# CMSE353 Fall 2021

# Access Control in Linux

Contents

[CMSE353 Fall 2021](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618531)

[Access Control in Linux](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618532)

[4. Unix file permissions and inodes 5](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618539)

[Example 1 6](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618540)

[Example 2 6](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618541)

[Example 3 7](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618542)

[Example 4 7](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618543)

[Example 5 7](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618544)

[5. Octal representation of permissions 7](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618545)

[Example 6 8](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618546)

[6. Changing file permissions on Linux system, adding users, groups, changing file owner and group 8](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618547)

[Example 7 9](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618548)

[Example 8 9](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618549)

[Example 9 10](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618550)

[Example 10 11](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618551)

[Example 11 11](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618552)

[Example 12 12](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618553)

[Example 14 12](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618554)

[Example 15 12](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618555)

[Example 16 13](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618556)

[Example 17 13](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618557)

[Example 18 13](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618558)

[Example 19 13](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618559)

[Example 20 14](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618560)

[Example 21 14](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618561)

[Example 22 14](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618562)

[Example 23 15](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618563)

[Example 24 15](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618564)

[Example 25 15](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618565)

[Challenge 1. 15](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618566)

[7. Command umask 15](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618567)

[Example 26 16](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618568)

[Challenge 2. 16](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618569)

[8. Set UID (SUID) 16](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618570)

[Example 27 16](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618571)

[Example 28 17](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618572)

[Example 29 17](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618573)

[Example 30 18](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618574)

[9. Writing a SUID program in C 18](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618575)

[Example 31 18](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618576)

[Example 32 19](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618577)

[Example 33 20](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618578)

[Example 34 20](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618579)

[Example 35 21](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618580)

[Challenge 3 21](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618581)

[Challenge 4 21](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618582)

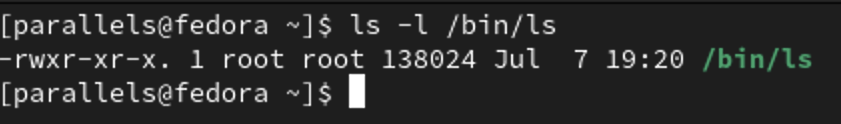
[Challenge 5 21](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618583)

[Challenge 6 21](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618584)

[10. Linux Extended ACLs 22](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618585)

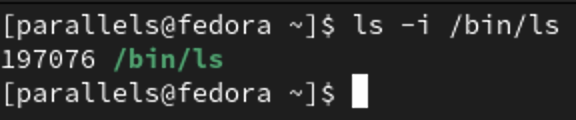
[Example 36 22](file:///C:/Users/SAI/Desktop/CMSE353-Lab/Lab%201%20Description%20-%20Access%20ontrol%2020102021%20(1).docx#_Toc85618586)  
  
**The distribution of tasks**Sinem & Emre :  
Seyit :  
Jamshid :

Example - 1



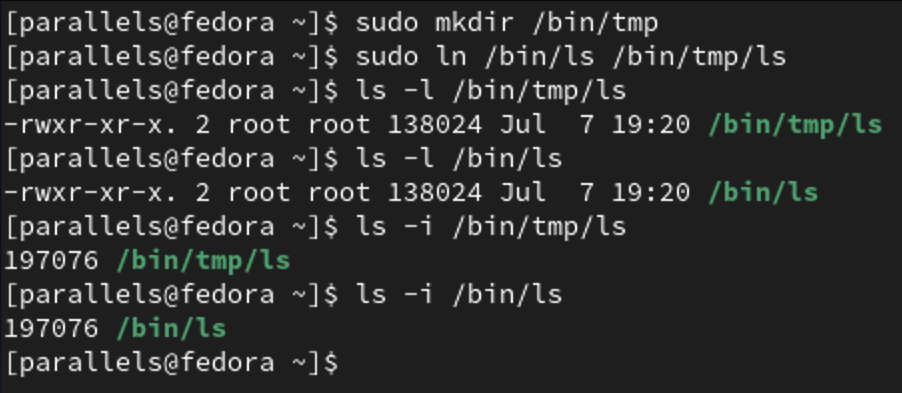
Defines the location of the file on disk, along with attributes including the Unix file permissions, and access times for the file.

