

Assignment 2

Question 2 Documentation

Program Logic / How the program works:

In order to set up a custom syscall for copying a floating-point 2d matrix into another, we will have to first declare the system call in the **/kernel/sys.c** file.

In the sys.c file I gave 4 parameters to the syscall with the name **kernel_2d_memcpy**, first the copyMatrix in which we have to copy the original matrix, second the originalMatrix, third the number of rows in the original matrix and finally the number of columns in the original matrix.

Then I used the **__copy_from_user()** function to first copy the original matrix into a temporary matrix and then used the function **__copy_to_user()** function to copy the temporary matrix to the copyMatrix and returned 0 from the syscall if successful otherwise I returned -1.

Once the syscall is defined we will go to the **/arch/x86/entry/syscalls/syscall_64.tbl** and add our syscall at number **448**, as a **common** and then syscall as **kernel_2d_memcpy** and **sys_kernel_2d_memcpy** as the syscall name.

Now simply I tested this syscall after recompilation of the kernel using a **test.c** file, In which I passed in the required parameters in the correct order as defined in **sys.c** file during declaration, for the syscall and got the desired output successfully.

Now I just generated the patch file using the **git diff** command on two Linux kernel folders, one with the changes done above and the other one being the stock version the default without any changes).

How to test the syscall:

After patching your Linux kernel with the patch file.

Open the directory with the test.c file and the type **make** to generate the **test** executable.

Now type **./test** to run the executable and see that the copy matrix that is printed is the same as the original matrix