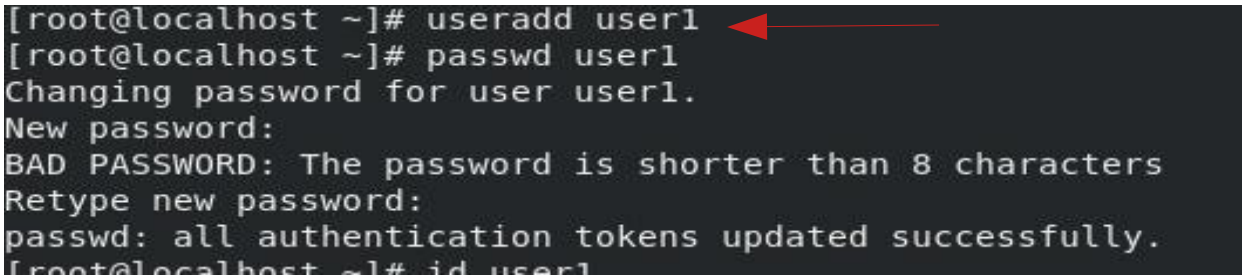
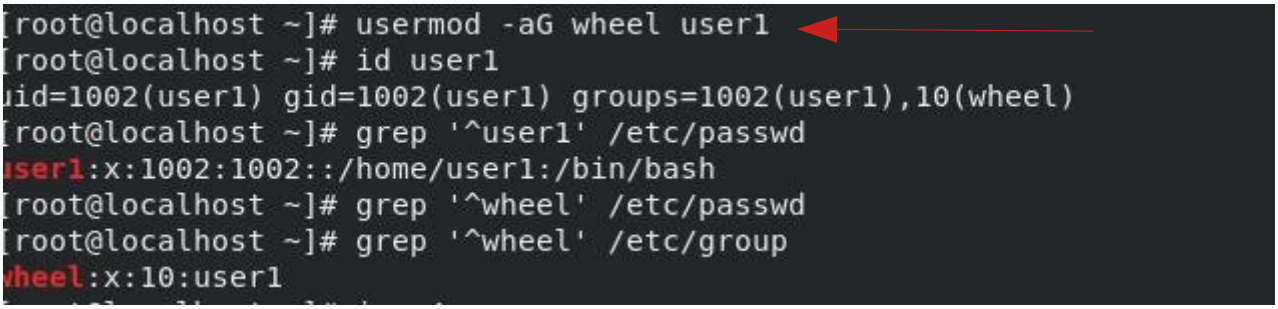
Task No:7

*Create a normal User give it only sudo particular permission *

STEP:-(1)---> First, I create a normal user named **user1** and set a password then run the following command:



STEP:-(2)---> After that, I verified it: then used the usermod command to add the user to the wheel group: because all members of the wheel group have sudo access.



STEP:-(3.)---> To test that the new sudo permissions are working, first use the **su** command to switch from the root user to the new user account.

[user1@localhost ~]$ su - user1

Password: **********

[user1@localhost ~]$

For example:- you can list the contents of the ***/root*** directory, which is normally only accessible to the root user:

When you use this command you notice that here the root user's password is not asked, enter the password of the user created by us like **User1**.

Note:- If i talk about what is **diffrence between su and sudo** then you can give answer su switches you to the root account and requires the root accounts's password but sudo is runs with single commands with root privilleges and it doest not switch to the root account's.

```
[user1@localhost ~]$ sudo ls -la /root
[sudo] password for user1:
total 48
dr-xr-x---.  7 root root   271 Jan 16 13:17 .
dr-xr-xr-x. 17 root root   224 Jan 11 11:24 ..
-rw-------.  1 root root  1269 Jan 11 11:53 anaconda-ks.cfg
drwx------.  3 root root    17 Jan 14 11:39 .ansible
-rw-------.  1 root root   604 Jan 16 12:55 .bash_history
-rw-r--r--.  1 root root    18 May 11  2019 .bash_logout
-rw-r--r--.  1 root root   176 May 11  2019 .bash_profile
-rw-r--r--.  1 root root   176 May 11  2019 .bashrc
drwx------.  4 root root    44 Jan 11 11:57 .cache
drwx------.  3 root root    18 Jan 11 11:58 .config
-rw-r--r--.  1 root root   100 May 11  2019 .cshrc
drwx------.  3 root root    25 Jan 11 11:57 .dbus
-rw-r--r--.  1 root root  1561 Jan 11 11:58 initial-setup-ks.cfg
drwxr-xr-x.  3 root root    19 Jan 16 13:17 .local
-rw-r--r--.  1 root root   129 May 11  2019 .tcshrc
-rw-------.  1 root root 11215 Jan 15 11:45 .viminfo
-rw-------.  1 root root   130 Jan 16 12:47 .xauthP3QXrC
[user1@localhost ~]$
```

