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: Путилин Д.Н.

Группа: М8О-207Б-21

Вариант: -

Преподаватель: Миронов Евгений Сергеевич

Оценка: \_\_\_\_\_\_\_\_\_\_\_

Дата: \_\_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_\_

Москва, 2022

**Содержание**

1. Репозиторий
2. Постановка задачи
3. Общие сведения о программе
4. Общий метод и алгоритм решения
5. Исходный код
6. Демонстрация работы программы
7. Выводы

**Репозиторий**

[https://github.com/](https://github.com/Liguha/OS)putilin21dn/OC

**Постановка задачи**

**Цель работы**

Приобретение практических навыков диагностики работы программного обеспечения.

**Задание**

При выполнении последующих лабораторных работ необходимо продемонстрировать ключевые системные вызовы, которые в них используются и то, что их использование соответствует варианту ЛР 3.

По итогам выполнения всех лабораторных работ отчет по данной должен содержать краткую сводку по исследованию последующих ЛР 3.

**Общие сведения о программе**

ssize\_t **read**(int fd, void \*buf, size\_t count);

read - чтение из файлового дескриптора

read() пытается считывать до подсчета байтов из файлового дескриптора fd в

буфер, начинающийся с buf.

ВОЗВРАЩАЕМОЕ ЗНАЧЕНИЕ

При успешном выполнении возвращается количество прочитанных байт (ноль указывает на конец

файла), и позиция файла увеличивается на это число. Это не ошибка, если это число меньше запрошенного количества байтов;

это может произойти, например, из-за того, что на самом деле доступно меньше байтов прямо сейчас (возможно, потому, что мы были близки к завершению файла, или потому, что мы считываются из канала или с терминала), или потому, что read() был впрерванный сигналом.

При ошибке возвращается значение -1, а значение errno устанавливается для указания ошибки. В

этом случае остается неуказанным, является ли позиция файла (если таковая имеется)

изменения.

int **openat**(int dirfd, const char \*pathname, int flags, mode\_t mode);

openat - откройте и, возможно, создайте файл

Системный вызов openat() работает точно так же, как open(), за исключением описанных здесь различий.

Аргумент dirfd используется в сочетании с аргументом pathname следующим

образом:

\* Если путь, указанный в pathname, является абсолютным, то dirfd - это ig‐nored.

\* Если путь, указанный в pathname, является относительным, а dirfd - это special значение AT\_FDCWD, тогда pathname интерпретируется относительно текущего рабочего каталога вызывающего процесса (например, open()).

\* Если путь, указанный в pathname, является относительным, то это inter‐ устанавливается относительно каталога, на который ссылается дескриптор файла dirfd (а не относительно текущего рабочего каталога вызывающего процесса, как это делается с помощью open() для относительного пути). В

этом случае dirfd должен быть каталогом, который был открыт для чтения (O\_RDONLY) или с использованием флага O\_PATH.

ВОЗВРАЩАЕМОЕ ЗНАЧЕНИЕ

При успешном выполнении openat() возвращают описание нового файла tor (неотрицательное целое число). При ошибке возвращается значение -1 и устанавливается значение errno чтобы указать на ошибку.

int **clone3**(int (\*fn)(void \*), void \*stack, int flags, void \*arg, pid\_t \*parent\_tid, void \*tls, pid\_t \*child\_tid);

clone3 - создайте дочерний процесс

Системный вызов clone3() предоставляет дополнительный набор функциональных возможностей более старого интерфейса clone(). Он также предоставляет ряд улучшений API‐улучшения, в том числе: пространство для дополнительных битов флагов; более чистое разделение в использовании различных аргументов; и возможность указать размер дочерней области стека.

ВОЗВРАЩАЕМОЕ ЗНАЧЕНИЕ

При успешном выполнении идентификатор потока дочернего процесса возвращается в потоке выполнения вызывающей стороны. При сбое возвращается значение -1 в контекст вызывающего абонента, дочерний процесс не создается, и значение errno устанавливается в error.

