CYB613: Operating Systems Theory and Administration
Lab4: File Permission

Objective:

To learn how to mange and secure files in Linux OS

Complete Lab 9.1 thru 9.5 from the lab textbook. Rename this lab sheet to CYB613.lab4.firstName.lastName.docx

Lab 9.1: Managing File Permissions

Aside of step 1 & Step 10, provide a screenshot evidence of the successful execution of the command run below each step in the lab.

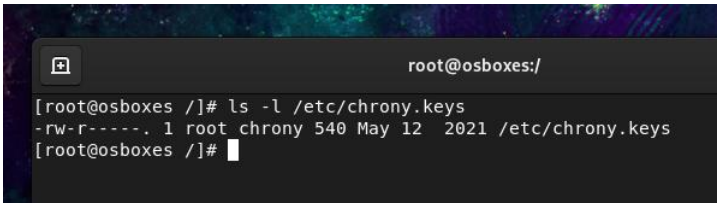
STEP 1. Open a terminal window. *(no need to provide screenshot for this step)*

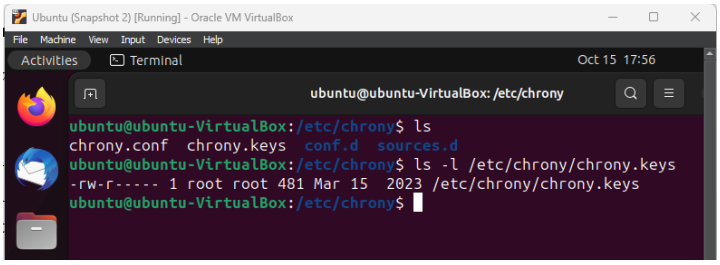
_____

STEP 2. Execute the correct command to display the permissions on the */etc/chrony.keys* file.

ls -l /etc/chrony.keys

or ls -l /etc/chrony/chrony.keys





STEP 3. Based on the output of the command from step 2, which user owns the */etc/chrony.keys* file?

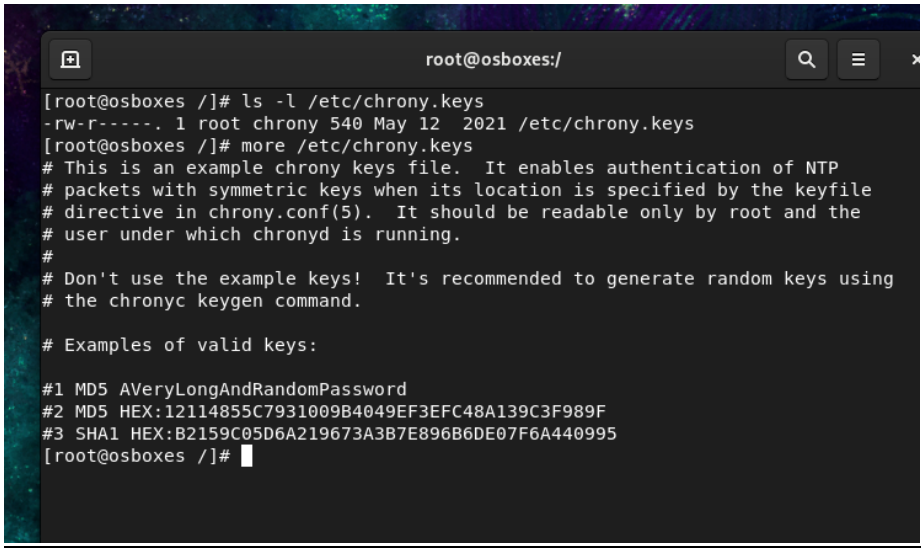
Owner

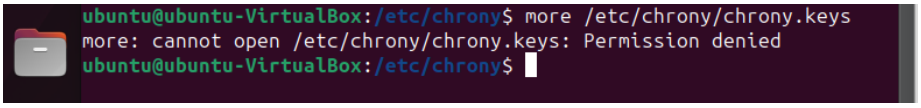
STEP 4. Based on the output of the command from step 2, which group owns the */etc/chrony.keys* file?

Group

STEP 5. Execute the *more/etc/chrony.keys* command and then explain why the command failed.

Because we don't have the necessary permissions to read the file.



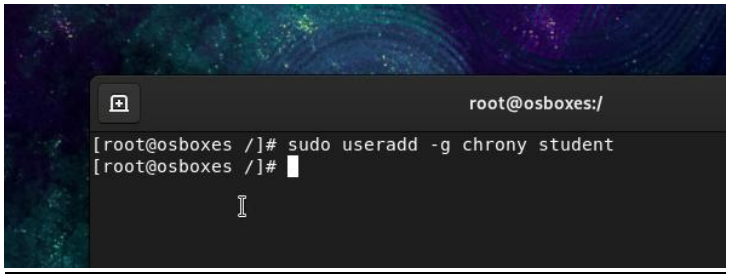


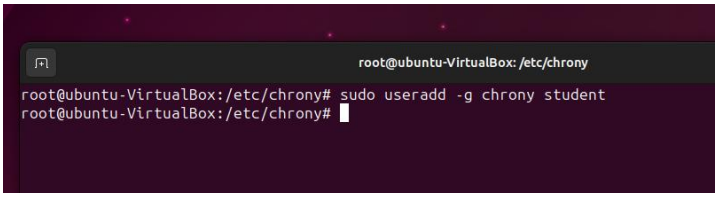
STEP 6. Switch to the root account using the *su* command.

Switching to root user using su or sudo su

STEP 7. Execute the correct command to add the student user to the chrony group.

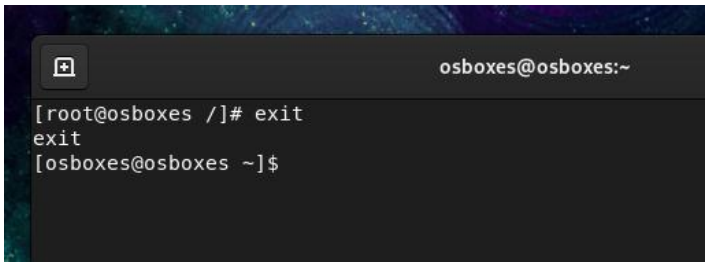
sudo useradd -g chrony student





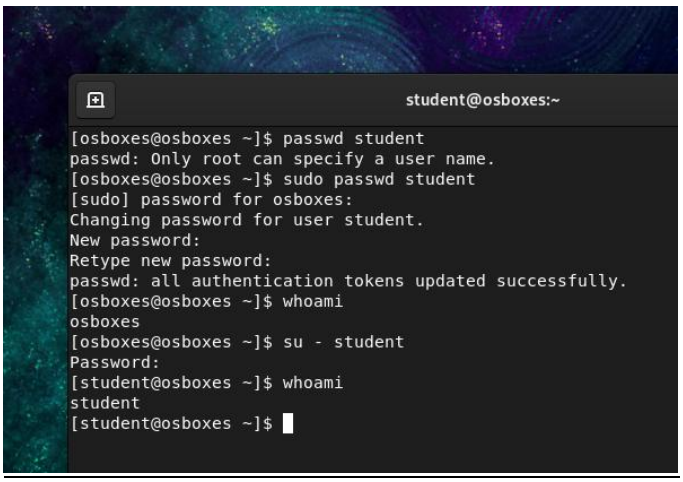
STEP 8. Log out of the system. (Note that this is necessary for the group ownership to take effect.)

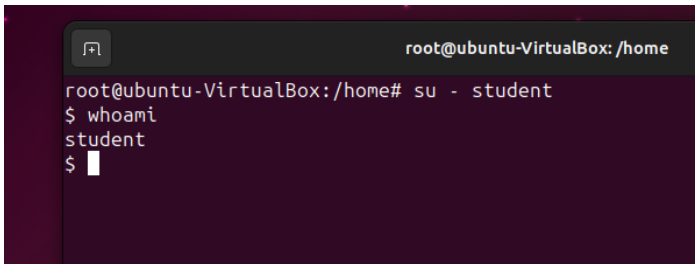
Logout or exit



STEP 9. Log in as the student user.

su – student

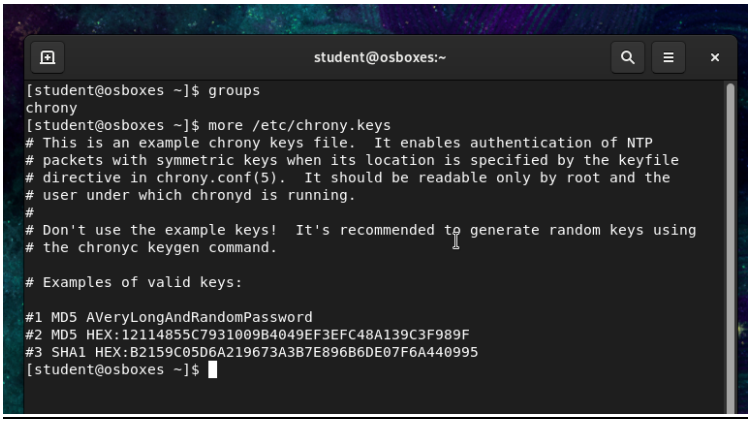




STEP 10. Open a terminal window. *(no need to provide screenshot for this step)*

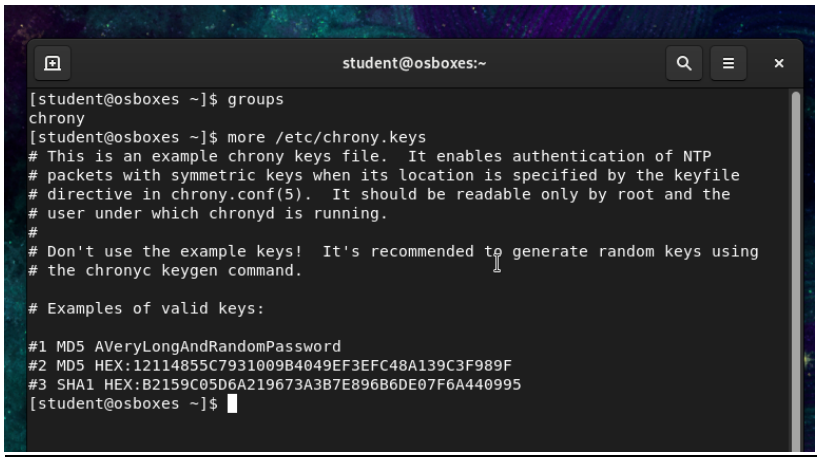
STEP 11. Execute the correct command to display the current user's groups.

