Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт №8 “Компьютерные науки и прикладная математика”

Кафедра №806 “Вычислительная математика и программирование”

Лабораторная работа №6 по курсу

«Операционные системы»

Группа: М8О-209БВ-24

Студент: Махова А.Б.

Преподаватель: Миронов Е.С.

Оценка: _____

Дата: 19.12.25

Москва, 2025

Постановка задачи

Приобретение практических навыков диагностики работы программного обеспечения. 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

Утилита **strace** предназначена для трассировки системных вызовов, выполняемых процессами в операционной системе Linux. Системные вызовы являются основным механизмом взаимодействия пользовательских программ с ядром ОС. Изучаемая утилита отображает информацию о каждом системном вызове, включая его имя, переданные аргументы, возвращаемое значение и возможные ошибки. Работа strace основана на механизме, который позволяет одному процессу отслеживать выполнение другого. При каждом системном вызове выполнение трассируемой программы временно приостанавливается, после чего strace выводит соответствующую информацию. Использование утилиты strace позволяет анализировать поведение программ, выявлять причины ошибок, изучать взаимодействие процессов с ядром операционной системы и наглядно исследовать принципы работы ОС в рамках лабораторных работ.

При работе с утилитой strace могут использоваться следующие **основные флаги:**

-f позволяет отслеживать системные вызовы дочерних процессов, создаваемых в ходе выполнения программы;

-o <файл> используется для сохранения вывода трассировки в указанный файл, что удобно для последующего анализа и оформления отчётов;

-p <pid> применяется для подключения к уже запущенному процессу по его идентификатору;

-e trace=<набор> ограничивает вывод только указанными системными вызовами или их группами, что уменьшает объём выводимой информации;

-c выводит статистику по количеству и времени выполнения системных вызовов;

-t и -tt добавляют временные метки к системным вызовам, позволяя анализировать временные характеристики работы программы;

-s <число> задаёт максимальную длину выводимых строк.

Таким образом, утилита strace является эффективным средством изучения системных вызовов и механизмов взаимодействия пользовательских программ с ядром операционной системы.

Выводы strace

Лабораторная работа №1

vscode ➜ /workspaces/MAI_OS/lab1/src (main) $ strace -f ./parent

execve("./parent", ["./parent"], 0xffffffb6a68c8 /* 35 vars */) = 0 < запуск родительского процесса

brk(NULL) = 0xaaaae1fc9000

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffffb49b4000

faccessat(AT_FDCWD, "/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=15048, ...}, AT_EMPTY_PATH) = 0

mmap(NULL, 15048, PROT_READ, MAP_PRIVATE, 3, 0) = 0xffffb49b0000

close(3) = 0

openat(AT_FDCWD, "/lib/aarch64-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\267\0\1\0\0\0\340u\2\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=1637400, ...}, AT_EMPTY_PATH) = 0

mmap(NULL, 1805928, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffffb47c6000

mmap(0xffffb47d0000, 1740392, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffffb47d0000

munmap(0xffffb47c6000, 40960) = 0

munmap(0xffffb4979000, 24168) = 0

mprotect(0xffffb4958000, 61440, PROT_NONE) = 0

mmap(0xffffb4967000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x187000) = 0xffffb4967000

mmap(0xffffb496d000, 48744, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0xffffb496d000

close(3) = 0

set_tid_address(0xffffb49b4f50)   = 5974

set_robust_list(0xffffb49b4f60, 24)   = 0

rseq(0xffffb49b5620, 0x20, 0, 0xd428bc00) = 0

mprotect(0xffffb4967000, 16384, PROT_READ) = 0

mprotect(0xaaaad5661000, 4096, PROT_READ) = 0

mprotect(0xffffb49b9000, 8192, PROT_READ) = 0

prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0

munmap(0xffffb49b0000, 15048)   = 0

write(1, "Input file: ", 12Input file:)   = 12

read(0, test.txt

"test.txt\n", 1023)   = 9

pipe2([3, 4], 0)   = 0 < pipe

clone(child_stack=NULL,
flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLDstrace: Process 6076 attached

, child_tidptr=0xffffb49b4f50) = 6076

[pid 5974] close(4 <unfinished ...>

[pid 6076] set_robust_list(0xffffb49b4f60, 24 <unfinished ...>

[pid 5974] <... close resumed>)   = 0

[pid 5974] read(3, <unfinished ...>

[pid 6076] <... set_robust_list resumed>) = 0

[pid 6076] close(3)   = 0

[pid 6076] openat(AT_FDCWD, "test.txt", O_RDONLY) = 3

[pid 6076] dup3(3, 0, 0)   = 0 < dup2(stdin)

