Task2

1) Analyze the structure of the /etc/passwd and /etc/group file, what fields are present in it, what users exist on the system? Specify several pseudo-users, how to define them?

/etc/passwd

1. Username: It is used when user logs in. It should be between 1 and 32 characters in length.

2. Password: An x character indicates that encrypted password is stored in /etc/shadow file. Please note that you need to use the passwd command to computes the hash of a password typed at the CLI or to store/update the hash of the password in /etc/shadow file.

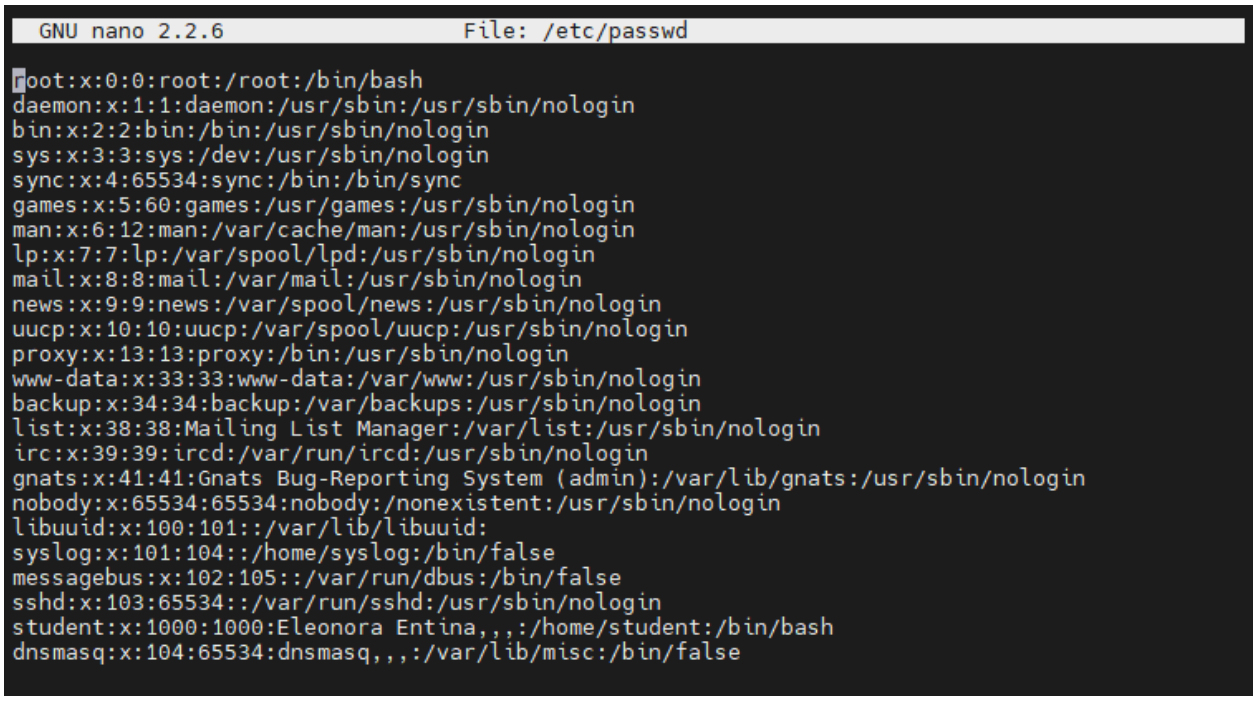
3. User ID (UID): Each user must be assigned a user ID (UID). UID 0 (zero) is reserved for root and UIDs 1-99 are reserved for other predefined accounts. Further UID 100-999 are reserved by system for administrative and system accounts/groups.

4. Group ID (GID): The primary group ID (stored in /etc/group file)

5. User ID Info (GECOS): The comment field. It allow you to add extra information about the users such as user’s full name, phone number etc. This field use by finger command.

6. Home directory: The absolute path to the directory the user will be in when they log in. If this directory does not exists then users directory becomes /

7. Command/shell: The absolute path of a command or shell (/bin/bash). Typically, this is a shell. Please note that it does not have to be a shell. For example, sysadmin can use the nologin shell, which acts as a replacement shell for the user accounts. If shell set to `/sbin/nologin` and the user tries to log in to the Linux system directly, the /sbin/nologin shell closes the connection.



