

**NATIONAL UNIVERSITY OF COMPUTER AND
EMERGING SCIENCES
PROGRAM: SOFTWARE ENGINEERING**



OPERATING SYSTEMS LAB
ASSIGNMENT-01

SUBMITTED BY:

Name: Ahmed Ali

Roll No: 22P-9318

Section: BS(SE)-5B

INSTRUCTOR NAME: Sir Saad Ahmad
A DEPARTMENT OF COMPUTER SCIENCE

So Before Moving to Actual Work of Assignment We will try to remove some dependencies issues. For that first of all I will update my system by using command `sudo apt update`.

```
amei-302@amei302-HP-EliteBook-840-G3:~$ sudo apt update
[sudo] password for amei-302:
Hit:1 http://pk.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://pk.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://pk.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:5 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages
Get:6 http://security.ubuntu.com/ubuntu jammy-security/main i386 Packages
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main arm64 Packages
Get:8 http://security.ubuntu.com/ubuntu jammy-security/main armhf Packages
Get:9 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages
Get:10 http://security.ubuntu.com/ubuntu jammy-security/restricted i386 Packages
Get:11 http://security.ubuntu.com/ubuntu jammy-security/restricted arm64 Packages
Get:12 http://security.ubuntu.com/ubuntu jammy-security/restricted armhf Packages
Get:13 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages
Get:14 http://security.ubuntu.com/ubuntu jammy-security/universe i386 Packages
```

Now for downloading and installing some necessary tools that will help in removing dependencies issues like gcc, ncurses, bison, flex, libssl-dev and libelf-dev we will use the following command:
`sudo apt install build-essential libncurses-dev bison flex libssl-dev libelf-dev`

```
amei-302@amei302-HP-EliteBook-840-G3:~$ sudo apt install build-essential libncurses-dev bison flex libssl-dev libelf-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu dpkg-dev fakeroot g++
  g++-11 gcc gcc-11 libalgorithm-diff-perl libalgorithm-diff-xs-perl
  libalgorithm-merge-perl libasan6 libbinutils libc-dev-bin libc-devtools
  libc6 libc6-dbg libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0 libctf0
  libdpkg-perl libfakeroot libfile-fcntllock-perl libfl-dev libfl2
  libgcc-11-dev libitm1 liblsan0 libnspr-dev libsigsegv2 libssl3
  libstdc++-11-dev libtirpc-dev libtsan0 libubsan1 linux-libc-dev
  lto-disabled-list m4 make manpages-dev rpcsvc-proto zlib1g-dev
Suggested packages:
  binutils-doc bison-doc debian-keyring flex-doc g++-multilib g++-11-multilib
  gcc-11-doc gcc-multilib autoconf automake libtool gcc-doc gcc-11-multilib
  gcc-11-locales glibc-doc git bzip ncurses-doc libssl-doc libstdc++-11-doc
  m4-doc make-doc
Recommended packages:
```

After installing necessary tools and removing dependencies issues we **Created the Backup Directory** using the command

```
amei-302@amei302-HP-EliteBook-840-G3:~$ mkdir -p ~/backup/
amei-302@amei302-HP-EliteBook-840-G3:~$ ls
backup Desktop Documents Downloads Music Pictures Public snap Templates Videos
```

Now Command for backing up the current kernel in backup directory that we have created:

simply copies your current kernel image (the one you're currently running) to the `~/backup/` directory. This provides a backup of the kernel before making any changes, like compiling or upgrading a new kernel.

```
amei-302@amei302-HP-EliteBook-840-G3:~$ sudo cp /boot/vmlinuz-$(uname -r) ~/backup/
amei-302@amei302-HP-EliteBook-840-G3:~$
```

Now your kernel has been backed up in a directory.

Next, download any version of kernel from <https://www.kernel.org>, I have downloaded the latest version kernel 6.11.1, I have pointed the cursor on the option as shown in image below click on it and the download will start automatically.



The screenshot shows the 'The Linux Kernel Archives' website. The header includes navigation links: About, Contact us, FAQ, Releases, Signatures, and Site news. A penguin logo is in the top right. A table lists download protocols and locations: HTTP (https://www.kernel.org/pub/), GIT (https://git.kernel.org/), and RSYNC (rsync://rsync.kernel.org/pub/). A yellow button labeled 'Latest Release 6.11.1' with a download icon is highlighted by a red arrow. Below the table, a list of kernel versions is shown with links for tarball, patch, view diff, browse, and changelog.

