Robert Jones
03/07/2025

1.
A) String query = "select name from students where id = " + id + ";";
      1. Id = 0 OR 1=1
         1=1 is always true
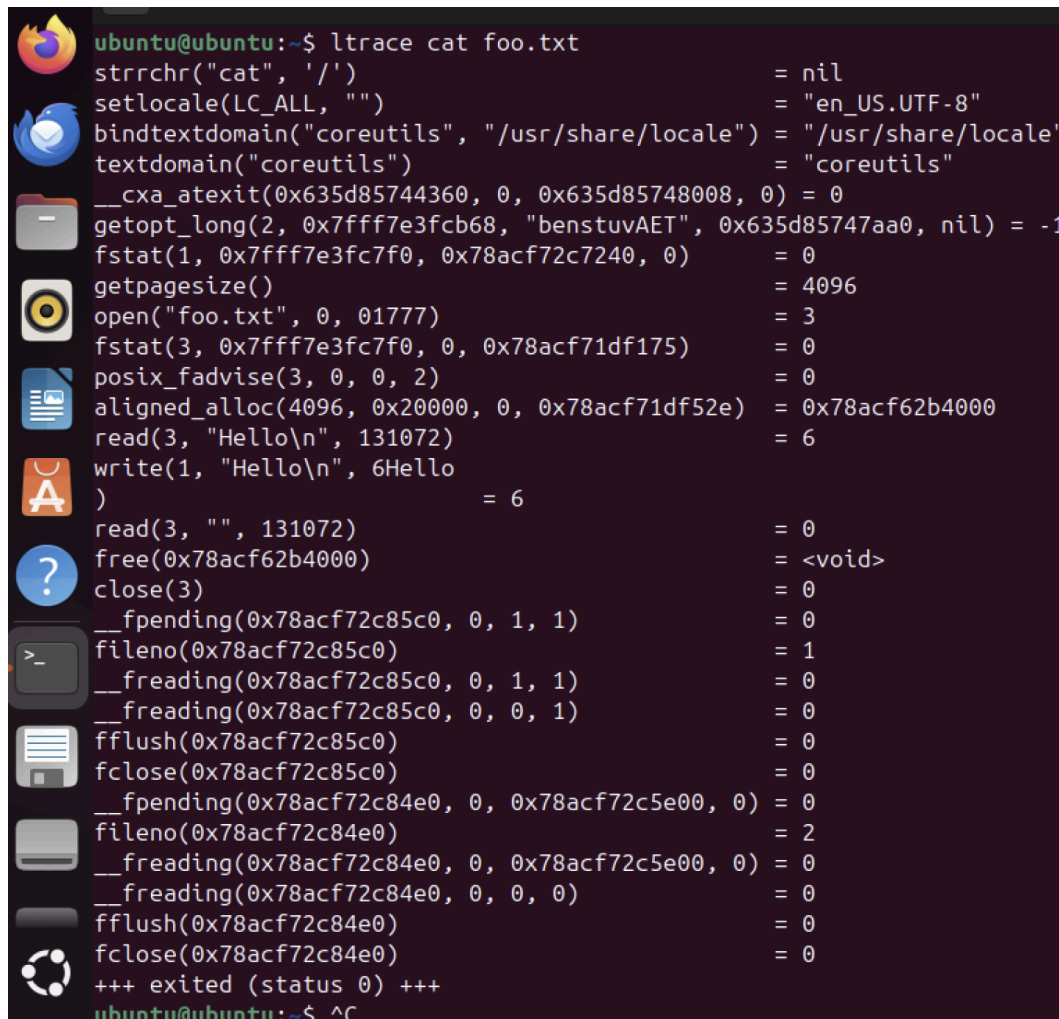   2. id = 0; DROP TABLE; Students
         The semicolon terminates the server side SQL query and appends DROP TABLE
students to delete the table.