STEP:(4.) --->

If i talk about admin privileges using sudo as it keeps track of user account in a log file then.

```
[user1@localhost ~]$ sudo grep user1 /var/log/secure | grep -i command
[sudo] password for user1:
Jan 16 13:52:29 localhost sudo[57753]:   user1 : TTY=pts/1 ; PWD=/home/user1 ; USER=root ; COMMAND=/bin/ls -la /root
[user1@localhost ~]$ sudo grep user1 /var/log/secure
Jan 16 13:50:22 localhost su[57687]: pam_unix(su-l:session): session opened for user user1 by user1(uid=1002)
Jan 16 13:52:29 localhost sudo[57753]:   user1 : TTY=pts/1 ; PWD=/home/user1 ; USER=root ; COMMAND=/bin/ls -la /root
Jan 16 13:52:29 localhost sudo[57753]: pam_unix(sudo:session): session opened for user root by user1(uid=0)
Jan 16 14:01:23 localhost sudo[57900]:   user1 : TTY=pts/1 ; PWD=/home/user1 ; USER=root ; COMMAND=/bin/grep user1 /var/log/secure
Jan 16 14:01:23 localhost sudo[57900]: pam_unix(sudo:session): session opened for user root by user1(uid=0)
Jan 16 14:01:47 localhost sudo[57920]:   user1 : TTY=pts/1 ; PWD=/home/user1 ; USER=root ; COMMAND=/bin/grep user1 /var/log/secure
Jan 16 14:01:47 localhost sudo[57920]: pam_unix(sudo:session): session opened for user root by user1(uid=0)
[user1@localhost ~]$ sudo tail -f /var/log/secure
Jan 16 13:52:29 localhost sudo[57753]: pam_unix(sudo:session): session opened for user root by user1(uid=0)
Jan 16 13:52:29 localhost sudo[57753]: pam_unix(sudo:session): session closed for user root
Jan 16 14:01:23 localhost sudo[57900]:   user1 : TTY=pts/1 ; PWD=/home/user1 ; USER=root ; COMMAND=/bin/grep user1 /var/log/secure
Jan 16 14:01:23 localhost sudo[57900]: pam_systemd(sudo:session): Cannot create session: Already running in a session or user slice
Jan 16 14:01:23 localhost sudo[57900]: pam_unix(sudo:session): session opened for user root by user1(uid=0)
Jan 16 14:01:23 localhost sudo[57900]: pam_unix(sudo:session): session closed for user root
Jan 16 14:01:47 localhost sudo[57920]:   user1 : TTY=pts/1 ; PWD=/home/user1 ; USER=root ; COMMAND=/bin/grep user1 /var/log/secure
Jan 16 14:01:47 localhost sudo[57920]: pam_systemd(sudo:session): Cannot create session: Already running in a session or user slice
Jan 16 14:01:47 localhost sudo[57920]: pam_unix(sudo:session): session opened for user root by user1(uid=0)
Jan 16 14:01:47 localhost sudo[57920]: pam_unix(sudo:session): session closed for user root
Jan 16 14:02:51 localhost sudo[57937]:   user1 : TTY=pts/1 ; PWD=/home/user1 ; USER=root ; COMMAND=/bin/tail -f /var/log/secure
Jan 16 14:02:51 localhost sudo[57937]: pam_systemd(sudo:session): Cannot create session: Already running in a session or user slice
Jan 16 14:02:51 localhost sudo[57937]: pam_unix(sudo:session): session opened for user root by user1(uid=0)
```

That’s it. The user now has sudo privileges.

