

# CAOS

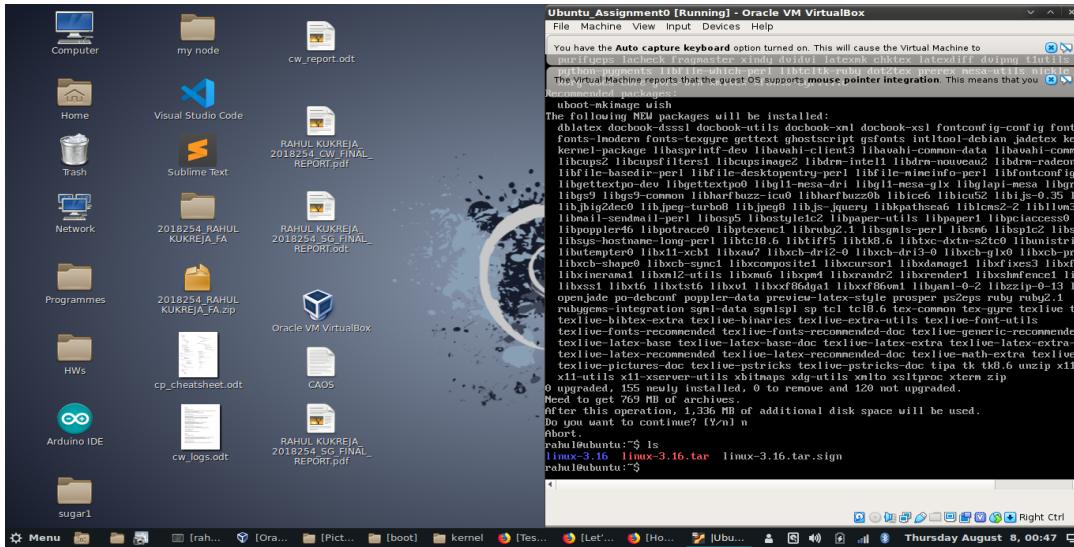
## ASSIGNMENT 0 PART 1

RAHUL KUKREJA - 2018254

Firstly, I installed Ubuntu 14.10 server mode into the virtual box.

Then, installed linux kernel version 3.16 and then unzipped it into the current directory.

Here is the screen-shot of my current status :

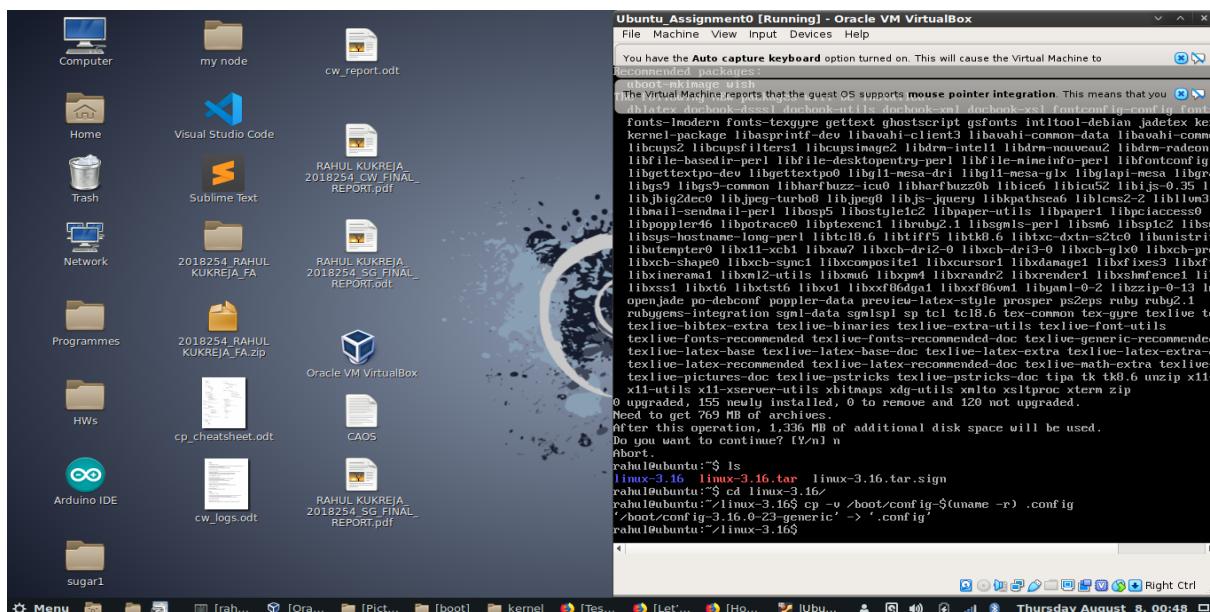


Then, I changed my current directory to the unzipped kernel directory : linux-3.16/

After this I copied the config file from the current ubuntu 14.10 distribution into the current directory i.e linux-3.16 using the command -

```
$ cp -v /boot/config-$(uname -r) .config
```

This command copied the config file (where uname -r denotes the current kernel version – the ubuntu kernel I'm working on) into the linux-3.16/ directory and saved it as .config.



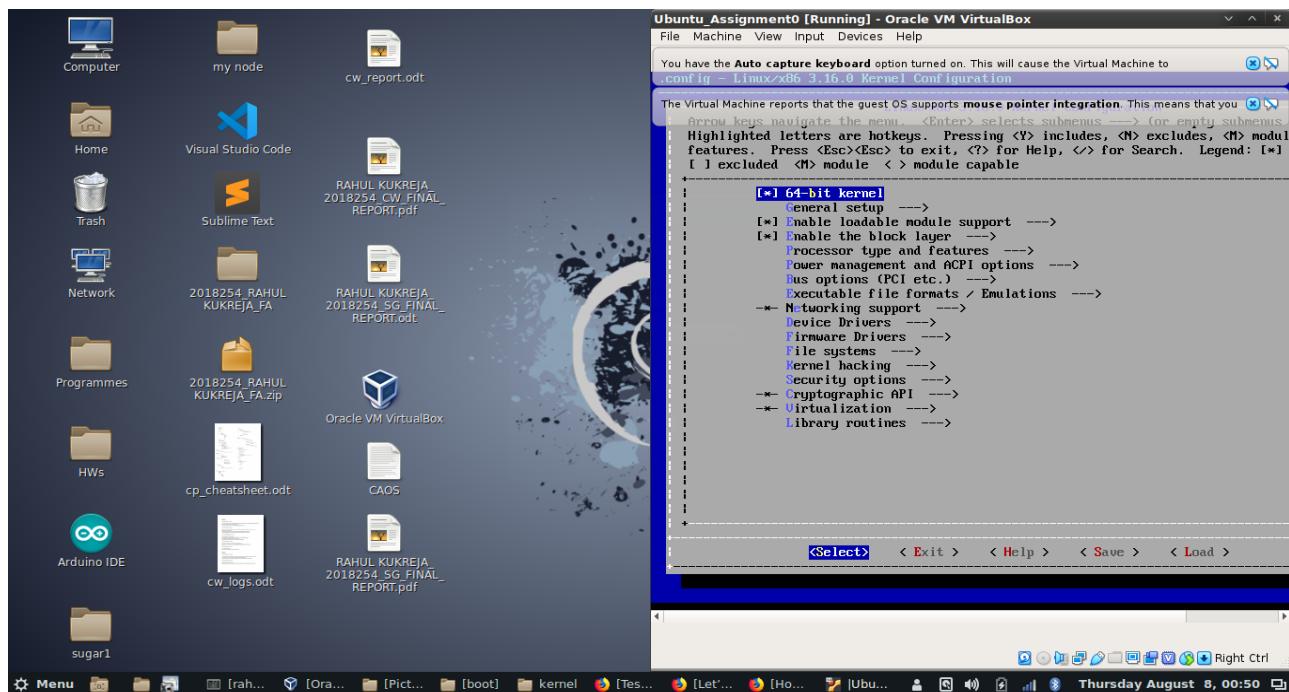
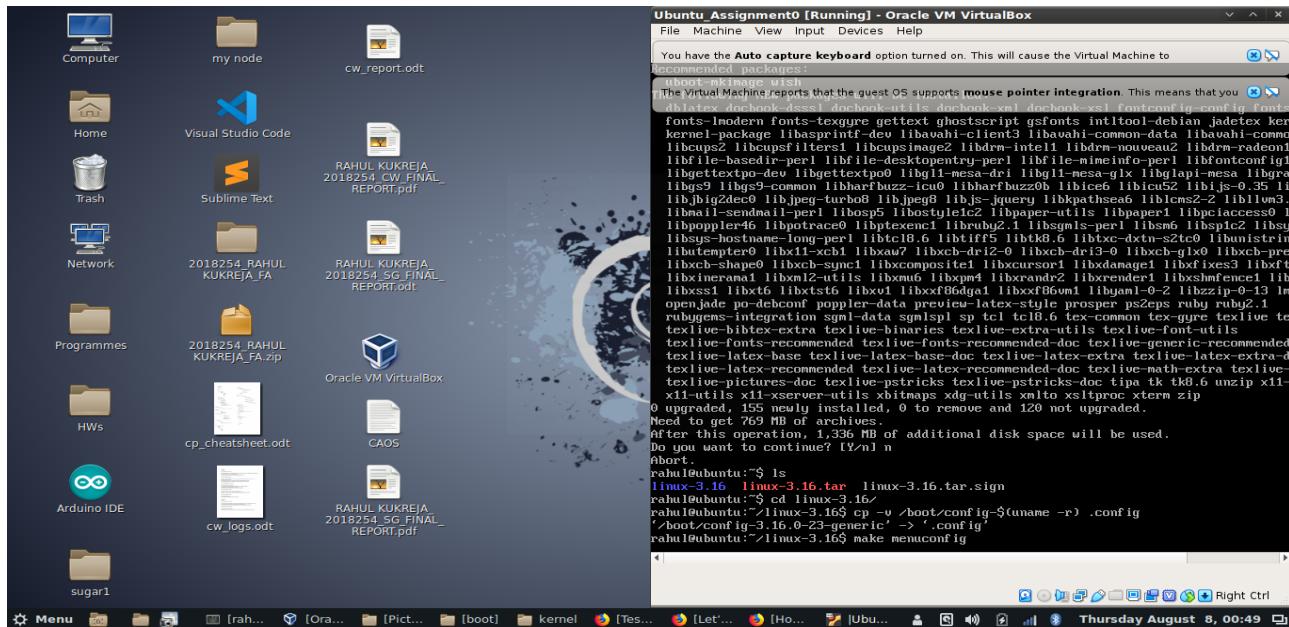
Then, to make append our name(as a customized string) into kernel version, we need to go into the configuration menu and add it to the version from there.

