AACS2284 OPERATING SYSTEMS (PRACTICAL)

Practical 3: Managing Linux Permissions and Ownership

Q1. Fill in the following commands:

id	Display information about a user's UID and which groups she is assigned to	
groups	Display information on the groups in which you are a member	
/etc/passwd	User and Group Configuration Files to store information for each user such as the user name, the UID, the home directory, and the standard shell	
/etc/group	User and Group Configuration Files to store group information such as the group name, the GID, the members of the group	
/etc/shadow	User and Group Configuration Files to store encrypted user password and password expiration information	
useradd	To create user account	useradd option <username>
usermod	To modify settings for an existing user account	usermod option <username>
userdel	To delete an existing user account	userdel option <username>
passwd	To establish or change the password of a user account	passwd <username>
groupadd	To add group account for the particular user	
groupdel	To delete group account	
groupmod	To modify the setting (GID, group name, users) for an existing group	
newgrp	To change the effective group of the executing user	
gpasswd	To change password for group accounts	
chmod	To change file permission	
umask	To modify default access permission	
chown	To change the file ownership	
chgrp	To change the file ownership for group	

Q2 Perform the following operations in UNIX command line interface:

a. Add a user called “**newuser**” in the system.

```
sudo adduser newuser
```

b. By using **su** command, switch to the **newuser** account.

```
su newuser
```

c. Create a file called “**newuser_file**” with the following mode “**rw-------**”.

```
touch newuser_file
chmod go-r newuser_file
chmod g-w newuser_file
Or
chmod 600 newuser_file
```

d. Logout **newuser** account and back to your current account.

```
exit
```

e. Create a directory called “**currentuser_dir**” under **/home/administrator** with the mode **“rw-r--r--”**.

```
mkdir currentuser_dir
chmod a-x currentuser_dir
            or
chmod 644 currentuser_dir
```

f. Create a file called “**currentuser_file**” with default mode inside **currentuser_dir**. Can you create the file in the directory? If not, make necessary modification on the directory’s mode. Enter “**Hallo World.**”

```
cat > currentuser_dir/currentuser_file
Permission Denied
chmod a +x currentuser_dir
cat > currentuser_dir/currentuser_file (CTRL + D to terminate)
```

g. View **currentuser_file**using **“cat”** command. What message did you get?

```
cat currentuser_dir/currentuser_file
Halo World
```

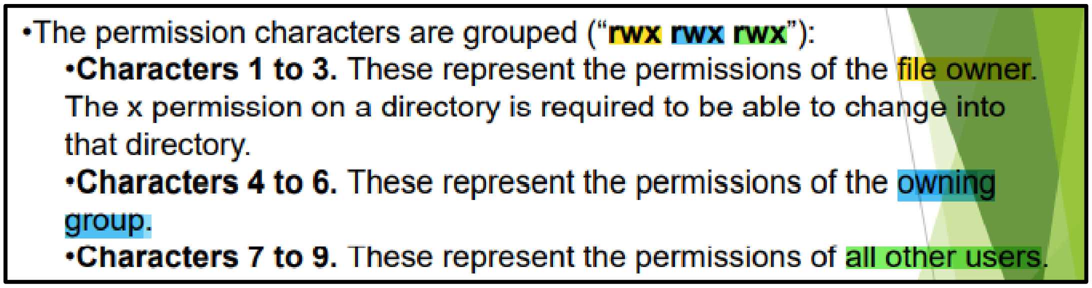
h. In the current account, view **newuser_file** using “**cat**” command. What message did you get? Why?

```
cat/home/newuser/newuser_file
Permission Denied
```

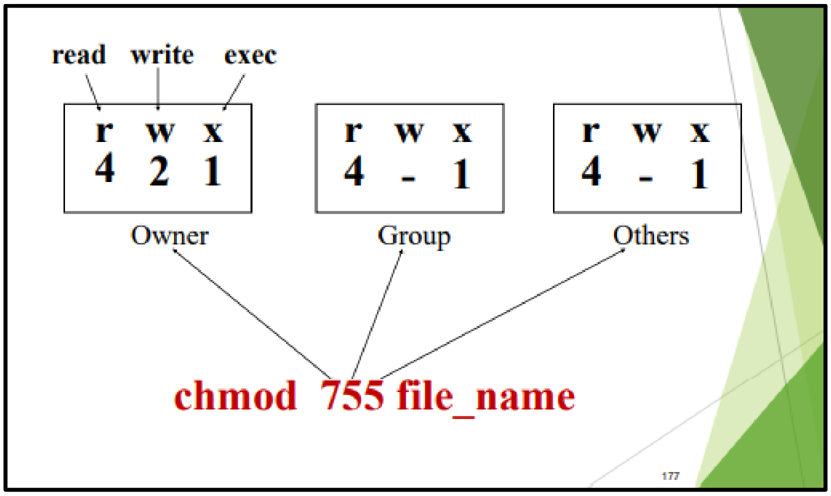
Because the file does not give permission to other users to view its content

