CS60038
Advances in Operating System Design

Prof. Arobinda Gupta & Prof Sandip Chakraborty

Assignment-1
Report

Group Members
Yashraj Singh - 20CS10079
Rishi Raj - 20CS30040

Part - 1

Steps to install a new kernel version in Ubuntu are as follows:

1. First we increase the **GRUB_TIMEOUT** in the file **/etc/default/grub** to 10 seconds using

```
$ sudo nano /etc/default/grub
```

2. Now, we update the grub using

```
$ sudo update-grub
```

3. Now we go to an empty directory, say **Downloads** in this case.

```
$ cd Downloads
```

4. Now we use **wget** command to download the kernel zip

```
$ wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.10.191.tar.xz
```

5. Next step is to install the dependencies using following commands

```
$ sudo apt install dwarves && sudo apt install -y zstd
$ sudo apt-get install build-essential libncurses-dev libssl-dev libelf-dev gcc bc bison flex
```

6. Making a directory and extract the kernel zip file in it

```
$ mkdir kernel
$ tar -xf linux-5.10.191.tar.xz -C kernel/ --strip-components=1
```

7. Now we copy the config of our current kernel in this directory so that it could support variety of devices

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

8. Disabling **environment variables**

```
$ scripts/config –disable SYSTEM_TRUSTED_KEYS
$ scripts/config –disable SYSTEM_REVOCATION_KEYS
```

