Task2

Task assignment.

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?

Structure of /etc/passwd File:

- Username: The user's login name.
- Password: Historically, the hashed password. On modern systems, this field often contains a placeholder character (e.g., x) and the actual password hash is stored in the /etc/shadow file.
- User ID (UID): A unique numerical identifier for the user.
- Group ID (GID): The numerical identifier of the primary group of the user.
- User Info: Additional information about the user (often the user's full name or a description).
- Home Directory: The user's home directory.
- Shell: The default shell for the user.

```
root@CsnKhai:/etc# cat passwd
root:x:0:0:root,,,:/root:/bin/bash
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/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
libuuid:x:100:101::/var/lib/libuuid:
syslog:x:101:104::/home/syslog:/bin/false
messagebus:x:102:105::/var/run/dbus:/bin/false
sshd:x:103:65534::/var/run/sshd:/usr/sbin/nologin
student:x:1000:1000:Bohdan Lesyk,,,,SoftServe student:/home/student:/bin/bash
root@CsnKhai:/etc#
```

Structure of /etc/group File:

- Group Name: The name of the group.
- Group Password: Historically, the hashed group password. Similar to the password field in /etc/passwd, this field often contains a placeholder character (e.g., x) and the actual password hash is stored in the /etc/gshadow file.
- Group ID (GID): A unique numerical identifier for the group.
- Group Members: A comma-separated list of usernames that are members of the group.

```
root@CsnKhai:/etc# cat 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:
```

Pseudo-users, also known as system users, are special user accounts on a Unix-like system that are not meant for human login but are used to manage system processes and provide specific functionalities. For example root or daemon users.

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

In the Linux operating system, UID stands for User Identifier. It is a unique numerical value assigned to each user account. The UID is used by the system to identify and differentiate between different users. UID values are typically assigned within specific ranges to differentiate between different types of users.

System Users are usually pseudo-users or accounts used for system processes. Their UIDs are often in the range of 1 to 999.

Regular Users are the accounts created for human users. Regular user UIDs usually start from 1000 and go upwards.

Reserved UIDs like the UID 0 for the superuser (root).

We can create a new user with specific UID of 2000.

```
student@CsnKhai:~$ sudo useradd -u 2000 newuser
[sudo] password for student:
student@CsnKhai:~$ id -u newuser
2000
student@CsnKhai:~$
```

3) What is GID? How to define it?

GID, is a numeric value used to represent a specific group. GIDs are used to associate ownership of files, processes, and other resources with specific groups.

```
student@CsnKhai:~$ sudo groupmod -g 3000 newuser
[sudo] password for student:
student@CsnKhai:~$ id -g newuser
3000
student@CsnKhai:~$
```

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:~$
```

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

```
student@CsnKhai:~$ sudo useradd newuser2
[sudo] password for student:
student@CsnKhai:~$
```

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

```
student@CsnKhai:~$ sudo usermod -l bohdan newuser
student@CsnKhai:~$
```

7) What is skell_dir? What is its structure?
Directory /etc/skel/ is used to initiate home directory when a user is first created.

```
student@CsnKhai:~$ cd /etc/skel
student@CsnKhai:/etc/skel$ ls -la
total 20
drwxr-xr-x  2 root root 4096 Sep 15  2015 .
drwxr-xr-x 83 root root 4096 Aug 17 11:31 ..
-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
student@CsnKhai:/etc/skel$
```

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

```
student@CsnKhai:~$ sudo userdel newuser2
student@CsnKhai:~$
```

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

```
student@CsnKhai:~$ sudo passwd -l bohdan
passwd: password expiry information changed.
student@CsnKhai:~$ sudo usermod -L bohdan
student@CsnKhai:~$ sudo passwd -u bohdan
passwd: password expiry information changed.
student@CsnKhai:~$ sudo usermod -U bohdan
student@CsnKhai:~$
```

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

```
student@CsnKhai:~$ sudo passwd -d bohdan
passwd: password expiry information changed.
student@CsnKhai:~$
```

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

```
student@CsnKhai:~$ ls -la
total 40
drwxr-xr-x 3 student student 4096 Aug 17 10:31 .
drwxr-xr-x 3 root    root    4096 Sep 15  2015 ..
-rw-------  1 student student  127 Aug 16 13:29 .bash_history
-rw-r--r--  1 student student  220 Sep 15  2015 .bash_logout
-rw-r--r--  1 student student 3637 Sep 15  2015 .bashrc
drwx------  2 student student 4096 Sep 15  2015 .cache
-rw-rw-r--  1 student student   13 Aug 16 14:13 example_file.txt
-rw-rw-r--  1 student student   66 Aug 16 13:31 .plan
-rw-r--r--  1 student student  675 Sep 15  2015 .profile
-rw-------  1 student student   53 Aug 17 10:31 .Xauthority
```

We can see file type and permissions, link count, owner, group, file size, modification date and time and name of file.

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

There are three main categories of access rights, often represented by acronyms, that determine what actions (read, write, execute) can be performed by different user roles (owner, group, others) on a file or directory. For example "rwxrw-r--" means the owner has read, write, and execute permissions, the group has read and write permissions, and others have only read permission.

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

The relationship between a file and a user is defined through file ownership and permissions.

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.

Ownership can be changed using the chown command. Permission can be changed using chmod command.

```
student@CsnKhai:~$ sudo chown bohdan example_file.txt
[sudo] password for student:
student@CsnKhai:~$ sudo chmod 777 example_file.txt
student@CsnKhai:~$ █
```

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

The octal representation of access rights is a way to represent file permissions using a three-digit octal number. Each digit corresponds to a different permission category: owner, group, and others.

Each permission (read, write, execute) is assigned a numeric value, Read(4), write(2), execute(1), By summing these values, you can create an octal number that represents the permissions for each category.

The umask utility allows you to view or to set the file mode creation mask, which determines the permissions bits for newly created files or directories.

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.

Identifier substitution, also known as variable substitution or parameter expansion, is a mechanism used to replace placeholders with actual values. It involves using special symbols or syntax to represent variables or values that will be substituted during execution.

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

The main attributes that should be present or considered in a command script file are execute, read and write permission and shebang (beginning of the script).