

8 Options to Trace/Debug Programs using Linux strace Command

 linuxide.com/linux-command/linux-strace-command-examples

August 18, 2014

The strace is the tool that helps in debugging issues by tracing system calls executed by a program. It is handy when you want to see how the program interacts with the operating system, like what system calls are executed in what order.

This simple yet very powerful tool is available for almost all the Linux based operating systems and can be used to debug a large number of programs.

1. Command Usage

Let's see how we can use strace command to trace the execution of a program.

In the simplest form, any command can follow strace. It will list a whole lot of system calls. Not all of it would make sense at first, but if you're really looking for something particular, then you should be able to figure something out of this output.

Lets see the system calls trace for simple ls command.

```
raghu@raghu-Linuxide ~ $ strace ls
```

```
raghu@raghu-Linuxide ~ $ strace ls
execve("/bin/ls", ["ls"], [/* 39 vars */]) = 0
brk(0)                                = 0x989c000
access("/etc/ld.so.nohwcap", F_OK)     = -1 ENOENT (No such file or directory)
mmap2(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xb77c8000
access("/etc/ld.so.preload", R_OK)     = -1 ENOENT (No such file or directory)
open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat64(3, {st_mode=S_IFREG|0644, st_size=86538, ...}) = 0
mmap2(NULL, 86538, PROT_READ, MAP_PRIVATE, 3, 0) = 0xb77b2000
close(3)                               = 0
access("/etc/ld.so.nohwcap", F_OK)     = -1 ENOENT (No such file or directory)
open("/lib/i386-linux-gnu/libselinux.so.1", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\1\1\1\0\0\0\0\0\0\0\0\0\3\0\3\0\1\0\0\0\340B\0\0004\0\0\0"... , 512) = 512
fstat64(3, {st_mode=S_IFREG|0644, st_size=120668, ...}) = 0
mmap2(NULL, 125884, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0xb7793000
mmap2(0xb77b0000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1c) = 0xb77b0000
close(3)                               = 0
access("/etc/ld.so.nohwcap", F_OK)     = -1 ENOENT (No such file or directory)
open("/lib/i386-linux-gnu/librt.so.1", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\1\1\1\0\0\0\0\0\0\0\0\0\3\0\3\0\1\0\0\0\0p\31\0\0004\0\0\0"... , 512) = 512
fstat64(3, {st_mode=S_IFREG|0644, st_size=30696, ...}) = 0
mmap2(NULL, 33352, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0xb778a000
```

This output shows the first few lines for strace command. The rest of the output is truncated.

```
getdents64(3, /* 0 entries */ , 32768) = 0
close(3)                               = 0
fstat64(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(136, 0), ...}) = 0
mmap2(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xb7776000
write(1, "dead.letter Desktop Documents "..., 86dead.letter Desktop Documents Downloads Music Pictures Public Templates Videos
) = 86
close(1)                               = 0
munmap(0xb7776000, 4096)               = 0
close(2)                               = 0
exit_group(0)                          = ?
raghu@raghu-Linuxide ~ $
```

The above part of the output shows the write system call where it outputs to STDOUT the current directory's listing. Following image shows the listing of the directory by ls command

(without strace).

```
| raghu@raghu-Linuxide ~ $ ls
raghu@raghu-Linuxide ~ $ ls
dead.letter Desktop Documents Downloads Music Pictures Public Templates Videos
raghu@raghu-Linuxide ~ $ █
```

1.1 Find configuration file read by program

One use of strace (Except debugging some problem) is that you can find out which configuration files are read by a program. For example,

```
| raghu@raghu-Linuxide ~ $ strace php 2>&1 | grep php.ini
raghu@raghu-Linuxide ~ $ strace php 2>&1 | grep php.ini
open("/usr/bin/php.ini", 0_RDONLY|O_LARGEFILE) = -1 ENOENT (No such file or directory)
open("/etc/php5/cli/php.ini", 0_RDONLY|O_LARGEFILE) = 3
```

1.2 Trace specific system call

The -e option to strace command can be used to display certain system calls only (for example, open, write etc.)

Lets trace only 'open' system call for cat command.

```
| raghu@raghu-Linuxide ~ $ strace -e open cat dead.letter
raghu@raghu-Linuxide ~ $ strace -e open cat dead.letter
open("/etc/ld.so.cache", 0_RDONLY|O_CLOEXEC) = 3
open("/lib/i386-linux-gnu/libc.so.6", 0_RDONLY|O_CLOEXEC) = 3
open("/usr/lib/locale/locale-archive", 0_RDONLY|O_LARGEFILE|O_CLOEXEC) = 3
open("dead.letter", 0_RDONLY|O_LARGEFILE) = 3
```

1.3 Stracing a process

The strace command can not only be used on the commands, but also on the running processes with -p option.

```
| raghu@raghu-Linuxide ~ $ sudo strace -p 1846
raghu@raghu-Linuxide ~ $ sudo strace -p 1846
Process 1846 attached - interrupt to quit
restart_syscall(<... resuming interrupted call ...>) = -1 ETIMEDOUT (Connection timed out)
futexp(0x9fe32e0, FUTEX_WAKE_PRIVATE, 1) = 0
clock_gettime(CLOCK_REALTIME, {1407781647, 393386549}) = 0
clock_gettime(CLOCK_REALTIME, {1407781647, 394867036}) = 0
futexp(0x9fe32b4, FUTEX_WAIT_PRIVATE, 209711, {0, 48519513}) = -1 ETIMEDOUT (Connection timed out)
futexp(0x9fe32e0, FUTEX_WAKE_PRIVATE, 1) = 0
```

