МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ

ФЕДЕРАЦИИ

Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт №8 "Компьютерные науки и прикладная математика"

Кафедра 806 "Вычислительная математика и программирование"

Лабораторная работа №8

По курсу «Операционные системы»

Студент: Григорьев Т. А.

Группа: М8О-208Б-23

Преподаватель: Живалев Е. А.

Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва, 2024

**Тема:** Диагностика программного обеспечения

**Цель работы:** Приобретение практических навыков диагностики работы программного обеспечения.

**Задачи:**

1. Провести диагностику одной из программ, написанных в ходе работ 1-7, с помощью утилиты *strace.*
2. Рассмотреть использовавшиеся системные вызовы.

**Ход работы:**

strace — это утилита для диагностики и отладки программ в операционных системах семейства Unix/Linux. Она позволяет отслеживать системные вызовы и сигналы, которые приложение выполняет или получает во время своей работы.

Трассировка использования *strace* для лабораторной работы №3*:*

execve("./lab3", ["./lab3"], 0x7ffd720bfd80 /\* 75 vars \*/) = 0

brk(NULL) = 0x5ad20e196000

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f47eea38000

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

fstat(3, {st\_mode=S\_IFREG|0644, st\_size=59307, ...}) = 0

mmap(NULL, 59307, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7f47eea29000

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\0\220\243\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

fstat(3, {st\_mode=S\_IFREG|0755, st\_size=2125328, ...}) = 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, 2170256, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f47ee800000

mmap(0x7f47ee828000, 1605632, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000) = 0x7f47ee828000

mmap(0x7f47ee9b0000, 323584, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1b0000) = 0x7f47ee9b0000

mmap(0x7f47ee9ff000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1fe000) = 0x7f47ee9ff000

mmap(0x7f47eea05000, 52624, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7f47eea05000

close(3) = 0

mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f47eea26000

arch\_prctl(ARCH\_SET\_FS, 0x7f47eea26740) = 0

set\_tid\_address(0x7f47eea26a10) = 17757

set\_robust\_list(0x7f47eea26a20, 24) = 0

rseq(0x7f47eea27060, 0x20, 0, 0x53053053) = 0

mprotect(0x7f47ee9ff000, 16384, PROT\_READ) = 0

mprotect(0x5ad20b442000, 4096, PROT\_READ) = 0

mprotect(0x7f47eea70000, 8192, PROT\_READ) = 0

prlimit64(0, RLIMIT\_STACK, NULL, {rlim\_cur=8192\*1024, rlim\_max=RLIM64\_INFINITY}) = 0

munmap(0x7f47eea29000, 59307) = 0

unlink("/dev/shm/sem.sem\_read1") = 0

unlink("/dev/shm/sem.sem\_read2") = 0

unlink("/dev/shm/sem.sem\_write") = 0

openat(AT\_FDCWD, "/dev/shm/sem.sem\_read1", O\_RDWR|O\_NOFOLLOW|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

getrandom("\xa3\xee\x00\x9b\x06\xaf\xac\x82", 8, GRND\_NONBLOCK) = 8

newfstatat(AT\_FDCWD, "/dev/shm/sem.BOoKWc", 0x7ffd8cdcb400, AT\_SYMLINK\_NOFOLLOW) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/dev/shm/sem.BOoKWc", O\_RDWR|O\_CREAT|O\_EXCL|O\_NOFOLLOW|O\_CLOEXEC, 0666) = 3

write(3, "\0\0\0\0\0\0\0\0\200\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0", 32) = 32

mmap(NULL, 32, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 3, 0) = 0x7f47eea37000

