

Quota Management

What you will practice

- Installing quota tools
- Creating a throw-away ext4 file-system that supports **user** and **group** quotas
- Turning quotas on/off and generating the quota database
- Setting soft/hard block limits, inode limits, and grace periods with both **edquota** and **setquota**
- Reading usage with **quota**, **repquota**, and **warnquota**
- Watching soft-limit grace periods expire and hard-limit writes get denied

Preparing the lab

1. Prerequisites (one-time)

Become root for the session

sudo -i

Install the quota utilities package

apt update && apt install quota -y

Command 1: **sudo -i**

Component	Explanation
sudo	Stands for "superuser do". It allows a permitted user to execute a command as the superuser (root) or another user, as specified by the security policy.
-i	Starts a login shell as the root user. This simulates a full root login, loading root's environment variables (e.g., <code>PATH</code> , <code>HOME</code> , etc.) and reading root's login files like <code>.profile</code> or <code>.bashrc</code> .

Command 2: **apt update && apt install quota -y**

Component	Explanation
apt	Advanced Package Tool. A command-line interface for handling packages on Debian-based systems like Ubuntu.
update	Retrieves and updates the list of available packages and their versions from configured repositories, but does not install or upgrade any packages.
&&	Logical AND operator. Ensures that the second command (apt install ...) runs only if the first command (apt update) succeeds.
apt install	Installs one or more packages on the system.
quota	The name of the package being installed. It provides user and group disk quota tools, allowing disk usage limits to be enforced.
-y	Automatically answers "yes" to any prompts, such as confirmation of package installation. Useful in scripts or automated setups.

```
tiago-paquete@Linux:~$ sudo -i
```

```
=====
[sudo] password for tiago-paquete:
=====
```

```
root@Linux:~# apt update && apt install quota -y
```

```
=====
Hit:1 https://packages.microsoft.com/repos/code stable InRelease
...
=====
```

2. Create a disposable 2 GiB “disk”

Why: Quotas only work on file-systems mounted with `usrquota` / `grpquota`.

How: Use a loop-back file so you don’t need a real extra disk.

2 GiB sparse file to act as a block device

`mkdir -p /srv/disks`

`fallocate -l 2G /srv/disks/quotatest.img`

Associate it with the next free loop device, e.g. `/dev/loop0`

`losetup -fP /srv/disks/quotatest.img`

Identify the loop name for later commands

`LOOPDEV=$(losetup -a | grep quotatest.img | cut -d: -f1)`

Make an EXT4 file-system

`mkfs.ext4 "$LOOPDEV"`

1. `mkdir -p /srv/disks`

Component	Explanation
<code>mkdir</code>	Command to create a directory.
<code>-p</code>	Creates parent directories as needed (no error if existing).
<code>/srv/disks</code>	Target directory to be created for storing the disk image.

2. `fallocate -l 2G /srv/disks/quotatest.img`

Component	Explanation
<code>fallocate</code>	Efficiently allocates space for a file without writing data.
<code>-l</code>	Specifies the length of the file to allocate.
<code>2G</code>	Size of the file: 2 GiB (Gibibytes).
<code>/srv/disks/quotatest.img</code>	Path and filename of the sparse image file to be created.

3. `losetup -fP /srv/disks/quotatest.img`

Component	Explanation
<code>losetup</code>	Tool to set up and manage loop devices.
<code>-f</code>	Finds the first unused (free) loop device.
<code>-P</code>	Forces kernel to scan for partitions (useful if the image contains partition table).
<code>/srv/disks/quotatest.img</code>	File to be associated with the loop device.

4. `LOOPDEV=$(losetup -a | grep quotatest.img | cut -d: -f1)`

Component	Explanation
<code>LOOPDEV=</code>	Assigns the output of the command substitution to the variable <code>LOOPDEV</code> .
<code>\$(...)</code>	Command substitution: runs the command inside and returns its output.
<code>losetup -a</code>	Lists all active loop devices and their associated files.
<code>grep quotatest.img</code>	Filters the output to include only the line with <code>quotatest.img</code> .
<code>cut -d: -f1</code>	Cuts the string at the colon (:) delimiter and extracts the first field, which is the loop device path (e.g., <code>/dev/loop0</code>).

