Lab 3, Week 4

Due date: Friday, February 03 23:30

Lab written by

- Le, Stanley A00961484
- Leung, Wen Kai Quinten A01055681
- Lim, Norvin A01328954

Make sure that you work with your lab partners. If a team member is absent contact them and work with them remotely via discord, zoom, ravens...

Don't divide up the work, work on each question together. It is important that you all learn the material in every lab.

Lab must be completed as a regular user, not the root user.

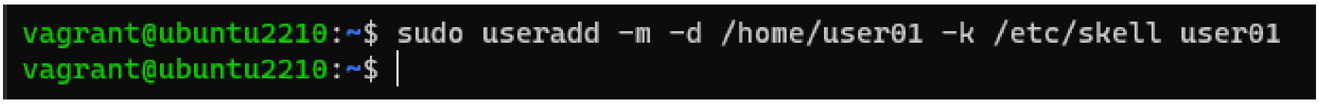
Instructions

Part one: 2 points

Create a new regular user with the `useradd` utility. Your new users should have:

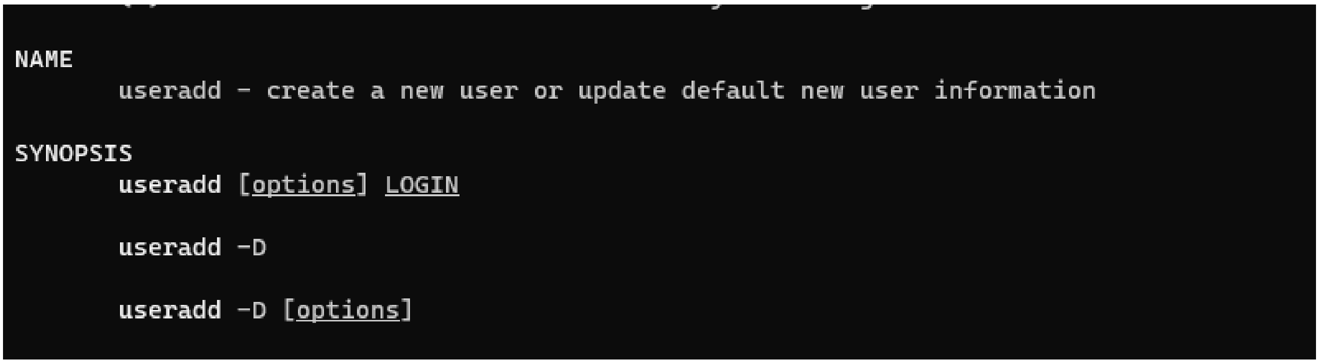
- a home directory /home/user-name
  - config files from /etc/skell should be copied to the new user's home directory
- use bash as their login shell
  USER ADD
  Include 4 screenshots for this.

Include 1 screenshot that demonstrates the command you used.

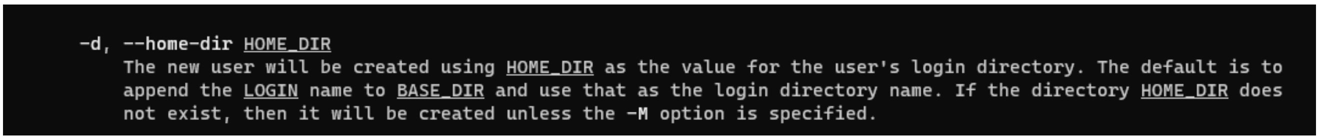


```
vagrant@ubuntu2210:~$ sudo useradd -m -d /home/user01 -k /etc/skell user01
vagrant@ubuntu2210:~$ 
```

