Josiah Norton
20251105

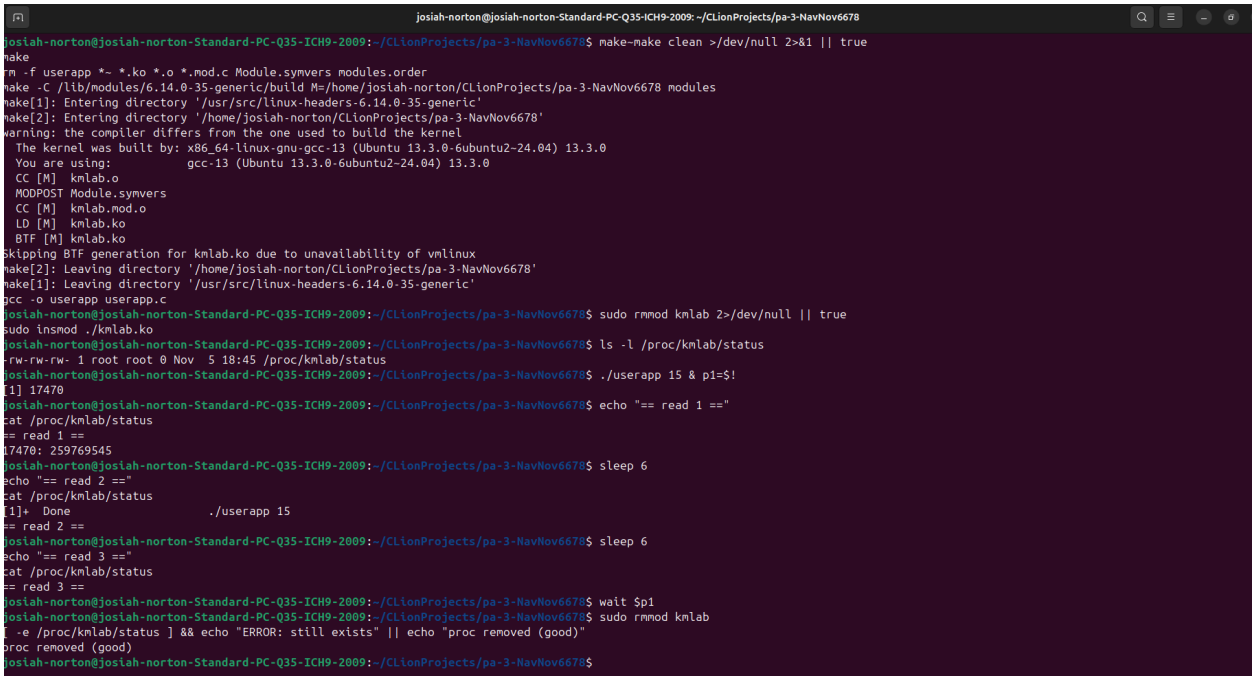
I started this assignment off by setting up a VM and downloading all of the files provided into that VM. Next I ran a quick test by just doing make to see if the given code would compile, which it did. The first thing I made was some global variables to use and keep track of all the different processes I was going to do. After this I made changes to the init and exit functions so that they would make the new files under proc, with the directory and status. The proc/kmlab/status connects the user programs and the kernel module. After this I made kmlab_file_ops to like that file to my read and write functions.

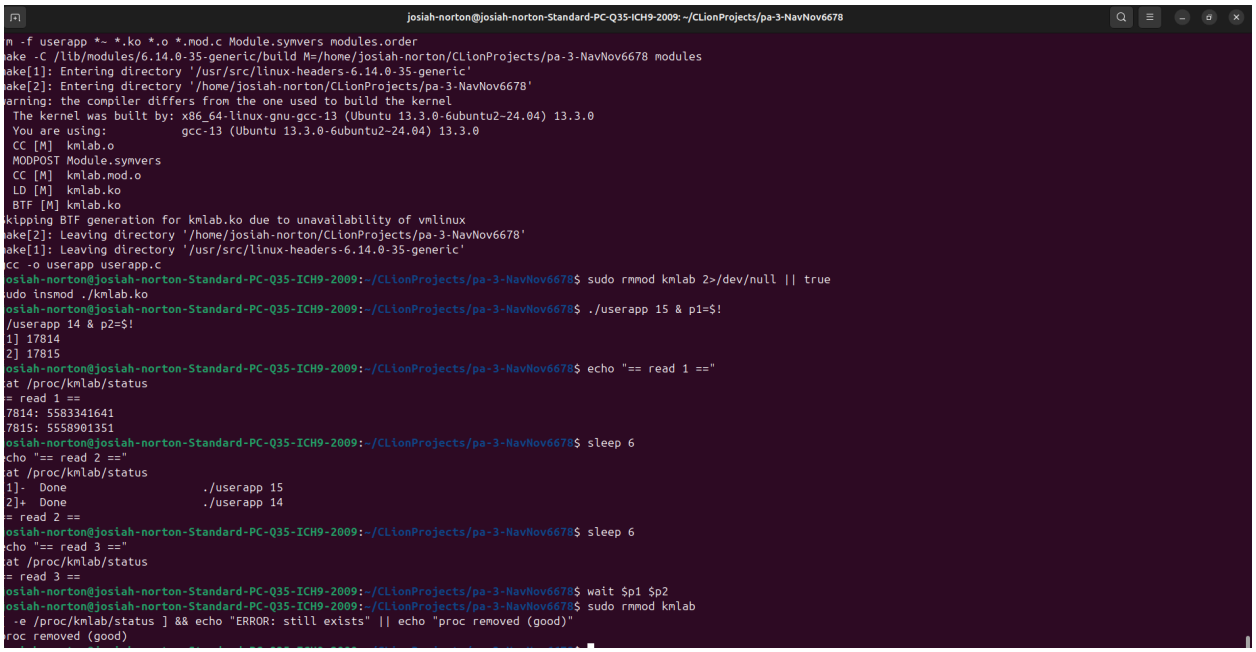
After doing this, I made the aforementioned read and write functions. Write runs whenever something is written into the status. It copies the PID out of the user space, and checks that it's a valid number, then uses the find function I made to see if the PID is already getting tracked. If it is not being tracked, we make a new entry, set the cpu time to zero, and add it to the linked list, proc_list_head under a spinlock. To my understanding the spinlock just holds onto a thread instead of putting it to sleep, that way it just loops till the lock is available. The iqr version of a spinlock just avoids race conditions or deadlocks caused by interruptions.