5. `mkfs.ext4 "$LOOPDEV"`

Component	Explanation
<code>mkfs.ext4</code>	Creates an ext4 filesystem on the specified device.
<code>"\$LOOPDEV"</code>	The loop device previously assigned (e.g., <code>/dev/loop0</code>). Quotation ensures correct parsing if variable contains spaces.

```
root@Linux:~# mkdir -p /srv/disks
```

```
root@Linux:~# fallocate -l 2G /srv/disks/quotatest.img
```

```
root@Linux:~# losetup -fP /srv/disks/quotatest.img
```

```
root@Linux:~# LOOPDEV=$(losetup -a | grep quotatest.img | cut -d: -f1)
```

```
root@Linux:~# echo $LOOPDEV
```

```
=====
/dev/loop16
=====
```

root@Linux:~# **mkfs.ext4 "\$LOOPDEV"**

=====

mkfs 1.47.0 (5-Feb-2023)

Discarding device blocks: done

Creating filesystem with 524288 4k blocks and 131072 inodes

Filesystem UUID: ebdfc028-cdb8-410e-9ba9-ac90af90063a

Superblock backups stored on blocks:

32768, 98304, 163840, 229376, 294912

Allocating group tables: done

Writing inode tables: done

Creating journal (16384 blocks): done

Writing superblocks and filesystem accounting information: done

=====

3. Mount it with quota options

Mount point

mkdir -p /mnt/quotatest

Add to /etc/fstab (persistent across reboots)

echo "\$LOOPDEV /mnt/quotatest ext4 defaults,usrquota,grpquota 0 2" >> /etc/fstab

Mount all entries (or just remount the new one)

mount -a

Verify the options:

mount | grep /mnt/quotatest

Command 1: **mkdir -p /mnt/quotatest**

Component	Explanation
mkdir	Command to create a directory or directories.
-p	Creates parent directories as needed. Does not raise an error if the directory exists.
/mnt/quotatest	The path of the directory to be created. This is the mount point for the filesystem.

Command 2:

echo "\$LOOPDEV /mnt/quotatest ext4 defaults,usrquota,grpquota 0 2" >> /etc/fstab

Component	Explanation
echo	Prints the string to standard output.
"\$LOOPDEV ..."	Expands the value of the variable LOOPDEV (e.g., /dev/loop0) and appends the full line.
/mnt/quotatest	The mount point where the device should be mounted.
ext4	The filesystem type to be used.
defaults	Uses default mount options: <i>rw, suid, dev, exec, auto, nouser, async</i> .
usrquota	Enables user disk quotas on the mounted filesystem.

grpquota	Enables group disk quotas on the mounted filesystem.
0	Dump flag: 0 means no need to back up this filesystem with dump .
2	Fsck order: 2 means this filesystem should be checked after the root filesystem.
>> /etc/fstab	Appends the line to the /etc/fstab file to make the mount persistent across reboots.

Command 3: mount -a

Component	Explanation
mount	Mounts filesystems.
-a	Mounts all filesystems listed in /etc/fstab except those with noauto option.

Command 4: mount | grep /mnt/quotatest

Component	Explanation
mount	Displays a list of currently mounted filesystems.
grep /mnt/quotatest	Filters output to show only lines containing /mnt/quotatest, verifying that the mount occurred and what options were used.

```
root@Linux:~# mkdir -p /mnt/quotatest
```

```
root@Linux:~# echo "$LOOPDEV /mnt/quotatest ext4
defaults,usrquota,grpquota 0 2" >> /etc/fstab
```

```
root@Linux:~# mount -a
```

```
=====
mount: (hint) your fstab has been modified, but systemd still uses
the old version; use 'systemctl daemon-reload' to reload.
```

```
root@Linux:~# mount | grep /mnt/quotatest
/dev/loop16 on /mnt/quotatest type ext4
(rw,relatime,quota,usrquota,grpquota)
=====
```

4. Prepare the quota database and switch quotas on

quotacheck -cugm /mnt/quotatest
quotaon /mnt/quotatest

quotacheck -cugm /mnt/quotatest initializes quota files and populates them with current usage data.

quotaon /mnt/quotatest activates the enforcement of user/group disk quotas on the specified mount point.

