Donghyun Park

Prof. Hakner

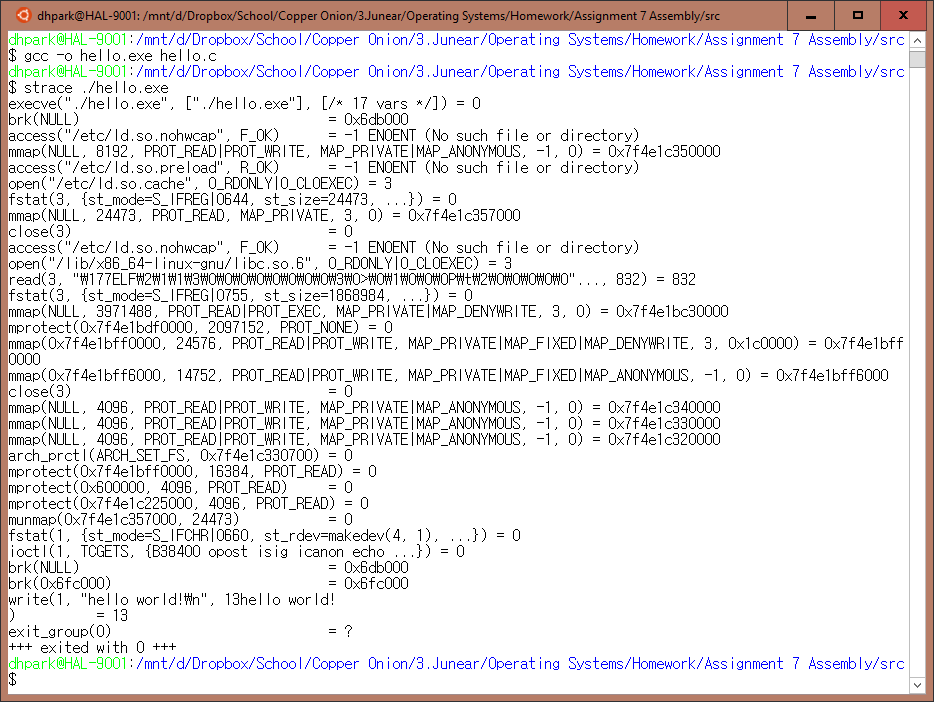
Operating Systems

12/27/17

PSet 7 – Strace and Assembly

Problem 1 – using strace

Not sure why, might be a windows locale thingy, but all \ symbols print out to be the Korean currency symbol ₩ for some reason.



Problem 2 – pure assembly

.text

.global \_start

\_start:

mov $1,%rax #write system call

mov $1,%rdi #output to fd 1(stdout)

mov $msg,%rsi #msg to output

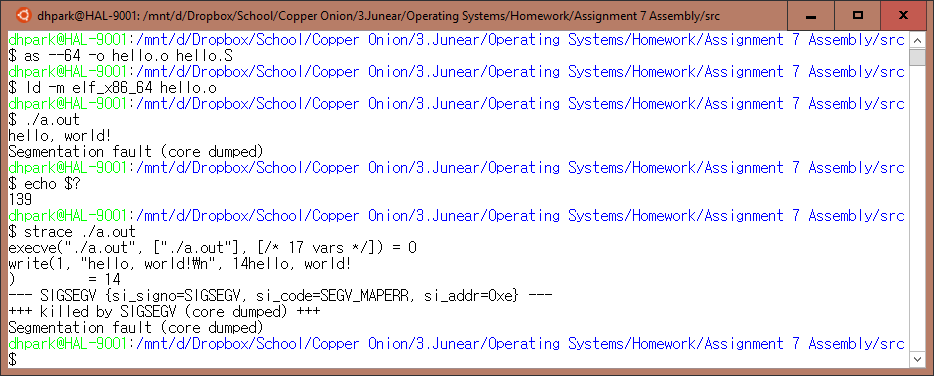
mov $14,%rdx #legnth of msg

syscall

.data

msg:

.ascii "hello, world!\n"



Problem 3 – exit code

After the write system call, the program terminates with the exit value 139 as seen by the echo $?, and the signal was SIGSEGV as seen by the strace. This makes sense because SIGSEGV is value 11, and the UNIX convention follows that the errno value of 128 added to 11 results in 139. The reason behind this error is due to the fact that the %rax register is volatile and thus not available at the program’s termination.

.text

.global \_start

\_start:

mov $1,%rax #write system call

mov $1,%rdi #output to fd 1(stdout)

mov $msg,%rsi #msg to output

mov $14,%rdx #legnth of msg

syscall

mov $1,%rdi #return code for exit syscall

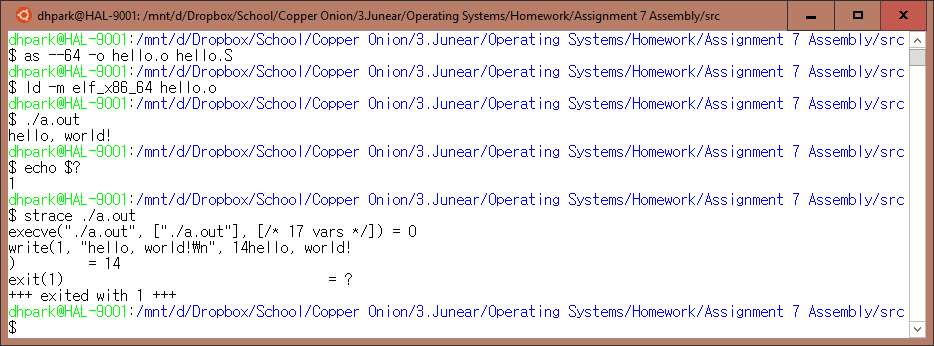
mov $60,%rax #exit system call

syscall

.data

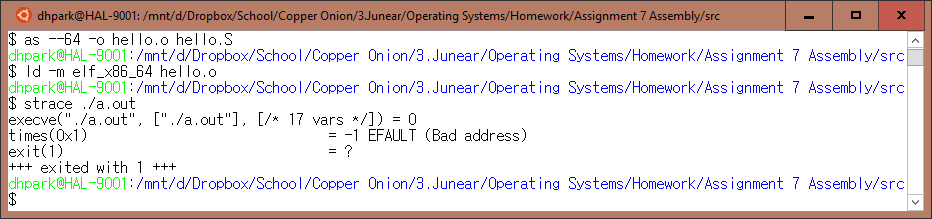
msg:

.ascii "hello, world!\n"



Problem 4 – system call validation

1. Replaced write system call value with 100



1. Passed invalid address for write string

