Assignment 1

Network Security applications development

Dimitry Rakhlei | A00849367 | 2018-01-22

# Overview

The goal of this project was to determine if processes were faster at computational tasks than threads or vise-versa. The task was to create two applications which would execute the same tasks and to measure their performance. I chose to perform this experiment on the GNU/Linux platform and therefore my outcomes may not be the same as someone’s wrote code for Windows. My initial assumption as that processes will be much faster than threads because of how Linux handles their creation. While debugging I found that many of the threading libraries appear to function much like the multi-processing library I was using (OpenMP).

This lead me to writing my programs using only the fork statements for processes and the basic POSIX thread API for threads.

# Design

Both applications decompose 4 numbers into their prime factors. The parent thread handles some of the IO and synchronization. Each application performs one distinct operation on each process/thread. The applications have methods for outputting detailed logs and simple time-based logs for processing.

The process application begins the timing procedure right before the fork loop and allows each individual child process to report back their execution time.

The thread application begins the timing procedure after creating the thread but before any of the computational work.

I found the example prime decomposition code perfect for testing parallel workloads on the CPU. The computational difficulty of the problem can be scaled easily with the size of the input and