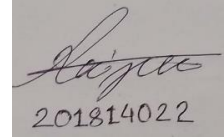# CSE - 308

Assignment - 3 Report

(a)   Kernel Version : 4.19.146

(b)   Major problem faced during the setup of host environment and solution :

- As I don't use Vmware Workstation pro I couldn't create the clone os like it was shown in the class.I copied the existing os and used add a new os then changed it's name to OS-clone and used it for kernel compilartion.
- Couldn't install quota-tools.
    - "sudo apt-get install quota-tools" command didn't work.
    - "E: Unable to locate package quota-tools" massage was given.
    - Solved using "sudo apt install quota" command.
- Couldn't install nfs-utils.
    - "sudo apt-get install nfs-utils" command didn't work.
    - "E: Unable to locate package quota-tools" massage was given.
    - Solved using "sudo apt-get install nfs-common" command.
- Couldn't install grub,mcelog and isdn4k-utils
    - "sudo apt-get install grub" didn't work.
    - "sudo apt-get install mcelog" didn't work.
    - "sudo apt-get install isdn4k-utils" didn't work.
    - "E: Unable to locate package quota-tools" massage was given.
    - Couldn't install these softwares.

(c)   Major problem faced during Compilation process and Solutions :

- After Setting up the host environment I used the command "make -j2" start the kernel compilation process.After a long 2 and a half hours when the compilation of this command was finished no "disk
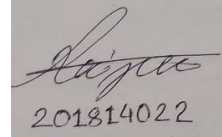
memory shortage" massage was shown.That's why I didn't expand my disk storage and didn't also download gparted.

After that I used the command "sudo make modules_install" to install the modules.

- This command was running all good but after a couple of minutes the "disk storage memory low" massage was shown.But I still thought that the installation will be done within this storage.I also cleared my trash and deleted unnecessary items from storage but after some time the installation process consumed the whole storage.
- After the storage got overwhelmed many error occurred in the installation process and that process ended after sometime.
- The I tried to install gparted, but alas! There was not even 2.5 mb free space in the storage.
- As I couldn't extend the harddisk storage at that point my whole work at that point from the beginning became pointless because after the 2 and a half hour compilation I didn't create a clone because I didn't think it was nacessary at that point.
- That's why I deleted the second clone and then created a new clone and started the whole process again.
- This time the difference was at the very beginning I downloaded gparted and made the harddisk storage 60GB and made the nacessary partition.And then done the whole process again and finished the compilation process.

(d) While exploring the source code of the kernel, which part of the kernel seemed interesting to me and why?

- While exploring the source code of the kernel I found the locking part of the kernel very interesting.I explored the source code of Spinlock.c and saw how kernel manages the resource for different processes by locking one process so that the process already running can finish it's task and then the other process can accure

the CPU core.Spinlocks are designed for multicore processors.

- I also found mapping.c interesting because of how it use the function mmap() and maps different files and documents into the storage.
- As the majority of the kernel is developed in C code and we finished our C course back in first year.Many files and programs seemed very familiar the the concept of interacting with hardware through these language is new to me that's why I struggled understanding them.But I got a overview of how kernel interacts with hardware and tell them what to do and manages their resource and time and debugs their problems basing on these programs.All in all it was interesting exploring the source code of kernel.

