

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@db1:/etc# cat /etc/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
systemd-network:x:100:102:systemd Network Management,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106:/:nonexistent:/usr/sbin/nologin
syslog:x:104:110:/:home/syslog:/usr/sbin/nologin
_apt:x:105:65534:/:nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,:/var/lib/tpm:/bin/false
uidd:x:107:112:/:run/uidd:/usr/sbin/nologin
tcpdump:x:108:113:/:nonexistent:/usr/sbin/nologin
landscape:x:109:115:/:var/lib/landscape:/usr/sbin/nologin
pollinate:x:110:1:/:var/cache/pollinate:/bin/false
usbmux:x:111:46:usbmux daemon,,:/var/lib/usbmux:/usr/sbin/nologin
sshd:x:112:65534:/:run/sshd:/usr/sbin/nologin
systemd-coredump:x:999:999:systemd Core Dumper:/:usr/sbin/nologin
db1:x:1000:1000:Ihor Mishchenko,,0915001000,0925002000,Lab account for lerning Linux:/home/db1:/bin/bash
lxd:x:1001:1001:/:var/lib/lxd/daemon:/bin/false
```

Db1-user

x-pass

1000 – UserID

1000 – Group ID

Ihor Mishchenko. 0915001000,0925002000, UserInfo

/home/db1 user home

/bin/bash – shell

Cat /etc/group

```
db1:x:1000:
```

Db1 group name

x- password

1000 – group ID

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

id - print real and effective user and group IDs

```
root@db1:/etc#
root@db1:/etc# id db1
uid=1000(db1) gid=1000(db1) groups=1000(db1),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),116(lxd)
```

3)What is GID? How to define it?

Cat/etc/group | grep db1

```
root@db1:/etc# cat /etc/group | grep db1
adm:x:4:syslog,db1
cdrom:x:24:db1
sudo:x:27:db1
dip:x:30:db1
plugdev:x:46:db1
lxd:x:116:db1
db1:x:1000:
```

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

Groups db1

```
root@db1:/etc# groups db1
db1 : db1 adm cdrom sudo dip plugdev lxd
root@db1:/etc#
```

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

add new user: `sudo useradd user01`

create for him a password: `sudo passwd user01`

create home directory: `sudo useradd -m user01`

creating a user with specific home directory: `sudo useradd -m -d /path_name user01`

creating a user with specific User ID: `sudo useradd -u 1500 user01`

creating a user with an expiry date: `sudo useradd -e 2021-12-31 user01`

creating a user and assign multiple groups: `sudo useradd -g users -G wheel,devops user01`

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

`sudo usermod -l user01 user02`

7) What is `skell_dir`? What is its structure?

Directory `/etc/skel/` (skel is derived from the "skeleton") is used to initiate home directory when a user is first created.

```
root@db1:/etc# ls -la /etc/skel/
total 20
drwxr-xr-x  2 root root 4096 Aug 24 08:45 .
drwxr-xr-x 100 root root 4096 Dec  2 16:14 ..
-rw-r--r--  1 root root  220 Feb 25  2020 .bash_logout
-rw-r--r--  1 root root 3771 Feb 25  2020 .bashrc
-rw-r--r--  1 root root  807 Feb 25  2020 .profile
root@db1:/etc#
```

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

`sudo userdel -r user01`

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

first way using `passwd` command to lock: `sudo passwd -l user01`

to unlock: `sudo passwd -u user01`

second way using `usermod` command:

to lock: `sudo usermod -L user01`

to unlock: `sudo usermod -U user01`

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

To do so, we need open the sudoers configuration file with sudo visudo, and add the following line to the file:

```
user01 ALL=(ALL) NOPASSWD:ALL
```

Then delete the password for your user by running this command:

```
sudo passwd -d `user01`
```

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

Long Output Format of ls command `ls -l`