Q3 Change the following files, which currently have **NO** permission settings, to have the specified permissions (use ls to check your result):

File	Permissions	Command
file1	rwxrwxrwx	**chmod a+rwx file OR** **chmod 777 file1**
File2	rwxr-xr-x	**chmod u+rwx,go=rx file2 OR** **chmod 755 file2**
file3	rw-r--r--	**chmod u=rw,go=r file OR** **chmod 644 file3**
file4	rwx------	**chmod u+rwx file4 OR** **chmod 700 file4**



- The permission characters are grouped ("**rwx rwx rwx**"):
  - **Characters 1 to 3.** These represent the permissions of the file owner. The x permission on a directory is required to be able to change into that directory.
  - **Characters 4 to 6.** These represent the permissions of the owning group.
  - **Characters 7 to 9.** These represent the permissions of all other users.



Q4. Perform the following operations:

Action	Command
A. Change to your home directory	**cd ~**
B. Make a directory named “myfolder”	**mkdir myfolder**
*** C. Allow group and others to be able to read and execute on your home directory (the access rights for home directory by default is already readable and executable. This question is just to show students the purpose of dot (.))	**chmod go+rx .**
*** D. Allow group and others to be able to read and execute on the myfolder directory (the access rights for myfolder by default is already readable and executable. This question is just to test students in using chmod command)	**chmod go+rx myfolder**
E. Verify the permissions on your home directory and on myfolder	**ls-l /home** **(to view home directory access rights, we need to view from home folder)** **ls -l** **(this for checking myfolder access right)**
F. Use touch to create an empty file named text1.txt and text2.txtinmyfolder directory	**touch myfolder/text1.txt** **touch myfolder/text2.txt**
G. Allow group and others to be able to write all files in the myfolder directory	**chmod go+w myfolder** **(this is to change myfolder access rights, optional and can be ignored)** **chmod go+w myfolder/*** **(this is to check access rights of all files in myfolder)**
H. Verify the permissions on the file(s) in myfolder directory	**ls -l myfolder/***

Extra exercises (Optional)

Q5.

1.	To combine **/etc/passwd** with **/etc/group** and send the output into the file **users_groups.txt** **Answer: cat/etc/passwd /etc/group > user_groups.txt**
2.	To create a symbolic link of **doc1** file to **softlink** in your home directory **Answer: ln -s doc1 softlink**
3.	Display the content of **/var/log/messages** page-by-page. **Answer: less / var/log/messages**
4.	Move and rename the **/tmp/file2** file to **~/my_file2** by using a single command. **Answer: mv/tmp/file2 ~/my_file2**
5.	Using wildcards token, list all the files with filenames consists of exactly 4 letters and which start with the letter “M”. **Answer: ls -M???**
6.	Find all files in the /home directory that have the word “AACS2284” as part of their filename. **Answer: find /home -name “AACS2284”**
7.	List all the directory names that exist under the /var directory **Answer: ls -d/var/***
8.	The vi editor can function in 2 modes, namely :____input________ and ___command___
9.	Write a command to show the date and time using each of the following formats: hh: 11mm: 24 **Answer: date + “hh: %H mm: %M”**
10.	For the **umask 272**, what will the permissions on all new files and new directories be? Express your answers in octal format. **Anwser:** New files ____404____ and new directories____505____ *By default, files ***don’t have execute*** so need to minus 1 again
11.	Lock the account of a user with the username **john.** **Answer: passwd -l john**

Q6. The questions below are interrelated and therefore must be done in sequence.

1.	Create a non-empty file called **Myfile** file under your default user directory. Create a new user called **John**	**cat > Myfile** **sudo useradd -m John**
2.	Switch user to the **root user** account. Change the ownership of **Myfile** file to **John**.	**sudo -i** **chown John /home/taruc/Myfile**
3.	Logout **root** user account.	**exit**
4.	Using a single command, switch to "**John**" account and his home directory. Create two directories and subdirectories called **dirA/letters** and **dirB/reports** under **John** directory by using a single command.	**su -l John** **mkdir -p dir A/letters dirB/reports**
5.	Deny all access to the "**letters**" directory by everyone except the owner.	**chmod 700 dirA/letters**
6.	Change directory to **dirB/reports** using absolute pathname. Create an empty file named **OSdoc** under **dirB/reports.** Set the permission for the file **OSdoc** to allow read and write by the file's owner and members of the file's owner group only.	**cd/home/John/dirB/reports** **touch OSdoc** **chmod 660 OSdoc**
7.	Logout **John** account and switch to your own account.	**exit**
8.	Change directory to John's **dirA/letters** using relative pathname. Can this be done? If not, why not?	**cd ../John/dirA/letters** **No write and execute permission to the directory**