Stephen Tambussi
COEN177L - Thursday 5:15PM
Lab 7

Guide to Modifying the Minix Scheduler

Prerequisite: must have vmware on ECC linux systems and access to these systems (NoMachine).

Virtual Machine Setup

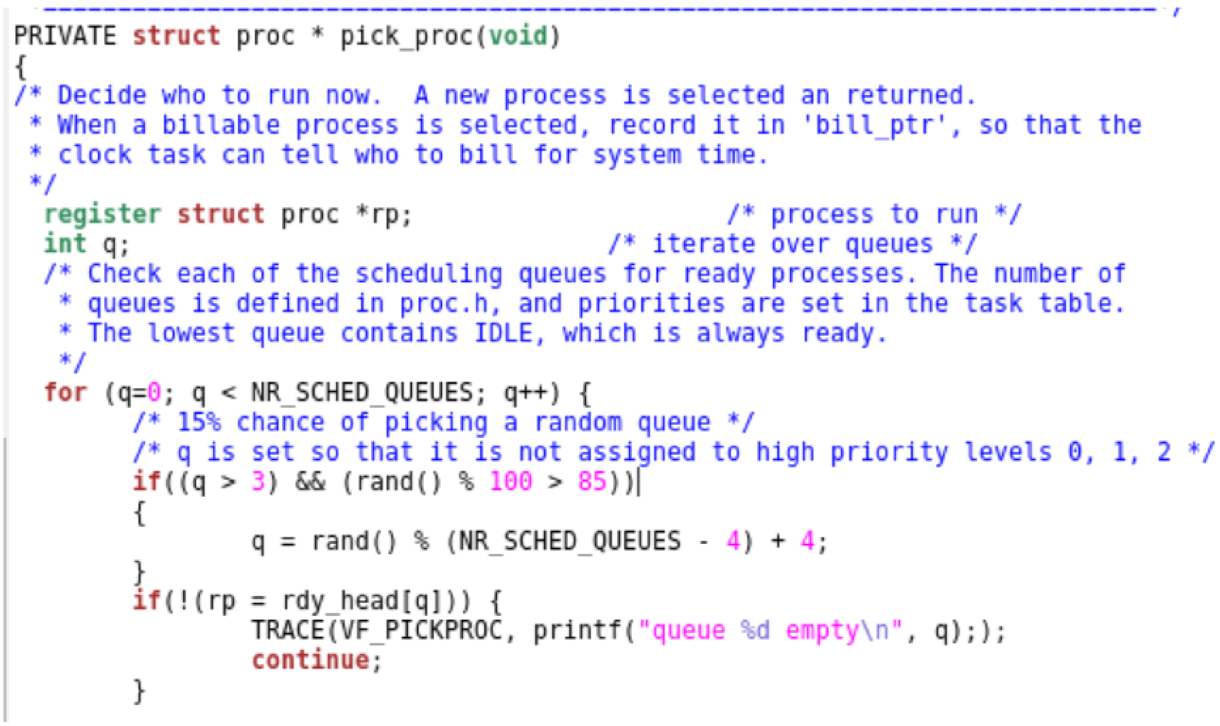
1. Open up a terminal window on the ECC linux lab computer.
2. At the prompt, type in: **setup vmware** to prepare the vmware software.
   a. This only needs to be done the first time you login to your machine each time.
3. Next, type in: **minix-get-image** to get a copy of the minix system image.
   a. This will completely erase any pre-existing old images (use this to restore the system if you break it).
4. Next, type in: **vmware &** to start the vmware software with the minix system image.
5. Once vmware starts, select the minix image and "power" on the virtual machine.
6. After the minix system boots and the command line becomes available, enter **root** as the username when prompted.
7. Next, type the command **passwd** and then enter your selected passcode when prompted.

FTP Setup

1. In the minix system, type in: **tcpd ftp /usr/bin/in.tfpd &** to launch the FTP daemon which will allow the local system to connect to the virtual machine system.
2. Next, type in: **ifconfig** to get the IP address of the VM.
3. Open up a terminal window on your local machine (if it wasn't already open) and using the VM's IP address, enter the command: **ftp <VM's IP address>** to launch ftp on the local machine.
4. When prompted, enter your username (*root*) and password for the VM.

Modifying Scheduler Source Code

1. On the local machine terminal window in ftp, go to the directory where the VM's kernel source code is by entering the command: **cd /usr/src/kernel**.
2. Once in that directory, locate the file titled **proc.c**.
3. Make sure that on your local machine you are in the directory that you want the files transferred to by entering: **lcd** to check the current location.
   a. Enter **lcd <directory path>** to change location on the local machine.
4. Enter: **get proc.c** to transfer the file from the VM to the local machine.
5. On the local machine, open **proc.c** with the text editor of your choice and locate the private function *pick_proc*.



```
  -------------------------------------------------------------------------*/
PRIVATE struct proc * pick_proc(void)
{
/* Decide who to run now.  A new process is selected an returned.
 * When a billable process is selected, record it in 'bill_ptr', so that the
 * clock task can tell who to bill for system time.
 */
  register struct proc *rp;                       /* process to run */
  int q;                                  /* iterate over queues */
  /* Check each of the scheduling queues for ready processes. The number of
   * queues is defined in proc.h, and priorities are set in the task table.
   * The lowest queue contains IDLE, which is always ready.
   */
  for (q=0; q < NR_SCHED_QUEUES; q++) {
        /* 15% chance of picking a random queue */
        /* q is set so that it is not assigned to high priority levels 0, 1, 2 */
        if((q > 3) && (rand() % 100 > 85))
        {
                q = rand() % (NR_SCHED_QUEUES - 4) + 4;
        }
        if(!(rp = rdy_head[q])) {
                TRACE(VF_PICKPROC, printf("queue %d empty\n", q););
                continue;
        }
```

6. As shown in the above screenshot, add the code to the *pick_proc* function.
7. Next, back on the local machine terminal with ftp, type in **put proc.c** to transfer the modified file back to the VM in the directory */usr/src/kernel*.
    a. This will overwrite the proc.c file that was previously in that directory.
8. Return to the minix system on the VM and navigate to the **/usr/src** directory. Type in **make world** to rebuild minix with your changes.
9. Once this process is done, enter **reboot** to restart the minix system.
    a. If the system crashes, restart this guide at step 3 in the “Virtual Machine Setup” section.
10. If nothing went wrong, then you should see the system booting up much more slowly than before.