Hence, to enter the configuration menu we need to enter the command:

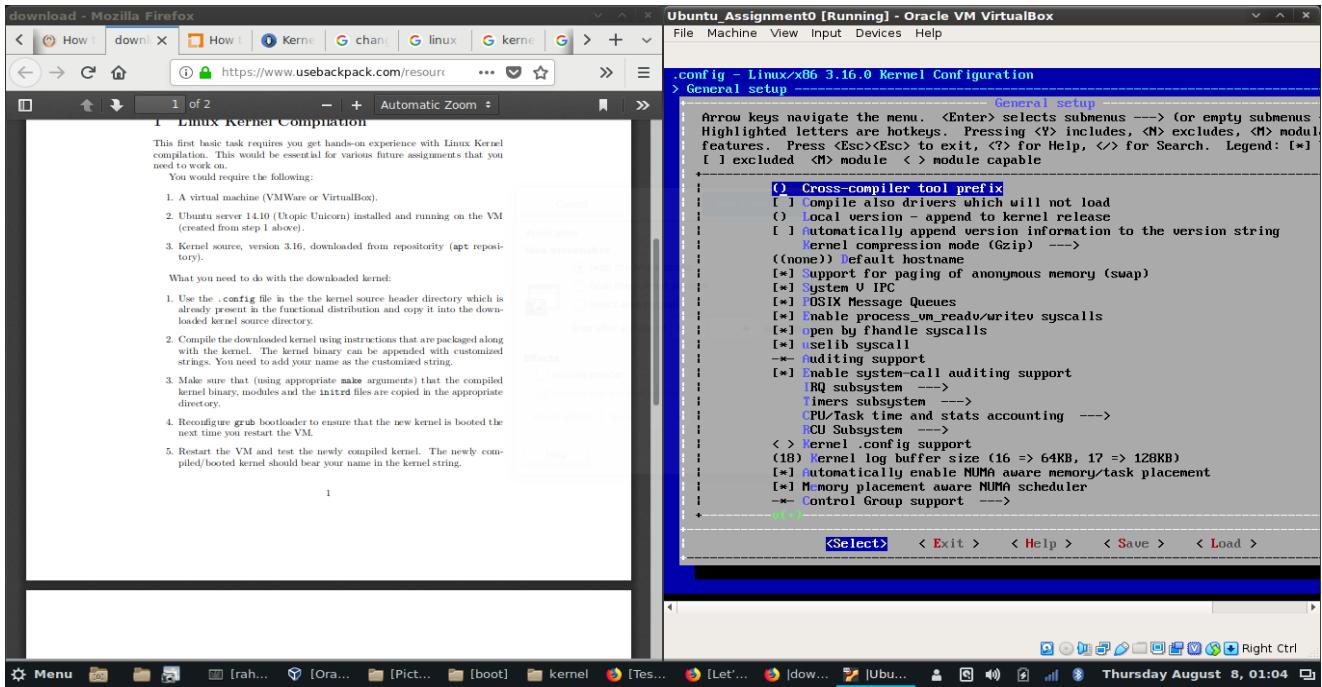
**\$ make menuconfig**

After we type this command, we'll enter the menu from where we can change various configuration settings of the kernel that we wish to compile.

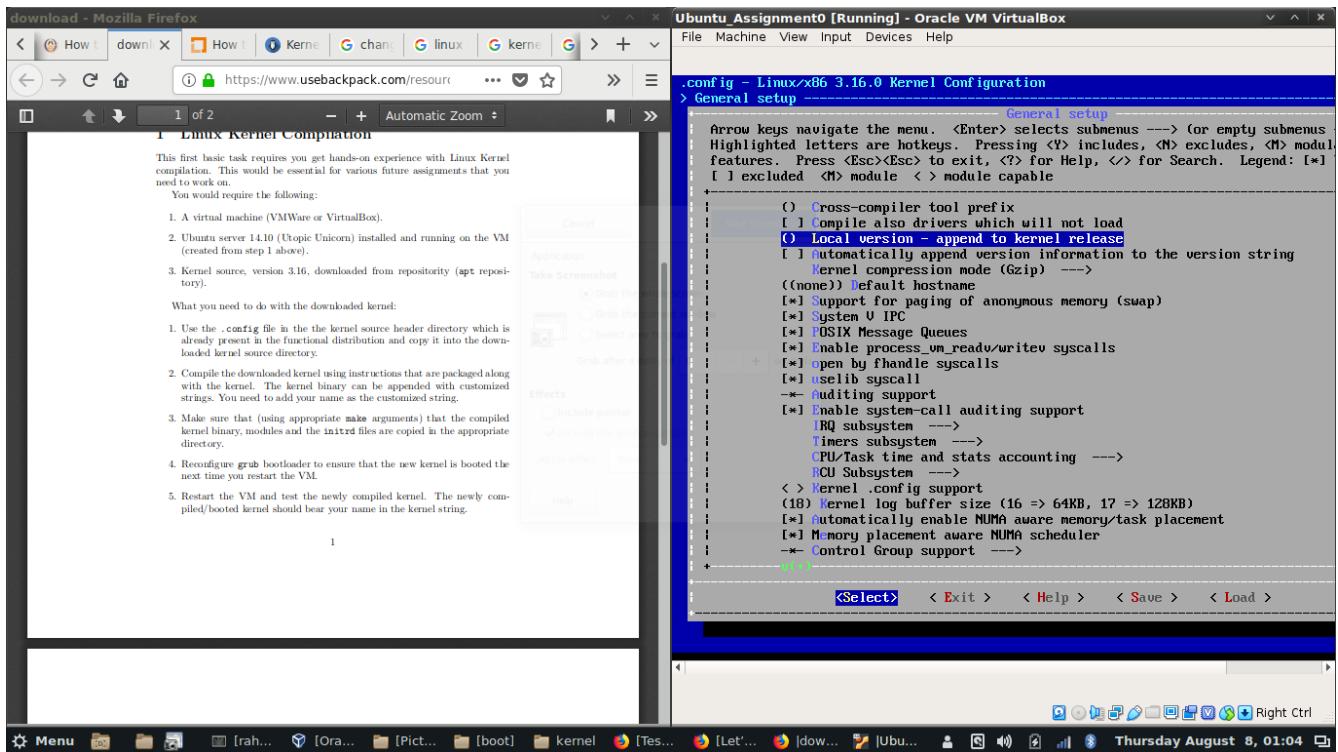
In other words, we can decide or change the configurations of our new kernel such as – what drivers to include and what to exclude and numerous others settings.



