1. **Configure smtp in localhost**

step-1--> Install the required packages

sudo apt-get install ssmtp --> to install SMTP package

sudo apt-get install mailutils --> to install mail package

step-2 --> After installing mail, go to google settings and generate app password

While generating select mail as an app and device as other and give name as mail\_txt

Then it will generate 12 digit password and save that password

Enable IMAP in your gmail account, by going to settings of gmail account and click on "forwording and pop imap" and enable IMAP

Now we have to add our app password and email id in ssmtp.conf. for that

step-3 --> add the below setting to ssmtp.conf

open terminal

sudo nano /etc/ssmtp/ssmtp.conf --> to open ssmtp.conf file and add the following credentials

SERVER=<your gmail account> --> add your gmail account AS SERVER

AuthUser=<source gmail account> --> mention the source gmail account name

AuthPass=<app password> --> enter the app password generated in google

useTLS=YES --> to send mail through secured layer

userSTARTTLS=YES --> to secure the password

rewriteDomain=gmail.com --> to set domain an gmail

fromlineoverride=YES --> to give acccess to user to send mail to required mail account

step-4 --> To send mail through terminal we use the following steps,

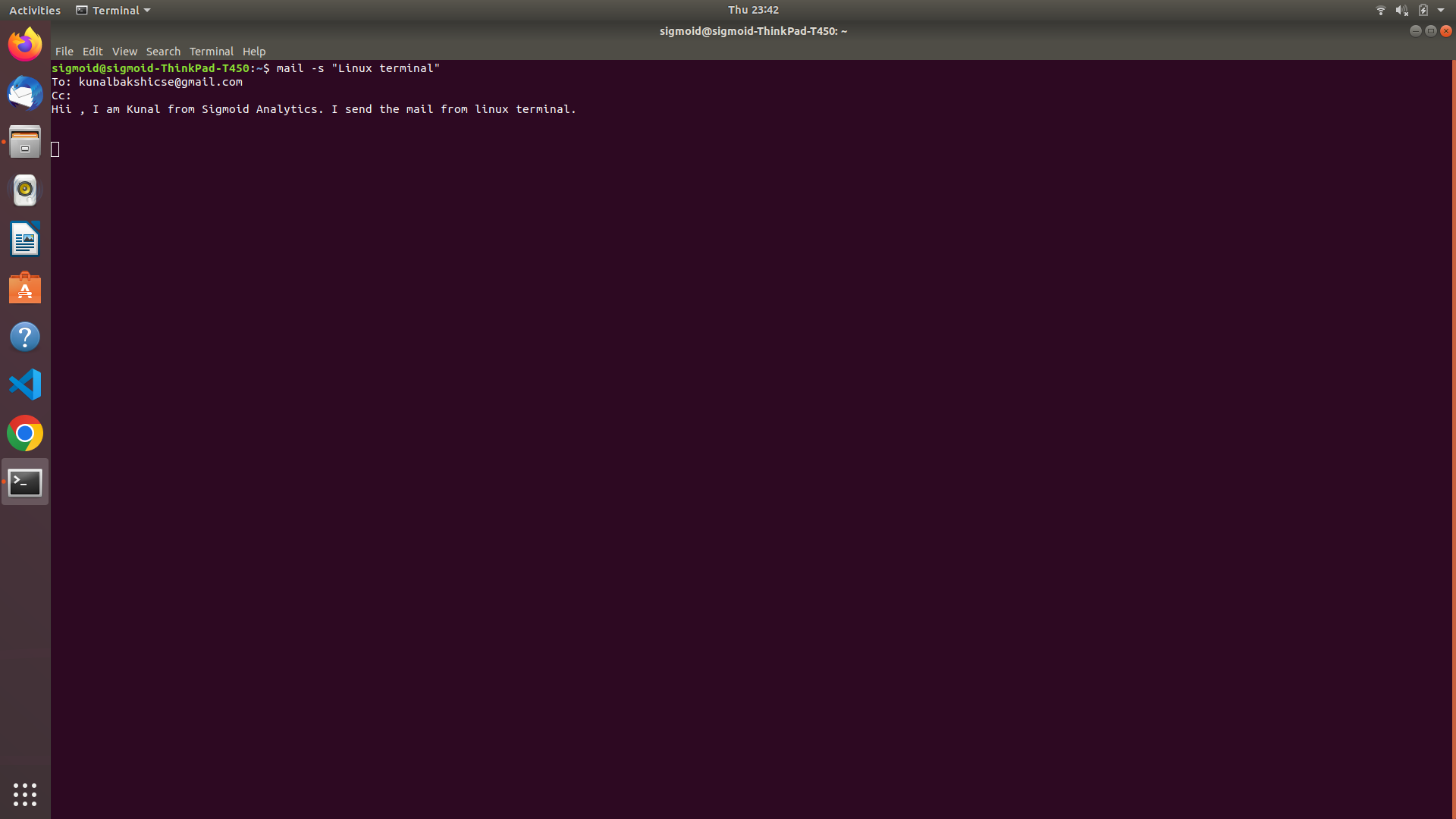
mail -s "<subject of mail>" --> command to send mail through terminal