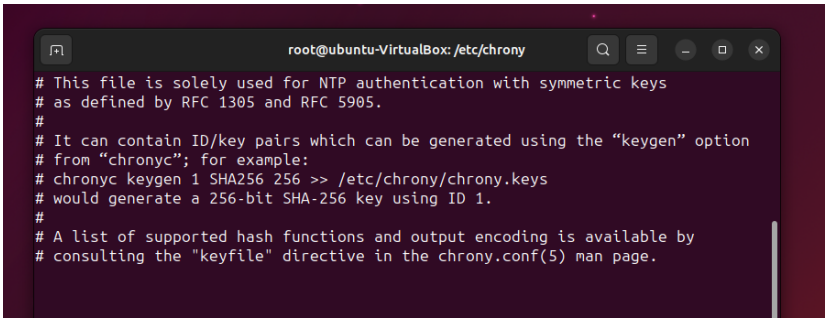
groups



STEP 12. Execute the *more/etc/chrony.keys* command to verify that this file's contents can now be displayed by the student user.

more/etc/chrony/chrony.keys

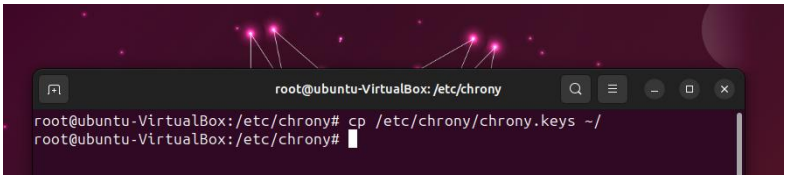




STEP 13. Copy the */etc/chrony.keys* file to the current directory (the home directory for the student user).

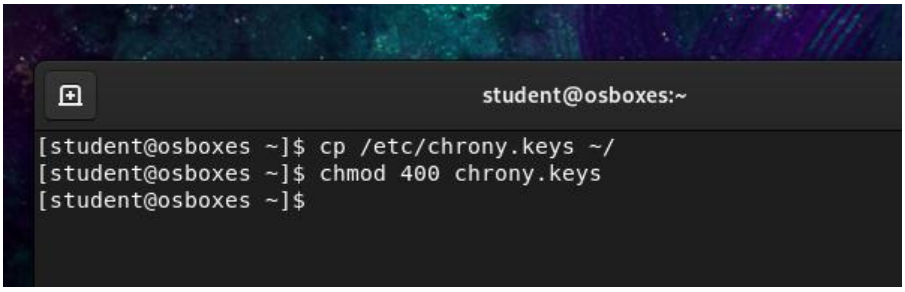
cp /etc/chrony/chrony.keys ~/

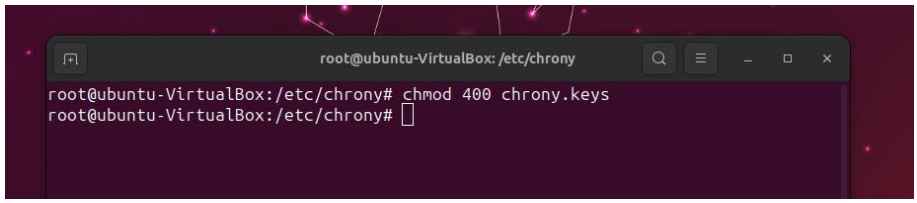
cp /etc/chrony.keys ~/



STEP 14. Using octal notation, change the permissions of the *chrony.keys* file that is in the current directory to *-r---------*.

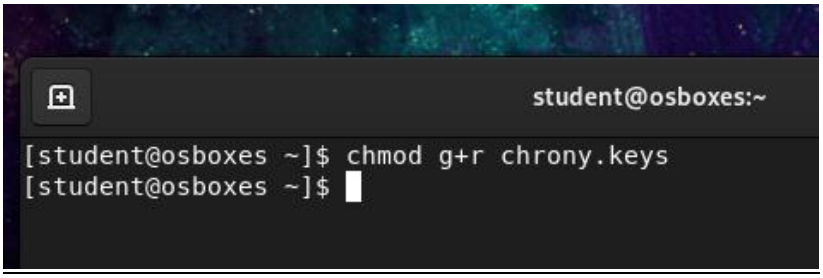
chmod 400 chrony.keys

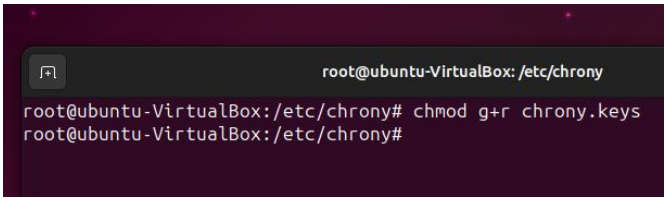




STEP 15. Using symbolic notation, change the permissions of the *chrony.keys* file that is in the current directory to allow group members the read permission.

chmod g+r chrony.keys

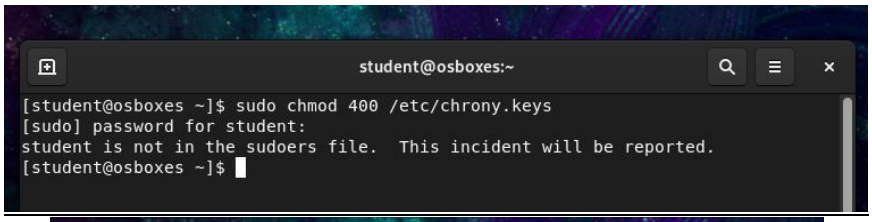


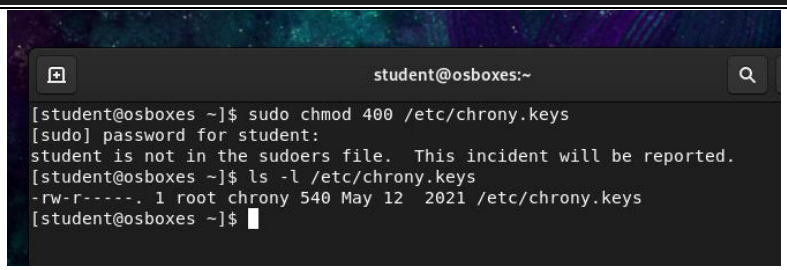


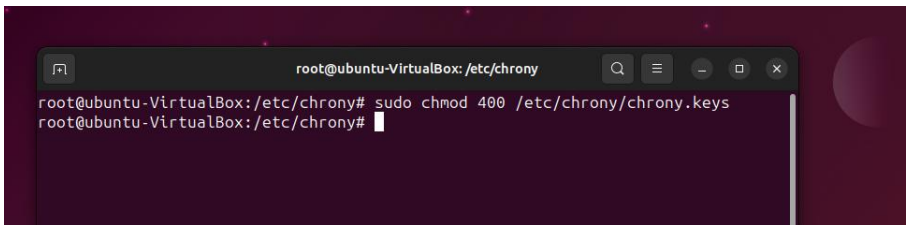
STEP 16. Using octal notation, try to change the permissions of the *chrony.keys* file that is in the */etc* directory to *-r---------*. Explain why this command fails.

sudo chmod 400 /etc/chrony.keys

//This command fails due to lack of permissions or due to SELinux or AppArmor Policies

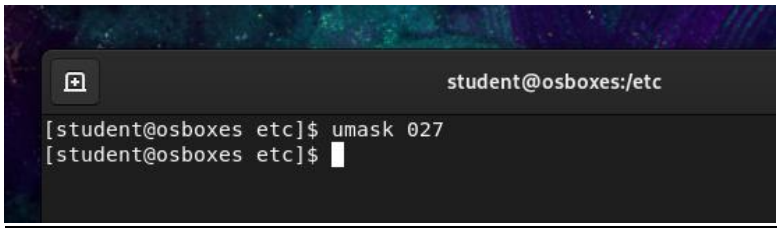


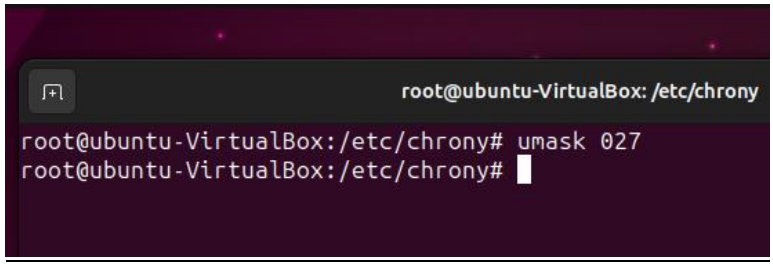




STEP 17. Change the mask value for the current shell so any new directory would have the following permissions: *drwxr-x---*

umask 027





STEP 18. Based on the mask value from step 17, what permissions would all new files that are created in this shell have?

