Lab 7 BSY IO 2

image

Task 1 – Compile and Install a Custom Kernel

```
root@io2:~# apt-get install build-essential gcc bc bison flex libssl-dev libncurses5-dev libelf-dev
root@io2:~# mkdir kernel
root@io2:~/kernel# cd kernel
root@io2:~/kernel# wget  https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.17.5.tar.xz
root@io2:~/kernel# xz -cd linux-5.17.5.tar.xz | tar xvf -
root@io2:~/kernel/linux-5.17.5# cd linux-5.17.5/
root@io2:~/kernel/linux-5.17.5# make menuconfig
# > Enable the block layer
#   -> IO Schedulers
#   -> BFQ I/) scheduler y
# > Disable certs
#   -> Cryptographic API
#    -> Certificates for signature checking
#       -> X.509 certificates to be preloaded into the system blacklist keyring
#           Change the 'debian/certs/debian-uefi-certs.pem' string to ''
root@io2:~/kernel/linux-5.17.5# make -j4 deb-pkg
root@io2:~/kernel/linux-5.17.5# cd ..
root@io2:~/kernel# dpkg -i linux-image-5.17.5-hka-bfq_5.17.5-hka-bfq-1_amd64.deb linux-headers-5.17.5-hka-bfq_5.17.5-hka-bfq-1_amd64.deb linux-image-5.17.5-hka-bfq-dbg_5.17.5-hka-bfq-1_amd64.deb linux-libc-dev_5.17.5-hka-bfq-1_amd64.deb
root@io2:~/kernel# reboot
root@io2:~# uname -a
Linux io2 5.17.5-hka-bfq #1 SMP PREEMPT Sat Apr 30 12:54:33 UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
```

Shared IO Access

```
root@io2:~# cat /sys/block/vda/queue/scheduler
[mq-deadline] bfq none
root@io2:~# echo "bfq" | sudo tee /sys/block/vda/queue/scheduler
```

```
bfq
root@io2:~# cat /sys/block/vda/queue/scheduler
mq-deadline [bfq] none

# Still no change, no idea why --> Setting oflag=direct in every dd command should make this work

# Class: Idle
# Don't run if other IO processes are running
root@io2:~# ionice -c 3 dd if=/dev/zero of=/home/ubuntu/testfile_4 bs=1024 count=10000000 &
# Class: Best-effort
root@io2:~# ionice -c 2 -n 0 dd if=/dev/zero of=/home/ubuntu/testfile_1 bs=1024 count=10000000 &
root@io2:~# ionice -c 2 -n 1 dd if=/dev/zero of=/home/ubuntu/testfile_2 bs=1024 count=10000000 &
# Run this after 10 seconds
# This should stop the other IO processes and starve them
# Class: Realtime
root@io2:~# ionice -c 1 -n 0 dd if=/dev/zero of=/home/ubuntu/testfile_3 bs=1024 count=10000000 &
```

Task 2 – Compile and Install a Custom Kernel Module

Build custom kernel module

```
root@io2:~# mkdir chardev
root@io2:~# cd chardev
root@io2:~/chardev# echo "obj-m := chardev.o" | tee Makefile
root@io2:~/chardev# make -C /lib/modules/$(uname -r)/build M=$(pwd) modules
```

Load custom kernel module

```
root@io2:~/chardev# modinfo ./chardev.ko
root@io2:~/chardev# insmod ./chardev.ko
root@io2:~/chardev# lsmod | grep char
chardev                16384  0
root@io2:~/chardev# rmmod chardev
```

- Verify that the module is initialized correctly. Where will you see the messages concerning module initialization?

```
root@io2:~/chardev# dmesg
...
[  263.897504] chardev: loading out-of-tree module taints kernel.
[  263.897815] chardev: module verification failed: signature and/or
required key missing - tainting kernel
[  263.898923] I was assigned major number 240. To talk to
[  263.898929] the driver, create a dev file with
[  263.898931] 'mknod /dev/chardev c 240 0'. # mknod - make block or
character special files
[  263.898934] Try various minor numbers. Try to cat and echo to
[  263.898935] the device file.
[  263.898936] Remove the device file and module when done.
root@io2:~/chardev# mknod /dev/chardev c 240 0
root@io2:~/chardev# mknod /dev/chardev c 240 0
mknod: /dev/chardev: File exists
root@io2:~/chardev# mknod /dev/chardev c 240 1
mknod: /dev/chardev: File exists
root@io2:~/chardev# mknod /dev/chardev c 240 2
mknod: /dev/chardev: File exists
root@io2:~/chardev# mknod /dev/chardev c 240 3
mknod: /dev/chardev: File exists
root@io2:~/chardev# cat /dev/chardev
I already told you 0 times Hello world!
root@io2:~/chardev# cat /dev/chardev
I already told you 1 times Hello world!
root@io2:~/chardev# cat /dev/chardev
I already told you 2 times Hello world!
root@io2:~/chardev# cat /dev/chardev
I already told you 3 times Hello world!
root@io2:~/chardev# cat /dev/chardev
I already told you 4 times Hello world!
root@io2:~/chardev# cat /dev/chardev
I already told you 5 times Hello world!
```

- Look at the /sys system directory and find the module you installed. Check its init state

```
root@io2:~/chardev# ls -l /sys/module/chardev/
total 0
-r--r--r-- 1 root root 4096 Apr 30 15:23 coresize
drwxr-xr-x 2 root root    0 Apr 30 15:23 holders
-r--r--r-- 1 root root 4096 Apr 30 15:24 initsize
-r--r--r-- 1 root root 4096 Apr 30 15:24 initstate
drwxr-xr-x 2 root root    0 Apr 30 15:24 notes
-r--r--r-- 1 root root 4096 Apr 30 15:23 refcnt
drwxr-xr-x 2 root root    0 Apr 30 15:24 sections
-r--r--r-- 1 root root 4096 Apr 30 15:24 srcversion
-r--r--r-- 1 root root 4096 Apr 30 15:24 taint
```

```
--w------- 1 root root 4096 Apr 30 15:23 uevent
root@io2:~/chardev# cat  /sys/module/chardev/initstate
live
```

- Remove the module. What happens to the module directory in /sys?

```
root@io2:~/chardev# rmmod chardev
# File still exists, but cannot be used
root@io2:~/chardev# cat /dev/chardev
cat: /dev/chardev: No such device or address
root@io2:~/chardev# ls -l /sys/module/ | grep char
root@io2:~/chardev#
```