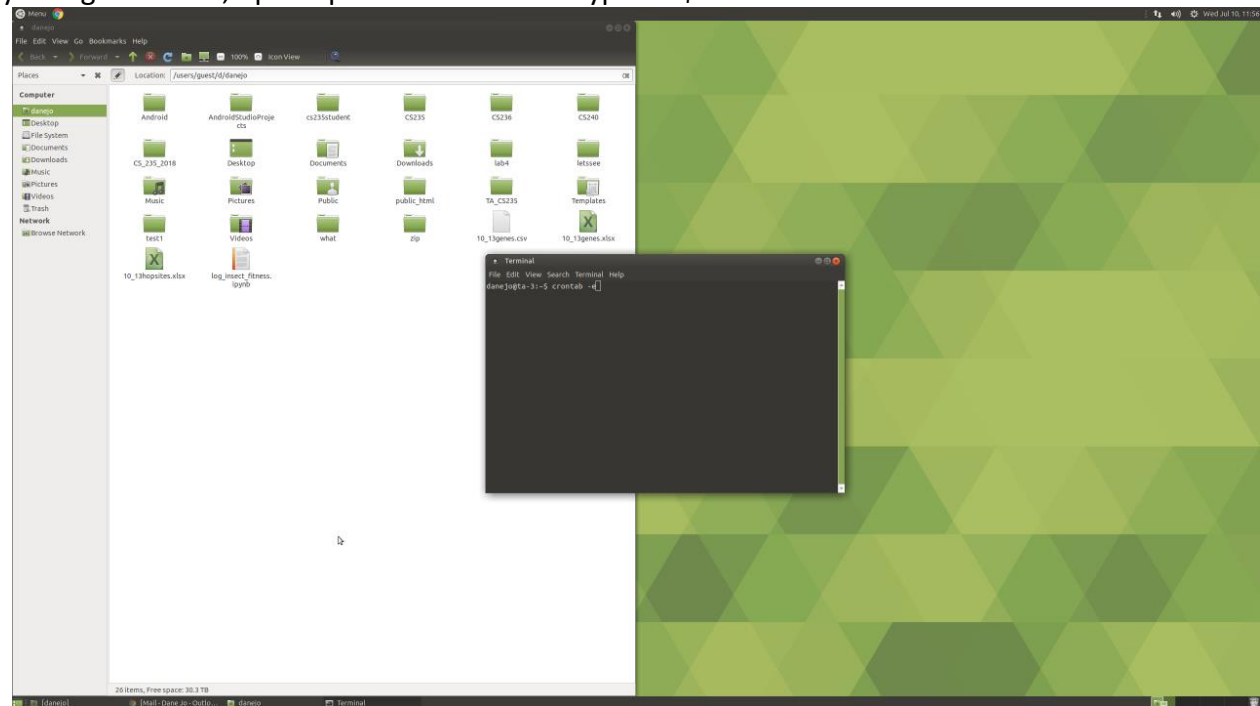


## Introduction


In this tutorial, I will show you how to run the driver on any TA machine. (I will use TA-1 through the tutorial.)

## Starting up The Driver

1. When you log into TA-1, open up the terminal and type in: `$ crontab -e`



- a.
2. When you click 'enter', you'll be greeted to a text editor (that uses vim) inside the terminal.



The screenshot shows a terminal window with the following content:

```

# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for "any").#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -czf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# h   dom mon dow   command

```

The terminal window has a title bar with "Terminal" and a menu bar with "File", "Edit", "View", "Search", and "Terminal Help". The status bar at the bottom shows the file path "/tmp/crontab.G3400g/crontab" with line 23 and column 889. It also includes icons for "Save Screenshot" and "Mail - Dane Ju - Outlook...".

```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# mail to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 5 * * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# h don mon dow command
# * * * * /users/groups/cs235ta/submission_driver/runCompileDriver.sh && /users/groups/cs235ta/submission_driver/cronlog.txt
```