9. Using **menuconfig** to edit the configuration file

```
$ make menuconfig
```

10. Now we use **localmodconfig** to make some final changes

```
$ make localmodconfig
```

11. Compile the kernel

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

12. Install the Kernel

```
$ make install
```

13. Enabling kernel for boot

```
$ sudo update-grub
```

14. Rebooting the system

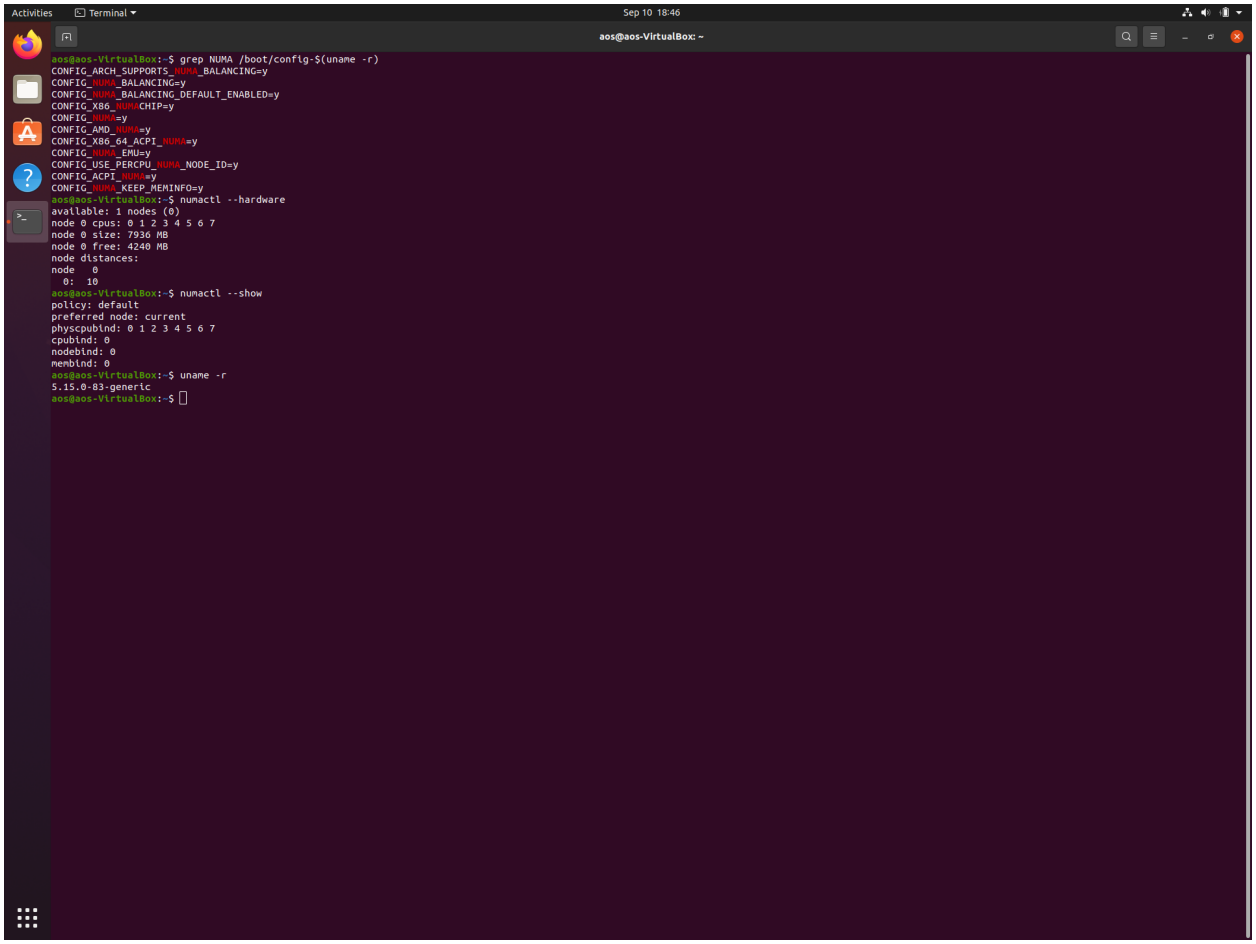
```
$ reboot
```

One point to remember is that after reboot, one should press the **Shift** key on the keyboard to enter the **grub menu**.

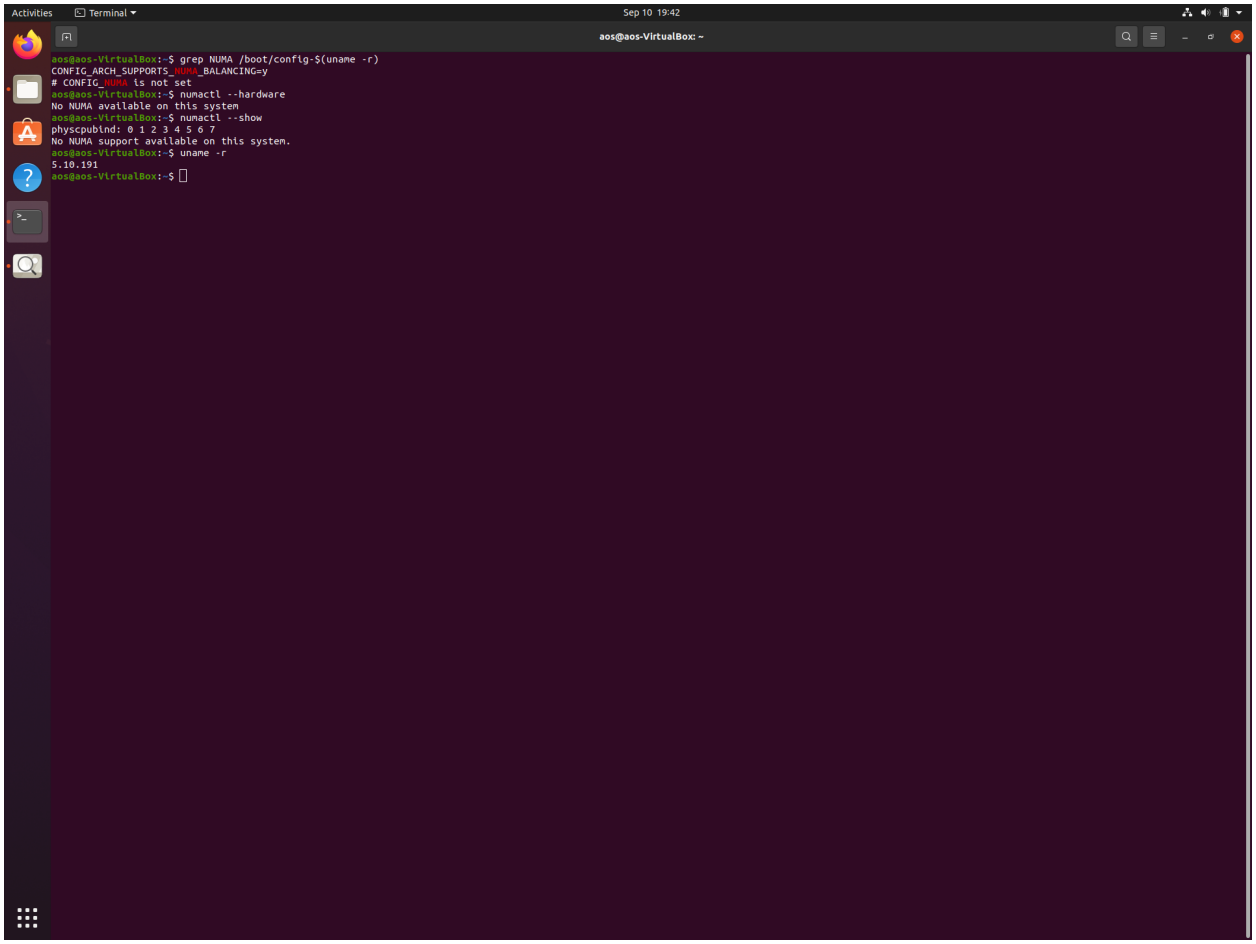
- **Removing NUMA memory allocation, scheduler and emulation**

