EPAM University Programs
DevOps external course
Module 4 Linux & Bash Essentials
TASK 4.7

Danylenko Homework

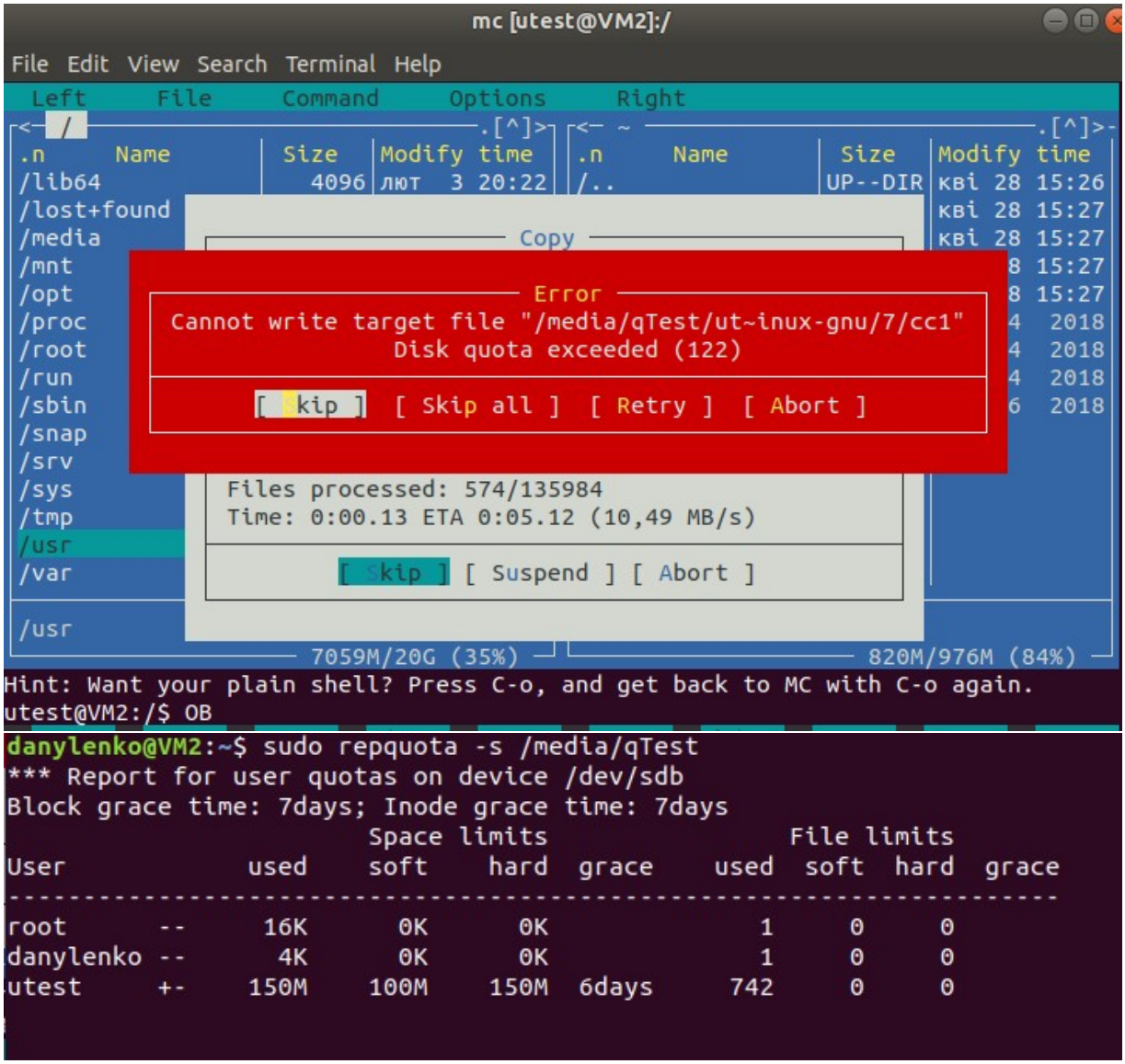
Part1. **Quota allocation mechanism.**

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.
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).

```
danylenko@VM2:~$ mount | grep "sda\|sdb"
/dev/sda1 on / type ext4 (rw,relatime,errors=remount-ro)
/dev/sdb on /media/qTest type ext4 (rw,relatime,quota,usrquota,grpquota)
danylenko@VM2:~$ sudo groupadd utest
danylenko@VM2:~$ sudo useradd -g utest -s /bin/bash -d /media/qTest/utest -m utest
danylenko@VM2:~$ sudo passwd utest
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

