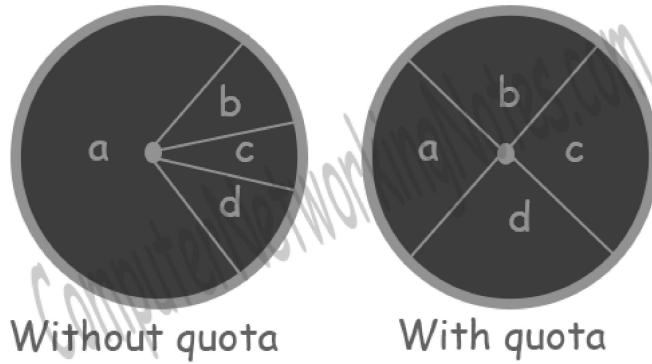


# How to manage disk quota in Linux step by step

This tutorial explains how to configure disk quota in Linux step by step with practical examples. Learn basic concepts of disk quota (Soft limit, Hard limit, Grace period, Block Size and Inode Number) and Linux disk quota management commands (quotacheck, edquota, quota, repquota, quotaoff and quotaon) in detail.

## Basic concepts of disk quota

If you are the only person who uses the disk, there is no need to implement quota at all. But if there are multiple users who use the same disk, quotas are the best ways to control the individual users from monopolizing entire disk space. A user limited by disk quotas cannot use additional disk space beyond his limit. For example suppose there are four users; user a, user b, user c and user d. Without quota any user can use entire disk space, leaving no space for other users. This situation is very common in shared environment such as web hosting, ISPs, file server, ftp server etc. But if disk quota is enabled, no user can use disk space beyond his limit.



## LAB Setup for disk quota practice

Although we can use a regular partition for practice, but if possible I suggest you to use a separate disk and create partition in that disk. If linux is installed in virtual system, you can add an additional disk for practice. If linux is installed in physical system, you can use a USB stick for practice.

*To learn how to add an additional disk in system and create partitions in that disk see the following tutorial which explains this process step by step with examples.*

*Manage Linux disk partition with Fdisk command*

For this tutorial, I assume that you have a separate partition or a partition which does not contain any important user data.

```
[root@server ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda        8:0    0 13G  0 disk
└─sda1     8:1    0 500M 0 part /boot
└─sda2     8:2    0 11.2G 0 part
  ├─rhel-root 253:0  0 9.8G 0 lvm /
  ├─rhel-swap 253:1  0 1000M 0 lvm [SWAP]
  └─rhel-home 253:2  0 500M 0 lvm /home
sdb        8:16   0  2G  0 disk
└─sdb1     8:17   0  2G 0 part We will perform all
[root@server ~]# fdisk -l /dev/sdb  exercises in this partition
```

```
Disk /dev/sdb: 2147 MB, 2147483648 bytes, 4194304 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x015d7a03
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		2048	4194303	2096128	83	Linux

During this practice we will execute commands which will overwrite exiting data with null characters. So make sure the partition you are going to use for practice does not contain any important user data.

We also need some user accounts and one group account to simulate the shared environment. Let's create four user accounts for practice.

Ads by Google

[Stop seeing this ad](#)[Why this ad? ⓘ](#)

```

Retype new password:
passwd: all authentication tokens updated successfully.
[root@server ~]# passwd b
Changing password for user b.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@server ~]# passwd c
Changing password for user c.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@server ~]# passwd d
Changing password for user d.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@server ~]#

```

Create a group **quotatest** and add user **c** and user **d** in that group.

```

[root@server ~]# groupadd quotatest Create group
[root@server ~]# usermod -g quotatest c add user C in group
[root@server ~]# usermod -g quotatest d add user D in group
[root@server ~]# grep quotatest /etc/group verify that both users
quotatest:x:1007:c,d are added successfully
[root@server ~]#
[root@server ~]# id c
uid=1005(c) gid=1007(quotatest) groups=1007(quotatest)
[root@server ~]# id d
uid=1006(d) gid=1007(quotatest) groups=1007(quotatest)
[root@server ~]#

```

Quota functionality is provided by **quota** package. To check whether this package is installed or not, use following commands

```
#rpm -qa quota
```

or

```
#yum list quota
```

For this tutorial, I assume that **quota** package is installed.