/etc/group

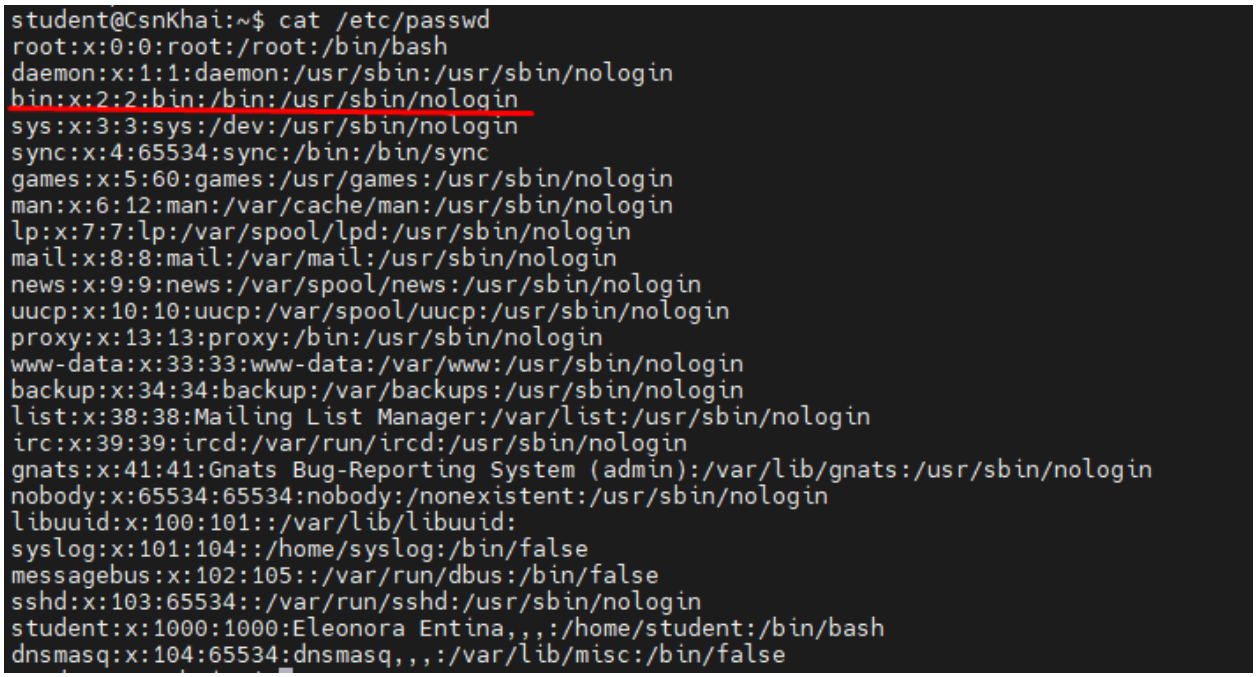
1. group_name: It is the name of group. If you run ls -l command, you will see this name printed in the group field.

2. Password: Generally password is not used, hence it is empty/blank. It can store encrypted password. This is useful to implement privileged groups.

3. Group ID (GID): Each user must be assigned a group ID. You can see this number in your /etc/passwd file.

4. Group List: It is a list of user names of users who are members of the group. The user names, must be separated by commas.

By definition, root always has UID 0, daemon has UID 1, and pseudo-user bin has UID 2



2) What are the uid ranges? What is UID? How to define it?

A UID (user identifier) is a number assigned by Linux to each user on the system. This number is used to identify the user to the system and to determine which system resources the user can access. UIDs are stored in the /etc/passwd file.

The third field represents the UID. The root user has the UID of 0. Most Linux distributions reserve the first 100 UIDs for system use. New users are assigned UIDs starting from 500 or 1000. For example, new users in Ubuntu start from 1000.

3) What is GID? How to define it?

Groups in Linux are defined by GIDs **(**group IDs**).** Just like with UIDs, the first 100 GIDs are usually reserved for system use. The GID of **0** corresponds to the root group and the GID of 100 usually represents the **users** group. GIDs are stored in the /etc/groups file

```
student@CsnKhai:~$ cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,student
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
cdrom:x:24:student
floppy:x:25:
tape:x:26:
sudo:x:27:student
audio:x:29:
dip:x:30:student
www-data:x:33:
backup:x:34:
operator:x:37:
list:x:38:
irc:x:39:
src:x:40:
gnats:x:41:
shadow:x:42:
utmp:x:43:
video:x:44:
```

The third field represents the GID. New groups are usually assigned GIDs starting from 1000.

4) How to determine belonging of user to the specific group?

```
student@CsnKhai:~$ groups student
student : student adm cdrom sudo dip plugdev lpadmin sambashare
student@CsnKhai:~$ id -Gn "student"|grep -c "adm"
1
```

5) What are the commands for adding a user to the system? What are the basic parameters required to create a user?

```
student@CsnKhai:~$ sudo useradd test
[sudo] password for student:
```

```
student@CsnKhai:~$ sudo adduser teest
Adding user `teest' ...
Adding new group `teest' (1002) ...
Adding new user `teest' (1002) with group `teest' ...
Creating home directory `/home/teest' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
Sorry, passwords do not match
passwd: Authentication token manipulation error
passwd: password unchanged
Try again? [y/N] y
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for teest
Enter the new value, or press ENTER for the default
        Full Name []: test
        Room Number []: 3
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
```

```
# system.
# Similar to DHSELL in adduser. However, we use "sh" here because
# useradd is a low level utility and should be as general
# as possible
SHELL=/bin/sh
#
# The default group for users
# 100=users on Debian systems
# Same as USERS_GID in adduser
# This argument is used when the -n flag is specified.
# The default behavior (when -n and -g are not specified) is to create a
# primary user group with the same name as the user being added to the
# system.
# GROUP=100
#
# The default home directory. Same as DHOME for adduser
# HOME=/home
#
# The number of days after a password expires until the account
# is permanently disabled
# INACTIVE=-1
#
# The default expire date
# EXPIRE=
#
# The SKEL variable specifies the directory containing "skeletal" user
# files; in other words, files such as a sample .profile that will be
# copied to the new user's home directory when it is created.
# SKEL=/etc/skel
#
# Defines whether the mail spool should be created while
# creating the account
# CREATE_MAIL_SPOOL=yes
```

