Ivan Chowdhury

ECE-357: Computer Operating Systems

Fall 2018

Professor Hakner

Problem Set 7 - Using Strace

Problem Set 7 Submission

Problem 2 - Pure Assembly

Build Process & Program Output

```
ubuntu@ubuntu:~/Documents/Source Code/osprog/p7/program$ as -o hello.o --64 hell
o.S
hello.S: Assembler messages:
hello.S: Warning: end of file in comment; newline inserted
ubuntu@ubuntu:~/Documents/Source Code/osprog/p7/program$ ld -m elf_x86_64 -o hel
lo hello.o
ubuntu@ubuntu:~/Documents/Source Code/osprog/p7/program$ ./hello
Hello, world.
Segmentation fault (core dumped)
```

Strace Output

```
ubuntu@ubuntu:~/Documents/Source Code/osprog/p7/program$ strace ./hello
execve("./hello", ["./hello"], 0x7ffee3cc0f50 /* 55 vars */) = 0
write(1, "Hello, world.\n", 14Hello, world.
)         = 14
--- SIGSEGV {si_signo=SIGSEGV, si_code=SI_KERNEL, si_addr=NULL} ---
+++ killed by SIGSEGV (core dumped) +++
Segmentation fault (core dumped)
```

Problem 3 - Exit Code

Build, Execution, Strace and Environment Variable Check

```
ubuntu@ubuntu:~/Documents/Source Code/osprog/p7/program$ as -o hellov2.o --64 he
llov2.S
hellov2.S: Assembler messages:
hellov2.S: Warning: end of file in comment; newline inserted
ubuntu@ubuntu:~/Documents/Source Code/osprog/p7/program$ ld -m elf_x86_64 -o hel
lov2 hellov2.o
ubuntu@ubuntu:~/Documents/Source Code/osprog/p7/program$ ./hellov2
Hello, world.
ubuntu@ubuntu:~/Documents/Source Code/osprog/p7/program$ strace ./hellov2
execve("./hellov2", ["./hellov2"], 0x7ffe5d233680 /* 55 vars */) = 0
write(1, "Hello, world.\n", 14Hello, world.
)         = 14
exit(5)                                 = ?
+++ exited with 5 +++
ubuntu@ubuntu:~/Documents/Source Code/osprog/p7/program$ echo $?
5
```

The first assembly program had no _exit system call invoked, so a segmentation fault occurred after the write system call was used, and the process received a SIGSEGV signal, thus killing the process. The second assembly program had an _exit system call explicitly used, with exit status 5, so the program successfully exits with exit status 5 after the write system call completes; this is verifiable by looking at environment variable $?, as shown above.

Problem 4 - System Call Validation

Strace Output

```
ubuntu@ubuntu:~/Documents/Source Code/osprog/p7/program$ strace ./hellov3
execve("./hellov3", ["./hellov3"], 0x7ffd0a9823b0 /* 55 vars */) = 0
write(1, 0x3e7, 14)                     = -1 EFAULT (Bad address)
exit(5)                                 = ?
+++ exited with 5 +++
```

Strace shows that the program completes execution and exits with exit status 5, similar to problem 3. However, the return value of the write system call is -1, with errno code EFAULT, rather than the number of bytes written like it was for previous problems. This shows that, since an incorrect parameter was used for the write system call (the second argument is meant to be a string containing the printed message, but is instead the number 999 in hexadecimal), the write system call returned -1 on error, and errno was set appropriately.