Linux Operating System: Assignment 1

Q1. Create the following users, groups, and group memberships:

Q: A group named admin
Ans: Below a group is created of name “admin”.
We check whether group is created using command: tail –4 /etc/group

```
(base) taseng@krypton:~$ sudo groupadd admin
(base) taseng@krypton:~$ tail -4 /etc/group
mlocate:x:136:
nvidia-persistenced:x:134:
mysql:x:137:
admin:x:1001:
(base) taseng@krypton:~$
```

Q: A user harry who belongs to admin as a secondary group
Ans: Below a user “harry” is created which belongs to group “admin” as a secondary member.
We check whether “harry” is under group “admin” using command: groups harry

```
(base) taseng@krypton:~$ sudo useradd -G admin harry
(base) taseng@krypton:~$ groups harry
harry : harry admin
```

Q: A user natasha who also belongs to admin as a secondary group
Ans: Below another user “natasha” is created which also belongs to the group “admin” as a secondary member.
We check whether “natasha” is under group “admin” using command: groups natasha

```
(base) taseng@krypton:~$ sudo useradd -G admin natasha
(base) taseng@krypton:~$ groups natasha
natasha : natasha admin
```

Q: A user “sarah” who does not have access to an interactive shell on the system, and who is not a member of admin
Ans: Below a user “sarah” is created with no access to interactive shell by making its shell name as “nologin”

```
(base) taseng@krypton:~$ sudo useradd -s /bin/nologin sarah
useradd: Warning: missing or non-executable shell '/bin/nologin'
(base) taseng@krypton:~$ tail -4 /etc/passwd
mysql:x:129:137:MySQL Server,,,:/nonexistent:/bin/false
harry:x:1001:1002::/home/harry:/bin/sh
natasha:x:1002:1003::/home/natasha:/bin/sh
sarah:x:1003:1004::/home/sarah:/bin/nologin
```

Sarah is not a member of admin is shown below:

```
(base) taseng@krypton:~$ groups sarah
sarah : sarah
```

Q: harry, natasha, and sarah should all have the password as "password@123"
Ans: Passwords for “harry”, “natasha” and “sarah” is set as “password@123” using command: sudo passwd username

```
(base) taseng@krypton:~$ sudo passwd harry
New password:
Retype new password:
passwd: password updated successfully
(base) taseng@krypton:~$ sudo passwd natasha
New password:
Retype new password:
passwd: password updated successfully
(base) taseng@krypton:~$ sudo passwd sarah
New password:
Retype new password:
passwd: password updated successfully
```

Q: user mac with user id 3553
Ans: Below user “mac” is created with user if “3553”

We check the id of mac using command: id mac

Also we verify this in the file /etc/passwd

```
(base) taseng@krypton:~$ sudo useradd -u 3553 mac
(base) taseng@krypton:~$ id mac
uid=3553(mac) gid=3553(mac) groups=3553(mac)
(base) taseng@krypton:~$ tail -4 /etc/passwd
harry:x:1001:1002::/home/harry:/bin/sh
natasha:x:1002:1003::/home/natasha:/bin/sh
sarah:x:1003:1004::/home/sarah:/bin/nologin
mac:x:3553:3553::/home/mac:/bin/sh
```

NOTE: Make sure that user harry, mac and natasha must have home dirs and shell /bin/bash

Ans: shell of users “harry”, “mac” and “natasha” is modified to “/bin/bash” as given below

```
(base) taseng@krypton:~$ sudo usermod -s /bin/bash harry
(base) taseng@krypton:~$ sudo usermod -s /bin/bash natasha
(base) taseng@krypton:~$ sudo usermod -s /bin/bash mac
(base) taseng@krypton:~$ tail -7 /etc/passwd
snap_daemon:x:584788:584788::/nonexistent:/usr/bin/false
nvidia-persistenced:x:127:134:NVIDIA Persistence Daemon,,,:/nonexistent:
/usr/sbin/nologin
mysql:x:129:137:MySQL Server,,,:/nonexistent:/bin/false
harry:x:1001:1002::/home/harry:/bin/bash
natasha:x:1002:1003::/home/natasha:/bin/bash
sarah:x:1003:1004::/home/sarah:/bin/nologin
mac:x:3553:3553::/home/mac:/bin/bash
```

Q2. Create a collaborative directory /common/adm with the following characteristics:

- The directory should be readable, writable, and accessible to members of owning group but not to any other user. (It is understood that root has access to all files and directories on the system.)

