EPAM University Programs

DevOps external course

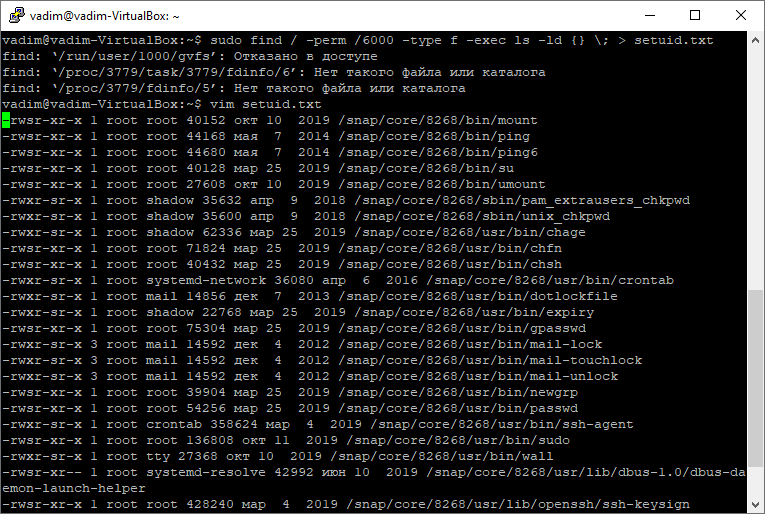
Module 4 Linux & Bash Essentials

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. Explain meaning of parameters of the above find command (hint: use find’s man page).



**sudo** – run the command **find** with root privileges

**find** – command to search files or directories

**/** - root of file system, place where are files need to search

**/6000** – options to search of permission access, in this exemple we put to search any permissions bits **suid** or **sgid** of files at mounted disks.

**-type f** - this optiont set to search only files at filesystem without directories

**-exec** – this option run the command **ls –ld** when the desired file is found

(man: Execute command; true if 0 status is returned)

**{}** – is replaced by the current file name being processed everywhere it occurs in the arguments to the command

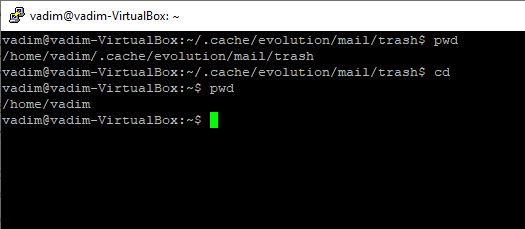
**\;** - end argument of the command **ls –ld**, and might need to be escaped (with a \) or quoted to protect them from expansion by the shell.

**>** - this option move output of the command **find** into the file **setuid.txt** also rewrite it if the file is already created

**setuid.txt** - the file with output command result **find**

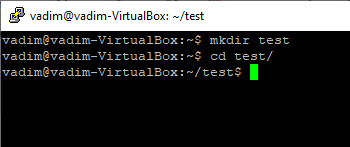
2. Discovering soft and hard links.

Comment on results of these commands (place the output into your report):   
cd – move to home directory of user



mkdir test – create directory test

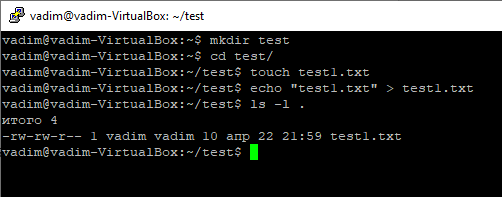
cd test – move into new directory test



touch test1.txt – create file test1.txt

echo “test1.txt” > test1.txt – overwrite text test1.txt into file test1.txt

ls -l . – list long command of the current directory

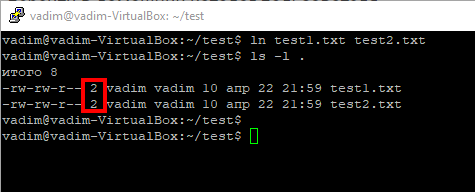


(a hard link)

ln test1.txt test2.txt create hard link **test2.txt** on the file **test1.txt**

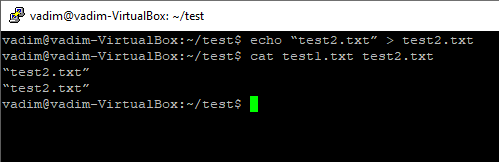
ls -l . - list long command of the current directory