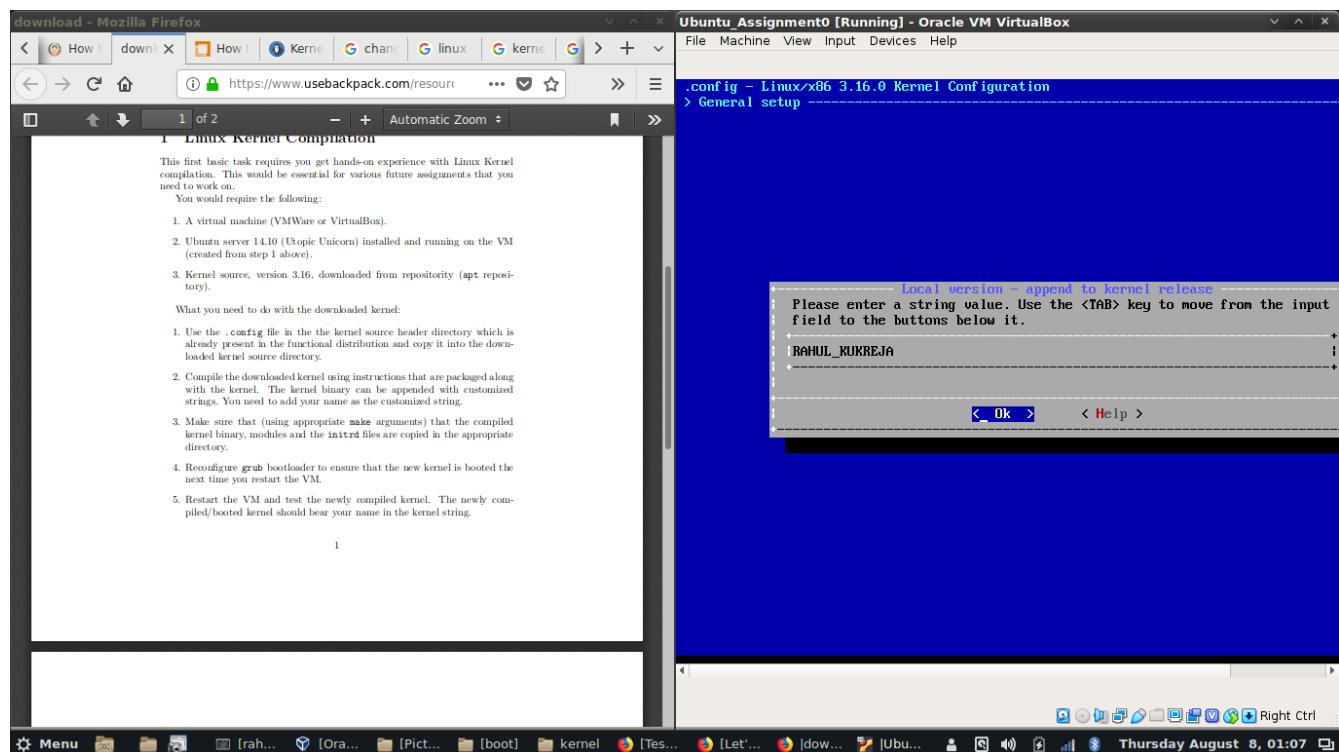
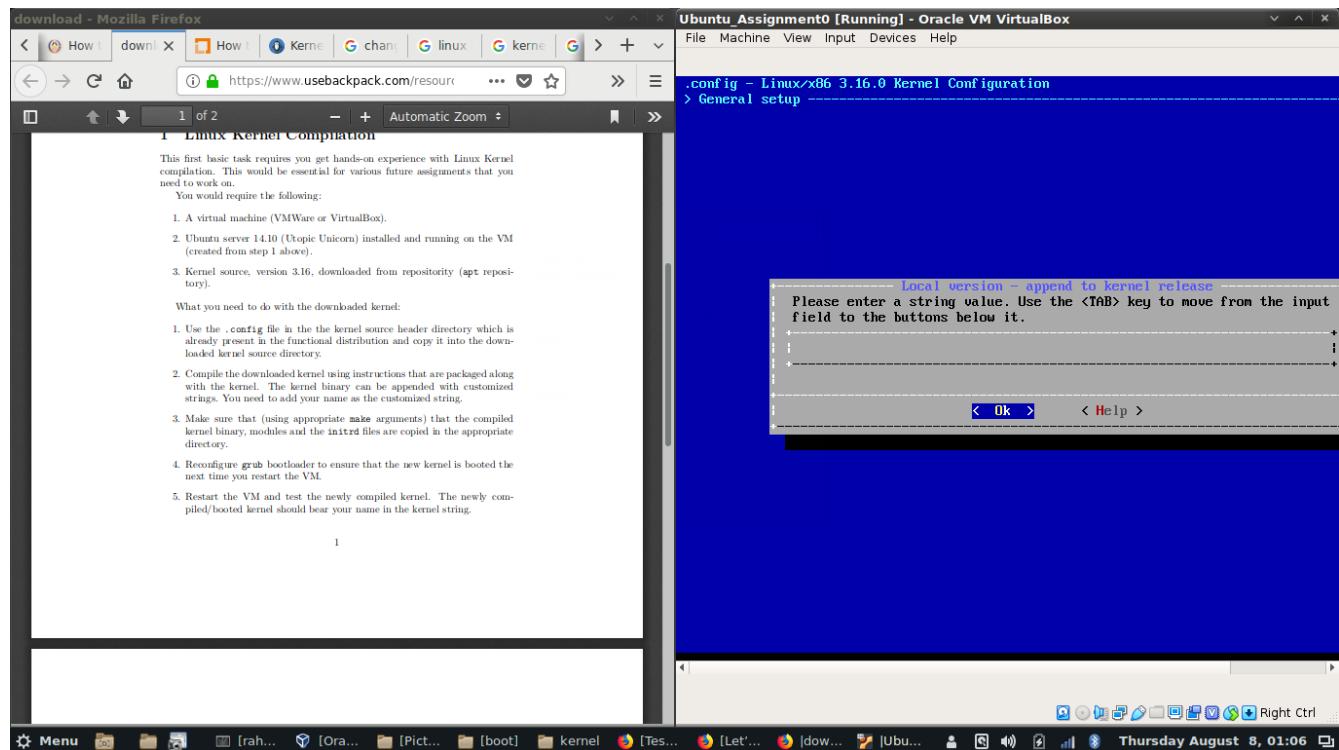
Once we have entered the configuration menu, we'll go to **General Setup** -->



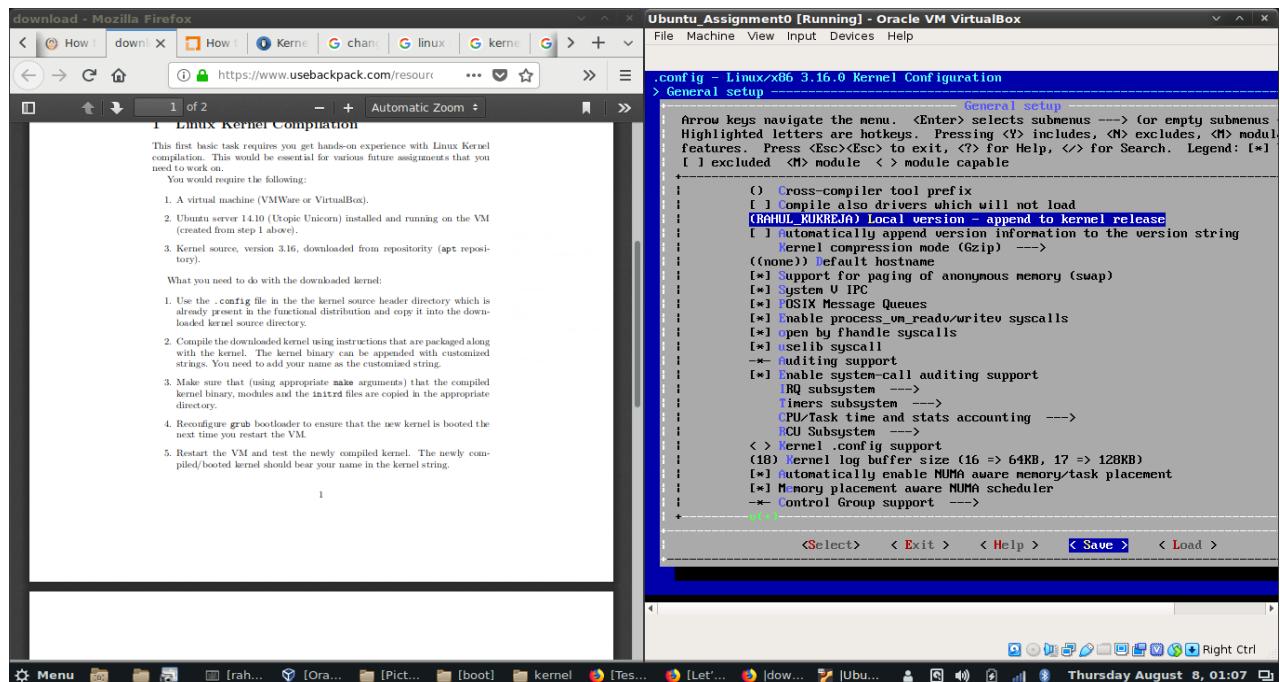
Then, we'll go to the option **Local version – append to kernel release**