Example - 2



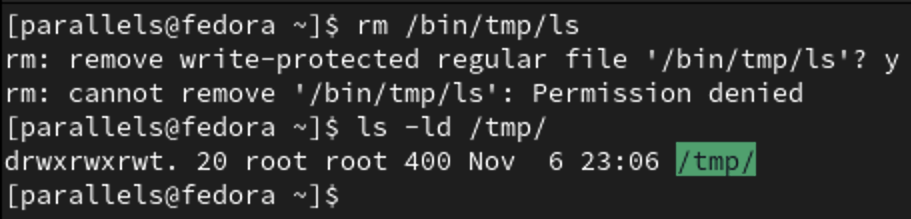
ls-i display the inode number of the given path.

Example – 3



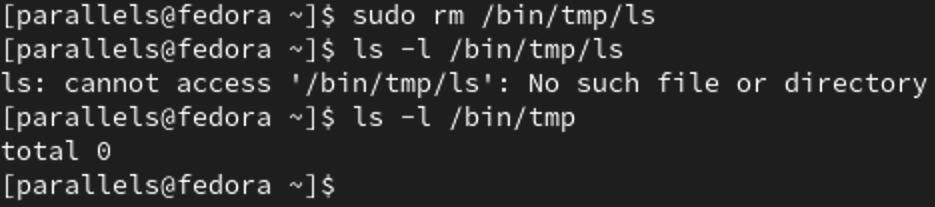
We see that the both files share the same inode number. if changing one of these files will affect the other.

Example – 4



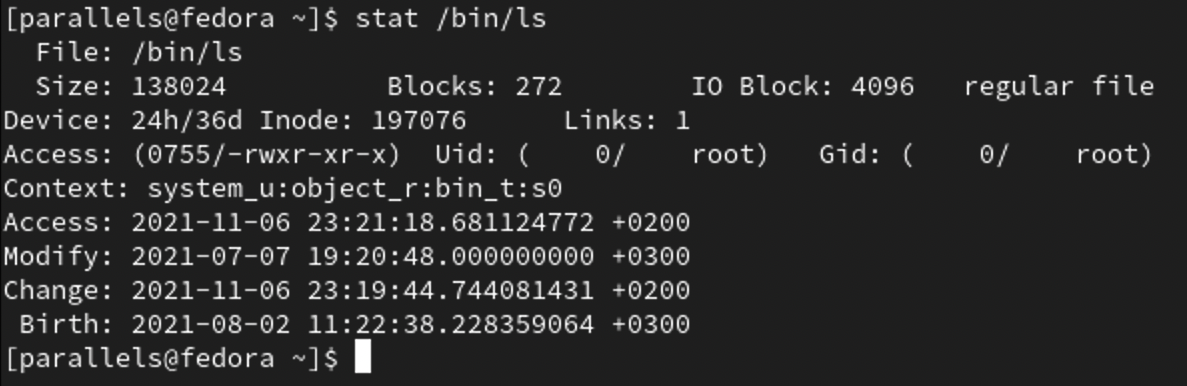
We can see the /bin/ls file as a normal user but can not delete that link from sticky bit for /tmp/ directory.