```

[root@server ~]# rpm -qa quota
quota-4.01-14.el7.x86_64
[root@server ~]# yum list quota
Loaded plugins: langpacks, product-id, search-disabled-repos,
: subscription-manager
This system is not registered to Red Hat Subscription Management. You can
use subscription-manager to register.
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makecache fast
Installed Packages
quota.x86_64          1:4.01-14.el7          @anaconda/7.3
[root@server ~]#

```

To learn, how to install a package use following tutorials which explain how to install and manage packages in linux step by step.

[How to configure yum Repository in Linux](#)

[RPM command in Linux Explained](#)

That's all setup we need for disk quota practice. Before we learn how to configure disk quota practically, let's understand two terms associated with disk quota.

## Block Size and Inode Number

## Happy Home Sale - Upto 45% Off

Choose the best sturdy wardrobes at Wakefit. 0% EMI. Free h delivery. CoD available

Wakefit

Shop ↗

Ads by Google

[Stop seeing this ad](#)[Why this ad? ⓘ](#)

If we want to control the size of files, we would configure the quota based on block size. If we want to control the number of files, we would configure the quota based on inode number. To control both, we would configure quota based on both block size and inode number.

It is highly recommended to configure quota based on both block size and inode number. If we skip any one method, a malicious user may use that method to abuse the system. Let's understand it with some examples.

#### **Situation 1 (Quota is configured only based on block size)**

1GB quota is configured based on block size for user **a**. Since quota for inode number is not configured, user can create files until entire 1GB space is not filled up. To abuse this system user can create relatively small size files. For example if he keeps file size only 1Kb, he can create 1000000 files (1 GB = 1000000Kb). 1000000 files means, 1000000 entries in inode table. This way only with 1GB space a user can make inode table unstable.

#### **Situation 2 (Quota is configured only based on inode table)**

100 inode numbers are configured as inode quota for user **a**. Since quota for block size is not configured, user can create 100 files (no matter how big or small in size they are). To abuse this system, user can create large size files. For example he can create a file of 1Tb in size. Yep, you read it right. Linux supports very big size files. For instance ext4 file system supports 16Tib individual file size. It means if disk is formatted with ext4 file system, we can create a single file of 16TiB in size. This way only 1 inode number is sufficient to fill up the entire disk space.

#### **Situation 3 (Quota is configured on both block size and inode number)**

1GB block size and 100 inode numbers are configured as quotas for user **a**. Since both block size and inode numbers are configured, user cannot abuse this system. No matter how small files in size he creates, he is not allowed to create more than 100 files. Just like this, no matter how big file in size he creates, he is not allowed to use more than 1GB disk space. As soon as 100 files are created, inode quota will block him from creating new file. Same way as soon as 1 GB space is consumed, block size quota will block him from using additional disk space. This way, if both block size and inode numbers are configured, user will not able to cheat the system.

## How to configure the disk quota

Disk quota can be configured in four steps

1. Enable quota
2. Remount file system
3. Create quota files
4. Configure quota policy

Let's understand each step in details

### Enabling quota

Linux uses `/etc/fstab` configuration file to mount all partitions in file system at boot time. This file contains all necessary information about the partition such as partition location at disk, mount point, attributes and other control options which are required to mount a partition. Each entry in this file has six fields.

#					
# /etc/fstab					
# Created by anaconda on Sun Mar 12 17:06:12 2017					
#					
# Accessible filesystems, by reference, are maintained under '/dev/disk'					
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info					
#					
/dev/mapper/rhel-root	/	xfs	defaults	0	0
UUID=5671c5f2-546e-4b5b-9baf-f731bc8df078	/boot	xfs	defaults	0	0
/dev/mapper/rhel-home	/home	xfs	defaults	0	0
/dev/mapper/rhel-swap	swap	swap	defaults	0	0
/dev/sdb1	/rhelab	ext4	defaults	1	2
1	2	3	4	5	6

Number	Filed	Description
1	What to mount	Device which we want to mount. We can use device name, UUID and label in this filed to represent the device.
2	Where to mount	The directory in main Linux File System where we want to mount the device.
3	File system	File system type of device.
4	Options	Mount options which control the mount process. To enable user quota add <b>usrquota</b> option and to enable group quota add <b>grpquota</b> option.

Ads by Google

[Stop seeing this ad](#)[Why this ad? ⓘ](#)

6	Automatic check	Whether this device should be checked while mounting or not. To disable use 0, to enable use 1 (for root partition) or 2 (for all partitions except root partition).
---	-----------------	--

In order to enable user quota, we have to add *usrquota* option in fourth field. Just like it, to enable group quota, we have to add *grpquota* option in fourth field. Let's enable both quotas for partition /dev/sdb1.

Following figure illustrates updated */etc/fstab* file