```
danylenko@VM2:~$ sudo quotacheck -ugm /media/qTest
danylenko@VM2:~$ ls /media/qTest
aquota.group  aquota.user  lost+found  utest
danylenko@VM2:~$ sudo quotaon -v /media/qTest
/dev/sdb [/media/qTest]: group quotas turned on
/dev/sdb [/media/qTest]: user quotas turned on
```

```
danylenko@VM2:~$ sudo setquota -u utest 100M 150M 0 0 /media/qTest
danylenko@VM2:~$ sudo quota -vs utest
Disk quotas for user utest (uid 1002):
     Filesystem   space   quota   limit   grace   files   quota   limit   grace
       /dev/sdb     28K    100M    150M              5       0       0
danylenko@VM2:~$ sudo repquota -s /media/qTest
*** Report for user quotas on device /dev/sdb
Block grace time: 7days; Inode grace time: 7days
                        Space limits                File limits
User            used    soft    hard  grace    used  soft  hard  grace
----------------------------------------------------------------------
root      --     16K      0K      0K              1     0     0
danylenko --      4K      0K      0K              1     0     0
utest     --     28K    100M    150M              5     0     0
```




```
danylenko@VM2:~$ sudo repquota -s /media/qTest
*** Report for user quotas on device /dev/sdb
Block grace time: 7days; Inode grace time: 7days
                        Space limits                File limits
User            used    soft    hard  grace    used  soft  hard  grace
----------------------------------------------------------------------
root      --     16K      0K      0K              1     0     0
danylenko --      4K      0K      0K              1     0     0
utest     +-    150M    100M    150M  6days     742     0     0
```

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.

```
utest@VM2:~$ date
вівторок, 28 квітня 2020 20:38:41 +0300
utest@VM2:~$ cd /home/danylenko
-bash: cd: /home/danylenko: Permission denied
utest@VM2:~$ date
вівторок, 28 квітня 2020 20:39:06 +0300
utest@VM2:~$ cd /home/danylenko
utest@VM2:/home/danylenko$
```

```
danylenko@VM2:~$ date
вівторок, 28 квітня 2020 20:38:48 +0300
danylenko@VM2:~$ setfacl -m u:utest:rx /home/danylenko
```

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).
To store both command and output I decided to put command into variable, and echo then eval this variable with pipe storing both into logger. Then to look at output I filtered syslog with grep by date, sometimes date and username, in the end of task used current hh:mm with date command. sleep 5s to wait until records appear into log before showing them.
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);
I had same output without sudo as with sudo.

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

```
danylenko@VM2:~$ cmd="blkid | grep sd"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -3
 /var/log/syslog
Apr 28 18:21:43 VM2 danylenko: blkid | grep sd
Apr 28 18:21:43 VM2 danylenko: /dev/sda1: UUID="7702faec-afd1-4c8b-9460-90dc0a28
371f" TYPE="ext4" PARTUUID="91a224bc-01"
Apr 28 18:21:43 VM2 danylenko: /dev/sdb: LABEL="quotes" UUID="55db64a3-3314-4774
-be52-a0dbdc474ba3" TYPE="ext4"
```

```
danylenko@VM2:~$ cmd="sudo tune2fs -l /dev/sda1 | grep acl"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -2
 /var/log/syslog
Apr 28 18:23:43 VM2 danylenko: sudo tune2fs -l /dev/sda1 | grep acl
Apr 28 18:23:43 VM2 danylenko: Default mount options:    user_xattr acl
```

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.* Verify that user *utest* is indeed capable of implementing granted him (her) privileges. For example, 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

```
danylenko@VM2:~$ cmd="mkdir /tmp/acl_test; chmod o=+rwx -v /tmp/acl_test"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -3
0 /var/log/syslog | grep "18:31"
Apr 28 18:31:49 VM2 danylenko: mkdir /tmp/acl_test; chmod o=+rwx -v /tmp/acl_tes
t
Apr 28 18:31:49 VM2 danylenko: mode of '/tmp/acl_test' changed from 0755 (rwxr-x
r-x) to 0757 (rwxr-xrwx)
```

