

CS3003D: OPERATING SYSTEMS

ASSIGNMENT-1

Name: Rose S Jose

Roll No: B190839CS

Batch: A

PROBLEM STATEMENT

Download the latest stable Linux kernel from kernel.org, compile it, and dual boot it with your current Linux version. Your current version as well as the new version should be present in the grub-menu.

METHODOLOGY

1. Ensure that you have a Linux version on your system.
2. Obtain the latest stable Linux kernel from kernel.org.
3. Install all development dependencies.
4. Configure and compile the kernel.
5. Install the compiled kernel and add grub entry.
6. Reboot the system.

EXPLANATION

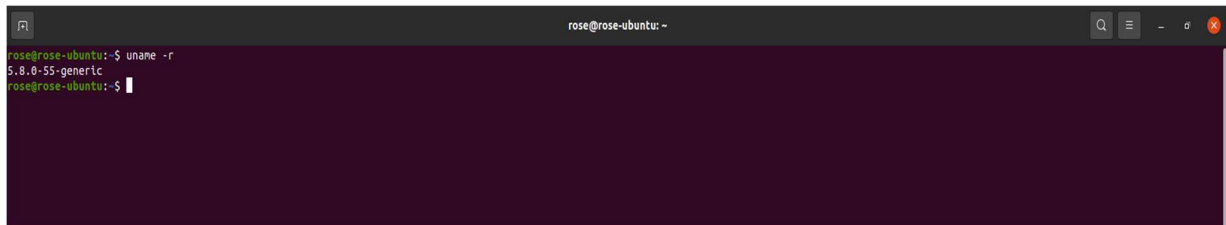
The kernel is the essential centre of a computer operating system (OS). It is the core that provides basic services for all other parts of the OS. It is the main layer between the OS and hardware, and it helps with process and memory management, file systems, device control and networking.

In this report steps to upgrade to the latest stable Linux kernel (5.14.1) is explained.

COMMANDS:

Step 1: \$uname -r

Get the current kernel version of your Linux distribution.



```
rose@rose-ubuntu: ~  
rose@rose-ubuntu:~$ uname -r  
5.8.0-55-generic  
rose@rose-ubuntu:~$
```

Step 2: \$mkdir OS_Assg

\$cd OS_Assg

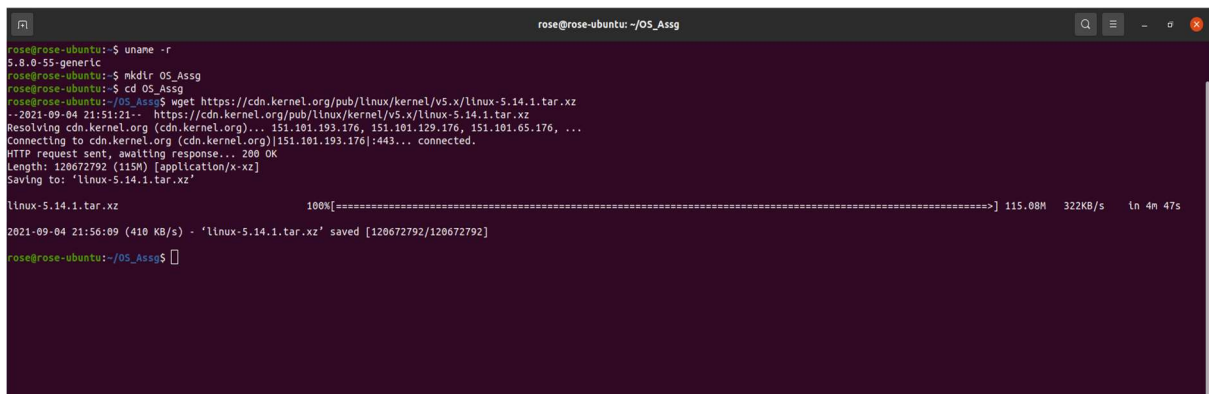
We are creating a directory name OS_Assg in HOME and change directory to OS_Assg so that all files downloaded and extracted is in a single place.



```
rose@rose-ubuntu: ~$ uname -r  
5.8.0-55-generic  
rose@rose-ubuntu:~$ mkdir OS_Assg  
rose@rose-ubuntu:~$ cd OS_Assg  
rose@rose-ubuntu:~/OS_Assg$
```