_____*drwxr-x---*_____

The permissions `drwxr-x---` can be classified as follows:
- The leading `d` indicates that it's a directory.
- Owner (User): Read, Write, and Execute (rwx)
- Group: Read and Execute (r-x)
- Others: No Permissions (---)

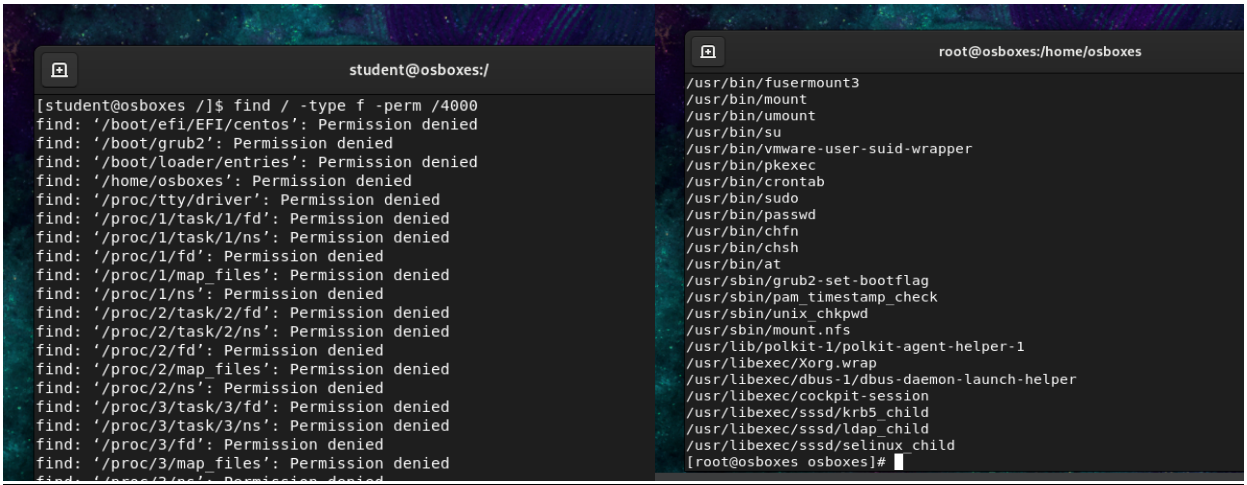
Lab 9.2 Managing Special Permissions

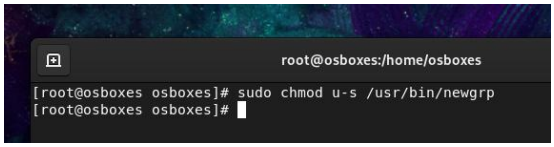
List the commands you will use without the need to provide a screenshot

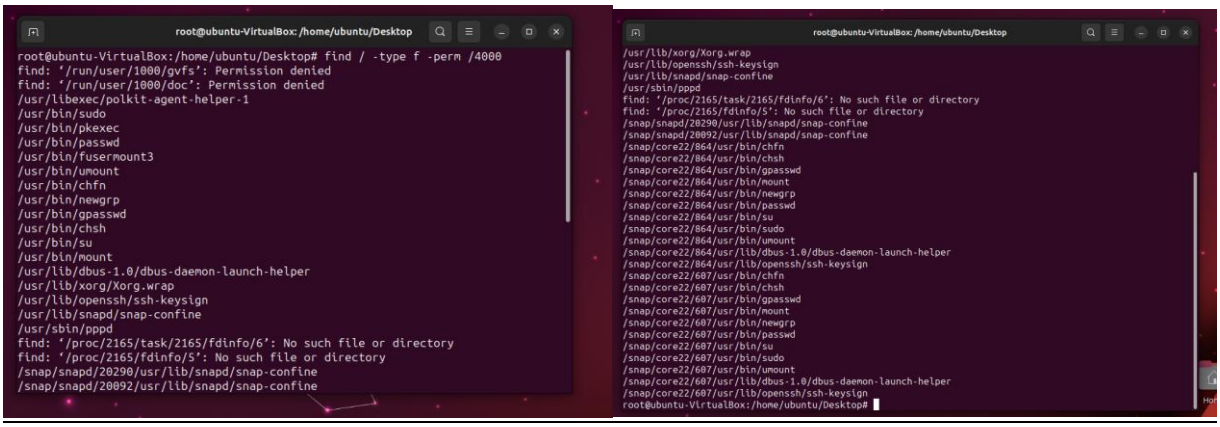
Scenario 1: You are concerned about SUID permissions on this system. Start by running the command to find all files that have SUID permissions set. Then change the *newgrp* command so it is not a SUID file.

