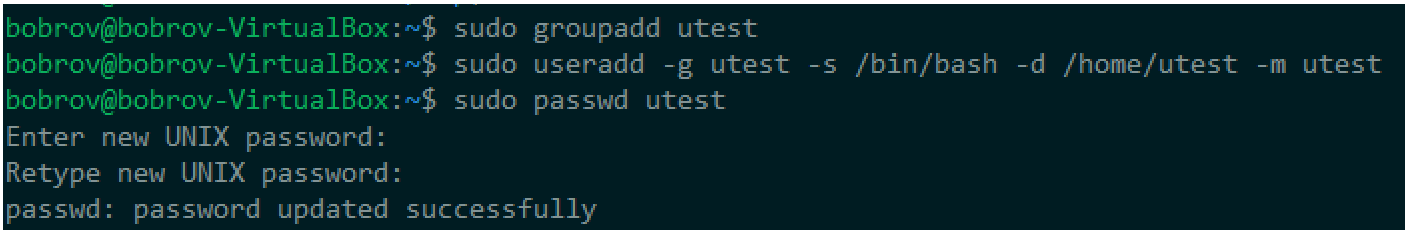
EPAM University Programs
DevOps external course
Module 4 Linux & Bash Essentials
TASK 4.7

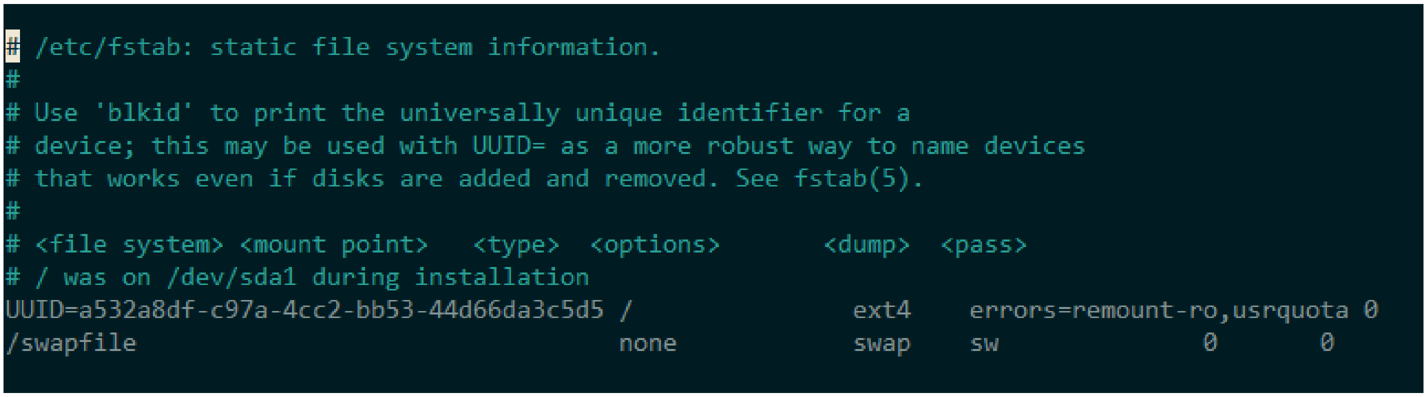
Part1. **Quota allocation mechanism.**

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. 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).



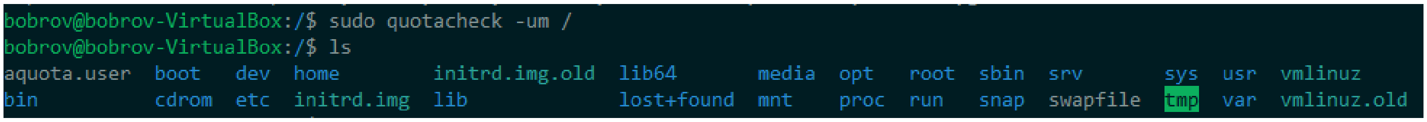
```
bobrov@bobrov-VirtualBox:~$ sudo groupadd utest
bobrov@bobrov-VirtualBox:~$ sudo useradd -g utest -s /bin/bash -d /home/utest -m utest
bobrov@bobrov-VirtualBox:~$ sudo passwd utest
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

```
sudo apt install quota
sudo nano /etc/fstab
```



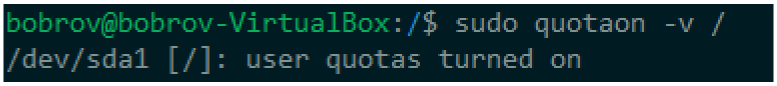
```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
# / was on /dev/sda1 during installation
UUID=a532a8df-c97a-4cc2-bb53-44d66da3c5d5 /               ext4    errors=remount-ro,usrquota 0
/swapfile                                 none            swap    sw              0       0
```

```
sudo mount -o remount /
sudo quotacheck -um /
```



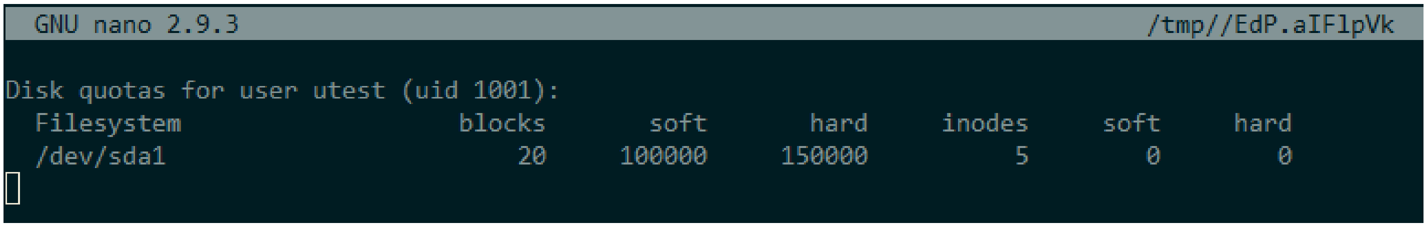
```
bobrov@bobrov-VirtualBox:/$ sudo quotacheck -um /
bobrov@bobrov-VirtualBox:/$ ls
aquota.user  boot   dev  home        initrd.img.old  lib64       media  opt   root  sbin  srv       sys  usr  vmlinuz
bin          cdrom  etc  initrd.img  lib             lost+found  mnt    proc  run   snap  swapfile  tmp  var  vmlinuz.old
```

