

Modern strace

Dmitry Levin

BaseALT

Linux Piter #4

Tracing output format

- print instruction pointer: `-i` option
- print timestamps: `-r`, `-t`, `-tt`, `-ttt`, and `-T` options
- string format: `-x` and `-xx` options
- string size: `-s` option
- abbreviate output: `-e abbrev=set`, `-v` option
- dereference structures: `-e verbose=set`
- print raw undecoded syscalls: `-e raw=set`
- print signals: `-e signal=set`
- dump the data read from the specified descriptors: `-e read=set`
- dump the data written to the specified descriptors: `-e write=set`

System call filtering

- trace only the specified set of system calls: **-e trace=*set***

System call statistics

- count time, calls, and errors for each system call: **-c** option
- sort the histogram printed by the **-c** option: **-S *sortby*** option

Tracing control

- attach to existing processes: **-p *pid*** option
- trace child processes: **-f** option
- write the trace to a file or pipeline: **-o *filename*** option
- write traces of processes to separate files: **-ff -o *filename***

Tracing output format

- pathnames accessed by name or descriptor: **-y** option
- network protocol associated with descriptors: **-yy** option
- stack of function calls: **-k** option
- open **-o** in append mode: **-A** option
- format of named constants and flags: **-X** option

System call filtering

- pathnames accessed by name or descriptor: **-P** option
- regular expressions: **-e trace=/*regex***
- optional specifications: **-e trace=?*spec***
- new syscall classes: **%stat, %lstat, %fstat, %statfs, %fstatfs, %%stat, %%statfs**

System call statistics

- wall clock time spent in syscalls: **-w** option
- combine statistics with regular output: **-C** option

Tracing control

- attach to multiple processes: **-p *pid_set*** option
- detach on execve: **-b execve** option
- run as a detached grandchild: **-D** option
- interruptibility: **-I** option
- postprocessing: **strace-log-merge**

System call tampering

- fault injection:
-e **inject=set:error=errno[:when=expr][:syscall=syscall]**
- return value injection:
-e **inject=set:retval=value[:when=expr][:syscall=syscall]**
- signal injection:
-e **inject=set:signal=set**
- delay injection:
-e **inject=set:delay_enter=usecs**
-e **inject=set:delay_exit=usecs**

glibc: open or openat?

```
glibc-2.25$ strace -qq -e open cat /dev/null
open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
open("/lib64/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
open("/dev/null", O_RDONLY) = 3
glibc-2.26$ strace -qq -e open cat /dev/null
glibc-2.26$ strace -qq -e openat cat /dev/null
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
openat(AT_FDCWD, "/lib64/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
openat(AT_FDCWD, "/dev/null", O_RDONLY) = 3
glibc-2.25$ strace -qq -e openat cat /dev/null
```

traditional approach is not portable

```
riscv$ strace -e open,openat
strace: invalid system call 'open'
```

naive approach is inexact

```
$ asinfo --set-arch x86_64,riscv --list-abi \
  --nargs --get-sname /open
```

		x86_64	x86_64	riscv	riscv
N	Syscall name	64bit	32bit	64bit	32bit
1	mq_open	4	4	4	4
2	open	3	3	-	-
3	open_by_handle_at	3	3	3	3
4	openat	4	4	4	4
5	perf_event_open	5	5	5	5

`asinfo` stands for **A**dvanced **S**yscall **I**NFOrmation tool

accurate and portable approach

```
$ strace -e '/^open(at)?$'
```


traditional syscall classes

- **desc**: take or return a descriptor
- **file**: take a file name
- **memory**: memory mapping, memory policy
- **process**: process management
- **signal**: signal related
- **ipc**: SysV IPC related
- **network**: network related

```
$ strace -e trace=class
```

```
$ strace -e class
```

all syscall classes now have % prefix

```
$ strace -e trace=%class
```

```
$ strace -e %class
```

new syscall classes

- %stat, %lstat, %fstat
- %statfs, %fstatfs
- %%stat = %stat + %lstat + %fstat + statx
- %%statfs = %statfs + %fstatfs + ustat

strace -y -e %%stat ls /var/empty

```
fstat(3</etc/ld.so.cache>, st_mode=S_IFREG|0644, st_size=30341, ...) = 0
...
fstat(3</proc/filesystems>, st_mode=S_IFREG|0444, st_size=0, ...) = 0
stat("/var/empty", st_mode=S_IFDIR|0555, st_size=40, ...) = 0
fstat(3</var/empty>, st_mode=S_IFDIR|0555, st_size=40, ...) = 0
+++ exited with 0 +++
```

```
%stat + %lstat + %fstat + statx = %%stat
```

```
$ asinfo --set-arch x86_64,riscv --list-abi --nargs --get-sname %%stat
```