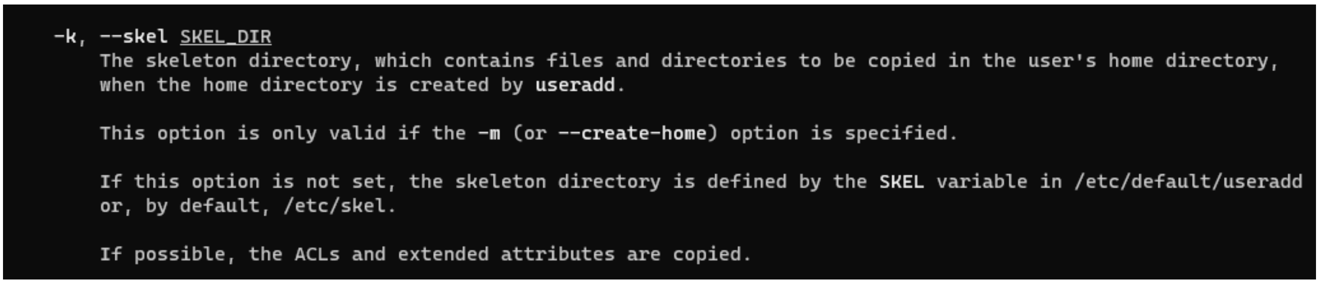
Include 2 screenshots that show where you found how to do both of the above bullet points in the man page for `useradd`



```
NAME
       useradd - create a new user or update default new user information

SYNOPSIS
       useradd [options] LOGIN

       useradd -D

       useradd -D [options]
```

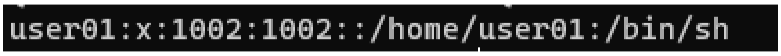


```
       -d, --home-dir HOME_DIR
           The new user will be created using HOME_DIR as the value for the user's login directory. The default is to
           append the LOGIN name to BASE_DIR and use that as the login directory name. If the directory HOME_DIR does
           not exist, then it will be created unless the -M option is specified.
```



```
       -k, --skel SKEL_DIR
           The skeleton directory, which contains files and directories to be copied in the user's home directory,
           when the home directory is created by useradd.

           This option is only valid if the -m (or --create-home) option is specified.

           If this option is not set, the skeleton directory is defined by the SKEL variable in /etc/default/useradd
           or, by default, /etc/skel.

           If possible, the ACLs and extended attributes are copied.
```

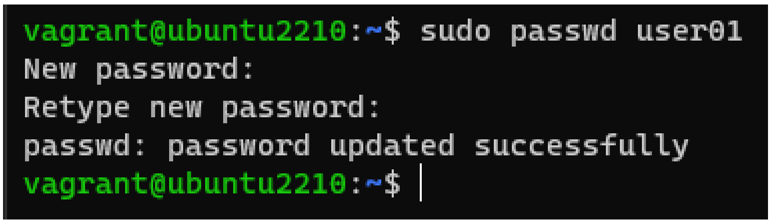
Include a screenshot that demonstrates you succeeded in creating a new user by finding the new line created in a file that contains information about users. We have looked at this file a few times.



```
user01:x:1002:1002::/home/user01:/bin/sh
```

Part two: 1 point

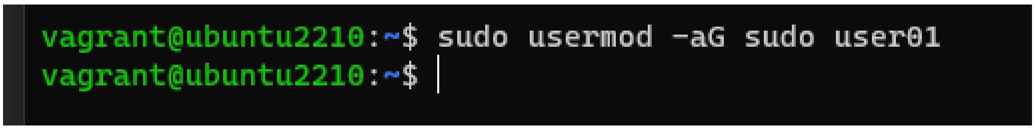
Give your new user a password.
Submit a screenshot of the command that you used.



```
vagrant@ubuntu2210:~$ sudo passwd user01
New password:
Retype new password:
passwd: password updated successfully
vagrant@ubuntu2210:~$
```

Part three: 1 point

Give your new user permission to use sudo.
Submit a screenshot of the command you used.

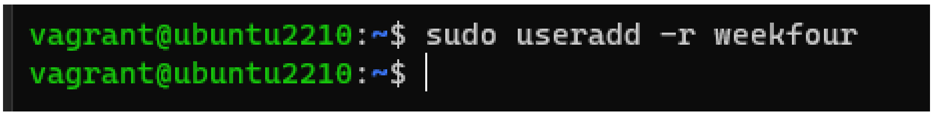


```
vagrant@ubuntu2210:~$ sudo usermod -aG sudo user01
vagrant@ubuntu2210:~$
```

Part four: 1 point

Create a new system user with the name “weekfour”

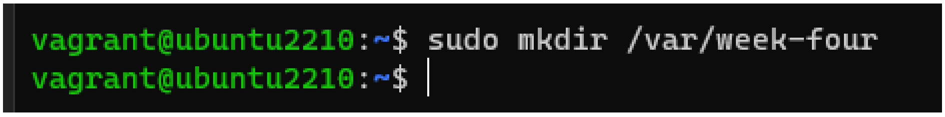
Submit a screenshot of the command you used.



```
vagrant@ubuntu2210:~$ sudo useradd -r weekfour
vagrant@ubuntu2210:~$ |
```