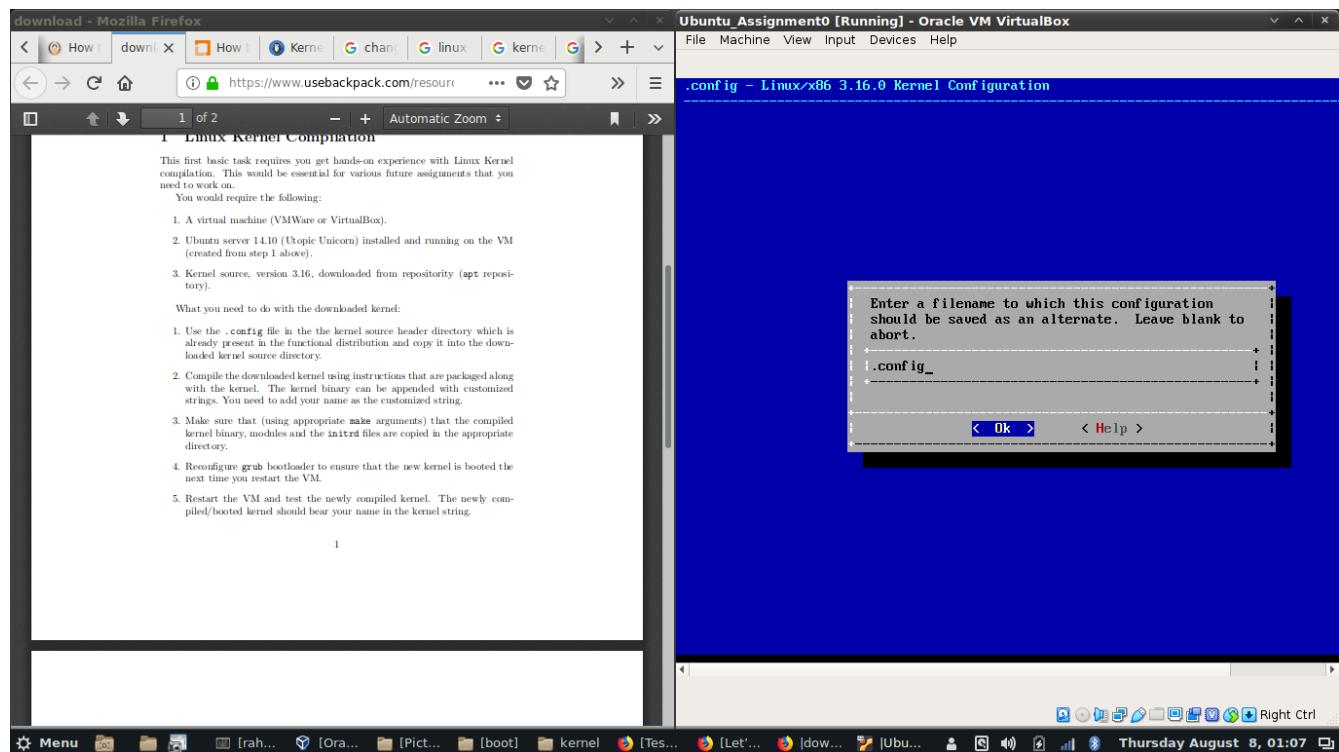
After this step, we'll add our name as a string (**RAHUL\_KUKREJA**)



Following this step, when you press OK,  
You'll be able to see your name on the **General Setup** screen.



Hit **Save** and save the changes to the **.config** file.

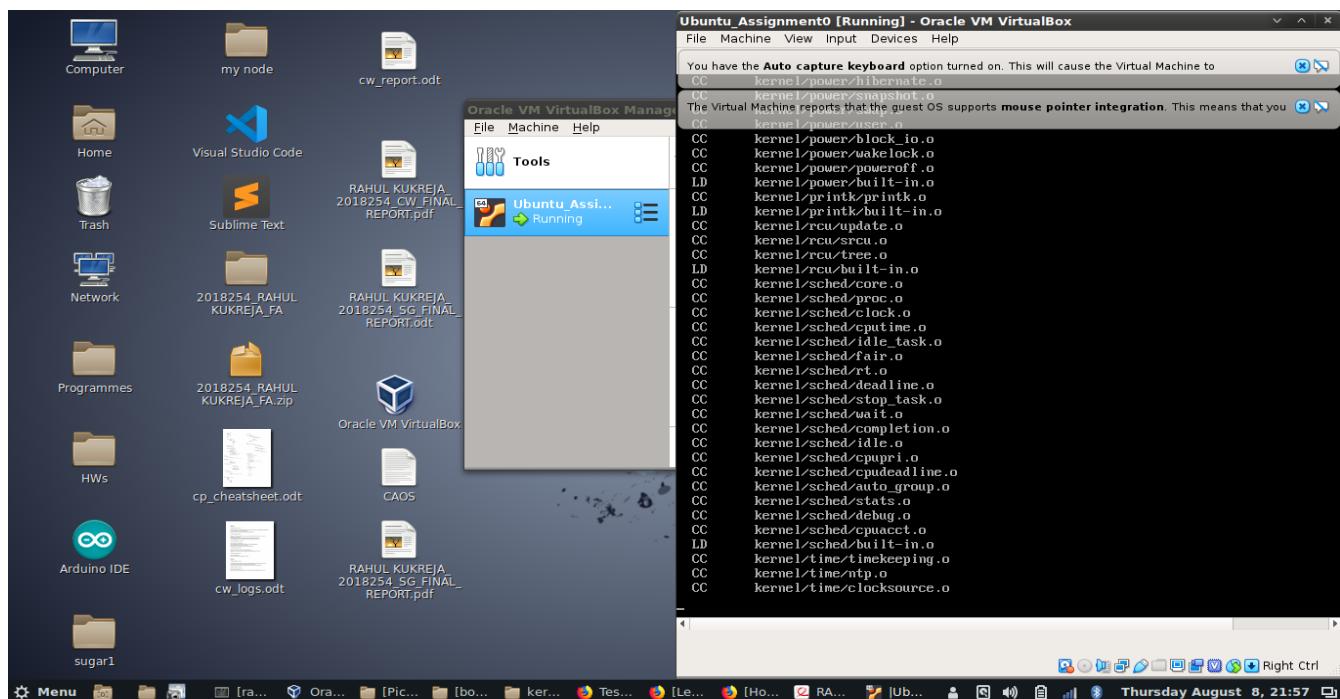
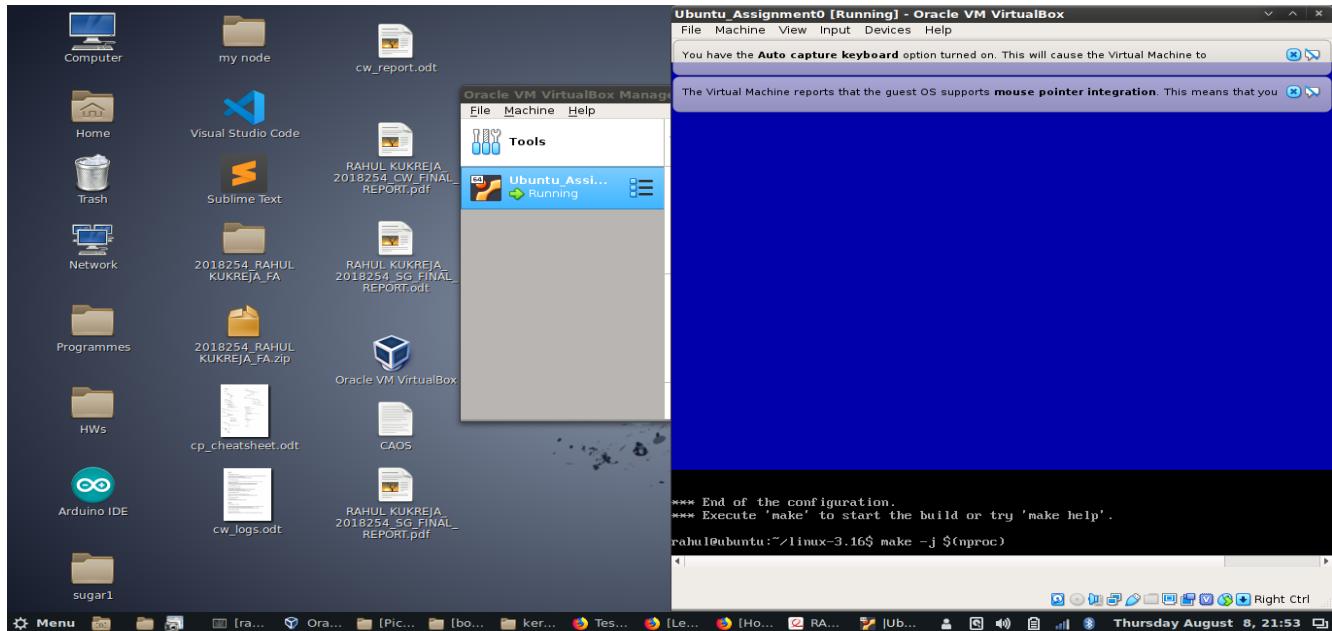


Once, we have completed the above steps.  
It's time to actually start **compiling the kernel**.