  NUMA memory allocation, scheduler and emulation was removed using **menuconfig**.
  For this we navigated to the submenu **Processor type and features** to find the **NUMA Memory Allocation and Scheduler Support** and disable it.

  Before ⇒


Activities
Terminal
Sep 10 18:46
aos@aos-VirtualBox: ~
aos@aos-VirtualBox:~$ grep NUMA /boot/config-$(uname -r)
CONFIG_ARCH_SUPPORTS_NUMA_BALANCING=y
CONFIG_NUMA_BALANCING=y
CONFIG_NUMA_BALANCING_DEFAULT_ENABLED=y
CONFIG_X86_NUMACHIP=y
CONFIG_NUMA=y
CONFIG_AMD_NUMA=y
CONFIG_X86_64_ACPI_NUMA=y
CONFIG_NUMA_EMU=y
CONFIG_USE_PERCPU_NUMA_NODE_ID=y
CONFIG_ACPI_NUMA=y
CONFIG_NUMA_KEEP_MEMINFO=y
aos@aos-VirtualBox:~$ numactl --hardware
available: 1 nodes (0)
node 0 cpus: 0 1 2 3 4 5 6 7
node 0 size: 7936 MB
node 0 free: 4240 MB
node distances:
node   0
  0: 10
aos@aos-VirtualBox:~$ numactl --show
policy: default
preferred node: current
physcpubind: 0 1 2 3 4 5 6 7
cpubind: 0
nodebind: 0
membind: 0
aos@aos-VirtualBox:~$ uname -r
5.15.0-83-generic
aos@aos-VirtualBox:~$