[pid 6076] close(3)   = 0

[pid 6076] dup3(4, 1, 0)   = 1 < dup2(stdout)

[pid 6076] close(4)   = 0

[pid 6076] execve("./child", ["child"], 0xffffeb613658 /* 35 vars */) = 0 < запуск дочернего процесса

[pid 6076] brk(NULL)   = 0xaaaae9cb8000

[pid 6076] mmap(NULL, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffff9d513000

[pid 6076] faccessat(AT_FDCWD, "/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)

[pid 6076] openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3

[pid 6076] newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=15048, ...}, AT_EMPTY_PATH) = 0

[pid 6076] mmap(NULL, 15048, PROT_READ, MAP_PRIVATE, 3, 0) = 0xffff9d50f000

[pid 6076] close(3)   = 0

[pid 6076] openat(AT_FDCWD, "/lib/aarch64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3

[pid 6076] read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\340u\2\0\0\0\0"..., 832) = 832

[pid 6076] newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=1637400, ...}, AT_EMPTY_PATH) = 0

[pid 6076] mmap(NULL, 1805928, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffff9d325000

[pid 6076] mmap(0xffff9d330000, 1740392, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffff9d330000

[pid 6076] munmap(0xffff9d325000, 45056) = 0

[pid 6076] munmap(0xffff9d4d9000, 20072) = 0

[pid 6076] mprotect(0xffff9d4b8000, 61440, PROT_NONE) = 0

[pid 6076] mmap(0xffff9d4c7000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x187000) = 0xffff9d4c7000

[pid 6076] mmap(0xffff9d4cd000, 48744, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0xffff9d4cd000

[pid 6076] close(3)   = 0

[pid 6076] set_tid_address(0xffff9d513f50) = 6076

[pid 6076] set_robust_list(0xffff9d513f60, 24) = 0

[pid 6076] rseq(0xffff9d514620, 0x20, 0, 0xd428bc00) = 0

[pid 6076] mprotect(0xffff9d4c7000, 16384, PROT_READ) = 0

[pid 6076] mprotect(0xaaaace2b1000, 4096, PROT_READ) = 0

[pid 6076] mprotect(0xffff9d518000, 8192, PROT_READ) = 0

[pid 6076] prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0

[pid 6076] munmap(0xffff9d50f000, 15048) = 0

[pid 6076] read(0, "12 47 -3 42\n8 12 46\n90 102 -90 2"..., 4095) = 43

[pid 6076] write(1, "98\n", 3 <unfinished ...>

[pid 5974] <... read resumed>"98\n", 4096) = 3

[pid 6076] <... write resumed>)   = 3

[pid 5974] write(1, "98\n", 3 <unfinished ...>

[pid 6076] write(1, "66\n", 398

<unfinished ...>

[pid 5974] <... write resumed>) = 3

[pid 6076] <... write resumed>) = 3

[pid 5974] read(3, <unfinished ...>

[pid 6076] write(1, "104\n", 4 <unfinished ...>

[pid 5974] <... read resumed>"66\n", 4096) = 3

[pid 6076] <... write resumed>) = 4

[pid 5974] write(1, "66\n", 3 <unfinished ...>

[pid 6076] write(1, "102\n", 466

<unfinished ...>

[pid 5974] <... write resumed>) = 3

[pid 6076] <... write resumed>) = 4

[pid 5974] read(3, <unfinished ...>

[pid 6076] read(0, <unfinished ...>

[pid 5974] <... read resumed>"104\n102\n", 4096) = 8

[pid 6076] <... read resumed>"", 4095) = 0

[pid 5974] write(1, "104\n102\n", 8 <unfinished ...>

[pid 6076] exit_group(0104

102

<unfinished ...>

[pid 5974] <... write resumed>) = 8

[pid 6076] <... exit_group resumed>) = ?

[pid 5974] read(3, "", 4096) = 0

[pid 5974] close(3 <unfinished ...>

[pid 6076] +++ exited with 0 +++

<... close resumed>) = 0

--- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=6076, si_uid=1000, si_status=0, si_utime=0, si_stime=0} ---

wait4(-1, NULL, 0, NULL) = 6076

exit_group(0) = ?

