

Московский Авиационный Институт  
(Национальный Исследовательский Университет)

Факультет информационных технологий и прикладной математики  
Кафедра вычислительной математики и программирования

**Лабораторная работа №1 по курсу**  
**«Операционные системы»**

Студент: Яруллин А.Р.  
Группа: М8О–201Б–21  
Вариант: 0  
Преподаватель: Миронов Е.С.  
Оценка: \_\_\_\_\_  
Дата: \_\_\_\_\_  
Подпись: \_\_\_\_\_

Москва, 2022

## Репозиторий

[https://github.com/nesrayr/OS\\_labs](https://github.com/nesrayr/OS_labs)

## Постановка задачи

### Цель работы

Приобретение практических навыков диагностики работы программного обеспечения на примере 3 лабораторной работы.

### Задание

Провести диагностику работы 3 лабораторной работы при помощи strace, объяснить результат работы strace.

### Вывод strace

```
yarullin@yarullin-OMEN-by-HP-Laptop-15-dc1xxx:~/OS-Labs/build/lab3$ cat write.log
```

```
17742 execve("./lab3", ["/lab3"], 0x7ffd73424258 /* 53 vars */) = 0
17742 brk(NULL) = 0x557f1b6c0000
17742 arch_prctl(0x3001 /* ARCH_??? */, 0x7ffc745bcd0) = -1 EINVAL (Invalid argument)
17742 access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)
17742 openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
17742 fstat(3, {st_mode=S_IFREG|0644, st_size=102997, ...}) = 0
17742 mmap(NULL, 102997, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f47370a2000
17742 close(3) = 0
17742 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libpthread.so.0", O_RDONLY|O_CLOEXEC) = 3
17742 read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\220q\0\0\0\0\0"... , 832) = 832
17742 pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0{E6\364\34\332\245\210\204\10\350-\0106\343="..., 68, 824) = 68
17742 fstat(3, {st_mode=S_IFREG|0755, st_size=157224, ...}) = 0
17742 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f47370a0000
17742 pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0{E6\364\34\332\245\210\204\10\350-\0106\343="..., 68, 824) = 68
17742 mmap(NULL, 140408, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f473707d000
17742 mmap(0x7f4737083000, 69632, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x6000) = 0x7f4737083000
17742 mmap(0x7f4737094000, 24576, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x17000) = 0x7f4737094000
17742 mmap(0x7f473709a000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1c000) = 0x7f473709a000
17742 mmap(0x7f473709c000, 13432, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f473709c000
```

```

17742 close(3) = 0
17742 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = 3
17742 read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\341\t\0\0\0\0"..., 832) = 832
17742 fstat(3, {st_mode=S_IFREG|0644, st_size=1956992, ...}) = 0
17742 mmap(NULL, 1972224, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f4736e9b000
17742 mprotect(0x7f4736f31000, 1290240, PROT_NONE) = 0
17742 mmap(0x7f4736f31000, 987136, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x96000) = 0x7f4736f31000
17742 mmap(0x7f4737022000, 299008, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x187000) = 0x7f4737022000
17742 mmap(0x7f473706c000, 57344, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1d0000) = 0x7f473706c000
17742 mmap(0x7f473707a000, 10240, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f473707a000
17742 close(3) = 0
17742 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libgcc_s.so.1", O_RDONLY|O_CLOEXEC) = 3
17742 read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\3405\0\0\0\0\0"..., 832) = 832
17742 fstat(3, {st_mode=S_IFREG|0644, st_size=104984, ...}) = 0
17742 mmap(NULL, 107592, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f4736e80000
17742 mmap(0x7f4736e83000, 73728, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x3000) = 0x7f4736e83000
17742 mmap(0x7f4736e95000, 16384, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x15000) = 0x7f4736e95000
17742 mmap(0x7f4736e99000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x18000) = 0x7f4736e99000
17742 close(3) = 0
17742 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
17742 read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\300A\2\0\0\0\0"..., 832) = 832
17742 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"..., 784, 64) = 784
17742 pread64(3, "\4\0\0\0\20\0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0"..., 32, 848) = 32
17742 pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\30x\346\264ur\0\226\236i\253-'o"..., 68, 880) = 68
17742 fstat(3, {st_mode=S_IFREG|0755, st_size=2029592, ...}) = 0
17742 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"..., 784, 64) = 784
17742 pread64(3, "\4\0\0\0\20\0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0"..., 32, 848) = 32
17742 pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\30x\346\264ur\0\226\236i\253-'o"..., 68, 880) = 68