To: <destination mail address> --> gmail address of destination

cc: <add cc mail address> --> if want to send same mail to other mail

<email body> --> body of the mail and click enter

ctrl+d --> to send the mail

**s**

1. **Create a user in your localhost, which should not be able to execute the sudo command.**

step-1) sudo su : To change the user into root

step-2) adduser <username> : To create a new user, and by default the new user will not have any permission to execute sudo command

step-3) passwd <username> : To create the password for that particular username

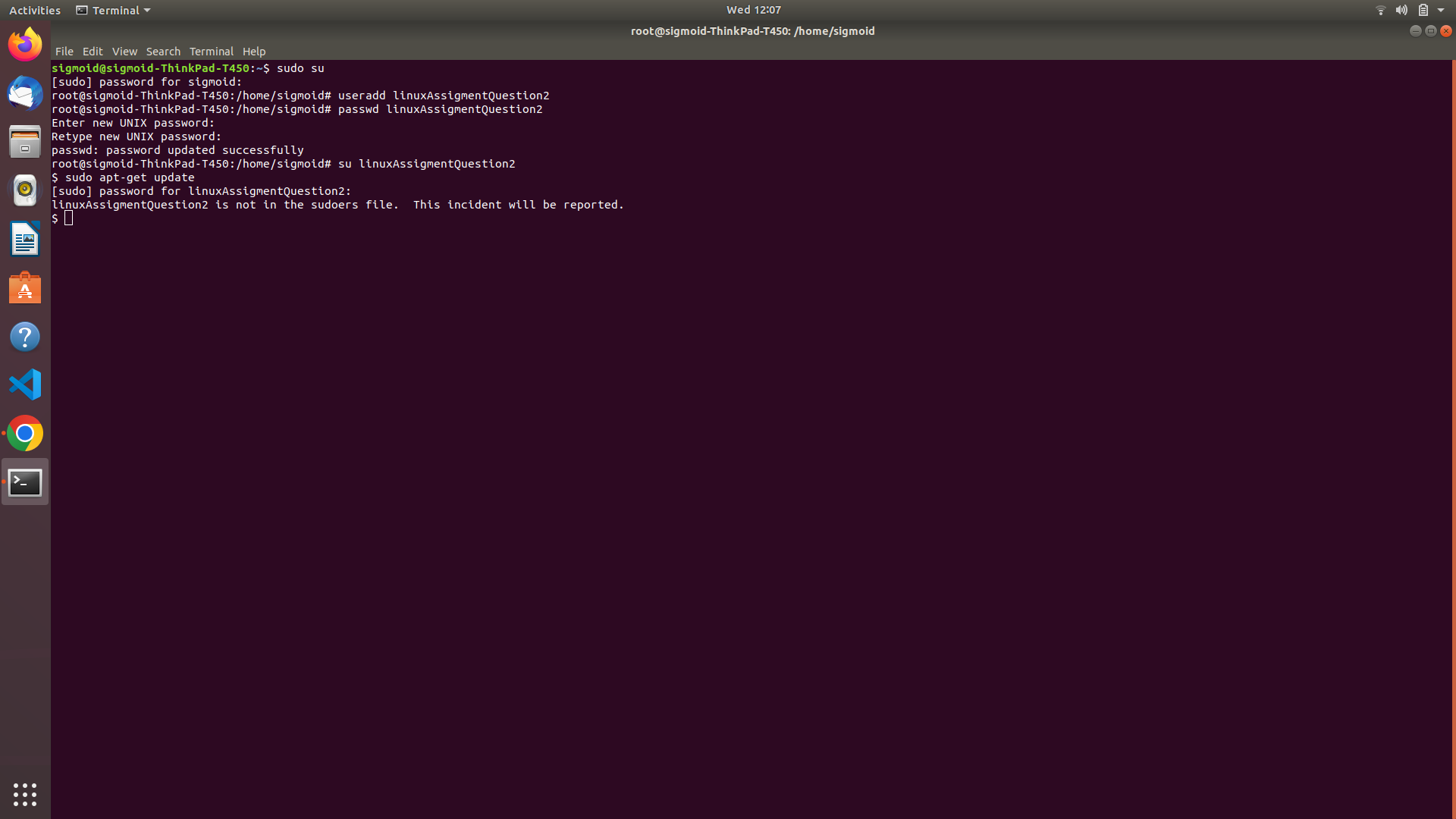
step-4) sudo nano /etc/sudoers : The sudoers file is a file Linux and Unix administrators use to allocate system rights to system users.

