

# Project 1b: xv6 Intro

We'll be doing kernel hacking projects in **xv6**. Xv6 is a port of a classic version of unix to a modern processor, Intel's x86. It is a clean and beautiful little kernel, and thus a perfect object for our study and usage.

This first project is just a warmup, and thus relatively light on work. The goal of the project is simple: to add a system call to xv6. Your system call, **getprocs()**, simply returns how many processes exist in the system at the time of the call.

## Details

Your new syscall should look like this: **int getprocs(void)**

Your system call returns the number of processes that exist in the system at the time of the call.

## Tips

Find some other system call, like **getpid()** or any other simple call. Basically, copy it in all the ways you think are needed. Then modify it to do what you need.

Most of the time will be spent on understanding the code. There shouldn't be a whole lot of code added.

Using gdb (the debugger) may be helpful in understanding code, doing code traces, and is helpful for later projects too. Get familiar with this fine tool!

## The Code

The source code for xv6 (and associated README) can be found in **~cs537-1/ta/xv6/**. Everything you need to build and run and even debug the kernel is in there.

You may also find the following readings about xv6 useful, written by the same team that ported xv6 to x86: [xv6 book](#). However, note that the kernel version we use is a little different than the book.

**Particularly useful for this project: Chapters 0, 1, 2.**