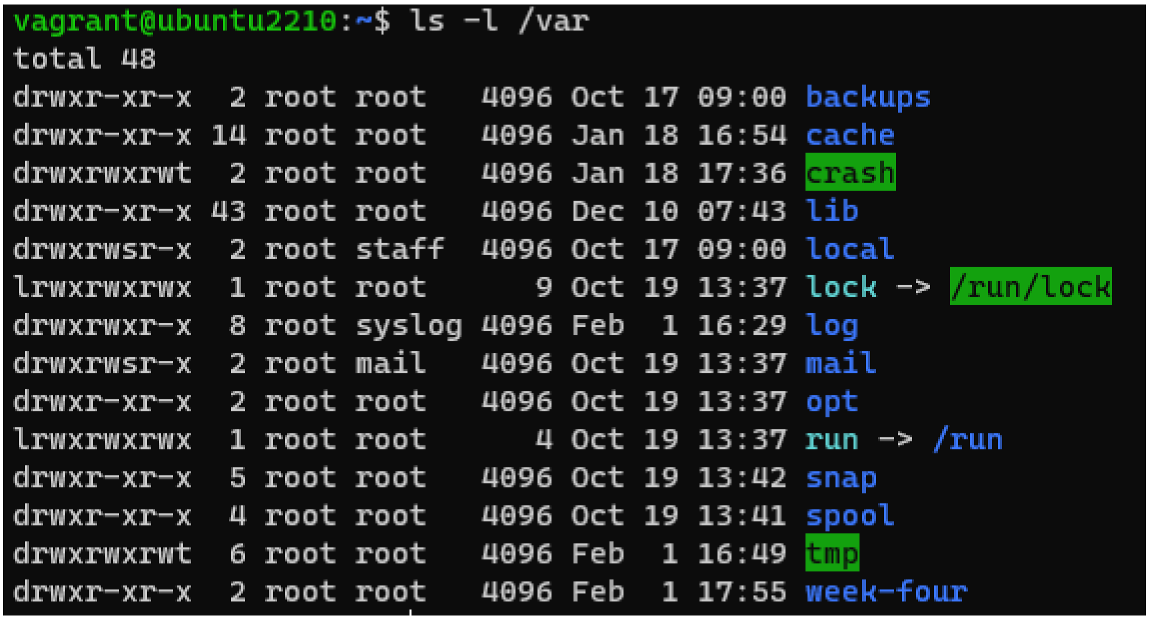
Part five: 2 points

create a new week4 directory in `/var`



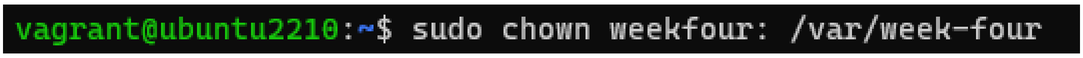
```
vagrant@ubuntu2210:~$ sudo mkdir /var/week-four
vagrant@ubuntu2210:~$ |
```

Make the weekfour user and group the owner of the week4 directory

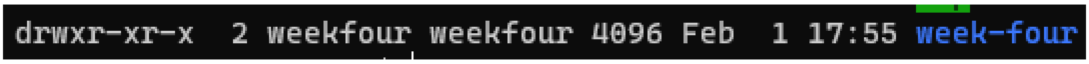


```
vagrant@ubuntu2210:~$ ls -l /var
total 48
drwxr-xr-x  2 root root   4096 Oct 17 09:00 backups
drwxr-xr-x 14 root root   4096 Jan 18 16:54 cache
drwxrwxrwt  2 root root   4096 Jan 18 17:36 crash
drwxr-xr-x 43 root root   4096 Dec 10 07:43 lib
drwxrwsr-x  2 root staff  4096 Oct 17 09:00 local
lrwxrwxrwx  1 root root      9 Oct 19 13:37 lock -> /run/lock
drwxrwxr-x  8 root syslog 4096 Feb  1 16:29 log
drwxrwsr-x  2 root mail   4096 Oct 19 13:37 mail
drwxr-xr-x  2 root root   4096 Oct 19 13:37 opt
lrwxrwxrwx  1 root root      4 Oct 19 13:37 run -> /run
drwxr-xr-x  5 root root   4096 Oct 19 13:42 snap
drwxr-xr-x  4 root root   4096 Oct 19 13:41 spool
drwxrwxrwt  6 root root   4096 Feb  1 16:49 tmp
drwxr-xr-x  2 root root   4096 Feb  1 17:55 week-four
```

Submit a screenshot of the command you used to change the ownership of the directory.