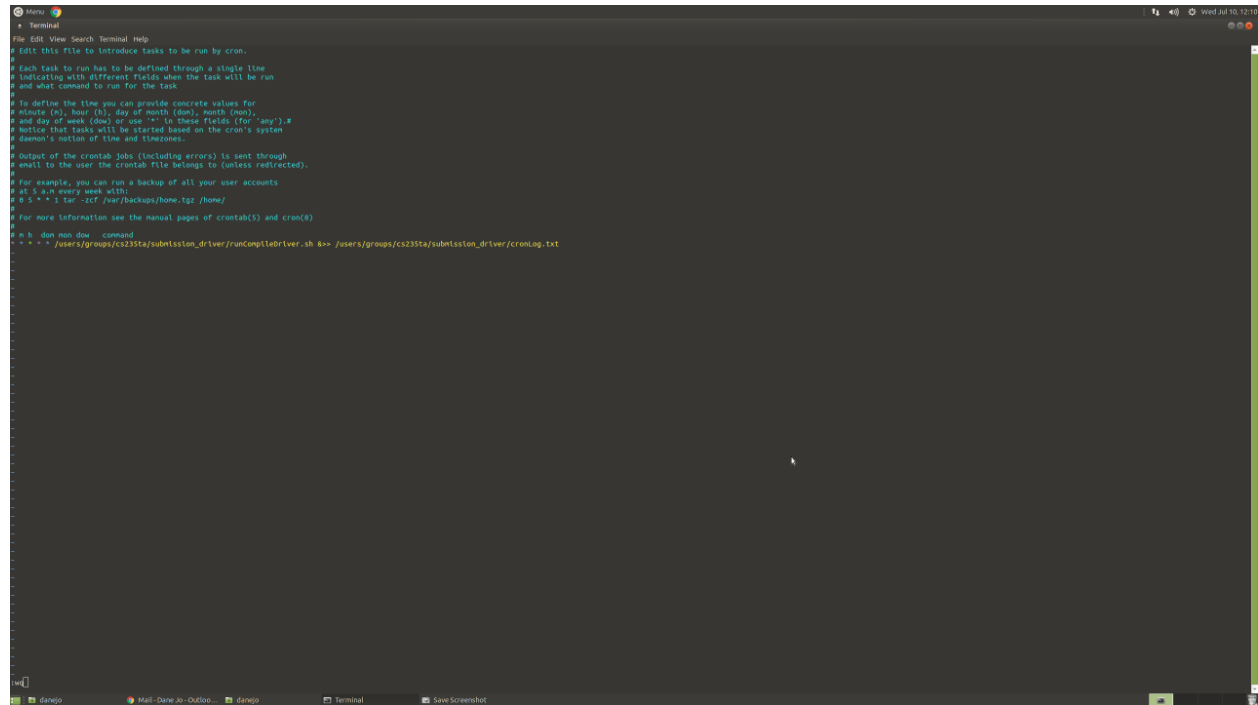
a.

- To save your edit, click the 'Esc' key on your keyboard and then click the 'Shift' + ';' keys to get the ':'. When you do this, you'll see at the bottom left corner a ':' with a flickering box waiting for you input.

```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# mail to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 5 * * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# h don mon dow command
# * * * * /users/groups/cs235ta/submission_driver/runCompileDriver.sh && /users/groups/cs235ta/submission_driver/cronlog.txt
```

a.