	Protocol	Location
	HTTP	https://www.kernel.org/pub/
	GIT	https://git.kernel.org/
	RSYNC	rsync://rsync.kernel.org/pub/

Latest Release
6.11.1 ↓

	mainline:	6.12-rc1	2024-09-29	[tarball]	[patch]	[view diff]	[browse]		
stable:	6.11.1	2024-09-30	[tarball]	[pgp]	[patch]	[view diff]	[browse]	[changelog]	
stable:	6.10.12	2024-09-30	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	6.6.53	2024-09-30	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	6.1.112	2024-09-30	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]

Command used to extract the kernel file is `tar -xvf linux-6.11.1.tar.xz`
(`tar -xvf <name of file with .tar extension>`)

```
amei-302@amei302-HP-EliteBook-840-G3:~$ ls
backup Desktop Documents Downloads linux-6.11.1 linux-6.11.1.tar.xz
amei-302@amei302-HP-EliteBook-840-G3:~$ tar -xvf linux-6.11.1
```

Kernel extraction Process:

```
linux-6.11.1/sound/soc/codecs/rt5677.h
linux-6.11.1/sound/soc/codecs/rt5682-i2c.c
linux-6.11.1/sound/soc/codecs/rt5682-sdw.c
linux-6.11.1/sound/soc/codecs/rt5682.c
linux-6.11.1/sound/soc/codecs/rt5682.h
linux-6.11.1/sound/soc/codecs/rt5682s.c
linux-6.11.1/sound/soc/codecs/rt5682s.h
linux-6.11.1/sound/soc/codecs/rt700-sdw.c
linux-6.11.1/sound/soc/codecs/rt700-sdw.h
linux-6.11.1/sound/soc/codecs/rt700.c
linux-6.11.1/sound/soc/codecs/rt700.h
linux-6.11.1/sound/soc/codecs/rt711-sdca-sdw.c
linux-6.11.1/sound/soc/codecs/rt711-sdca-sdw.h
linux-6.11.1/sound/soc/codecs/rt711-sdca.c
linux-6.11.1/sound/soc/codecs/rt711-sdca.h
linux-6.11.1/sound/soc/codecs/rt711-sdw.c
linux-6.11.1/sound/soc/codecs/rt711-sdw.h
linux-6.11.1/sound/soc/codecs/rt711.c
linux-6.11.1/sound/soc/codecs/rt711.h
```

Moving into new kernel directory

```
amei-302@amei302-HP-EliteBook-840-G3:~$ cd linux-6.11.1/  
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$
```

Now we will configure and customize the kernel:
type the command of **make menuconfig** a screen will appear in front of you

```
.config - Linux/x86 6.11.1 Kernel Configuration  
  
Linux/x86 6.11.1 Kernel Configuration  
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighte  
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] bu  
  
General setup --->  
[*] 64-bit kernel  
Processor type and features --->  
[*] Mitigations for CPU vulnerabilities --->  
Power management and ACPI options --->  
Bus options (PCI etc.) --->  
Binary Emulations --->  
[*] Virtualization --->  
General architecture-dependent options --->  
[*] Enable loadable module support --->  
[*] Enable the block layer --->  
Executable file formats --->  
Memory Management options --->  
[*] Networking support --->  
Device Drivers --->  
File systems --->  
Security options --->  
-* Cryptographic API --->  
Library routines --->  
Kernel hacking --->
```

I have disabled the MMC/SD/SDIO card support from device drivers which means the memory card/ micro sd card drivers.

```
on  
  
Device Drivers  
lects submenus ---> (or empty submenus ----). High  
  
^(-)  
[*] USB support --->  
< > MMC/SD/SDIO card support ----  
<M> Universal Flash Storage Controller --->  
<M> Sony MemoryStick card support --->  
-* LED Support --->
```

Then I disabled the Memory technology devices MTD which is used for embedded systems and flash memory chips from device drivers

```

<M> RaplD10 support --->
Generic Driver Options ---->
Bus devices --->
Cache Drivers ----
[*] Connector - unified userspace <-> kernelspace
Firmware Drivers --->
<M> GNSS receiver support --->
< > Memory Technology Device (MTD) support ----
[ ] Device Tree and Open Firmware support ----
<M> Parallel port support --->
-* Plug and Play support --->
[*] Block devices --->
    NVME Support --->
    Misc devices --->

```

As My pc does not support touch input so I moved to Device Drivers>Input device support>Touchscreens and disabled Touchscreen drivers.

```

[*] Keyboards --->
[*] Mice --->
[*] Joysticks/Gamepads --->
[*] Tablets --->
[ ] Touchscreens ----
[*] Miscellaneous devices --->
{M} Synaptics RMI4 bus support
<M> RMI4 I2C Support
<M> RMI4 F01 Support

```

Save it with .config name and press <esc><esc> for doing more customization, It will go back.

