Özgür Öney 21101821 CS342 – 2

Assoc. Prof. Dr. Özcan Öztürk

Homework 1

Assigned: Sep 11, 2015, Friday

Due date: Sep 19, 2015, Saturday, 23:55pm.

Q2) Kernel executable is in the path lib/modules/3.19.0-28-generic/kernel 3.19.0-28-generic is the version of my running kernel. After given instructions are followed and root directory has been changed to source code's directory, following subdirectories were reached.

```
🔊 🖨 🗊 ozgur@ozgur: ~/indirilenler/linux-source-3.19.0_3.19.0-28.30_all/usr/src/linux-source-3.19.0
ozgur@ozgur:~/İndirilenler/linux-source-3.19.0_3.19.0-28.30_all/usr/src/linux-so
urce-3.19.0/linux-source-3.19.0$ ls
         Documentation include
                                 kernel
                                                               security virt
                                              net
block
         drivers
                        init
                                 lib
                                              README
                                                               sound
                                 MAINTAINERS REPORTING-BUGS tools
COPYING dropped.txt
                        ipc
CREDITS firmware
                        Kbuild Makefile
                                                               ubuntu
                                              samples
                        Kconfig mm
                                              scripts
ozgur@ozgur:~/İndirilenler/linux-source-3.19.0_3.19.0-28.30_all/usr/src/linux-so
urce-3.19.0/linux-source-3.19.0$
```

Q4) **strace** command in Linux is used to trace system calls. Since every user process has an interact with kernel by invoking system calls, **strace** utility logs these system calls respectively. Thanks to these, programs' interactions with the system could be monitored and tracked down. Following figure shows what Terminal did to show trace of executable after compilation.

strace man command is used to display the **man** page of the "**strace**" command, whereby man is the default built user manual page under many Linux version. Just after **strace man** command is used in Terminal, following screen could be seen.

```
STRACE(1)
                                                                                                                                                                                                                                                  STRACE(1)
                                                                                                             General Commands Manual
NAME
            strace - trace system calls and signals
            strace [-CdffhiqrtttTvVxxy] [-In] [-b<u>execve]</u> [-e<u>expr]</u>... [-a<u>column] [-ofile] [-sstrsize]</u> [-P<u>path]</u>... -p<u>pid</u>... / [-D] [-E<u>var[=val]]</u>... [-u<u>username] command</u> [<u>args</u>]
            strace -c[df] [-In] [-bexecve] [-eexpr]... [-Ooverhead] [-Ssortby] -ppid... / [-D] [-Evar[=val]]... [-uusername] command [args]
DESCRIPTION
            In the simplest case strace runs the specified <u>command</u> until it exits. It intercepts and records the system calls which are called by a process and the signals which are received by a process. The name of each system call, its arguments and its return value are printed on standard error or to the file specified with the -o option.
            strace is a useful diagnostic, instructional, and debugging tool. System administrators, diagnosticians and trouble-shooters will find it invaluable for solving problems with programs for which the source is not readily available since they do not need to be recompiled in order to trace them. Students, hackers and the overly-curious will find that a great deal can be learned about a system and its system calls by tracing even ordinary programs. And programmers will find that since system calls and signals are events that happen at the user/kernel interface, a close examination of this boundary is very useful for bug isolation, sanity checking and
             attempting to capture race conditions.
            Each line in the trace contains the system call name, followed by its arguments in parentheses and its return value. An example from stracing the command ``cat /dev/null'' is:
            open("/dev/null", O_RDONLY) = 3
            Errors (typically a return value of -1) have the errno symbol and error string appended.
            open("/foo/bar", O_RDONLY) = -1 ENOENT (No such file or directory)
            Signals are printed as a signal symbol and a signal string. An excerpt from stracing and interrupting the command ``sleep 666'' is:
            sigsuspend([] <unfinished ...>
--- SIGINT (Interrupt) ---
+++ killed by SIGINT +++
If a system call is being executed and meanwhile another one is being called from a different thread/process then strace will try to preserve the order of those events and mark the ongoing call as being <u>unfinished</u>. When the call returns it will be marked as Manual page strace(1) line 1 (press h for help or q to quit)
```