Example – 5



If we add sudo to delete a sticky bit

Example – 6



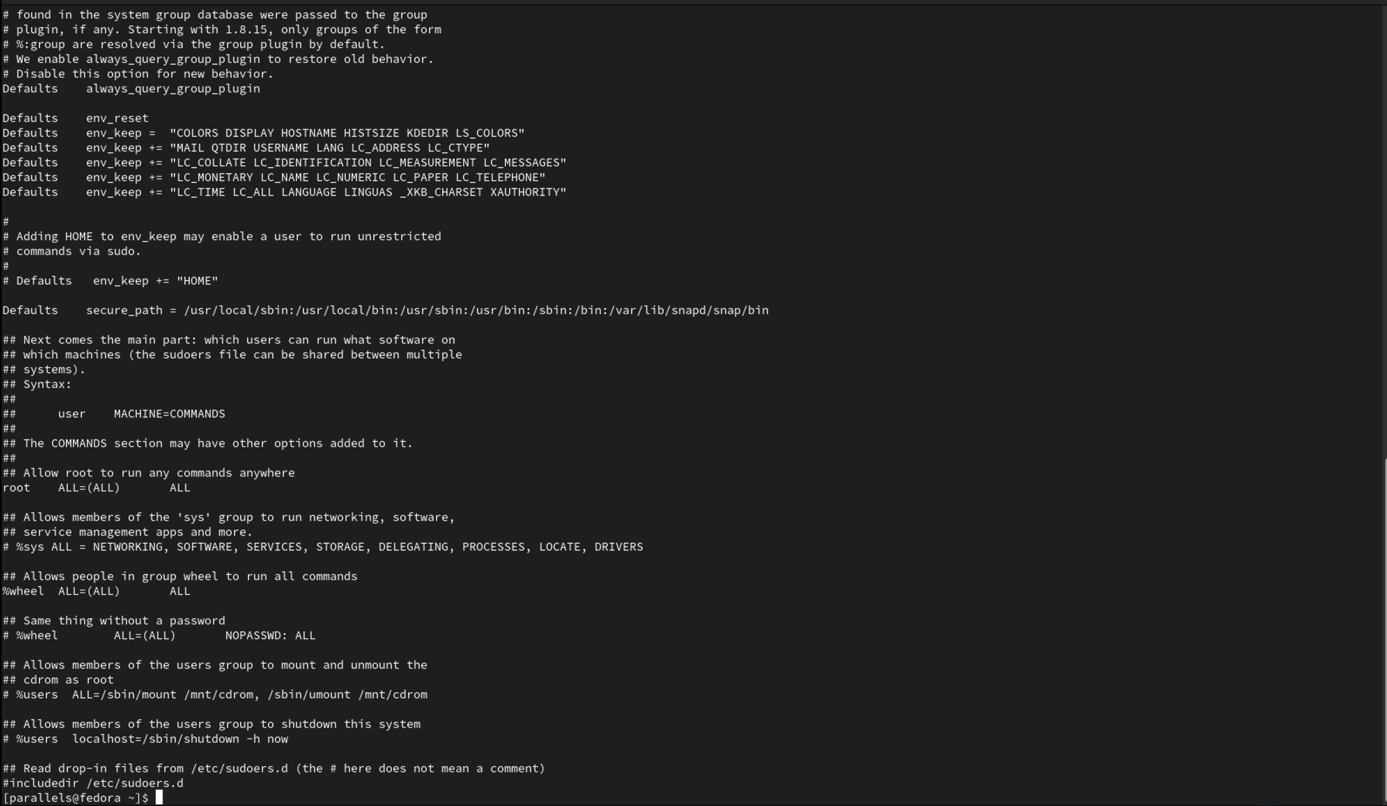
