

COEN 177: Operating Systems

Lab assignment 7: Minix Scheduling

Objectives

1. To understand Minix scheduler implementation.
2. To modify the Minix scheduler and observe changes.

Guidelines

In this lab, you will continue to work on the Minix operating system. You will need to install, run, and rebuild Minix, this time with changes to the scheduling algorithm. Minix uses a multi-level priority queue scheduler, and modification of the scheduling algorithm, particularly severe changes to high priority queues, may make the system unstable. You may need to experiment with what level of change allows the system to continue to run while still being noticeable.

Getting started with Minix on the ECC Systems

For consistency, it is strongly recommended that you run the provided Minix image on the ECC systems. If you have not done so already, follow the instructions from lab 6 to install and run Minix on the ECC Systems.

The Minix Scheduler

The goal of this assignment is to gain experience in modifying an operating system kernel, specifically the process scheduling algorithm. You are to modify the queue selection algorithm to skew the priority scheduling. Currently, the algorithm selects the highest priority queue with jobs available every time it runs, so your task is to make it sometimes choose to ignore an available job queue and run some lower priority queue instead. Note that if you modify the priority queue imprudently, the operating system will cease to function. Your goal is to achieve the following:

- upon attempting to select the next job queue, modify the selection to add a random possibility of choosing to move on to a lower priority job queue instead (the lower the probability you choose for this, the more consistent with the current selection mechanism you will be. Some experimentation may be required to select a reasonable value).

Hint: the name of the function called to **pick** the process to run next is one of the few instances where that word appears, and the grep utility - or, alternately, the Minix documentation - will be very useful for finding it.

You will be graded based on both your implementation of this modification, and on how well you explain the mechanism. It is therefore important to realize that this is both an exercise in coding, as well as an exercise in familiarizing yourself with, and understanding to the point of being able to explain, an unfamiliar code base.

Observing a change

A second challenge in this assignment is to demonstrate how the scheduler has been modified. You are to note an observable change in behavior between the unmodified and modified schedulers. To achieve this, you may need to run a simple test program (a shell command with a lot of output, or one that runs relatively slowly already, may be good candidates depending on the probability chosen).

Additional Resources:

- Minix Wiki: <https://en.wikipedia.org/wiki/MINIX>
- Minix user guide: <https://wiki.minix3.org/doku.php?id=usersguide:start>
- Minix installation guide: <https://wiki.minix3.org/doku.php?id=usersguide:doinginstallation>

Requirements to complete the lab

1. Show the TA your running Minix system.
2. Write up a description of your steps, which files you modified and what the modifications were (a *short* code snippet may be useful), and what changes you observed afterwards.

Please start the text file with a descriptive block that includes at minimum the following information:

//Name:

//Date:

```
//Title: Lab7 -  
//Description:
```