

I <3 strace

By Marc Paquette

What is strace?

Strace shows what system calls a program is making to the linux kernel

Basic strace

strace <executable>

```
Shell
bosh/0:~# strace ls
execve("/bin/ls", ["ls"], [/* 14 vars */]) = 0
brk(0) = 0x245a000
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or
directory)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1,
0) = 0x7f3c59d35000
access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or
directory)
open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=24749, ...}) = 0
mmap(NULL, 24749, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f3c59d2e000
close(3) = 0
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or
directory)
open("/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"..., 832) = 832
fstat(3, {st_mode=S_IFREG|0644, st_size=134296, ...}) = 0
mmap(NULL, 2238192, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3
, 0) = 0x7f3c598f2000
mprotect(0x7f3c59912000, 2093056, PROT_NONE) = 0
mmap(0x7f3c59b11000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXE
D|MAP_DENYWRITE, 3, 0x1f000) = 0x7f3c59b11000
mmap(0x7f3c59b13000, 5872, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXE
D|MAP_ANONYMOUS, -1, 0) = 0x7f3c59b13000
close(3) = 0
```

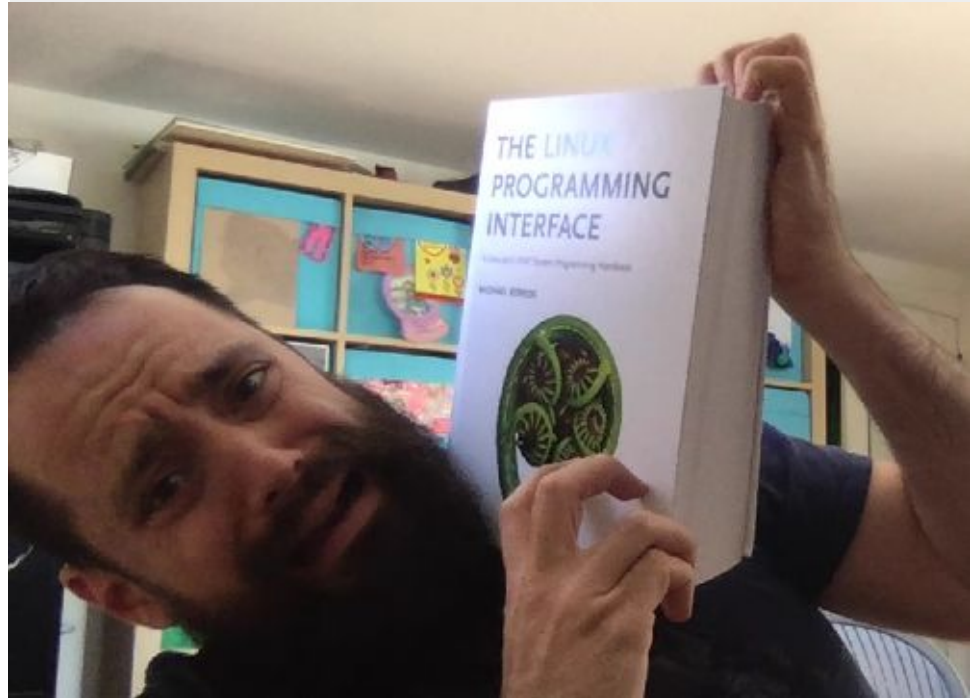
Basic strace

strace <executable>

```
Shell
bosh/0:~# strace ls
execve("/bin/ls", ["ls"], [/* 14 vars */]) = 0
brk(0) = 0x245a000
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or
directory)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1,
0) = 0x7f3c59d35000
access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or
directory)
open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=24749, ...}) = 0
mmap(NULL, 24749, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f3c59d2e000
close(3) = 0
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or
directory)
open("/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
fstat(3, {st_mode=S_IFREG|0644, st_size=134296, ...}) = 0
mmap(NULL, 2238192, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3
, 0) = 0x7f3c598f2000
mprotect(0x7f3c59912000, 2093056, PROT_NONE) = 0
mmap(0x7f3c59b11000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXE
D|MAP_DENYWRITE, 3, 0x1f000) = 0x7f3c59b11000
mmap(0x7f3c59b13000, 5872, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXE
D|MAP_ANONYMOUS, -1, 0) = 0x7f3c59b13000
close(3) = 0
```

System calls!?