After ⇒

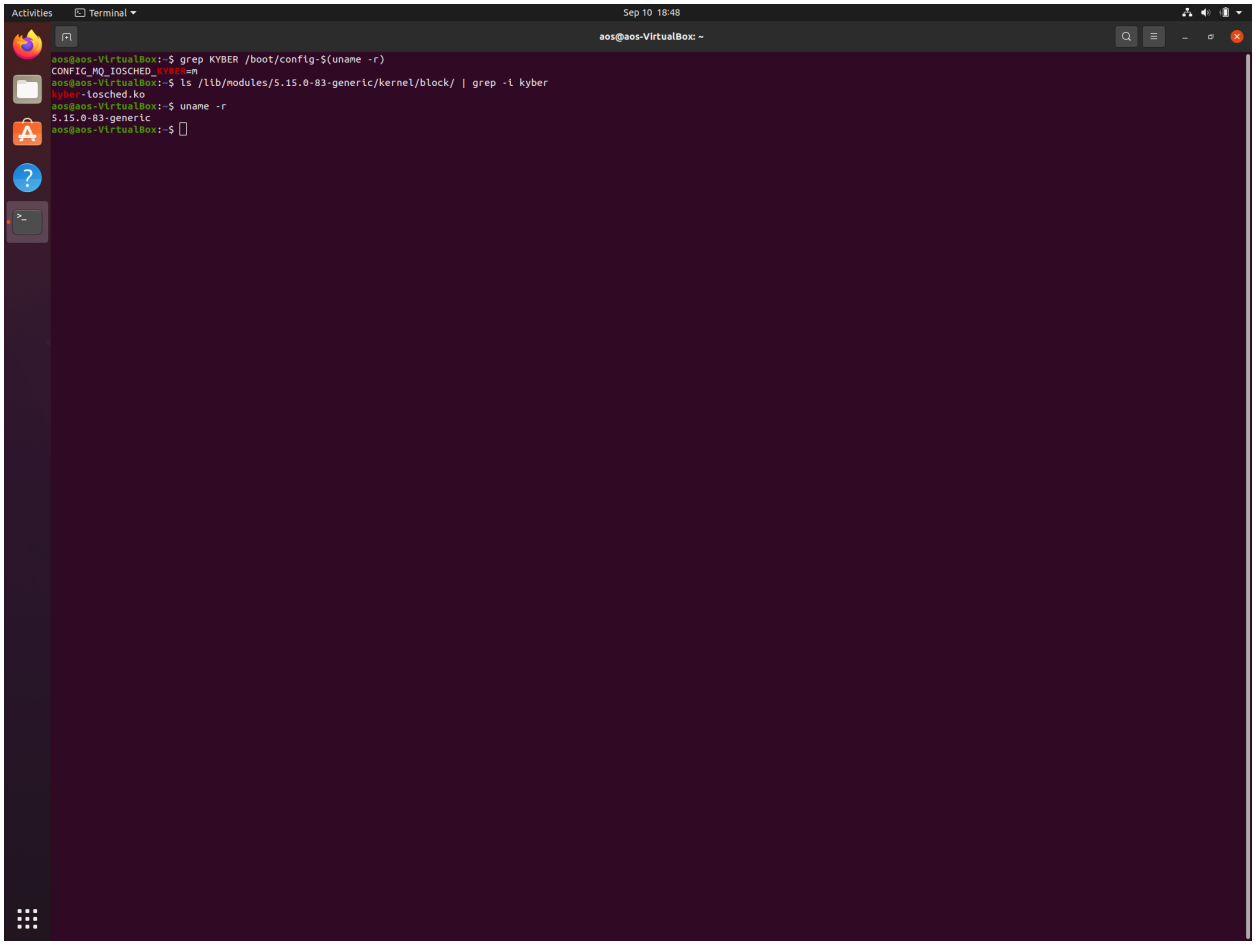

Activities
Terminal
Sep 10 19:42
aos@aos-VirtualBox: ~
aos@aos-VirtualBox:~$ grep NUMA /boot/config-$(uname -r)
CONFIG_ARCH_SUPPORTS_NUMA_BALANCING=y
# CONFIG_NUMA is not set
aos@aos-VirtualBox:~$ numactl --hardware
No NUMA available on this system
aos@aos-VirtualBox:~$ numactl --show
physcpubind: 0 1 2 3 4 5 6 7
No NUMA support available on this system.
aos@aos-VirtualBox:~$ uname -r
5.10.191
aos@aos-VirtualBox:~$