		x86_64	x86_64	riscv	riscv
N	Syscall name	64bit	32bit	64bit	32bit
1	fstat	2	2	2	-
2	fstat64	-	2	-	2
3	fstatat64	-	4	-	4
4	lstat	2	2	-	-
5	lstat64	-	2	-	-
6	newfstatat	4	-	4	-
7	oldfstat	-	2	-	-
8	oldlstat	-	2	-	-
9	oldstat	-	2	-	-
10	stat	2	2	-	-
11	stat64	-	2	-	-
12	statx	5	5	5	5

```
$ strace -yy -e %desc,%network netcat 127.0.0.1 22 </dev/null
...
socket(AF_INET, SOCK_STREAM, IPPROTO_TCP) = 3<TCP: [518663]>
connect(3<TCP: [518663]>, sa_family=AF_INET, sin_port=htons(22),
        sin_addr=inet_addr("127.0.0.1"), 16) = 0
poll([fd=3<TCP: [127.0.0.1:45678->127.0.0.1:22]>, events=POLLIN,
      fd=0</dev/null<char 1:3>>, events=POLLIN], 2, -1) = 1 ([fd=0, revents=POLLIN])
read(0</dev/null<char 1:3>>, "", 2048) = 0
shutdown(3<TCP: [127.0.0.1:45678->127.0.0.1:22]>, SHUT_WR) = 0
poll([fd=3<TCP: [127.0.0.1:45678->127.0.0.1:22]>, events=POLLIN, fd=-1], 2, -1)
    = 1 ([fd=3, revents=POLLIN])
read(3<TCP: [127.0.0.1:45678->127.0.0.1:22]>, "SSH-2.0-OpenSSH_7.9\r\n", 2048) = 21
write(1</dev/pts/9<char 136:9>>, "SSH-2.0-OpenSSH_7.9\r\n", 21) = 21
poll([fd=3<TCP: [127.0.0.1:45678->127.0.0.1:22]>, events=POLLIN, fd=-1], 2, -1)
    = 1 ([fd=3, revents=POLLIN|POLLHUP])
read(3<TCP: [127.0.0.1:45678->127.0.0.1:22]>, "", 2048) = 0
shutdown(3<TCP: [127.0.0.1:45678->127.0.0.1:22]>, SHUT_RD)
    = -1 ENOTCONN (Transport endpoint is not connected)
close(3<TCP: [127.0.0.1:45678->127.0.0.1:22]>) = 0
+++ exited with 0 +++
```

```
strace -qq -P /dev/full cat /dev/null > /dev/full
```

```
fstat(1, st_mode=S_IFCHR|0666, st_rdev=makedev(1, 7), ...) = 0  
close(1) = 0
```

```
strace -k -qq -P /dev/full cat /dev/null > /dev/full
```

```
fstat(1, st_mode=S_IFCHR|0666, st_rdev=makedev(1, 7), ...) = 0  
> /lib64/libc-2.27.so(__fxstat64+0x13) [0xe79c3]  
> /bin/cat(main+0x1b3) [0x4017e3]  
> /lib64/libc-2.27.so(__libc_start_main+0xe6) [0x21bd6]  
> /bin/cat(_start+0x29) [0x402179]  
close(1) = 0  
> /lib64/libc-2.27.so(__close_nocancel+0x7) [0xe8b47]  
> /lib64/libc-2.27.so(_IO_file_close_it@@GLIBC_2.2.5+0x67) [0x79fd7]  
> /lib64/libc-2.27.so(fclose@@GLIBC_2.2.5+0x136) [0x6d376]  
> /bin/cat(close_stream+0x19) [0x404ce9]  
> /bin/cat(close_stdout+0x11) [0x402691]  
> /lib64/libc-2.27.so(__run_exit_handlers+0x170) [0x379c0]  
> /lib64/libc-2.27.so(exit+0x19) [0x37ab9]  
> /lib64/libc-2.27.so(__libc_start_main+0xed) [0x21bdd]  
> /bin/cat(_start+0x29) [0x402179]
```

strace -e /open cat /dev/null

```
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
openat(AT_FDCWD, "/lib64/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
openat(AT_FDCWD, "/dev/null", O_RDONLY) = 3
+++ exited with 0 +++
```

