设置目录权限时，“o+t”、“o-t”权限模式可分别用于添加、移除粘滞位权限。此时的权限为已经由.x变为.t。此时如果在dir1目录底下创建一个文件a.txt,切换到普通用户环境底下，再一次，发现此时已经无法删除该文件。

关于粘滞位的取消命令（chmod o-t tmp），取消完成之后就和初始状态一样。

28.

root@DESKTOP-TRMDDFQ:~# usermod -L user2

账号禁用的三种方法：  
（1）# usermod -L <username>  
# usermod -U <username> // 解除禁用  
（2）修改/etc/passwd文件  
 1）把第二个字段中的"x"变成其它的字符，该账号就不能登录  
 2）把/[bin](https://www.baidu.com/s?wd=bin&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "https://zhidao.baidu.com/question/_blank)/bash修改成/s[bin](https://www.baidu.com/s?wd=bin&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "https://zhidao.baidu.com/question/_blank)/nologin  
（3）修改/etc/shadow文件  
 1）在第二个密码字段的前面加上一个“!”，该账号就不能登录，这个其实就是usermod -L命令的结果  
 2）在最后两个冒号之间加上数字"1"，表示该账号的密码自1970年1月1日起，过一天后立即过期，当然现在自然就不能登录了。  
如果想解禁，把修改的东西去掉就可以了。

**Lab 11**

1.

root@DESKTOP-TRMDDFQ:~#

Checking t/whether the printer schedular is running in Linux

# lpc status {all 1 printer….}

2.

root@DESKTOP-TRMDDFQ:~#

# vi /etc/printcap

Local Printer

Hp|Our printer:\

:sd=/var/spool/lpd/hp:\

:mx#0:\

:sh:\

:hp=/dev/lp0:\

3.

root@DESKTOP-TRMDDFQ:~#

Sending a print request in Linux

# lpq [-l] [-Pprinter] [job #….]

Example to print a file

# lpq –P hp /radiant/kishore

4.

root@DESKTOP-TRMDDFQ:~#

Restarting the deamon in Linux

# /etc/rc.d/init.d/lpd stop

# /etc/rc.d/init.d/lpd start

5.

root@DESKTOP-TRMDDFQ:~#

Cancel the print job in Linux

Lprm [-Pprinter] [-l] [job #…..] [user…..]

**Lab 12**

1. mkdir dir1 dir2 dir3

cd dir1 : touch file1

cd dir2 : touch file 2

cd dir3 : touch file 3

2. tar cvf /dev/fd0 dir1 dir2 dir3

3. tar tvf /dev/fd0

4. tar cvf /dev/fdo dir1

tar cvf /dev/fdo dir2

tar cvf /dev/fdo dir3

5. tar tvf /dev/fd0

6. cd dir2

ls –l |cpio –o c> /dev/fd0

7. cpio –l c< /dev/fd0

8.dd if = ./ of= /dev/fd0

9.dd if = /dev /fd0of=/home

**Lab 13**

1.

root@DESKTOP-TRMDDFQ:~#

Configuring quotas for user in Linux

# touch [filename]

# touch /home/quota.user

2.

root@DESKTOP-TRMDDFQ:~#

Change the permission

# chmod 600 /home/quota.user

3.

root@DESKTOP-TRMDDFQ:~#

Make an entry in /etc/fstab

# vi /etc/fstab

/dev/device /dir/to/mount ftype parameters fs\_freq fs\_paano

/dev/hda6 home ext2 defaults,usrquota 1 1

4.

root@DESKTOP-TRMDDFQ:~#

Configuring quotas for users in Sun Solaris

Examples- Configuring quotas for user

# touch [filename]

# touch /export /home /quotas

5.

root@DESKTOP-TRMDDFQ:~#

Change the permission

# chmod 600 /export/home/quotas

6.

root@DESKTOP-TRMDDFQ:~#

Make an entry in /etc/vfstab

# vi /etc/vfstab

deviceamount devicetofsck mountpoint fstype fsckpass mountatboot mounoptions

/dev/dsk/c0td0s7 /dev rdsk/c0t3d0s7 /export/home ufs 2 yes rq

7.

root@DESKTOP-TRMDDFQ:~#

Turn on the quotas

# quotaon [options] [filesystems]

# quotaon –va /home

/dev/hda6-quota turned on

8.

root@DESKTOP-TRMDDFQ:~#

Assigning quota for a user

# edquota –u kishore

Quotas for user kishore

/dev/hda6: blocks in use : 0, limits (soft=0, hard=0)

inodes in use:0,limits (soft=0,hard=0)

9.