- **Removing Kyber I/O Scheduler**

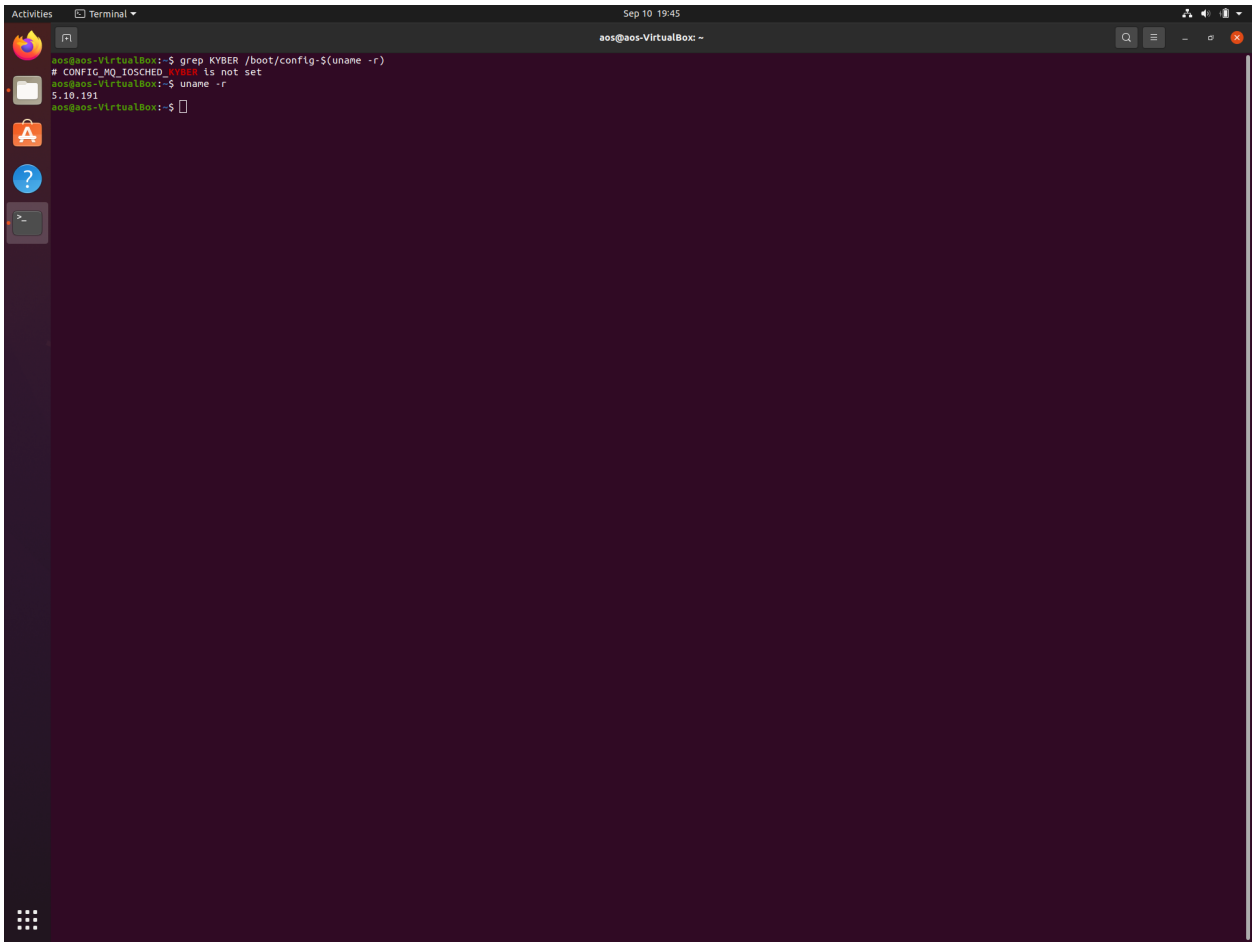
  This was also removed using **menuconfig**.

  For this we navigated to the **Kyber I/O Scheduler** option inside the submenu of **IO Schedulers**. Then we disabled the scheduler.

