

Московский Авиационный Институт
(Национальный Исследовательский Университет)

Факультет информационных технологий и прикладной математики
Кафедра вычислительной математики и программирования

**Лабораторная работа №8 по курсу
«Операционные системы»**

**ДИАГНОСТИКА РАБОТЫ
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Студент: Слесарчук Василий Анатольевич

Группа: М8О–210Б–22

Вариант:

Преподаватель: Соколов Андрей Алексеевич

Оценка: _____

Дата: _____

Подпись: _____

Москва, 2023.

Постановка задачи

Цель работы

Целью является приобретение практических навыков в диагностике работы программы.

Задание

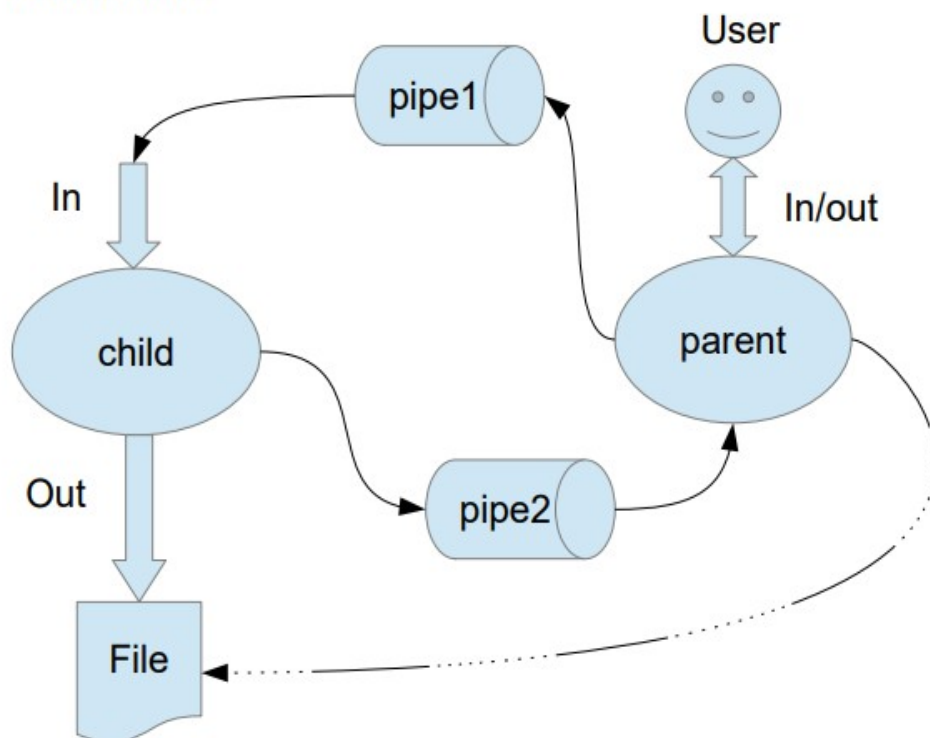
При выполнении лабораторных работ по курсу ОС необходимо продемонстрировать ключевые системные вызовы, которые в них используются и то, что их использование соответствует варианту ЛР.

Общий метод и алгоритм решения.

Для выполнения лабораторной работы необходимо:

1. Изучить основные системные вызовы.
2. Вспомнить логику написанной программы и места в которых будут происходить специфические системные вызовы.