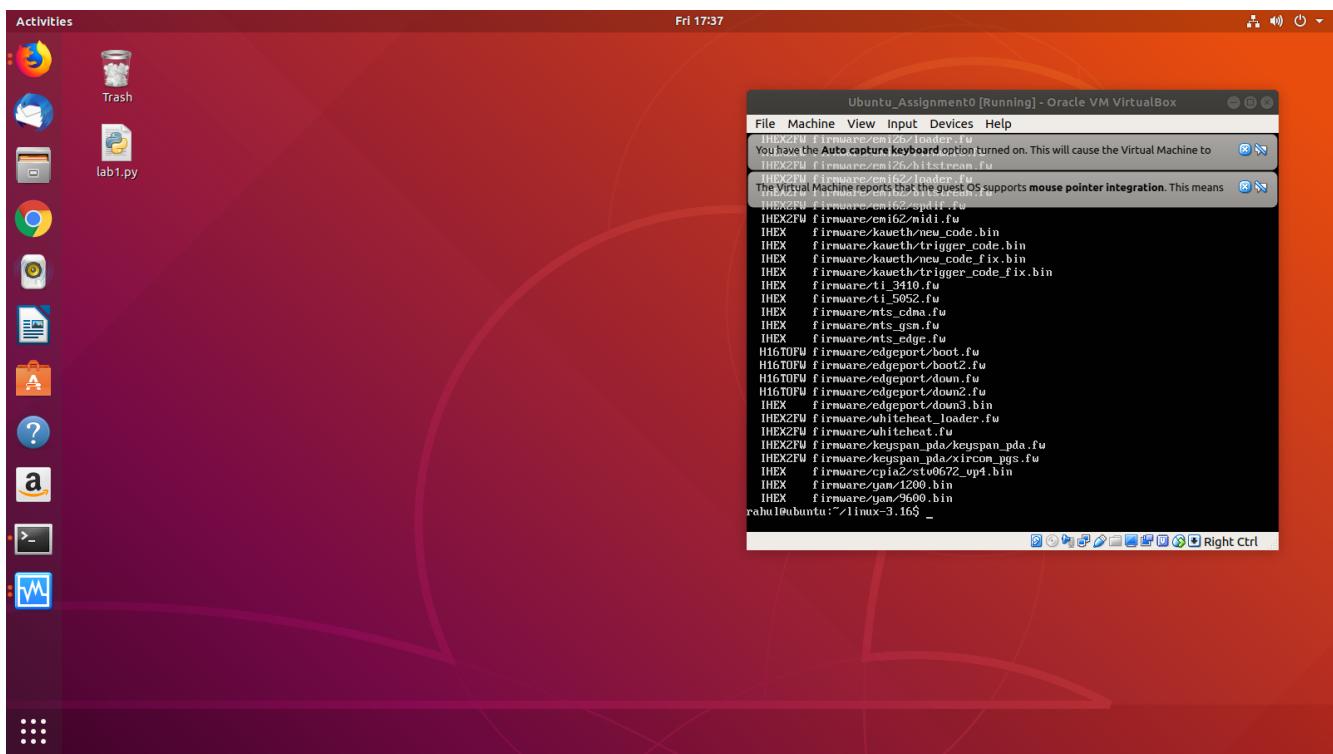
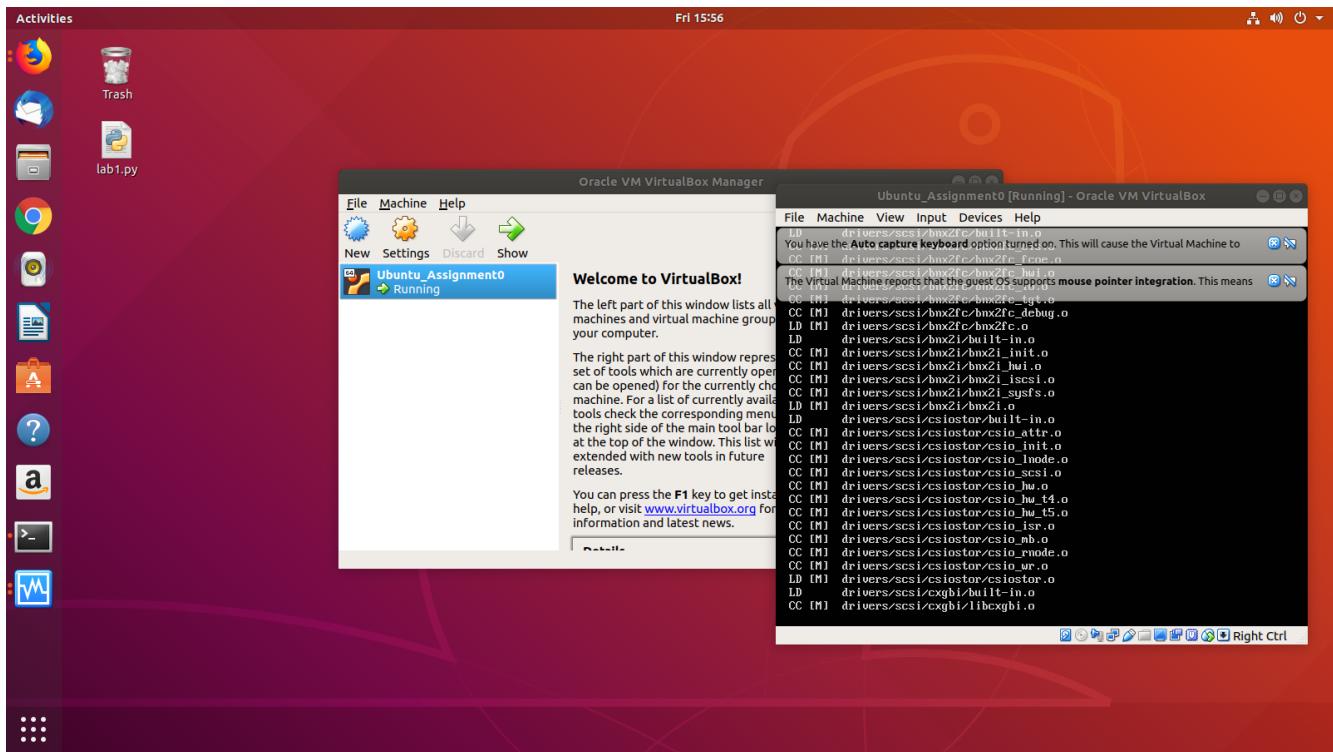
In order to start compilation, we'll type the command -

```
$ make -j $(nproc)
```

Here, -j \$(nproc) will fasten up the **make** process by giving the process all possible cores.  
**nproc – total number of cores available**



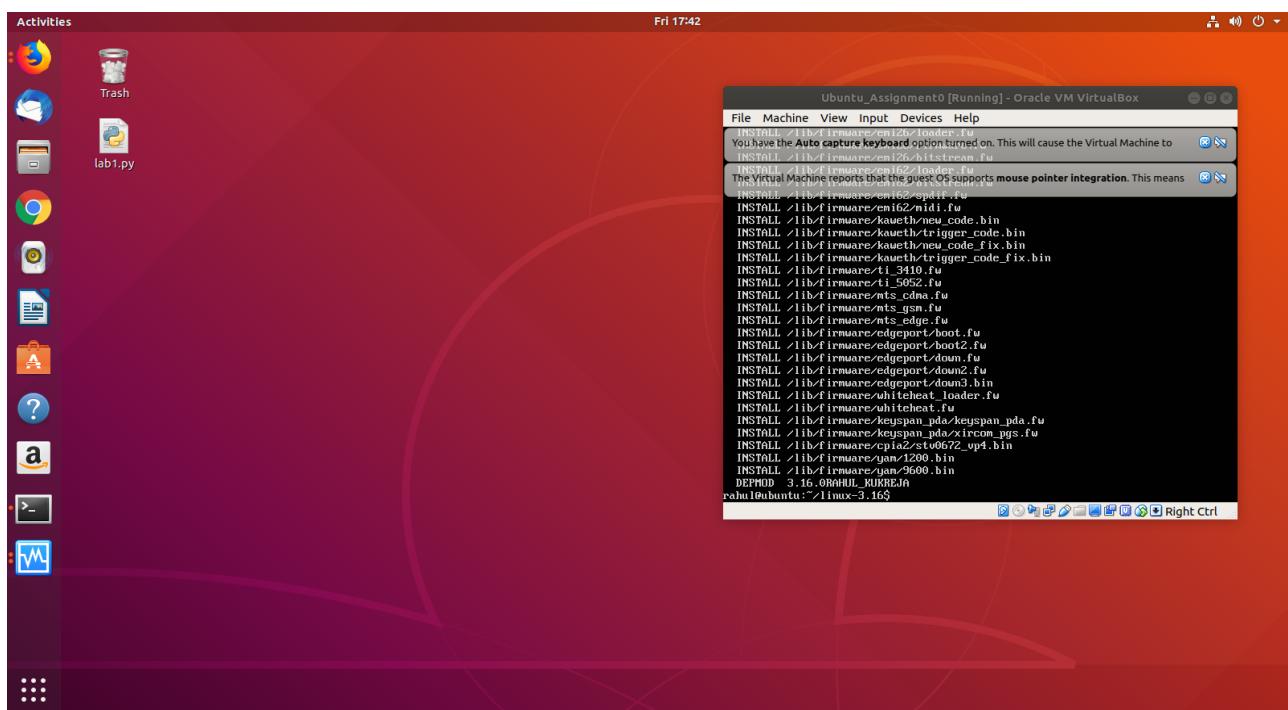
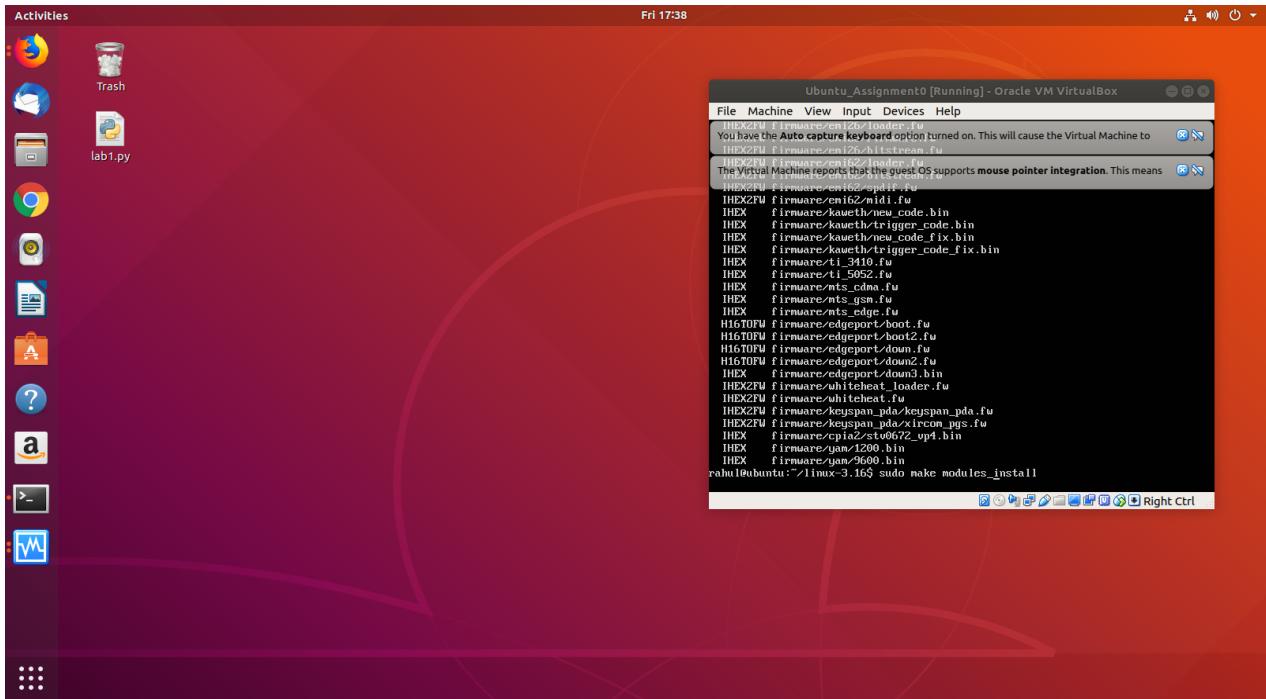
The compilation process continues and takes a considerable amount of time.



