

strace: ???????? syscall'?

Strace очень полезная тулза, чтобы разобраться, что вообще делает приложение. Можно запустить через strace или подцепиться к процессу.

Запустить:

```
root@test0:~# strace ls -la
execve("/usr/bin/ls", ["ls", "-la"], 0x7ffda9fafe78 /* 20 vars */) = 0
brk(NULL)                                = 0x561515250000
arch_prctl(0x3001 /* ARCH_??? */, 0x7ffff9801360) = -1 EINVAL (Invalid argument)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f66a4721000
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
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=34320, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 34320, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f66a4718000
close(3)                                  = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libselinux.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\0"..., 832) = 832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=166280, ...}, AT_EMPTY_PATH) = 0
...
```

Подцепиться:

```
root@test0:~# ps auxf | grep [c]ron
root          416  0.0  0.2  9496  2212 ?        Ss   Aug05   0:18 /usr/sbin/cron -f -P
root@test0:~# strace -p 416
strace: Process 416 attached
restart_syscall(<... resuming interrupted read ...>) = 0
newfstatat(AT_FDCWD, "/etc/localtime", {st_mode=S_IFREG|0644, st_size=1535, ...}, 0) = 0
newfstatat(AT_FDCWD, "crontabs", {st_mode=S_IFDIR|S_ISVTX|0730, st_size=4096, ...}, 0) = 0
newfstatat(AT_FDCWD, "/etc/crontab", {st_mode=S_IFREG|0644, st_size=1136, ...}, 0) = 0
newfstatat(AT_FDCWD, "/etc/cron.d", {st_mode=S_IFDIR|0755, st_size=4096, ...}, 0) = 0
newfstatat(AT_FDCWD, "/etc/cron.d/e2scrub_all", {st_mode=S_IFREG|0644, st_size=201, ...}, 0) = 0
clone(child_stack=NULL, flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLD,
child_tidptr=0x7f233edb0b10) = 687725
clock_nanosleep(CLOCK_REALTIME, 0, {tv_sec=60, tv_nsec=0}, {tv_sec=59, tv_nsec=982248922}) = ?
```

```

ERESTART_RESTARTBLOCK (Interrupted by signal)
--- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=687725, si_uid=0, si_status=0,
si_utime=0, si_stime=0} ---
wait4(-1, [{WIFEXITED(s) && WEXITSTATUS(s) == 0}], WNOHANG, NULL) = 687725
wait4(-1, 0x7ffda4fdc294, WNOHANG, NULL) = -1 ECHILD (No child processes)
rt_sigreturn({mask=[]})          = -1 EINTR (Interrupted system call)
newfstatat(AT_FDCWD, "/etc/localtime", {st_mode=S_IFREG|0644, st_size=1535, ...}, 0) = 0
clock_nanosleep(CLOCK_REALTIME, 0, {tv_sec=60, tv_nsec=0},

```

Strace показывает **вызовы ядра** (обращение|работа с ФС/сетью/выделение памяти|ресурсов(cgroups)/отправляемые|принимаемые сигналы), но **не показывает** переменные внутри процесса и происходящую логику. Обычно этого достаточно чтобы понять, какие трудности и невзгоды возникли у приложения, но в сложных случаях иногда приходится обращаться к gdb.

Еще из полезных флагов:

```

# Если приложение создает потоки - мониторить и их
strace -f

# Если надо посмотреть какие данные приложение читает/пишет (в сокет), можно увеличить длину
строки:
strace -s 2048

# Статистика вызовов, без самого трейса:
strace -c

# Записать в файл
strace -o strace.log

# Фильтр по определенным вызовам
root@test0:~# strace -e openat,close ls -la > /dev/null
openat(AT_FDCWD, "/etc/ld.so.cache", 0_RDONLY|0_CLOEXEC) = 3
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libselinux.so.1", 0_RDONLY|0_CLOEXEC) = 3
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", 0_RDONLY|0_CLOEXEC) = 3
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libpcr2-8.so.0", 0_RDONLY|0_CLOEXEC) = 3
close(3)                                = 0
openat(AT_FDCWD, "/proc/filesystems", 0_RDONLY|0_CLOEXEC) = 3