B) String query = "select name from students where id = '" + id + ";";
      Id = '0' OR '1' = '1'
      Id = '0'; DROP TABLE; Students

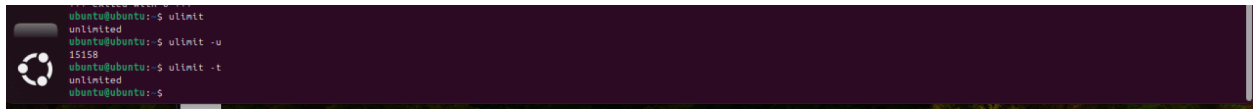2. Linux commands through Ubuntu VM

```
ubuntu@ubuntu:~$ ltrace cat foo.txt
strrchr("cat", '/')                                = nil
setlocale(LC_ALL, "")                              = "en_US.UTF-8"
bindtextdomain("coreutils", "/usr/share/locale") = "/usr/share/locale"
textdomain("coreutils")                            = "coreutils"
__cxa_atexit(0x635d85744360, 0, 0x635d85748008, 0) = 0
getopt_long(2, 0x7fff7e3fcb68, "benstuvAET", 0x635d85747aa0, nil) = -1
fstat(1, 0x7fff7e3fc7f0, 0x78acf72c7240, 0)        = 0
getpagesize()                                      = 4096
open("foo.txt", 0, 01777)                          = 3
fstat(3, 0x7fff7e3fc7f0, 0, 0x78acf71df175)        = 0
posix_fadvise(3, 0, 0, 2)                          = 0
aligned_alloc(4096, 0x20000, 0, 0x78acf71df52e)   = 0x78acf62b4000
read(3, "Hello\n", 131072)                         = 6
write(1, "Hello\n", 6Hello
)                                        = 6
read(3, "", 131072)                                = 0
free(0x78acf62b4000)                               = <void>
close(3)                                           = 0
__fpending(0x78acf72c85c0, 0, 1, 1)               = 0
fileno(0x78acf72c85c0)                            = 1
__freading(0x78acf72c85c0, 0, 1, 1)               = 0
__freading(0x78acf72c85c0, 0, 0, 1)               = 0
fflush(0x78acf72c85c0)                            = 0
fclose(0x78acf72c85c0)                            = 0
__fpending(0x78acf72c84e0, 0, 0x78acf72c5e00, 0) = 0
fileno(0x78acf72c84e0)                            = 2
__freading(0x78acf72c84e0, 0, 0x78acf72c5e00, 0) = 0
__freading(0x78acf72c84e0, 0, 0, 0)               = 0
fflush(0x78acf72c84e0)                            = 0
fclose(0x78acf72c84e0)                            = 0
+++ exited (status 0) +++
ubuntu@ubuntu:~$ ^C
```
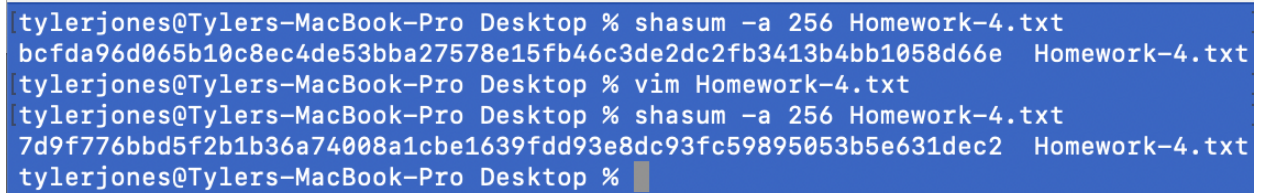
```
ubuntu@ubuntu:~$ strace cat foo.txt
execve("/usr/bin/cat", ["cat", "foo.txt"], 0x7ffd209d36a8 /* 46 vars */) = 0
brk(NULL)                               = 0x613e594a0000
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x767ddf0c6000
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=63427, ...}) = 0
mmap(NULL, 63427, PROT_READ, MAP_PRIVATE, 3, 0) = 0x767ddf0b6000
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\0\3\0>\0\1\0\0\0\220\243\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
fstat(3, {st_mode=S_IFREG|0755, st_size=2125328, ...}) = 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, 2170256, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x767ddeea4000
mmap(0x767ddeecc000, 1605632, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x767ddeecc000
mmap(0x767ddf054000, 323584, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1b0000) = 0x767ddf054000
mmap(0x767ddf0a3000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1fe000) = 0x767ddf0a3000
mmap(0x767ddf0a9000, 52624, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x767ddf0a9000
close(3)                                = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x767ddeea1000
arch_prctl(ARCH_SET_FS, 0x767ddeea1740) = 0
set_tid_address(0x767ddeea1a10)         = 5223
set_robust_list(0x767ddeea1a20, 24)     = 0
rseq(0x767ddeea2060, 0x20, 0, 0x53053053) = 0
mprotect(0x767ddf0a3000, 16384, PROT_READ) = 0
mprotect(0x613e578db000, 4096, PROT_READ) = 0
mprotect(0x767ddf0fe000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
munmap(0x767ddf0b6000, 63427)           = 0
getrandom("\x59\x7a\xf3\xc6\xed\x92\x98\x17", 8, GRND_NONBLOCK) = 8
brk(NULL)                               = 0x613e594a0000
brk(0x613e594c1000)                     = 0x613e594c1000
openat(AT_FDCWD, "/usr/lib/locale/locale-archive", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=14596880, ...}) = 0
mmap(NULL, 14596880, PROT_READ, MAP_PRIVATE, 3, 0) = 0x767dde0b5000
close(3)                                = 0
fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
openat(AT_FDCWD, "foo.txt", O_RDONLY)   = 3
fstat(3, {st_mode=S_IFREG|0664, st_size=6, ...}) = 0
fadvise64(3, 0, 0, POSIX_FADV_SEQUENTIAL) = 0
mmap(NULL, 139264, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x767dde093000
read(3, "Hello\n", 131072)              = 6
write(1, "Hello\n", 6Hello
)                                       = 6
read(3, "", 131072)                     = 0
munmap(0x767dde093000, 139264)          = 0
close(3)                                = 0
close(1)                                = 0
close(2)                                = 0
exit_group(0)                           = ?
+++ exited with 0 +++
```