After the compilation is finished, we'll have to install some necessary modules for the complete compilation of the kernel using the command,

**\$ make modules\_install**

This will automatically install all the required modules and files such as some driver/ and firmware/ file.

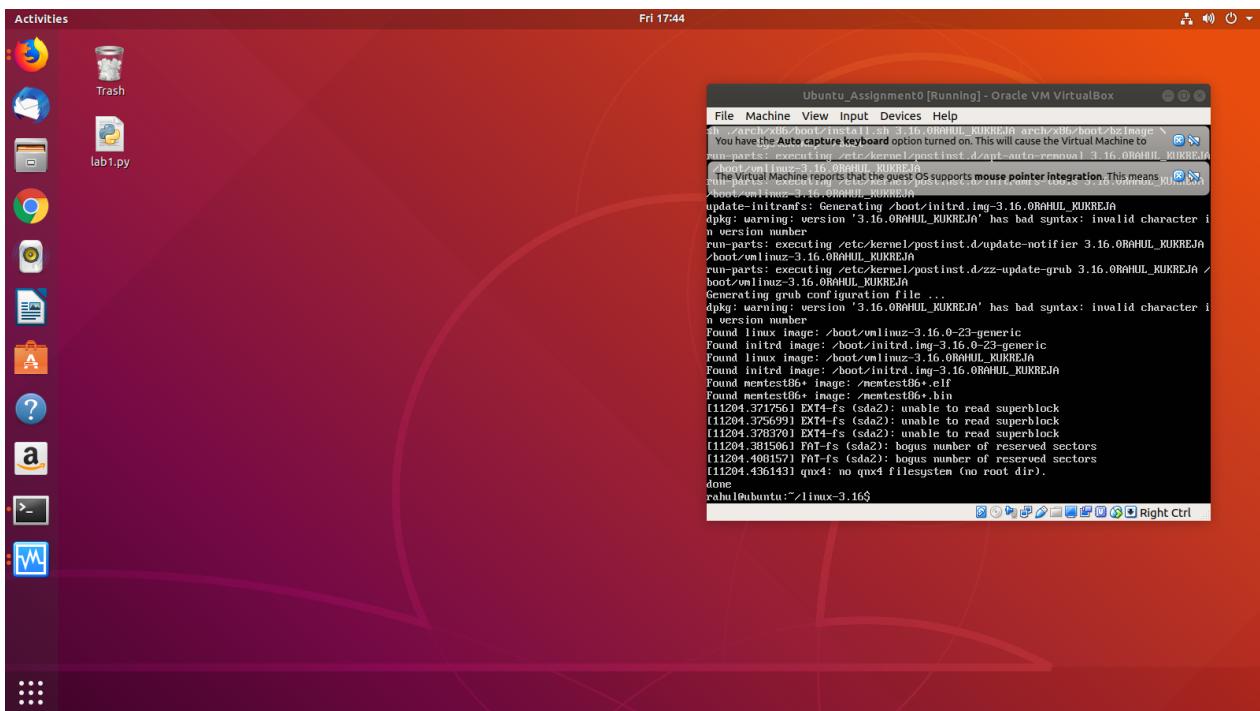
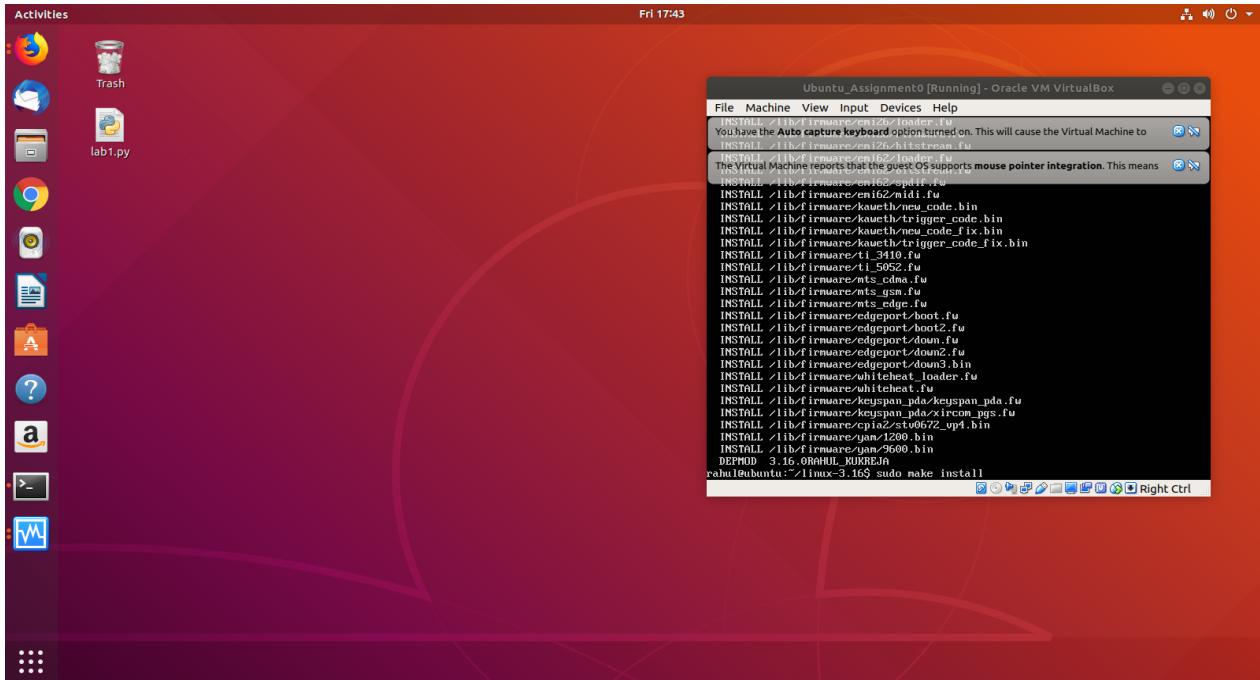


Finally, to complete the installation process, we'll type the command -

**\$ make install**

This will finish up the compilation process.