```
# /etc/fstab
# Created by anaconda on Sun Mar 12 17:06:12 2017
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/mapper/rhel-root    /          xfs      defaults        0  0
UUID=5671c5f2-546e-4b5b-9baf-f731bc8df078 /boot    xfs      defaults        0  0
/dev/mapper/rhel-home    /home     xfs      defaults        0  0
/dev/mapper/rhel-swap    swap      swap     defaults        0  0
/dev/sdb1      /rhcelab   ext4    defaults,usrquota,grpquota  1  2
```

Any changes made in */etc/fstab* file will not apply until next time system reboots. This is also applies on the disk quota options which we have recently added in this file. We have two choices here; either restart the system or remount the associated partition. Wherever possible we should always choose the first option. But in several situations immediate restart is not possible. In that case we can use second option.

## Remounting file system

If partition is not used by any process, we can remount it with following command.

```
#mount -o remount [partition]
```

Following figure illustrate this operation

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

If partition is remounted without any error or warning, use **mount | grep [partition]** command to confirm that quota options are successfully applied.

```
[root@server ~]# mount | grep /rhcelab
/dev/sdb1 on /rhcelab type ext4 (rw,relatime,seclabel,quota,usrquota,
grpquota,data=ordered)
[root@server ~]#
```

Some common reasons which trigger errors here are typing mistake in fstab file, mount point unavailable, file system is not formatted and wrong partition is selected. If there is any error, correct that before moving in next step.

## Creating quota files

In third step we will run following command.

```
#quotacheck -cug [partition where quota is enabled]
```

This command will create necessary files for quota. Let's understand this command in detail.

**quotacheck** :- This command is used to check the quota implementation in partition.

**c** :- This option is used to create the quota files in specified partition.

**u** :- This option is used to check the user quota.

**g** :- This option is used to check the group quota.

Basically this command will check quota entries in specified partition. If *aquota.user* and *aquota.group* files are not available in specified partition, it will create them.

```
File Edit View Search Terminal Help
[root@server ~]# quotacheck -cug /rhcelab
[root@server ~]# ls /rhcelab/
aquota.group  aquota.user  lost+found
[root@server ~]#
```

We need to run above command only once for each partition where we want to configure the disk quota.

Ads by Google

[Stop seeing this ad](#)[Why this ad? ⓘ](#)

**a** :- This option is used to check all quota enabled partitions

**v** :- This option is used to print real time updates as command proceeds

**u** :- This option is used to check user disk quota information

**g** :- This option is used to check group disk quota information

```
[root@server ~]# quotacheck -avug
quotacheck: Your kernel probably supports journaled quota but you are not using
it. Consider switching to journaled quota to avoid running quotacheck after an
unclean shutdown.
quotacheck: Scanning /dev/sdb1 [/rhcelab] done
quotacheck: Checked 2 directories and 2 files
[root@server ~]#
```

quota.user and aquota.group files are used to store quota configuration values for user and group respectively. Quota database keeps track of disk usage. How much space is allowed to a particular user is configured in quota.user file while how much space has been used by that user is tracked in quota database table. quota.user and aquota.group both are binary files which mean we cannot read or write them directly.

## Configuring quota policies

To configure quota policies, we have to define three values; soft limit, hard limit and grace period.

**Soft limit**:- This limit is flexible. User or group is allowed to cross this limit temporary.

**Hard limit**:- This is fixed limit. User or group is not allowed to cross this limit.

**Grace period**:- This is the time period in which user or group is allowed to use additional space beyond the soft limit.

To understand quota policies practically, let's create some dummy requirements.

User /Group	Block Size Soft Limit	Block Size Hard Limit	Grace period	Inode Soft Limit	Inode Hard limit	Grace period
a	100Mb	200Mb	2 Minutes	10	10	Nil
b	200Mb	200Mb	Nil	10	20	5 hours
quotatest	1000Mb	1500Mb	1 Day	100	150	5 days

**Nil**:- If both soft limit and hard limit are same, there is no need to configure this value.

To configure quota `edquota` command is used. To configure quota for user **a** use following command

```
#edquota a
```

Above command will open user quota configuration file.