**Исходный код**

|  |
| --- |
| **report** |
| dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/build$ strace ./main.out  execve("./main.out", ["./main.out"], 0x7fff5127a510 /\* 63 vars \*/) = 0  brk(NULL) = 0x559428492000  arch\_prctl(0x3001 /\* ARCH\_??? \*/, 0x7fff7a988db0) = -1 EINVAL (Недопустимый аргумент)  mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7fa02e235000  access("/etc/ld.so.preload", R\_OK) = -1 ENOENT (Нет такого файла или каталога)  openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC) = 3  newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=70013, ...}, AT\_EMPTY\_PATH) = 0  mmap(NULL, 70013, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7fa02e223000  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\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832  newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=2252064, ...}, AT\_EMPTY\_PATH) = 0  mmap(NULL, 2267264, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7fa02de00000  mmap(0x7fa02de9a000, 1114112, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x9a000) = 0x7fa02de9a000  mmap(0x7fa02dfaa000, 454656, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1aa000) = 0x7fa02dfaa000  mmap(0x7fa02e019000, 57344, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x218000) = 0x7fa02e019000  mmap(0x7fa02e027000, 10368, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7fa02e027000  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\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832  newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=125464, ...}, AT\_EMPTY\_PATH) = 0  mmap(NULL, 127688, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7fa02e203000  mmap(0x7fa02e206000, 94208, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x3000) = 0x7fa02e206000  mmap(0x7fa02e21d000, 16384, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1a000) = 0x7fa02e21d000  mmap(0x7fa02e221000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1d000) = 0x7fa02e221000  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\3206\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  newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=2072888, ...}, AT\_EMPTY\_PATH) = 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, 2117488, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7fa02da00000  mmap(0x7fa02da22000, 1544192, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x22000) = 0x7fa02da22000  mmap(0x7fa02db9b000, 356352, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x19b000) = 0x7fa02db9b000  mmap(0x7fa02dbf2000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1f1000) = 0x7fa02dbf2000  mmap(0x7fa02dbf8000, 53104, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7fa02dbf8000  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\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832  newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=948816, ...}, AT\_EMPTY\_PATH) = 0  mmap(NULL, 950520, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7fa02e11a000  mmap(0x7fa02e128000, 516096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0xe000) = 0x7fa02e128000  mmap(0x7fa02e1a6000, 372736, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x8c000) = 0x7fa02e1a6000  mmap(0x7fa02e201000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0xe6000) = 0x7fa02e201000  close(3) = 0  mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7fa02e118000  arch\_prctl(ARCH\_SET\_FS, 0x7fa02e119400) = 0  set\_tid\_address(0x7fa02e1196d0) = 9579  set\_robust\_list(0x7fa02e1196e0, 24) = 0  