**The command will automatically add our new compiled kernel into the boot sequence.**



## Re-configuration of grub -

Now, in order to make the compiled kernel boot automatically on reboot.  
We'll have to configure the grub using the following commands :

```
$ grep gnulinux /boot/grub/grub.cfg
```

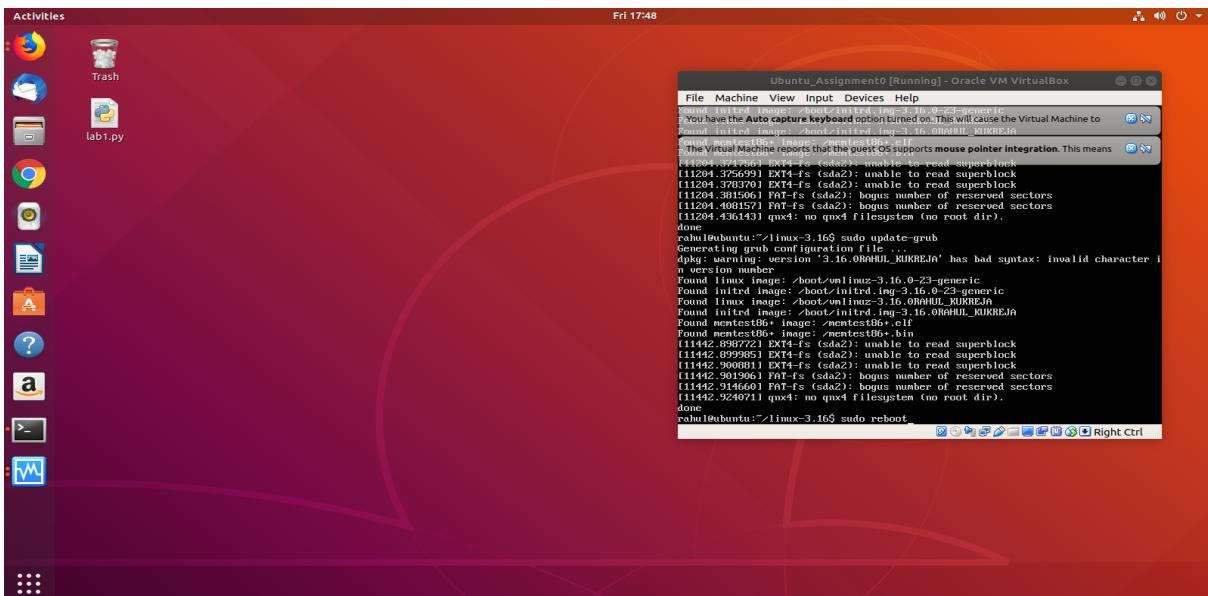
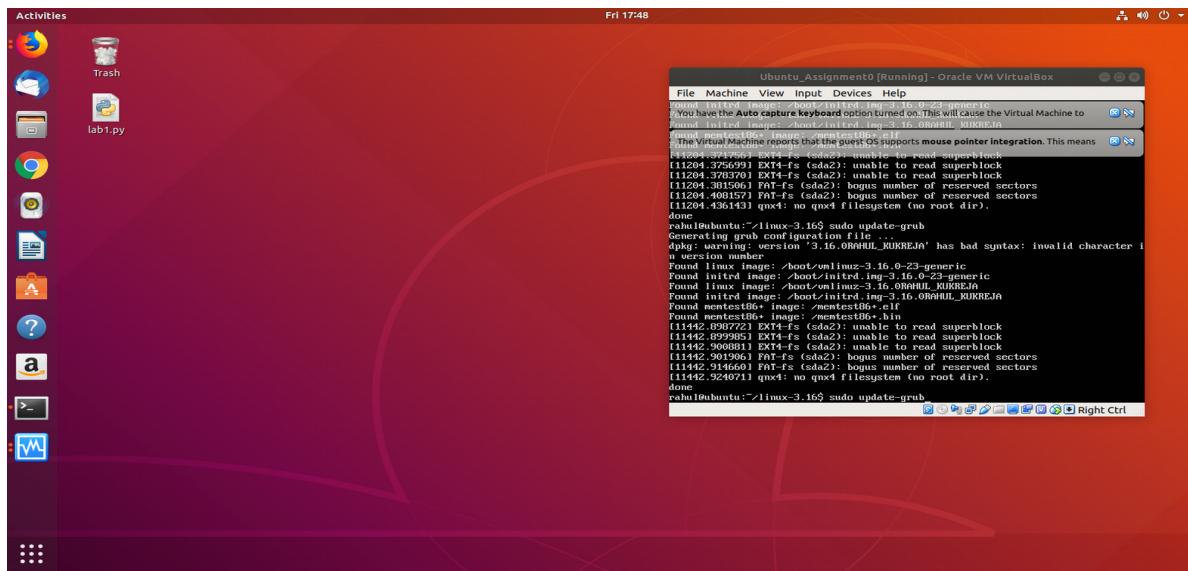
This will show a list of kernels.

**id** of a kernel is the index of that particular kernel in the list.

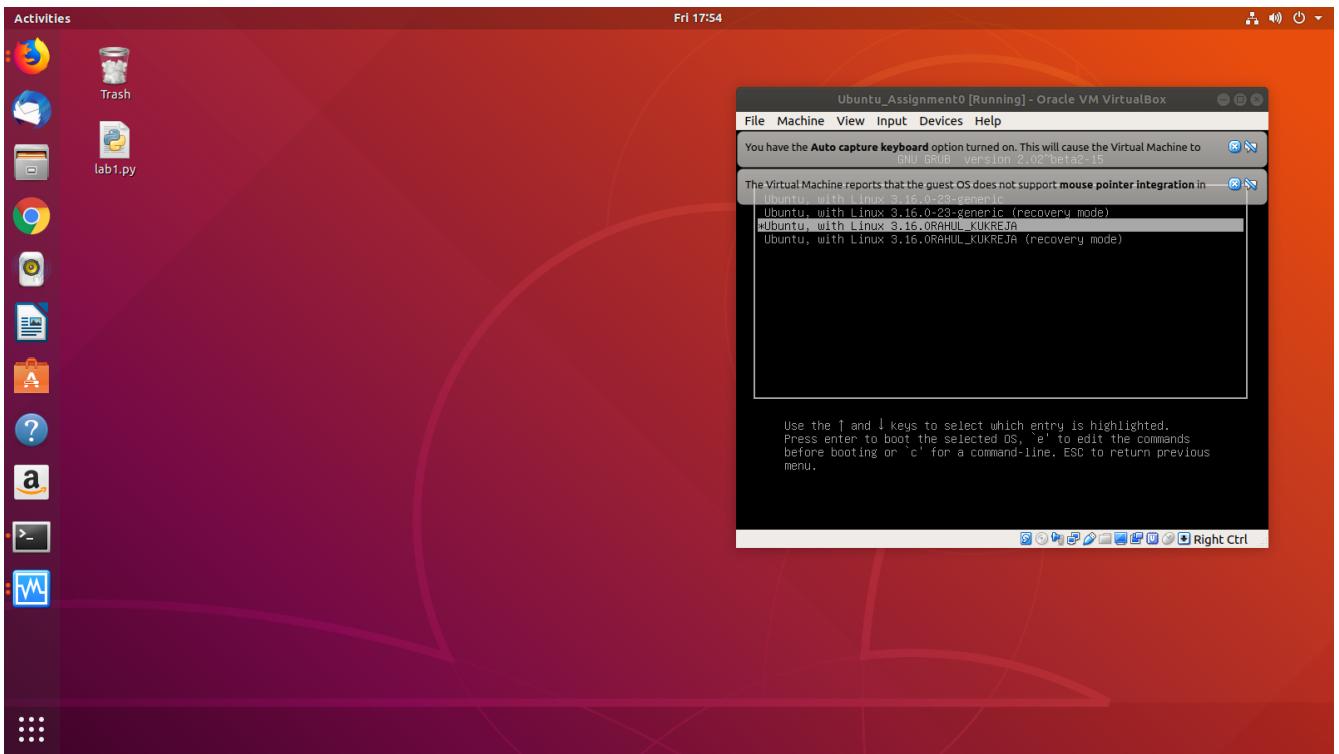
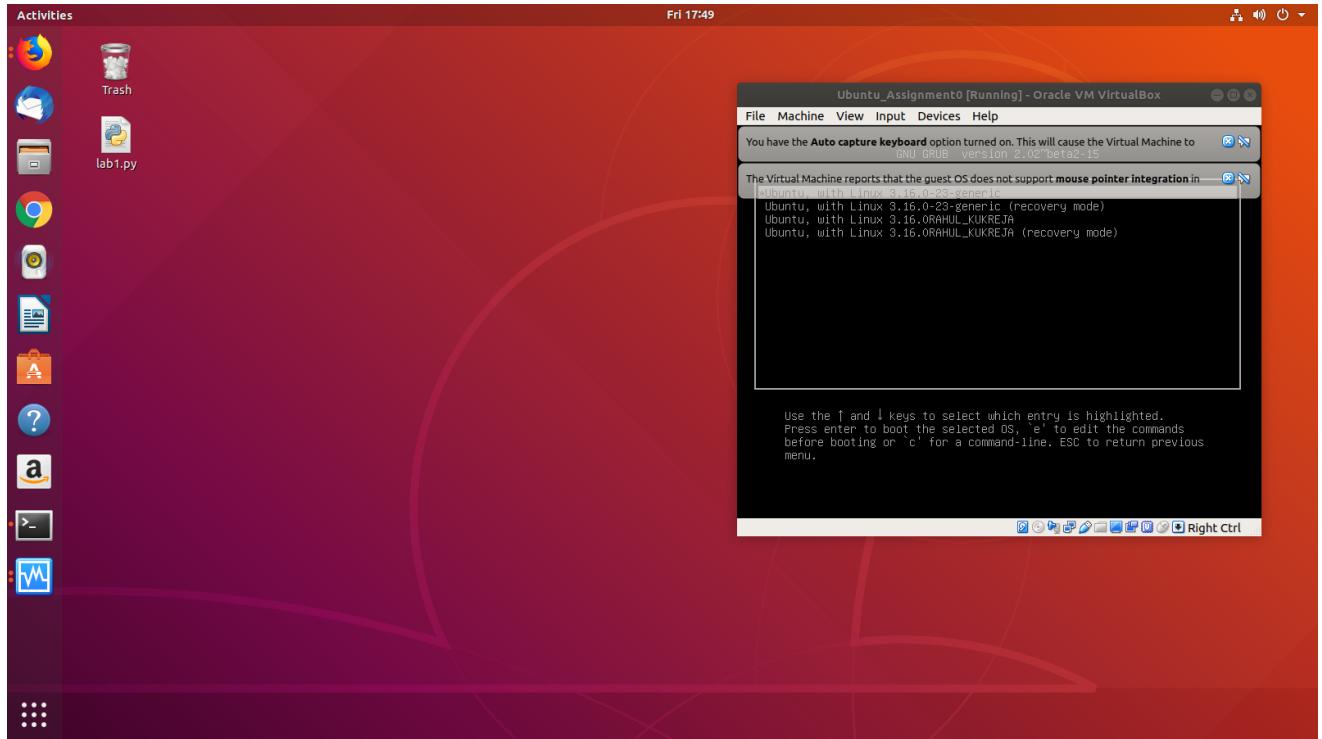
The index of our compiled kernel is 2. Hence, we'll type the command :

```
$ grub-set-default 2
```

