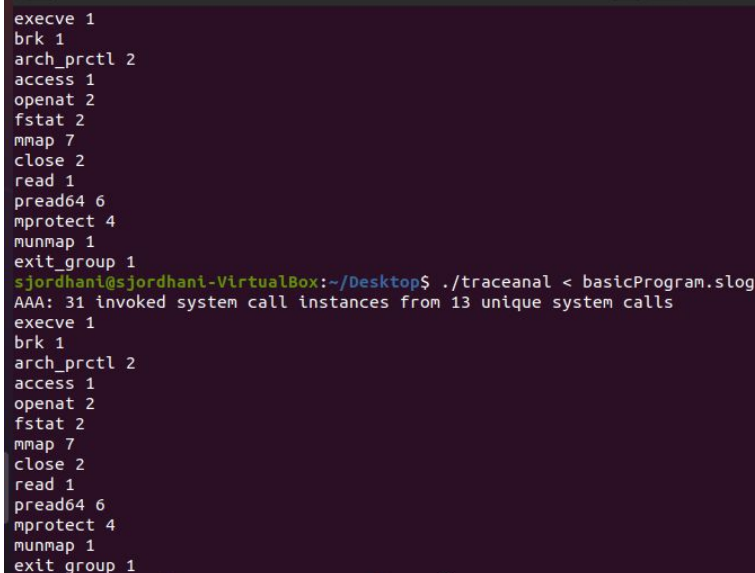


### Project 2: Part 3

1.) In order to examine start-up behaviors in programs, I created two basic programs that did not make any system calls and then analyzed it by looking at the log file and traceanal. After doing some research and determining which operations made system calls and which didn't, I made one program that simply returned 0 but did not print anything out, and another that simply did an arithmetic operation with integers but also did not print anything out. When comparing the traceanal output for both programs, it seemed that they both use the same system calls in the same sequence as shown in the screenshot below.



```
execve 1
brk 1
arch_prctl 2
access 1
openat 2
fstat 2
mmap 7
close 2
read 1
pread64 6
mprotect 4
munmap 1
exit_group 1
sjordhani@sjordhani-VirtualBox:~/Desktop$ ./traceanal < basicProgram.slog
AAA: 31 invoked system call instances from 13 unique system calls
execve 1
brk 1
arch_prctl 2
access 1
openat 2
fstat 2
mmap 7
close 2
read 1
pread64 6
mprotect 4
munmap 1
exit_group 1
```

Additionally, when I compared the log file and traceanal output of the basic programs to that of a command such as “ls”, it seemed that the command which did make system calls (ls), had the same first 10 system calls as that of the programs that did not make system calls. These 10 first system calls included, in order, “execve, brk, arch\_prctl, access, openat, fstat, mmap, close, openat, read”. In conclusion, they did exhibit a similar start-up in terms of system calls and relative sequence.

2.) When running the ls command twice and retrieving the slog file for each and then running traceanal on both of them, it seems that although the count of system calls varies between one another, the sequences of the calls that are made in each case are perfectly similar. After I tested running the same command twice, I ran and compared the traceanal results of ls and ls -R. As we know, “ls” displays a list of files in the current directory that you are in, while “ls -R” displays the files in all directories. As a result, when comparing the results of a command with the results of the command with different command line arguments, it seems that the count of system calls

changes however there is no major variation in the sequence apart from one swap of calls that occurs in the `ls -R` command. Additionally, we can conclude that the sequence does not change if the amount of data or duration varies considering `ls -R` takes longer and works to grab more data than `ls` because `ls -R` is looking through all directories in the system.

3.) For my final test, I compared the traceanal output when running `ls` command and when running the `grep` command on a txt file. Once again we see a similar start-up behavior between the two commands, however if we were to compare the command-specific portion we would see that they each have different calls that are being made and in different sequences. If we look after the `prlimit64` system call for both commands we see that for `grep` it follows with “`futex`, `sigaltstack`, `stat`, `write`, `exit_group`” while `ls` command follows with “`statfs`, `ioctl`, `getdents64`, `write`, `exit_group`”. However, another similarity that we observe is that they both end with the system calls `write` and then `exit_group`.

```
sjordhani@sjordhani-VirtualBox:~/Desktop$ ./traceanal < grep.slog
AAA: 109 invoked system call instances from 22 unique system calls
execve 1
brk 3
arch_prctl 2
access 1
openat 21
fstat 11
mmap 23
close 11
read 8
pread64 8
mprotect 7
munmap 1
set_tid_address 1
set_robust_list 1
rt_sigaction 3
rt_sigprocmask 1
prlimit64 1
futex 1
sigaltstack 1
stat 1
write 1
exit_group 1
```

```
sjordhani@sjordhani-VirtualBox:~/Desktop$ ./traceanal < ls.slog
AAA: 109 invoked system call instances from 22 unique system calls
execve 1
brk 3
arch_prctl 2
access 2
openat 9
fstat 10
mmap 27
close 11
read 7
mprotect 9
pread64 8
munmap 1
set_tid_address 1
set_robust_list 1
rt_sigaction 2
rt_sigprocmask 1
prlimit64 1
statfs 2
ioctl 2
getdents64 2
write 6
exit_group 1
```