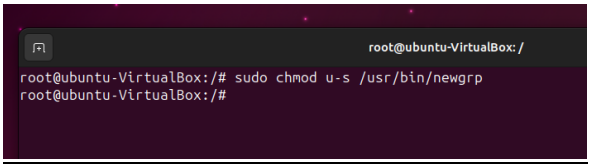
find / -type f -perm /4000

sudo chmod u-s /usr/bin/newgrp



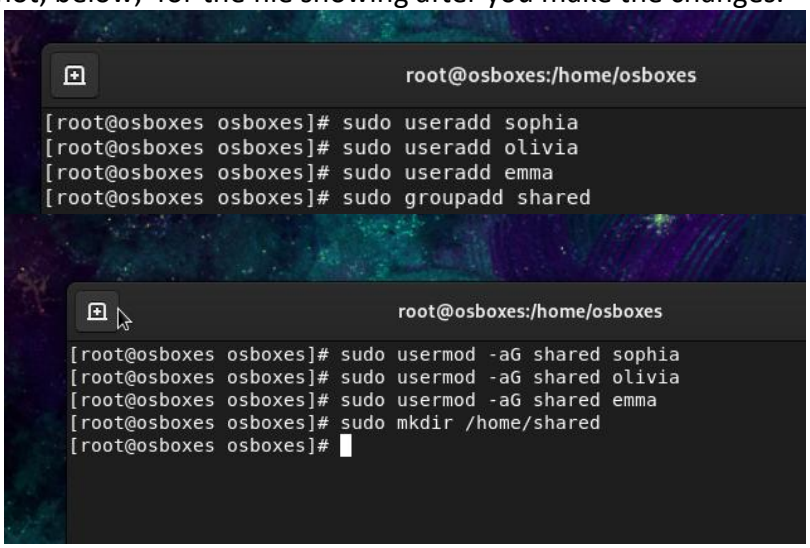


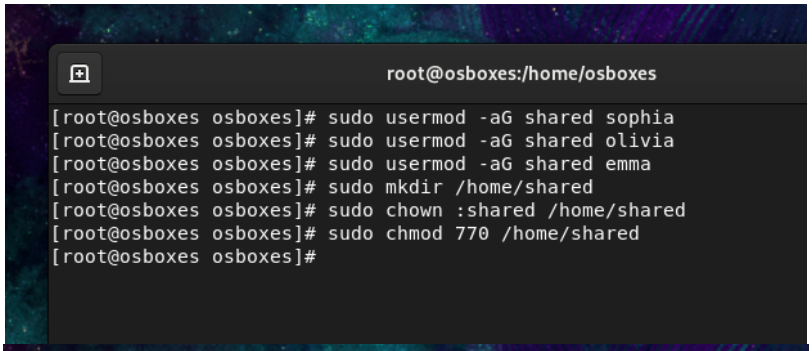


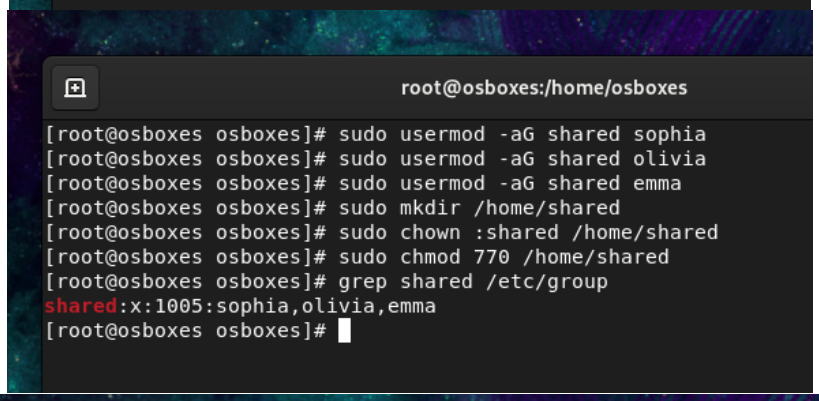


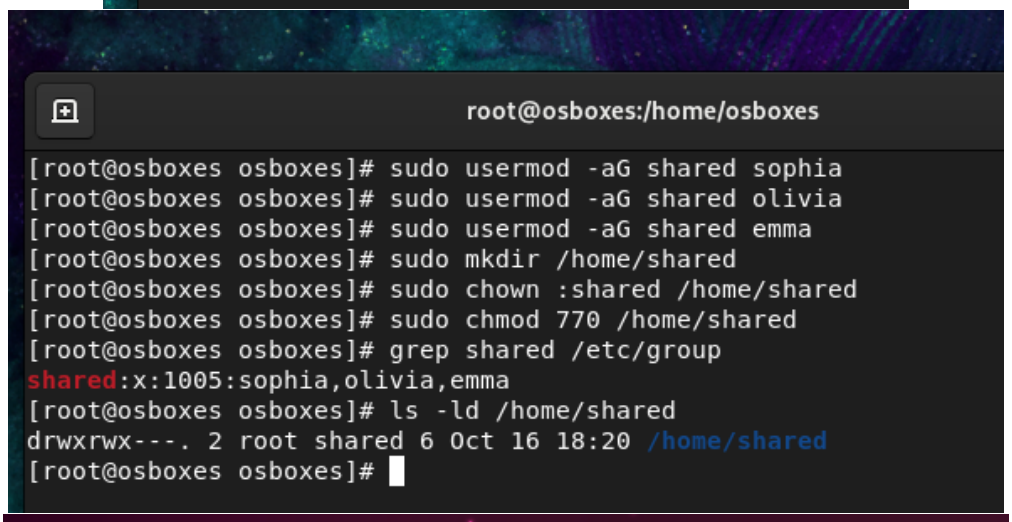
Scenario 2: Three users need to share files on the system, but no other user can have access to these files. Create user accounts for sophia, olivia, and emma, using default values for account parameters. Create a common group named shared, which will be used to allow these three users to share files with each other. Finally, create a directory named */home/shared* for only these three users to access and to also automatically give group ownership of all new files to the shared group.

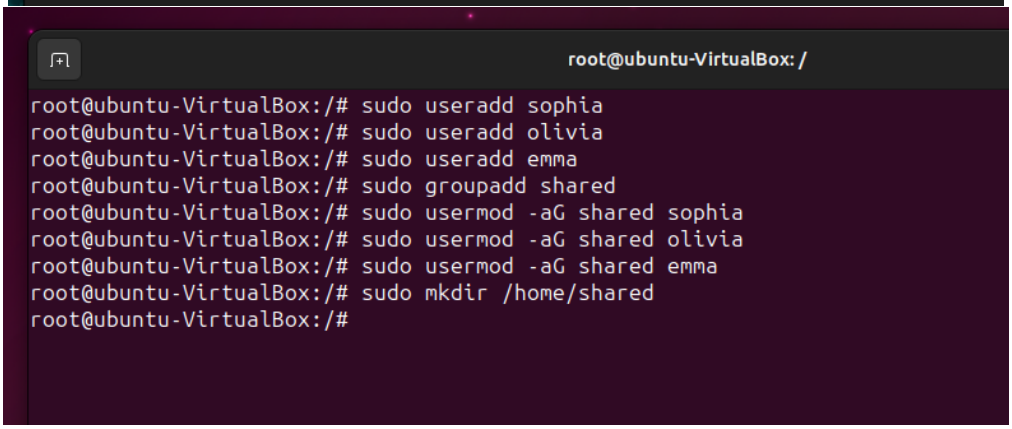
paste a screen shot, below, for the file showing after you make the changes.

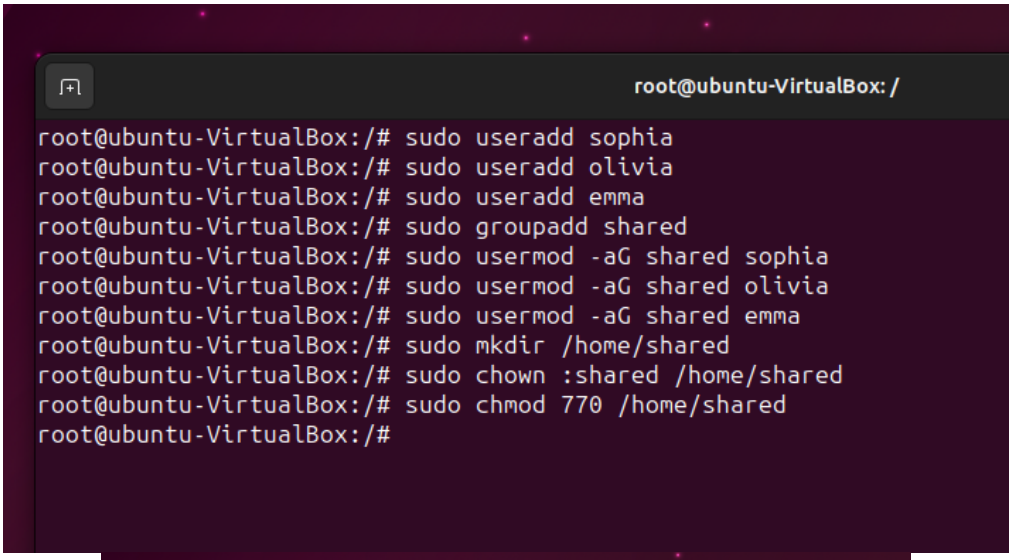


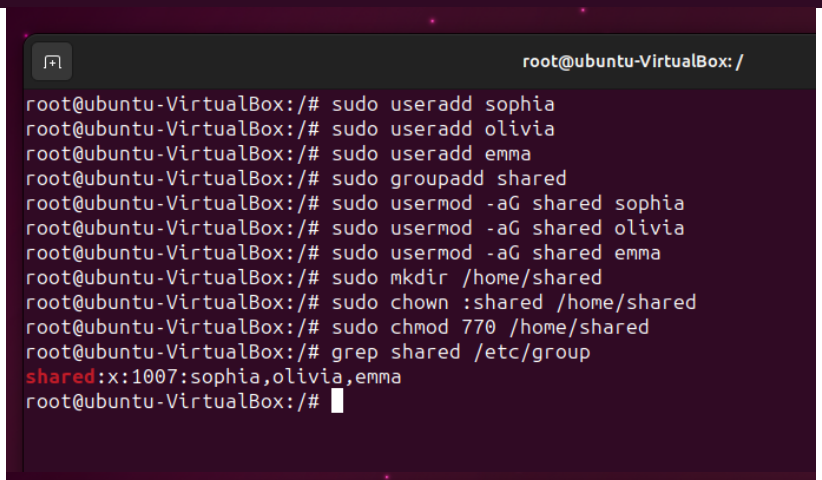

root@osboxes:/home/osboxes
[root@osboxes osboxes]# sudo usermod -aG shared sophia
[root@osboxes osboxes]# sudo usermod -aG shared olivia
[root@osboxes osboxes]# sudo usermod -aG shared emma
[root@osboxes osboxes]# sudo mkdir /home/shared
[root@osboxes osboxes]# sudo chown :shared /home/shared
[root@osboxes osboxes]# sudo chmod 770 /home/shared
[root@osboxes osboxes]#

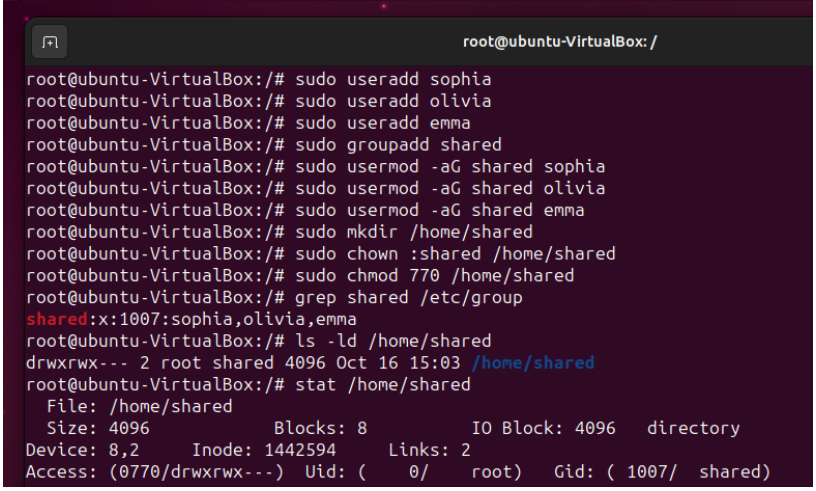

root@osboxes:/home/osboxes
[root@osboxes osboxes]# sudo usermod -aG shared sophia
[root@osboxes osboxes]# sudo usermod -aG shared olivia
[root@osboxes osboxes]# sudo usermod -aG shared emma
[root@osboxes osboxes]# sudo mkdir /home/shared
[root@osboxes osboxes]# sudo chown :shared /home/shared
[root@osboxes osboxes]# sudo chmod 770 /home/shared
[root@osboxes osboxes]# grep shared /etc/group
shared:x:1005:sophia,olivia,emma
[root@osboxes osboxes]#


root@osboxes:/home/osboxes
[root@osboxes osboxes]# sudo usermod -aG shared sophia
[root@osboxes osboxes]# sudo usermod -aG shared olivia
[root@osboxes osboxes]# sudo usermod -aG shared emma
[root@osboxes osboxes]# sudo mkdir /home/shared
[root@osboxes osboxes]# sudo chown :shared /home/shared
[root@osboxes osboxes]# sudo chmod 770 /home/shared
[root@osboxes osboxes]# grep shared /etc/group
shared:x:1005:sophia,olivia,emma
[root@osboxes osboxes]# ls -ld /home/shared
drwxrwx---. 2 root shared 6 Oct 16 18:20 /home/shared
[root@osboxes osboxes]#


root@ubuntu-VirtualBox: /
root@ubuntu-VirtualBox:/# sudo useradd sophia
root@ubuntu-VirtualBox:/# sudo useradd olivia
root@ubuntu-VirtualBox:/# sudo useradd emma
root@ubuntu-VirtualBox:/# sudo groupadd shared
root@ubuntu-VirtualBox:/# sudo usermod -aG shared sophia
root@ubuntu-VirtualBox:/# sudo usermod -aG shared olivia
root@ubuntu-VirtualBox:/# sudo usermod -aG shared emma
root@ubuntu-VirtualBox:/# sudo mkdir /home/shared
root@ubuntu-VirtualBox:/#