```
[root@server ~]# edquota a
Disk quotas for user a (uid 1003):
Filesystem          blocks   soft   hard   inodes   soft   hard
  /dev/sdb1            0       0       0       0       0       0
```

This file has seven columns

Column	Name	Description
1	Filesystem	Partition where this quota will apply
2	blocks	Number of blocks currently used by this user
3	soft	Soft block size limit for user
4	hard	Hard block size limit for user
5	inodes	Number of inodes currently used by this user
6	soft	Soft inodes limit for user

^

Ads by Google

[Stop seeing this ad](#)[Why this ad? ⓘ](#)

Let's update this file

```
Disk quotas for user a (uid 1003):
Filesystem    blocks   soft    hard   inodes   soft    hard
/dev/sdb1        0 100000 200000     0      10      10
~
~:wq
[root@server ~]#
```

**Default block size is 1Kb. (1block = 1Kb).**

Following same way, configure the quota limit for user b

```
[root@server ~]# edquota b
Disk quotas for user b (uid 1004):
Filesystem    blocks   soft    hard   inodes   soft    hard
/dev/sdb1        0 200000 200000     0      10      20
~
~:wq
[root@server ~]#
```

Group quota is also defined in same manner. By default **edquota** command is used to set quota for users. To set quota for group we have to use **-g** option with this command. Let's define group quota for group *quotatest*.

#### edquota group

By default grace period is set to seven days. It means user or group will be able to use resources (block size or inodes after soft limit) till seven days. After seven days they will be denied to use any additional resources. We can adjust grace period as per our requirement.

To set grace period for user, use following command

```
edquota -T [username]
```

To set grace period for group, use following command

```
edquota -T -g [groupname]
```

To adjust global grace period, use following command

```
edquota -T
```

Following figure illustrates default configuration file.

```
Times to enforce softlimit
Time units may be: days, hours, minutes, or seconds
Filesystem          block grace      inode grace
/dev/sdb1            unset           unset
~ File system where quota is applied
~                  Grace period for block size
~                  for inodes
```

To define quota time period valid time units are days, hours, minutes and seconds.

Let's configure grace period for user a

```
[root@server ~]# edquota -T a
Times to enforce softlimit for user a (uid 1003):
Time units may be: days, hours, minutes, or seconds
Filesystem          block grace      inode grace
/dev/sdb1            2minutes       7days
~
~:wq
[root@server ~]#
```

Following same way configure grace period for user b

```
[root@server ~]# edquota -T b
Times to enforce softlimit for user b (uid 1004):
Time units may be: days, hours, minutes, or seconds
Filesystem          block grace      inode grace
/dev/sdb1            7days          5hours
~
~:wq
[root@server ~]#
```

← Ads by Google Stop seeing this ad Why this ad? ⓘ

```
[root@server ~]#
```

Never put space between value and unit for example "5 days" will be wrong entry, correct entry will be "5days". A space between value and unit or undefined value will generate ***edquota cannot read individual grace time from file*** error message.

Finally enable quota with following command

```
[root@server rhcelab]# quotaon /rhcelab/  
[root@server rhcelab]# █
```

## Testing disk quota

To verify disk quota setup, we can use following commands.

Command	Description
quota [user name]	To view quota uses by user
quota -g [group name]	To view quota uses by group
repquota -a	To view quota uses by all users and groups

System cannot generate quota reports, until user or group use the resources. If users or groups haven't used any block size or inode number, we will get following message.

Disk quotas for user [name] uid : none

Above message indicates that particular user or group has not used any quota resources (block size or inode) to display.