```
sudo quotaon -v /
```



```
bobrov@bobrov-VirtualBox:/$ sudo quotaon -v /
/dev/sda1 [/]: user quotas turned on
```

sudo edquota -u utest

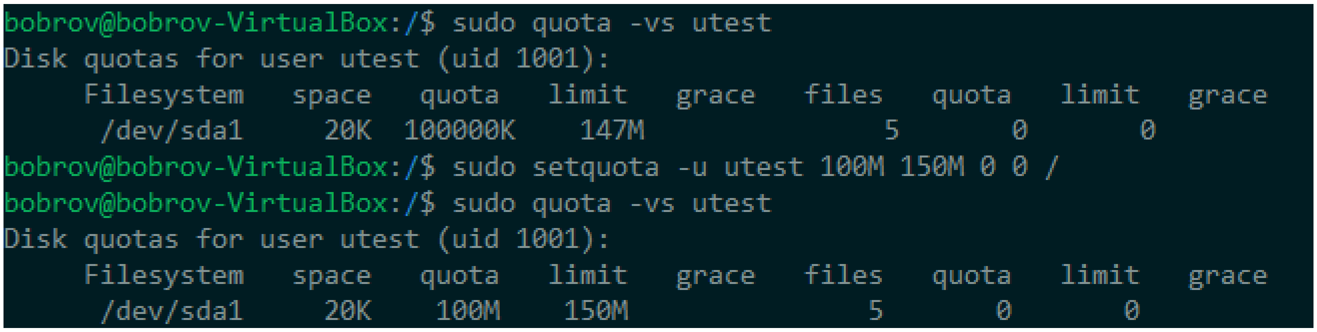


```
 GNU nano 2.9.3                                                        /tmp//EdP.aIFlpVk

Disk quotas for user utest (uid 1001):
  Filesystem                   blocks       soft       hard     inodes     soft     hard
  /dev/sda1                        20     100000     150000          5        0        0
```

Check the established quotas and edit the values:

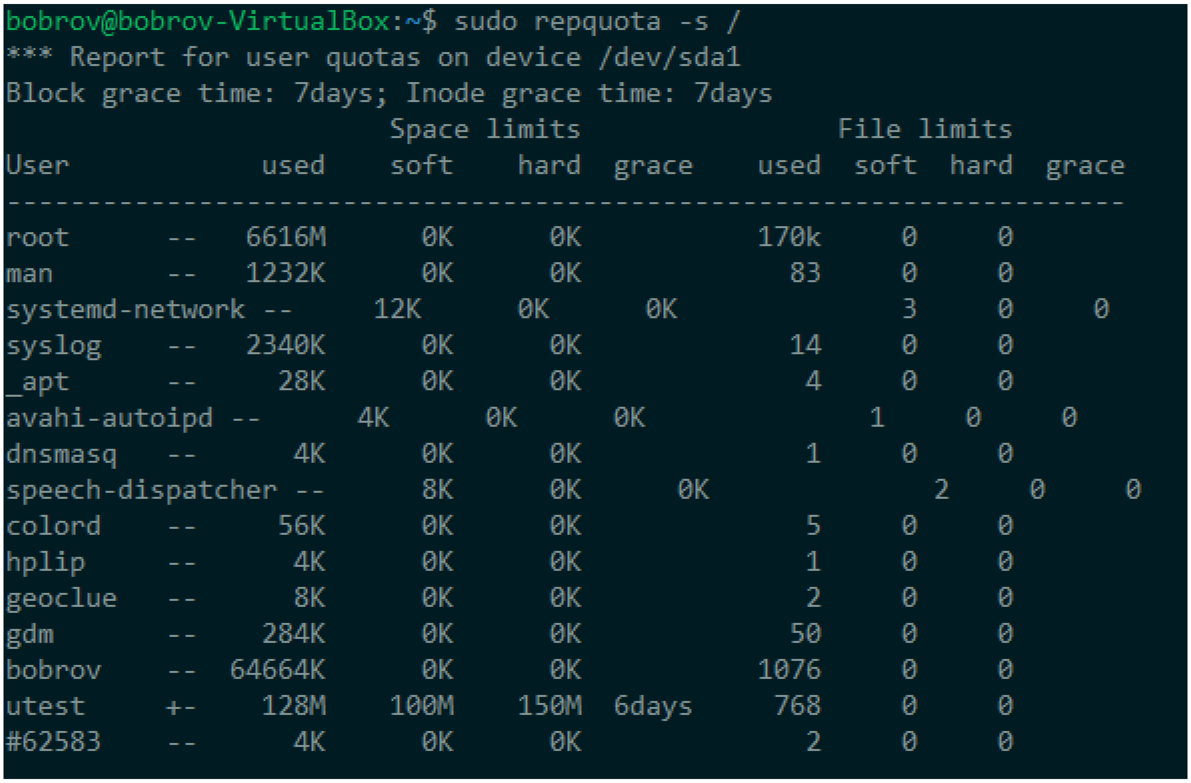
sudo quota -vs utest

sudo setquota -u utest 100M 150M 0 0 /



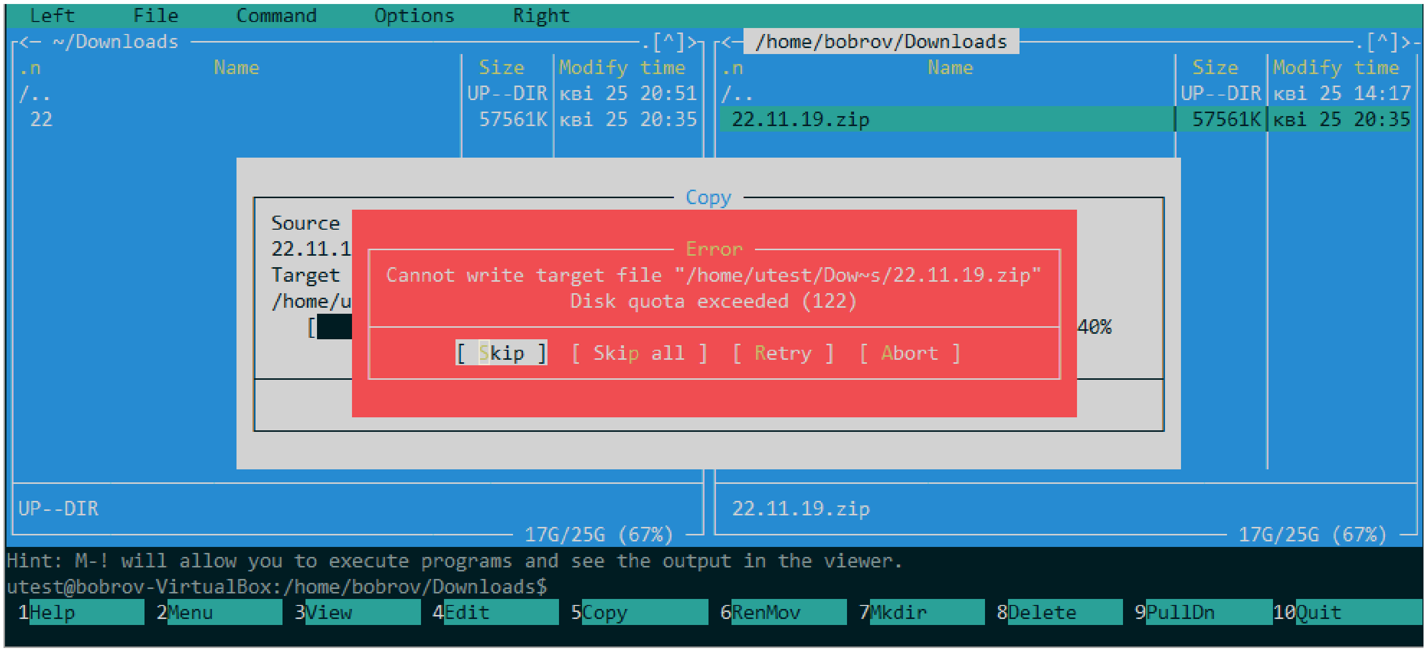
```
bobrov@bobrov-VirtualBox:/$ sudo quota -vs utest
Disk quotas for user utest (uid 1001):
     Filesystem   space   quota   limit   grace   files   quota   limit   grace
      /dev/sda1     20K  100000K   147M                5       0       0
bobrov@bobrov-VirtualBox:/$ sudo setquota -u utest 100M 150M 0 0 /
bobrov@bobrov-VirtualBox:/$ sudo quota -vs utest
Disk quotas for user utest (uid 1001):
     Filesystem   space   quota   limit   grace   files   quota   limit   grace
      /dev/sda1     20K    100M    150M               5       0       0
```

sudo repquota -s /



