EPAM University Programs

DevOps external course

Module 4 Linux & Bash Essentials

TASK 4.7

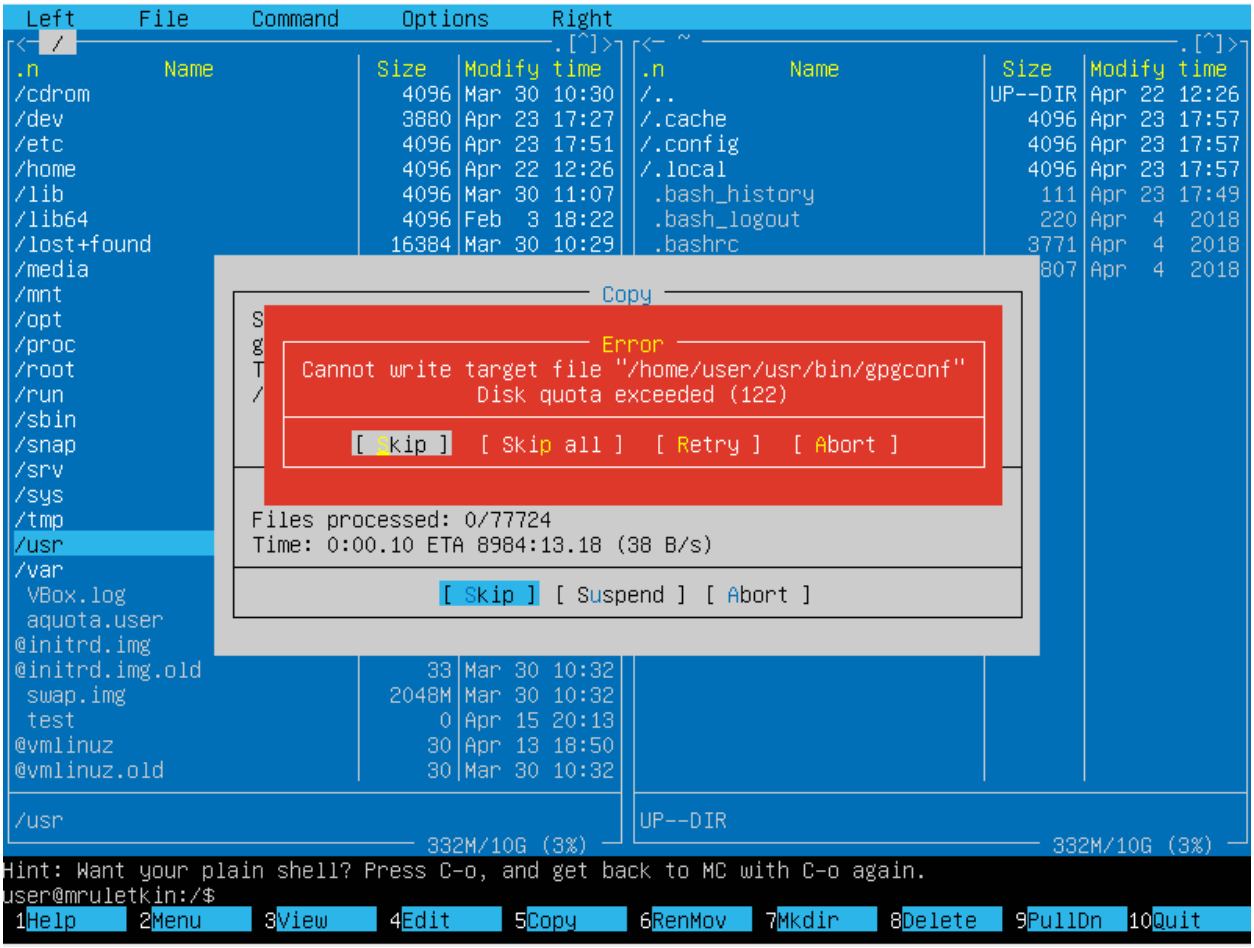
Part1. **Quota allocation mechanism.**

Employing commands from presentation #4.6, create a new user, say, *utest*. Based on the quota mechanism, limit the available disk space for this user to **soft**: 100M and **hard**: 150M.

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/4fbf83d5-6513-4726-8f38-3a4011f7e5de / ext4 defaults,usrquota     0 0
/swap.img       none    swap    sw      0 0
```

```
root@mruletkin:~# mount -o remount /
root@mruletkin:~# mount | grep quota
/dev/sda2 on / type ext4 (rw,relatime,quota,usrquota,data=ordered)
root@mruletkin:~# quotacheck -favugm
quotacheck: Your kernel probably supports journaled quota but you are not using it. Consider switchi
ng to journaled quota to avoid running quotacheck after an unclean shutdown.
quotacheck: Scanning /dev/sda2 [/] done
quotacheck: Old group file name could not been determined. Usage will not be subtracted.
quotacheck: Checked 21163 directories and 103754 files
root@mruletkin:~# quotaon -avug
quotaon: using //aquota.user on /dev/sda2 [/]: Device or resource busy
root@mruletkin:~# edquota -u user
```