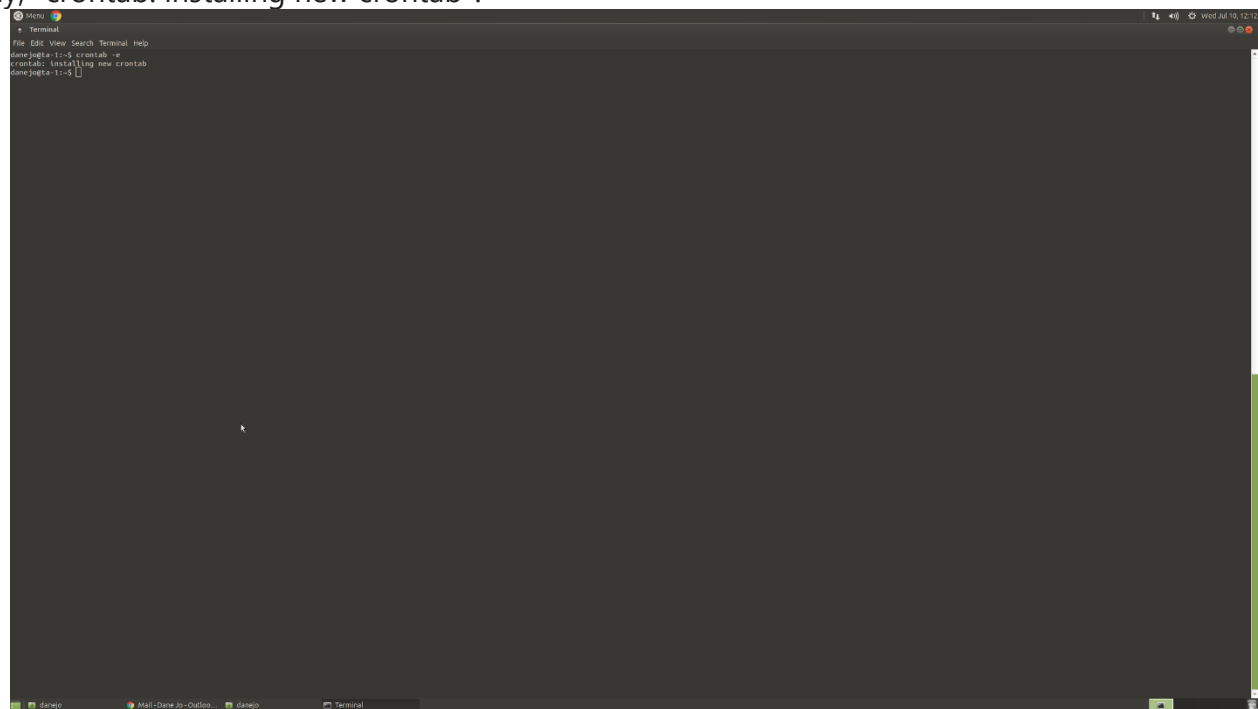
- Next, type 'wq' (below is the picture of you typing 'wq') and press enter.



```
crontab(5)
# This file is to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# mail to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 5 * * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# h don run dow command
# * * * * * /users/groups/cs235ta/submission_driver/runCompileDriver.sh &> /users/groups/cs235ta/submission_driver/cronLog.txt
```

a.

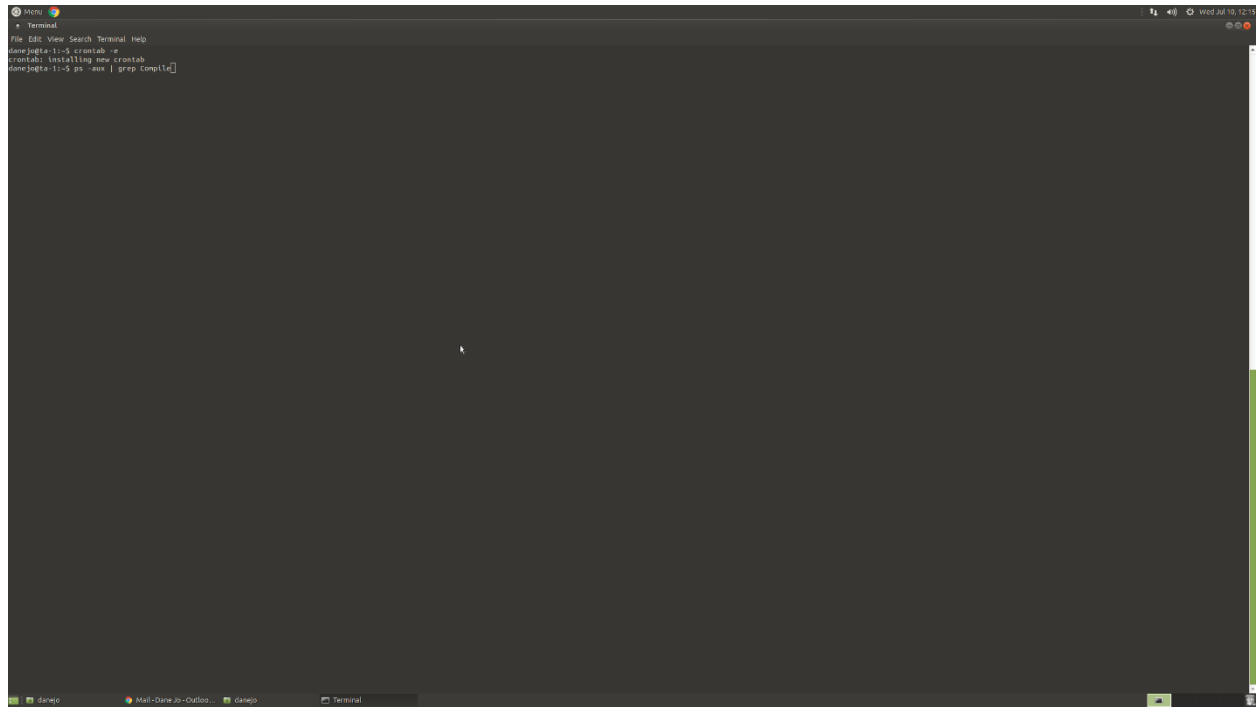
- By doing the previous step, you'll go back to the terminal and in the terminal, it will way say, "crontab: installing new crontab".



```
danjpa-11:~$ crontab -e
crontab: installing new crontab
danjpa-11:~$
```