```
utest@VM2:~$ touch /tmp/acl_test/utest.txt
utest@VM2:~$
```

ls -ld /tmp/acl_test

ls -l /tmp/acl_test

```
danylenko@VM2:~$ cmd="ls -ld /tmp/acl_test"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -30 /
var/log/syslog | grep "18:42"
Apr 28 18:42:36 VM2 danylenko: ls -ld /tmp/acl_test
Apr 28 18:42:36 VM2 danylenko: drwxr-xrwx 2 danylenko danylenko 4096 квi 28 18:40 /
tmp/acl_test
danylenko@VM2:~$ cmd="ls -l /tmp/acl_test"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -30 /
var/log/syslog | grep "18:42"
Apr 28 18:42:36 VM2 danylenko: ls -ld /tmp/acl_test
Apr 28 18:42:36 VM2 danylenko: drwxr-xrwx 2 danylenko danylenko 4096 квi 28 18:40 /
tmp/acl_test
Apr 28 18:42:58 VM2 danylenko: ls -l /tmp/acl_test
Apr 28 18:42:58 VM2 danylenko: total 0
Apr 28 18:42:58 VM2 danylenko: -rw-rw-r-- 1 utest utest 0 квi 28 18:40 utest.txt
danylenko@VM2:~$
```

To check ACL permissions do:

getfacl /tmp/acl_test

getfacl /tmp/acl_test/utest.txt

```
danylenko@VM2:~$ cmd="getfacl /tmp/acl_test"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger)
getfacl: Removing leading '/' from absolute path names
danylenko@VM2:~$ cmd="getfacl /tmp/acl_test/utest.txt"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -30 /
var/log/syslog | grep "18:49"
getfacl: Removing leading '/' from absolute path names
Apr 28 18:49:03 VM2 danylenko: getfacl /tmp/acl_test
Apr 28 18:49:03 VM2 danylenko: # file: tmp/acl_test
Apr 28 18:49:03 VM2 danylenko: # owner: danylenko
Apr 28 18:49:03 VM2 danylenko: # group: danylenko
Apr 28 18:49:03 VM2 danylenko: user::rwx
Apr 28 18:49:03 VM2 danylenko: group::r-x
Apr 28 18:49:03 VM2 danylenko: other::rwx
Apr 28 18:49:03 VM2 danylenko:
Apr 28 18:49:25 VM2 danylenko: getfacl /tmp/acl_test/utest.txt
Apr 28 18:49:25 VM2 danylenko: # file: tmp/acl_test/utest.txt
Apr 28 18:49:25 VM2 danylenko: # owner: utest
Apr 28 18:49:25 VM2 danylenko: # group: utest
Apr 28 18:49:25 VM2 danylenko: user::rw-
Apr 28 18:49:25 VM2 danylenko: group::rw-
Apr 28 18:49:25 VM2 danylenko: other::r--
Apr 28 18:49:25 VM2 danylenko:
```

3. Employ ACL to block any activity except for reading, for user *utest* with respect to directory */tmp/acl_test* (hint: use **setfacl**). Test if the actions are effectively prohibited

```
danylenko@VM2:~$ cmd="setfacl -m u:utest:r /tmp/acl_test"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -30 /
var/log/syslog | grep "18:54"
Apr 28 18:54:39 VM2 danylenko: setfacl -m u:utest:r /tmp/acl_test
```

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*).

```
utest@VM2:~$ touch /tmp/acl_test/prohibited.txt
touch: cannot touch '/tmp/acl_test/prohibited.txt': Permission denied
utest@VM2:~$ echo "new content" > /tmp/acl_test/utest.txt
-bash: /tmp/acl_test/utest.txt: Permission denied
utest@VM2:~$ _
```

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 action is allowed with **chmod** (conventional mechanism). (Hint: repeat step 3, but given the new context).