Step 3: \$wget <https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.14.1.tar.xz>

This will download the Linux-5.14.1 kernel code published on kernel.org as a tar file which is latest stable Linux kernel. The tar file is about 120.7mb.



```
rose@rose-ubuntu: ~/OS_Assg  
rose@rose-ubuntu:~$ uname -r  
5.8.0-55-generic  
rose@rose-ubuntu:~$ mkdir OS_Assg  
rose@rose-ubuntu:~$ cd OS_Assg  
rose@rose-ubuntu:~/OS_Assg$ wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.14.1.tar.xz  
--2021-09-04 21:51:21-- https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.14.1.tar.xz  
Resolving cdn.kernel.org (cdn.kernel.org)... 151.101.193.176, 151.101.129.176, 151.101.65.176, ...  
Connecting to cdn.kernel.org (cdn.kernel.org)[151.101.193.176]:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 120672792 (115M) [application/x-xz]  
Saving to: 'linux-5.14.1.tar.xz'  
  
linux-5.14.1.tar.xz 100%[=====] 115.08M 322KB/s in 4m 47s  
2021-09-04 21:56:09 (410 KB/s) - 'linux-5.14.1.tar.xz' saved [120672792/120672792]  
rose@rose-ubuntu:~/OS_Assg$
```

Step 4: \$tar -xf linux-5.14.1.tar.xz

This will extract files from the tarball archive into the folder linux-5.14.1.

```
rose@rose-ubuntu: ~/$ uname -r
5.8.0-55-generic
rose@rose-ubuntu: ~/$ mkdir OS_Assg
rose@rose-ubuntu: ~/$ cd OS_Assg
rose@rose-ubuntu: ~/OS_Assg$ wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.14.1.tar.xz
--2021-09-04 21:51:21-- https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.14.1.tar.xz
Resolving cdn.kernel.org (cdn.kernel.org)... 151.101.193.176, 151.101.129.176, 151.101.65.176, ...
Connecting to cdn.kernel.org (cdn.kernel.org)|151.101.193.176|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 120672792 (115M) [application/x-xz]
Saving to: 'linux-5.14.1.tar.xz'

linux-5.14.1.tar.xz           100%[=====] 115.08M  322KB/s   in 4m 47s

2021-09-04 21:56:09 (410 KB/s) - 'linux-5.14.1.tar.xz' saved [120672792/120672792]

rose@rose-ubuntu: ~/OS_Assg$ tar -xf linux-5.14.1.tar.xz
rose@rose-ubuntu: ~/OS_Assg$
```

Step 5: \$cd linux-5.14.1

Once done extracting the archive, the directory is changed to the extracted folder.

Step 6: \$cp /boot/config-5.8.0-55-generic .config

Before we compile the kernel, we need to make sure we configure it to specify which modules we want to be installed and which not to. Either we can configure the kernel from scratch using **make config** that starts a configure script which will ask you many questions regarding enabling or disabling modules and installing drivers or simply use the one that came with our Linux.

The command copies the config file of the existing Linux to the new config file.

```
rose@rose-ubuntu: ~/$ uname -r
5.8.0-55-generic
rose@rose-ubuntu: ~/$ mkdir OS_Assg
rose@rose-ubuntu: ~/$ cd OS_Assg
rose@rose-ubuntu: ~/OS_Assg$ wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.14.1.tar.xz
--2021-09-04 21:51:21-- https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.14.1.tar.xz
Resolving cdn.kernel.org (cdn.kernel.org)... 151.101.193.176, 151.101.129.176, 151.101.65.176, ...
Connecting to cdn.kernel.org (cdn.kernel.org)|151.101.193.176|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 120672792 (115M) [application/x-xz]
Saving to: 'linux-5.14.1.tar.xz'

linux-5.14.1.tar.xz           100%[=====] 115.08M  322KB/s   in 4m 47s

2021-09-04 21:56:09 (410 KB/s) - 'linux-5.14.1.tar.xz' saved [120672792/120672792]

rose@rose-ubuntu: ~/OS_Assg$ tar -xf linux-5.14.1.tar.xz
rose@rose-ubuntu: ~/OS_Assg$ cd linux-5.14.1/
rose@rose-ubuntu: ~/OS_Assg/linux-5.14.1$ cp /boot/config-5.8.0-55-generic .config
rose@rose-ubuntu: ~/OS_Assg/linux-5.14.1$
```

