CMPS 240 – Sprint 1
Team: *Cachers*
Members: *Abdul Rahman Kobeissi*
Stephane Najjar
Yara Khalifeh

OBJECTIVE:

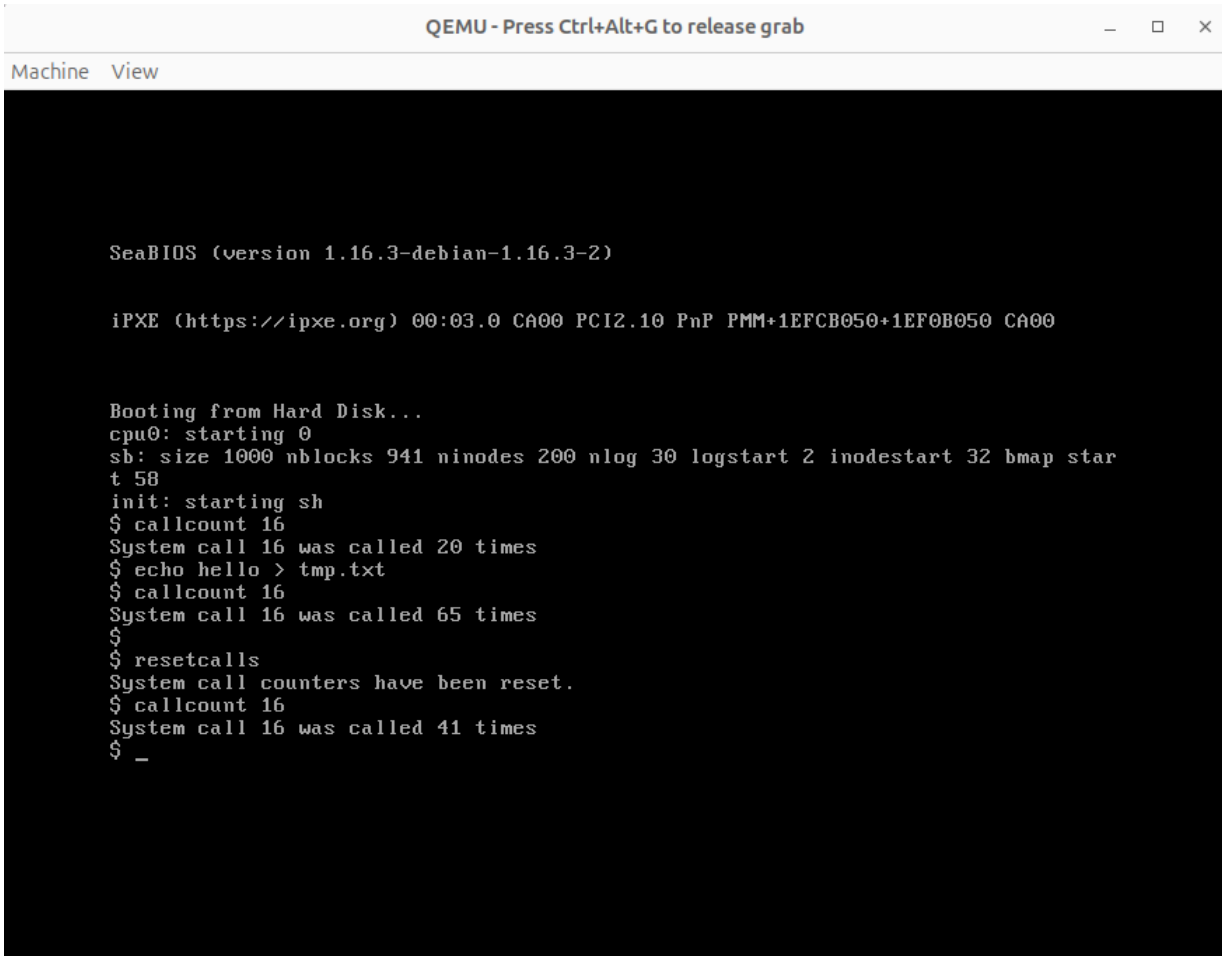
Our objective was to extend MIT xv6 with a new system call “syscallcount(int id)” that returns how many times a given system call has been invoked since boot.
A user program `callcount` demonstrates the feature.

We also added another system call resetcallcount(void) that resets the syscall counters to zero. A user program “resetcalls” shows that the counters are correctly cleared and start counting again from recent activity.

MODIFICATIONS:

FILES	CHANGES FOR SYSCALLCOUNT	CHANGES FOR RESETCALLCOUNT
Makefile	Added _callcount to UPROGS list	Added _resetcalls to UPROGS list
callcount.c	New file added, a user test program that prints syscall counter	
defs.h	Added “#define NSYSCALLS 24”, which defines the size of syscall counter array	
resetcalls.c		New file added, a user test program that calls resetcallcount() syscall
syscall.c	● Added “syscall_counter” array, which keeps track of how often each syscall	

	is called ● Incremented counter in dispatcher ● New “sys_syscallcount()” function, which returns how often each syscall is called ● Registered syscall in table	
syscall.h	Added “#define SYS_syscallcount 22”, which assigns syscall ID for `syscallcount`	Added “#define SYS_syscallcount 23”, which assigns syscall ID for `resetcallcount`
sysproc.c		Added `sys_resetcallcount()` function, which is a kernel implementation that resets syscall counters to zero
user.h	Added `int syscallcount(int id);`, a user-space prototype	Added `int resetcallcount(void);`, a user-space prototype
usys.S	Added `SYSCALL(syscallcount)`,	Added `SYSCALL(resetcallcount)`



Following the echo command, the “callcount 16” output has risen from 20 to 65, showing that our new syscallcount() system call correctly counts every extra write() call made in user space.

Following resetcalls command, the “callcount 16” output has dropped from 65 to 41, confirming that our reset functionality works and counters start over from recent system activity.