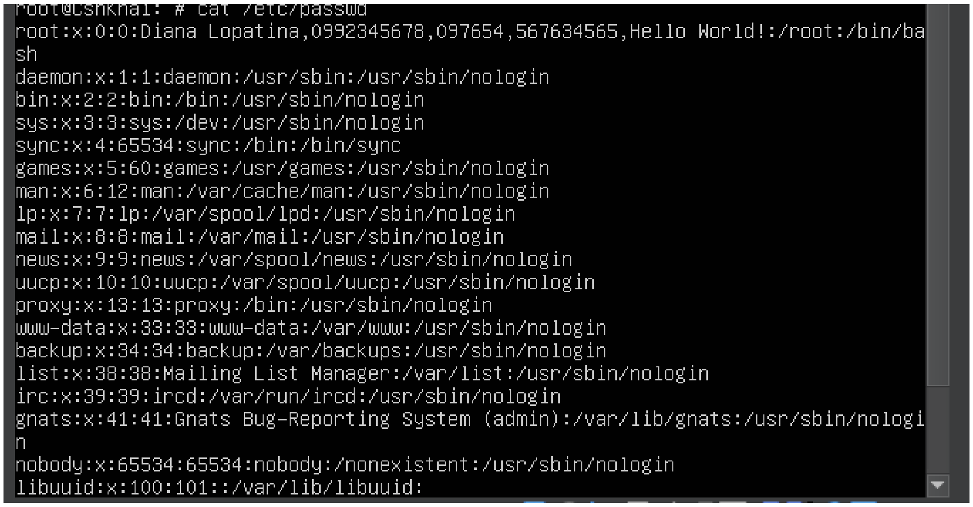
Task2

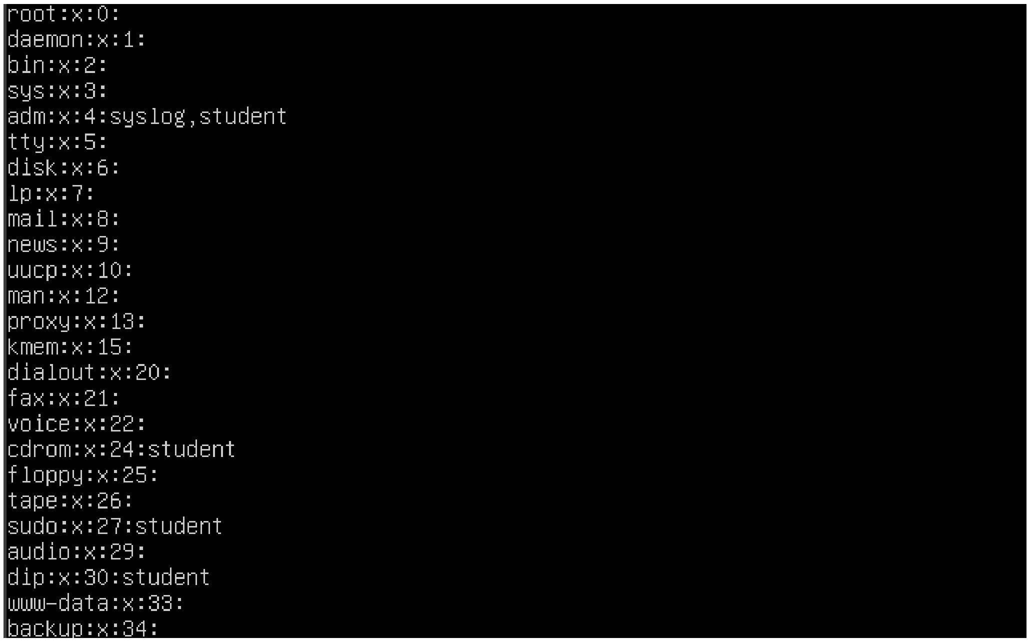
1) Analyze the structure of the /etc/passwd and /etc/group file, what fields are present in it, what users exist on the system? Specify several pseudo-users, how to define them?



```
root@Cshkhal: # cat /etc/passwd
root:x:0:0:Diana Lopatina,0992345678,097654,567634565,Hello World!:/root:/bin/ba
sh
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologi
n
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
libuuid:x:100:101::/var/lib/libuuid:
```

