Lab (2)

1. Create a user account with the following attribute
   Username: ahmed
   Fullname/comment: ahmed ali
   Password: ahmed

```
[amr@localhost ~]$ sudo useradd ahmed  -c "ahmed ali" -md /home/ahmed -s /bin/bash
[sudo] password for amr:
[amr@localhost ~]$ tail -n5 /etc/passwd
gnome-initial-setup:x:976:975::/run/gnome-initial-setup/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
amr:x:1000:1000:amr:/home/amr:/bin/bash
ahmed:x:1001:1001:ahmed ali:/home/ahmed:/bin/bash
[amr@localhost ~]$ sudo passwd ahmed
Changing password for user ahmed.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[amr@localhost ~]$
```

2. Create a user account with the following attribute
   Username: baduser
   Full name/comment: Bad User
   Password: baduser

```
[amr@localhost ~]$ sudo useradd baduser  -c "Bad User" -s /bin/bash
[amr@localhost ~]$ tail -n5 /etc/passwd
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
amr:x:1000:1000:amr:/home/amr:/bin/bash
ahmed:x:1001:1001:ahmed ali:/home/ahmed:/bin/bash
baduser:x:1002:1002:Bad User:/home/baduser:/bin/bash
[amr@localhost ~]$ sudo passwd baduser
Changing password for user baduser.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[amr@localhost ~]$
```

3. Create a supplementary (Secondary) group called pgroup with group ID of 30000

```
[amr@localhost ~]$ sudo groupadd -g 30000 pgroup
[amr@localhost ~]$ tail -n5 /etc/group
tcpdump:x:72:
amr:x:1000:
ahmed:x:1001:
baduser:x:1002:
pgroup:x:30000:
[amr@localhost ~]$
```

4. Create a supplementary group called badgroup

```
[amr@localhost ~]$ sudo groupadd badgroup
[amr@localhost ~]$ tail -n5 /etc/group
amr:x:1000:
ahmed:x:1001:
baduser:x:1002:
pgroup:x:30000:
badgroup:x:30001:
```

5. Add ahmed user to the pgroup group as a supplementary group

```
[amr@localhost ~]$ sudo usermod ahmed -aG pgroup
[amr@localhost ~]$ groups ahmed
ahmed : ahmed pgroup
[amr@localhost ~]$
```

6. Modify the password of ahmed's account to password

```
[amr@localhost ~]$ sudo passwd ahmed
Changing password for user ahmed.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
[amr@localhost ~]$
```

7. Modify ahmed's account so the password expires after 30 days

```
[amr@localhost ~]$ sudo chage -M 30 ahmed
[amr@localhost ~]$ chage -l ahmed
chage: Permission denied.
[amr@localhost ~]$ sudo chage -l ahmed
Last password change                                    : Apr 12, 2023
Password expires                                        : May 12, 2023
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 0
Maximum number of days between password change          : 30
Number of days of warning before password expires       : 7
[amr@localhost ~]$
```

8. Lock bad user account so he can't log in

```
[amr@localhost ~]$ sudo usermod -L baduser
```

9. Delete bad user account

```
[amr@localhost ~]$ sudo userdel baduser
[amr@localhost ~]$ tail -n5 /etc/passwd
gnome-initial-setup:x:976:975::/run/gnome-initial-setup/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
amr:x:1000:1000:amr:/home/amr:/bin/bash
ahmed:x:1001:1001:ahmed ali:/home/ahmed:/bin/bash
[amr@localhost ~]$
```

10. Delete the supplementary group called badgroup.

```
[amr@localhost ~]$ sudo groupdel badgroup
[amr@localhost ~]$ tail -n5 /etc/group
slocate:x:21:
tcpdump:x:72:
amr:x:1000:
ahmed:x:1001:
pgroup:x:30000:ahmed
[amr@localhost ~]$
```

13. Create a folder called myteam in your home directory and change its permissions to read only for the owner.

```
[amr@localhost ~]$ mkdir myteam
[amr@localhost ~]$ chmod 400 myteam/
[amr@localhost ~]$ ls -ld myteam/
dr--------. 2 amr amr 6 Apr 12 00:23 myteam/
[amr@localhost ~]$
```