rseq(0x7fa02e119d20, 0x20, 0, 0x53053053) = 0  mprotect(0x7fa02dbf2000, 16384, PROT\_READ) = 0  mprotect(0x7fa02e201000, 4096, PROT\_READ) = 0  mprotect(0x7fa02e221000, 4096, PROT\_READ) = 0  mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7fa02e116000  mprotect(0x7fa02e019000, 45056, PROT\_READ) = 0  mprotect(0x559428175000, 4096, PROT\_READ) = 0  mprotect(0x7fa02e26b000, 8192, PROT\_READ) = 0  prlimit64(0, RLIMIT\_STACK, NULL, {rlim\_cur=8192\*1024, rlim\_max=RLIM64\_INFINITY}) = 0  munmap(0x7fa02e223000, 70013) = 0  getrandom("\xc3\xa5\x6f\x92\x79\xc0\x25\x0c", 8, GRND\_NONBLOCK) = 8  brk(NULL) = 0x559428492000  brk(0x5594284b3000) = 0x5594284b3000  futex(0x7fa02e02773c, FUTEX\_WAKE\_PRIVATE, 2147483647) = 0  newfstatat(0, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...}, AT\_EMPTY\_PATH) = 0  read(0, ^Cstrace: Process 9579 detached  <detached ...>  dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/build$ strace ./main.out 4 10  execve("./main.out", ["./main.out", "4", "10"], 0x7ffe1638c310 /\* 63 vars \*/) = 0  brk(NULL) = 0x561c8492b000  arch\_prctl(0x3001 /\* ARCH\_??? \*/, 0x7ffc8fe8d4c0) = -1 EINVAL (Недопустимый аргумент)  mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7fe5f701a000  access("/etc/ld.so.preload", R\_OK) = -1 ENOENT (Нет такого файла или каталога)  openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC) = 3  newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=70013, ...}, AT\_EMPTY\_PATH) = 0  mmap(NULL, 70013, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7fe5f7008000  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\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832  newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=2252064, ...}, AT\_EMPTY\_PATH) = 0  mmap(NULL, 2267264, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7fe5f6c00000  mmap(0x7fe5f6c9a000, 1114112, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x9a000) = 0x7fe5f6c9a000  mmap(0x7fe5f6daa000, 454656, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1aa000) = 0x7fe5f6daa000  mmap(0x7fe5f6e19000, 57344, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x218000) = 0x7fe5f6e19000  mmap(0x7fe5f6e27000, 10368, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7fe5f6e27000  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\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832  newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=125464, ...}, AT\_EMPTY\_PATH) = 0  mmap(NULL, 127688, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7fe5f6fe8000  mmap(0x7fe5f6feb000, 94208, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x3000) = 0x7fe5f6feb000  mmap(0x7fe5f7002000, 16384, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1a000) = 0x7fe5f7002000  mmap(0x7fe5f7006000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1d000) = 0x7fe5f7006000  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\3206\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  newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=2072888, ...}, AT\_EMPTY\_PATH) = 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, 2117488, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7fe5f6800000  mmap(0x7fe5f6822000, 1544192, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x22000) = 0x7fe5f6822000  mmap(0x7fe5f699b000, 356352, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x19b000) = 0x7fe5f699b000  mmap(0x7fe5f69f2000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1f1000) = 0x7fe5f69f2000  mmap(0x7fe5f69f8000, 53104, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7fe5f69f8000  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\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832  newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=948816, ...