Now I go in Networking support>Networking options, and disable older and unused network protocols.

```

Memory Management options --->
[*] Networking support --->
    Device Drivers --->
    File systems --->
    Security options --->
    -* Cryptographic API --->
    Library routines --->
    Kernel hacking --->

```

↓

```

--- Networking support
[ ] Networking options --->
[*] Amateur Radio support --->
<M> CAN bus subsystem support

```

First of all I disabled Appletalk protocol support:

```

[*] MVRP (Multiple VLAN Registra
<M> ANSI/IEEE 802.2 LLC type 2 Sup
< > Appletalk protocol support
<M> CCITT X.25 Packet Layer

```

now I press <esc><esc> and come in Networking support option. Here I disabled the bluetooth subsystem support as I do not use bluetooth support in my pc


```

Networking options --->
[*] Amateur Radio support --->
<M> CAN bus subsystem support --->
< > Bluetooth subsystem support ----
{M} RxRPC session sockets
[*] IPv6 support for RxRPC
[*] Transmit packet loss into RxRPC pack

```

Now I press<esc><esc> and go in device drivers there I disabled the InfiniBand Support which is used for high-performance computing clusters.

```

<M> Sony MemoryStick card support --->
-* LED Support --->
[*] Accessibility support --->
< > InfiniBand support ----
<*> EDAC (Error Detection And Correction)
[*] Real Time Clock --->

```

In the Device drivers I moved in IEEE 1394 (FireWire) support and disabled all drivers from there as I do not use any fireWire devices on my system.

```

[*] Fusion MPT device support --->
IEEE 1394 (FireWire) support --->
[*] Macintosh device drivers --->

```



```

< > FireWire driver stack
< > Nopsy - a FireWire traffic sniffer for PCILynx cards

```

Disabling old/unused sound cards drivers, In the Device driver disable sound card support

```

[*] Compute Acceleration Framework --->
< > Sound card support ----
[*] HID bus support --->

```

Disabling Virtual Terminal

Device Drivers>Character devices>Virtual Terminal

this is safe to disable if you're using a graphical interface and don't need virtual terminals for troubleshooting)

```

[*] Enable TTY
[*] Virtual terminal
[*] Unix98 PTY support

```

Drivers to Keep

Input Devices (Keyboard/Mouse):

- Enabling drivers for standard input devices like keyboards and mice.
- Location: Device Drivers>Input device support>Keyboards/Mouse.

```
*** Input Device Drivers ***
[*] Keyboards --->
[*] Mice --->
[*] Joysticks/Gamepads --->
[*] Tablets --->
```

Graphics Card Drivers:

- Enabling and keeping drivers for your GPU
- Location: Device Drivers>Graphics support

```
<M> Multimedia support --->
[*] Graphics support --->
[*] Compute Acceleration Framework --->
< > Sound card support ----
[*] HID bus support --->
```

```
-*- Auxiliary Display support --->
<M> Parallel port LCD/Keypad Panel support (OLD OPTION)
<*> /dev/agpgart (AGP Support) --->
[*] Laptop Hybrid Graphics - GPU switching support
<*> Direct Rendering Manager (XFree86 4.1.0 and higher D
    Frame buffer Devices --->
    Backlight & LCD device support --->
[*] Bootup logo --->
```

USB Support:

- Keep **USB device support** for peripherals like external drives, keyboards, and mice.
- Location: Device Drivers>USB support

```
[*] HID bus support --->
[*] USB support --->
< > MMC/SD/SDIO card support ----
<M> Universal Flash Storage Controller --->
<M> Sony MemoryStick card support --->
```

Ethernet/Wi-Fi Drivers:

- Keep drivers for wireless (Wi-Fi) network cards.
- Location: Device Drivers>Network device support>Wireless LAN

```

[*] Six bit SLIP encapsulation
<M> USB Network Adapters --->
[*] Wireless LAN --->
[*] Wan interfaces support --->
<M> IEEE 802.15.4 drivers --->

```

File System Drivers:

- Keep drivers for the file systems you use (e.g., **ext4**).
- Location: File systems>

```

[*] Networking support --->
    Device Drivers --->
    File systems --->
    Security options --->
    *- Cryptographic API --->

```

I enabled the Ext4 debugging support for making kernel more efficient.

```

[*] Use ext4 for ext2 file systems
[*] Ext4 POSIX Access Control Lists
[*] Ext4 Security Labels
[*] Ext4 debugging support
[ ] JBD2 (ext4) debugging support

```

After this I Select the save command and then Exit, and again I come back on terminal.

Compiling Kernel:

```

amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ make -j$(nproc)
SYNC    include/config/auto.conf.cmd
HOSTCC  scripts/kconfig/conf.o
HOSTLD  scripts/kconfig/conf
WRAP    arch/x86/include/generated/uapi/asm/bpf_perf_event.h
WRAP    arch/x86/include/generated/uapi/asm/errno.h
WRAP    arch/x86/include/generated/uapi/asm/fcntl.h
WRAP    arch/x86/include/generated/uapi/asm/ioctl.h
WRAP    arch/x86/include/generated/uapi/asm/ioctls.h
WRAP    arch/x86/include/generated/uapi/asm/ipcbuf.h
WRAP    arch/x86/include/generated/uapi/asm/param.h
UPD     include/config/kernel.release
WRAP    arch/x86/include/generated/uapi/asm/poll.h
WRAP    arch/x86/include/generated/uapi/asm/resource.h
WRAP    arch/x86/include/generated/uapi/asm/socket.h
WRAP    arch/x86/include/generated/uapi/asm/sockios.h
WRAP    arch/x86/include/generated/uapi/asm/timex.h

```

Instead of manually specifying the number of cores, `$(nproc)` automatically adjusts to the number of cores on your machine.

While the compilation of kernel I have faced one issue of permission that was occurring due to the file owner name was root so I used sudo chown command and rename the file owner name according to my system name.

```
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ cd ..  
amei-302@amei302-HP-EliteBook-840-G3:~$ sudo chown -R amei-302:amei-302 linux-6.11.1
```

Now all file names are according to my system name.

```
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ ls -l security/selinux/av_permissions.h  
-rw-r--r-- 1 amei-302 amei-302 105476 18:56 1 اکتوبر security/selinux/av_permissions.h  
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$
```

```
amei-302@amei302-HP-EliteBook-840-G3:~$ cd linux-6.11.1/  
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ ls -l security/selinux/  
total 840  
-rw-rw-r-- 1 amei-302 amei-302 31760 19:31 30 ستمبر avc.c  
-rw-rw-r-- 1 amei-302 amei-302 100288 18:56 1 اکتوبر avc.o  
-rw-rw-r-- 1 amei-302 amei-302 105476 18:56 1 اکتوبر av_permissions.h  
-rw-rw-r-- 1 amei-302 amei-302 1512 18:57 1 اکتوبر built-in.a  
-rw-rw-r-- 1 amei-302 amei-302 8683 18:56 1 اکتوبر flask.h  
-rw-rw-r-- 1 amei-302 amei-302 198074 19:31 30 ستمبر hooks.c  
-rw-rw-r-- 1 amei-302 amei-302 133864 18:56 1 اکتوبر hooks.o  
-rw-rw-r-- 1 amei-302 amei-302 5686 19:31 30 ستمبر ibpkey.c  
-rw-rw-r-- 1 amei-302 amei-302 2805 19:31 30 ستمبر ima.c  
drwxrwxr-x 2 amei-302 amei-302 4096 19:31 30 ستمبر include  
-rw-rw-r-- 1 amei-302 amei-302 3155 19:31 30 ستمبر Kconfig  
-rw-rw-r-- 1 amei-302 amei-302 1621 19:31 30 ستمبر Makefile  
-rw-rw-r-- 1 amei-302 amei-302 0 18:56 1 اکتوبر modules.order  
-rw-rw-r-- 1 amei-302 amei-302 6870 19:31 30 ستمبر netif.c  
-rw-rw-r-- 1 amei-302 amei-302 5888 18:56 1 اکتوبر netif.o  
-rw-rw-r-- 1 amei-302 amei-302 16387 19:31 30 ستمبر netlabel.c  
-rw-rw-r-- 1 amei-302 amei-302 9856 18:57 1 اکتوبر netlabel.o  
-rw-rw-r-- 1 amei-302 amei-302 2358 19:31 30 ستمبر netlink.c  
-rw-rw-r-- 1 amei-302 amei-302 4024 18:56 1 اکتوبر netlink.o  
-rw-rw-r-- 1 amei-302 amei-302 7641 19:31 30 ستمبر netnode.c  
-rw-rw-r-- 1 amei-302 amei-302 5872 18:56 1 اکتوبر netnode.o  
-rw-rw-r-- 1 amei-302 amei-302 5952 19:31 30 ستمبر netport.c  
-rw-rw-r-- 1 amei-302 amei-302 4664 18:56 1 اکتوبر netport.o  
-rw-rw-r-- 1 amei-302 amei-302 9040 19:31 30 ستمبر nlmsgtab.c  
-rw-rw-r-- 1 amei-302 amei-302 3216 18:56 1 اکتوبر nlmsgtab.o  
-rw-rw-r-- 1 amei-302 amei-302 50825 19:31 30 ستمبر selinuxfs.c  
-rw-rw-r-- 1 amei-302 amei-302 54864 18:56 1 اکتوبر selinuxfs.o  
drwxrwxr-x 2 amei-302 amei-302 4096 18:57 1 اکتوبر ss  
-rw-rw-r-- 1 amei-302 amei-302 3354 19:31 30 ستمبر status.c  
-rw-rw-r-- 1 amei-302 amei-302 2832 18:56 1 اکتوبر status.o  
-rw-rw-r-- 1 amei-302 amei-302 11265 19:31 30 ستمبر xfrm.c  
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$
```