```
danylenko@VM2:~$ cmd="chmod u-rwx,g-rwx,o-rwx -v /tmp/acl_test; setfacl -m u:utest:
rwx /tmp/acl_test ; getfacl /tmp/acl_test"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -30 /
var/log/syslog | grep "19:08"
getfacl: Removing leading '/' from absolute path names
Apr 28 19:08:28 VM2 danylenko: chmod u-rwx,g-rwx,o-rwx -v /tmp/acl_test; setfacl -m
 u:utest:rwx /tmp/acl_test ; getfacl /tmp/acl_test
Apr 28 19:08:28 VM2 danylenko: mode of '/tmp/acl_test' changed from 0757 (rwxr-xrwx
) to 0000 (---------)
Apr 28 19:08:28 VM2 danylenko: # file: tmp/acl_test
Apr 28 19:08:28 VM2 danylenko: # owner: danylenko
Apr 28 19:08:28 VM2 danylenko: # group: danylenko
Apr 28 19:08:28 VM2 danylenko: user::---
Apr 28 19:08:28 VM2 danylenko: user:utest:rwx
Apr 28 19:08:28 VM2 danylenko: group::r-x
Apr 28 19:08:28 VM2 danylenko: mask::rwx
Apr 28 19:08:28 VM2 danylenko: other::---
Apr 28 19:08:28 VM2 danylenko:
```

```
Вівторок, 28 квітня 2020 19:09:07 +0300
utest@VM2:~$ touch /tmp/acl_test/prohibited.txt
utest@VM2:~$ echo "new content" > /tmp/acl_test/utest.txt
utest@VM2:~$
```

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 **setfacl** command).

```
danylenko@VM2:~$ cmd="setfacl -d -m u:utest:r,o::r /tmp/acl_test ; getfacl /tmp/acl
_test"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -30 /
var/log/syslog | grep "19:29"
getfacl: Removing leading '/' from absolute path names
Apr 28 19:29:43 VM2 danylenko: setfacl -d -m u:utest:r,o::r /tmp/acl_test ; getfacl
 /tmp/acl_test
Apr 28 19:29:43 VM2 danylenko: # file: tmp/acl_test
Apr 28 19:29:43 VM2 danylenko: # owner: danylenko
Apr 28 19:29:43 VM2 danylenko: # group: danylenko
Apr 28 19:29:43 VM2 danylenko: user::---
Apr 28 19:29:43 VM2 danylenko: user:utest:rwx
Apr 28 19:29:43 VM2 danylenko: group::r-x
Apr 28 19:29:43 VM2 danylenko: mask::rwx
Apr 28 19:29:43 VM2 danylenko: other::---
Apr 28 19:29:43 VM2 danylenko: default:user::---
Apr 28 19:29:43 VM2 danylenko: default:user:utest:r--
Apr 28 19:29:43 VM2 danylenko: default:group::r-x
Apr 28 19:29:43 VM2 danylenko: default:mask::r-x
Apr 28 19:29:43 VM2 danylenko: default:other::r--
Apr 28 19:29:43 VM2 danylenko:
```

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 **getfacl**.

```
вівторок, 28 квітня 2020 19:35:03 +0300
utest@VM2:~$ touch /tmp/acl_test/utest2.txt
utest@VM2:~$ getfacl /tmp/acl_test/utest2.txt | logger
```

Queried permitions and send them to logger from utest.

Looked at them through log from danylenko user

```
danylenko@VM2:~$ tail -200 /var/log/syslog | grep "19:35" | grep "utest:"
Apr 28 19:35:16 VM2 utest: # file: tmp/acl_test/utest2.txt
Apr 28 19:35:16 VM2 utest: # owner: utest
Apr 28 19:35:16 VM2 utest: # group: utest
Apr 28 19:35:16 VM2 utest: user::---
Apr 28 19:35:16 VM2 utest: user:utest:r--
Apr 28 19:35:16 VM2 utest: group::r-x#011#effective:r--
Apr 28 19:35:16 VM2 utest: mask::r--
Apr 28 19:35:16 VM2 utest: other::r--
Apr 28 19:35:16 VM2 utest:
```

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 **getfacl**.