OH NOES! HELP!!!11one!1!



How to read the output

```
Shell
bosh/0:~# strace ls
execve("/bin/ls", ["ls"], [/ * 14 vars */]) = 0
brk(0) = 0x245a000
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or
directory)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1,
0) = 0x7f3c59d35000
access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or
directory)
open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=24749, ...}) = 0
mmap(NULL, 24749, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f3c59d2e000
close(3) = 0
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or
directory)
open("/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\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0[\0\0\0\0\0\0\0"...
, 832) = 832
fstat(3, {st_mode=S_IFREG|0644, st_size=134296, ...}) = 0
mmap(NULL, 2238192, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3
, 0) = 0x7f3c598f2000
mprotect(0x7f3c59912000, 2093056, PROT_NONE) = 0
mmap(0x7f3c59b11000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXE
D|MAP_DENYWRITE, 3, 0x1f000) = 0x7f3c59b11000
mmap(0x7f3c59b13000, 5872, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXE
D|MAP_ANONYMOUS, -1, 0) = 0x7f3c59b13000
close(3) = 0
```







How to read the *system calls*

man man

```
MAN(1) Manual pager utils
NAME
    man - an interface to the on-line reference manuals

SYNOPSIS
    man -f [whatis options] page ...
    man [-?V] BLAH
    BLAH BLAH BLAH BLAH

DESCRIPTION
    BLAH BLAH BLAH BLAH BLAH BLAH BLAH

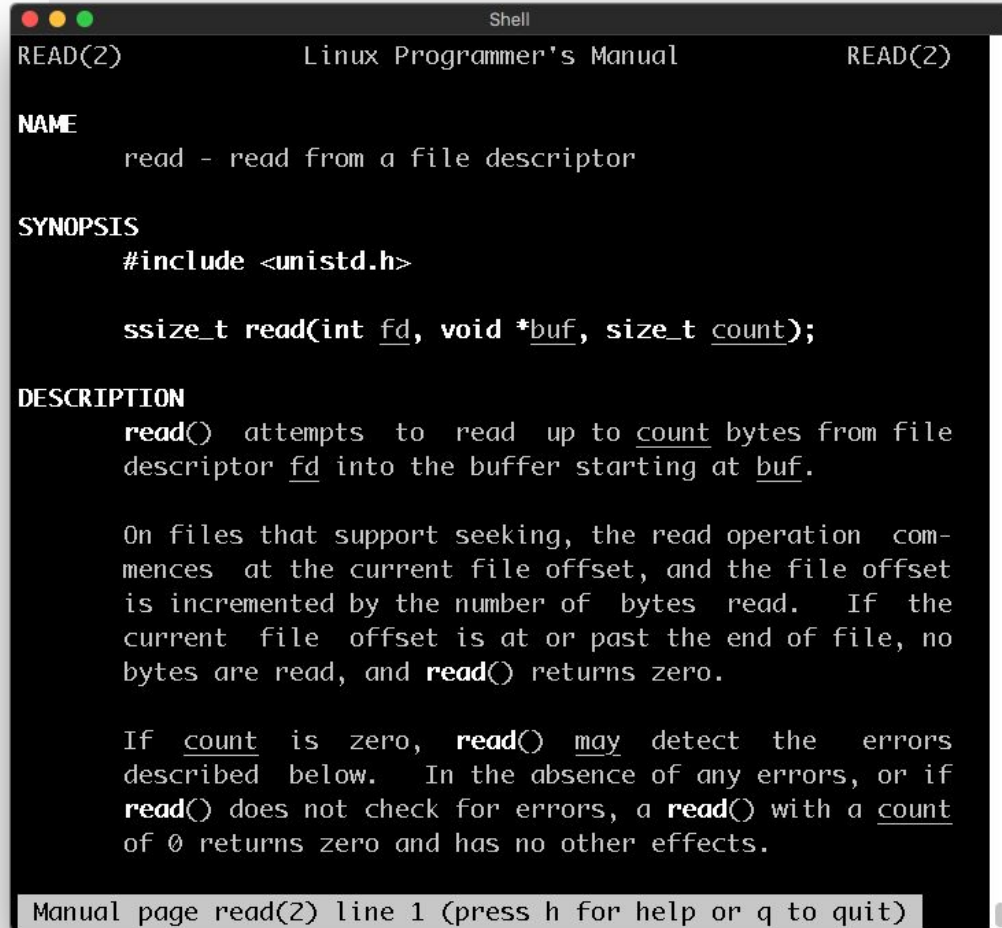
    The table below shows the section numbers of the manual followed by
    the types of pages they contain.
    1 Executable programs or shell commands
    2 System calls (functions provided by the kernel)
    3 Library calls (functions within program libraries)
    4 Special files (usually found in /dev)
    5 File formats and conventions eg /etc/passwd
    6 Games
    7 Miscellaneous (including macro packages and conventions), e.g.
    man(7), groff(7)
    8 System administration commands (usually only for root)
    9 Kernel routines [Non standard]

    A manual page consists of several sections.

-- INSERT --
```

