CS 492 Lab 2 Part E Report

Ben Rose

How I added syscalls to the kernel:

Step 1: Created lab2-my-syscall folder to create detailed definitions for how my\_syscall and my\_syscall2 will work, and my\_syscall.c contains these detailed definitions.

Step 2: Edited linux-4.9/include/linux/syscalls.h with my\_syscall and my\_syscall2 function primitives.

Step 3: Added asmlinkage statements for sys\_my\_syscall and sys\_my\_syscall2 to the end of syscalls.h

Step 4: Added my\_syscall and my\_syscall 2 as entries 548 and 549 to arch/x86/entry/syscalls/syscall\_64.tbl, with 64 in place of x32, and sys\_my\_syscall and sys\_my\_syscall2 as the corresponding system call for my\_syscall and my\_syscall2 respectively

Step 5: Added #define \_\_NP\_my\_syscall 548 and #define \_\_NP\_my\_syscall2 549 after #define \_\_NP\_pkey\_fr ee 331 in arch/x86/include/generated/uapi/asm/unistd\_64.h

Step 6: Added \_\_SYSCALL\_64 entries at the end of arch/x86/include/generated/asm/syscalls\_64.h for sys\_my\_syscall and sys\_my\_syscall2

Step 7: run make oldconfig from the Student user, run make -j 3 all from the Student user, run make modules\_install from the Root user, run make install from the Root user, and then run reboot from the Root user, restarting the machine.

After these seven steps were completed, the syscalls have been successfully added.