As a result, **aquota.user** and **aquota.group** files are created in **/mnt/quotatest**.

Component	Explanation
quotacheck	A command used to scan a filesystem for disk usage and update the quota database files (aquota.user , aquota.group).
-c	Create new quota files if they do not exist.
-u	Check user quotas. Creates or updates the aquota.user file.
-g	Check group quotas. Creates or updates the aquota.group file.
-m	Perform the check even if the filesystem is marked as read-write. Suppresses warnings about the FS being mounted.
/mnt/quotatest (in quotacheck)	Target mount point where quotas are to be checked or initialized. Must be the root of a quota-enabled filesystem.
quotaon	A command that enables disk quota enforcement on a mounted filesystem.
/mnt/quotatest (in quotaon)	Mount point of the filesystem where quotas are being enabled.

```
root@Linux:~# quotacheck -cugm /mnt/quotatest
root@Linux:~# quotaon /mnt/quotatest
```


5. Create practice identities & workspace

A shared project directory

`mkdir /mnt/quotatest/projects`

`groupadd acme`

`chown :acme /mnt/quotatest/projects`

`chmod 2775 /mnt/quotatest/projects` # setgid, so new files inherit the group

Command / Component	Explanation
<code>mkdir /mnt/quotatest/projects</code>	Creates a directory at the path <code>/mnt/quotatest/projects</code> . <code>mkdir</code> stands for "make directory".
<code>/mnt/quotatest/projects</code>	The full path where the new directory will be created. <code>/mnt</code> is commonly used for mounted filesystems.
<code>groupadd acme</code>	Creates a new user group named <code>acme</code> . <code>groupadd</code> is used to define a new group in the system.
<code>acme</code>	The name of the new group to be added.
<code>chown :acme /mnt/quotatest/projects</code>	Changes the group ownership of the directory <code>/mnt/quotatest/projects</code> to the <code>acme</code> group. The colon <code>:</code> before the group name means you're only changing the group, not the owner.
<code>:acme</code>	Shorthand syntax to indicate that only the group ownership should be changed to <code>acme</code> . No user ownership change is applied.
<code>chmod 2775 /mnt/quotatest/projects</code>	Sets permissions and special modes on the directory. <code>2</code> sets the setgid (Set Group ID) bit, and <code>775</code> gives rw-rw-r-x permissions.
<code>2</code> (in <code>2775</code>)	Setgid bit for directories. Ensures new files/directories created inside inherit the group of the parent directory (i.e., <code>acme</code>).
<code>775</code>	Standard Unix file permissions: Owner (rwx), Group (rwx), Others (r-x). Allows group members to read, write, and execute, while others can read and execute only.
<code>/mnt/quotatest/projects</code> (in <code>chmod</code>)	Target directory whose permissions are being modified.

```

root@Linux:~# mkdir /mnt/quotatest/projects
root@Linux:~# groupadd acme
root@Linux:~# chown :acme /mnt/quotatest/projects
root@Linux:~# chmod 2775 /mnt/quotatest/projects

```

Verification Command	Explanation
<code>ls -ld /mnt/quotatest/projects</code>	Lists detailed information about the directory. You'll verify: <ul style="list-style-type: none"> • Ownership: <code>root:acme</code> (group changed) • Permissions: <code>drwxrwsr-x</code> (SetGID bit set: <code>s</code> in group execute)
<code>getfacl /mnt/quotatest/projects</code>	Displays Access Control List (ACL) entries. This confirms both traditional permissions and any ACLs (if configured). Not always required unless ACLs are used.
<code>stat /mnt/quotatest/projects</code>	Provides complete metadata including owner, group, and permission bits, including the setgid bit in the "Access" line.
<code>getent group acme</code>	Verifies that the acme group exists and shows its members (will be empty unless users are added).
<code>find /mnt/quotatest/projects -type d -exec ls -ld {} \;</code>	Recursively lists all directories under <code>/mnt/quotatest/projects</code> to ensure inherited group ownership and permissions. Useful after adding files/directories.