```
bobrov@bobrov-VirtualBox:~$ sudo repquota -s /
*** Report for user quotas on device /dev/sda1
Block grace time: 7days; Inode grace time: 7days
                        Space limits                File limits
User            used    soft    hard  grace    used  soft  hard  grace
----------------------------------------------------------------------
root      --   6616M      0K      0K           170k     0     0
man       --   1232K      0K      0K             83     0     0
systemd-network --     12K      0K      0K              3     0     0
syslog    --   2340K      0K      0K             14     0     0
_apt      --     28K      0K      0K              4     0     0
avahi-autoipd --      4K      0K      0K              1     0     0
dnsmasq   --      4K      0K      0K              1     0     0
speech-dispatcher --      8K      0K      0K              2     0     0
colord    --     56K      0K      0K              5     0     0
hplip     --      4K      0K      0K              1     0     0
geoclue   --      8K      0K      0K              2     0     0
gdm       --    284K      0K      0K             50     0     0
bobrov    --  64664K      0K      0K           1076     0     0
utest     +-    128M    100M    150M  6days     768     0     0
#62583    --      4K      0K      0K              2     0     0
```

```
sudo apt install mc
mc run
```



Left File Command Options Right
1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit

```
<- ~/Downloads                                  .[^]>  <- /home/bobrov/Downloads                        .[^]>
.n              Name            Size   |Modify time   .n                 Name           Size   |Modify time
/..                            UP--DIR |квi 25 20:51  /..                              UP--DIR |квi 25 14:17
 22                            57561K  |квi 25 20:35   22.11.19.zip                    57561K  |квi 25 20:35

                         Copy
   Source
   22.11.1             Error
   Target   Cannot write target file "/home/utest/Dow~s/22.11.19.zip"
   /home/u            Disk quota exceeded (122)
      [
              [ Skip ]  [ Skip all ]  [ Retry ]  [ Abort ]         40%

UP--DIR                                            22.11.19.zip
                              17G/25G (67%)                                         17G/25G (67%)
Hint: M-! will allow you to execute programs and see the output in the viewer.
utest@bobrov-VirtualBox:/home/bobrov/Downloads$
```

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).

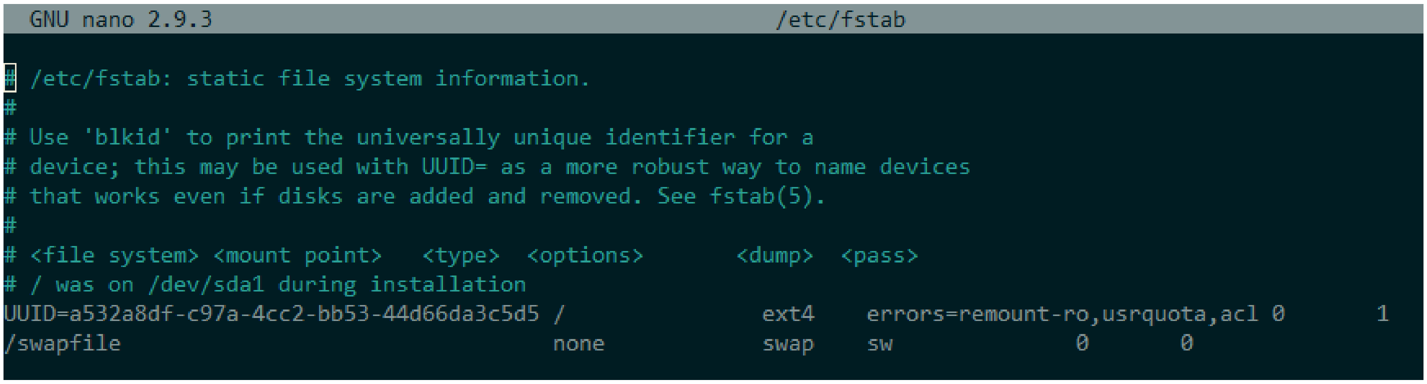
1. Based on given in presentation #4.7 instructions, turn on and set up the ACL. *Caution*! The fact that a file system has been mounted with the "acl" flag on by default, doesn't mean that the ACL package is installed.

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).

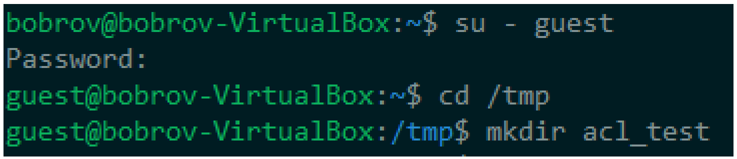
Edit file **/etc/fstab** add option **acl**



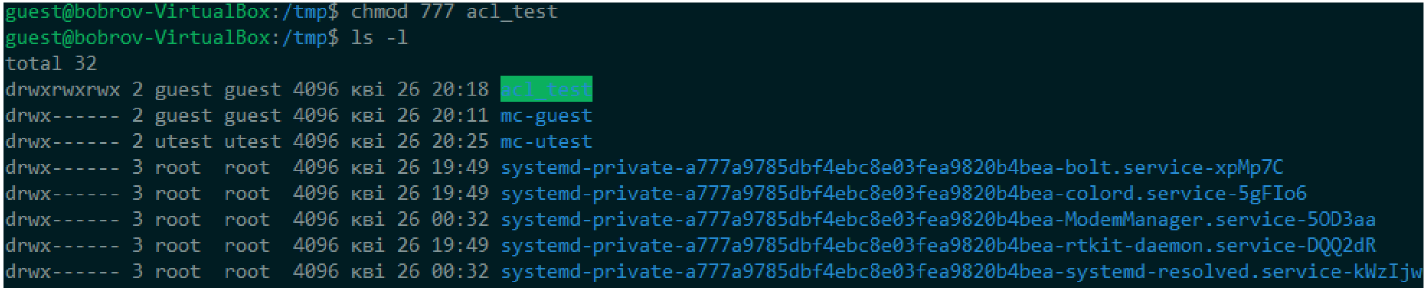
```
  GNU nano 2.9.3                                     /etc/fstab