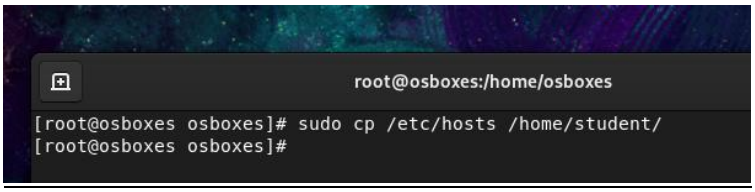
Lab 9.3 Enabling Access Control Lists

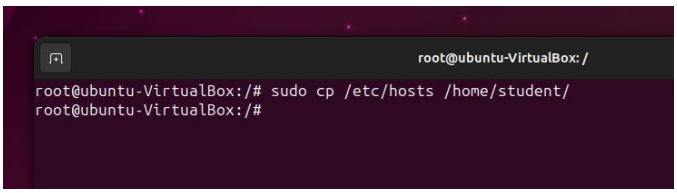
Aside of step 1, provide a screenshot evidence of the successful execution of the command run below each step in the lab.

STEP 1. Open a terminal window. *(no need to provide screenshot for this step)*

STEP 2. Copy the */etc/hosts* file to the student user's home directory.

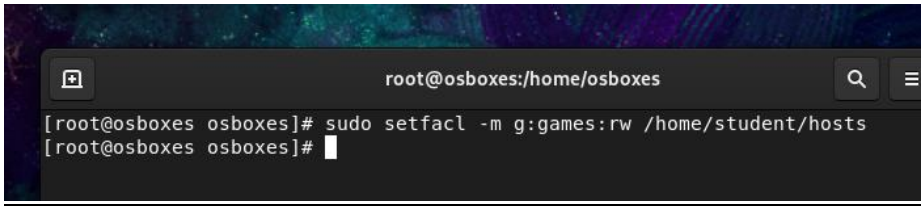
sudo cp /etc/hosts /home/student/

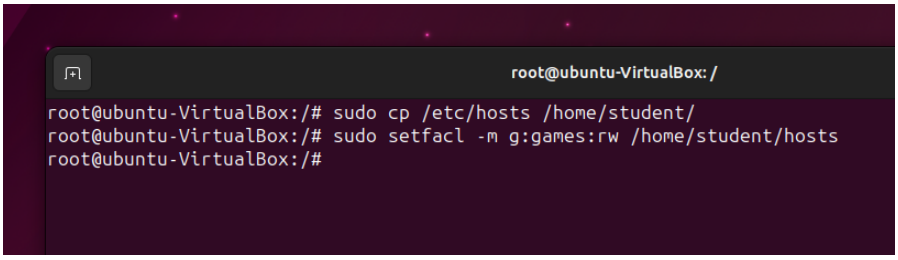




STEP 3. Set an access control list (ACL) for the games group that allows read and write permissions for that group on the *hosts* file.

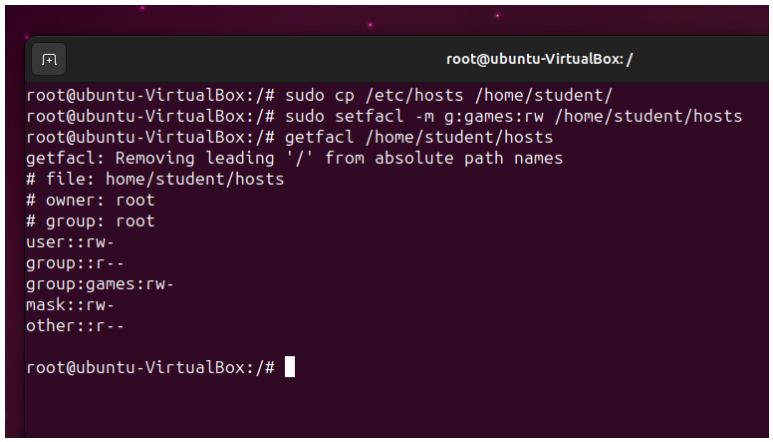
sudo setfacl -m g:games:rw /home/student/hosts





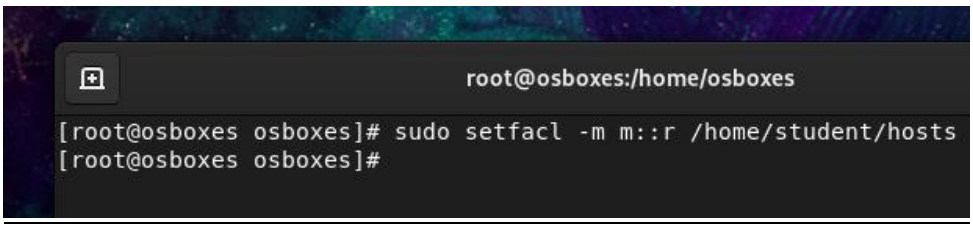
STEP 4. Display the ACLs for the *hosts* file.

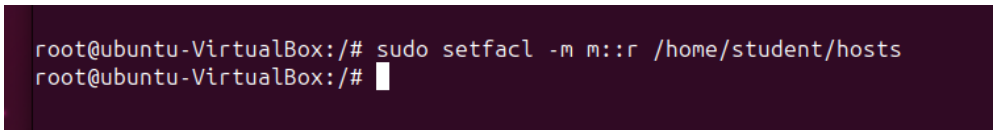
getfacl /home/student/hosts



STEP 5. Change the ACL mask value to read-only for the *hosts* file.

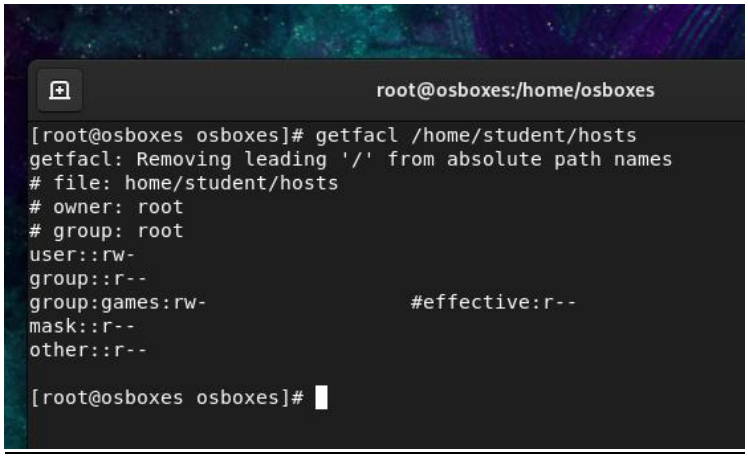
sudo setfacl -m m::r /home/student/hosts

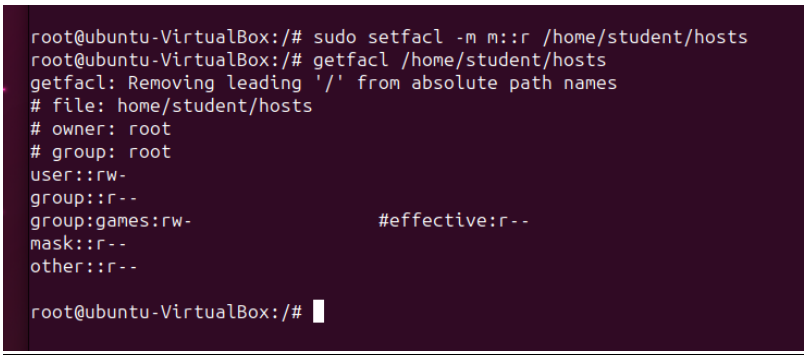




STEP 6. Display the ACLs for the *hosts* file to verify the ACL mask.