(e)   Screenshots of the process:
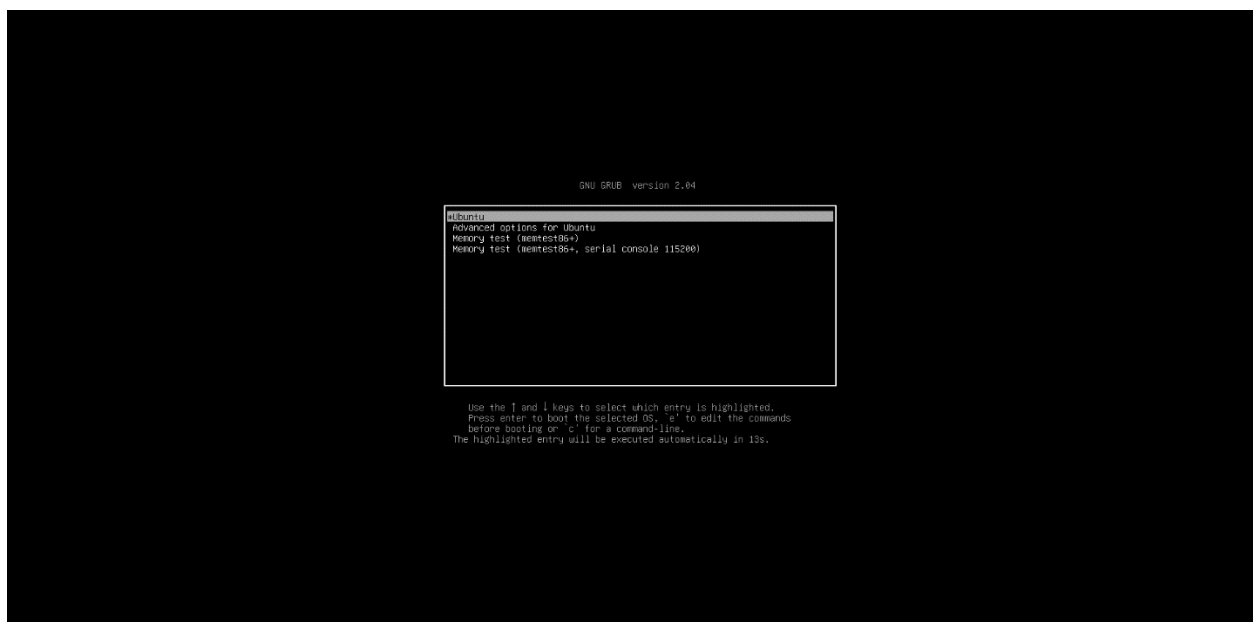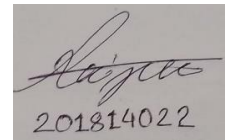
```
kazi@kazi-virtual-machine: ~

kazi@kazi-virtual-machine:~$ fdformat --version
fdformat from util-linux 2.34
kazi@kazi-virtual-machine:~$ depmod -V
kmod version 27
+XZ -ZLIB +LIBCRYPTO -EXPERIMENTAL
kazi@kazi-virtual-machine:~$ e2fsck -V
e2fsck 1.45.5 (07-Jan-2020)
        Using EXT2FS Library version 1.45.5, 07-Jan-2020
kazi@kazi-virtual-machine:~$ fsck.jfs -V

Command 'fsck.jfs' not found, but can be installed with:

sudo apt install jfsutils

kazi@kazi-virtual-machine:~$
```

```
                          Linux/x86 4.19.146 Kernel Configuration
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press
  <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                   *** Compiler: gcc (Ubuntu 9.3.0-10ubuntu2) 9.3.0 ***
                   General setup  --->
               [*] 64-bit kernel
                   Processor type and features  --->
                   Power management and ACPI options  --->
                   Bus options (PCI etc.)  --->
                   Binary Emulations  --->
                   Firmware Drivers  --->
               [*] 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  --->

                 <Select>    < Exit >    < Help >    < Save >    < Load >
```

```
CC [M]  drivers/video/fbdev/n411.o
CC [M]  drivers/video/fbdev/hgafb.o
CC [M]  drivers/video/fbdev/sstfb.o
CC [M]  drivers/video/fbdev/cirrusfb.o
CC [M]  drivers/video/fbdev/metronomefb.o
CC [M]  drivers/video/fbdev/s1d13xxxfb.o
CC [M]  drivers/video/fbdev/sm501fb.o
CC [M]  drivers/video/fbdev/udlfb.o
CC [M]  drivers/video/fbdev/smscufx.o
CC [M]  drivers/video/fbdev/xen-fbfront.o
CC [M]  drivers/video/fbdev/carminefb.o
CC [M]  drivers/video/fbdev/hyperv_fb.o
CC [M]  drivers/video/fbdev/ocfb.o
CC [M]  drivers/video/fbdev/sm712fb.o
CC [M]  drivers/video/fbdev/uvesafb.o
CC [M]  drivers/video/fbdev/vga16fb.o
AR      drivers/video/fbdev/built-in.a
AR      drivers/video/built-in.a
AR      drivers/built-in.a
GEN     .version
CHK     include/generated/compile.h
AR      built-in.a
LD      vmlinux.o
```
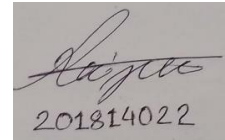
(f)   References :

[1]   William E. Shotts. Jr.,The Linux Command Line : A Complete

Introduction,Part 3 : Common tasks and essential tools,pp. 150-

200.