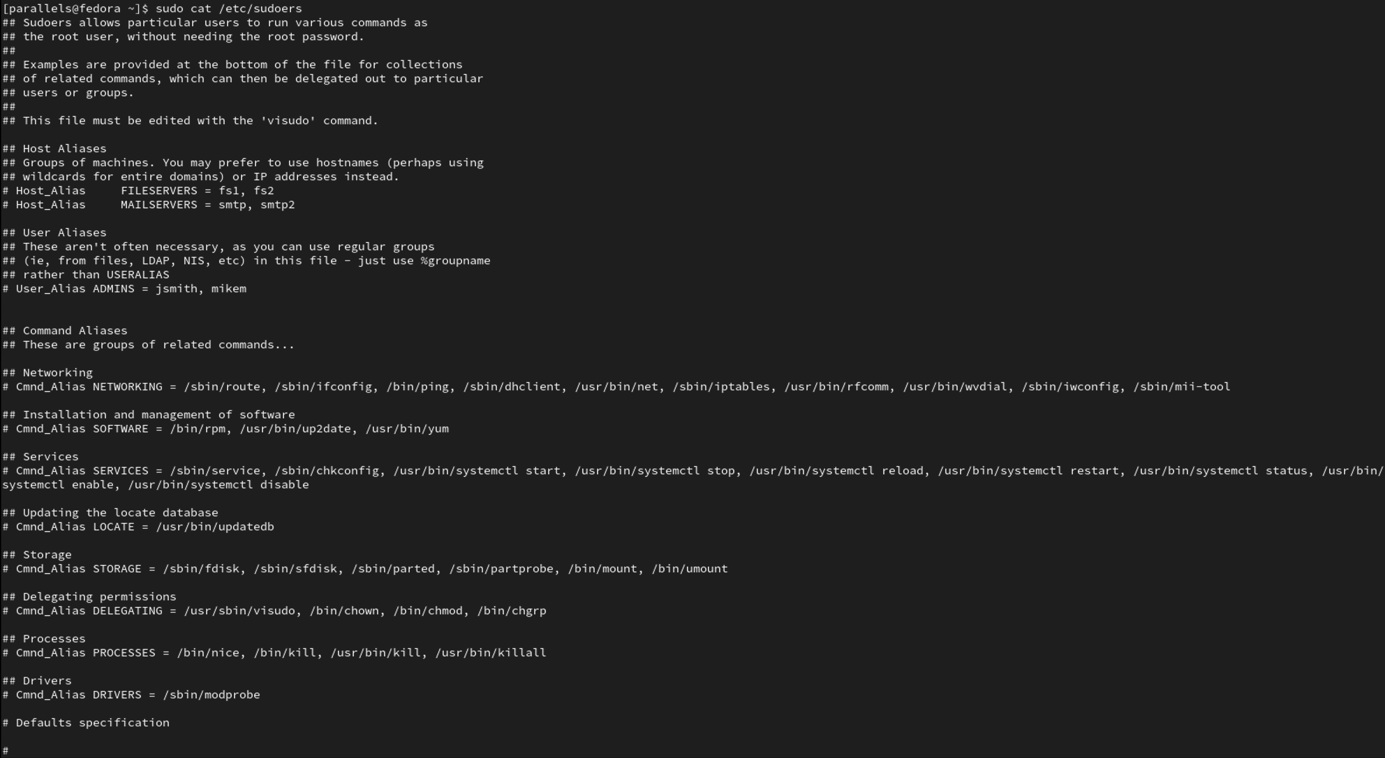
Stat command display the output includes the access rights, along