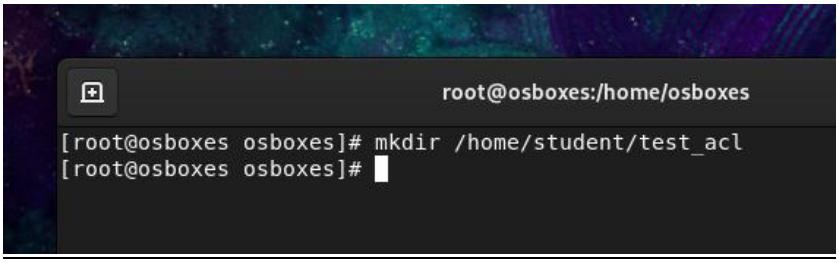
getfacl /home/student/hosts

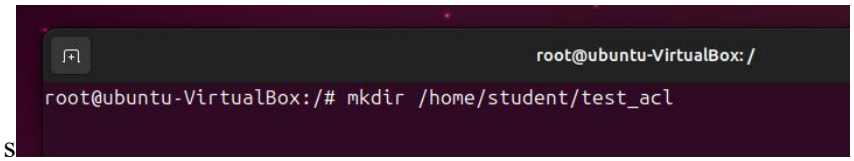




STEP 7. Create a directory called *test_acl*.

mkdir /home/student/test_acl

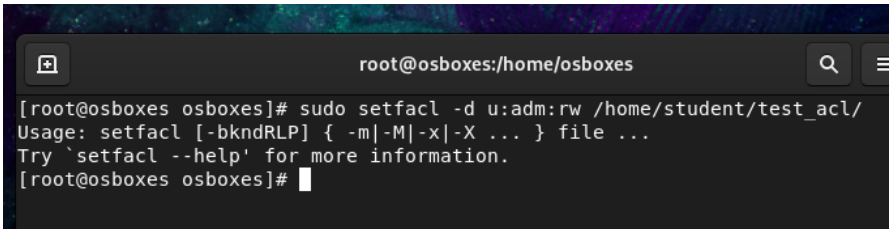


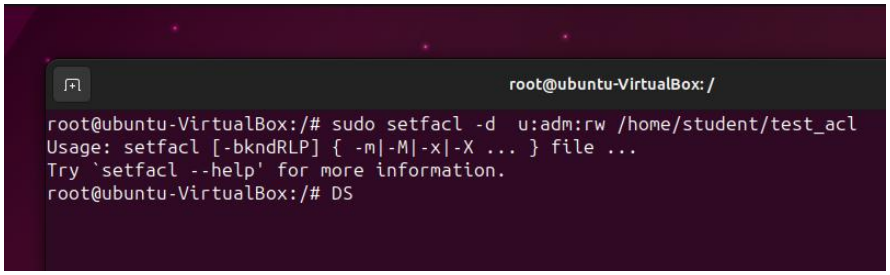


s

STEP 8. Create a default ACL for the *test_acl* directory so the adm user has read and write permissions for all new files and directories created in the *test_acl* directory.

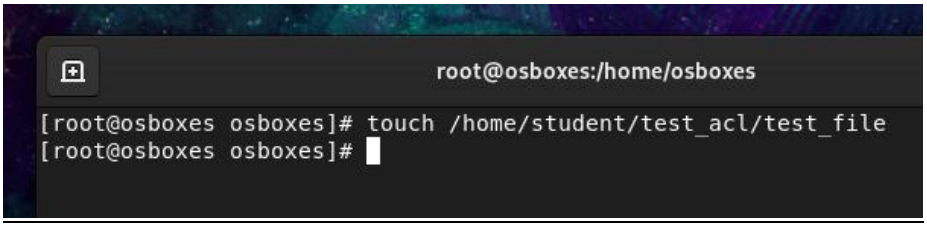
sudo setfacl -d u:adm:rw /home/student/test_acl

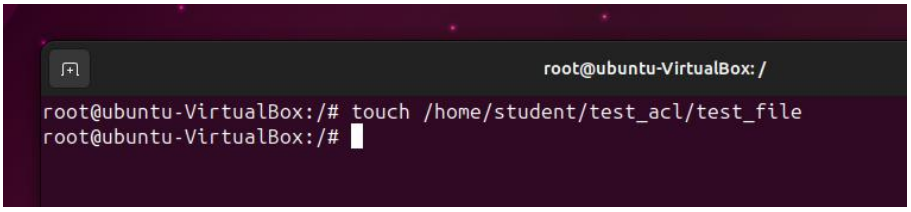




STEP 9. Use the *touch* command to create a new file named *test_file* in the *test_acl* directory.

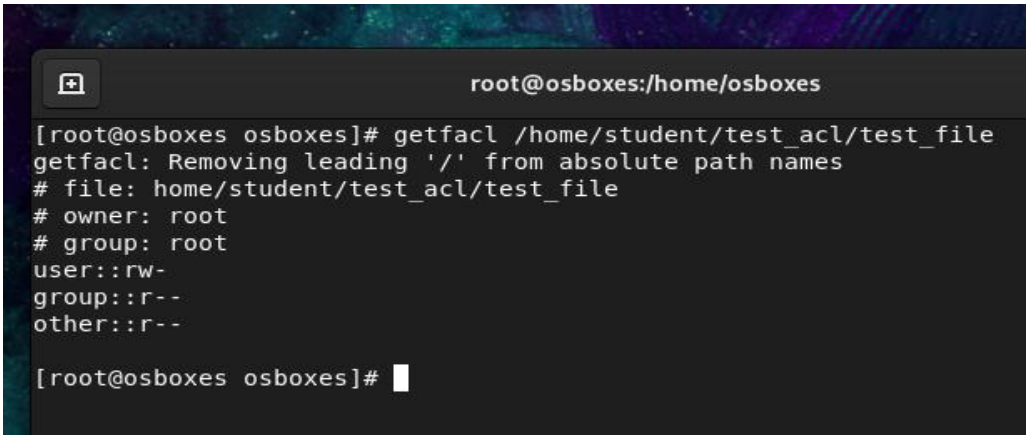
touch /home/student/test_acl/test_file

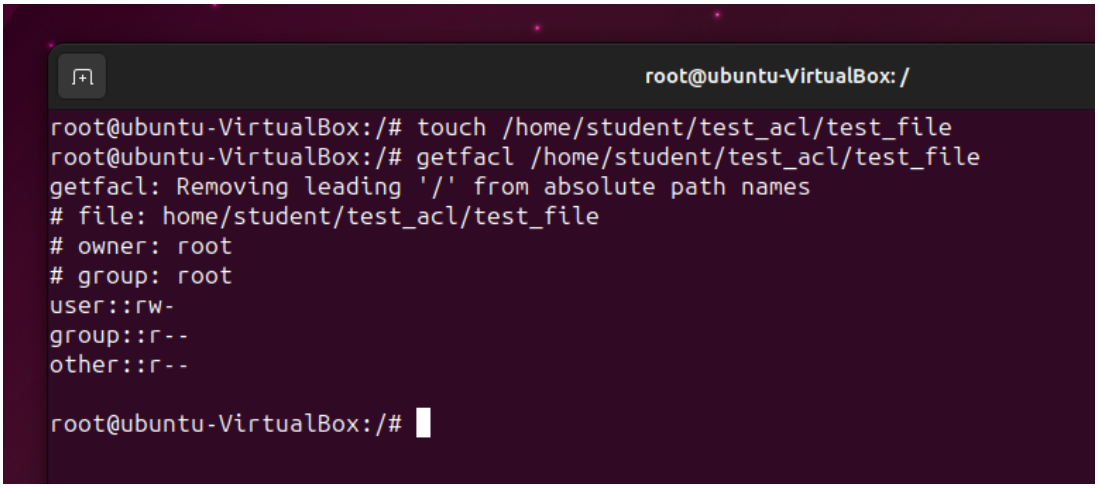




STEP 10. Verify the default ACLs by viewing the ACLs for the *test_file* file.

getfacl /home/student/test_acl/test_file





Lab 9.4 Managing File Ownership and Attributes

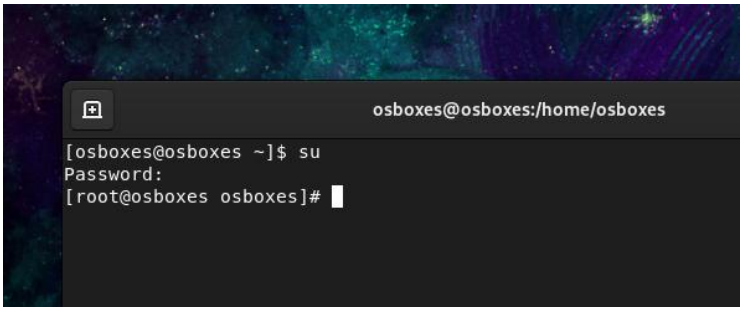
Aside of step 1, provide a screenshot evidence of the successful execution of the command run below each step in the lab.

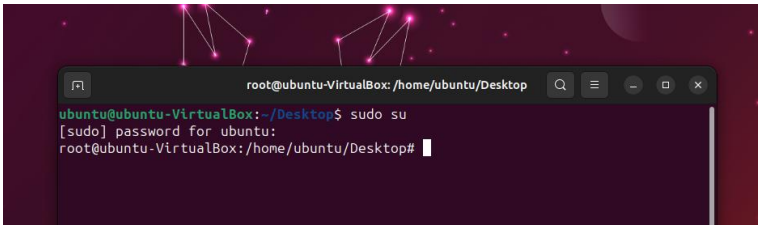
STEP 1. Open a terminal window. *(no need to provide screenshot for this step)*

_____

STEP 2. Use the *su* command to switch to the root account.