```
Disk quotas for user user (uid 1002):
  Filesystem                  blocks       soft       hard     inodes     soft     hard
  /dev/sda2                       20        100        150          5        0        0
```



Then, using Midnight Commander (since MC shows warnings about exceeding the limits of available to a user disk space), copy content of /usr directory to utest's home directory (actually, /usr isn't mandatory, you are free to copy any other data, the only condition is sufficient total size of the files to copy).

Note: if /home is not a mount point, then the **mount** and **quotaon** commands should be called with respect to the root partition /.

Note 2: Please, put into your report screenshots of your terminal window with the executed commands, along with screenshots of MC panels over which quota warnings are shown (i.e. warnings about exceeding soft and hard limits).

Part2. **Access Control Lists, ACLs**

In what follows, we assume that there are two users: *guest* (included into the list of sudoers) and *utest*. None of the users is the superuser (i.e. UIDs of the users differ from 0).

The most task: to allow user *utest* visit *guest*'s home directory.

The average task: to acquaint yourself with the basics of ACL and verify the fact that ACL privileges override the **chmod** ones.

Before proceeding to the task execution, please, visit the linux.org page describing ACL, https://linuxconfig.org/how-to-manage-acls-on-linux.

Every step of execution should be stored into some file **/var/log** directory (use logger, please).

1. Based on given in presentation #4.7 instructions, turn on and set up the ACL. *Caution*! The fact that a file system has been mounted with the "acl" flag on by default, doesn't mean that the ACL package is installed.

Prior to any action, it is advised to check if the "acl" flag is on, using

tune2fs -l /dev/sda*

(a particular name of the device file sda*, is to be determined by calling to **blkid**, invoke it twice:

(i) on behalf of *guest* (i.e. without the superuser privileges);

(ii) with **sudo** (i.e. with the superuser privileges). Note the level of details provided by different **blkid** outputs).

2. Log in as *guest*. Create in */tmp* a directory called *acl_test*. By means of **chmod,** allow user utest to perform all possible operations (rwx) with respect to *acl_test.*

```
mykhailo_litvinov@mruletkin:/tmp$ tail /var/log/syslog
Apr 25 11:50:27 mruletkin systemd[2151]: Reached target Default.
Apr 25 11:50:27 mruletkin systemd[2151]: Startup finished in 31ms.
Apr 25 11:55:32 mruletkin mykhailo_litvinov: sudo tune2fs -l /dev/sda2
Apr 25 11:56:18 mruletkin kernel: [  574.930189] EXT4-fs (sda2): re-mounted. Opts: acl
Apr 25 11:56:45 mruletkin mykhailo_litvinov: sudo mount -o remount -o acl /dev/sda2
Apr 25 11:59:58 mruletkin mykhailo_litvinov: mkdir acl_test
Apr 25 12:00:05 mruletkin mykhailo_litvinov: chmod o+rwx acl_test/
```

Verify that user *utest* is indeed capable of implementing granted him (her) privileges. For example, acer logging in as *utest*, create a file in */tmp/acl_test*, say, *utest.txt* with the aid of **touch**. Query information about the directory and file by calling to

ls -ld /tmp/acl_test

ls -l /tmp/acl_test

To check ACL permissions do:

ge4acl /tmp/acl_test

ge4acl /tmp/acl_test/utest.txt

```
mykhailo_litvinov@mruletkin:/tmp$ su user
Password:
user@mruletkin:/tmp$ cd
user@mruletkin:~$ ls -ld /tmp/acl_test
drwxrwxrwx 2 mykhailo_litvinov mykhailo_litvinov 4096 Apr 25 12:21 /tmp/acl_test
user@mruletkin:~$ ls -l /tmp/acl_test
total 0
-rw-rw-r-- 1 user user 0 Apr 25 12:22 utest.txt
user@mruletkin:~$ getfacl /tmp/acl_test
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test
# owner: mykhailo_litvinov
# group: mykhailo_litvinov
user::rwx
group::rwx
other::rwx

user@mruletkin:~$ getfacl /tmp/acl_test/utest.txt
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test/utest.txt
# owner: user
# group: user
user::rw-
group::rw-
other::r--

user@mruletkin:~$
```

```
Apr 25 12:23:04 mruletkin mykhailo_litvinov: ls -ld /tmp/acl_test
Apr 25 12:23:15 mruletkin mykhailo_litvinov: ls -l /tmp/acl_test
Apr 25 12:23:50 mruletkin mykhailo_litvinov: getfacl /tmp/acl_test
Apr 25 12:24:23 mruletkin mykhailo_litvinov: getfacl /tmp/acl_test/utest.txt
mykhailo_litvinov@mruletkin:/tmp$ _
```

3. Employ ACL to block any activity except for reading, for user *utest* with respect to directory */tmp/acl_test* (hint: use **se4acl**).