Verify:

```

root@Linux:~# ls -ld /mnt/quotatest/projects

```

```

=====
drwxrwsr-x 2 root acme 4096 May  4 19:57 /mnt/quotatest/projects
=====

```

```

root@Linux:~# getfacl /mnt/quotatest/projects

```

```

=====
getfacl: Removing leading '/' from absolute path names
# file: mnt/quotatest/projects
# owner: root
# group: acme
# flags: -s-
user::rwx
group::rwx
other::r-x
=====

```

```

root@Linux:~# stat /mnt/quotatest/projects

```

```

=====
File: /mnt/quotatest/projects
  Size: 4096      Blocks: 8          IO Block: 4096   directory
Device: 7,16 Inode: 8193       Links: 2

```

```
Access: (2775/drwxrwsr-x)  Uid: (    0/    root)  Gid: ( 1005/    acme)
Access: 2025-05-04 19:57:24.708249614 +0200
Modify: 2025-05-04 19:57:24.708249614 +0200
Change: 2025-05-04 19:57:46.941897013 +0200
  Birth: 2025-05-04 19:57:24.708249614 +0200
```

=====

```
root@Linux:~# getent group acme
```

=====

```
acme:x:1005:
```

=====

```
root@Linux:~# find /mnt/quotatest/projects -type d -exec ls -ld {} \;
```

=====

```
drwxrwsr-x 2 root acme 4096 May  4 19:57 /mnt/quotatest/projects
```

=====

Scenario tasks

Task A – Set user quotas

User	Soft	Hard	Soft	Hard	Grace
alice	10 000	15 000	1 000	1 200	7 days
bob	20 000	25 000	2 000	2 500	7 days
intern01	5 000	7 000	500	700	7 days

Blocks are 1 KiB (default for setquota/edquota on ext4).

Option 1 – interactive (edquota)

edquota -u alice

Repeat for bob and intern01

edquota -t # set 7-day grace for blocks & inodes

Option 1 – interactive (edquota)

Component	Explanation
edquota	Command to edit disk quotas using a text editor (usually vi or nano).
-u	Specifies that the quota is being set for a user (as opposed to a group).
alice,bob, intern01	Username for whom the disk quota will be edited interactively.
edquota -t	Opens grace period settings for blocks and inodes in the text editor.
7-day grace	Refers to a grace period of 7 days during which users can exceed soft limits before being enforced.
blocks & inodes	Applies the grace period to both disk space (blocks) and file count (inodes) .

```
root@Linux:~# edquota -u alice
```

```
=====
GNU nano 7.2 /tmp//EdP.aIpi5HI
Disk quotas for user alice (uid 1001):
  Filesystem            blocks      soft      hard      inodes
soft      hard
  /dev/loop16           0      10000    150000         0
1000      1200
=====
```

```
root@Linux:~# edquota -t
```

```
=====
GNU nano 7.2 /tmp//EdP.a0A8E3t
Grace period before enforcing soft limits for users:
Time units may be: days, hours, minutes, or seconds
  Filesystem      Block grace period      Inode grace period
  /dev/loop16      7days                    7days
=====
```

Option 2 – one-liners (setquota)

```
# setquota -u <user> <softBlks> <hardBlks> <softIno> <hardIno> <fs>
```

```
setquota -u alice 10000 15000 1000 1200 /mnt/quotatest
```

```
setquota -u bob 20000 25000 2000 2500 /mnt/quotatest
```

```
setquota -u intern01 5000 7000 500 700 /mnt/quotatest
```

```
setquota -u -t 604800 604800 /mnt/quotatest # 604 800 s = 7 days
```