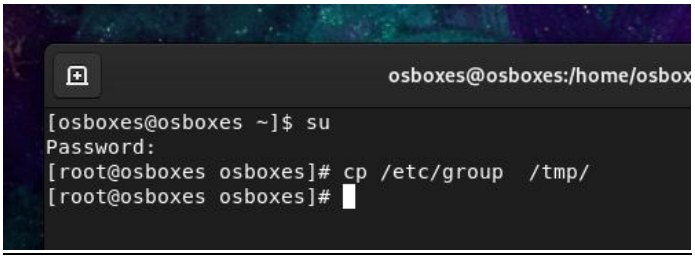
_____su_____





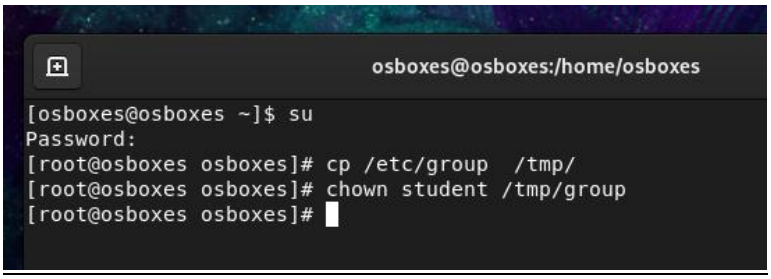
STEP 3. Copy the */etc/group* file to the */tmp* directory.

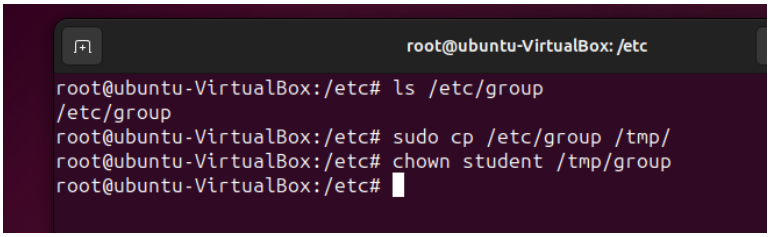
cp /etc/group /tmp/



STEP 4. Change the user ownership of the */tmp/group* file to the student user.

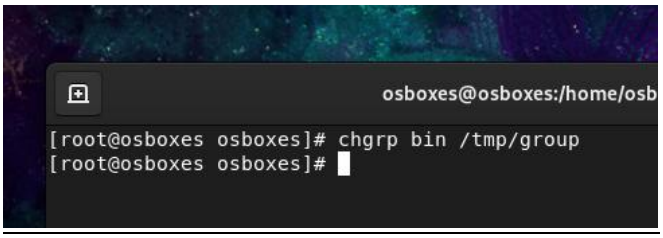
chown student /tmp/group

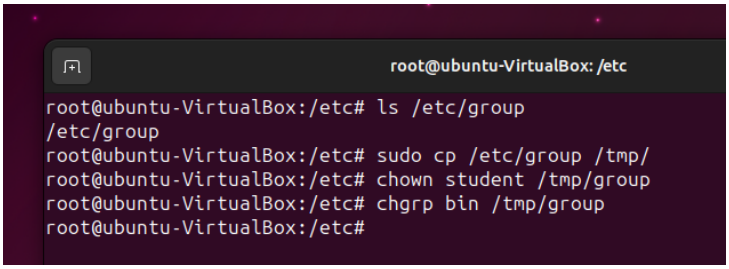




STEP 5. Change the group ownership of the */tmp/group* file to the bin group.

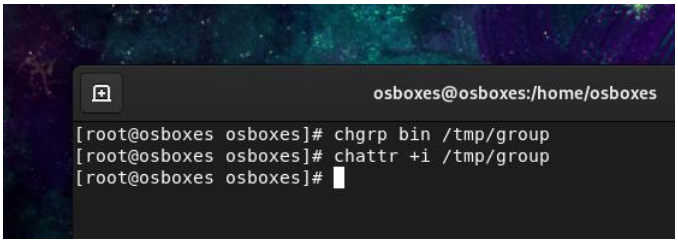
chgrp bin /tmp/group

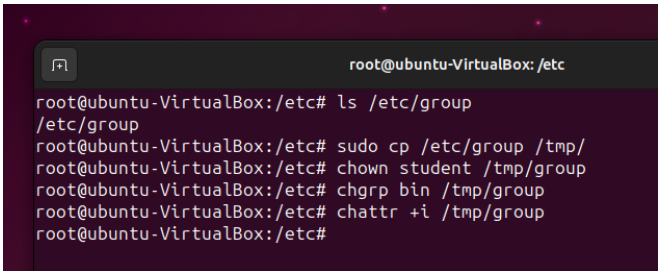




STEP 6. Make the */tmp/group* file immutable.

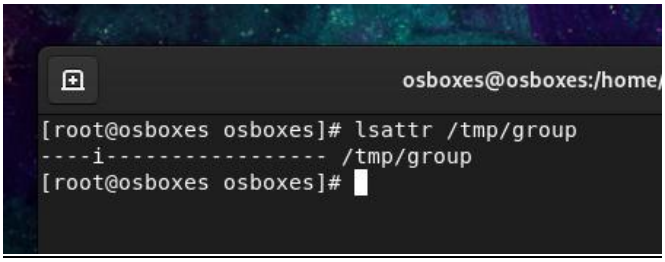
chattr +i /tmp/group

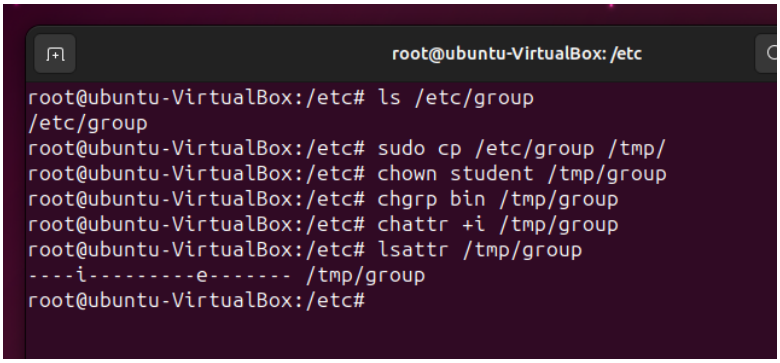




STEP 7. Run the correct command to display the file attributes of the */tmp/group* file.

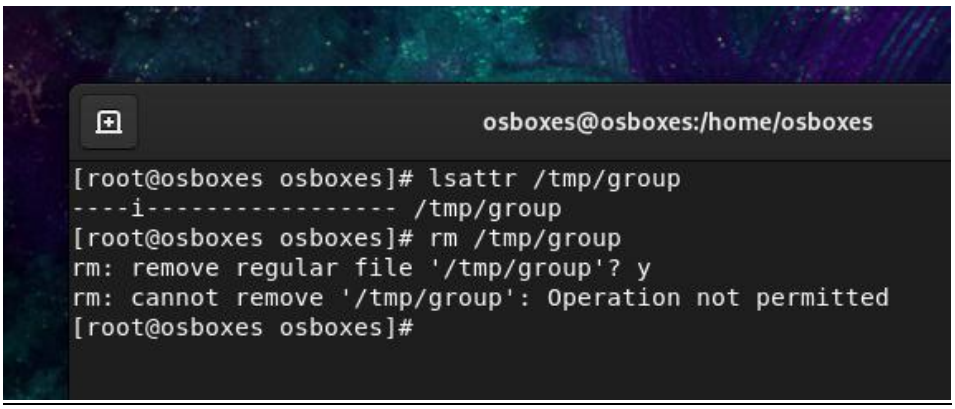
lsattr /tmp/group

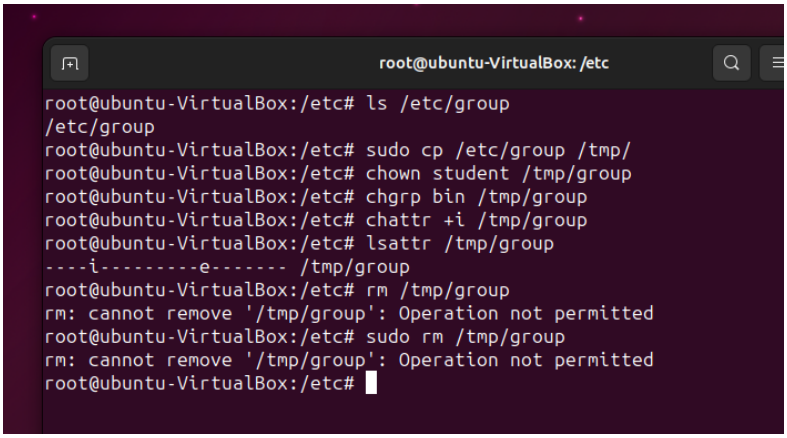




STEP 8. Attempt to remove the */tmp/group* file. Explain why this fails.