in sudoers file we should give sudo permissions to new users

step-5) <username> ALL=(ALL:ALL)ALL : This will permit the new user to execute sudo commands

step-6) sudo su <newuser> : Switch to new user

step-7) sudo apt-get update : This command is used to execute and check the sudo permissions are given



1. **Configure your system in such a way that when a user type and executes a describe command from anywhere of the system it must list all the files and folders of the user's current directory.**

step-1) umask : This will return 0002,Where the last three digits represent the octal values of the umask for a file .

The default value of newly created file is 666 ; that gives (-rw-rw-rw-) read write permissions to a file

Then the default value(666) gets substracted with umask(002) and result will be (666-002=664)

Then permission (664) will be allocated to the file i.e(-rw-rw-r--) . so to cofigure our system to not to do any activity

step-2) umask 0666 : Changing the value of umask from 0002 to 0666

So now the newly created file will have (666-666=000) 000 permissions i.e (---------). So there will be no permission

to do any activity

step-3) To cofigure system such that the newly created folder should not have any permission

step-4) umask 0777 : The default permissiom for newly created folder will be (777) i.e (rwxrwxrwx) . After substracting umask

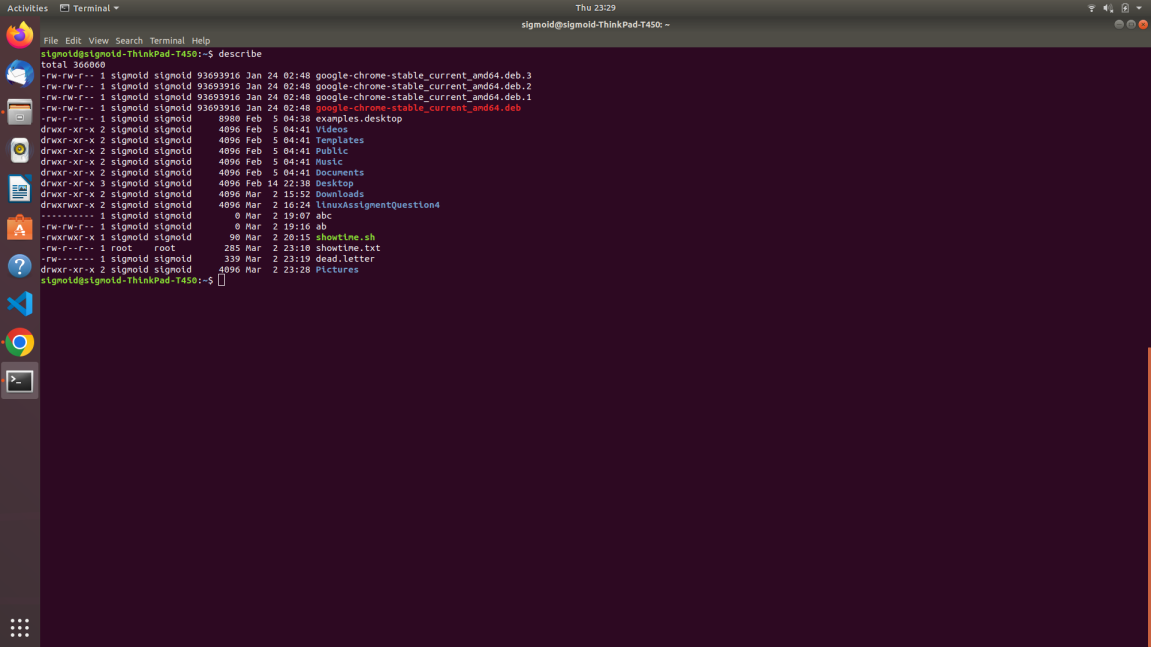
value(002) they finally have (775)