strace -X verbose -e /open cat /dev/null

```
openat(-100 /* AT_FDCWD */, "/etc/ld.so.cache",
        0x80000 /* O_RDONLY|O_CLOEXEC */) = 3
openat(-100 /* AT_FDCWD */, "/lib64/libc.so.6",
        0x80000 /* O_RDONLY|O_CLOEXEC */) = 3
openat(-100 /* AT_FDCWD */, "/dev/null", 0 /* O_RDONLY */) = 3
+++ exited with 0 +++
```

strace -X raw -e /open cat /dev/null

```
openat(-100, "/etc/ld.so.cache", 0x80000) = 3
openat(-100, "/lib64/libc.so.6", 0x80000) = 3
openat(-100, "/dev/null", 0) = 3
+++ exited with 0 +++
```

strace -c sleep 1

% time	seconds	usecs/call	calls	errors	syscall
31.45	0.000078	19	4		close
25.40	0.000063	15	4		mprotect
12.90	0.000032	32	1		nanosleep
11.29	0.000028	5	5		mmap
8.47	0.000021	5	4		brk
8.06	0.000020	20	1		munmap
2.42	0.000006	6	1		arch_prctl
0.00	0.000000	0	1		read
0.00	0.000000	0	2		fstat
0.00	0.000000	0	1	1	access
0.00	0.000000	0	1		execve
0.00	0.000000	0	2		openat
100.00	0.000248		27	1	total

```
strace -c -w sleep 1
```

% time	seconds	usecs/call	calls	errors	syscall
99.91	1.000184	1000183	1		nanosleep
0.04	0.000367	366	1		execve
0.02	0.000216	53	4		close
0.01	0.000087	17	5		mmap
0.01	0.000075	18	4		mprotect
0.01	0.000052	13	4		brk
0.00	0.000037	18	2		openat
0.00	0.000027	26	1		munmap
0.00	0.000024	12	2		fstat
0.00	0.000019	19	1	1	access
0.00	0.000015	14	1		read
0.00	0.000013	13	1		arch_prctl
100.00	1.001116		27	1	total

foreground strace

```
$ echo $$ && strace -e none sh -c 'echo $PPID'  
1234  
23456  
+++ exited with 0 +++  
$ echo $$ && strace -e none sh -c 'echo $PPID'  
1234  
23459  
+++ exited with 0 +++
```

background strace

```
$ echo $$ && strace -D -e none sh -c 'echo $PPID'  
1234  
1234  
+++ exited with 0 +++
```

```
$ strace -o log -ff -tt -e trace=execve,nanosleep sh -c 'sleep 0.1 & sleep 0.2 & sleep 0.3'
$ strace-log-merge log
13475 21:13:52.040837 execve("/bin/sh", ["sh", "-c", "sleep 0.1 & sleep 0.2 & sleep 0."...],
    0x7ffde54b2450 /* 33 vars */) = 0
13478 21:13:52.044050 execve("/bin/sleep", ["sleep", "0.3"], 0x5631be4f87a8 /* 33 vars */) = 0
13476 21:13:52.044269 execve("/bin/sleep", ["sleep", "0.1"], 0x5631be4f87a8 /* 33 vars */) = 0
13477 21:13:52.044389 execve("/bin/sleep", ["sleep", "0.2"], 0x5631be4f87a8 /* 33 vars */) = 0
13478 21:13:52.046207 nanosleep({tv_sec=0, tv_nsec=300000000}, NULL) = 0
13476 21:13:52.046303 nanosleep({tv_sec=0, tv_nsec=100000000}, NULL) = 0
13477 21:13:52.046318 nanosleep({tv_sec=0, tv_nsec=200000000}, NULL) = 0
13476 21:13:52.146852 +++ exited with 0 +++
13475 21:13:52.146942 --- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED,
    si_pid=13476, si_uid=1000, si_status=0, si_utime=0, si_stime=0} ---
13477 21:13:52.247782 +++ exited with 0 +++
13475 21:13:52.247885 --- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED,
    si_pid=13477, si_uid=1000, si_status=0, si_utime=0, si_stime=0} ---
13478 21:13:52.347680 +++ exited with 0 +++
13475 21:13:52.347786 --- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED,
    si_pid=13478, si_uid=1000, si_status=0, si_utime=0, si_stime=0} ---
13475 21:13:52.348069 +++ exited with 0 +++
```

```
-e inject=set:error=errno[:when=expr][:syscall=syscall]
```

inject=set – fault injection for the specified set of syscalls

error=errno – the error code to fail syscalls with

when=expr – when to inject, in *first[+[step]]* form

syscall=syscall – inject the specified syscall instead of -1

```
strace -e /open -e inject=all:error=EACCES:when=3 \  
cat /dev/full /dev/null
```