```
root@db1:/etc# ls -l
total 820
-rw-r--r-- 1 root root      3028 Aug 24 08:42 adduser.conf
drwxr-xr-x 2 root root     4096 Dec  9 17:12 alternatives
drwxr-xr-x 3 root root     4096 Aug 24 08:47 apparmor
drwxr-xr-x 7 root root     4096 Nov 23 10:04 apparmor.d
drwxr-xr-x 3 root root     4096 Nov 23 09:31 appport
drwxr-xr-x 7 root root     4096 Nov 23 09:29 apt
-rw-r----- 1 root daemon   144 Nov 12 2018 at.deny
-rw-r--r-- 1 root root    2319 Feb 25 2020 bash.bashrc
-rw-r--r-- 1 root root     45 Jan 26 2020 bash_completion
drwxr-xr-x 2 root root     4096 Nov 23 09:31 bash_completion.d
-rw-r--r-- 1 root root     367 Apr 14 2020 bindresvport.blacklist
drwxr-xr-x 2 root root     4096 Apr 22 2020 binfo.d
drwxr-xr-x 2 root root     4096 Aug 24 08:47 byobu
drwxr-xr-x 3 root root     4096 Aug 24 08:42 ca-certificates
-rw-r--r-- 1 root root    6570 Nov 23 09:31 ca-certificates.conf
-rw-r--r-- 1 root root    6569 Aug 24 08:45 ca-certificates.conf.dpkg-old
drwxr-xr-x 2 root root     4096 Aug 24 08:47 calendar
drwxr-xr-x 4 root root     4096 Nov 23 09:30 cloud
drwxr-xr-x 2 root root     4096 Nov 23 09:34 console-setup
drwxr-xr-x 2 root root     4096 Aug 24 08:47 cron.d
drwxr-xr-x 2 root root     4096 Dec  9 16:24 cron.daily
drwxr-xr-x 2 root root     4096 Aug 24 08:43 cron.hourly
drwxr-xr-x 2 root root     4096 Aug 24 08:43 cron.monthly
-rw-r--r-- 1 root root    1042 Feb 13 2020 crontab
drwxr-xr-x 2 root root     4096 Aug 24 08:47 cron.weekly
drwxr-xr-x 2 root root     4096 Aug 24 08:47 cryptsetup-initramfs
-rw-r--r-- 1 root root     54 Aug 24 08:46 crypttab
drwxr-xr-x 4 root root     4096 Aug 24 08:42 dbus-1
drwxr-xr-x 3 root root     4096 Aug 24 08:46 dconf
-rw-r--r-- 1 root root    2969 Aug  3 2019 debconf.conf
-rw-r--r-- 1 root root     13 Dec  5 2019 debian_version
drwxr-xr-x 3 root root     4096 Nov 23 09:31 default
-rw-r--r-- 1 root root     604 Sep 15 2018 deluser.conf
drwxr-xr-x 2 root root     4096 Aug 24 08:43 depmod.d
drwxr-xr-x 4 root root     4096 Aug 24 08:45 dhcp
drwxr-xr-x 4 root root     4096 Aug 24 08:42 dpkg
-rw-r--r-- 1 root root     685 Feb 14 2020 e2scrub.conf
-rw-r--r-- 1 root root     106 Aug 24 08:45 environment
-rw-r--r-- 1 root root    1816 Dec 27 2019 ethertypes
drwxr-xr-x 4 root root     4096 Aug 24 08:45 fonts
-rw-r--r-- 1 root root     657 Nov 23 09:30 fstab
-rw-r--r-- 1 root root     280 Jun 20 2014 fuse.conf
drwxr-xr-x 3 root root     4096 Aug 24 08:47 fwupd
-rw-r--r-- 1 root root    2584 Feb  1 2020 gai.conf
drwxr-xr-x 2 root root     4096 Aug 24 08:47 groff
-rw-r--r-- 1 root root     782 Nov 23 10:04 group
-rw-r--r-- 1 root root     769 Nov 23 09:34 group-
drwxr-xr-x 2 root root     4096 Nov 23 09:30 grub.d
-rw-r--r-- 1 root shadow   653 Nov 23 10:04 gshadow
-rw-r--r-- 1 root shadow   643 Nov 23 09:34 gshadow-
drwxr-xr-x 3 root root     4096 Aug 24 08:45 gss
-rw-r--r-- 1 root root    5060 Aug 21 2019 hdparm.conf
-rw-r--r-- 1 root root     92 Dec  5 2019 host.conf
```

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

Permission settings grouped in a string of characters (-, r, w, x) classified into four sections:

File type. There are three possibilities for the type. It can either be a regular file (-), a directory (d) or a link (l).

File permission of the user (owner)

File permission of the owner's group

File permission of other users

The characters **r**, **w**, and **x** stand for **read**, **write**, and **execute**. Instead of letters, the octal format represents privileges with numbers:

r(ead) has the value of 4

w(rite) has the value of 2

(e)x(ecute) has the value of 1

no permission has the value of 0

The privileges are summed up and depicted by one number. Therefore, the possibilities are:

7 – for read, write, and execute permission

6 – for read and write privileges

5 – for read and execute privileges

4 – for read privileges

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.

```
sudo chown root hard_link_labwork2
```

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

When a file is created, the permission flags are set according to the file mode creation mask, which can be set using the umask command. The file mode creation mask (sometimes referred to as "the umask") is a three-digit octal value whose nine bits correspond to fields 2-10 of the permission flags.

Octal value Binary representation Meaning

0 0 No read, write, and execute permissions (—)

1 1 Only execute permission (-x)

2 10 Only write permission (-w-)

3 11 Write and execute permissions (-wx)

4 100 Only read permission (r—)

5 101 Read and execute permissions (r-x)

6 110 Read and write permissions (rw-)

7 111 Read, write, and execute permissions (rwx)

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

The sticky bit was initially introduced to 'stick' an executable program's text segment in the swap space even after the program has completed execution, to speed up the subsequent runs of the same program. However, these days the sticky bit means something entirely different.

When a directory has the sticky bit set, its files can be deleted or renamed only by the file owner, directory owner and the root user. `mkdir allAccess`

```

chmod 777 allAccess/
cd allAccess
touch file1.txt
cd -
sudo adduser guest
sudo adduser guest1
su guest
cd allAccess
touch file2.txt
exit
su guest1
cd allAccess
touch file3.txt
exit
cd allAccess
chmod 777 file1.txt
sudo chmod 777 file2.txt
sudo chmod 777 file3.txt
cd -
chmod +t allAccess/
su guest

```

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

The files and directories can have following attributes:

Value	Meaning
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

Value Meaning lsattr