+++ exited with 0 +++

clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process
6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6531
clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process
6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6532
clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process
6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6533
clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process
6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6534
clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process
6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6535
clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process
6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6536
clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process
6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6537
clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process
6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6538
clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process
6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6539
clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process
6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6540
clone(child_stack=0xffffabc0e960,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY

SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6541
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6542
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6543
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6544
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6545
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6546
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6547
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6548
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6549
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6550
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6551
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY

SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6552
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6553
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6554
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6555
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6556
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6557
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6558
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6559
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6560
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6561
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY
SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6562
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY

SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6563
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6564
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6565
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6566
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6567
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6568
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6569
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6570
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6571
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6572
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6573
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SY

SVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6574
clone(child_stack=0xffffabc0e960, flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 6531 attached
, parent_tid=[6531], tls=0xffffabc0f8e0, child_tidptr=0xffffabc0f1f0) = 6575

Лабораторная работа №3

vscode ➜ /workspaces/MAI_OS/lab3 (main) $ strace ./client

execve("./client", ["./client"], 0xffffde9e7ed0 /* 35 vars */) = 0

brk(NULL) = 0xaaaade327000

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffff99e38000

faccessat(AT_FDCWD, "/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=15048, ...}, AT_EMPTY_PATH) = 0

mmap(NULL, 15048, PROT_READ, MAP_PRIVATE, 3, 0) = 0xffff99e34000

close(3) = 0

openat(AT_FDCWD, "/lib/aarch64-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\267\0\1\0\0\0\340u\2\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=1637400, ...}, AT_EMPTY_PATH) = 0

mmap(NULL, 1805928, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffff99c4a000

mmap(0xffff99c50000, 1740392, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffff99c50000

munmap(0xffff99c4a000, 24576) = 0

munmap(0xffff99df9000, 40552) = 0

mprotect(0xffff99dd8000, 61440, PROT_NONE) = 0

mmap(0xffff99de7000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x187000) = 0xffff99de7000

mmap(0xffff99ded000, 48744, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0xffff99ded000

close(3) = 0

set_tid_address(0xffff99e38f50) = 35722

set_robust_list(0xffff99e38f60, 24) = 0

rseq(0xffff99e39620, 0x20, 0, 0xd428bc00) = 0

mprotect(0xffff99de7000, 16384, PROT_READ) = 0

mprotect(0xaaaabeb41000, 4096, PROT_READ) = 0

mprotect(0xffff99e3d000, 8192, PROT_READ) = 0

prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0

munmap(0xffff99e34000, 15048) = 0

**openat(AT_FDCWD, "/dev/shm/sum_sh_memory",
O_RDWR|O_CREAT|O_TRUNC|O_NOFOLLOW|O_CLOEXEC, 0666) = 3**

ftruncate(3, 4096) = 0

mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_SHARED, 3, 0) = 0xffff99e37000

**openat(AT_FDCWD, "/dev/shm/sem.sum_semaphore", O_RDWR|O_NOFOLLOW) = -1
ENOENT (No such file or directory)**

getrandom("\x09\x15\x1a\x2a\x1a\x37\x3a\x06", 8, GRND_NONBLOCK) = 8

newfstatat(AT_FDCWD, "/dev/shm/sem.rq2AZl", 0xffffd2cc89b8, AT_SYMLINK_NOFOLLOW) = -1
ENOENT (No such file or directory)

openat(AT_FDCWD, "/dev/shm/sem.rq2AZl", O_RDWR|O_CREAT|O_EXCL, 0666) = 4

write(4, "\1\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\0", 32) = 32

mmap(NULL, 32, PROT_READ|PROT_WRITE, MAP_SHARED, 4, 0) = 0xffff99e36000

linkat(AT_FDCWD, "/dev/shm/sem.rq2AZl", AT_FDCWD, "/dev/shm/sem.sum_semaphore", 0) = 0

newfstatat(4, "", {st_mode=S_IFREG|0644, st_size=32, ...}, AT_EMPTY_PATH) = 0

getrandom("\x2d\x33\x81\xed\x5c\xa9\x69\xda", 8, GRND_NONBLOCK) = 8

brk(NULL) = 0xaaaade327000

brk(0xaaaade348000) = 0xaaaade348000

unlinkat(AT_FDCWD, "/dev/shm/sem.rq2AZl", 0) = 0

close(4) = 0

write(1, "Input file : ", 13Input file :) = 13

read(0, a.txt

"a.txt\n", 127) = 6

**clone(child_stack=NULL,
flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLD,
child_tidptr=0xffff99e38f50) = 35816**