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

```
openat(AT_FDCWD, "/lib64/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
```

```
openat(AT_FDCWD, "/dev/full", O_RDONLY)  
= -1 EACCES (Permission denied) (INJECTED)
```

```
cat: /dev/full: Permission denied
```

```
openat(AT_FDCWD, "/dev/null", O_RDONLY) = 3
```

```
+++ exited with 1 +++
```

python3.5 bug: error opening /dev/urandom

```
$ strace -P /dev/urandom -e inject=%file:error=ENOENT python3
openat(AT_FDCWD, "/dev/urandom", O_RDONLY|O_CLOEXEC)
 = -1 ENOENT (No such file or directory) (INJECTED)
Fatal Python error: Failed to open /dev/urandom
--- SIGSEGV {si_signo=SIGSEGV, si_code=SEGV_MAPERR, si_addr=0x50} ---
+++ killed by SIGSEGV +++
Segmentation fault
```

python3.5 bug: error reading /dev/urandom

```
$ strace -a0 -e read -P /dev/urandom -e inject=all:error=EIO python3
read(3, 0x8db610, 24) = -1 EIO (Input/output error) (INJECTED)
Fatal Python error: Failed to read bytes from /dev/urandom
--- SIGSEGV {si_signo=SIGSEGV, si_code=SEGV_MAPERR, si_addr=0x50} ---
+++ killed by SIGSEGV +++
Segmentation fault
```

glibc \leq 2.25 dynamic linker bug

```
$ strace -e mprotect -efault=all:error=EPERM:when=1 pwd
mprotect(0x7fabcd00f000, 2097152, PROT_NONE)
= -1 EPERM (Operation not permitted) (INJECTED)
mprotect(0x7fabcd20f000, 16384, PROT_READ) = 0
mprotect(0x606000, 4096, PROT_READ)      = 0
mprotect(0x7fabcd441000, 4096, PROT_READ) = 0
/
+++ exited with 0 +++
```

glibc \geq 2.26: with a proper check

```
$ strace -e mprotect -efault=all:error=EPERM:when=1 pwd
mprotect(0x7fabcd00f000, 2097152, PROT_NONE)
= -1 EPERM (Operation not permitted) (INJECTED)
pwd: error while loading shared libraries: libc.so.6:
cannot change memory protections
+++ exited with 127 +++
```

```
-e inject=set:retval=value[:when=expr][:syscall=syscall]
```

inject=set – fault injection for the specified set of syscalls

retval=value – the return value to return

when=expr – when to inject, in *first[+[step]]* form

syscall=syscall – inject the specified syscall instead of -1

example: recovery of temporary files

```
$ cat script.sh
t='mktemp'; trap 'rm -f "$t"' 0; echo secret $$ > $t
$ strace -qq -f -e signal=none -e /unlink
  -e inject=all:retval=0 sh script.sh
[pid 347] unlinkat(AT_FDCWD, "/tmp/tmp.l1AlwyCYH3", 0) = 0 (INJECTED)
$ cat /tmp/tmp.l1AlwyCYH3
secret 345
```

syscall delay injection

```
strace -e inject=set:delay__enter=usecs
```

```
strace -e inject=set:delay__exit=usecs
```

```
dd if=/dev/zero of=/dev/null bs=1M count=10
```

```
10+0 records in
```

```
10+0 records out
```

```
10485760 bytes (10 MB, 10 MiB) copied, 0.00211354 s, 5.0 GB/s
```

```
strace -einject=write:delay__exit=100000 -ewrite -o/dev/null \
```

```
dd if=/dev/zero of=/dev/null bs=1M count=10
```

```
10+0 records in
```

```
10+0 records out
```

```
10485760 bytes (10 MB, 10 MiB) copied, 1.10658 s, 9.5 MB/s
```

Questions?

homepage

<https://strace.io>

strace.git

<https://github.com/strace/strace.git>

<https://gitlab.com/strace/strace.git>

mailing list

strace-devel@lists.strace.io

IRC channel

[#strace@freenode](#)