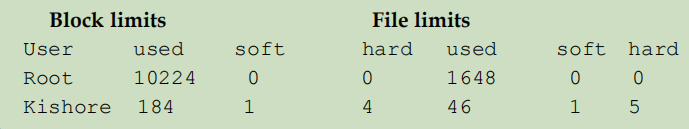
root@DESKTOP-TRMDDFQ:~#

Checking quota for the file system

# repquota [options] [file system]

# repquota –va /home

\*\*\* Report for user quotas on /dev/hda6(/home)



Block limits File limits

User used soft hard used soft hard

Root 10224 0 0 1648 0 0

Kishore 184 1 4 46 1 5

10.

root@DESKTOP-TRMDDFQ:~#

Turn off the quotas:

To turn off quotas:

# quotaoff [options] [ filesystem]

# quotaoff –va /home

/dev/hda6 –quota turned off

**Lab 14**

2.

root@DESKTOP-TRMDDFQ:~#

# vi /etc/at.deny

……………… enter the login name of the user to be denied

user1

:wq!

3.

root@DESKTOP-TRMDDFQ:~#

# at 11:30

at> echo WELCOME TO RADIANT>/dev/console

4.

root@DESKTOP-TRMDDFQ:~#

get the jobs information

#atq

#atrm <at job number>

5.

root@DESKTOP-TRMDDFQ:~#

#atq

6. /var/spool/at

7. atrm <at job number>

8. at –d

9.

root@DESKTOP-TRMDDFQ:~#

# vi /etc/at.allow

………….. enter the login name………………

user1

wq!

10.

root@DESKTOP-TRMDDFQ:~#

# at midnight

at> rm <filename>

wq!

11.

root@DESKTOP-TRMDDFQ:~#

# crontab –e

……………enter the values like this…………………

3006 \*\*\*<command to be executed>

:wq!

12.

root@DESKTOP-TRMDDFQ:~#

# crontab –u <user name> -r

13.

root@DESKTOP-TRMDDFQ:~#

# crontab –u <user name> -l

/var/spool/cron/<user name>

# vi /etc/cron.deny

…………………..enter the username to be restricted

user1

:wq!

# vi /etc/cron.allow

…………………..enter the user names to be allowed

user1

user2

user3

:wq!

**Lab 15**

ps命令用于显示当前进程 (process) 的状态。

语法：

ps [options] [--help]