Again I compiled the kernel using command make -j\$(nproc) and it has successfully compiled the kernel according to customization.



Installing the Kernel and Modules

For installing the modules we will use the command `sudo make modules_install`, This command will install all modules of kernel that were compiled during the kernel build process.

These modules are loadable kernel modules that can be loaded or unloaded dynamically at runtime, this will copy all the compiled modules to the appropriate location on your system

`lib/modules/kernel-6.11.1`.

```
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ sudo make modules_install
[sudo] password for amei-302:
SYMLINK /lib/modules/6.11.1/build
INSTALL /lib/modules/6.11.1/modules.order
INSTALL /lib/modules/6.11.1/modules.builtin
INSTALL /lib/modules/6.11.1/modules.builtin.modinfo
INSTALL /lib/modules/6.11.1/kernel/fs/efivarfs/efivarfs.ko
INSTALL /lib/modules/6.11.1/kernel/drivers/thermal/intel/x86_pkg_temp_thermal.ko
INSTALL /lib/modules/6.11.1/kernel/net/netfilter/nf_log_syslog.ko
INSTALL /lib/modules/6.11.1/kernel/net/netfilter/xt_mark.ko
INSTALL /lib/modules/6.11.1/kernel/net/netfilter/xt_nat.ko
INSTALL /lib/modules/6.11.1/kernel/net/netfilter/xt_LOG.ko
INSTALL /lib/modules/6.11.1/kernel/net/netfilter/xt_MASQUERADE.ko
INSTALL /lib/modules/6.11.1/kernel/net/netfilter/xt_addrtype.ko
INSTALL /lib/modules/6.11.1/kernel/net/ipv4/netfilter/iptables_nat.ko
DEPMOD /lib/modules/6.11.1
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$
```

Next, we will install the newly compiled kernel image and related files: ***sudo make install*** this will copy the newly compiled kernel image to ***/boot*** preparing the system to boot with the newly compiled kernel.

