```
.TH P1
.SH Name
P1 - System Calls
.SH Synopsis
Creating multiple system calls, library functions, and harness functions.

.SH Files changed
/usr/rep/src/reptilian-kernel/arch/x86/entry/syscalls/syscall_64.tbl
/usr/rep/src/reptilian-kernel/include/linux/syscalls.h
/usr/rep/src/reptilian-kernel/kernel/sys.c

.SH How I implemented each new system call
.B Step(1)
Added my syscalls to the "syscall_64.tbl" to act as the entry point for
the system
calls:
435   common     get_proc_log_level      __x64_sys_get_proc_log_level
436     common  set_proc_log_level      __x64_sys_set_proc_log_level
437     common  proc_log_message        __x64_sys_proc_log_message
.PP

.B Step(2)
To link the system calls properly, I add the asmlinkage prototypes to
"syscalls.h":
asmlinkage int sys_get_proc_log_level(void);
asmlinkage int sys_set_proc_log_level(int new_level);
asmlinkage int sys_proc_log_message(int level, char *message);
.PP

.B Step(3)
I implemented my sys calls at the end of sys.c, this part is the logic of
the system
calls such as new_level is input or if the message should be able to
print.
.PP

.B Step(4)
I created a process_log directory in which contains my files
process_log.c,
process_log.h, in addition I created a makefile that uses ".c" and ".h"
files to
create "process_log.o" and "libprocess_log.a".
.PP

.SH Testing
I tested the my process_log files by using the tests provided in the SRC
file of the
project.

.SH Problems
My program does not correctly print messages for the proc_log_message and
for the
harness test my virtual machine shuts down, which I was unable to figure
out why.
```

```
In addition, it was a little confusing on where to start and finding which
files to
place the place my system call information and changes. Once I was able to
figure
out where to place entry points and prototypes, I was able to write the
easily in
the process_log directory.

.SH Sources
https://www.kernel.org/doc/html/v4.10/process/adding-syscalls.html
https://brennan.io/2016/11/14/kernel-dev-
ep3/#:~:text=In%20fact%2C%20a%20system%20call,do%20something%20on%20your%2
0behalf.&text=In%20fact%2C%20the%20modern%20way,to%20make%20a%20system%20c
all.
https://dev.to/omergulen/how-to-add-system-call-syscall-that-prints-
elapsed-time-of-a-process-with-given-pid-to-the-kernel-and-test-it-335c

.SH Video link
https://youtu.be/B5c8YMsnYjI


.SH AUTHOR
Jesse Maki
```