The read function runs whenever we need to read the file. It allocates a kernel buffer, locks the list, and loops through all the proc_list entries, using snprinf to format each one into a line to display the PID and cpu time. When it finishes this, it unlocks the list, and copies the text back to the terminal using copy to the user.

After finishing the read and write functions, I added background updating. In the init function I added a timer setup and workqueue, the timer runs every 5 seconds, and it queues up the work_fn function. Inside the work_fn function, I loop through the linked list and call the get cpu id function for each process so I can update the cpu times. If this fails because the process does not exist, then I remove the entry from the list and free it.

Finally, my exit functions clean up everything. It deletes the timer, flushes and destroys the work queue, and removes the proc/kmlab folder with the remove proc subtree function. It then free’s everything which is in spinlock and makes sure there are no leaks or crashes when leaving the kernel module.


josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009: ~/CLionProjects/pa-3-NavNov6678
josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ make-make clean >/dev/null 2>&1 || true
make
rm -f userapp *~ *.ko *.o *.mod.c Module.symvers modules.order
make -C /lib/modules/6.14.0-35-generic/build M=/home/josiah-norton/CLionProjects/pa-3-NavNov6678 modules
make[1]: Entering directory '/usr/src/linux-headers-6.14.0-35-generic'
make[2]: Entering directory '/home/josiah-norton/CLionProjects/pa-3-NavNov6678'
warning: the compiler differs from the one used to build the kernel
The kernel was built by: x86_64-linux-gnu-gcc-13 (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0
You are using: gcc-13 (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0
CC [M] kmlab.o
MODPOST Module.symvers
CC [M] kmlab.mod.o
LD [M] kmlab.ko
BTF [M] kmlab.ko
Skipping BTF generation for kmlab.ko due to unavailability of vmlinux
make[2]: Leaving directory '/home/josiah-norton/CLionProjects/pa-3-NavNov6678'
make[1]: Leaving directory '/usr/src/linux-headers-6.14.0-35-generic'
gcc -o userapp userapp.c
josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ sudo rmmod kmlab 2>/dev/null || true
sudo insmod ./kmlab.ko
josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ ls -l /proc/kmlab/status
-rw-rw-rw- 1 root root 0 Nov 5 18:45 /proc/kmlab/status
josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ ./userapp 15 & p1=$!
[1] 17470
josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ echo "== read 1 =="
cat /proc/kmlab/status
== read 1 ==
17470: 259769545
josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ sleep 6
echo "== read 2 =="
cat /proc/kmlab/status
[1]+ Done ./userapp 15
== read 2 ==
josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ sleep 6
echo "== read 3 =="
cat /proc/kmlab/status
== read 3 ==
josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ wait $p1
josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ sudo rmmod kmlab
[-e /proc/kmlab/status] && echo "ERROR: still exists" || echo "proc removed (good)"
proc removed (good)
josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$


josiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009: ~/CLionProjects/pa-3-NavNov6678
m -f userapp *~ *.ko *.o *.mod.c Module.symvers modules.order
ake -C /lib/modules/6.14.0-35-generic/build M=/home/josiah-norton/CLionProjects/pa-3-NavNov6678 modules
ake[1]: Entering directory '/usr/src/linux-headers-6.14.0-35-generic'
ake[2]: Entering directory '/home/josiah-norton/CLionProjects/pa-3-NavNov6678'
arning: the compiler differs from the one used to build the kernel
The kernel was built by: x86_64-linux-gnu-gcc-13 (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0
You are using: gcc-13 (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0
CC [M] kmlab.o
MODPOST Module.symvers
CC [M] kmlab.mod.o
LD [M] kmlab.ko
BTF [M] kmlab.ko
kipping BTF generation for kmlab.ko due to unavailability of vmlinux
ake[2]: Leaving directory '/home/josiah-norton/CLionProjects/pa-3-NavNov6678'
ake[1]: Leaving directory '/usr/src/linux-headers-6.14.0-35-generic'
cc -o userapp userapp.c
osiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ sudo rmmod kmlab 2>/dev/null || true
udo insmod ./kmlab.ko
osiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ ./userapp 15 & p1=$!
/userapp 14 & p2=$!
1] 17814
2] 17815
osiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ echo "== read 1 =="
at /proc/kmlab/status
= read 1 ==
7814: 5583341641
7815: 5558901351
osiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ sleep 6
cho "== read 2 =="
at /proc/kmlab/status
1]- Done ./userapp 15
2]+ Done ./userapp 14
= read 2 ==
osiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ sleep 6
cho "== read 3 =="
at /proc/kmlab/status
= read 3 ==
osiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ wait $p1 $p2
osiah-norton@josiah-norton-Standard-PC-Q35-ICH9-2009:~/CLionProjects/pa-3-NavNov6678$ sudo rmmod kmlab
-e /proc/kmlab/status] && echo "ERROR: still exists" || echo "proc removed (good)"
roc removed (good)