link("/dev/shm/sem.BOoKWc", "/dev/shm/sem.sem\_read1") = 0

fstat(3, {st\_mode=S\_IFREG|0664, st\_size=32, ...}) = 0

getrandom("\xef\x70\xd9\xbd\xd1\x2c\x37\x96", 8, GRND\_NONBLOCK) = 8

brk(NULL) = 0x5ad20e196000

brk(0x5ad20e1b7000) = 0x5ad20e1b7000

unlink("/dev/shm/sem.BOoKWc") = 0

close(3) = 0

openat(AT\_FDCWD, "/dev/shm/sem.sem\_read2", O\_RDWR|O\_NOFOLLOW|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

getrandom("\xea\x16\x5c\x6e\x1b\x51\xf1\xc3", 8, GRND\_NONBLOCK) = 8

newfstatat(AT\_FDCWD, "/dev/shm/sem.4No7af", 0x7ffd8cdcb400, AT\_SYMLINK\_NOFOLLOW) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/dev/shm/sem.4No7af", O\_RDWR|O\_CREAT|O\_EXCL|O\_NOFOLLOW|O\_CLOEXEC, 0666) = 3

write(3, "\0\0\0\0\0\0\0\0\200\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0", 32) = 32

mmap(NULL, 32, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 3, 0) = 0x7f47eea36000

link("/dev/shm/sem.4No7af", "/dev/shm/sem.sem\_read2") = 0

fstat(3, {st\_mode=S\_IFREG|0664, st\_size=32, ...}) = 0

unlink("/dev/shm/sem.4No7af") = 0

close(3) = 0

openat(AT\_FDCWD, "/dev/shm/sem.sem\_write", O\_RDWR|O\_NOFOLLOW|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

getrandom("\x5c\x3d\xf4\x49\x1e\xee\x46\xd0", 8, GRND\_NONBLOCK) = 8

newfstatat(AT\_FDCWD, "/dev/shm/sem.KId41p", 0x7ffd8cdcb400, AT\_SYMLINK\_NOFOLLOW) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/dev/shm/sem.KId41p", O\_RDWR|O\_CREAT|O\_EXCL|O\_NOFOLLOW|O\_CLOEXEC, 0666) = 3

write(3, "\0\0\0\0\0\0\0\0\200\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0", 32) = 32

mmap(NULL, 32, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 3, 0) = 0x7f47eea35000

link("/dev/shm/sem.KId41p", "/dev/shm/sem.sem\_write") = 0

fstat(3, {st\_mode=S\_IFREG|0664, st\_size=32, ...}) = 0

unlink("/dev/shm/sem.KId41p") = 0

close(3) = 0

openat(AT\_FDCWD, "/dev/shm/memory", O\_RDWR|O\_CREAT|O\_NOFOLLOW|O\_CLOEXEC, 0666) = 3

ftruncate(3, 1024) = 0

mmap(NULL, 1024, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 3, 0) = 0x7f47eea34000

fstat(0, {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0x4), ...}) = 0

read(0, out1

"out1\n", 1024) = 5

read(0, out2

"out2\n", 1024) = 5

clone(child\_stack=NULL, flags=CLONE\_CHILD\_CLEARTID|CLONE\_CHILD\_SETTID|SIGCHLD, child\_tidptr=0x7f47eea26a10) = 17812

clone(child\_stack=NULL, flags=CLONE\_CHILD\_CLEARTID|CLONE\_CHILD\_SETTID|SIGCHLD, child\_tidptr=0x7f47eea26a10) = 17813

read(0, fsdkljfksdjfdksljldk

"fsdkljfksdjfdksljldk\n", 1024) = 21

futex(0x7f47eea36000, FUTEX\_WAKE, 1) = 1

futex(0x7f47eea35000, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 0, NULL, FUTEX\_BITSET\_MATCH\_ANY) = -1 EAGAIN (Resource temporarily unavailable)

read(0, sdlkfjlksdjfkldsjlkfsd

"sdlkfjlksdjfkldsjlkfsd\n", 1024) = 23

futex(0x7f47eea36000, FUTEX\_WAKE, 1) = 1

futex(0x7f47eea35000, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 0, NULL, FUTEX\_BITSET\_MATCH\_ANY) = -1 EAGAIN (Resource temporarily unavailable)

