**CS 111 Design Project Lab 1**

**Design Document**

Spencer Tung 004355860

Li Tseng 304272081

**Design Overview**

> Security Objective

What we want to do in this design project is to prevent a process overload attack when using our shell in lab 1ab. When there is an overload attack, out shell may be overloaded with too many bad processes so that our shell can not have processes to be used by other users. Our implementation will try to mitigate this kind of attack and not let them happen if one process tries to fork too many processes.

> Overall Design

First, we aim to limit the maximum number of processes that our shell can fork at any given time. If a process hits the limit, we will kill it if it continues to attempt to fork.

Second, we want to ensure fairness in other programs by checking the number of children that a parent has forked using *pstree*. When the number of processes exceed a certain limit, we will start increasing its nice factor to slowdown its forking.

**Plan for Implementation**

**Summary of Results**

> Robustness Analysis

First, by limiting the maximum number of processes on the shell, we can prevent our shell from being overloaded since it will stop forking when it already has a certain number of processes.

Second, by increasing the nice factor when a process trying to fork too many child processes, we can let it fork slower. Hence, it won’t occupy the shared resources on the shell, which may let other users cannot use the resources.