/etc/passwd – contains information about all user accounts found in the system. We can find a list of system accounts, saving useful information from each account such as user id, group is, home directory, shell, etc. The file contains lines of the following form, separated by colons: username: pswd: uid: gid: uid comments: directory: shell

/etc/group – file applies to the general security sheme for Unix-like system: user, group, and file access. The format for this file is a follow: group_name:password

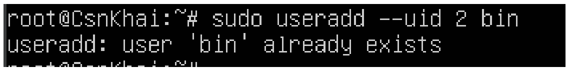


```
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,student
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
cdrom:x:24:student
floppy:x:25:
tape:x:26:
sudo:x:27:student
audio:x:29:
dip:x:30:student
www-data:x:33:
backup:x:34:
```

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

The system UIDs from 0 to 99 should be statically allocated by the system, and shall not be created by applications. UID – unique identifier of the user within the system. UID values typically range from 0 to a certain max value. The specific ranges can vary depending on the distribution and system configuration, but the most common ranges are System User (from 0 to 999 are often reserved for system accounts and administrative users) and Regular User (starting from 1000 and above are usually reserved for regular user accounts). This is usually a positive number not more than 65535 (sometimes 32-bit). Some identifiers are reserved for special use. These include 0 (root), 1-999(daemons, pseudo-users, system and reserved users), 1000+ (regular users).

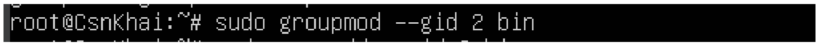
useradd [-c uid comment] [-d dir] [-e expire] [-f inactive] [-g gid] [-m [-k skel_dir]] [-s shell] [-u uid [-o]] username



```
root@CsnKhai:~# sudo useradd --uid 2 bin
useradd: user 'bin' already exists
```

3) What is GID? How to define it?

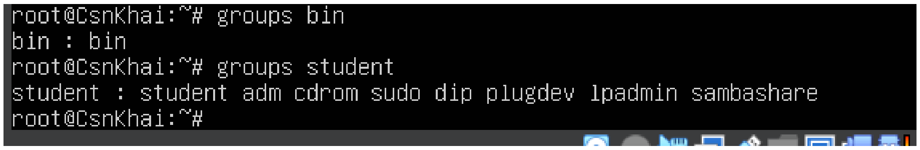
GID – unique identifier of the group within the system to which the user belongs.



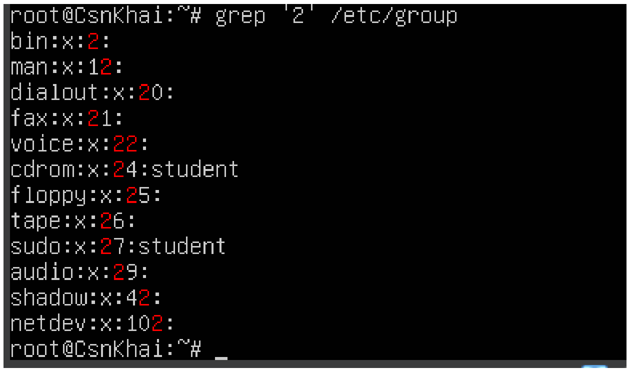
```
root@CsnKhai:~# sudo groupmod --gid 2 bin
```

GID – defines the ID or name of the group to which the user belongs.

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



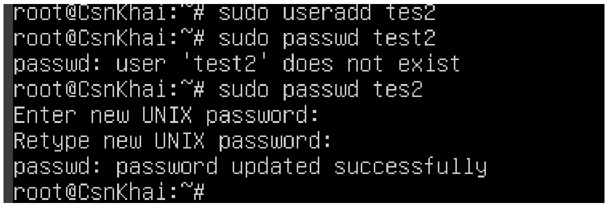
```
root@CsnKhai:~# groups bin
bin : bin
root@CsnKhai:~# groups student
student : student adm cdrom sudo dip plugdev lpadmin sambashare
root@CsnKhai:~#
```



```
root@CsnKhai:~# grep '2' /etc/group
bin:x:2:
man:x:12:
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:
shadow:x:42:
netdev:x:102:
root@CsnKhai:~# _
```

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

parameters required to create a user?