read(0, sdjlfsdjlkfjsdlkfsjdklsjdk

"sdjlfsdjlkfjsdlkfsjdklsjdk\n", 1024) = 27

futex(0x7f47eea36000, FUTEX\_WAKE, 1) = 1

futex(0x7f47eea35000, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 0, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0

read(0, sdfjsdnfncxnmsfd

"sdfjsdnfncxnmsfd\n", 1024) = 17

futex(0x7f47eea36000, FUTEX\_WAKE, 1) = 1

read(0, q

"q\n", 1024) = 2

futex(0x7f47eea37000, FUTEX\_WAKE, 1) = 1

futex(0x7f47eea36000, FUTEX\_WAKE, 1) = 1

wait4(-1, NULL, 0, NULL) = 17812

--- SIGCHLD {si\_signo=SIGCHLD, si\_code=CLD\_EXITED, si\_pid=17812, si\_uid=1000, si\_status=0, si\_utime=0, si\_stime=0} ---

wait4(-1, NULL, 0, NULL) = 17813

munmap(0x7f47eea37000, 32) = 0

unlink("/dev/shm/sem.sem\_read1") = 0

munmap(0x7f47eea36000, 32) = 0

unlink("/dev/shm/sem.sem\_read2") = 0

munmap(0x7f47eea35000, 32) = 0

unlink("/dev/shm/sem.sem\_write") = 0

munmap(0x7f47eea34000, 1024) = 0

close(3) = 0

unlink("/dev/shm/memory") = 0

exit\_group(0) = ?

+++ exited with 0 +++

Использовавшиеся системные вызовы:

*1.* **execve** - Запускает программу, заменяя текущий процесс новым.

*2.* **brk** - Управляет размером кучи процесса, выделяя или освобождая память.

3. **mmap** - Отображает файл или область памяти в адресное пространство процесса, используется для выделения памяти.

4. **access** - Проверяет доступность файла для чтения, записи или выполнения.

5. **openat** - Открывает файл относительно указанной директории.

6. **fstat** - Получает информацию о файле, такую как размер и права доступа.

7. **pread64** - Читает данные из файла с указанным смещением.

8. **close** - Закрывает файловый дескриптор, освобождая ресурсы.

9. **arch**\_**prctl** - Устанавливает значение регистра FS, используется для управления потоками.

10. **set\_tid\_address** - Устанавливает адрес для хранения идентификатора потока.

11. **mprotect** - Изменяет права доступа к области памяти, например, защищает её от записи.

12. **prlimit64** - Получает или устанавливает ограничения на ресурсы процесса, такие как размер стека.

13. **unlink** - Удаляет файл, используется для очистки временных файлов или семафоров.

14. **getrandom** - Генерирует случайные данные из системного генератора случайных чисел.

15. **clone** - Создает новый поток или процесс для параллельной работы.

16. **futex** - Управляет синхронизацией между потоками, например, ожиданием или пробуждением.

17. **wait4** - Ожидает завершения дочернего процесса или потока.

18. **exit\_group** - Завершает выполнение всех потоков процесса, завершая программу.

*Эти системные вызовы показывают, как программа взаимодействует с ядром Linux для выполнения задач, таких как управление памятью, работа с файлами, синхронизация потоков и завершение работы.*

**Вывод:** В ходе лабораторной работы была проведена диагностика программы с использованием инструмента strace, что позволило проанализировать системные вызовы, обеспечивающие взаимодействие программы с ресурсами операционной системы. В процессе исследования были изучены механизмы работы с памятью, синхронизацией процессов, вводом-выводом и обменом сообщениями. Полученные результаты подтвердили корректность работы программы и её соответствие теоретическим ожиданиям. Анализ системных вызовов продемонстрировал их ключевую роль в обеспечении стабильного функционирования программы и её взаимодействия с операционной системой..