Группа вариантов 4



Пример работы

```
> strace -e trace=mmap,openat,write,read,close -o trace.info ./
```

```
write path to file
```

```
SUS
```

```
Enter something strings. If you want to stop, press Ctrl+D
```

```
Aboba666
```

```
AMONGUS
```

```
among us
```

```
5U5Y BAKA
```

```
String start with uppercase
```

```
String start with uppercase
```

```
String does not start with uppercase
```

```
String does not start with uppercase
```

```
[trace.info]
```

```
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f5340221000
```

```
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
```

```
mmap(NULL, 110759, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f5340205000
```

```
close(3) = 0
```

```
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = 3
```

```
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0"... , 832) = 832
```

```
mmap(NULL, 2275520, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f533fe00000
```

```
mmap(0x7f533fe9a000, 1118208, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x9a000) = 0x7f533fe9a000
```

```
mmap(0x7f533ffab000, 454656, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1ab000) = 0x7f533ffab000
```

```
mmap(0x7f534001b000, 57344, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x21a000) = 0x7f534001b000
```

```
mmap(0x7f5340029000, 10432, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f5340029000
```

```
close(3) = 0
```

```
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libgcc_s.so.1", O_RDONLY|O_CLOEXEC) = 3
```

```
read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0"... , 832) = 832
```

```
mmap(NULL, 127720, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f53401e5000
```

```
mmap(0x7f53401e8000, 94208, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x3000) = 0x7f53401e8000
```

```
mmap(0x7f53401ff000, 16384, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1a000) = 0x7f53401ff000
```

```
mmap(0x7f5340203000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1d000) = 0x7f5340203000
```

```
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\3\0>\0\1\0\0\0\0\0\0\0\0\0"... , 832) = 832
```

```
mmap(NULL, 2264656, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f533fa00000
```

```
mmap(0x7f533fa28000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|
MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7f533fa28000
mmap(0x7f533fbbd000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x1bd000) = 0x7f533fbbd000
mmap(0x7f533fc16000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x215000) = 0x7f533fc16000
mmap(0x7f533fc1c000, 52816, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_ANONYMOUS, -1, 0) = 0x7f533fc1c000
close(3) = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libm.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832
mmap(NULL, 942344, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f53400fe000
mmap(0x7f534010c000, 507904, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0xe000) = 0x7f534010c000
mmap(0x7f5340188000, 372736, PROT_READ, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x8a000) = 0x7f5340188000
mmap(0x7f53401e3000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0xe4000) = 0x7f53401e3000
close(3) = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -
1, 0) = 0x7f53400fc000
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -
1, 0) = 0x7f53400fa000
write(1, "write path to file\n", 19) = 19
read(0, "SUS\n", 1024) = 4
openat(AT_FDCWD, "SUS", O_WRONLY|O_CREAT|O_APPEND, 0666) = 3
openat(AT_FDCWD, "/dev/shm/input", O_RDWR|O_CREAT|O_NOFOLLOW|O_CLOEXEC,
0666) = 4
mmap(NULL, 1024, PROT_READ|PROT_WRITE, MAP_SHARED, 4, 0) = 0x7f534025a000
close(4) = 0
openat(AT_FDCWD, "/dev/shm/error", O_RDWR|O_CREAT|O_NOFOLLOW|O_CLOEXEC,
0666) = 4
mmap(NULL, 1024, PROT_READ|PROT_WRITE, MAP_SHARED, 4, 0) = 0x7f5340220000
close(4) = 0
write(1, "Enter something strings. If you "..., 59) = 59
read(0, "Aboba666\n", 1024) = 9
read(0, "AMONGUS\n", 1024) = 8
read(0, "among us\n", 1024) = 9
read(0, "5U5Y BAKA\n", 1024) = 10
read(0, "", 1024) = 0
--- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=44143, si_uid=1000,
si_status=0, si_utime=0, si_stime=0} ---
write(1, "String start with uppercase\n", 28) = 28
write(1, "String start with uppercase\n", 28) = 28
write(1, "String does not start with upper"..., 37) = 37
write(1, "String does not start with upper"..., 37) = 37
+++ exited with 0 +++
```

Вывод

При отладке при помощи strace видно, что код написанный мной начинает исполняться со строки `write(1, "write path to file\n", 19) = 19`. Можно следить за тем, как открываются и закрываются общие области памяти и как идет запись в файлы. В ходе выполнения ЛР я создал файл вывода с правами супер пользователя, из за чего он оставался пустым. Проанализировав вывод команды strace я быстро отладил ошибку. Системные вызовы перед моей программой осуществляют загрузку библиотек. В ходе этой лабораторной работы я научился диагностировать работу программного обеспечения при помощи консольной утилиты strace.