Note:-

If you get a notification like **“User is not in the sudoers file,”** It means process is not successfull then your user is not sudo privileges.After that you can add **user1** the sudoers file- **/etc/sudoers.**

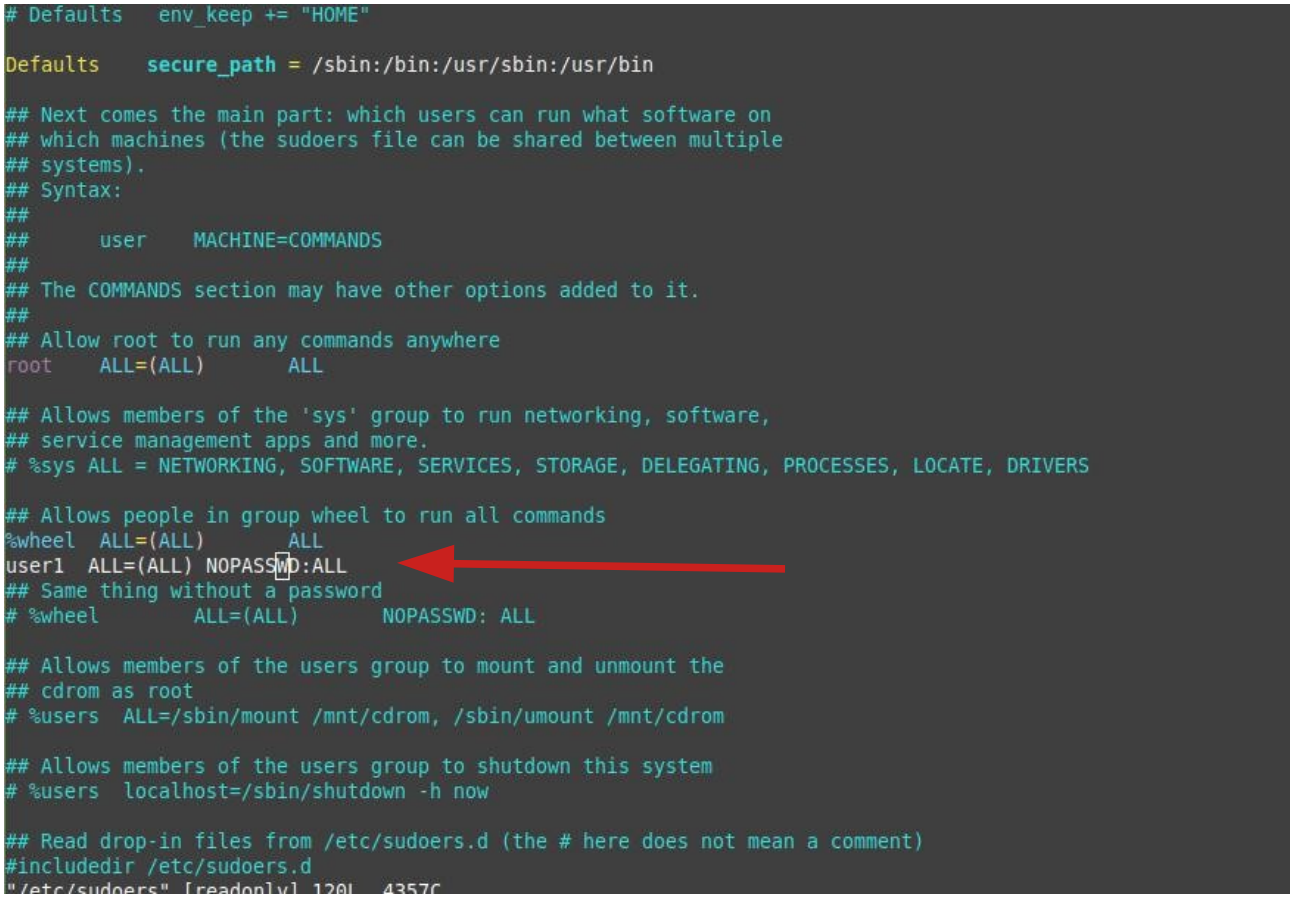
Here i wil use vim **editor** to open this file **/etc/sudoers** file to changes sudoers file**.**

[user1@localhost ~]$ su – root

password:- ************

[root@localhost ~]# vim /etc/sudoers

user1 ALL=(ALL) NOPASSWD:ALL



You have allowed the user to run sudo commands without password authentication by adding the line above.Now new user named **user1** is also full sudo access.