```

```

17742 mmap(NULL, 2037344, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f4736c8e000
17742      mmap(0x7f4736cb0000,          1540096,          PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x22000) = 0x7f4736cb0000
17742      mmap(0x7f4736e28000,          319488,          PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x19a000) = 0x7f4736e28000
17742      mmap(0x7f4736e76000,          24576,          PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1e7000) = 0x7f4736e76000
17742      mmap(0x7f4736e7c000,          13920,          PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f4736e7c000
17742 close(3) = 0
17742 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libm.so.6", O_RDONLY|O_CLOEXEC) = 3
17742 read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\300\323\0\0\0\0\0"..., 832) = 832
17742 fstat(3, {st_mode=S_IFREG|0644, st_size=1369384, ...}) = 0
17742 mmap(NULL, 1368336, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f4736b3f000
17742      mmap(0x7f4736b4c000,          684032,          PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xd000) = 0x7f4736b4c000
17742      mmap(0x7f4736bf3000,          626688,          PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xb4000) = 0x7f4736bf3000
17742      mmap(0x7f4736c8c000,          8192,          PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x14c000) = 0x7f4736c8c000
17742 close(3) = 0
17742 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7f4736b3d000
17742 mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1,
0) = 0x7f4736b3a000
17742 arch_prctl(ARCH_SET_FS, 0x7f4736b3a740) = 0
17742 mprotect(0x7f4736e76000, 16384, PROT_READ) = 0
17742 mprotect(0x7f4736c8c000, 4096, PROT_READ) = 0
17742 mprotect(0x7f4736e99000, 4096, PROT_READ) = 0
17742 mprotect(0x7f473706c000, 45056, PROT_READ) = 0
17742 mprotect(0x7f473709a000, 4096, PROT_READ) = 0
17742 mprotect(0x557f1b093000, 4096, PROT_READ) = 0
17742 mprotect(0x7f47370e9000, 4096, PROT_READ) = 0
17742 munmap(0x7f47370a2000, 102997) = 0
17742 set_tid_address(0x7f4736b3aa10) = 17742
17742 set_robust_list(0x7f4736b3aa20, 24) = 0

```

[illegible]

## Описание работы

**brk(NULL)** - Устанавливает конец сегмента данных в значение NULL, возвращает указатель на начало новой области памяти

openat() - Открывает относительно дескриптора указанного каталога с правами доступа. Возвращает новый файловый дескриптор .

`mmap()` - Создает отображение файла в память. Возвращает указатель на начало отраженной памяти.

`read()` - Читает данные из файла в буффер указанный вторым аргументом. Возвращает число успешно считанных байт.

`arch_prctl()` - Устанавливает специфичное для архитектуры состояние. Возвращает 0 – успешное выполнение.

**write()** Записывает данные из буфера (второй аргумент) в файл. Возвращает число успешно записанных байт.

**read()** Читает данные в буффер из файл. Возвращает число успешно считанных байт.

clone() - Создает процесс-потомок с флагами, задает положение стека для процесса-потомка, задает указатель на id. Возвращает pid процесса-потомка.

### **Вывод**

При помощи strace можно удобно анализировать работу программы, смотреть на различные системные вызовы их параметры, что каждый вызов возвращает, также можно смотреть системные вызовы по процессам, все это помогает искать неполадки в работе программы и устранять их.