# /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/sda1 during installation
UUID=a532a8df-c97a-4cc2-bb53-44d66da3c5d5 /               ext4    errors=remount-ro,usrquota,acl 0       1
/swapfile                                 none            swap    sw              0       0
```

run command and add result in **/var/log/syslog** with teg **testacl2**
tune2fs -l /dev/sda1 | logger -t testacl2
sudo tune2fs -l /dev/sda1 | logger -t testacl2

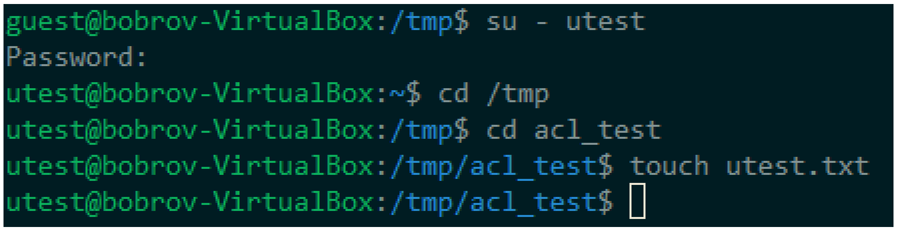
2. Log in as *guest*. Create in */tmp* a directory called *acl_test*. By means of *chmod*, allow user utest to perform all possible operations (rwx) with respect to *acl_test.* Verify that user *utest* is indeed capable of implementing granted him (her) privileges. For example, acer logging in as *utest*, 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



```
bobrov@bobrov-VirtualBox:~$ su - guest
Password:
guest@bobrov-VirtualBox:~$ cd /tmp
guest@bobrov-VirtualBox:/tmp$ mkdir acl_test
```

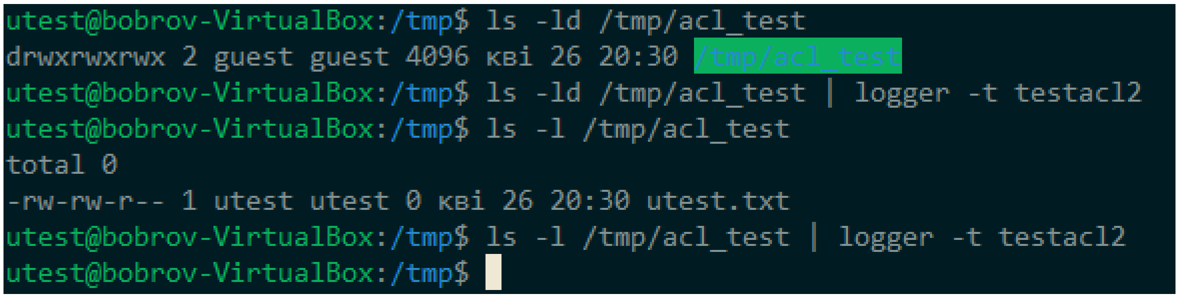