(pay attention to the number of links to test1.txt and test2.txt)



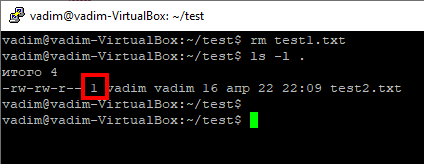
echo “test2.txt” > test2.txt - overwrite text test2.txt into file test2.txt

cat test1.txt test2.txt – show the contents of files test1.txt and test2.txt, the contents are the same.



rm test1.txt – remove the file test1.txt

ls -l . - list long command of the current directory **test**

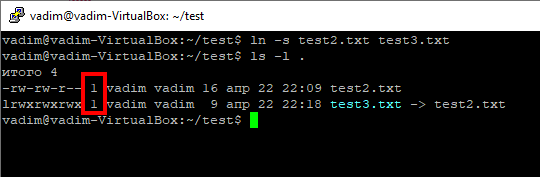


(now a soft link)

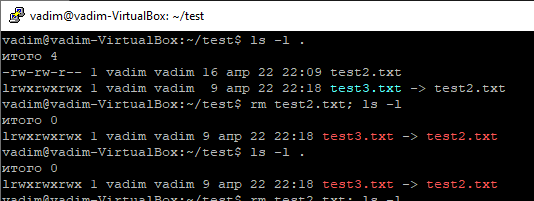
ln -s test2.txt test3.txt

ls -l .

(pay attention to the number of links to the created files)



rm test2.txt; ls -l . remove the file test2.txt and list long command of the current directory