(SEE BELOW FOR ATTACHED SCREENSHOT ON NEXT PAGE)

```

amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ sudo make install
INSTALL /boot
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 6.11.1 /boot/vmlinuz-6.11.1
update-initramfs: Generating /boot/initrd.img-6.11.1
W: Possible missing firmware /lib/firmware/i915/bmg_dmc.bin for built-in driver i915
W: Possible missing firmware /lib/firmware/i915/xe2lpd_dmc.bin for built-in driver i915
W: Possible missing firmware /lib/firmware/rtl_nic/rtl8126a-2.fw for built-in driver r8169
run-parts: executing /etc/kernel/postinst.d/unattended-upgrades 6.11.1 /boot/vmlinuz-6.11.1
run-parts: executing /etc/kernel/postinst.d/update-notifier 6.11.1 /boot/vmlinuz-6.11.1
run-parts: executing /etc/kernel/postinst.d/xx-update-initrd-links 6.11.1 /boot/vmlinuz-6.11.1
I: /boot/initrd.img.old is now a symlink to initrd.img-6.8.0-40-generic
I: /boot/initrd.img is now a symlink to initrd.img-6.11.1
run-parts: executing /etc/kernel/postinst.d/zz-shim 6.11.1 /boot/vmlinuz-6.11.1
run-parts: executing /etc/kernel/postinst.d/zz-update-grub 6.11.1 /boot/vmlinuz-6.11.1
Sourcing file `/etc/default/grub'
Sourcing file `/etc/default/grub.d/init-select.cfg'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.11.1
Found initrd image: /boot/initrd.img-6.11.1
Found linux image: /boot/vmlinuz-6.8.0-40-generic
Found initrd image: /boot/initrd.img-6.8.0-40-generic
Found linux image: /boot/vmlinuz-6.5.0-18-generic
Found initrd image: /boot/initrd.img-6.5.0-18-generic
Memtest86+ needs a 16-bit boot, that is not available on EFI, exiting
Warning: os-prober will be executed to detect other bootable partitions.
Its output will be used to detect bootable binaries on them and create new boot entries.
Found Windows Boot Manager on /dev/sda2@/EFI/Microsoft/Boot/bootmgfw.efi
Found Windows Boot Manager on /dev/sdb1@/efi/Microsoft/Boot/bootmgfw.efi
Adding boot menu entry for UEFI Firmware Settings ...
done
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ █

```

Updating the Bootloader:

GRUB (Grand Unified Bootloader) is a bootloader used in Linux systems to load the operating system kernel, it allows users to select from multiple operating systems or kernel versions during system startup. GRUB is highly flexible, supporting different file systems and Operating Systems, and provides a user-friendly menu for booting.

```

amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ sudo update-grub
Sourcing file `/etc/default/grub'
Sourcing file `/etc/default/grub.d/init-select.cfg'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.11.1
Found initrd image: /boot/initrd.img-6.11.1
Found linux image: /boot/vmlinuz-6.8.0-40-generic
Found initrd image: /boot/initrd.img-6.8.0-40-generic
Found linux image: /boot/vmlinuz-6.5.0-18-generic
Found initrd image: /boot/initrd.img-6.5.0-18-generic
Memtest86+ needs a 16-bit boot, that is not available on EFI, exiting
Warning: os-prober will be executed to detect other bootable partitions
Its output will be used to detect bootable binaries on them and create
Found Windows Boot Manager on /dev/sda2@/EFI/Microsoft/Boot/bootmgfw.efi
Found Windows Boot Manager on /dev/sdb1@/efi/Microsoft/Boot/bootmgfw.efi
Adding boot menu entry for UEFI Firmware Settings ...
done
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ █

```

My Current Version Of Kernel

```
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ uname -r  
6.8.0-40-generic  
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$
```

Reboot and Select the New Kernel:

```
amei-302@amei302-HP-EliteBook-840-G3:~/linux-6.11.1$ sudo reboot
```

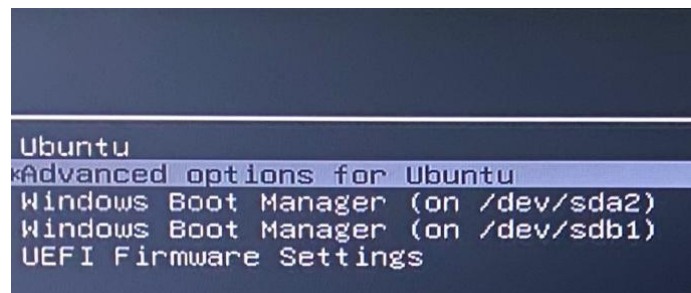
When the system boots up, select your newly installed kernel from the GRUB menu.

