EPAM University Programs
DevOps external course
Module 4 Linux& Bash Essentials
TASK 4.5

1. To discover files with active sticky bits, use the following version of the **find** command:
sudo find / -perm /6000 -type f -exec ls -ld {} \;>setuid.txt
Put into your report a fragment of setuid.txt file.

```
  GNU nano 2.9.3                    setuid.txt

-rwxr-sr-x 1 root shadow 34816 лют 27  2019 /sbin/pam_extrausers_chkpwd
-rwxr-sr-x 1 root shadow 34816 лют 27  2019 /sbin/unix_chkpwd
-rwsr-xr-x 1 root root 43088 бер  5 19:23 /bin/mount
-rwsr-xr-x 1 root root 64424 чер 28  2019 /bin/ping
-rwsr-xr-x 1 root root 44664 бер 22  2019 /bin/su
-rwsr-xr-x 1 root root 26696 бер  5 19:23 /bin/umount
-rwsr-xr-x 1 root root 30800 сер 11  2016 /bin/fusermount
-rwsr-xr-- 1 root dip 382696 лют 11 17:05 /usr/sbin/pppd
```

Explain meaning of parameters of the above **find** command (hint: use find's man page).
-perm /mode - Any of the permission bits mode are set for the file.
6000 - we are looking for files with user and group setuid bits set.
- type - type of a file
f - regular file
- exec - execute a command for each result
ls - list directory contents
-ld - using a long listing format and list directory entries instead of contents

2. Discovering soft and hard links.

Comment on results of these commands (place the output into your report):

#navigate to the home directory

cd

#create a new directory

mkdir test

navigate to the directory

cd test

#create a new file

touch test1.txt

#Record stdout of echo command in the file

echo “test1.txt” > test1.txt

#show a content of the current directory using a long listing format

ls -l .

(a hard link)

#create a hard link

ln test1.txt test2.txt

#show a content of the current directory using a long listing format

ls -l .

(pay attention to the number of links to test1.txt and test2.txt #now the number of links is 2)

```
vm1kern@vm1kern-VirtualBox:~$ cd
vm1kern@vm1kern-VirtualBox:~$ mkdir test
vm1kern@vm1kern-VirtualBox:~$ cd test
vm1kern@vm1kern-VirtualBox:~/test$ touch test1.txt
vm1kern@vm1kern-VirtualBox:~/test$ echo "test1.txt">test1.txt
vm1kern@vm1kern-VirtualBox:~/test$ ls -l
total 4
-rw-r--r-- 1 vm1kern vm1kern 10 кві 22 00:18 test1.txt
vm1kern@vm1kern-VirtualBox:~/test$ ln test1.txt test2.txt
vm1kern@vm1kern-VirtualBox:~/test$ ls -l
total 8
-rw-r--r-- 2 vm1kern vm1kern 10 кві 22 00:18 test1.txt
-rw-r--r-- 2 vm1kern vm1kern 10 кві 22 00:18 test2.txt
```

#Record stdout of echo command in the file
echo “test2.txt” > test2.txt
#concatenate files and print their output in the console
cat test1.txt test2.txt
#remove the file test1.txt
rm test1.txt
#show a content of the current directory using a long listing format
ls -l .
(now a soft link)
#create a soft link to the file test2.txt
ln -s test2.txt test3.txt
#show a content of the current directory using a long listing format
ls -l .
(pay attention to the number of links to the created files #with a soft link the number of links remains unchanged)

```
vm1kern@vm1kern-VirtualBox:~/test$ ln -s test2.txt test3.txt
vm1kern@vm1kern-VirtualBox:~/test$ ls -l
total 4
-rw-r--r-- 1 vm1kern vm1kern 10 кві 22 00:27 test2.txt
lrwxrwxrwx 1 vm1kern vm1kern  9 кві 22 00:37 test3.txt -> test2.txt
```

#remove the file test2.txt
rm test2.txt;
#show a content of the current directory using a long listing format
ls -l .

```
vm1kern@vm1kern-VirtualBox:~/test$ rm test2.txt
vm1kern@vm1kern-VirtualBox:~/test$ ls -l
total 0
lrwxrwxrwx 1 vm1kern vm1kern 9 кві 22 00:37 test3.txt -> test2.txt
```

3. I/O redirect.

Execute these commands; comment on the output.

mount #show all mounts

blkid #locate/prink block devices

mount | **grep**sda - #print lines matching a pattern with "sda"

dmesg | **grep**sda - #print lines matching a pattern with "sda" from the kernel ring buffer

sudo grep -R -e “root” /etc> root_entries.txt

(place only a reasonable fragment of root_entries.txt into your report)