a.

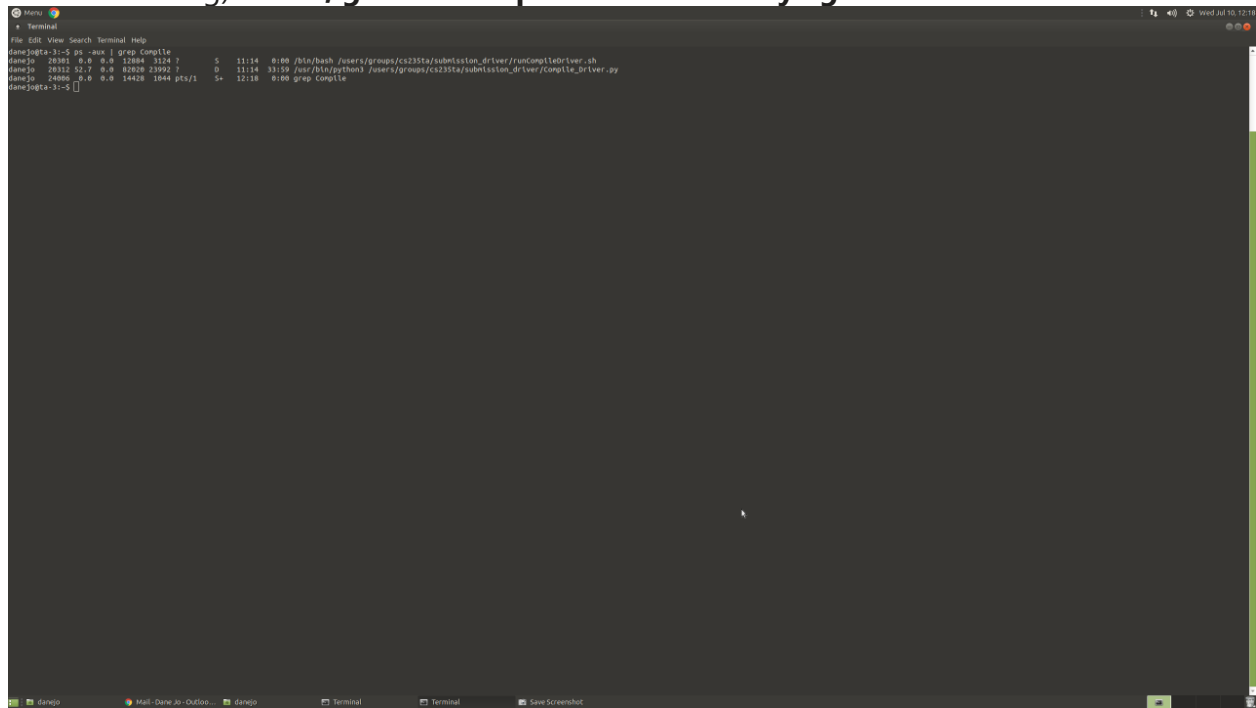
- To check if the driver is running, type in terminal: `$ ps -aux | grep Compile`



```
Menu
Terminal
File Edit View Search Terminal Help
danej@ta-1i-5:~$ crontab -e
crontab: installing new crontab
danej@ta-1i-5:~$ ps -aux | grep Compile
```

a.