```
[root@server ~]# quota a
Disk quotas for user a (uid 1003): none
[root@server ~]# quota b
Disk quotas for user b (uid 1004): none
[root@server ~]# quota -g quotatest
Disk quotas for group quotatest (gid 1007): none
[root@server ~]# repquota -a
*** Report for user quotas on device /dev/sdb1
Block grace time: 7days; Inode grace time: 7days
                                         Block limits                               File limits
User          used    soft    hard   grace   used    soft    hard   grace
-----
root         --     20      0      0           2      0      0      0
```

[root@server ~]# █

```
[root@server ~]# mkdir /rhcelab/user-a
[root@server ~]# mkdir /rhcelab/user-b
[root@server ~]# mkdir /rhcelab/group-quotatest
[root@server ~]# chown -R a:a /rhcelab/user-a
[root@server ~]# chown -R b:b /rhcelab/user-b
[root@server ~]# chgrp -R quotatest /rhcelab/group-quotatest/
[root@server ~]# ls -l /rhcelab/
total 44
-rw-----. 1 root root      7168 Aug 22 19:44 aquota.group
-rw-----. 1 root root      7168 Aug 22 19:44 aquota.user
drwxr-xr-x. 2 root quotatest 4096 Aug 23 13:34 group-quotatest
drwx----- 2 root root    16384 Aug 22 19:42 lost+found
drwxr-xr-x. 2 a       a        4096 Aug 23 13:34 user-a
drwxr-xr-x. 2 b       b        4096 Aug 23 13:34 user-b
[root@server ~]#
```

## Quota configuration testing from user a

User *a* is allowed 100Mb disk space. He is also allowed to use additional 100Mb space for 2 minutes. He can create maximum 10 files or directories in this space.

To test this configuration switch to user **a** and change directory to `/rchelab`.

Ads by Google

[Stop seeing this ad](#)[Why this ad? ⓘ](#)

Now list the content and switch to *user-a* directory and create 5 directories and 4 files

```
[a@server rhcelab]$ ls -l
total 44
-rw----- 1 root root 7168 Aug 22 19:44 aquota.group
-rw----- 1 root root 7168 Aug 22 19:44 aquota.user
drwxr-xr-x 2 root quotatest 4096 Aug 23 13:34 group-quotatest
drwx----- 2 root root 16384 Aug 22 19:42 lost+found
drwxr-xr-x 2 a a 4096 Aug 24 13:14 user-a
drwxr-xr-x 2 b b 4096 Aug 23 13:34 user-b
[a@server rhcelab]$ cd user-a
[a@server user-a]$ mkdir {d1,d2,d3,d4,d5}
[a@server user-a]$ touch {f1,f2,f3,f4}
[a@server user-a]$ ls
d1 d2 d3 d4 d5 f1 f2 f3 f4
[a@server user-a]$
```

**1 + 5 = 6 directories**  
**4 files**  
**total inodes used = 10**

If files or directories name are supplied in {} bracket, they will be processed individually. The {} brackets is used to create multiple files or directories with single command.

As per our setup user **a** is allowed maximum 10 inodes and as above output shows, he has been used all allowed inodes. So he should not be able to create any new file or directory now. Let's test this restriction

```
[a@server user-a]$ mkdir d6
sdb1: write failed, user file limit reached.
mkdir: cannot create directory 'd6': Disk quota exceeded
[a@server user-a]$ touch f5
touch: cannot touch 'f5': Disk quota exceeded
[a@server user-a]$
```

As we can see in above output user **a** is not allowed to create any additional file or directory beyond his limit (10 inodes). This restriction confirms that our inodes quota configuration is properly setup and working as expected.

Now we will test block size configuration. Block size configuration has two limits; soft 100Mb and hard 200Mb. Soft limit can be extended for two minutes. Let's create a dummy data file to utilize all space defined in soft limit.

```
[a@server user-a]$ du -h *
4.0K d1
4.0K d2
4.0K d3
4.0K d4
4.0K d5
0 f1
0 f2
0 f3
0 f4
[a@server user-a]$ dd if=/dev/zero of=/rhcelab/user-a/f1 count=95 bs=1M
95+0 records in
95+0 records out
99614720 bytes (100 MB) copied, 0.43199 s, 231 MB/s
[a@server user-a]$ du -h *
4.0K d1
4.0K d2
4.0K d3
4.0K d4
4.0K d5
95M f1
0 f2
0 f3
0 f4
[a@server user-a]$
```

I used **dd** command to copy the 95Mb null bytes in *f1* file.

As we can see in above output user **a** is allowed to add any length of data in file until he remains under the soft limit. Before we test the soft limit and grace period, open an another terminal and check the current uses of user **a**

Ads by Google

[Stop seeing this ad](#) [Why this ad? ⓘ](#)

User	used	Block limits			File limits		
		soft	hard	grace	used	soft	hard
root	--	24	0	0		3	0
a	--	97304	100000	200000		10	10

Until soft limit is not crossed,  
no value will be displayed here

[root@server ~]#

As we can see in above output user a has been used all allowed inodes. So far block size is concerned, he still has 2629 blocks (100000 - 97304) available under his soft limit.

Did you notice there is no value listed in grace period field while we configured this value also?

Grace period is just like a timer which will start only when soft limit is crossed and user still has 2629 blocks in his soft limit. To see it practically, let's cross the soft limit