How to read the output

man 2 read



```
Shell
READ(2)                                Linux Programmer's Manual                                READ(2)

NAME
    read - read from a file descriptor

SYNOPSIS
    #include <unistd.h>

    ssize_t read(int fd, void *buf, size_t count);

DESCRIPTION
    read() attempts to read up to count bytes from file descriptor fd into the buffer starting at buf.

    On files that support seeking, the read operation commences at the current file offset, and the file offset is incremented by the number of bytes read. If the current file offset is at or past the end of file, no bytes are read, and read() returns zero.

    If count is zero, read() may detect the errors described below. In the absence of any errors, or if read() does not check for errors, a read() with a count of 0 returns zero and has no other effects.

Manual page read(2) line 1 (press h for help or q to quit)
```

How to read the output

```
Shell
bosh/0:~# strace ls
execve("/bin/ls", ["ls"], [/ * 14 vars */]) = 0
brk(0) = 0x245a000
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or
directory)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1,
0) = 0x7f3c59d35000
access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or
directory)
open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=24749, ...}) = 0
mmap(NULL, 24749, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f3c59d2e000
close(3) = 0
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or
directory)
open("/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"... , 832) = 832
fstat(3, {st_mode=S_IFREG|0644, st_size=134296, ...}) = 0
mmap(NULL, 2238192, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3
, 0) = 0x7f3c598f2000
mprotect(0x7f3c59912000, 2093056, PROT_NONE) = 0
mmap(0x7f3c59b11000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXE
D|MAP_DENYWRITE, 3, 0x1f000) = 0x7f3c59b11000
mmap(0x7f3c59b13000, 5872, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXE
D|MAP_ANONYMOUS, -1, 0) = 0x7f3c59b13000
close(3) = 0
```

Strace args

Arguments ahoy!

Process strace

```
strace -p <pid>
```

[illegible]

Cumulative strace

strace -c

```
bosh/0:~# strace -c wc -l very_large_file.txt
96671367 very_large_file.txt
% time      seconds  usecs/call   calls   errors syscall
-----
100.00      0.016000          0    52427         read
  0.00      0.000000          0         1         write
  0.00      0.000000          0         4         open
  0.00      0.000000          0         6         close
  0.00      0.000000          0         4         fstat
  0.00      0.000000          0         9         mmap
  0.00      0.000000          0         4         mprotect
  0.00      0.000000          0         2         munmap
  0.00      0.000000          0         3         brk
  0.00      0.000000          0         3         access
  0.00      0.000000          0         1         execve
  0.00      0.000000          0         1         arch_prctl
  0.00      0.000000          0         1         fadvise64
-----
100.00      0.016000          0    52466         3 total
```