Step 7: \$sudo apt install build-essential libncurses-dev flex bison libssl-dev libelf-dev

The command will install all the required dependencies for the kernel compilation.

```
Get:62 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 DEP-11 Metadata [2,468 B]
Get:63 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 c-n-f Metadata [540 B]
Fetched 9,825 kB in 46s (209 kB/s)
Reading package lists... Done
rose@rose-ubuntu:~/05_Assg/linux-5.14.1$ sudo apt install build-essential libncurses-dev flex bison libssl-dev libelf-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
build-essential is already the newest version (12.8ubuntu1.1).
The following package was automatically installed and is no longer required:
  libfprint-2-tod1
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  libelf-dev libelf2 libsigsegv2 libssl1.1 m4
Suggested packages:
  bison-doc flex-doc ncurses-doc libssl-doc m4-doc
The following NEW packages will be installed:
  bison flex libelf-dev libelf2 libncurses-dev libsigsegv2 libssl-dev m4
The following packages will be upgraded:
  libssl1.1
1 upgraded, 9 newly installed, 0 to remove and 170 not upgraded.
Need to get 2,904 kB/4,595 kB of archives.
After this operation, 14.3 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 libssl1.1 amd64 1.1.1f-1ubuntu2.8 [1,320 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 libssl-dev amd64 1.1.1f-1ubuntu2.8 [1,584 kB]
Fetched 2,904 kB in 3s (868 kB/s)
Preconfiguring packages ...
Selecting previously unselected package libsigsegv2:amd64.
(Reading database ... 287821 files and directories currently installed.)
Preparing to unpack .../0-libsigsegv2_2.12-2_amd64.deb ...
Unpacking libsigsegv2:amd64 (2.12-2) ...
Selecting previously unselected package m4.
Preparing to unpack .../1-m4_1.4.18-4_amd64.deb ...
Unpacking m4 (1.4.18-4) ...
Selecting previously unselected package flex.
Preparing to unpack .../2-flex_2.6.4-6.2_amd64.deb ...
Unpacking flex (2.6.4-6.2) ...
Preparing to unpack .../3-libssl1.1_1.1.1f-1ubuntu2.8_amd64.deb ...
Unpacking libssl1.1:amd64 (1.1.1f-1ubuntu2.8) over (1.1.1f-1ubuntu2.4) ...
Selecting previously unselected package bison.
Preparing to unpack .../4-bison_2.3a3.5.1+dfsg-1_amd64.deb ...
Unpacking bison (2.3a3.5+dfsg-1) ...
```

