CS6233: Introduction to Operating Systems

Assignment 2

09-23-23

Kushagra Khatwani (kk5395)

Strace Log

PFA the strace log when the program is run successfully.

lab2_strace_log.log

```
execve("./lab2_1.o", ["./lab2_1.o"], 0x7ffe625d18f0 /brk(NULL)                                 = 0x5555d4719000
arch_prctl(0x3001 /mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fd82ee45000
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=53475, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 53475, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7fd82ee37000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\237\2\0\0\0\0\0"..., 832) = 832
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784
pread64(3, "\4\0\0\0 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) = 48
pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\"\233}\305\t\5?\344\337^)\350b\231\21\360"..., 68, 896) = 68
newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=2216304, ...}, AT_EMPTY_PATH) = 0
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784
mmap(NULL, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fd82ec00000
mmap(0x7fd82ec28000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7fd82ec28000
mmap(0x7fd82edbd000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1bd000) = 0x7fd82edbd000
mmap(0x7fd82ee15000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x214000) = 0x7fd82ee15000
mmap(0x7fd82ee1b000, 52816, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7fd82ee1b000
close(3)                                = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fd82ee34000
arch_prctl(ARCH_SET_FS, 0x7fd82ee34740) = 0
set_tid_address(0x7fd82ee34a10)         = 18202
set_robust_list(0x7fd82ee34a20, 24)     = 0
rseq(0x7fd82ee350e0, 0x20, 0, 0x53053053) = 0
mprotect(0x7fd82ee15000, 16384, PROT_READ) = 0
mprotect(0x5555d3058000, 4096, PROT_READ) = 0
mprotect(0x7fd82ee7f000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192munmap(0x7fd82ee37000, 53475)           = 0
newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}, AT_EMPTY_PATH) = 0
getrandom("\xbd\xe6\x58\xb4\x69\xb5\x91\xb5", 8, GRND_NONBLOCK) = 8
brk(NULL)                               = 0x5555d4719000
brk(0x5555d473a000)                     = 0x5555d473a000
newfstatat(0, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}, AT_EMPTY_PATH) = 0
write(1, "Enter filename: ", 16)        = 16
read(0, "lab2_test.txt\n", 1024)        = 14
write(1, "Entered filename: lab2_test.txt\n", 32) = 32
write(1, "Enter filename: ", 16)        = 16
read(0, "lab2_out.txt\n", 1024)         = 13
write(1, "Entered filename: lab2_out.txt\n", 31) = 31
openat(AT_FDCWD, "lab2_test.txt", O_RDONLY) = 3
openat(AT_FDCWD, "lab2_out.txt", O_WRONLY|O_CREAT|O_EXCL|O_TRUNC, 0666) = 4
newfstatat(3, "", {st_mode=S_IFREG|0664, st_size=27, ...}, AT_EMPTY_PATH) = 0
read(3, "Hello World, this is lab 2\n", 4096) = 27
newfstatat(4, "", {st_mode=S_IFREG|0664, st_size=0, ...}, AT_EMPTY_PATH) = 0
read(3, "", 4096)                       = 0
write(1, "\n", 1)                       = 1
write(1, "Contents of lab2_test.txt transf"..., 68) = 68
write(4, "Hello World, this is lab 2\n", 27) = 27
close(4)                                = 0
close(3)                                = 0
lseek(0, -1, SEEK_CUR)                  = -1 ESPIPE (Illegal seek)
exit_group(0)                           = ?
+++ exited with 0 +++
```

Answers

Answer 1.

Functionality	System Calls
Opening a file	open_at() newfstatat()
Closing a file	close()
Reading a file	read()
writing a file	write()

Answer 2.

Functionality	System Calls
Opening a file	2
Closing a file	1
Reading a file	1
writing a file	1

Answer 3.

The value of file descriptor for read file in my program was **3**. Yes, we can expect the value of read file to change, for instance if write file was opened first then the value of file descriptor would have been different (4 in my case).

Answer 3.

The value of file descriptor for write file in my program was **4**. Yes, we can expect the value of write file to change, for instance if write file was opened first then the value of file descriptor would have been different (3 in my case).