Follow threads strace

strace -f

```
Shell
bosh/0:~# ps -T -p 340849
  PID     SPID  TTY      TIME  CMD
 340849   340849 ?        00:00:00 dadoo
 340849   340850 ?        00:00:00 dadoo
 340849   340851 ?        00:00:00 dadoo
 340849   340852 ?        00:00:00 dadoo
 340849   340853 ?        00:00:00 dadoo
bosh/0:~# strace -p 340849
Process 340849 attached
futex(0xaa5890, FUTEX_WAIT, 0, NULL^CProcess 340849 detached
<detached ...>
bosh/0:~# strace -f -p 340849
Process 340849 attached with 5 threads
[pid 340853] futex(0xc420029990, FUTEX_WAIT, 0, NULL <unfinished ...>
[pid 340851] futex(0xc420028810, FUTEX_WAIT, 0, NULL <unfinished ...>
[pid 340849] futex(0xaa5890, FUTEX_WAIT, 0, NULL <unfinished ...>
[pid 340852] futex(0xac4880, FUTEX_WAIT, 0, NULL <unfinished ...>
[pid 340850] restart_syscall(<... resuming interrupted call ...>^CPro
cess 340849 detached
Process 340850 detached
<detached ...>
Process 340851 detached
Process 340852 detached
Process 340853 detached
bosh/0:~#
bosh/0:~#
bosh/0:~#
```


Time strace

strace -t

strace -tt

strace -T

```
Shell
bosh/0:~# strace -t wc -l very_small_file.txt
04:36:12 execve("/usr/bin/wc", ["wc", "-l", "very_small_file.txt"], [/* 14 vars */]) = 0
04:36:12 brk(0) = 0x111f
```

```
Shell
bosh/0:~# strace -tt wc -l very_small_file.txt
04:36:45.697492 execve("/usr/bin/wc", ["wc", "-l", "very_small_file.txt"], [/* 14 vars */]) = 0
04:36:45.698025 brk(0) = 0x2130
000
```

```
Shell
bosh/0:~# strace -T wc -l very_small_file.txt
execve("/usr/bin/wc", ["wc", "-l", "very_small_file.txt"], [/* 14 vars */]) = 0 <0.000669>
brk(0) = 0x208d000 <0.000086>
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or directory) <0.000177>
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f92d8b18000 <0.000137>
access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No
```

How to read the output

Remember me?

```
Shell
bosh/0:~# strace ls
execve("/bin/ls", ["ls"], [/ * 14 vars */]) = 0
brk(0) = 0x245a000
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or
directory)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1,
0) = 0x7f3c59d35000
access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or
directory)
open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=24749, ...}) = 0
mmap(NULL, 24749, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f3c59d2e000
close(3) = 0
access("/etc/ld.so.nohwcap", F_OK) = -1 ENOENT (No such file or
directory)
open("/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\0"..., 832) = 832
fstat(3, {st_mode=S_IFREG|0644, st_size=134296, ...}) = 0
mmap(NULL, 2238192, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f3c598f2000
, 0) = 0x7f3c598f2000
mprotect(0x7f3c59912000, 2093056, PROT_NONE) = 0
mmap(0x7f3c59b11000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1f000) = 0x7f3c59b11000
mmap(0x7f3c59b13000, 5872, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f3c59b13000
close(3) = 0
```