i.e (drwxrwxr-x) for created folder

step-5) So after changing umask value to 0777 , now if the user creates any folder then the permission for that folder will be given by

(777-777)

i.e (---------) . So there will be no permission to do any activity



1. **Users can put a compressed file at any path of the linux file system. The name of the file will be research and the extension will be of compression type, example for gzip type extension will be .gz.**

**You have to find the file and check the compression type and uncompress it.**

gzip type extension will be .gz.

step-1) find /home/sigmoid -iname gzip.gz -exec gzip -df {}\;

\* find : is a command name

\* /home/sigmoid : is to specify from where to start searching

\* -iname : name is to search by name and -i for ignoring case sensitiveness

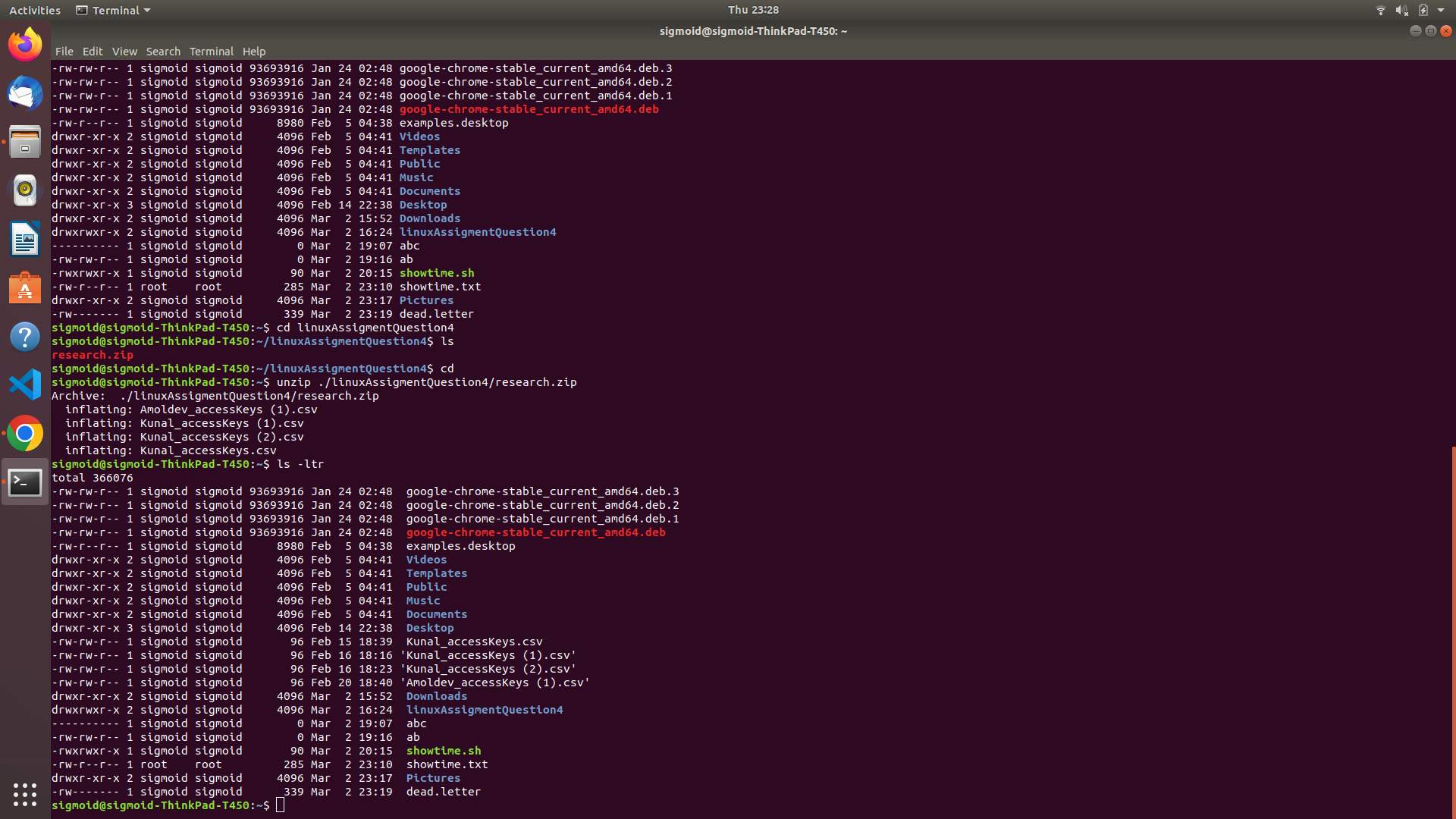
\* gzip.gz : name of the compressed file