9. If you see **3 lines of tasks**, that means **the driver is running** (below is a picture of the driver now running). **If not, give the script a minute and try again.**



```
Menu
Terminal
File Edit View Search Terminal Help
danej@ta-3i-5:~$ ps -aux | grep Compile
danej 26081 0.0 0.0 12884 3124 ? S 11:14 0:00 /bin/bash /users/groups/cs235ta/submission_driver/runCompiledDriver.sh
danej 26312 52.0 0.0 82808 23992 ? D 11:14 33:59 /usr/bin/python3 /users/groups/cs235ta/submission_driver/Compile_driver.py
danej 24866 0.0 0.0 14408 1644 pts/1 S+ 12:18 0:00 grep Compile
danej@ta-3i-5:~$
```

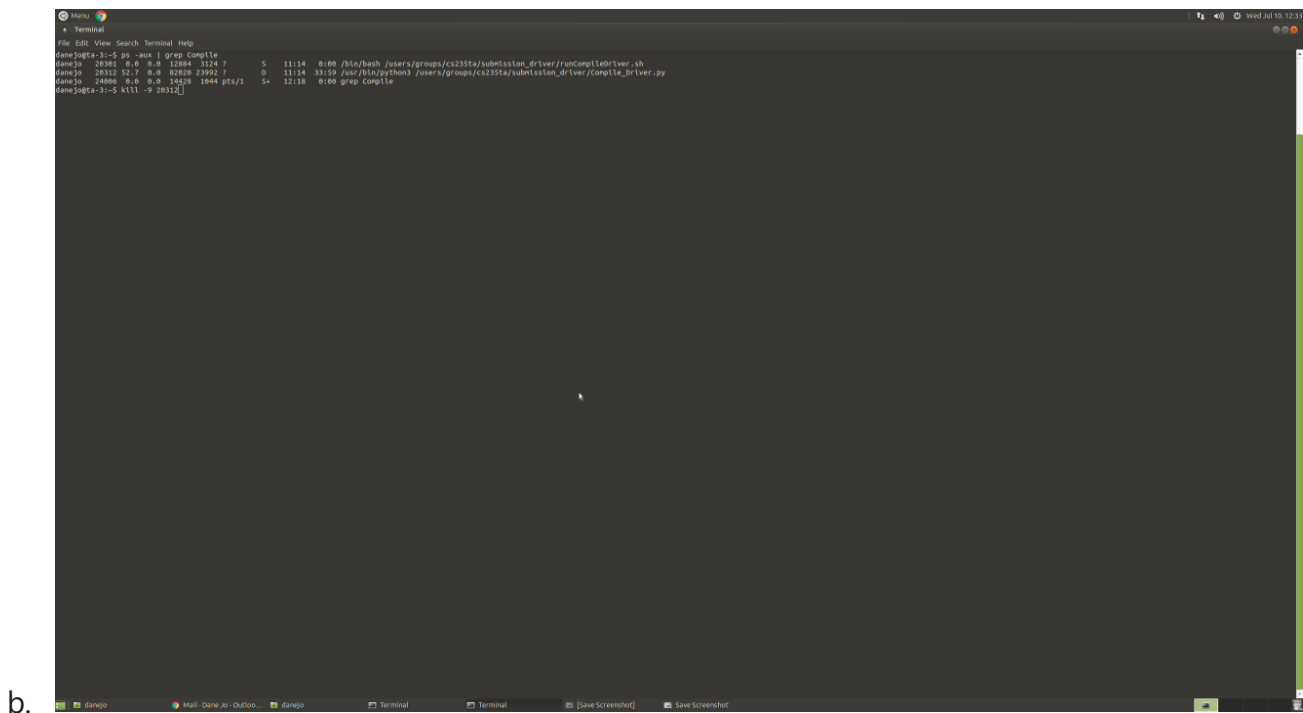
a.