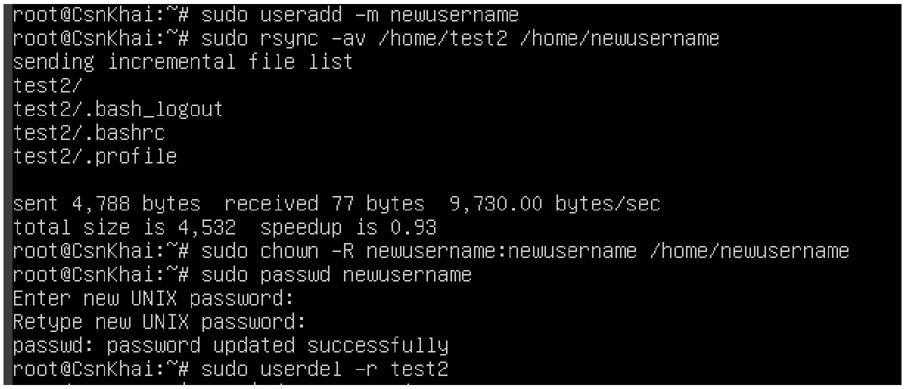


```
root@CsnKhai:~# sudo useradd tes2
root@CsnKhai:~# sudo passwd test2
passwd: user 'test2' does not exist
root@CsnKhai:~# sudo passwd tes2
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@CsnKhai:~#
```

Basic parameters required to create:

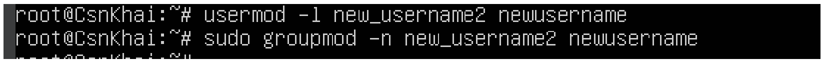
Username (username), Home Directory (-m or --create-home), Default Shell (-s or --shell), User ID (-u or --uid), Primary Group (-g or --gid), Password (passwd)

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



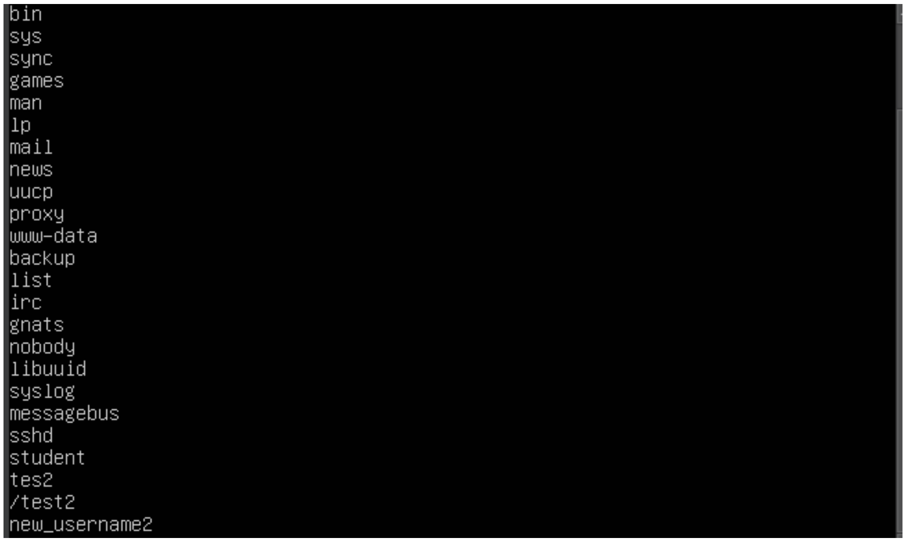
```
root@CsnKhai:~# sudo useradd -m newusername
root@CsnKhai:~# sudo rsync -av /home/test2 /home/newusername
sending incremental file list
test2/
test2/.bash_logout
test2/.bashrc
test2/.profile

sent 4,788 bytes  received 77 bytes  9,730.00 bytes/sec
total size is 4,532  speedup is 0.93
root@CsnKhai:~# sudo chown -R newusername:newusername /home/newusername
root@CsnKhai:~# sudo passwd newusername
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@CsnKhai:~# sudo userdel -r test2
```



```
root@CsnKhai:~# usermod -l new_username2 newusername
root@CsnKhai:~# sudo groupmod -n new_username2 newusername
```