When running ltrace, the first system call is to strrchr, checking for the firs instance of "/" in the string "cat" to check if there is a directory path. There is not so nil is returned. When running strace, the first system call is to execve because it is reading the system's responsibility to start the cat command.

3.
<script>
document.write(unescape('%3C%68%74%6D%6C%3E%0A%3C%74%69%74%6C%65%3E%45%6E%63%72%79%70%74%65%64%20%48%54%4D%4C%3C%2F%74%69%74%6C%65%3E%0A%3C%62%6F%64%79%3E%48%65%6C%6C%6F%20%77%6F%72%6C%64%21%3C%2F%62%6F%64%79%3E%0A%3C%2F%68%74%6D%6C%3E'));
</script>

  <html>
   <title>Encrypted HTML</title>
   <body>Hello world!</body>
  </html>

4. ulimit -u can be used to limit the number of a process a user can create and ulimit -t can be used to limit the number of seconds to be used by each process. These two will prevent exhaustion of processes and CPU time.

```
ubuntu@ubuntu:~$ ulimit
unlimited
ubuntu@ubuntu:~$ ulimit -u
15158
ubuntu@ubuntu:~$ ulimit -t
unlimited
ubuntu@ubuntu:~$
```

5.

```
[tylerjones@Tylers-MacBook-Pro Desktop % shasum -a 256 Homework-4.txt
bcfda96d065b10c8ec4de53bba27578e15fb46c3de2dc2fb3413b4bb1058d66e  Homework-4.txt
[tylerjones@Tylers-MacBook-Pro Desktop % vim Homework-4.txt
[tylerjones@Tylers-MacBook-Pro Desktop % shasum -a 256 Homework-4.txt
7d9f776bbd5f2b1b36a74008a1cbe1639fdd93e8dc93fc59895053b5e631dec2  Homework-4.txt
tylerjones@Tylers-MacBook-Pro Desktop %
```