clock_nanosleep(CLOCK_REALTIME, 0, {tv_sec=0, tv_nsec=1000000}, NULL) = 0

write(1, "6\n", 26

) = 2

clock_nanosleep(CLOCK_REALTIME, 0, {tv_sec=0, tv_nsec=1000000}, NULL) = 0

write(1, "45\n", 345

) = 3

clock_nanosleep(CLOCK_REALTIME, 0, {tv_sec=0, tv_nsec=1000000}, NULL) = 0

write(1, "0\n", 20

) = 2

clock_nanosleep(CLOCK_REALTIME, 0, {tv_sec=0, tv_nsec=1000000}, NULL) = 0

clock_nanosleep(CLOCK_REALTIME, 0, {tv_sec=0, tv_nsec=1000000}, NULL) = ?
ERESTART_RESTARTBLOCK (Interrupted by signal)

--- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=35816, si_uid=1000, si_status=0, si_utime=0, si_stime=0} ---

restart_syscall(<... resuming interrupted clock_nanosleep ...>) = 0

clock_nanosleep(CLOCK_REALTIME, 0, {tv_sec=0, tv_nsec=1000000}, NULL) = 0

wait4(35816, NULL, 0, NULL) = 35816

munmap(0xffff99e36000, 32) = 0

unlinkat(AT_FDCWD, "/dev/shm/sem.sum_semaphore", 0) = 0

munmap(0xffff99e37000, 4096) = 0

unlinkat(AT_FDCWD, "/dev/shm/sum_sh_memory", 0) = 0

close(3) = 0

exit_group(0) = ?

+++ exited with 0 +++

Лабораторная работа №4

execve("./runtime", ["./runtime"], 0xffffe68c1a18 /* 35 vars */) = 0

brk(NULL) = 0xaaab1e700000

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffff833c8000

faccessat(AT_FDCWD, "/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=15048, ...}, AT_EMPTY_PATH) = 0

```
mmap(NULL, 15048, PROT_READ, MAP_PRIVATE, 3, 0) = 0xffff833c4000
close(3)                                = 0
openat(AT_FDCWD, "/lib/aarch64-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\267\0\1\0\0\0\340u\2\0\0\0\0\0"..., 832) = 832
newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=1637400, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 1805928, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0xffff831da000
mmap(0xffff831e0000, 1740392, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffff831e0000
munmap(0xffff831da000, 24576)           = 0
munmap(0xffff83389000, 40552)           = 0
mprotect(0xffff83368000, 61440, PROT_NONE) = 0
mmap(0xffff83377000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x187000) = 0xffff83377000
mmap(0xffff8337d000, 48744, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0xffff8337d000
close(3)                                = 0
set_tid_address(0xffff833c8f50)         = 17121
set_robust_list(0xffff833c8f60, 24)     = 0
rseq(0xffff833c9620, 0x20, 0, 0xd428bc00) = 0
mprotect(0xffff83377000, 16384, PROT_READ) = 0
mprotect(0xaaaae7eb1000, 4096, PROT_READ) = 0
mprotect(0xffff833cd000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
munmap(0xffff833c4000, 15048)           = 0
newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}, AT_EMPTY_PATH) = 0
getrandom("\xbc\x01\x6f\x31\x7d\xed\xd1\xa5", 8, GRND_NONBLOCK) = 8
brk(NULL)                               = 0xaaab1e700000
brk(0xaaab1e721000)                     = 0xaaab1e721000
write(1, "Please enter: M arg1 arg2 ... ar"..., 35Please enter: M arg1 arg2 ... argN
) = 35
write(1, "M - Selected function from libra"..., 35M - Selected function from library
) = 35
write(1, "M = -1 - Complete the execution\n", 32M = -1 - Complete the execution
) = 32
```

write(1, "M = 0 - Change the library \n", 28M = 0 - Change the library

) = 28

write(1, "M = 1 - Count the amount of prim"..., 71M = 1 - Count the amount of prime numbers in [A, B]; arg1, arg2 - A, B