with the last time the file was accessed, modified, and when the inode was last changed

Example – 7



We can read every user

Example – 8

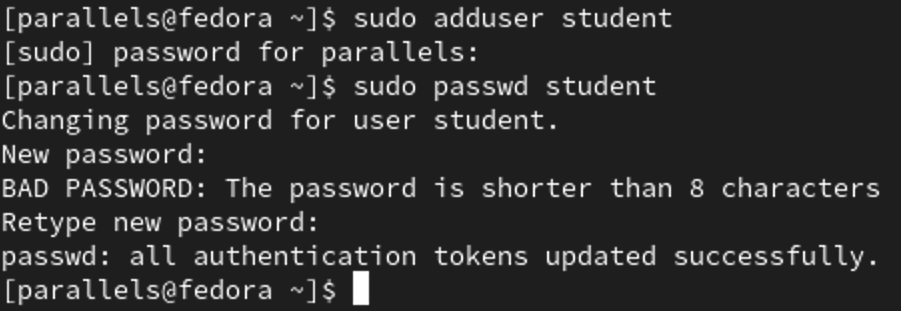
The users allowed doing sudo (from sudo group) are enlisted file

Example – 9



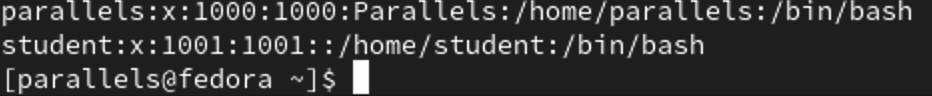
We see that sudo group users have the same permissions as root

Example – 10



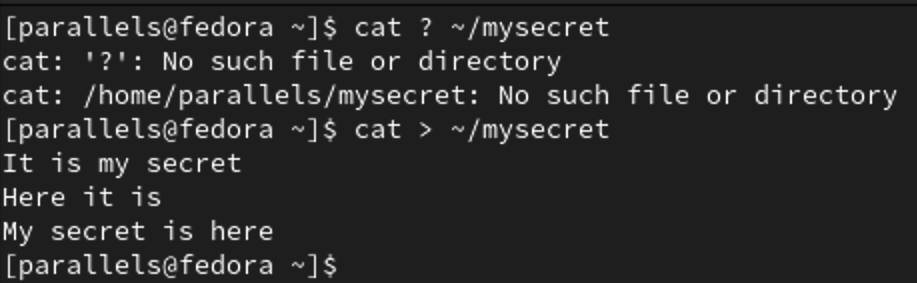
If we want to create a new user, you can use sudo adduser command

Example – 11



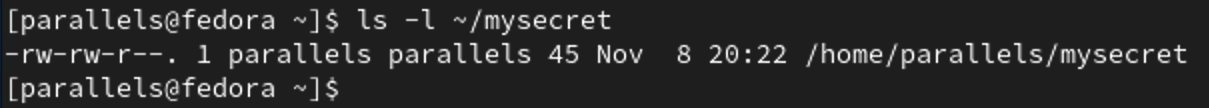
A new user student is created

Example – 12



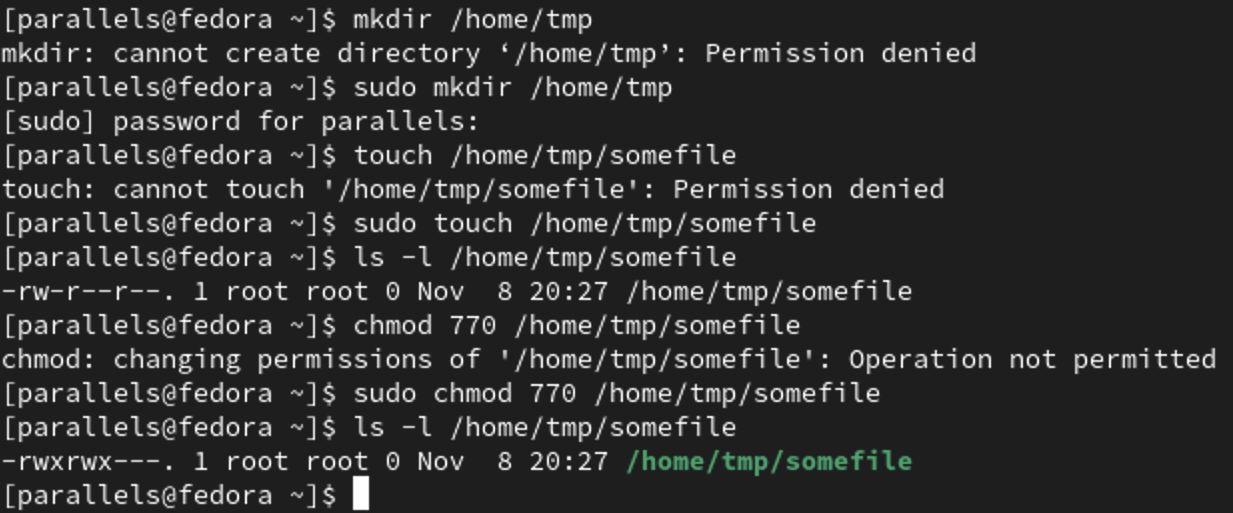
The cat command is used to read the contents of the file. But if the > sign is placed after the cat command, a new file is created.