```
danylenko@VM2:~$ cmd="sudo setfacl -m mask::r /tmp/acl_test/utest.txt ;sudo getfacl
 /tmp/acl_test/utest.txt"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -30 /
var/log/syslog | grep "$(date +%H:%M)"
getfacl: Removing leading '/' from absolute path names
Apr 28 19:54:14 VM2 danylenko: sudo setfacl -m mask::r /tmp/acl_test/utest.txt ;sud
o getfacl /tmp/acl_test/utest.txt
Apr 28 19:54:15 VM2 danylenko: # file: tmp/acl_test/utest.txt
Apr 28 19:54:15 VM2 danylenko: # owner: utest
Apr 28 19:54:15 VM2 danylenko: # group: utest
Apr 28 19:54:15 VM2 danylenko: user::rw-
Apr 28 19:54:15 VM2 danylenko: group::rw-#011#effective:r--
Apr 28 19:54:15 VM2 danylenko: mask::r--
Apr 28 19:54:15 VM2 danylenko: other::r--
Apr 28 19:54:15 VM2 danylenko:
```

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

To remove ACL permitions I had to return chmod permitions that I removed earlier, for simplicity added max permitions.

```
danylenko@VM2:~$ cmd="sudo chmod u=rwx,g=rwx,o=rwx -vR /tmp/acl_test/"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -30 /
var/log/syslog | grep "$(date +%H:%M)"
Apr 28 20:25:26 VM2 danylenko: sudo chmod u=rwx,g=rwx,o=rwx -vR /tmp/acl_test/
Apr 28 20:25:26 VM2 danylenko: mode of '/tmp/acl_test/' changed from 0654 (rw-r-xr-
-) to 0777 (rwxrwxrwx)
Apr 28 20:25:26 VM2 danylenko: mode of '/tmp/acl_test/utest.txt' changed from 0644
(rw-r--r--) to 0777 (rwxrwxrwx)
Apr 28 20:25:26 VM2 danylenko: mode of '/tmp/acl_test/utest2.txt' changed from 0044
 (---r--r--) to 0777 (rwxrwxrwx)
Apr 28 20:25:26 VM2 danylenko: mode of '/tmp/acl_test/prohibited.txt' changed from
0664 (rw-rw-r--) to 0777 (rwxrwxrwx)
danylenko@VM2:~$ cmd="sudo setfacl -b -R /tmp/acl_test"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -30 /
var/log/syslog | grep "$(date +%H:%M)"
Apr 28 20:26:05 VM2 danylenko: sudo setfacl -b -R /tmp/acl_test
```

Checking:

```
danylenko@VM2:~$ cmd="sudo getfacl -R /tmp/acl_test"
danylenko@VM2:~$ (echo $cmd | logger ; eval $cmd | logger ; sleep 5s) && tail -30 /var/log/syslog | grep "$(da
te +%H:%M)"
getfacl: Removing leading '/' from absolute path names
Apr 28 20:28:02 VM2 danylenko: sudo getfacl -R /tmp/acl_test
Apr 28 20:28:02 VM2 danylenko: # file: tmp/acl_test
Apr 28 20:28:02 VM2 danylenko: # owner: danylenko
Apr 28 20:28:02 VM2 danylenko: # group: danylenko
Apr 28 20:28:02 VM2 danylenko: user::rwx
Apr 28 20:28:02 VM2 danylenko: group::rwx
Apr 28 20:28:02 VM2 danylenko: other::rwx
Apr 28 20:28:02 VM2 danylenko:
Apr 28 20:28:02 VM2 danylenko: # file: tmp/acl_test/utest.txt
Apr 28 20:28:02 VM2 danylenko: # owner: utest
Apr 28 20:28:02 VM2 danylenko: # group: utest
Apr 28 20:28:02 VM2 danylenko: user::rwx
Apr 28 20:28:02 VM2 danylenko: group::rwx
Apr 28 20:28:02 VM2 danylenko: other::rwx
Apr 28 20:28:02 VM2 danylenko:
Apr 28 20:28:02 VM2 danylenko: # file: tmp/acl_test/utest2.txt
Apr 28 20:28:02 VM2 danylenko: # owner: utest
Apr 28 20:28:02 VM2 danylenko: # group: utest
Apr 28 20:28:02 VM2 danylenko: user::rwx
Apr 28 20:28:02 VM2 danylenko: group::rwx
Apr 28 20:28:02 VM2 danylenko: other::rwx
Apr 28 20:28:02 VM2 danylenko:
Apr 28 20:28:02 VM2 danylenko: # file: tmp/acl_test/prohibited.txt
Apr 28 20:28:02 VM2 danylenko: # owner: utest
```