While rebooting the GRUB Bootloader interface will appear in front of you;

1. Select Advance Option for UBUNTU

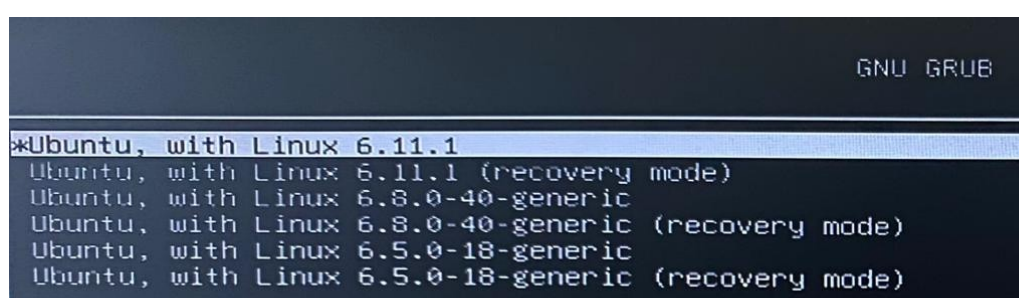
- There List of Kernels will appear, Select the newly compiled kernel by remembering its version like in my case it was linux-6.11.1, So I know which kernel to Select.

I have captured some pictures, They are attached Below:

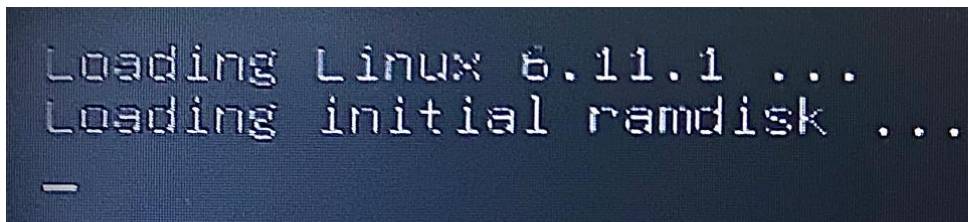


while rebooting when the GRUB Bootloader interface appears select **Advanced options for UBUNTU** and hit enter key.

Now a list of kernels that are compiled at your system will appear here. I selected the Linux 6.11.1 (the latest kernel) as I have downloaded it, made customization in the kernel according to myself and compiled it and press enter key.