Example – 13



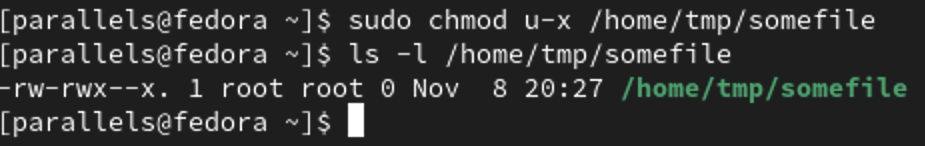
First view the permissions of your newly created file.

Example – 14



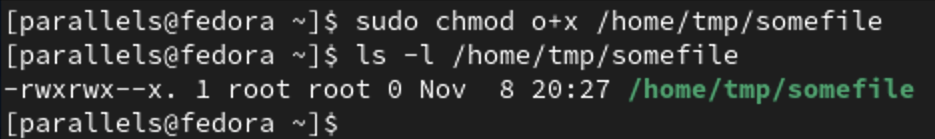
The chmod command can be used to set permissions on a file.

Example – 15



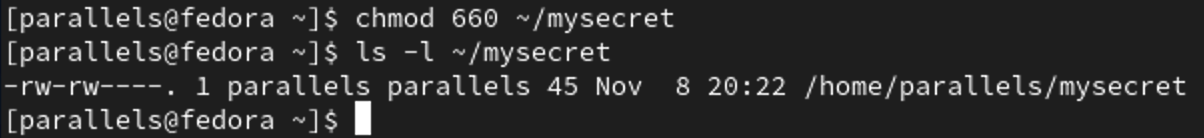
You can make relative changes: u-x would remove the owner (u) the ability to execute (x) the file.

Example – 16



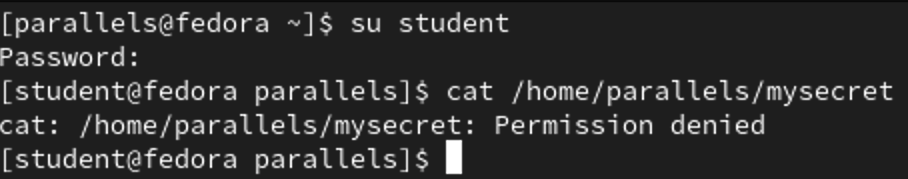
You can make relative changes: o+x would add the other user (o) the ability to execute (x) the file.

Example – 17



chmod can set permissions based on absolute octal values, or relative changes. You could use chmod to set permissions on a file based on octet: 660 would give the owner and group rw (read-write), and others no permissions.

Example – 18



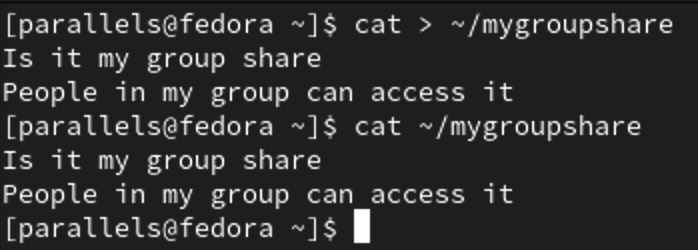
su command is used to make changes among user.

Example – 19



We gave read and write access to the file we created with chmod 666 command. Then we let other users run root directory with chmod 701 command.

Example – 20



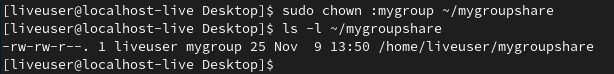
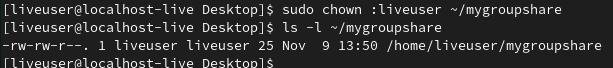
The cat command is used to read the contents of the file. But if the > sign is placed after the cat command, a new file is created.  
  
  
**Example 21**  
  
to add user seyit123 to group liveuser  
  

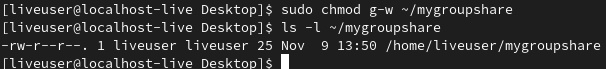

  
  
Now seyıt123 added to lıveuser group



metin içeren bir resim

Açıklama otomatik olarak oluşturuldu  
 **Example 22**  
  
Let’s create a group . name is mygroup  
  
  
  
Its here.  
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu.  
  