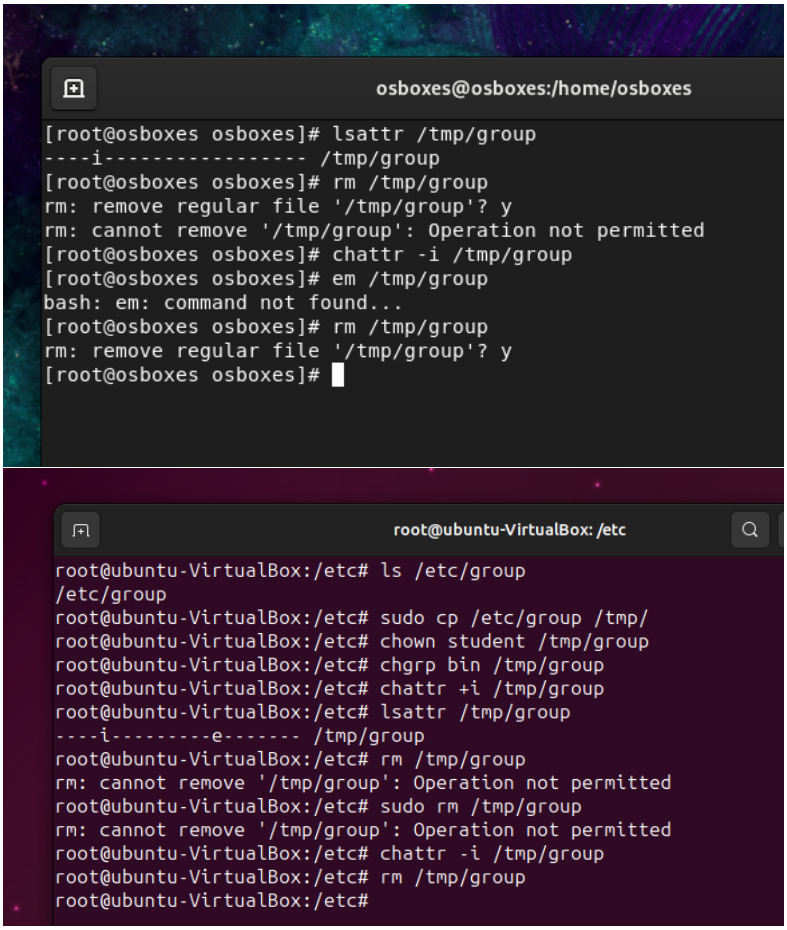
rm /tmp/group





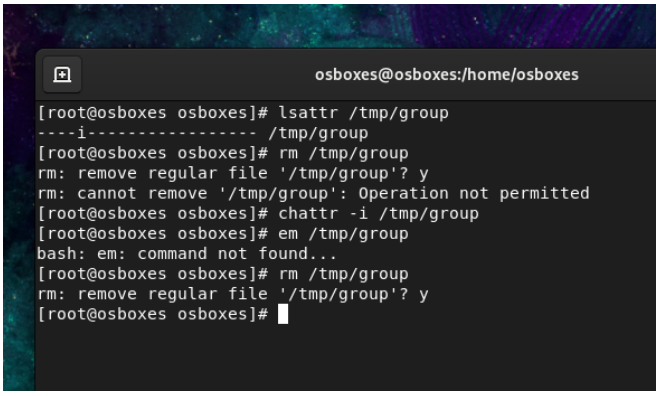
STEP 9. Remove the immutable attribute from the */tmp/group* file.

chattr -i /tmp/group



STEP 10. Remove the /tmp/group file.

rm /tmp/group



Lab 9.5 Monitoring Security Issues with SELinux

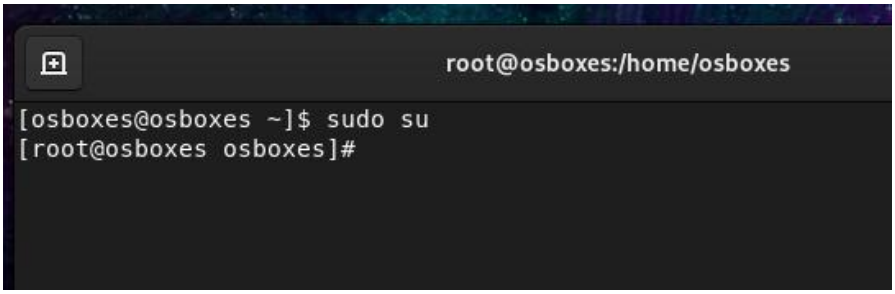
Aside of step 1, provide a screenshot evidence of the successful execution of the command run below each step in the lab.

STEP 1. Open a terminal window. *(no need to provide screenshot for this step)*

_____

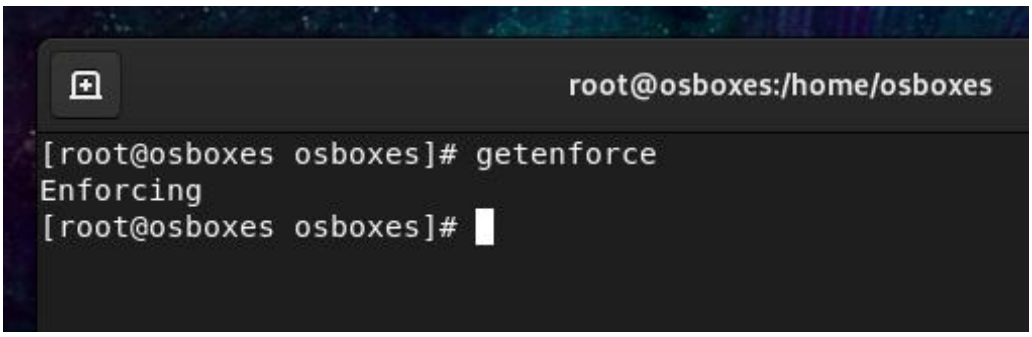
STEP 2. Use the *su* command to switch to the root account.

_____su_____



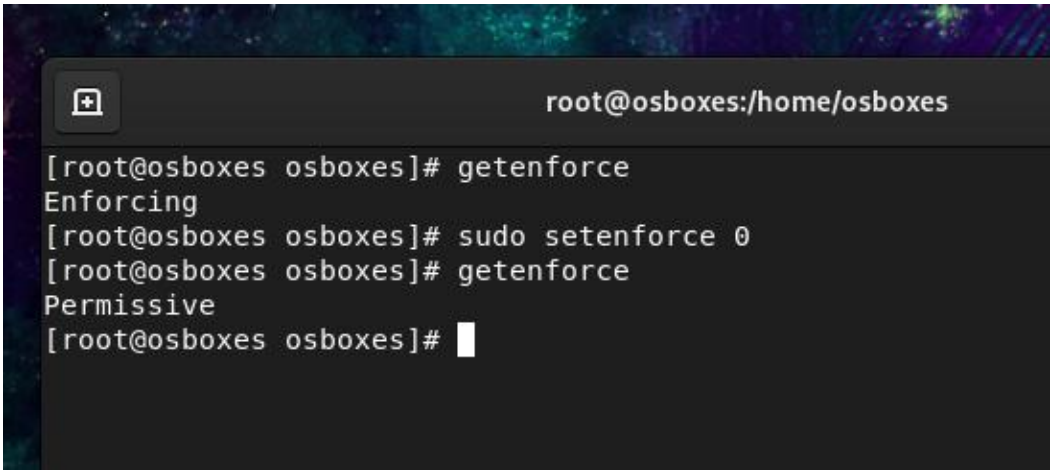
STEP 3. Execute the correct command to determine what mode (enforcing or permissive) SELinux is currently in.
<<check the remarks in the book for this step>>

_____getenforce_____



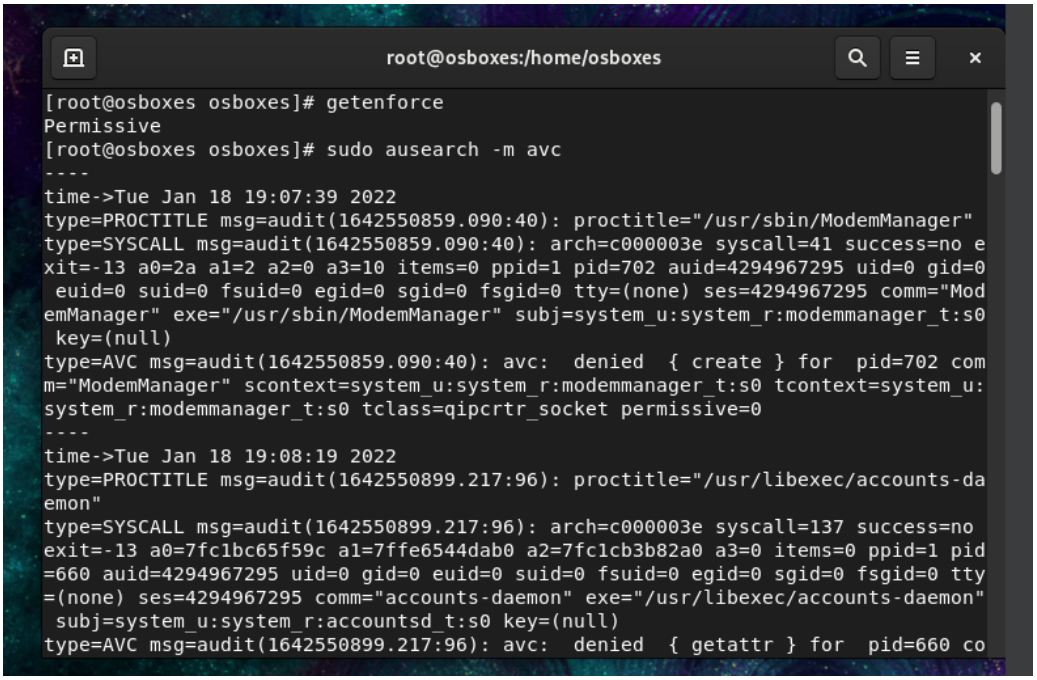
STEP 4. Change the current mode to permissive.

sudo setenforce 0



STEP 5. Display any log entries related to SELinux.

sudo ausearch -m avc



STEP 6. Change the current mode to enforcing. <<check the remarks in the book for this step>>

sudo setenforce 1