```
guest@bobrov-VirtualBox:/tmp$ chmod 777 acl_test
guest@bobrov-VirtualBox:/tmp$ ls -l
total 32
drwxrwxrwx 2 guest guest 4096 кві 26 20:18 acl_test
drwx------ 2 guest guest 4096 кві 26 20:11 mc-guest
drwx------ 2 utest utest 4096 кві 26 20:25 mc-utest
drwx------ 3 root  root  4096 кві 26 19:49 systemd-private-a777a9785dbf4ebc8e03fea9820b4bea-bolt.service-xpMp7C
drwx------ 3 root  root  4096 кві 26 19:49 systemd-private-a777a9785dbf4ebc8e03fea9820b4bea-colord.service-5gFIo6
drwx------ 3 root  root  4096 кві 26 00:32 systemd-private-a777a9785dbf4ebc8e03fea9820b4bea-ModemManager.service-5OD3aa
drwx------ 3 root  root  4096 кві 26 19:49 systemd-private-a777a9785dbf4ebc8e03fea9820b4bea-rtkit-daemon.service-DQQ2dR
drwx------ 3 root  root  4096 кві 26 00:32 systemd-private-a777a9785dbf4ebc8e03fea9820b4bea-systemd-resolved.service-kWzIjw
```



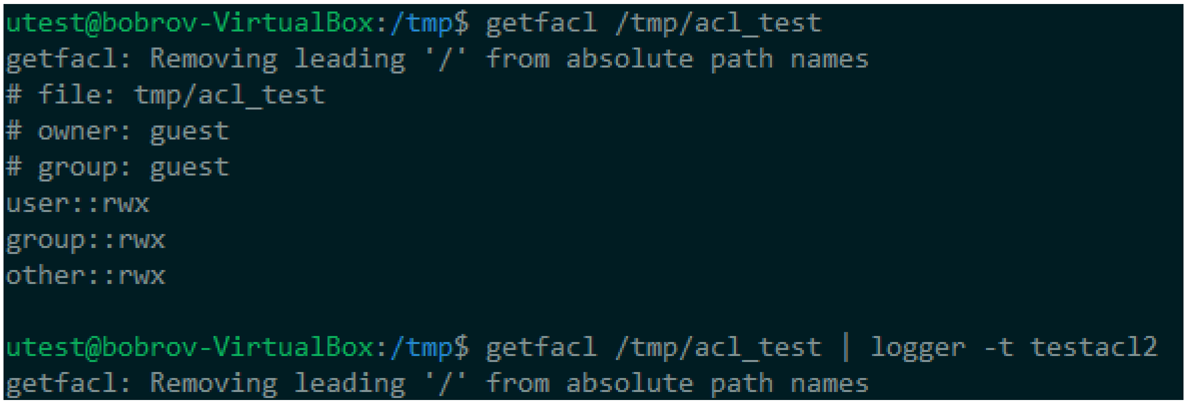
```
guest@bobrov-VirtualBox:/tmp$ su - utest
Password:
utest@bobrov-VirtualBox:~$ cd /tmp
utest@bobrov-VirtualBox:/tmp$ cd acl_test
utest@bobrov-VirtualBox:/tmp/acl_test$ touch utest.txt
utest@bobrov-VirtualBox:/tmp/acl_test$ 
```

ls -ld /tmp/acl_test
ls -l /tmp/acl_test



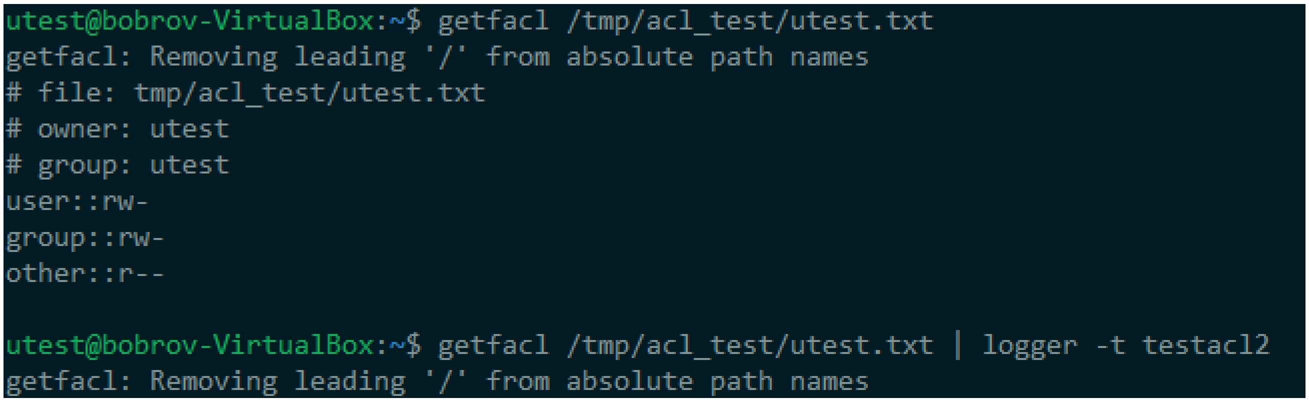
```
utest@bobrov-VirtualBox:/tmp$ ls -ld /tmp/acl_test
drwxrwxrwx 2 guest guest 4096 кві 26 20:30 /tmp/acl_test
utest@bobrov-VirtualBox:/tmp$ ls -ld /tmp/acl_test | logger -t testacl2
utest@bobrov-VirtualBox:/tmp$ ls -l /tmp/acl_test
total 0
-rw-rw-r-- 1 utest utest 0 кві 26 20:30 utest.txt
utest@bobrov-VirtualBox:/tmp$ ls -l /tmp/acl_test | logger -t testacl2
utest@bobrov-VirtualBox:/tmp$ 
```

To check ACL permissions do:
getfacl /tmp/acl_test



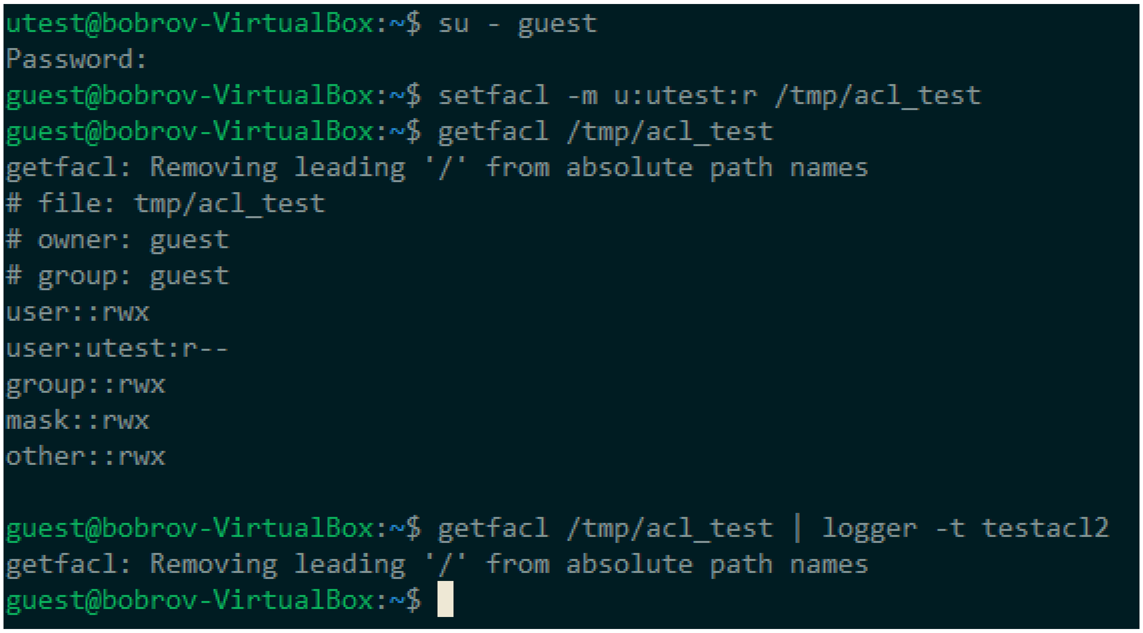
```
utest@bobrov-VirtualBox:/tmp$ getfacl /tmp/acl_test
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test
# owner: guest
# group: guest
user::rwx
group::rwx
other::rwx

utest@bobrov-VirtualBox:/tmp$ getfacl /tmp/acl_test | logger -t testacl2
getfacl: Removing leading '/' from absolute path names
```

getfacl /tmp/acl_test/utest.txt



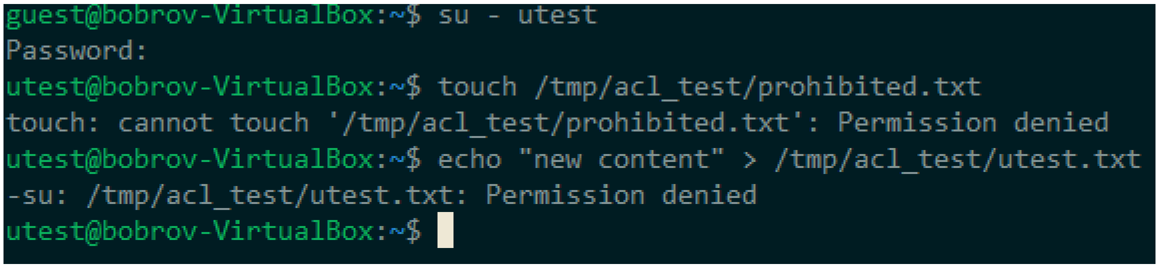
```
utest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test/utest.txt
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test/utest.txt
# owner: utest
# group: utest
user::rw-
group::rw-
other::r--