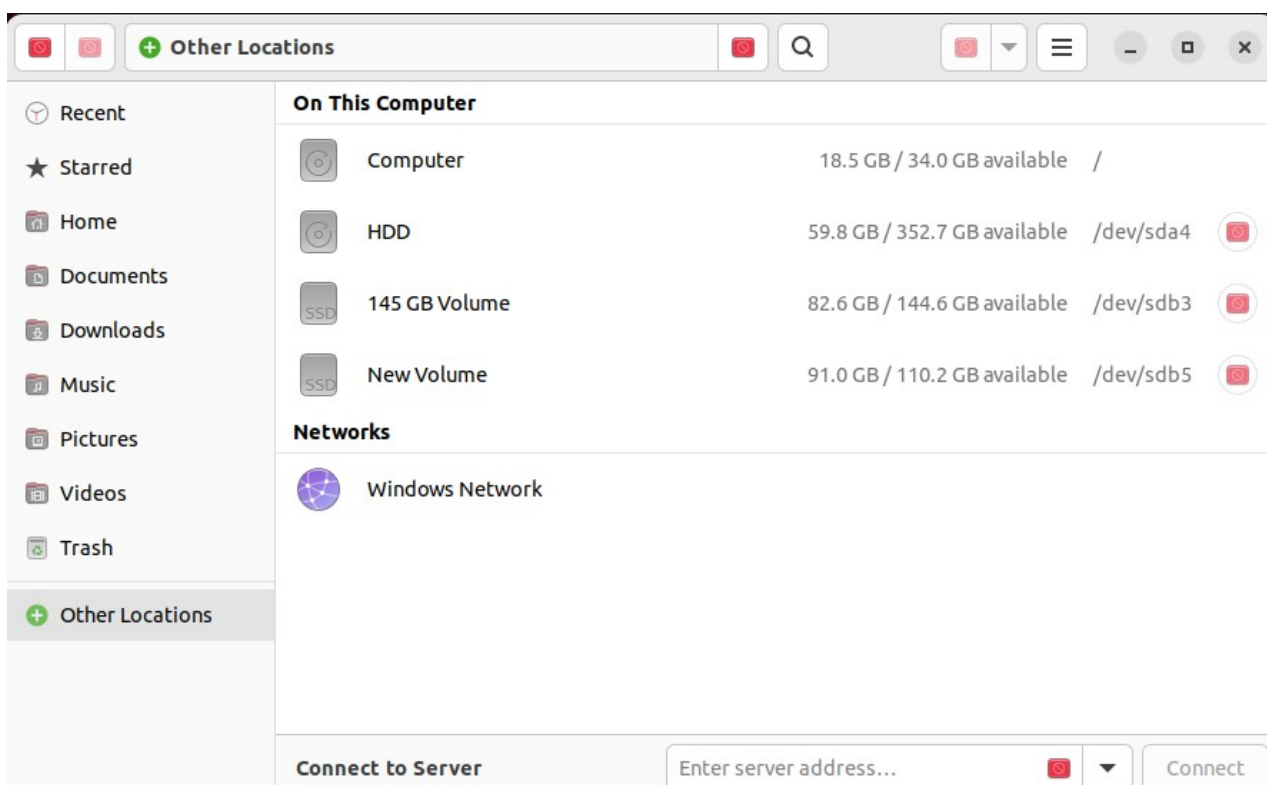
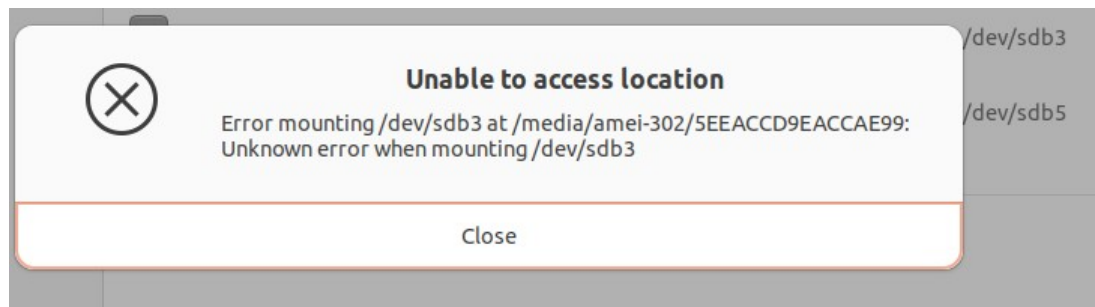
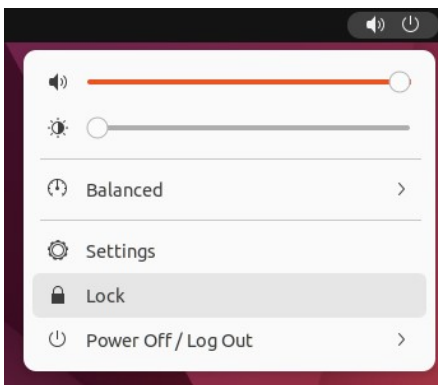


After You press Enter Key, it will take a while to process the kernel.

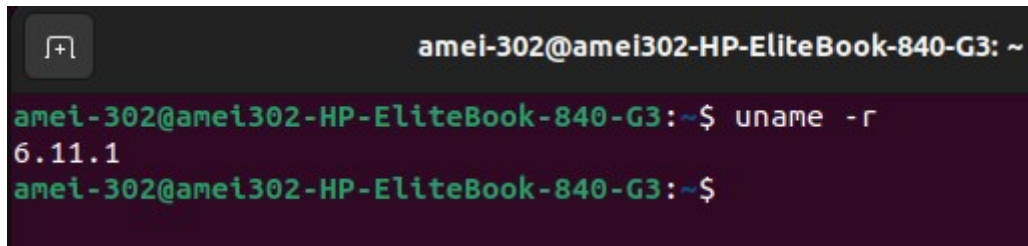


After execution of kernel, The first thing I noticed on my screen is:

- 1) the sound option is showing but I cannot hear the voice of any song.
- 2) No internet
- 3) unable to access the drives other than Ubuntu
- 4) symbols of back and forward are changed



New Version Of Kernel:

A terminal window with a dark background. The title bar at the top reads 'amei-302@amei302-HP-EliteBook-840-G3: ~'. The terminal shows a command 'uname -r' being executed, which returns the output '6.11.1'. The prompt 'amei-302@amei302-HP-EliteBook-840-G3:~\$' is visible at the bottom.

```
amei-302@amei302-HP-EliteBook-840-G3: ~  
amei-302@amei302-HP-EliteBook-840-G3:~$ uname -r  
6.11.1  
amei-302@amei302-HP-EliteBook-840-G3:~$
```