Option 2 – one-liners (setquota)

Component	Explanation
setquota	Command to set quotas directly (non-interactive) for users or groups.
-u	Specifies that the quota is being applied to a user .
alice,bob, intern01	Target usernames for quota assignment.
10000,20000,5000	Soft block limit (in kilobytes) — limit before grace period kicks in.
15000,25000,7000	Hard block limit — absolute maximum disk usage.
1000,2000,500	Soft inode limit — limit on number of files before grace period applies.
1200,2500,700	Hard inode limit — max number of files allowed.
/mnt/quotatest	Filesystem mount point where quota is enforced.

Set global grace period using setquota -t

Component	Explanation
setquota -t	Command to set global grace periods for the specified filesystem.
/mnt/ quotatest	Filesystem to which grace period settings will be applied.

604800	Grace period in seconds for blocks — 604800 seconds = 7 days.
604800	Grace period in seconds for inodes — 604800 seconds = 7 days.

```

root@Linux:~# setquota -u alice 10000 15000 1000 1200 /mnt/quotatest
root@Linux:~# setquota -u bob 20000 25000 2000 2500 /mnt/quotatest
root@Linux:~# setquota -u intern01 5000 7000 500 700 /mnt/quotatest
root@Linux:~# setquota -u -t 604800 604800 /mnt/quotatest

```

Simulate usage (write files)

```

root@Linux:~# sudo -u alice bash -c "dd if=/dev/zero of=/mnt/quotatest/
alice_test.img bs=1M count=105"

root@Linux:~# sudo -u bob bash -c "dd if=/dev/zero of=/mnt/quotatest/
bob_test.img bs=1M count=5"

root@Linux:~# sudo -u intern01 bash -c "dd if=/dev/zero of=/mnt/quotatest/
intern01_test.img bs=1M count=5"

```

Verify:

```

root@Linux:~# repquota /mnt/quotatest
=====
*** Report for user quotas on device /dev/loop16
Block grace time: 7days; Inode grace time: 7days
      Block limits            File limits
User      used  soft  hard  grace  used soft hard  grace
-----
root    --    24    0    0           3  0  0
alice   +- 15000 10000 15000 6days    1 1000 1200
bob     --   5120 20000 25000           1 2000 2500
intern01 +-  5120  5000  7000 6days    1  500  700
=====

```

Component	Explanation
repquota	A command-line utility used to report the disk usage and limits for users and groups on filesystems with quota enabled.
/mnt/quotatest	This is the mount point of the filesystem for which quotas are being reported. It must be a filesystem that has disk quotas enabled and

Task B – Set a group quota (unchanged)

acme: soft = 50 000 blocks, hard = 55 000 blocks
setquota -g acme 50000 55000 0 0 /mnt/quotatest

Component	Explanation
setquota	Command used to set disk quotas for users or groups on a Linux system. Requires quota to be enabled and mounted properly on the target filesystem.
-g	Flag that specifies the quota is for a group (as opposed to a user with -u).
acme	Name of the group for which the quota is being set.
50000	Soft block limit: Group acme can use up to 50,000 blocks (typically 1 block = 1 KB) before receiving warnings.
55000	Hard block limit: Absolute limit on disk usage. The group acme cannot exceed 55,000 blocks under any circumstance.
0	Soft inode limit: No soft limit set on the number of inodes (i.e., files).
0	Hard inode limit: No hard limit set on the number of inodes.
/mnt/quotatest	Filesystem mount point: The path where the quota is applied. Must be a mount point with group quota enabled in /etc/fstab.