utest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test/utest.txt | logger -t testacl2
getfacl: Removing leading '/' from absolute path names
```

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



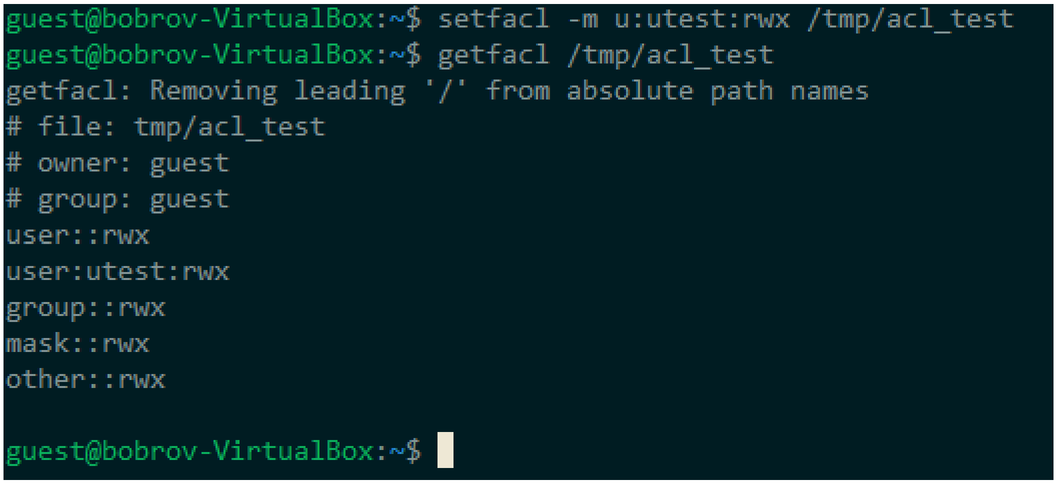
```
utest@bobrov-VirtualBox:~$ su - guest
Password:
guest@bobrov-VirtualBox:~$ setfacl -m u:utest:r /tmp/acl_test
guest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test
# owner: guest
# group: guest
user::rwx
user:utest:r--
group::rwx
mask::rwx
other::rwx

guest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test | logger -t testacl2
getfacl: Removing leading '/' from absolute path names
guest@bobrov-VirtualBox:~$
```

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*).

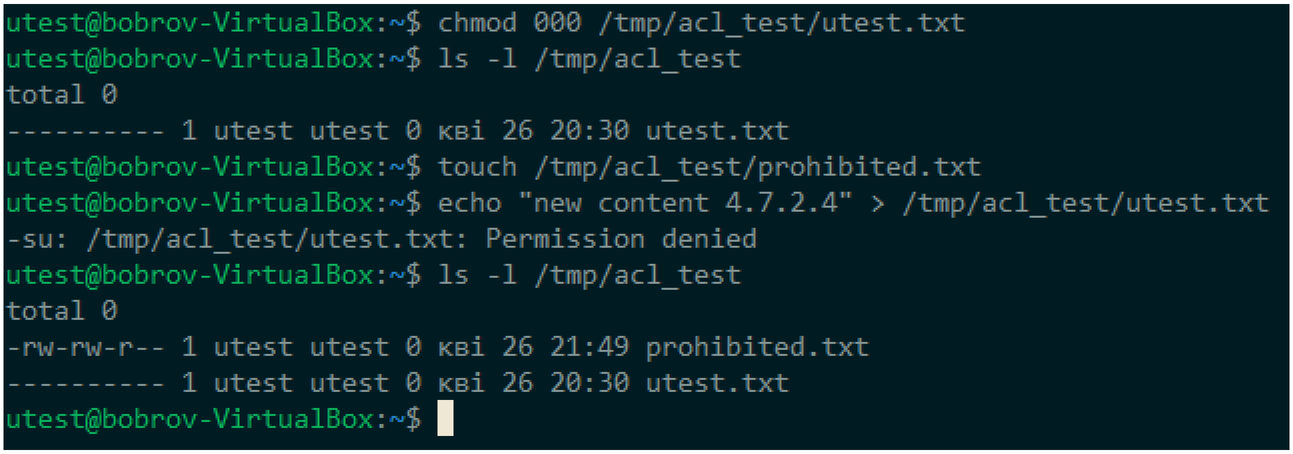


```
guest@bobrov-VirtualBox:~$ su - utest
Password:
utest@bobrov-VirtualBox:~$ touch /tmp/acl_test/prohibited.txt
touch: cannot touch '/tmp/acl_test/prohibited.txt': Permission denied
utest@bobrov-VirtualBox:~$ echo "new content" > /tmp/acl_test/utest.txt
-su: /tmp/acl_test/utest.txt: Permission denied
utest@bobrov-VirtualBox:~$
```

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).

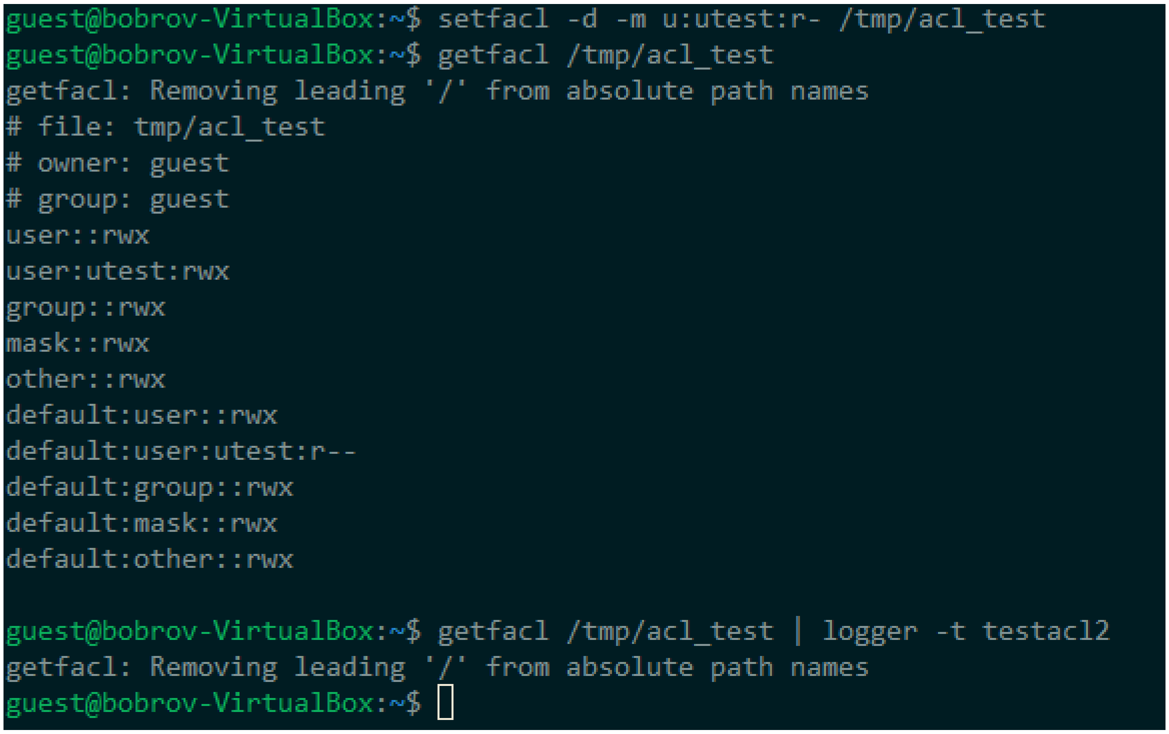