```

```

close(3)                                = 0
openat(AT_FDCWD, "/usr/lib/locale/locale-archive", 0_RDONLY|0_CLOEXEC) = 3
close(3)                                = 0
openat(AT_FDCWD, "/usr/share/locale/locale.alias", 0_RDONLY|0_CLOEXEC) = 3
close(3)                                = 0
openat(AT_FDCWD, "/usr/share/locale/en_US.UTF-8/LC_TIME/coreutils.mo", 0_RDONLY) = -1 ENOENT
(No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en_US.utf8/LC_TIME/coreutils.mo", 0_RDONLY) = -1 ENOENT
(No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en_US/LC_TIME/coreutils.mo", 0_RDONLY) = -1 ENOENT (No
such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en.UTF-8/LC_TIME/coreutils.mo", 0_RDONLY) = -1 ENOENT (No
such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en.utf8/LC_TIME/coreutils.mo", 0_RDONLY) = -1 ENOENT (No
such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en/LC_TIME/coreutils.mo", 0_RDONLY) = -1 ENOENT (No such
file or directory)
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/gconv/gconv-modules.cache", 0_RDONLY) = 3
...
+++ exited with 0 +++

```

Выводить только failed сисколы:

```

root@test0:~# strace -e status=failed ls -la > /dev/null
arch_prctl(0x3001 /* ARCH_??? */, 0x7ffdd9f08a00) = -1 EINVAL (Invalid argument)
access("/etc/ld.so.preload", R_OK)          = -1 ENOENT (No such file or directory)
statfs("/sys/fs/selinux", 0x7ffdd9f08a40) = -1 ENOENT (No such file or directory)
statfs("/selinux", 0x7ffdd9f08a40)          = -1 ENOENT (No such file or directory)
access("/etc/selinux/config", F_OK)          = -1 ENOENT (No such file or directory)
ioctl(1, TCGETS, 0x7ffdd9f08970)            = -1 ENOTTY (Inappropriate ioctl for device)
ioctl(1, TIOCGWINSZ, 0x7ffdd9f08a60)        = -1 ENOTTY (Inappropriate ioctl for device)
openat(AT_FDCWD, "/usr/share/locale/en_US.UTF-8/LC_TIME/coreutils.mo", 0_RDONLY) = -1 ENOENT
(No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en_US.utf8/LC_TIME/coreutils.mo", 0_RDONLY) = -1 ENOENT
(No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en_US/LC_TIME/coreutils.mo", 0_RDONLY) = -1 ENOENT (No
such file or directory)

```

Прочие фильтры:

```

#                successful    Trace system calls that returned without an error code.  The
-z option has the effect of status=successful.

```

```
#                failed      Trace system calls that returned with an error code.  The -Z
option has the effect of status=failed.

#                unfinished   Trace system calls that did not return.  This might happen,
for example, due to an execve call in a neighbour thread.

#                unavailable   Trace system calls that returned but strace failed to fetch
the error status.

#                detached     Trace system calls for which strace detached before the
return.


# Можно комбинировать:
root@test0:~# strace -e openat,close -e status=failed ls -la > /dev/null
openat(AT_FDCWD, "/usr/share/locale/en_US.UTF-8/LC_TIME/coreutils.mo", O_RDONLY) = -1 ENOENT
(No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en_US.utf8/LC_TIME/coreutils.mo", O_RDONLY) = -1 ENOENT
(No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en_US/LC_TIME/coreutils.mo", O_RDONLY) = -1 ENOENT (No
such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en.UTF-8/LC_TIME/coreutils.mo", O_RDONLY) = -1 ENOENT (No
such file or directory)
```

Revision #2

Created 2026-08-28 11:42:10 MSK by Ivan

Updated 2026-08-28 12:03:51 MSK by Ivan