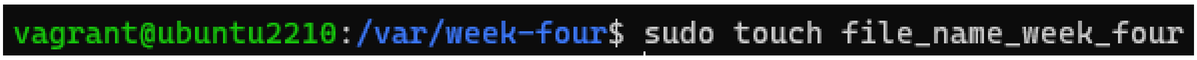
```
vagrant@ubuntu2210:~$ sudo chown weekfour: /var/week-four
```



```
drwxr-xr-x  2 weekfour weekfour 4096 Feb  1 17:55 week-four
```

Part six: 2 points

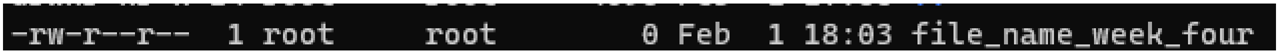
create a new file in the week4 directory (file name is up to you)



```
vagrant@ubuntu2210:/var/week-four$ sudo touch file_name_week_four
```

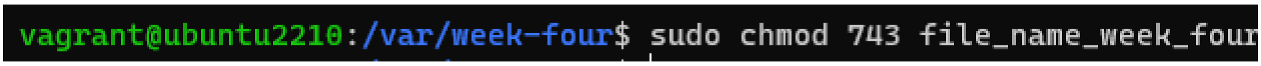
Change the file's permission to match the following.

`rwxr---wx`

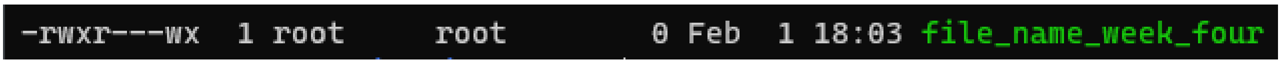


```
-rw-r--r--  1 root     root          0 Feb  1 18:03 file_name_week_four
```

Submit a screenshot of the command you used to change the file's permission.



```
vagrant@ubuntu2210:/var/week-four$ sudo chmod 743 file_name_week_four
```



```
-rwxr---wx  1 root     root          0 Feb  1 18:03 file_name_week_four
```