```
guest@bobrov-VirtualBox:~$ setfacl -m u:utest:rwx /tmp/acl_test
guest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test
# owner: guest
# group: guest
user::rwx
user:utest:rwx
group::rwx
mask::rwx
other::rwx

guest@bobrov-VirtualBox:~$
```

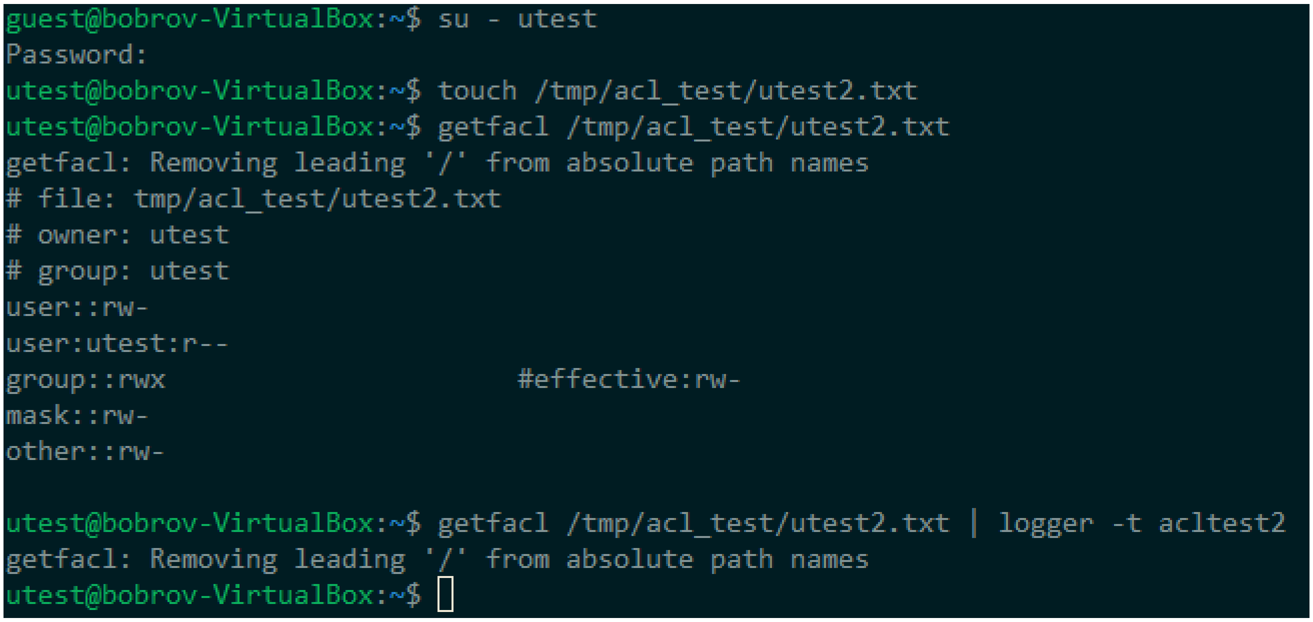


```
utest@bobrov-VirtualBox:~$ chmod 000 /tmp/acl_test/utest.txt
utest@bobrov-VirtualBox:~$ ls -l /tmp/acl_test
total 0
---------- 1 utest utest 0 кві 26 20:30 utest.txt
utest@bobrov-VirtualBox:~$ touch /tmp/acl_test/prohibited.txt
utest@bobrov-VirtualBox:~$ echo "new content 4.7.2.4" > /tmp/acl_test/utest.txt
-su: /tmp/acl_test/utest.txt: Permission denied
utest@bobrov-VirtualBox:~$ ls -l /tmp/acl_test
total 0
-rw-rw-r-- 1 utest utest 0 кві 26 21:49 prohibited.txt
---------- 1 utest utest 0 кві 26 20:30 utest.txt
utest@bobrov-VirtualBox:~$
```

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). 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*.

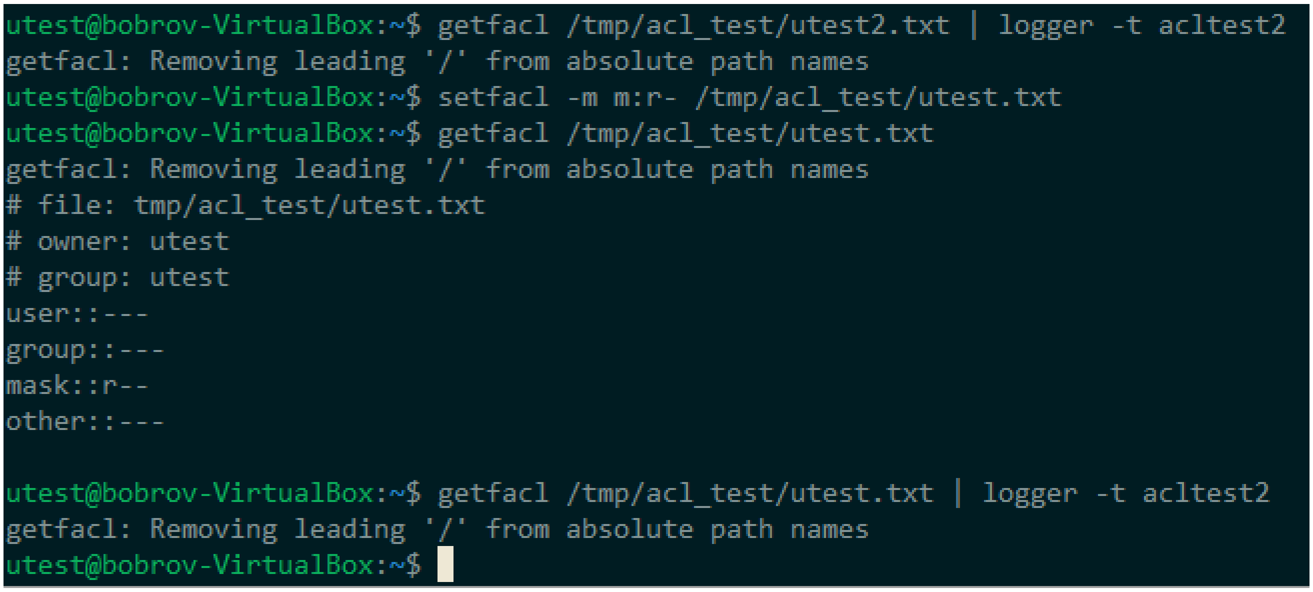