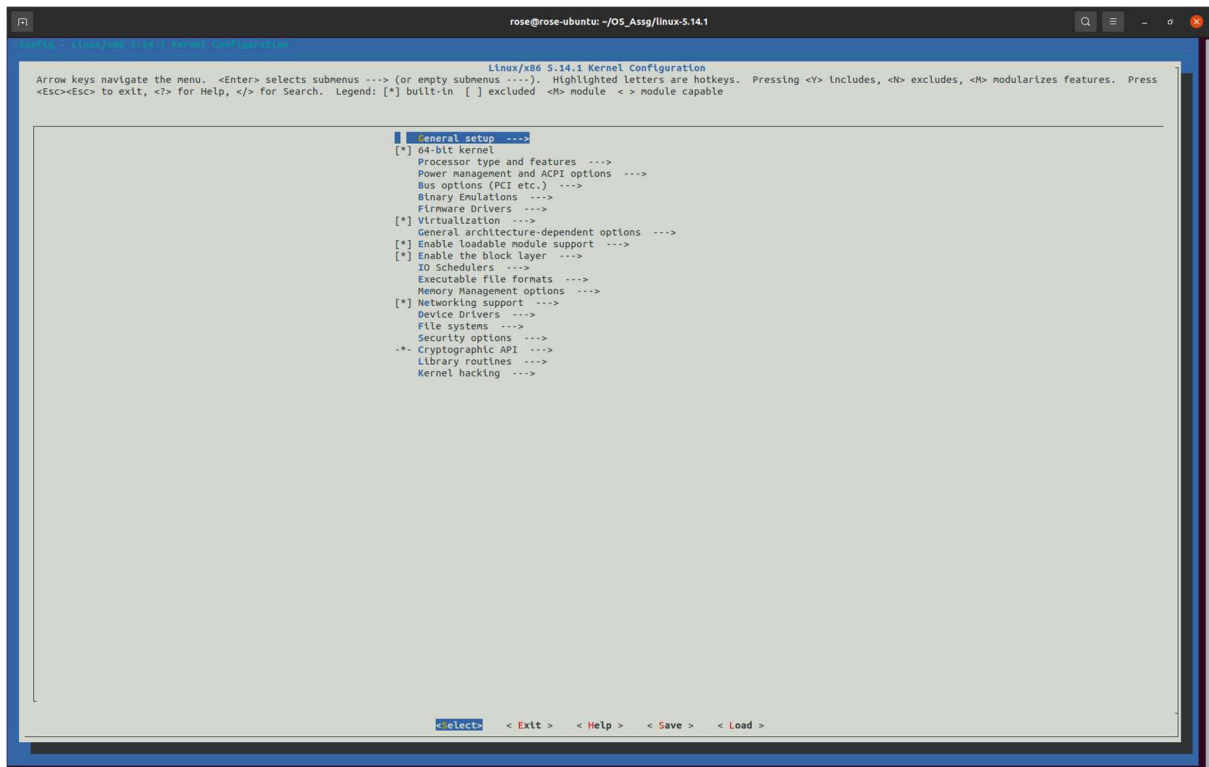
Step 8: \$make menuconfig

Just to make sure that you have all the required files in the directory and you are good to compile the code check the menuconfig.

```
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for install-info (6.7.0.dfsg.2-5) ...
rose@rose-ubuntu:~/05_Assg/linux-5.14.1$ make menuconfig
HOSTCC scripts/basic/fixdep
UPD scripts/kconfig/mconf.cfg
HOSTCC scripts/kconfig/mconf.o
HOSTCC scripts/kconfig/xdialog/checklist.o
HOSTCC scripts/kconfig/xdialog/inputbox.o
HOSTCC scripts/kconfig/xdialog/menubox.o
HOSTCC scripts/kconfig/xdialog/textbox.o
HOSTCC scripts/kconfig/xdialog/uttl.o
HOSTCC scripts/kconfig/xdialog/yesno.o
HOSTCC scripts/kconfig/confdata.o
HOSTCC scripts/kconfig/expr.o
LEX scripts/kconfig/lexer.lex.c
YACC scripts/kconfig/parser.tab.[ch]
HOSTCC scripts/kconfig/lexer.lex.o
HOSTCC scripts/kconfig/menu.o
HOSTCC scripts/kconfig/parser.tab.o
HOSTCC scripts/kconfig/preprocess.o
HOSTCC scripts/kconfig/symbol.o
HOSTCC scripts/kconfig/uttl.o
HOSTLD scripts/kconfig/mconf
.config:2367:warning: symbol value 'n' invalid for MTD_NAND_ECC_SW_HAMMING
.config:2544:warning: symbol value 'n' invalid for PVPANIC
.config:8467:warning: symbol value 'n' invalid for ASHMEM
.config:9474:warning: symbol value 'n' invalid for ANDROID_BINDER_IPC
.config:9475:warning: symbol value 'n' invalid for ANDROID_BINDERFS
.config:10612:warning: symbol value '1' invalid for KASAN_STACK

*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.
```

This command will open up a configuration tool that allows you to go through every module available and enable or disable what you need or don't need.



You can either choose to edit the given configuration or just save and exit.

Step 9: \$make -j4

This command will start compiling the kernel code.

Before compiling you need to make sure that you have enough disk space allotted as this took me around 25GB of space.

The -j parameter is to customize the number of threads you are allocating for the compiling process. In this case, it is 4. You can have 5 or 6 depending on your systems performance.