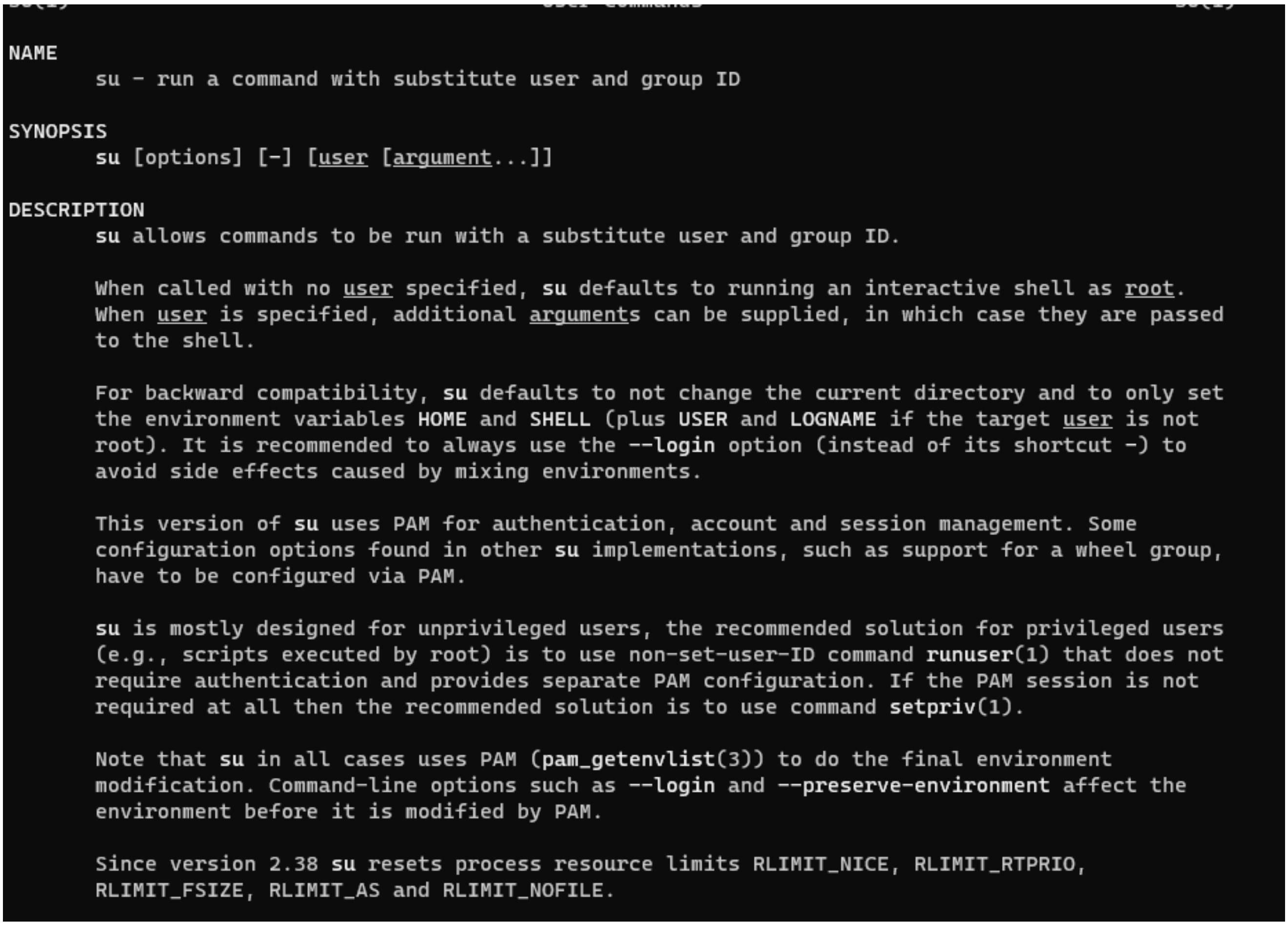
Include a note on why you think these permissions are odd.
This only needs to be 1-2 sentences.

- Owner - This is fine and perfectly normal to have owner with all three permissions
- Group - This allows other members of the group to read the file however
- Other - This allows other users to write and execute the file

The odd things about these settings are that Group treats the file as a read only file that cannot be executed. However, the Other permission allow you to write onto the same file and execute it implying the same file serve different purposes.

Part seven: 2 points

How could you change to your new regular user created in part one? Without logging out and logging back in again. In addition to becoming the new user, you want to inherit their environment variables and shell configuration.
`su <user name>`
Include the command you found and citations to any resources you used.



```
NAME
       su - run a command with substitute user and group ID

SYNOPSIS
       su [options] [-] [user [argument...]]

DESCRIPTION
       su allows commands to be run with a substitute user and group ID.

       When called with no user specified, su defaults to running an interactive shell as root.
       When user is specified, additional arguments can be supplied, in which case they are passed
       to the shell.

       For backward compatibility, su defaults to not change the current directory and to only set
       the environment variables HOME and SHELL (plus USER and LOGNAME if the target user is not
       root). It is recommended to always use the --login option (instead of its shortcut -) to
       avoid side effects caused by mixing environments.

       This version of su uses PAM for authentication, account and session management. Some
       configuration options found in other su implementations, such as support for a wheel group,
       have to be configured via PAM.

       su is mostly designed for unprivileged users, the recommended solution for privileged users
       (e.g., scripts executed by root) is to use non-set-user-ID command runuser(1) that does not
       require authentication and provides separate PAM configuration. If the PAM session is not
       required at all then the recommended solution is to use command setpriv(1).

       Note that su in all cases uses PAM (pam_getenvlist(3)) to do the final environment
       modification. Command-line options such as --login and --preserve-environment affect the
       environment before it is modified by PAM.

       Since version 2.38 su resets process resource limits RLIMIT_NICE, RLIMIT_RTPRIO,
       RLIMIT_FSIZE, RLIMIT_AS and RLIMIT_NOFILE.
```

Linux: How to Run a command as another user
You don’t need to format your citation using any specific style guide.
A man page can be a resource.

Total: 11 Points

Submission Instructions

Submit a .pdf using the dropbox on D2L

Title your pdf “your-name-lab3.pdf” ie "nathan-mcninch-lab3.pdf”

File must be a .pdf.
Only one submission per team, make sure that all team members names are on the file.