  Before ⇒



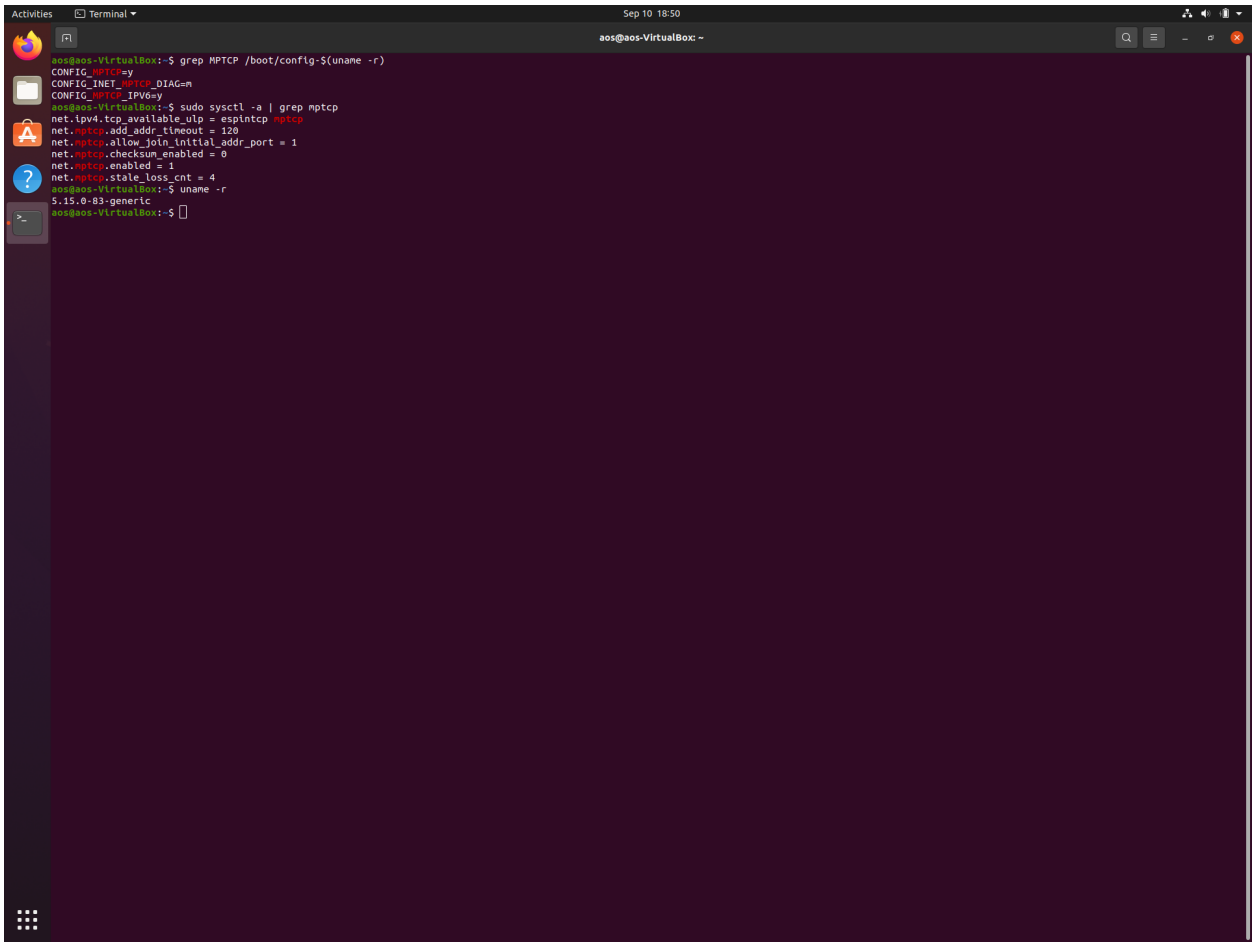
After ⇒


Activities
Terminal
Sep 10 19:45
aos@aos-VirtualBox: ~
aos@aos-VirtualBox:~$ grep KYBER /boot/config-$(uname -r)
CONFIG_MQ_IOSCHED_KYBER is not set
aos@aos-VirtualBox:~$ uname -r
5.10.191
aos@aos-VirtualBox:~$