Result:



```
bin
sys
sync
games
man
lp
mail
news
uucp
proxy
www-data
backup
list
irc
gnats
nobody
libuuid
syslog
messagebus
sshd
student
tes2
/test2
new_username2
```

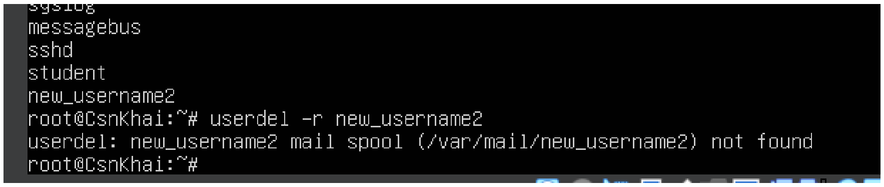
7) What is skel_dir? What is its structure?

Skel_dir – contains files that must be copied to the new user’s home directory. When a new user account is created, the contents of the skel directory are typically copied into the user's home directory to provide a basic set of files, settings, and configurations.

Basic structure:

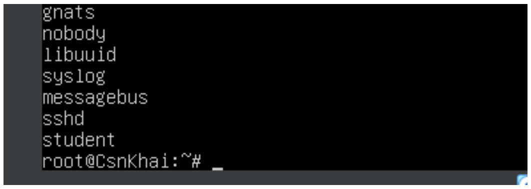
```
skel_dir/
├── .bashrc
├── .bash_logout
├── .profile
├── .config/
|   └── user-dirs.dirs
├── .local/
|   └── share/
|       ├── applications/
|       |   └── mimeapps.list
|       └── icons/
|           └── default/
|               └── index.theme
└── Documents/
    ├── README.txt
    └── Examples/
```

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



```
syslog
messagebus
sshd
student
new_username2
root@CsnKhai:~# userdel -r new_username2
userdel: new_username2 mail spool (/var/mail/new_username2) not found
root@CsnKhai:~#
```

Result:



```
gnats
nobody
libuuid
syslog
messagebus
sshd
student
root@CsnKhai:~# _
```

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

Lock a user account command with **-l options.**

(passwd -l username)

Unlock a user account command with **-u options**

(passwd -u username_)

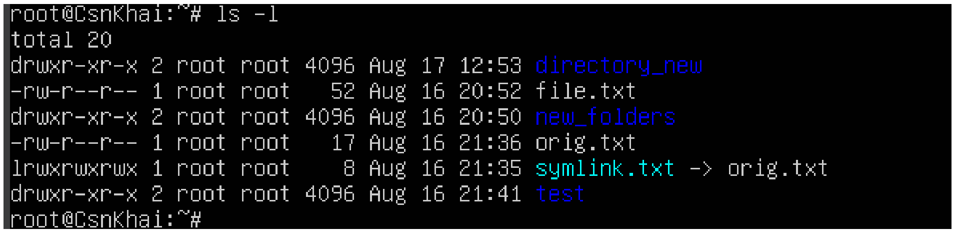
10) How to remove a user's password and provide him with a password-free

login for a subsequent password change?

Use the passwd command to remove the user's password. This will effectively allow the user to log in without providing a password. For remove use password you need to use **passwd -d username_; passwd -f :** Forces the user to change password at the next login by expiring the password for name. **passwd -e** or **passwd --expire** : Immediately expire an account's password. This in effect can force a user to change his/her password at the user's next login.

