

– General Linux 1 –

(Linux Professional Institute Certification)

104-4 Managing disk quota [1]

a

```
.~.  
/V\   by: geoffrey robertson  
//  \  geoffrey@zip.com.au  
@._.@
```

\$Id: 104-4.tex,v 1.2 2003/05/30 04:51:59 waratah Exp \$

^aCopyright © 2002 Geoffrey Robertson. Permission is granted to make and distribute verbatim copies or modified versions of this document provided that this copyright notice and this permission notice are preserved on all copies under the terms of the GNU General Public License as published by the Free Software Foundation—either version 2 of the License or (at your option) any later version.

List of Slides

Objective—Managing disk quota

Candidates should be able to manage disk quotas for users. This objective includes setting up a disk quota for a filesystem, editing, checking, and generating user quota reports.

Key files, terms and utilities

- quota
- edquota
- repquota
- quotaon

Disk Quotas

Quotas are used to enforce a disk space limit or an inode maximum on individuals or groups.

Types of quota limits:

user hard: The maximum size for a user.

user soft: A user's warning threshold.

group hard: The maximum size for a group.

group soft: A group's warning threshold.

grace period: A soft limit time restriction.

Setting Up Quotas

You must configure your init script to check and turn on QUOTA at boot time.

The golden rule is turn on quota *after* your `/etc/fstab` has been mounted.

```
#check quota and then turn quota on.
if [ -x /usr/sbin/quotacheck ]; then
    echo "Checking quotas. This may take sometime. "
    /usr/sbin/quotacheck -avug
    echo "Done."
fi
if [ -x /usr/sbin/quotaon ]; then
    echo "Turning on quota."
    /usr/sbin/quotaon -avug
fi
```

Entries in /etc/fstab

No Quota :

```
/dev/hda2 /home ext2 default 1 1
```

User Quota Set :

```
/dev/hda2 /home ext2 default,usrquota 1 1
```

Group Quota Set :

```
/dev/hda2 /home ext2 default,grpquota 1 1
```

Create a Quota Record

Go to the partition that you wish to restrict usage in and as root:

```
# touch /partition/quota.user  
# touch /partition/quota.group  
# chmod 600 /partition/quota.user  
# chmod 600 /partition/quota.group
```

Then reboot the system to take affect. (Remount the partition)

Assigning quota for users and groups

Setup quota for a user/group

```
# edquota -u USER
```

This will bring you into vi:

```
Quota for user USER:  
(Quota for group GROUP:)  
/dev/hda2:  
block in use: 2594, limits (soft = 5000, hard = 6500)  
inodes in use: 356, limits (soft = 1000, hard = 1500)  
~  
~
```

block in use: is the total number of block(in kilobytes).

inodes in use: is the total of files.

limits (soft =) indicates the maximum amount of hdd usage combined with the GRACE PERIOD acts as the border. There is warning to tell you, that you are close to this limit

limits (hard =) only works when the GRACE PERIOD is set. This is the ABSOLUTE limit of disk usage

Grace period.

Setting the grace period is in the format sec(onds), min(utes), hour(s), day(s), week(s), and month(s).

Setting a Grace Period

Setting the grace period is in the format sec(onds), min(utes), hour(s), day(s), week(s), and month(s).

```
# edquota -t
```

```
-----  
Time units may be: days, hours, minutes, or seconds  
Grace period before enforcing soft limits for users:  
/dev/hda2: block grace period: 0 days,  
file grace period: 0 days  
~  
~  
~  
-----
```

Quota Commands

quotacheck

Quotacheck is used to scan the disk usage and update the "quota.user". This should be automated as a cronjob periodically (weekly).

- v** this will produce some useful information about what it is doing.
- d** debug (very detail -v).
- u** user (all files with uid).
- g** group (all file with gid).
- a** all.
- R** except root (when used with -a).

repquota

Repquota produces a summarised report on the quotas on the system.

- a** all file system indicated in /etc/fstab.
- v** report all quotas, even if there is no usage.
- g** report group quotas.
- u** report user quotas (default).

quotaon / quotaoff

Turn quota on.

- a** all (as in /etc/fstab).
- v** display message for each file system.
- g** manipulate user quotas.
- u** manipulate group quotas.

MAKE

Make is a tool that is a very simple idea that allows large systems to grow naturally. Sounds like a lot of Unix tools.

The concept is simple, you have a file that is built from another file (or more files). so for example our site has some latex code, we want to produce a pdf format however it takes a couple of steps to do this. I will work backwards to show it a little clearer:

The following will check the timestamp on the dvi and if it is newer than the pdf file it will run the commands.

```
notes.pdf:      notes.dvi
                dv12pdf notes.dvi
```

note that the indent **MUST** be a space.

Now we want to check that the dvi file is up to date.

```
notes.dvi:      notes.tex
                latex notes.tex
```

the make program will work out all the timestamps and work out that notes.dvi must be built first.

This simple example shows that we can build up a number of parts to make a program.

SO if we talk about programs we might have something like this:

```
xyz.o: xyz.c xyz.h common.h  
      gcc xyz.c -o xyz.o
```

This will produce a new xyz.o if xyz.c or xyz.h or common.h changes.

Finally make give us some built in processes so we can use shorthand like:

```
xyz.o: xyz.c xyz.h common.h
```

The make process implies the compiler and option settings. In fact this is preferred and some features are built into make to allow you to extend it, like macros called CFLAGS, etc.

It is outside the scope of the short talk to cover this in more detail.