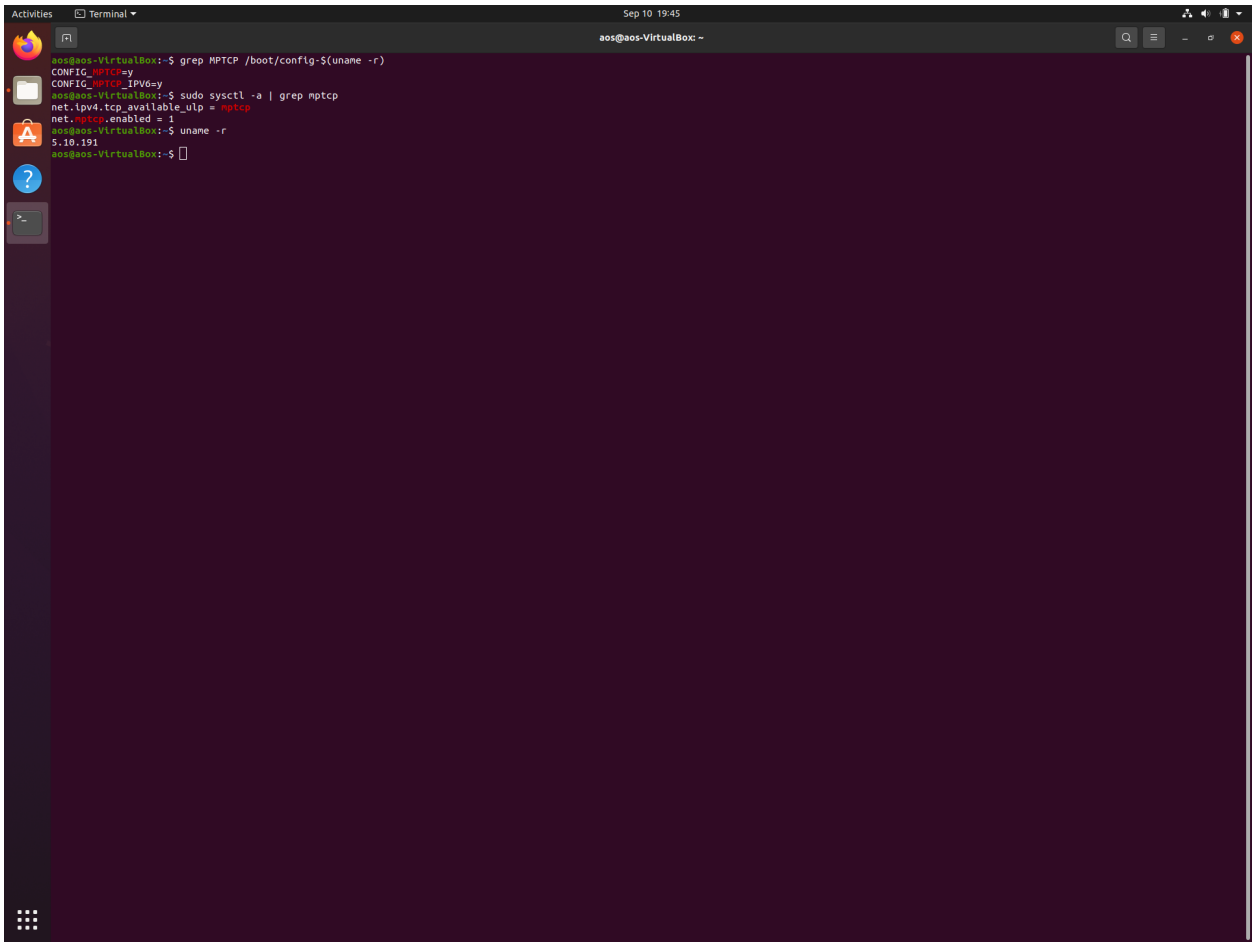
- **Including Multipath TCP (MPTCP)**

  By default, this opinion is already enables in the kernel version 5.15.0 Hence, we did not make any change for it in **menuconfig**.

Before ⇒



After ⇒


Activities
Terminal
Sep 10 19:45
aos@aos-VirtualBox: ~
aos@aos-VirtualBox:~$ grep MPTCP /boot/config-$(uname -r)
CONFIG_MPTCP=y
CONFIG_MPTCP_IPV6=y
aos@aos-VirtualBox:~$ sudo sysctl -a | grep mptcp
net.ipv4.tcp_available_ulp = mptcp
net.mptcp.enabled = 1
aos@aos-VirtualBox:~$ uname -r
5.10.191
aos@aos-VirtualBox:~$