Ans: Directories “common/adm” is created using parent to child method using command: sudo mkdir -p common/adm

```
(base) taseng@krypton:/$ sudo mkdir -p common/adm
(base) taseng@krypton:/$ ls /common/
adm
(base) taseng@krypton:/$ ls -l common
total 4
drwxr-xr-x 2 root root 4096 Sep  6 22:59 adm
```

Below we change the permission of the directory “common” recursively and make it readable, writable and executable for other group members, but not to others.

```
(base) taseng@krypton:/$ sudo chmod g+w,o-rx common -R
(base) taseng@krypton:/$ sudo ls -l common
total 4
drwxrwx--- 2 root root 4096 Sep  6 22:59 adm
(base) taseng@krypton:/$
```

Q3. You need to change the permissions of a file named schedule.txt such that the file owner can edit the file, users who are members of the group that owns the file can edit it, and users who are not owners and do not belong to the owning group can view it but not modify it. Which command will do this?

Ans: A file “schedule.txt” is created using “touch” command.

Permissions for the file is changed using “chmod” command. File owner is given read and write permissions, members of its group are given read and write permissions and others are given only read permission.

```
(base) taseng@krypton:/$ sudo touch schedule.txt
(base) taseng@krypton:/$ ls -l schedule.txt
-rw-r--r-- 1 root root 0 Sep  6 23:21 schedule.txt
(base) taseng@krypton:/$ sudo chmod u=rw,g=rw,o=r schedule.txt
(base) taseng@krypton:/$ ls -l schedule.txt
-rw-rw-r-- 1 root root 0 Sep  6 23:21 schedule.txt
(base) taseng@krypton:/$
```

Q4. Create a dir and creates 10000 files under it, assign permissions to all files as user owner can read, edit and execute, group owner can read & execute, and read permission to others.

Ans: Below a directory “cdac” is created with 10000 files inside it using the following command:

```
(base) taseng@krypton:~$ mkdir cdac && touch cdac/file{1..10000}
ls -l cdac/
```

```
-rw-rw-r-- 1 taseng taseng 0 Sep  6 23:36 file9991
-rw-rw-r-- 1 taseng taseng 0 Sep  6 23:36 file9992
-rw-rw-r-- 1 taseng taseng 0 Sep  6 23:36 file9993
-rw-rw-r-- 1 taseng taseng 0 Sep  6 23:36 file9994
-rw-rw-r-- 1 taseng taseng 0 Sep  6 23:36 file9995
-rw-rw-r-- 1 taseng taseng 0 Sep  6 23:36 file9996
-rw-rw-r-- 1 taseng taseng 0 Sep  6 23:36 file9997
-rw-rw-r-- 1 taseng taseng 0 Sep  6 23:36 file9998
-rw-rw-r-- 1 taseng taseng 0 Sep  6 23:36 file9999
```

Permissions to all files inside are given where user owner can read, write and execute the file, group owner and members of the group can read and execute and others can only read.

```
(base) taseng@krypton:~$ chmod u=rwx,g=rx,o=r cdac -R
```

```
-rwxr-xr-- 1 taseng taseng 0 Sep  6 23:36 file9994
-rwxr-xr-- 1 taseng taseng 0 Sep  6 23:36 file9995
-rwxr-xr-- 1 taseng taseng 0 Sep  6 23:36 file9996
-rwxr-xr-- 1 taseng taseng 0 Sep  6 23:36 file9997
-rwxr-xr-- 1 taseng taseng 0 Sep  6 23:36 file9998
-rwxr-xr-- 1 taseng taseng 0 Sep  6 23:36 file9999
```