It almost 2 hours for the compilation. Make sure that you don't turn off your system or let it sleep during the process.

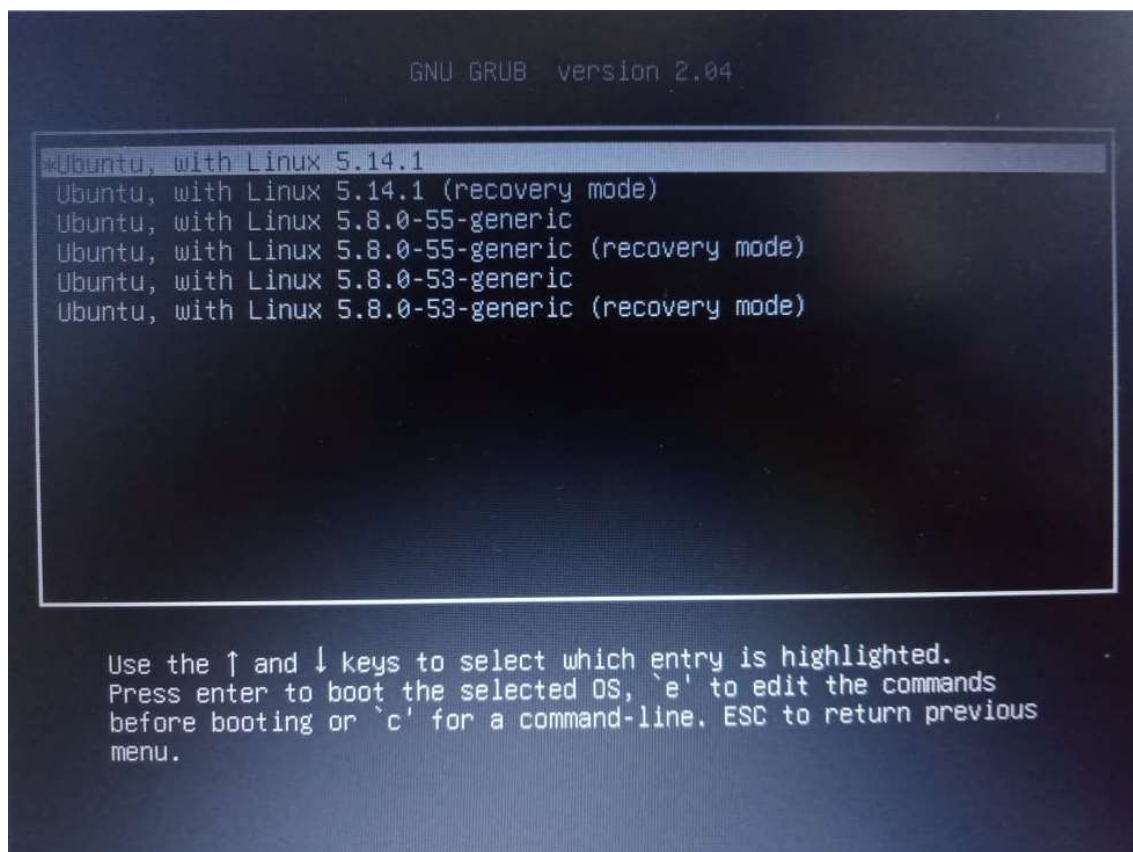

```
SIGN /lib/modules/5.14.1/kernel/sound/usb/usx2y/snd-usx2y.ko
INSTALL /lib/modules/5.14.1/kernel/sound/x86/snd-hdmi-lpe-audio.ko
SIGN /lib/modules/5.14.1/kernel/sound/x86/snd-hdmi-lpe-audio.ko
INSTALL /lib/modules/5.14.1/kernel/sound/xen/snd_xen_front.ko
SIGN /lib/modules/5.14.1/kernel/sound/xen/snd_xen_front.ko
DEPMOD /lib/modules/5.14.1
root@rose-ubuntu:~/OS_Assg/linux-5.14.1# sudo make install
arch/x86/Makefile:148: CONFIG_X86_X32 enabled but no binutils support
sh ./arch/x86/boot/install.sh \
  5.14.1 arch/x86/boot/bzImage \
  System.map "/boot"
run-parts: executing /etc/kernel/postinst.d/apt-auto-removal 5.14.1 /boot/vmlinuz-5.14.1
run-parts: executing /etc/kernel/postinst.d/dkms 5.14.1 /boot/vmlinuz-5.14.1
* dkms: running auto installation service for kernel 5.14.1
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 5.14.1 /boot/vmlinuz-5.14.1
update-initramfs: Generating /boot/initrd.img-5.14.1
run-parts: executing /etc/kernel/postinst.d/unattended-upgrades 5.14.1 /boot/vmlinuz-5.14.1
run-parts: executing /etc/kernel/postinst.d/update-notifier 5.14.1 /boot/vmlinuz-5.14.1
run-parts: executing /etc/kernel/postinst.d/vboxadd 5.14.1 /boot/vmlinuz-5.14.1
VirtualBox Guest Additions: Building the modules for kernel 5.14.1.
VirtualBox Guest Additions: Look at /var/log/vboxadd-setup.log to find out what
went wrong
run-parts: executing /etc/kernel/postinst.d/zz-update-grub 5.14.1 /boot/vmlinuz-5.14.1
Sourcing file /etc/default/grub
Sourcing file /etc/default/grub.d/init-select.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-5.14.1
Found initrd image: /boot/initrd.img-5.14.1
Found linux image: /boot/vmlinuz-5.8.0-55-generic
Found initrd image: /boot/initrd.img-5.8.0-55-generic
Found linux image: /boot/vmlinuz-5.8.0-53-generic
Found initrd image: /boot/initrd.img-5.8.0-53-generic
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
done
root@rose-ubuntu:~/OS_Assg/linux-5.14.1#
```