File descriptors suck (Yeah they do) strace

strace -y #file descriptors

```
strace -yy #sockets
```

```
bosh/0:~# strace -y ls
execve("/bin/ls", ["ls"], [/ * 14 vars */]) = 0
brk(0)                                = 0x23a2000
access("/etc/ld.so.nohwcap", F_OK)    = -1 ENOENT (No such file or dire
ctory)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7efcacb9e000
access("/etc/ld.so.preload", R_OK)    = -1 ENOENT (No such file or dire
ctory)
open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3</etc/ld.so.cache>, {st_mode=S_IFREG|0644, st_size=24749, ...}) =
0
mmap(NULL, 24749, PROT_READ, MAP_PRIVATE, 3</etc/ld.so.cache>, 0) = 0x7ef
cacb97000
close(3</etc/ld.so.cache>)            = 0
access("/etc/ld.so.nohwcap", F_OK)    = -1 ENOENT (No such file or dire
ctory)
open("/lib/x86_64-linux-gnu/libselinux.so.1", O_RDONLY|O_CLOEXEC) = 3
read(3</lib/x86_64-linux-gnu/libselinux.so.1>, "\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\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0
....", 832) = 832
fstat(3</lib/x86_64-linux-gnu/libselinux.so.1>, {st_mode=S_IFREG|0644, st
_size=134296, ...}) = 0
mmap(NULL, 2238192, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3</li
b/x86_64-linux-gnu/libselinux.so.1>, 0) = 0x7efcac75b000
mprotect(0x7efcac77b000, 2093056, PROT_NONE) = 0
mmap(0x7efcac97a000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MA
P_DENYWRITE, 3</lib/x86_64-linux-gnu/libselinux.so.1>, 0x1f000) = 0x7efca
c97a000
mmap(0x7efcac97c000, 5872, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MA
P_ANONYMOUS, -1, 0) = 0x7efcac97c000
close(3</lib/x86_64-linux-gnu/libselinux.so.1>) = 0
```

GPDB or CF Example

BOSH director's PostgreSQL

We can see the queries being run

We can see the location of the table involved in the query

We can see the Object ID of the table and the database

We can see block size of 8192

```
Shell
bosh/0:~# strace -p 1627978 -yy
Process 1627978 attached
recvfrom(8, "Q\\0\\0\\0\\35select * from pg_tables;\\0", 8192, 0, NULL, NULL) =
 30
kill(5310, SIGUSR1) = 0
open("base/16387/11894", 0_RDONLY) = 45
lseek(45</var/vcap/store/postgres-9.4/base/16387/11894>, 40960, SEEK_SET)
= 40960
read(45</var/vcap/store/postgres-9.4/base/16387/11894>, "\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\324\\5\\0\\t\\360\\37\\4 \\0\\0\\0\\0\\0\\211 \\0\\340\\237 \\0"... , 8192) = 8192
lseek(17</var/vcap/store/postgres-9.4/base/16387/11891>, 163840, SEEK_SET)
= 163840
read(17</var/vcap/store/postgres-9.4/base/16387/11891>, "\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\4\\0\\364\\0\\20\\1\\0 \\4 \\0\\0\\0\\0p\\237\\30\\1\\340\\236\\30\\1"... , 8192) = 8192
open("base/16387/11972", 0_RDONLY) = 46
read(46</var/vcap/store/postgres-9.4/base/16387/11972>, "\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0000\\0\\360\\37\\360\\37\\4 \\0\\0\\0\\0b1\\5\\0\\2\\0\\0\\0"... , 8192) = 8192
read(46</var/vcap/store/postgres-9.4/base/16387/11972>, "\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\314\\1\\250\\25\\360\\37\\4 \\0\\0\\0\\0\\330\\2370\\0\\300\\2370\\0"... , 8192) =
8192
open("base/16387/11966", 0_RDONLY) = 47
read(47</var/vcap/store/postgres-9.4/base/16387/11966>, "\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\4\\0H\\0\\340\\3\\0 \\4 \\0\\0\\0\\0\\200\\237\\370\\0\\260\\227\\234\\17"... , 8192) = 8
192
open("base/16387/11970", 0_RDONLY) = 48
read(48</var/vcap/store/postgres-9.4/base/16387\\1\\0\\0\\0\\0000\\0\\360\\37\\360\\37\\4 \\0\\0\\0\\0b1\\5\\0\\
```

```
Shell
bosh=# select * from pg_tables;
-[ RECORD 1 ]-----
-----
schemaname | pg_catalog
tablename  | pg_statistic
tableowner  | vcap
tablespace | 
hasindexes  | t
hasrules    | f
hastriggers | f
-[ RECORD 2 ]-----
```

Additional resources

Strace man page

<http://man7.org/linux/man-pages/man1/strace.1.html>

Strace zine (way better than this presentation!)

<https://jvns.ca/strace-zine-v2.pdf>

Linux Programming Interface (Book)

<https://smile.amazon.com/Linux-Programming-Interface-System-Handbook/dp/1593272200>

```
write(1, "The End\n")
```