```
[root@server ~]# su a
[a@server root]$ cd /rhcelab/user-a
[a@server user-a]$ date
Fri Aug 25 18:24:33 IST 2017
[a@server user-a]$ dd if=/dev/zero of=/rhcelab/user-a/f2 count=50 bs=1M
sdb1: warning, user block quota exceeded.
50+0 records in
50+0 records out
52428800 bytes (52 MB) copied, 1.31406 s, 39.9 MB/s
[a@server user-a]$ exit
exit
[root@server ~]# date
Fri Aug 25 18:25:12 IST 2017
[root@server ~]# quota a
Disk quotas for user a (uid 1003):
  Filesystem blocks  quota  limit   grace   files  quota  limit   grace
    /dev/sdb1  148504* 100000  200000  00:02      10*    10     10
[root@server ~]#
```

As soon as user crosses the soft limit  
grace period timer starts

As we can see in above output as soon as user crossed the soft limit, grace period timer started. User is allowed to use additional space until this timer keeps running. Once timer is stopped he will be denied from using any additional space. Right now user still has 51496 (200000-148504) blocks available, before it reaches to hard limit. Let's use additional 10Mb space.

```
[root@server ~]# quota a
Disk quotas for user a (uid 1003):
  Filesystem blocks  quota  limit   grace   files  quota  limit   grace
    /dev/sdb1  148504* 100000  200000  00:02      10*    10     10
[root@server ~]# su a
[a@server root]$ cd /rhcelab/user-a
[a@server user-a]$ dd if=/dev/zero of=/rhcelab/user-a/f3 count=10 bs=1M
10+0 records in
10+0 records out
10485760 bytes (10 MB) copied, 0.251022 s, 41.8 MB/s
[a@server user-a]$
```

As we can see in above output grace period timer is running and hard limit is not crossed, so additional 10Mb space is allowed. Now let the grace period expire and try to use additional 10Mb space from remaining space.

```
[a@server user-a]$ exit
exit
[root@server ~]# quota a
Disk quotas for user a (uid 1003):
  Filesystem blocks  quota  limit   grace   files  quota  limit   grace
    /dev/sdb1  158744* 100000  200000  none      10*    10     10
[root@server ~]# su a
[a@server root]$ cd /rhcelab/user-a
[a@server user-a]$ dd if=/dev/zero of=/rhcelab/user-a/f4 count=10 bs=1M
sdb1: write failed, user block quota exceeded too long.
dd: error writing '/rhcelab/user-a/f4': Disk quota exceeded
1+0 records in
0+0 records out
0 bytes (0 B) copied, 0.00264897 s, 0.0 kB/s
[a@server user-a]$
```

As we can see in above output user is not allowed to use additional space even he has 41256 (200000-158744) blocks available. To use this remaining 41256 blocks he also needs time in grace period which is already expired.



Ads by Google

[Stop seeing this ad](#)[Why this ad? ⓘ](#)