}, AT\_EMPTY\_PATH) = 0  mmap(NULL, 950520, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7fe5f6eff000  mmap(0x7fe5f6f0d000, 516096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0xe000) = 0x7fe5f6f0d000  mmap(0x7fe5f6f8b000, 372736, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x8c000) = 0x7fe5f6f8b000  mmap(0x7fe5f6fe6000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0xe6000) = 0x7fe5f6fe6000  close(3) = 0  mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7fe5f6efd000  arch\_prctl(ARCH\_SET\_FS, 0x7fe5f6efe400) = 0  set\_tid\_address(0x7fe5f6efe6d0) = 9721  set\_robust\_list(0x7fe5f6efe6e0, 24) = 0  rseq(0x7fe5f6efed20, 0x20, 0, 0x53053053) = 0  mprotect(0x7fe5f69f2000, 16384, PROT\_READ) = 0  mprotect(0x7fe5f6fe6000, 4096, PROT\_READ) = 0  mprotect(0x7fe5f7006000, 4096, PROT\_READ) = 0  mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7fe5f6efb000  mprotect(0x7fe5f6e19000, 45056, PROT\_READ) = 0  mprotect(0x561c8361e000, 4096, PROT\_READ) = 0  mprotect(0x7fe5f7050000, 8192, PROT\_READ) = 0  prlimit64(0, RLIMIT\_STACK, NULL, {rlim\_cur=8192\*1024, rlim\_max=RLIM64\_INFINITY}) = 0  munmap(0x7fe5f7008000, 70013) = 0  getrandom("\x0c\xb2\x2b\xb0\x39\x7c\xa5\x6f", 8, GRND\_NONBLOCK) = 8  brk(NULL) = 0x561c8492b000  brk(0x561c8494c000) = 0x561c8494c000  futex(0x7fe5f6e2773c, FUTEX\_WAKE\_PRIVATE, 2147483647) = 0  newfstatat(0, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...}, AT\_EMPTY\_PATH) = 0  read(0, 5  "5\n", 1024) = 2  read(0, 3  "3\n", 1024) = 2  read(0, 10  "10\n", 1024) = 3  read(0, 5  "5\n", 1024) = 2  read(0, 5  "5\n", 1024) = 2  read(0, 2  "2\n", 1024) = 2  read(0, 3  "3\n", 1024) = 2  read(0, 7  "7\n", 1024) = 2  read(0, 9  "9\n", 1024) = 2  read(0, 2  "2\n", 1024) = 2  read(0, 4  "4\n", 1024) = 2  read(0, 9  "9\n", 1024) = 2  read(0, 6  "6\n", 1024) = 2  read(0, 10  "10\n", 1024) = 3  read(0, 7  "7\n", 1024) = 2  read(0, 3  "3\n", 1024) = 2  read(0, 4  "4\n", 1024) = 2  read(0, 7  "7\n", 1024) = 2  read(0, 6  "6\n", 1024) = 2  read(0, 9  "9\n", 1024) = 2  read(0, 6  "6\n", 1024) = 2  read(0, 5  "5\n", 1024) = 2  read(0, 5  "5\n", 1024) = 2  read(0, 2  "2\n", 1024) = 2  read(0, 1  "1\n", 1024) = 2  read(0, 3  "3\n", 1024) = 2  read(0, 0  "0\n", 1024) = 2  rt\_sigaction(SIGRT\_1, {sa\_handler=0x7fe5f688d1a0, sa\_mask=[], sa\_flags=SA\_RESTORER|SA\_ONSTACK|SA\_RESTART|SA\_SIGINFO, sa\_restorer=0x7fe5f683bcf0}, NULL, 8) = 0  rt\_sigprocmask(SIG\_UNBLOCK, [RTMIN RT\_1], NULL, 8) = 0  mmap(NULL, 8392704, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS|MAP\_STACK, -1, 0) = 0x7fe5f5fff000  mprotect(0x7fe5f6000000, 8388608, PROT\_READ|PROT\_WRITE) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f67ff990, parent\_tid=0x7fe5f67ff990, exit\_signal=0, stack=0x7fe5f5fff000, stack\_size=0x7fff80, tls=0x7fe5f67ff6c0} => {parent\_tid=[9850]}, 88) = 9850  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  mmap(NULL, 8392704, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS|MAP\_STACK, -1, 0) = 0x7fe5f57fe000  mprotect(0x7fe5f57ff000, 8388608, PROT\_READ|PROT\_WRITE) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f5ffe990, parent\_tid=0x7fe5f5ffe990, exit\_signal=0, stack=0x7fe5f57fe000, stack\_size=0x7fff80, tls=0x7fe5f5ffe6c0} => {parent\_tid=[9851]}, 88) = 9851  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  mmap(NULL, 