6) How do I change the name (account name) of an existing user?

```
student@CsnKhai:~$ sudo usermod -l new_test test
```

7) What is skell_dir? What is its structure?

The */etc/skel* directory contains files and directories that are automatically copied over to a new user's when it is created from useradd command.

```
student@CsnKhai:~$ ls -la /etc/skel/
total 20
drwxr-xr-x  2 root root 4096 Sep 15  2015 .
drwxr-xr-x 84 root root 4096 Feb 18 17:59 ..
-rw-r--r--  1 root root  220 Apr  9  2014 .bash_logout
-rw-r--r--  1 root root 3637 Apr  9  2014 .bashrc
-rw-r--r--  1 root root  675 Apr  9  2014 .profile
```

8) How to remove a user from the system (including his mailbox)?

```
student@CsnKhai:~$ sudo killall -u new_test
```

9) What commands and keys should be used to lock and unlock a user account?

```
student@CsnKhai:~$ sudo passwd -l new_test
passwd: password expiry information changed.
```

```
student@CsnKhai:~$ sudo passwd -u new_test
```

10) How to remove a user's password and provide him with a password-free login for subsequent password change?

```
student@CsnKhai:~$ sudo passwd --delete teest
passwd: password expiry information changed.
student@CsnKhai:~$ usermod -s /usr/sbin/nologin teest
usermod: Permission denied.
usermod: cannot lock /etc/passwd; try again later.
student@CsnKhai:~$ sudo usermod -s /usr/sbin/nologin teest
student@CsnKhai:~$ sudo chage -l teest
Last password change                                    : Feb 18, 2022
Password expires                                        : never
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 0
Maximum number of days between password change          : 99999
Number of days of warning before password expires       : 7
```

11) Display the extended format of information about the directory, tell about the information columns displayed on the terminal.

```
student@CsnKhai:~$ ls -ld /etc
drwxr-xr-x 84 root root 4096 Feb 18 18:15 /etc
```

12) What access rights exist and for whom (i. e., describe the main roles)? Briefly describe the acronym for access rights.

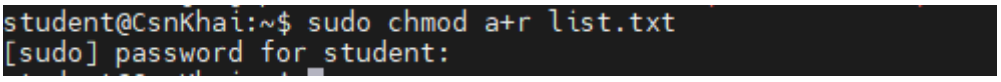
On a Linux system, each file and directory is assigned access rights for the owner of the file, the members of a group of related users, and everybody else. Rights can be assigned to read a file, to write a file, and to execute a file.

To see the permission settings for a file, we can use the ls command.

13) What is the sequence of defining the relationship between the file and the user?

By default, the *owner* of a file is the user who *created* it and the *group* assigned to a file is the *primary group* of the user.

14) What commands are used to change the owner of a file (directory), as well as the mode of access to the file? Give examples, demonstrate on the terminal.



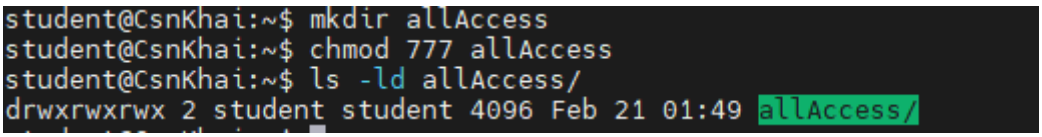
15) What is an example of octal representation of access rights? Describe the umask command.

Umask, or the user file-creation mode, is a Linux command that is used to assign the default file

permission sets for newly created folders and files. The term mask references the grouping of the permission bits, each of which defines how its corresponding permission is set for newly created files.

16) Give definitions of sticky bits and mechanism of identifier substitution. Give an example of files and directories with these attributes.

A Sticky bit is a permission bit that is set on a file or a directory that lets only the owner of the file/directory or the root user to delete or rename the file. No other user is given privileges to delete the file created by some other user.



17) What file attributes should be present in the command script

The files and directories can have following attributes:

- **a** - append only
- **c** - compressed
- **d** - no dump
- **e** - extent format
- **i** - immutable
- **j** - data journaling
- **s** - secure deletion
- **t** - no tail-merging

- **u** - undeletable
- **A** - no atime updates
- **D** - synchronous directory updates
- **S** - synchronous updates
- **T** - top of directory hierarchy