```
guest@bobrov-VirtualBox:~$ setfacl -d -m u:utest:r- /tmp/acl_test
guest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test
# owner: guest
# group: guest
user::rwx
user:utest:rwx
group::rwx
mask::rwx
other::rwx
default:user::rwx
default:user:utest:r--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

guest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test | logger -t testacl2
getfacl: Removing leading '/' from absolute path names
guest@bobrov-VirtualBox:~$
```



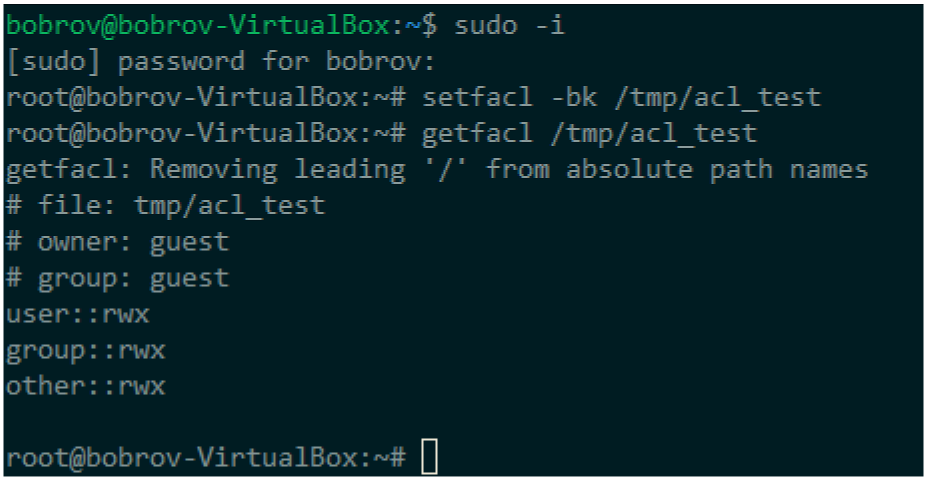
```
guest@bobrov-VirtualBox:~$ su - utest
Password:
utest@bobrov-VirtualBox:~$ touch /tmp/acl_test/utest2.txt
utest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test/utest2.txt
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test/utest2.txt
# owner: utest
# group: utest
user::rw-
user:utest:r--
group::rwx                      #effective:rw-
mask::rw-
other::rw-

utest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test/utest2.txt | logger -t acltest2
getfacl: Removing leading '/' from absolute path names
utest@bobrov-VirtualBox:~$
```

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*.

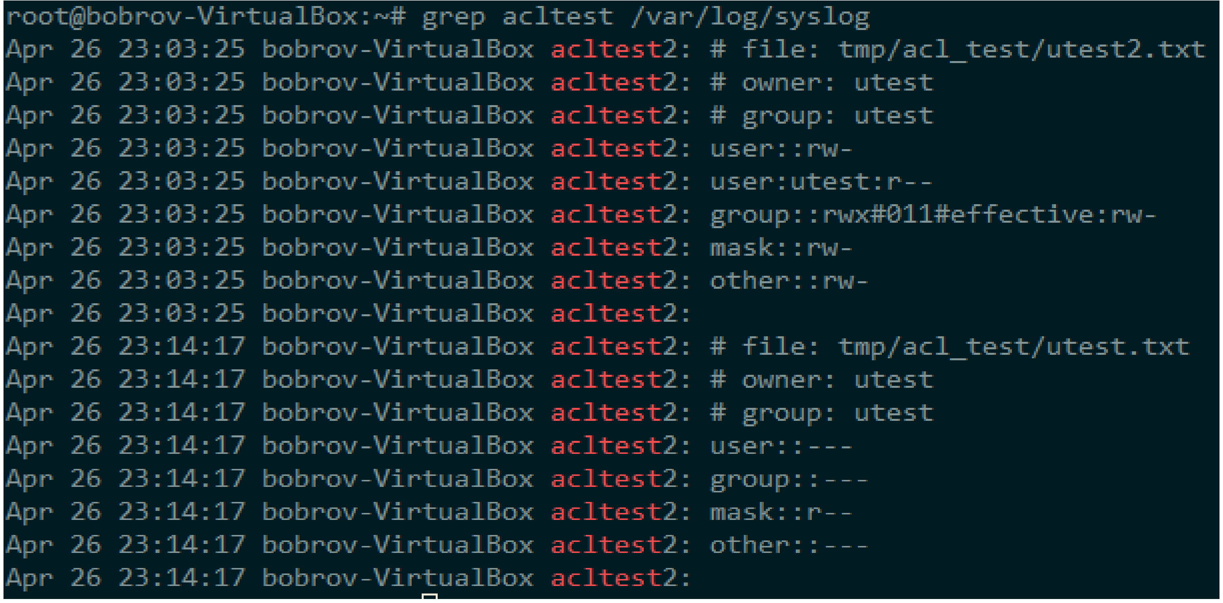


```
utest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test/utest2.txt | logger -t acltest2
getfacl: Removing leading '/' from absolute path names
utest@bobrov-VirtualBox:~$ setfacl -m m:r- /tmp/acl_test/utest.txt
utest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test/utest.txt
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test/utest.txt
# owner: utest
# group: utest
user::---
group::---
mask::r--
other::---

utest@bobrov-VirtualBox:~$ getfacl /tmp/acl_test/utest.txt | logger -t acltest2
getfacl: Removing leading '/' from absolute path names
utest@bobrov-VirtualBox:~$
```

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



```
bobrov@bobrov-VirtualBox:~$ sudo -i
[sudo] password for bobrov:
root@bobrov-VirtualBox:~# setfacl -bk /tmp/acl_test
root@bobrov-VirtualBox:~# getfacl /tmp/acl_test
getfacl: Removing leading '/' from absolute path names
# file: tmp/acl_test
# owner: guest
# group: guest
user::rwx
group::rwx
other::rwx

root@bobrov-VirtualBox:~#
```



```
root@bobrov-VirtualBox:~# grep acltest /var/log/syslog
Apr 26 23:03:25 bobrov-VirtualBox acltest2: # file: tmp/acl_test/utest2.txt
Apr 26 23:03:25 bobrov-VirtualBox acltest2: # owner: utest
Apr 26 23:03:25 bobrov-VirtualBox acltest2: # group: utest
Apr 26 23:03:25 bobrov-VirtualBox acltest2: user::rw-
Apr 26 23:03:25 bobrov-VirtualBox acltest2: user:utest:r--
Apr 26 23:03:25 bobrov-VirtualBox acltest2: group::rwx#011#effective:rw-
Apr 26 23:03:25 bobrov-VirtualBox acltest2: mask::rw-
Apr 26 23:03:25 bobrov-VirtualBox acltest2: other::rw-
Apr 26 23:03:25 bobrov-VirtualBox acltest2:
Apr 26 23:14:17 bobrov-VirtualBox acltest2: # file: tmp/acl_test/utest.txt
Apr 26 23:14:17 bobrov-VirtualBox acltest2: # owner: utest
Apr 26 23:14:17 bobrov-VirtualBox acltest2: # group: utest
Apr 26 23:14:17 bobrov-VirtualBox acltest2: user::---
Apr 26 23:14:17 bobrov-VirtualBox acltest2: group::---
Apr 26 23:14:17 bobrov-VirtualBox acltest2: mask::r--
Apr 26 23:14:17 bobrov-VirtualBox acltest2: other::---
Apr 26 23:14:17 bobrov-VirtualBox acltest2:
```