11) Display the extended format of information about the directory, tell about

the information columns displayed on the terminal.



```
root@CsnKhai:~# ls -l
total 20
drwxr-xr-x 2 root root 4096 Aug 17 12:53 directory_new
-rw-r--r-- 1 root root   52 Aug 16 20:52 file.txt
drwxr-xr-x 2 root root 4096 Aug 16 20:50 new_folders
-rw-r--r-- 1 root root   17 Aug 16 21:36 orig.txt
lrwxrwxrwx 1 root root    8 Aug 16 21:35 symlink.txt -> orig.txt
drwxr-xr-x 2 root root 4096 Aug 16 21:41 test
root@CsnKhai:~#
```

- File Type and Permissions (-rw-r--r--): The first column represents the file type and permissions. This is a combination of 10 characters that indicate the file type (e.g., - for a regular file, d for directory, l for symbolic link) and the permissions for the owner, group, and others. The permissions consist of read (r), write (w), and execute (x) permissions.
- Number of Links (1): The second column represents the number of hard links to the file or directory. This number includes the directory's own entry plus any hard links created using the ln command.
- The owner (user): The third column indicates the username of the file's owner.
- Group (group): The fourth column indicates the group to which the file or directory belongs.
- Size (size): The fifth column represents the size of the file in bytes. For directories, this value might not be meaningful.
- Date and Time (date/time): The sixth column shows the date and time when the file or directory was last modified. The format can vary based on the age of the file. If the file was modified recently, it displays the time, and for older files, it displays the month and day;
- Filename (filename): The seventh and final column displays the name of the file or directory.

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

Briefly describe the acronym for access rights.

Main roles: as the owner (user), as a member of the group that owns the file (group), and as an outsider (other), have no ownership relations of the file.

The acronym for access rights – **«rwx».**

r – Read, it allows the user to view the content of a file or list the contents of a directory.

w – write, It allows users to modify the content of a file or create, rename, or delete files within a directory.

x – execute, it allows users to run executable files or scripts. For directories, it allows users to enter and access the directory's contents.

The attribute string is 3 three **rwx** that describe the file permissions of the owner of this file (the

first triplet, "u"), the group that owns the file (the second triplet, "g") and outsiders (the third triplet, "o");

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

user?

When the relationship between the file and the user who started the process, the role is determined as follows:

If the UID of the file is the same as the UID of the process, the user is the owner of the file

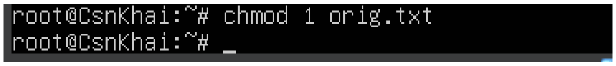
If the GID of the file matches the GID of any group the user belongs to, he is a member of the group to

which the file belongs.

If neither the UID no the GID of a file overlaps with the UID of the process and the list of groups that the user running it belongs to, that user is an outsider

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, and demonstrate on the terminal.



```
root@CsnKhai:~# chmod 1 orig.txt
root@CsnKhai:~# _
```

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

umask command.

To represent access rights using octal notation, you simply sum up the values of the permissions you want to set for each role. For example:

rwx (read, write, execute) corresponds to 7 (4 + 2 + 1).

rw- (read, write, no execute) corresponds to 6 (4 + 2).

r-- (read only) corresponds to 4 (4).

For example, umask 0 will cause files to be created with "rw-rw-rw-" attributes and directories "rwxrwxrwx". The umask 022 command removes write permissions from the default attributes for everyone except the owner (it turns out "rw-r - r--" and "rwxr-xr-x", respectively), and with umask 077 new files and directories become completely are not available ("rw -------" and "rwx ------") to everyone except their owners

16) Give definitions of sticky bits and the mechanism of identifier substitution. Give

an example of files and directories with these attributes.

Sticky Bit is mainly used on folders in order to avoid deletion of a folder and it's content by other users though they having write permissions on the folder contents. If Sticky bit is enabled on a folder, the folder contents are deleted by only owner who created them and the root user. No one else can delete other users data in this folder(Where sticky bit is set). This is a security measure to avoid deletion of critical folders and their content(sub-folders and files), though other users have full permissions.

The mechanism of identifier substitution is used in Unix-like operating systems to allow users to execute a file with the privileges of the file's owner or group.

The last special permission has been dubbed the "sticky bit." This permission does not affect individual files. However, at the directory level, it restricts file deletion. Only the owner (and root) of a file can remove the file within that directory. A common example of this is the /tmp directory

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

-a file, -b file, -c file, -d dir, -e file, -f file, -g file, -h file, -k file, -p file, -r file, -s file, -t file, -u file, -w file, -O file, -G file, -L file, -S file, -N file.