```
[a@server root]$ cd /rhcelab/user-a
[a@server user-a]$ rm f1 as soon as user brings consumption below the soft limit
[a@server user-a]$ exit grace period timer will be removed
exit
[root@server ~]# quota a
Disk quotas for user a (uid 1003):
  Filesystem blocks quota limit grace files quota limit grace
    /dev/sdb1 61464 100000 200000          9     10     10
[root@server ~]# su a
[a@server root]$ cd /rhcelab/user-a
[a@server user-a]$ dd if=/dev/zero of=/rhcelab/user-a/f1 count=100 bs=1M
sdb1: warning, user block quota exceeded. as soon as soft limit is crossed,
100+0 records in
100+0 records out
104857600 bytes (105 MB) copied, 2.29392 s, 45.7 MB/s
[a@server user-a]$ █
[root@server ~]# quota a
Disk quotas for user a (uid 1003):
  Filesystem blocks quota limit grace files quota limit grace
    /dev/sdb1 148504* 100000 200000 00:02   10*     10     10
```

#### Key points

If soft limit and hard limit are same, grace period is not required.

If soft limit and hard limit are different, grace period is required.

Soft limit must be configured lower than hard limit.

As soon as user crosses his soft limit grace period timer starts.

User is allowed to use additional space (*hard limit – soft limit*) until grace period timer is running.

Once grace period is expired, user is not allowed to use additional space.

Grace period timer will be removed automatically once user brings his consumption below the soft limit.

## Quota configuration testing from user b

User **b** is allowed 200Mb block size and 20 inodes with soft limit 10 inodes and 5 hours grace period. Since both soft and hard limits are same for block size, grace period is not configured.

We can test this setup with following steps.

### Block size testing

Switch to user **b** and change directory to `/rhcelab/user-b`.

Create a file of 195Mb in size with following command

```
#dd if=/dev/zero of=/rhcelab/user-b/file1 count=195 bs=1M
```

Exit from user **b** and verify block size quota uses with following command

```
#quota b
```

Switch to user **b** again and try to create a file 10Mb in size with following command

```
#dd if=/dev/zero of=/rhcelab/user-b/file2 count=10 bs=1M
```

If this time user is denied, block size quota configuration is setup correctly. If user is allowed to create this file, block size quota configuration is not setup properly.

### Inodes number testing

Switch to user **b** and change directory to `/rhcelab/user-b`

Create 10 directories with following command

```
#mkdir {d1,d2,d3,d4,d5,d6,d7,d8}
```

Exit from user **b** and verify inodes quota uses with following command

```
#quota b
```

Switch to user **b** again and try to create one more directory with following command

```
#mkdir d9
```

User should be allowed to create directory but this time he should get ***disk quota exceeds*** warning message.

Exit from user **b** and check inodes number quota again

Ads by Google

[Stop seeing this ad](#)[Why this ad? ⓘ](#)

## Quota configuration testing from group quotatest

The group quotatest has soft limit 1000Mb and hard limit 1500Mb with grace period of 1 day for block size. For inodes it has soft limit of 100 inodes and hard limit of 150 inodes with 5 days grace period.

Before you start testing from group, make sure that user **c** and **d** are the members of group and group has proper permission on testing folder.

```
[root@server ~]# grep quotatest /etc/group
quotatest:x:1007:c,d
[root@server ~]# id c
uid=1005(c) gid=1007(quotatest) groups=1007(quotatest)
[root@server ~]# id d
uid=1006(d) gid=1007(quotatest) groups=1007(quotatest)
[root@server ~]# chmod -R 770 /rhcelab/group-quotatest/
[root@server ~]# ls -l /rhcelab
total 44
-rw----- 1 root root 7168 Aug 25 16:29 aquota.group
-rw----- 1 root root 7168 Aug 25 16:29 aquota.user
drwxrwx--- 2 root quotatest 4096 Aug 25 21:12 group-quotatest
drwx---- 2 root root 16384 Aug 25 16:03 lost+found
drwxr-xr-x 7 a root 4096 Aug 25 18:30 user-a
drwxr-xr-x 2 b root 4096 Aug 25 18:06 user-b
[root@server ~]#
```

Switch to user **c** and create a file (800Mb in size) and a directory. In directory creates 80 empty files with following command.

```
#touch test_{1..80}.txt
```

Verify files and directories with **ls** command.