参数：

|  |  |
| --- | --- |
| -a | 显示所有终端机下执行的程序，除了阶段作业领导者之外 |
| a | 显示现行终端机下的所有程序，包括其他用户的程序 |
| -A | 显示所有程序 |
| -c | 显示CLS和PRI栏位 |
| c | 列出程序时，显示每个程序真正的指令名称，而不包含路径，选项或常驻服务的标示 |
| -C <指令名称> | 指定执行指令的名称，并列出该指令的程序的状况 |
| -d | 显示所有程序，但不包括阶段作业领导者的程序 |
| -e | 此选项的效果和指定”A”选项相同 |
| e | 列出程序时，显示每个程序所使用的环境变量 |
| -f | 显示UID,PPIP,C与STIME栏位 |
| f | 用ASCII字符显示树状结构，表达程序间的相互关系 |
| -g <群组名称> | 此选项的效果和指定”-G”选项相同，当亦能使用阶段作业领导者的名称来指定 |
| g | 显示现行终端机下的所有程序，包括群组领导者的程序 |
| -G <群组识别码> | 列出属于该群组的程序的状况，也可使用群组名称来指定 |
| h | 不显示标题列 |
| -H | 显示树状结构，表示程序间的相互关系 |
| -j或j | 采用工作控制的格式显示程序状况 |
| -l或l | 采用详细的格式来显示程序状况 |
| L | 列出栏位的相关信息 |
| -m或m | 显示所有的执行绪 |
| n | 以数字来表示USER和WCHAN栏位 |
| -N | 显示所有的程序，除了执行ps指令终端机下的程序之外 |
| -p <程序识别码> | 指定程序识别码，并列出该程序的状况 |
| p <程序识别码> | 此选项的效果和指定”-p”选项相同，只在列表格式方面稍有差异 |
| r | 只列出现行终端机正在执行中的程序 |
| -s <阶段作业> | 指定阶段作业的程序识别码，并列出隶属该阶段作业的程序的状况 |
| s | 采用程序信号的格式显示程序状况 |
| S | 列出程序时，包括已中断的子程序资料 |
| -t <终端机编号> | 指定终端机编号，并列出属于该终端机的程序的状况 |
| t <终端机编号> | 此选项的效果和指定”-t”选项相同，只在列表格式方面稍有差异 |
| -T | 显示现行终端机下的所有程序 |
| -u <用户识别码> | 此选项的效果和指定”-U”选项相同 |
| u | 以用户为主的格式来显示程序状况 |
| -U <用户识别码> | 列出属于该用户的程序的状况，也可使用用户名称来指定 |
| U <用户名称> | 列出属于该用户的程序的状况 |
| v | 采用虚拟内存的格式显示程序状况 |
| -V或V | 显示版本信息 |
| -w或w | 采用宽阔的格式来显示程序状况 |
| x | 显示所有程序，不以终端机来区分 |
| X | 采用旧式的Linux i386登陆格式显示程序状况 |
| -y | 配合选项”-l”使用时，不显示F(flag)栏位，并以RSS栏位取代ADDR栏位 |
| -- <程序识别码> | 此选项的效果和指定”p”选项相同 |
| --cols <每列字符数> | 设置每列的最大字符数 |
| --columns <每列字符数> | 此选项的效果和指定”--cols”选项相同 |
| --cumulative | 此选项的效果和指定”S”选项相同 |
| --deselect | 此选项的效果和指定”-N”选项相同 |
| --forest | 此选项的效果和指定”f”选项相同 |
| --headers | 重复显示标题列 |
| --help | 在线帮助 |
| --info | 显示排错信息 |
| --lines <显示列数> | 设置显示画面的列数 |
| --no-headers | 此选项的效果和指定”h”选项相同，只在列表格式方面稍有差异 |
| --group <群组名称> | 此选项的效果和指定”-G”选项相同 |
| --Group <群组识别码> | 此选项的效果和指定”-G”选项相同 |
| --pid <程序识别码> | 此选项的效果和指定”-p”选项相同 |
| --rows <显示列数> | 此选项的效果和指定”--lines”选项相同 |
| --sid <阶段作业> | 此选项的效果和指定”-s”选项相同 |
| -tty <终端机编号> | 此选项的效果和指定”-t”选项相同 |
| --user <用户名称> | 此选项的效果和指定”-U”选项相同 |
| --User <用户识别码> | 此选项的效果和指定”-U”选项相同 |
| --version | 此选项的效果和指定”-V”选项相同 |
| --widty <每列字符数> | 此选项的效果和指定”-cols”选项相同 |

au(x) 输出格式 :

USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND

USER: 行程拥有者

PID: pid

%CPU: 占用的 CPU 使用率

%MEM: 占用的记忆体使用率

VSZ: 占用的虚拟记忆体大小

RSS: 占用的记忆体大小

TTY: 终端的次要装置号码 (minor device number of tty)

STAT: 该行程的状态:

D: 无法中断的休眠状态 (通常 IO 的进程)

R: 正在执行中

S: 静止状态

T: 暂停执行

Z: 不存在但暂时无法消除

W: 没有足够的记忆体分页可分配

<: 高优先序的行程

N: 低优先序的行程

L: 有记忆体分页分配并锁在记忆体内 (实时系统或捱A I/O)

START: 行程开始时间

TIME: 执行的时间

COMMAND:所执行的指令

1.

# List the processes for the current shell

root@DESKTOP-TRMDDFQ:~# ps

PID TTY TIME CMD

3 tty1 00:00:00 init

17 tty1 00:00:00 sudo

18 tty1 00:00:00 bash

33 tty1 00:00:00 adduser

47 tty1 00:00:00 chfn

118 tty1 00:00:00 su

137 tty1 00:00:00 sudo

138 tty1 00:00:00 bash

199 tty1 00:00:00 ps

2.

root@DESKTOP-TRMDDFQ:~# ps -ef # 显示所有命令，连带命令行

UID PID PPID C STIME TTY TIME CMD

root 1 0 0 10:12 ? 00:00:00 /init

root 3 1 0 10:12 tty1 00:00:00 /init

yql 4 3 0 10:12 tty1 00:00:00 -bash

root 17 4 0 10:44 tty1 00:00:00 sudo -i

root 18 17 0 10:44 tty1 00:00:00 -bash

root 33 18 0 10:44 tty1 00:00:00 adduser

root 47 33 0 10:44 tty1 00:00:00 /usr/bin/chfn user1

root 118 18 0 11:47 tty1 00:00:00 su yql

yql 119 118 0 11:47 tty1 00:00:00 bash

root 137 119 0 11:48 tty1 00:00:00 sudo -i

root 138 137 0 11:48 tty1 00:00:00 -bash

root 200 138 0 22:53 tty1 00:00:00 ps -ef

3.