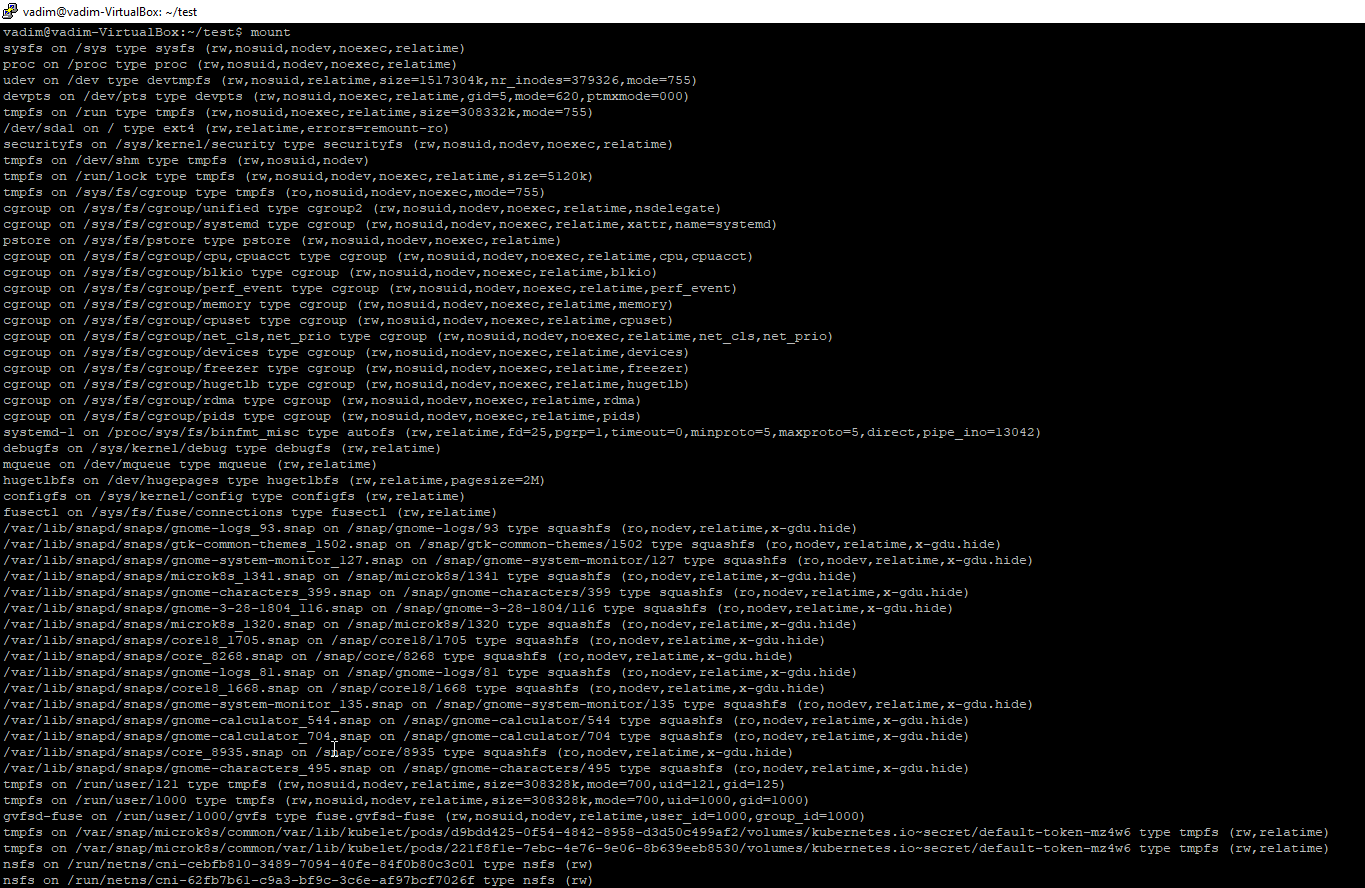


Soft link refers to a non-existent file test2.txt and turns red.

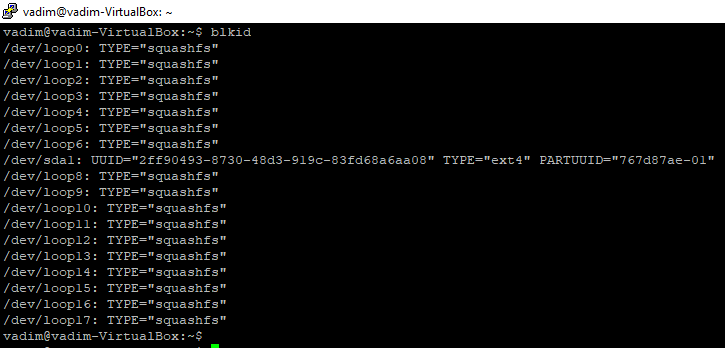
3. I/O redirect.

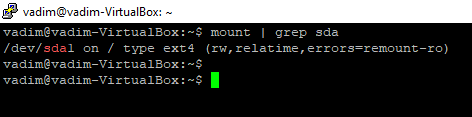
Execute these commands; comment on the output.

mount – list all mounted devices in system

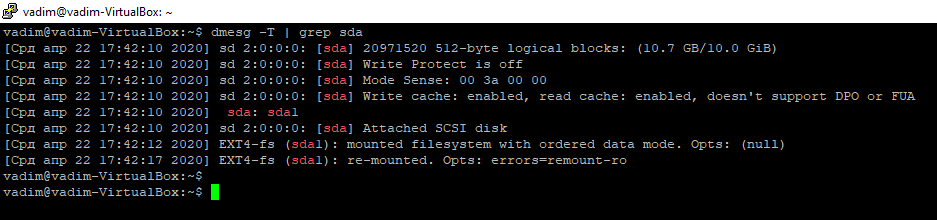


blkid – locate/print block device attributes – name device, UUID, type file system of mounted device and partuuid



mount | grep sda – list and filter devices of hard disk is connected into machine. Also we can take information about this hard disk – mounted point, type of file system and device access options

dmesg | grep sda – this command print kernel information about device hard disk **sda.** This information can be performed during the operation of the operating system.



sudo grep -R -e “root” /etc > root\_entries.txt – this command run to search recursively the string of files and links at pattern “root” into directory /etc

(place only a reasonable fragment of root\_entries.txt into your report)