```
[root@server ~]# su c
[c@server root]$ cd /rhcelab/group-quotatest/
[c@server group-quotatest]$ dd if=/dev/zero of=/rhcelab/group-quotatest/c-file-1 count=800
bs=1M
800+0 records in
800+0 records out
838860800 bytes (839 MB) copied, 7.77966 s, 108 MB/s
[c@server group-quotatest]$ mkdir c-dir
[c@server group-quotatest]$ cd c-dir/
[c@server c-dir]$ touch test_{1..80}.txt
[c@server c-dir]$ ls
d{1..80} test_1.txt test_2.txt test_3.txt test_4.txt test_5.txt test_6.txt
test_7.txt
test_10.txt test_20.txt test_30.txt test_40.txt test_50.txt test_60.txt test_70.txt
test_80.txt
test_11.txt test_21.txt test_31.txt test_41.txt test_51.txt test_61.txt test_71.txt
test_8.txt
test_12.txt test_22.txt test_32.txt test_42.txt test_52.txt test_62.txt test_72.txt
test_9.txt
test_13.txt test_23.txt test_33.txt test_43.txt test_53.txt test_63.txt test_73.txt
test_14.txt test_24.txt test_34.txt test_44.txt test_54.txt test_64.txt test_74.txt
test_15.txt test_25.txt test_35.txt test_45.txt test_55.txt test_65.txt test_75.txt
test_16.txt test_26.txt test_36.txt test_46.txt test_56.txt test_66.txt test_76.txt
test_17.txt test_27.txt test_37.txt test_47.txt test_57.txt test_67.txt test_77.txt
test_18.txt test_28.txt test_38.txt test_48.txt test_58.txt test_68.txt test_78.txt
test_19.txt test_29.txt test_39.txt test_49.txt test_59.txt test_69.txt test_79.txt
[c@server c-dir]$
```

Exit from user **c** and verify quota limit.

```
[c@server c-dir]$ exit
exit
[root@server ~]# quota -g quotatest
Disk quotas for group quotatest (gid 1007):
  Filesystem    blocks   quota   limit   grace   files   quota   limit   grace
    /dev/sdb1    819212  1000000  1500000          84      100     150
[root@server ~]#
```

Now login from user **d** and create a file 400Mb in size to cross the soft limit of block size. To cross the soft limit of inodes create 30 empty directories. As soon as user would cross the soft limit, he should get warning message for related quota limit.





Ads by Google

[Stop seeing this ad](#)[Why this ad? ⓘ](#)

```
314572800 bytes (315 MB) copied, 2.05164 s, 153 MB/s
[d@server group-quotatest]$ mkdir d{1..30}
sdb1: warning, group file quota exceeded. This warning message confirms
[d@server group-quotatest]$ ls                               that inodes number quota is set up correctly
c-dir    d10   d13   d16   d19   d21   d24   d27   d3    d5   d8
c-file-1 d11   d14   d17   d2   d22   d25   d28   d30   d6   d9
d1      d12   d15   d18   d20   d23   d26   d29   d4    d7   d-file-1
[d@server group-quotatest]$
```

Exit from user **d** and view the quota uses for group. Grace period timer should be started for both limits.

```
[d@server group-quotatest]$ exit
exit
[root@server ~]# quota -g quotatest
Disk quotas for group quotatest (gid 1007):
  Filesystem blocks  quota  limit  grace   files  quota  limit  grace
    /dev/sdb1 1126532* 1000000 1500000  1day     115*   100    150  5days
[root@server ~]#
```

Above output confirms that grace period for group quota is also configured successfully.

## Important commands for quota management

Command	Description
quotacheck	This command is used to check quota implementation and update quota database from file system. This command is also used to create aquota.user and aquota.group files, if they are not created manually.
edquota	This command is used to configure quota values for user and group.
quota	This command is used to view the quota uses for specific user or group.
repquota	This command is used to view the quota uses for all users and group.
quotaoff	This command is used to turnoff quota temporary.
quotaon	This command is used to enable quota again if it is disabled.

In this tutorial we learned common file system disk management step by step with examples. Usually this process should work on maximum file systems; however some file systems such as xfs have their own quota management tools. So if this approach does not work as expected, please check the manual page of corresponding file system to figure out, how that particular file system works with disk quota.

That's all for this tutorial. In next tutorial I will explain another linux topic in details with examples.

By ComputerNetworkingNotes Updated on 2021-06-25 10:04:38 IST

---

ComputerNetworkingNotes    Linux Tutorials    How to manage disk quota in Linux step by step

How to Configure Printer Server in Linux

How to configure DNS Server in Linux

We do not accept any kind of Guest Post. Except Guest post submission, for any other query (such as advertising opportunity, product advertisement, feedback, suggestion, error reporting and technical issue) or simply just say to hello mail us    ComputerNetworkingNotes@gmail.com

Computer Networking Notes and Study Guides © 2022. All Rights Reserved.

About Privacy Policy Terms and Conditions