8392704, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS|MAP\_STACK, -1, 0) = 0x7fe5f4ffd000  mprotect(0x7fe5f4ffe000, 8388608, PROT\_READ|PROT\_WRITE) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f57fd990, parent\_tid=0x7fe5f57fd990, exit\_signal=0, stack=0x7fe5f4ffd000, stack\_size=0x7fff80, tls=0x7fe5f57fd6c0} => {parent\_tid=[9852]}, 88) = 9852  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  mmap(NULL, 8392704, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS|MAP\_STACK, -1, 0) = 0x7fe5f47fc000  mprotect(0x7fe5f47fd000, 8388608, PROT\_READ|PROT\_WRITE) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f4ffc990, parent\_tid=0x7fe5f4ffc990, exit\_signal=0, stack=0x7fe5f47fc000, stack\_size=0x7fff80, tls=0x7fe5f4ffc6c0} => {parent\_tid=[9853]}, 88) = 9853  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  futex(0x7fe5f67ff990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9850, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f4ffc990, parent\_tid=0x7fe5f4ffc990, exit\_signal=0, stack=0x7fe5f47fc000, stack\_size=0x7fff80, tls=0x7fe5f4ffc6c0} => {parent\_tid=[9854]}, 88) = 9854  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f57fd990, parent\_tid=0x7fe5f57fd990, exit\_signal=0, stack=0x7fe5f4ffd000, stack\_size=0x7fff80, tls=0x7fe5f57fd6c0} => {parent\_tid=[9855]}, 88) = 9855  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f5ffe990, parent\_tid=0x7fe5f5ffe990, exit\_signal=0, stack=0x7fe5f57fe000, stack\_size=0x7fff80, tls=0x7fe5f5ffe6c0} => {parent\_tid=[9856]}, 88) = 9856  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f67ff990, parent\_tid=0x7fe5f67ff990, exit\_signal=0, stack=0x7fe5f5fff000, stack\_size=0x7fff80, tls=0x7fe5f67ff6c0} => {parent\_tid=[9857]}, 88) = 9857  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  futex(0x7fe5f67ff990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9857, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f67ff990, parent\_tid=0x7fe5f67ff990, exit\_signal=0, stack=0x7fe5f5fff000, stack\_size=0x7fff80, tls=0x7fe5f67ff6c0} => {parent\_tid=[9858]}, 88) = 9858  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f5ffe990, parent\_tid=0x7fe5f5ffe990, exit\_signal=0, stack=0x7fe5f57fe000, stack\_size=0x7fff80, tls=0x7fe5f5ffe6c0} => {parent\_tid=[9859]}, 88) = 9859  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f57fd990, parent\_tid=0x7fe5f57fd990, exit\_signal=0, stack=0x7fe5f4ffd000, stack\_size=0x7fff80, tls=0x7fe5f57fd6c0} => {parent\_tid=[9860]}, 88) = 9860  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f4ffc990, parent\_tid=0x7fe5f4ffc990, exit\_signal=0, stack=0x7fe5f47fc000, stack\_size=0x7fff80, tls=0x7fe5f4ffc6c0} => {parent\_tid=[9861]}, 88) = 9861  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f4ffc990, parent\_tid=0x7fe5f4ffc990, exit\_signal=0, stack=0x7fe5f47fc000, stack\_size=0x7fff80, tls=0x7fe5f4ffc6c0} => {parent\_tid=[9862]}, 88) = 9862  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f57fd990, parent\_tid=0x7fe5f57fd990, exit\_signal=0, stack=0x7fe5f4ffd000, stack\_size=0x7fff80, tls=0x7fe5f57fd6c0} => {parent\_tid=[9863]}, 88) = 9863  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f5ffe990, parent\_tid=0x7fe5f5ffe990, exit\_signal=0, stack=0x7fe5f57fe000, stack\_size=0x7fff80, tls=0x7fe5f5ffe6c0} => {parent\_tid=[9864]}, 88) = 9864  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f67ff990, parent\_tid=0x7fe5f67ff990, exit\_signal=0, stack=0x7fe5f5fff000, stack\_size=0x7fff80, tls=0x7fe5f67ff6c0} => {parent\_tid=[9865]}, 88) = 9865  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  