) = 71

write(1, "M = 2 - Array sorting; arg1 - si"..., 97M = 2 - Array sorting; arg1 - size of unsorted array, arg2 ... argN - elements of unsorted array

) = 97

openat(AT_FDCWD, "./lib2/lib2.so", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=12640, ...}, AT_EMPTY_PATH) = 0

getcwd("/workspaces/MAI_OS/lab4/src", 128) = 28

mmap(NULL, 139368, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffff831bd000

mmap(0xffff831c0000, 73832, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffff831c0000

munmap(0xffff831bd000, 12288)   = 0

munmap(0xffff831d3000, 49256)   = 0

mprotect(0xffff831c2000, 61440, PROT_NONE) = 0

mmap(0xffff831d1000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0xffff831d1000

close(3)   = 0

mprotect(0xffff831d1000, 4096, PROT_READ) = 0

write(1, "\n", 1

)   = 1

write(1, "2nd dynamic library loaded\n", 272nd dynamic library loaded

) = 27

write(1, "Now Eratosthenes Sieve is used f"..., 58Now Eratosthenes Sieve is used for counting prime numbers

) = 58

write(1, "Now QuickSort is used to sort th"..., 40Now QuickSort is used to sort the array

) = 40

write(1, "\n", 1

)   = 1

newfstatat(0, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}, AT_EMPTY_PATH) = 0

```
read(0, 1 1 10
"1 1 10\n", 1024)               = 7
write(1, "\n", 1
)                       = 1
write(1, "The amount of prime numbers - 4\n"..., 33The amount of prime numbers - 4


) = 33
read(0, 2 5 5 4 3 2 1
"2 5 5 4 3 2 1\n", 1024)        = 14
write(1, "Sorted array - 1 2 3 4 5 \n", 26Sorted array - 1 2 3 4 5
) = 26
write(1, "\n", 1
)                       = 1
read(0, 0
"0\n", 1024)                    = 2
write(1, "\n", 1
)                       = 1
munmap(0xffff831c0000, 73832)           = 0
openat(AT_FDCWD, "./lib1/lib1.so", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832
newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=8096, ...}, AT_EMPTY_PATH) = 0
getcwd("/workspaces/MAI_OS/lab4/src", 128) = 28
mmap(NULL, 135208, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0xffff831be000
mmap(0xffff831c0000, 69672, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffff831c0000
munmap(0xffff831be000, 8192)            = 0
munmap(0xffff831d2000, 53288)           = 0
mprotect(0xffff831c1000, 61440, PROT_NONE) = 0
mmap(0xffff831d0000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffff831d0000
close(3)                                = 0
mprotect(0xffff831d0000, 4096, PROT_READ) = 0
write(1, "1st dynamic library loaded\n", 271st dynamic library loaded
```

```
) = 27
write(1, "Now The naive algorithm is used "..., 59Now The naive algorithm is used for counting prime
numbers
) = 59
write(1, "Now BubbleSort is used to sort t"..., 41Now BubbleSort is used to sort the array
) = 41
write(1, "Your library has been changed su"..., 45Your library has been changed successfuly!!!
) = 45
write(1, "\n", 1
)                       = 1
read(0, 1 1 10
"1 1 10\n", 1024)               = 7
write(1, "\n", 1
)                       = 1
write(1, "The amount of prime numbers - 4\n"..., 33The amount of prime numbers - 4

) = 33
read(0, 2 5 5 4 3 2 1
"2 5 5 4 3 2 1\n", 1024)        = 14
write(1, "Sorted array - 1 2 3 4 5 \n", 26Sorted array - 1 2 3 4 5
) = 26
write(1, "\n", 1
)                       = 1
read(0, -1
"-1\n", 1024)                   = 3
munmap(0xffff831c0000, 69672)           = 0
write(1, "The program successfuly finished"..., 33The program successfuly finished
) = 33
lseek(0, -1, SEEK_CUR)                  = -1 ESPIPE (Illegal seek)
exit_group(0)                           = ?
+++ exited with 0 +++
```

Вывод

В ходе выполнения лабораторных работ с использованием утилиты **strace** были изучены основные механизмы взаимодействия пользовательских программ с ядром операционной системы Linux. На практике были проанализированы системные вызовы, связанные с созданием и завершением процессов, межпроцессным взаимодействием, перенаправлением потоков ввода-вывода, управлением памятью и обработкой сигналов.