\* -exec : this will execute the command to its right only if the command to its left results 0

\* gzip -df : used to unzip a zipped file using gzip

\* {} : placeholder,that holdes the name of a file to unzip

\* \; : to terminate the -exec command

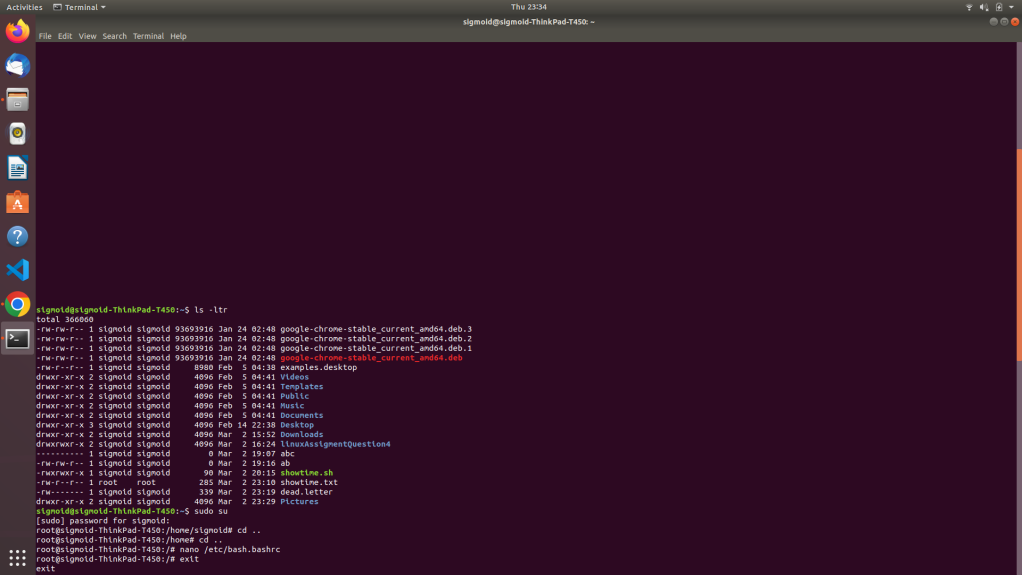


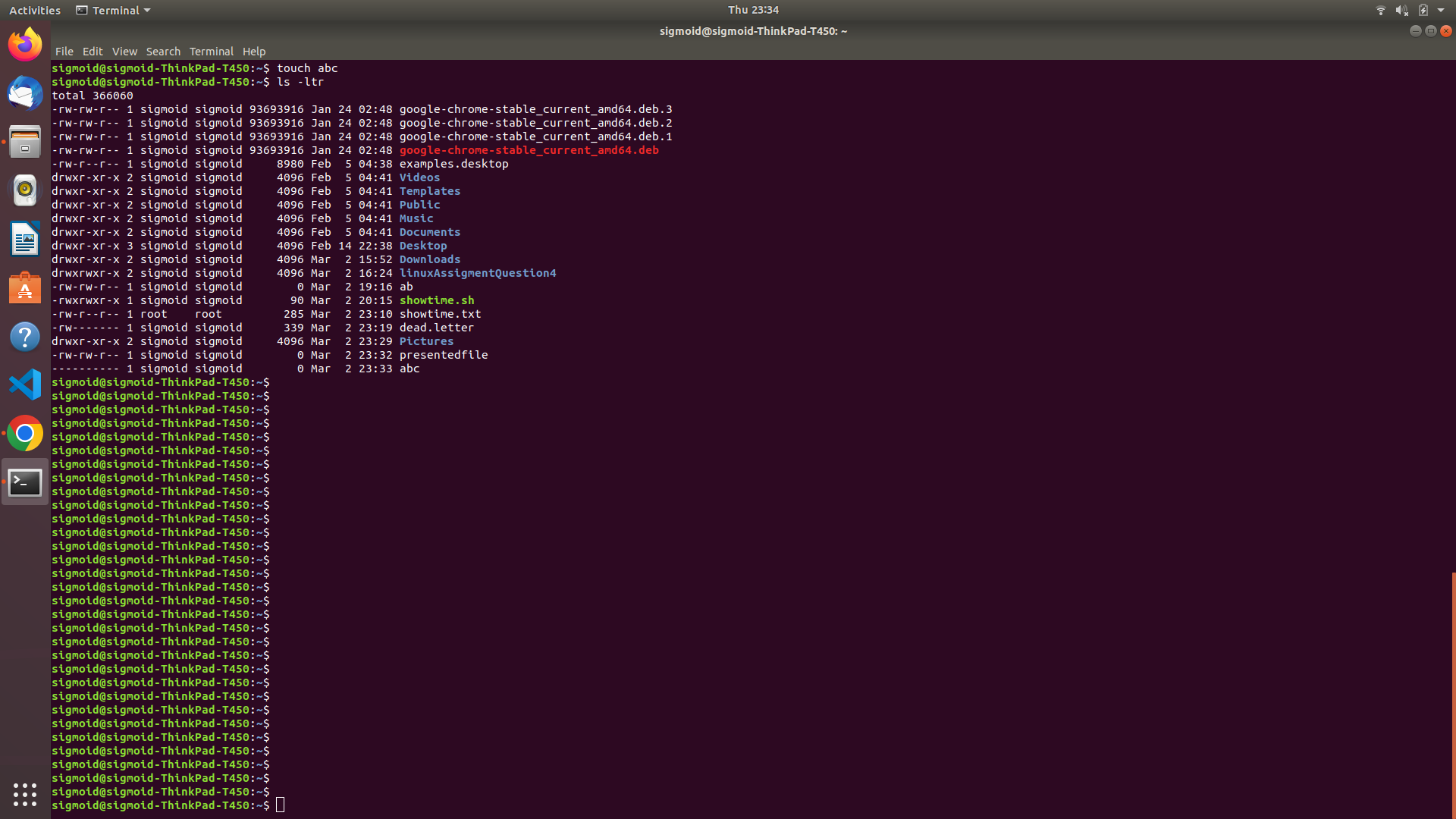
1. **Configure your system in such a way that any user of your system creates a file then there should not be permission to do any activity in that file.**

step-1)--> First go to the root. cmd-"sudo su"

step-2)--> Then go to the .bashrc file and open in editor. cmd-"nano /etc/bash.bashrc

step-3)--> Afterthat at the bottom of the file write "umask u-rwx,g-rwx,o-rwx" and save that file.

* 



1. **Create a service with the name showtime , after starting the service, every minute it should print the current time in a file in the user home directory.**

step-1)--> First we have to create a file using. cmd-"touch filename.sh"

step-2)--> Then we will write a script inside that to write you can use nano or vi editor. cmd-"nano filename.sh" or ex-"vi filename.sh"

script:-

#!/bin/bash

while true

do

date >> /home/sigmoid/showtime.txt

sleep 30

done

step-3)--> Afterthat we will convert that file into executable file. cmd-"chmod +x filename.sh"

step-4)--> Then we will create a service file. cmd-"touch filename.service"

step-5)--> Now we will write that executable filepath into this service file.

script:-

[Service]

ExecStart=/home/sigmoid/showtime.sh

step-6)--> Afterthat we will move that file to '/etc/systemd/system' path. cmd-"mv filename.service /etc/systemd/system/"

step-7)--> Then to start the service. cmd-"sudo service servicename start"