**Example 23**  
  
To change owner and group of a file  
  
   
  
**Example 24**Change back mygroupshare group to alex  
  
and give alex group **only read** Access to mygroupshare

  
  
**Example 24 – B:** Create a new group called “staff”, and create a file that you and a fellow classmate

(other user) can collaborate on (both edit). Test whether you have correctly set permissions. Both users should be able to edit the file, yet other users should not have write access.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

  
  
  
I didnt understand permission settings in the discription so i did both way i understood.  
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
**Example 25:**Dir ; shows us a lıst of dırectory’s file and subdırectorıes  
mkdır: creates dırectory/folder  
touch: accordıng to selected fıle we can make fıle  
  
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
**Challenge 1  
mkdir test: created test folder  
touch test/test1 test/test2 test/test3 : creates fıles ın to test folder  
From man chmod (chmod ınformatıon) we found chmod -R (-R means recursıvely)  
and changed all permıssıons of folder’s fıles**metin, levha içeren bir resim

Açıklama otomatik olarak oluşturuldu **Example 26  
  
umask let us to change default settings of fıles whıch ıs gonna be new created.**My default umask settings  
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
**Challenge 2:** Using the umask builtin command, set your umask so that new files are only rw

accessible by you (but not to your group or others):

umask XXX

where XXX is the new umask to use.

Test your new umask value by creating a new file and checking its permissions:

touch newfilename

ls -l newfilename

Do the permissions read “rw-------”? If not, change the umask and try again.

I found umask that ıs 177.   
How i found? I want only rw permissions for user.  
So its for user = r(4)+w(2)=6  
For group = 0  
for others= 0  
In conclusion its 600 , I extracted ıt from 777. 777-600=177 umask.  
  
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
**8.8 Set UID (SUID)**  
  
UID: ıt’s orıgınal user  
EUID: Prıvıleged user has hıgh permıssıons  
  
**Example 27**  
How to look at Effectıve UID ıs specıfıed:  
ls -l /usr/bın/passwd  
  
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
It has “s” so that means ıts the fıle UID wıll be used as EUID  
Also, ıt means UID and EUID are same.  
  
**Example 28**  
statistic about my fıle  
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
  
**Gıves user id**  
id -u  
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
id -u -r   
**gives EUID**  
  
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
**Example 29**  
If i want to see all the fıles runnıng wıth user ıd equal to root;  
ps -af  
  
ps (process stated)= gıves ınfo about runnıng processes  
af(fully formated lıst)=formats precesses as a full lıst  
  
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
**Example 30**  
or we can check with ; sudo find -perm -4000 -type f -prınt  
  
metin, siyah, ekran, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu  
  
sudo: super admin   
fınd: help find the files  
-perm: means permission  
-4000: gıve us files run with root permission  
- type f -print: this funtion same as -af which modifiy it as list.

9.9. Writing a SUID program in C

**Example 31**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Making file ‘mysecret’ accessible only by owner.

**Example 32**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Creating a accessmysecret.c file and compiling it



**Example 33**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Setting file permission to setuid with command chmod and checking permission of the file.

**Example 34**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Running the program

**Example 35**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Running the program under different user. The uid and gid id is different than the one before

**Challenge 3**

**Challenge 4**

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

By adding if condition to the beginning of our program we can put restrictions to our file.

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Challenge 5**

**Graphical user interface, text, application

Description automatically generated**

Same as challenge 4, we can use if-else condition to print different contents of our file.

**Text

Description automatically generated**

**Challenge 6**

****

For this challenge we use if condition again. In this example we gave user “student” with UID=1002 to access to our file.

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**10 Linus extended ACLs**

**Example 36**

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

We used setfacl command to give student user readign permission of “mysecret” file. And with getfacl on our file we can check who has which kind of permission on our file.