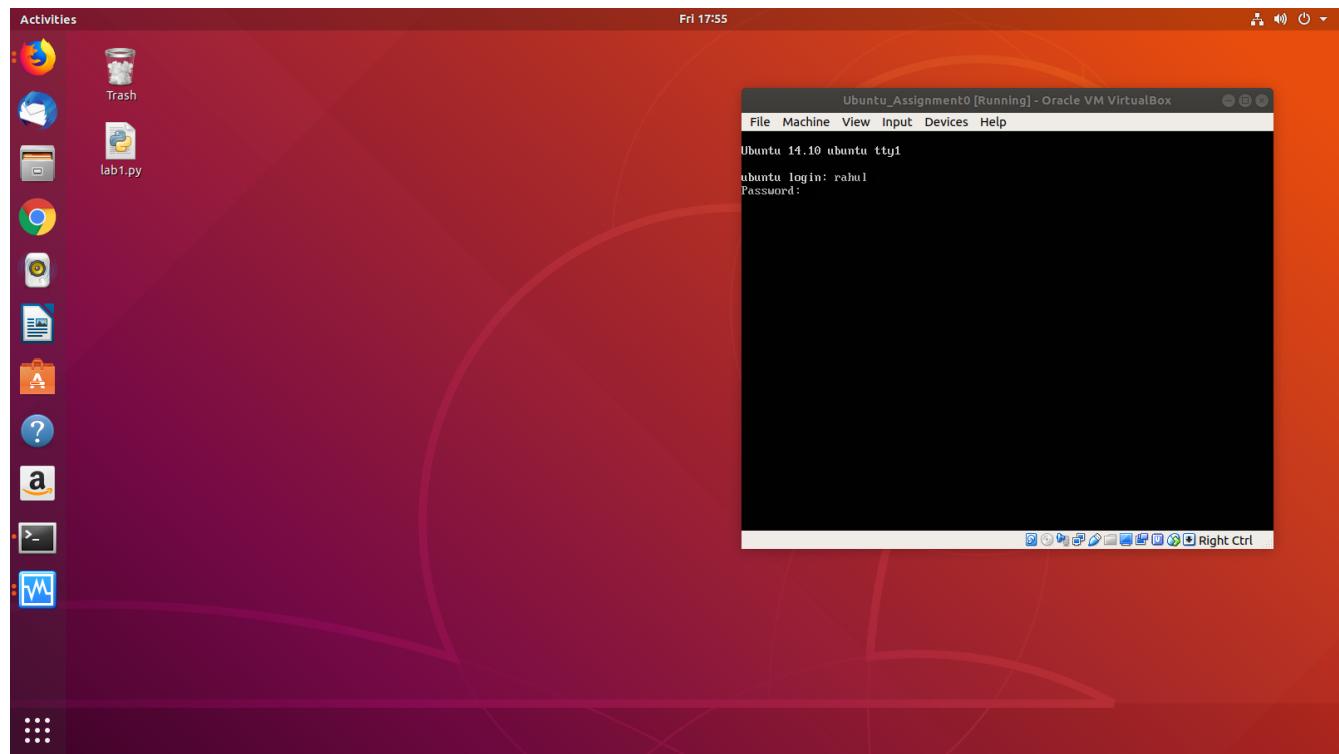
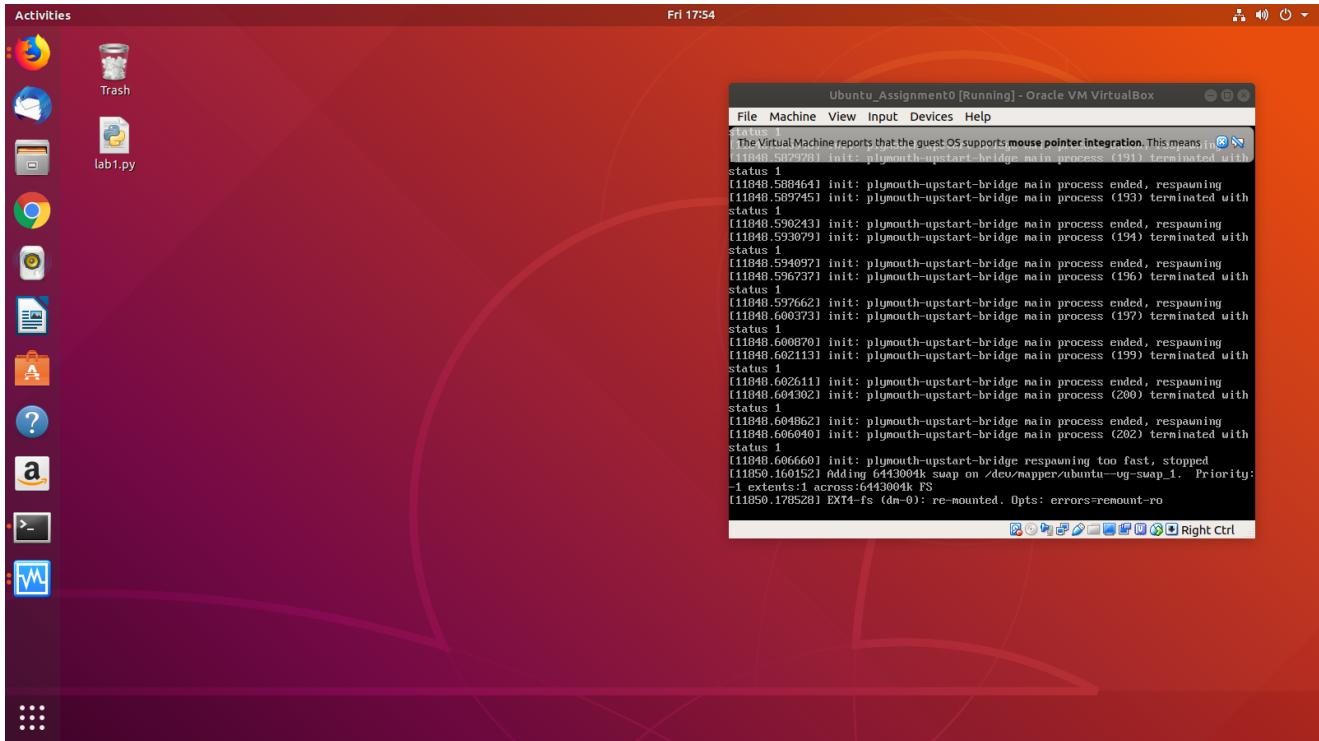
This will set our kernel as the default kernel on **reboot**.



On rebooting,  
we'll see our compiled kernel – **Linux 3.16.0RAHUL\_KUKREJA** automatically in the boot sequence  
on the grub menu.



Now, boot into your compiled kernel.



Finally, to ensure that we have correctly compiled and booted into our kernel, we'll check the kernel version of our booted kernel which should show our name (**RAHUL\_KUKREJA**) in the version name.

To check that, type the command -

```
$ uname -a
```

This command shows the current linux kernel version.