Step 11: \$sudo reboot

You can directly restart or type the above command in the terminal. Once the system starts it will open GRUB menu. Open advanced settings and select the newly booted kernel i.e. version 5.14.1

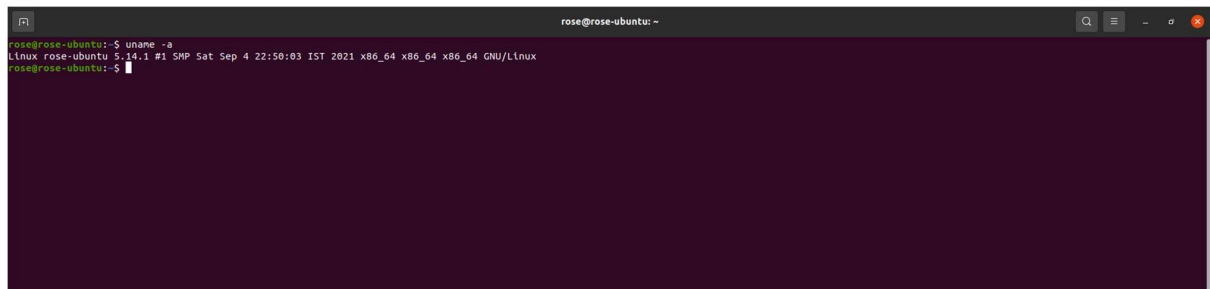
CONCLUSION

When we restart the system we can see option to select the kernel module from GRUB menu in advanced options. Since it is the latest version (5.14.1), it will be selected by default.



\$uname -a

This command will give the current version of the kernel which should be 5.14.1

A terminal window titled 'rose@rose-ubuntu: ~' with a dark background and light green text. The prompt is 'rose@rose-ubuntu:~\$'. The command 'uname -a' has been entered and executed. The output is displayed on the next line: 'Linux rose-ubuntu 5.14.1 #1 SMP Sat Sep 4 22:50:03 IST 2021 x86_64 x86_64 x86_64 GNU/Linux'. The prompt 'rose@rose-ubuntu:~\$' is visible again on the third line.

```
rose@rose-ubuntu:~$ uname -a
Linux rose-ubuntu 5.14.1 #1 SMP Sat Sep 4 22:50:03 IST 2021 x86_64 x86_64 x86_64 GNU/Linux
rose@rose-ubuntu:~$
```