futex(0x7fe5f4ffc990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9862, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f67ff990, parent\_tid=0x7fe5f67ff990, exit\_signal=0, stack=0x7fe5f5fff000, stack\_size=0x7fff80, tls=0x7fe5f67ff6c0} => {parent\_tid=[9866]}, 88) = 9866  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f5ffe990, parent\_tid=0x7fe5f5ffe990, exit\_signal=0, stack=0x7fe5f57fe000, stack\_size=0x7fff80, tls=0x7fe5f5ffe6c0} => {parent\_tid=[9867]}, 88) = 9867  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f57fd990, parent\_tid=0x7fe5f57fd990, exit\_signal=0, stack=0x7fe5f4ffd000, stack\_size=0x7fff80, tls=0x7fe5f57fd6c0} => {parent\_tid=[9868]}, 88) = 9868  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f4ffc990, parent\_tid=0x7fe5f4ffc990, exit\_signal=0, stack=0x7fe5f47fc000, stack\_size=0x7fff80, tls=0x7fe5f4ffc6c0} => {parent\_tid=[9869]}, 88) = 9869  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  futex(0x7fe5f57fd990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9868, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f4ffc990, parent\_tid=0x7fe5f4ffc990, exit\_signal=0, stack=0x7fe5f47fc000, stack\_size=0x7fff80, tls=0x7fe5f4ffc6c0} => {parent\_tid=[9870]}, 88) = 9870  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f57fd990, parent\_tid=0x7fe5f57fd990, exit\_signal=0, stack=0x7fe5f4ffd000, stack\_size=0x7fff80, tls=0x7fe5f57fd6c0} => {parent\_tid=[9871]}, 88) = 9871  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f5ffe990, parent\_tid=0x7fe5f5ffe990, exit\_signal=0, stack=0x7fe5f57fe000, stack\_size=0x7fff80, tls=0x7fe5f5ffe6c0} => {parent\_tid=[9872]}, 88) = 9872  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f67ff990, parent\_tid=0x7fe5f67ff990, exit\_signal=0, stack=0x7fe5f5fff000, stack\_size=0x7fff80, tls=0x7fe5f67ff6c0} => {parent\_tid=[9873]}, 88) = 9873  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  futex(0x7fe5f4ffc990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9870, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f67ff990, parent\_tid=0x7fe5f67ff990, exit\_signal=0, stack=0x7fe5f5fff000, stack\_size=0x7fff80, tls=0x7fe5f67ff6c0} => {parent\_tid=[9874]}, 88) = 9874  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f5ffe990, parent\_tid=0x7fe5f5ffe990, exit\_signal=0, stack=0x7fe5f57fe000, stack\_size=0x7fff80, tls=0x7fe5f5ffe6c0} => {parent\_tid=[9875]}, 88) = 9875  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f57fd990, parent\_tid=0x7fe5f57fd990, exit\_signal=0, stack=0x7fe5f4ffd000, stack\_size=0x7fff80, tls=0x7fe5f57fd6c0} => {parent\_tid=[9876]}, 88) = 9876  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f4ffc990, parent\_tid=0x7fe5f4ffc990, exit\_signal=0, stack=0x7fe5f47fc000, stack\_size=0x7fff80, tls=0x7fe5f4ffc6c0} => {parent\_tid=[9877]}, 88) = 9877  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f4ffc990, parent\_tid=0x7fe5f4ffc990, exit\_signal=0, stack=0x7fe5f47fc000, stack\_size=0x7fff80, tls=0x7fe5f4ffc6c0} => {parent\_tid=[9878]}, 88) = 9878  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f57fd990, parent\_tid=0x7fe5f57fd990, exit\_signal=0, stack=0x7fe5f4ffd000, stack\_size=0x7fff80, tls=0x7fe5f57fd6c0} => {parent\_tid=[9879]}, 88) = 9879  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f5ffe990, parent\_tid=0x7fe5f5ffe990, exit\_signal=0, stack=0x7fe5f57fe000, stack\_size=0x7fff80, tls=0x7fe5f5ffe6c0} => {parent\_tid=[9880]}, 88) = 9880  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f67ff990, parent\_tid=0x7fe5f67ff990, exit\_signal=0, stack=0x7fe5f5fff000, stack\_size=0x7fff80, tls=0x7fe5f67ff6c0} => {parent\_tid=[9881]}, 