

Московский Авиационный Институт
(Национальный Исследовательский Университет)
Факультет информационных технологий и прикладной математики
Кафедра вычислительной математики и программирования

Лабораторная работа №1 по курсу
«Операционные системы»

Тема работы
“Использование утилиты strace”

Студент: Ханнанов Руслан Маратович
Группа: М8О-208Б-20
Вариант: -
Преподаватель: Миронов Евгений Сергеевич
Оценка: _____
Дата: _____
Подпись: _____

Москва, 2021

Содержание

1. Репозиторий
2. Постановка задачи
3. Демонстрация работы с утилитой `strace` и подробное объяснение каждого системного вызова
4. Выводы

Репозиторий

<https://github.com/Naksen/OS>

Постановка задачи

Задача: подробно рассказать о каждом системном вызове, отображенном утилитой strace. Разбор системных вызовов осуществляется на примере четвертой лабораторной работы курса “Операционные системы”.

Демонстрация работы с утилитой strace и подробное объяснение каждого системного вызова

Исходный код strace:

```
execve("./main_1", ["/main_1"], 0x7fffd3ecc270 /* 28 vars */) = 0
brk(NULL) = 0x7fffd864000
(0x3001 /* ARCH_??? */, 0x7ffff4e15910) = -1 EINVAL (Invalid argument)
access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/tls/haswell/x86_64/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/tls/haswell/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/tls/x86_64/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/tls/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/haswell/x86_64/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/haswell/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/x86_64/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/libd1.so", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\200\20\0\0\0\0\0"... , 832) = 832
fstat(3, {st_mode=S_IFREG|0755, st_size=16264, ...}) = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f320a770000
getcwd("/home/naksan/cprog/5_lab_OS", 128) = 28
mmap(NULL, 16440, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f320a730000
3
```

```

mmap(0x7f320a731000, 4096, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x7f320a731000

mmap(0x7f320a732000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x2000) = 0x7f320a732000

mmap(0x7f320a733000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f320a733000

close(3) = 0

openat(AT_FDCWD, "/tls/haswell/x86_64/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No
such file or directory)

openat(AT_FDCWD, "/tls/haswell/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file
or directory)

openat(AT_FDCWD, "/tls/x86_64/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file
or directory)

openat(AT_FDCWD, "/tls/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "/haswell/x86_64/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No
such file or directory)

openat(AT_FDCWD, "/haswell/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "/x86_64/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "/libc.so.6", O_RDONLY|O_CLOEXEC) = -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=33622, ...}) = 0

mmap(NULL, 33622, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f320a776000

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\3\0>\0\1\0\0\360q\2\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"..., 784, 64) = 784

pread64(3, "\4\0\0\0\20\0\0\05\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0", 32, 848) = 32

pread64(3, "\4\0\0\0\24\0\0\03\0\0\0GNU\0\t\233\222%\274\260\320\31\331\326\10\204\276X>\263"...,
68, 880) = 68

fstat(3, {st_mode=S_IFREG|0755, st_size=2029224, ...}) = 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"..., 784, 64) = 784

pread64(3, "\4\0\0\0\20\0\0\05\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0", 32, 848) = 32

pread64(3, "\4\0\0\0\24\0\0\03\0\0\0GNU\0\t\233\222%\274\260\320\31\331\326\10\204\276X>\263"...,
68, 880) = 68

```

```

mmap(NULL, 2036952, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f320a530000
mprotect(0x7f320a555000, 1847296, PROT_NONE) = 0
mmap(0x7f320a555000, 1540096, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x25000) = 0x7f320a555000
mmap(0x7f320a6cd000, 303104, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x19d000) = 0x7f320a6cd000
mmap(0x7f320a718000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1e7000) = 0x7f320a718000
mmap(0x7f320a71e000, 13528, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f320a71e000
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\300\363\0\0\0\0\0"..., 832) = 832
fstat(3, {st_mode=S_IFREG|0644, st_size=1369352, ...}) = 0
mmap(NULL, 1368336, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f320a3e1000
mmap(0x7f320a3f0000, 684032, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xf000) = 0x7f320a3f0000
mmap(0x7f320a497000, 618496, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0xb6000) = 0x7f320a497000
mmap(0x7f320a52e000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x14c000) = 0x7f320a52e000
close(3) = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f320a3d0000
arch_prctl(ARCH_SET_FS, 0x7f320a3d0740) = 0
mprotect(0x7f320a718000, 12288, PROT_READ) = 0
mprotect(0x7f320a52e000, 4096, PROT_READ) = 0
mprotect(0x7f320a733000, 4096, PROT_READ) = 0
mprotect(0x7f320a782000, 4096, PROT_READ) = 0
mprotect(0x7f320a76d000, 4096, PROT_READ) = 0
munmap(0x7f320a776000, 33622) = 0
fstat(0, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
brk(NULL) = 0x7ffed864000
brk(0x7ffed885000) = 0x7ffed885000
read(0, "2\n", 1024) = 2
read(0, "10\n", 1024) = 3

```

```
fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
```

```
write(1, "E(10) = 2.593742\n", 17)    = 17
```

```
read(0, "", 1024)                    = 0
```

```
exit_group(0)                        = ?
```

```
+++ exited with 0 +++
```

Разбор команд:

execve(...) - запускает программу

brk(...) - устанавливает конец сегмента данных в значение NULL

access(...) - проверяет права доступа к файлу, возвращает -1 как код ошибки

openat(...) - открывает файл, имеет в качестве возвращаемого значения файловый дескриптор

fstat(...) - собирает информацию из файла

mmap(...) - отображает файл на память

mprotect(...) - контролирует доступ к памяти

close(...) - закрывает файловый дескриптор

read(...) - считывает из файлового дескриптора

arch_prctl(...) - устанавливает специфичное для архитектуры значение ядра

munmap(...) - освобождает память, отведенную для отображения файла

write(...) - пишет в консоль

ftruncate(...) - увеличивает размер файла до n байтов

clone(...) - создает новый процесс

unlink(...) - удаляет файл

lseek(...) - устанавливает смещение для позиционирования операций чтения/записи

Выводы

В данной лабораторной работе я научился пользоваться strace и с помощью его отслеживать системные вызовы. Изучил сами системные вызовы и для чего они используются.