root@DESKTOP-TRMDDFQ:~# ps –ecl #显示进程的全局优先级并找出提供的列

F S UID PID PPID CLS PRI ADDR SZ WCHAN TTY TIME CMD

0 S 0 1 0 TS 19 - 2081 - ? 00:00:00 init

0 S 0 3 1 TS 19 - 2084 - tty1 00:00:00 init

0 S 1000 4 3 TS 19 - 4198 - tty1 00:00:00 bash

0 S 0 17 4 TS 19 - 4317 - tty1 00:00:00 sudo

0 S 0 18 17 TS 19 - 4199 - tty1 00:00:00 bash

0 T 0 33 18 TS 19 - 5898 - tty1 00:00:00 adduser

0 T 0 47 33 TS 19 - 3875 - tty1 00:00:00 chfn

0 S 0 118 18 TS 19 - 4130 - tty1 00:00:00 su

0 S 1000 119 118 TS 19 - 4173 - tty1 00:00:00 bash

0 S 0 137 119 TS 19 - 4317 - tty1 00:00:00 sudo

0 S 0 138 137 TS 19 - 4199 - tty1 00:00:00 bash

0 R 0 201 138 TS 19 - 4271 - tty1 00:00:00 ps

4.

# 使用默认参数更改进程的优先级

root@DESKTOP-TRMDDFQ:~# /usr/bin/nice +4 (default four units)

PRI表示Priority，NI表示Nice，都是优先级的意思，数字越小表示优先级越高。普通用户只能调高自己的NI（0~19），用户只能修改NI，不能直接修改PRI；而root用户可以调整任意用户进程NI（-20~19）。

（1）一开始执行程序就指定nice值

语法：nice [-n <优先等级>][--help][--version][执行指令]

nice -n -5 /usr/local/mysql/bin/mysqld\_safe &

参数：

-n<优先等级>或-<优先等级>或–adjustment=<优先等级>设置欲执行的指令的优先权等级。等级的范围从-20-19，其中-20最高，19最低，只有系统管理者可以设置负数的等级。

（2）调整已存在进程的nice

语法：renice [优先等级][-g <程序群组名称>...][-p <程序识别码>...][-u <用户名称>...]

renice -5 -p 5200 # PID为5200的进程nice设为-5

参数：

-g <程序群组名称> 　使用程序群组名称，修改所有隶属于该程序群组的程序的优先权。

-p <程序识别码> 　改变该程序的优先权等级，此参数为预设值。

-u <用户名称> 　指定用户名称，修改所有隶属于该用户的程序的优先权。

（3）用top命令更改已存在进程的nice

top # 进入top后按“r”–>输入进程PID–>输入nice值

5.

root@DESKTOP-TRMDDFQ:~# vmstat # 显示虚拟内存统计信息

procs -----------memory---------- ---swap-- -----io---- -system-- ------cpu-----

r b swpd free buff cache si so bi bo in cs us sy id wa st

vmstat(Virtual Memory Statistics 虚拟内存统计) 命令用来显示Linux系统虚拟内存状态，也可以报告关于进程、内存、I/O等系统整体运行状态。

语法：

vmstat [-a] [-n] [-t] [-S unit] [delay [ count]]

vmstat [-s] [-n] [-S unit]

vmstat [-m] [-n] [delay [ count]]

vmstat [-d] [-n] [delay [ count]]

vmstat [-p disk partition] [-n] [delay [ count]]

vmstat [-f]

vmstat [-V]

参数：

-a：显示活跃和非活跃内存

-f：显示从系统启动至今的fork数量 。

-m：显示slabinfo

-n：只在开始时显示一次各字段名称。

-s：显示内存相关统计信息及多种系统活动数量。

delay：刷新时间间隔。如果不指定，只显示一条结果。

count：刷新次数。如果不指定刷新次数，但指定了刷新时间间隔，这时刷新次数为无穷。

-d：显示磁盘相关统计信息。

-p：显示指定磁盘分区统计信息

-S：使用指定单位显示。参数有 k 、K 、m 、M ，分别代表1000、1024、1000000、1048576字节（byte）。默认单位为K（1024 bytes）

-V：显示vmstat版本信息。

Procs（进程）

|  |  |
| --- | --- |
| r: | 运行队列中进程数量，这个值也可以判断是否需要增加CPU。（长期大于1） |
| b | 等待IO的进程数量。 |

Memory（内存）

|  |  |
| --- | --- |
| swpd | 使用虚拟内存大小，如果swpd的值不为0，但是SI，SO的值长期为0，这种情况不会影响系统性能。 |
| free | 空闲物理内存大小。 |
| buff | 用作缓冲的内存大小。 |
| cache | 用作缓存的内存大小，如果cache的值大的时候，说明cache处的文件数多，如果频繁访问到的文件都能被cache处，那么磁盘的读IO bi会非常小。 |