14. Log out and log in by another user

```
[amr@localhost ~]$ whoami
amr
[amr@localhost ~]$ su - ahmed
Password:
[ahmed@localhost ~]$ whoami
ahmed
```

15. Try to access (by cd command) the folder (myteam)

```
[ahmed@localhost ~]$ cd /home/amr/myteam
bash: cd: /home/amr/myteam: Permission denied
[ahmed@localhost ~]$ su amr
Password:
[amr@localhost ahmed]$ cd myteam
bash: cd: myteam: Permission denied
[amr@localhost ahmed]$
```

16. Using the command Line

* Change the permissions of oldpasswd file to give owner read and write permissions and for group write and execute and execute only for the others (using chmod in 2 different ways)

chmod 631 oldpasswd

chmod u=rw,g=wx,o=x oldpasswd

ls -l oldpasswd

```
[amr@localhost ~]$ ls -l oldpasswd
-rwxrwxrwx. 1 amr amr 2504 Apr  7 06:30 oldpasswd
[amr@localhost ~]$ chmod 631 oldpasswd
[amr@localhost ~]$ ls -l oldpasswd
-rw--wx--x. 1 amr amr 2504 Apr  7 06:30 oldpasswd
[amr@localhost ~]$ chmod 777 oldpasswd
[amr@localhost ~]$ ls -l oldpasswd
-rwxrwxrwx. 1 amr amr 2504 Apr  7 06:30 oldpasswd
[amr@localhost ~]$ chmod u=rw,g=wx,o=x oldpasswd
[amr@localhost ~]$ ls -l oldpasswd
-rw--wx--x. 1 amr amr 2504 Apr  7 06:30 oldpasswd
[amr@localhost ~]$
```

* Change your default permissions to be as above.

When we make a new directory, the permissions will be calculated as

(full permissions for directory) – (umask value) i.e. 777 – 631 = 146

```
[amr@localhost ~]$ umask 0146
[amr@localhost ~]$ umask
0146
[amr@localhost ~]$
```

* What is the maximum permission a file can have, by default when it is just created? And what is that for directory.

File -> The full permission set for a file is 666 (read/write permission for all)

Directory -> The full permission set for a directory is 777 (read/write/execute)

* Change your default permissions to be no permission to everyone then create a directory and a file

```
[amr@localhost ~]$ umask 0777
[amr@localhost ~]$ mkdir newdir
[amr@localhost ~]$ touch newfile
[amr@localhost ~]$ ls -l newfile & ls -ld newdir
[1] 7377
d---------. 2 amr amr 6 Apr 12 01:29 newdir
----------. 1 amr amr 0 Apr 12 01:29 newfile
[amr@localhost ~]$
```

17. What are the minimum permission needed for:
* Copy a directory (permission for source directory and permissions for target parent directory)
For source directory: execute and read permission
For target parent directory: execute and write permission.

* Copy a file (permission for source file and permission for target parent File)
For source file: read permission.
For target file directory: you don't need any permission since it doesn't exist before you copy it. Or write permission if the file exists.

* Delete a file
No permission needed on the file, but write and execute permission on the parent directory.

* Change to a directory
Execute permission.

* List a directory content (ls command)
Read permission

* View a file content (more/cat command)
Read permission.

* Modify a file content
Write permission.

18. Create a file with permission 444. Try to edit in it and to remove it? Note what happened.

As per below screenshots when I try to edit in the file it shows me that this is a read only file and if I want to override it, and when I try to remove the file it asks me if I want to delete a write protected file or not and if I pressed yes it will delete it.

```
[amr@localhost ~]$ touch file
[amr@localhost ~]$ chmod 444 file
[amr@localhost ~]$ ls -l file
-r--r--r--. 1 amr amr 0 Apr 12 01:40 file
[amr@localhost ~]$ vim file
[amr@localhost ~]$
```

```
~
-- INSERT -- W10: Warning: Changing a readonly file
```

```
[amr@localhost ~]$ rm file
rm: remove write-protected regular empty file 'file'? y
[amr@localhost ~]$
```

19. What is the difference between the “x” permission for a file and for a directory?

For a directory which enables the user to get into the directory or search a directory.

For a file the execute permission lets you execute an executable file.