88) = 9881  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  futex(0x7fe5f57fd990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9879, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  futex(0x7fe5f67ff990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9881, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f67ff990, parent\_tid=0x7fe5f67ff990, exit\_signal=0, stack=0x7fe5f5fff000, stack\_size=0x7fff80, tls=0x7fe5f67ff6c0} => {parent\_tid=[9882]}, 88) = 9882  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f5ffe990, parent\_tid=0x7fe5f5ffe990, exit\_signal=0, stack=0x7fe5f57fe000, stack\_size=0x7fff80, tls=0x7fe5f5ffe6c0} => {parent\_tid=[9883]}, 88) = 9883  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f57fd990, parent\_tid=0x7fe5f57fd990, exit\_signal=0, stack=0x7fe5f4ffd000, stack\_size=0x7fff80, tls=0x7fe5f57fd6c0} => {parent\_tid=[9884]}, 88) = 9884  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f4ffc990, parent\_tid=0x7fe5f4ffc990, exit\_signal=0, stack=0x7fe5f47fc000, stack\_size=0x7fff80, tls=0x7fe5f4ffc6c0} => {parent\_tid=[9885]}, 88) = 9885  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  futex(0x7fe5f67ff990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9882, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  futex(0x7fe5f57fd990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9884, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  futex(0x7fe5f4ffc990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9885, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f4ffc990, parent\_tid=0x7fe5f4ffc990, exit\_signal=0, stack=0x7fe5f47fc000, stack\_size=0x7fff80, tls=0x7fe5f4ffc6c0} => {parent\_tid=[9886]}, 88) = 9886  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f57fd990, parent\_tid=0x7fe5f57fd990, exit\_signal=0, stack=0x7fe5f4ffd000, stack\_size=0x7fff80, tls=0x7fe5f57fd6c0} => {parent\_tid=[9887]}, 88) = 9887  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f5ffe990, parent\_tid=0x7fe5f5ffe990, exit\_signal=0, stack=0x7fe5f57fe000, stack\_size=0x7fff80, tls=0x7fe5f5ffe6c0} => {parent\_tid=[9888]}, 88) = 9888  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0  clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7fe5f67ff990, parent\_tid=0x7fe5f67ff990, exit\_signal=0, stack=0x7fe5f5fff000, stack\_size=0x7fff80, tls=0x7fe5f67ff6c0} => {parent\_tid=[9889]}, 88) = 9889  rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0  futex(0x7fe5f5ffe990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 9888, NULL, FUTEX\_BITSET\_MATCH\_ANY) = 0  newfstatat(1, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...}, AT\_EMPTY\_PATH) = 0  write(1, "Filtr narashivania \n", 20Filtr narashivania  ) = 20  write(1, "3 10 5 5 2 \n", 123 10 5 5 2  ) = 12  write(1, "3 7 9 2 4 \n", 113 7 9 2 4  ) = 11  write(1, "9 6 10 7 3 \n", 129 6 10 7 3  ) = 12  write(1, "4 7 6 9 6 \n", 114 7 6 9 6  ) = 11  write(1, "5 5 2 1 3 \n", 115 5 2 1 3  ) = 11  write(1, "\nFiltr erosii \n", 16  Filtr erosii  ) = 16  write(1, "3 10 5 5 2 \n", 123 10 5 5 2  ) = 12  write(1, "3 7 9 2 4 \n", 113 7 9 2 4  ) = 11  write(1, "9 6 10 7 3 \n", 129 6 10 7 3  ) = 12  write(1, "4 7 6 9 6 \n", 114 7 6 9 6  ) = 11  write(1, "5 5 2 1 3 \n", 115 5 2 1 3  ) = 11  lseek(0, -1, SEEK\_CUR) = -1 ESPIPE (Недопустимая операция смещения)  exit\_group(0) = ?  +++ exited with 0 +++ |

**Выводы**

Проделав данную лабораторную работу, я приобрел навыки, необходимые для работы с strace, а также изучил системные вызовы.