Swap

|  |  |
| --- | --- |
| si | 每秒从交换区写到内存的大小，由磁盘调入内存。 |
| so | 每秒写入交换区的内存大小，由内存调入磁盘。 |

内存够用的时候，这2个值都是0，如果这2个值长期大于0时，系统性能会受到影响，磁盘IO和CPU资源都会被消耗。

IO

|  |  |
| --- | --- |
| bi | 每秒读取的块数 |
| bo | 每秒写入的块数 |

随机磁盘读写的时候，这2个值越大（如超出1024k)，CPU在IO等待的值也会越大。

system（系统）

|  |  |
| --- | --- |
| in | 每秒中断数，包括时钟中断。 |
| cs | 每秒上下文切换数。 |

值越大，由内核消耗的CPU时间会越大。

CPU（以百分比表示）

|  |  |
| --- | --- |
| us | 用户进程执行时间百分比(user time) us的值比较高时，说明用户进程消耗的CPU时间多，但是如果长期超50%的使用，那么我们就该考虑优化程序算法或者进行加速。 |
| sy: | 内核系统进程执行时间百分比(system time) sy的值高时，说明系统内核消耗的CPU资源多，这并不是良性表现，我们应该检查原因。 |
| wa | IO等待时间百分比 wa的值高时，说明IO等待比较严重，这可能由于磁盘大量作随机访问造成，也有可能磁盘出现瓶颈（块操作）。 |
| id | 空闲时间百分比 |

6.

# 显示自上次引导系统以来发生的系统事件数量

root@DESKTOP-TRMDDFQ:~# vmstat –s

7.

root@DESKTOP-TRMDDFQ:~# vmstat –S # 显示交换统计信息

-S：使用指定单位显示。参数有 k 、K 、m 、M ，分别代表1000、1024、1000000、1048576字节（byte）。默认单位为K（1024 bytes）

8.

root@DESKTOP-TRMDDFQ:~# sar –a

[sar](http://lovesoo.org/tag/sar" \o "View all posts in sar" \t "http://lovesoo.org/_blank)（System Activity Reporter系统活动情况报告）是目前[Linux](http://lovesoo.org/tag/linux" \o "View all posts in Linux" \t "http://lovesoo.org/_blank)上最为全面的系统[性能分析](http://lovesoo.org/tag/%e6%80%a7%e8%83%bd%e5%88%86%e6%9e%90" \o "View all posts in 性能分析" \t "http://lovesoo.org/_blank)工具之一，可以从多方面对系统的活动进行报告，包括：文件的读写情况、系统调用的使用情况、[磁盘](http://lovesoo.org/tag/%e7%a3%81%e7%9b%98" \o "View all posts in 磁盘" \t "http://lovesoo.org/_blank)[I/O](http://lovesoo.org/tag/io" \o "View all posts in I/O" \t "http://lovesoo.org/_blank)、[CPU](http://lovesoo.org/tag/cpu" \o "View all posts in CPU" \t "http://lovesoo.org/_blank)效率、[内存](http://lovesoo.org/tag/%e5%86%85%e5%ad%98" \o "View all posts in 内存" \t "http://lovesoo.org/_blank)使用状况、进程活动及IPC有关的活动等。

语法：

sar [ 选项 ] [ <时间间隔> [ <次数> ] ]

参数：

-A:所有报告的总和

-b:显示I/O和传递速率的统计信息（每10秒采样一次，连续采样3次，报告缓冲区的使用情况：sar -b 10 3）

-B:显示换页状态

-d:输出每一块磁盘的使用信息

-e:设置显示报告的结束时间

-f:从制定的文件读取报告

-i:设置状态信息刷新的间隔时间

-P:报告每个CPU的状态

-R:显示内存状态

–u:输出cpu使用情况和统计信息

–v:显示索引节点、文件和其他内核表的状态

-w:显示交换分区的状态

-x:显示给定进程的装

-r:报告内存利用率的统计信息

9.

root@DESKTOP-TRMDDFQ:~# sar –b # 查看IO和传递速率

10.

root@DESKTOP-TRMDDFQ:~# sar –d # 查看磁盘使用情况

11.

root@DESKTOP-TRMDDFQ:~# sar –m

12.

root@DESKTOP-TRMDDFQ:~# sar -r # 查看内存使用情况

13.

root@DESKTOP-TRMDDFQ:~# sar -w # 输出交换活动信息