1.4 Statistical summary of strace

The summary of the system calls, time of execution, errors etc. can be displayed in a neat manner with -c option:

```
| raghu@raghu-Linuxide ~ $ strace -c ls
```

```

raghu@raghu-Linuxide ~ $ strace -c ls
dead.letter Desktop Documents Downloads Music Pictures Public Templates Videos
% time      seconds  usecs/call      calls      errors syscall
-----
70.35      0.001248          1248          1          write
10.94      0.000194           19         10          open
 9.92      0.000176           7         25         mmap2
 8.79      0.000156          12         13          close
 0.00      0.000000           0           9          read
 0.00      0.000000           0           1          execve
 0.00      0.000000           0           9          9 access
 0.00      0.000000           0           3          brk
 0.00      0.000000           0           2          ioctl
 0.00      0.000000           0           3          munmap
 0.00      0.000000           0           1          uname
 0.00      0.000000           0           9          mprotect
 0.00      0.000000           0           2          rt_sigaction
 0.00      0.000000           0           1          rt_sigprocmask
 0.00      0.000000           0           1          getrlimit
 0.00      0.000000           0          11          fstat64
 0.00      0.000000           0           2          getdents64
 0.00      0.000000           0           1          1 futex
 0.00      0.000000           0           1          set_thread_area
 0.00      0.000000           0           1          set_tid_address
 0.00      0.000000           0           2          1 statfs64
 0.00      0.000000           0           1          openat
 0.00      0.000000           0           1          set_robust_list
-----
100.00      0.001774          110          11 total
raghu@raghu-Linuxide ~ $

```

1.5 Saving output

The output of strace command can be saved into a file with -o option.

```

raghu@raghu-Linuxide ~ $ sudo strace -o process_strace -p 3229

raghu@raghu-Linuxide ~ $ sudo strace -o process_strace -p 3229
Process 3229 attached - interrupt to quit
Process 3229 detached
raghu@raghu-Linuxide ~ $

```

The above command is run with sudo as it will display error in case the user ID does not match with the process owner.

1.6 Displaying timestamp

The timestamp can be displayed before each output line with -t option.

```

raghu@raghu-Linuxide ~ $ strace -t ls

```

```
raghu@raghu-Linuxide ~ $ strace -t ls
00:13:43 execve("/bin/ls", ["ls"], [/* 39 vars */]) = 0
00:13:43 brk(0) = 0x865c000
00:13:43 access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or directory)
00:13:43 mmap2(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xb773e000
00:13:43 access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)
00:13:43 open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
00:13:43 fstat64(3, {st_mode=S_IFREG|0644, st_size=86538, ...}) = 0
00:13:43 mmap2(NULL, 86538, PROT_READ, MAP_PRIVATE, 3, 0) = 0xb7728000
00:13:43 close(3) = 0
00:13:43 access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or directory)
00:13:43 open("/lib/i386-linux-gnu/libselinux.so.1", O_RDONLY|O_CLOEXEC) = 3
00:13:43 read(3, "\177ELF\1\1\1\0\0\0\0\0\0\0\3\0\3\0\1\0\0\0\340B\0\0004\0\0\0"... , 512) = 512
```

1.7 The Finer timestamp

The `-tt` option displays timestamp followed by microsecond.

```
raghu@raghu-Linuxide ~ $ strace -tt ls
```

```
raghu@raghu-Linuxide ~ $ strace -tt ls
00:14:58.705781 execve("/bin/ls", ["ls"], [/* 39 vars */]) = 0
00:14:58.714520 brk(0) = 0x86ea000
00:14:58.716235 access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or directory)
00:14:58.717986 mmap2(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xb774d000
00:14:58.727937 access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)
00:14:58.728671 open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
00:14:58.733752 fstat64(3, {st_mode=S_IFREG|0644, st_size=86538, ...}) = 0
00:14:58.735310 mmap2(NULL, 86538, PROT_READ, MAP_PRIVATE, 3, 0) = 0xb7737000
00:14:58.736490 close(3) = 0
00:14:58.737803 access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or directory)
```

The `-ttt` displays microseconds like above, but instead of printing current time, it displays the number of seconds since the epoch.

```
raghu@raghu-Linuxide ~ $ strace -ttt ls
```

```
raghu@raghu-Linuxide ~ $ strace -ttt ls
1407782765.485879 execve("/bin/ls", ["ls"], [/* 39 vars */]) = 0
1407782765.498613 brk(0) = 0x83ea000
1407782765.500229 access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or directory)
1407782765.509669 mmap2(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xb7753000
1407782765.511465 access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)
1407782765.512511 open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
1407782765.518956 fstat64(3, {st_mode=S_IFREG|0644, st_size=86538, ...}) = 0
1407782765.520321 mmap2(NULL, 86538, PROT_READ, MAP_PRIVATE, 3, 0) = 0xb773d000
1407782765.521596 close(3) = 0
1407782765.522910 access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or directory)
```

1.8 Relative Time

The `-r` option displays the relative timestamp between the system calls.

```
raghu@raghu-Linuxide ~ $ strace -r ls
```

```
raghu@raghu-Linuxide ~ $ strace -r ls
0.000000 execve("/bin/ls", ["ls"], [/* 39 vars */]) = 0
0.014999 brk(0) = 0x901b000
0.005126 access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or directory)
0.006688 mmap2(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xb7731000
0.001288 access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)
0.001435 open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
0.003601 fstat64(3, {st_mode=S_IFREG|0644, st_size=86538, ...}) = 0
0.001617 mmap2(NULL, 86538, PROT_READ, MAP_PRIVATE, 3, 0) = 0xb771b000
0.001205 close(3) = 0
0.007156 access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or directory)
```