Verify:

root@Linux:~# **repquota -g /mnt/quotatest**

*** Report for group quotas on device /dev/loop16

Block grace time: 7days; Inode grace time: 7days

Group		Block limits			grace	File limits		
		used	soft	hard		used	soft	hard
root	--	20	0	0	2	0	0	
alice	--	15000	0	0	1	0	0	
bob	--	5120	0	0	1	0	0	
intern01	--	5120	0	0	1	0	0	
acme	--	4	50000	55000	1	0	0	

Task C – Verify quotas (edited users)

User-centric views

`quota -uv alice`

`quota -uv bob`

`quota -uv intern01`

Global report

`repquota -a`

Command/Component	Explanation
quota	Displays disk usage and limits for a user or group.
-u	Specifies that the target is a user (as opposed to a group).
-v	Enables verbose output – shows both used and unused limits.
alice, bob, intern01	Username – the specific accounts whose disk quota information is being shown.
repquota	Summarizes quotas for all users and/or groups on all mounted filesystems.
-a	Means all – run the report on all filesystems with quotas enabled.

```
root@Linux:~# quota -uv alice
```

```
=====
Disk quotas for user alice (uid 1001):
      Filesystem  blocks    quota   limit   grace   files   quota   limit
grace
    /dev/loop16  15000*  10000   15000   6days        1   1000   1200
=====
```

```
root@Linux:~# quota -uv bob
```

```
=====
Disk quotas for user bob (uid 1002):
      Filesystem  blocks    quota   limit   grace   files   quota   limit
grace
    /dev/loop16    5120   20000   25000                1   2000   2500
=====
```



```
root@Linux:~# quota -uv intern01
```

```
=====
```

```
Disk quotas for user intern01 (uid 1003):
```

Filesystem	blocks	quota	limit	grace	files	quota	limit
grace							
/dev/loop16	5120*	5000	7000	6days	1	500	700

```
=====
```

```
root@Linux:~# repquota -a
```

```
=====
```

```
*** Report for user quotas on device /dev/loop16
```

```
Block grace time: 7days; Inode grace time: 7days
```

User		used	Block limits		grace	used	File limits		grace
			soft	hard			soft	hard	
root	--	24	0	0		3	0	0	
alice	+-	15000	10000	15000	6days	1	1000	1200	
bob	--	5120	20000	25000		1	2000	2500	
intern01	+-	5120	5000	7000	6days	1	500	700	

```
=====
```

Task D – Test soft- vs. hard-limit (use bob)

```
# Become bob
su - bob
```

```
cd /mnt/quotatest/projects
mkdir bob
dd if=/dev/zero of=bob/file1 bs=1M count=9 # ~9 000 KiB (< soft)
dd if=/dev/zero of=bob/file2 bs=1M count=19 # exceed soft but < hard
quota -v # see grace timer start
```

```
bob@Linux:~$ cd /mnt/quotatest/projects
```

```
bob@Linux:/mnt/quotatest/projects$ ls -l
```

```
total 0
```

```
bob@Linux:/mnt/quotatest/projects$ mkdir bob
```

```
bob@Linux:/mnt/quotatest/projects$ dd if=/dev/zero of=bob/file1 bs=1M
count=9
```

```
9+0 records in
9+0 records out
9437184 bytes (9.4 MB, 9.0 MiB) copied, 0.00803959 s, 1.2 GB/s
```

```
bob@Linux:/mnt/quotatest/projects$ quota -v
```

```
Disk quotas for user bob (uid 1002):
```

Filesystem	blocks	quota	limit	grace	files	quota	limit
grace							
/dev/loop16	14340	20000	25000		3	2000	2500

```
bob@Linux:/mnt/quotatest/projects$ dd if=/dev/zero of=bob/file21 bs=1M
count=19
```

```
dd: error writing 'bob/file21': Disk quota exceeded
```

```
11+0 records in
10+0 records out
10915840 bytes (11 MB, 10 MiB) copied, 0.0108202 s, 1.0 GB/s
```

```
=====
Disk quotas for user bob (uid 1002):
    Filesystem  blocks    quota   limit   grace   files   quota   limit
grace
    /dev/loop16 25000* 20000   25000   7days         4    2000   2500
=====
```