```
mykhailo_litvinov@mruletkin:/tmp$ setfacl -d -m u:user:r /tmp/acl_test
mykhailo_litvinov@mruletkin:/tmp$ getfacl /tmp/acl_test
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test
# owner: mykhailo_litvinov
# group: mykhailo_litvinov
user::rwx
group::rwx
other::rwx
default:user::rwx
default:user:user:r--
default:group::rwx
default:mask::rwx
default:other::rwx
```

```
Apr 25 13:03:41 mruletkin mykhailo_litvinov: setfacl -d -m u:user:r /tmp/acl_test
Apr 25 13:03:51 mruletkin mykhailo_litvinov: getfacl /tmp/acl_test
mykhailo_litvinov@mruletkin:/tmp$ _
```

Test if the actions are effectively prohibited

touch /tmp/acl_test/prohibited.txt

Is it possible to invoke this command?

echo “new content” > /tmp/acl_test/utest.txt

Test if user *utest* can be prevented from modifying content of the file *utest.txt* by means of ACL. (Note that user *utest* is the owner of the file *tmp/acl_test/utest.txt*).

```
user@mruletkin:~$ touch /tmp/acl_test/prohibited.txt
touch: cannot touch '/tmp/acl_test/prohibited.txt': Permission denied
user@mruletkin:~$ echo "new content">/tmp/acl_test/utest.txt
bash: /tmp/acl_test/utest.txt: Permission denied
user@mruletkin:~$ _
```

```
Apr 25 13:40:01 mruletkin mykhailo_litvinov: touch /tmp/acl_test/prohibited.txt
Apr 25 13:40:23 mruletkin mykhailo_litvinov: echo new content>/tmp/acl_test/utest.txt
mykhailo_litvinov@mruletkin:/tmp/acl_test$ _
```

4. Consider a situation when at the ACL level user *utest* is allowed to have all possible privileges with respect to */tmp/acl_test*, while no ac=on is allowed with **chmod** (conventional mechanism). (Hint: repeat step 3, but given the new context).

```
mykhailo_litvinov@mruletkin:/tmp/acl_test$ chmod o-rwx /tmp/acl_test
mykhailo_litvinov@mruletkin:/tmp/acl_test$ setfacl -m u:user:rwx /tmp/acl_test
mykhailo_litvinov@mruletkin:/tmp/acl_test$ logger "chmod o-rwx /tmp/acl_test"
mykhailo_litvinov@mruletkin:/tmp/acl_test$ logger "setfacl -m u:user:rwx /tmp/acl_test"
mykhailo_litvinov@mruletkin:/tmp/acl_test$ su user
Password:
user@mruletkin:/tmp/acl_test$ cd
user@mruletkin:~$ touch /tmp/acl_test/prohibited.txt
user@mruletkin:~$ echo "new content">/tmp/acl_test/utest.txt
user@mruletkin:~$ logger touch /tmp/acl_test/prohibited.txt
user@mruletkin:~$ logger "echo "new content">/tmp/acl_test/utest.txt"
user@mruletkin:~$
```

5. For user *utest*, set default ACLs to the directory */tmp/acl_test* which allow read-only access (hint: use the -d option of the **se4acl** command). Being logged in as *utest*, invoke **touch** to create the file *utest2.txt* in the */tmp/acl_test* directory. Query permissions on this file using **ge4acl**.

```
mykhailo_litvinov@mruletkin:/tmp/acl_test$ setfacl -d -m u:user:r /tmp/acl_test
mykhailo_litvinov@mruletkin:/tmp/acl_test$ su user
Password:
user@mruletkin:/tmp/acl_test$ touch utest2.txt
user@mruletkin:/tmp/acl_test$ getfacl utest2.txt
# file: utest2.txt
# owner: user
# group: user
user::rw-
user:user:r--
group::rwx                      #effective:rw-
mask::rw-
other::rw-

user@mruletkin:/tmp/acl_test$
```

```
mykhailo_litvinov@mruletkin:/tmp/acl_test$ tail -3 /var/log/syslog
Apr 25 13:57:10 mruletkin mykhailo_litvinov: setfacl -d -m u:user:r /tmp/acl_test
Apr 25 13:57:29 mruletkin mykhailo_litvinov: touch utest2.txt
Apr 25 13:57:37 mruletkin mykhailo_litvinov: getfacl utest2.txt
mykhailo_litvinov@mruletkin:/tmp/acl_test$ _
```

6. Set the maximum permissions mask on the */tmp/acl_test/utest.txt* file in such a way as to allow read-only access. Check permissions with **ge4acl**.

```
user@mruletkin:/tmp/acl_test$ setfacl -m u:user:r-- utest2.txt
user@mruletkin:/tmp/acl_test$ getfacl utest2.txt
# file: utest2.txt
# owner: user
# group: user
user::rw-
user:user:r--
group::rwx
mask::rwx
other::rw-

user@mruletkin:/tmp/acl_test$ _
```

7. Delete all ACL entries relative to the */tmp/acl_test* directory.

```
Apr 25 14:04:11 mruletkin mykhailo_litvinov: setfacl -m u:user:r-- utest2.txt
Apr 25 14:04:18 mruletkin mykhailo_litvinov: getfacl utest2.txt
Apr 25 14:06:13 mruletkin mykhailo_litvinov: setfacl -x u:user: /tmp/acl_test
mykhailo_litvinov@mruletkin:~$
```