10. Once you see your job running, congratulations; you got the driver up and running!

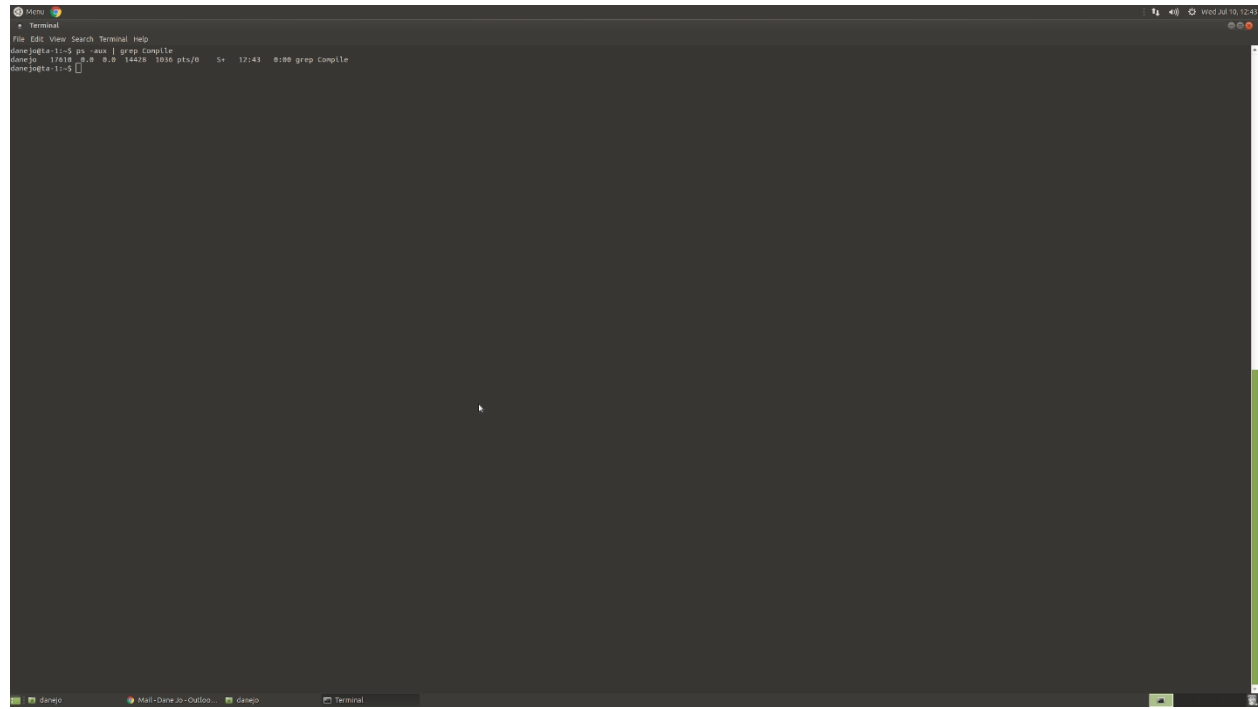
## How to Kill the Driver Manually

1. To kill the driver, first, access the computer's terminal where the driver is running.
2. Access the crontab by following step #1 again.

3. When you enter the text editor, enable editing and delete the command that is running the driver (see the command that we wrote on step #4).
4. Save your changes by following step #5 and #6.
5. Next, to actually kill the driver, type in the same command as step #8.
6. Again, you'll see all the current task running on the computer. **You'll only need to kill 1 task.** Notice that each task will have a netid and a unique task id number next to it. To kill the task (or in this case, the driver), type in the terminal: \$ kill -9 \_\_\_\_\_type\_in\_the\_unique\_task\_id\_number\_\_\_\_\_
  - a. For example; in the picture from step #9, we want to kill the task running '/usr/bin/python3 /users/groups/cs235ta/submission\_driver/Compile\_Driver.py'. So in the terminal, I would type, \$ kill -9 20312



7. After you press enter, type in the same command as step #8.
8. You'll notice now that there will only be one task running (or was running. That was your last command--the 'ps -aux' one).



A terminal window with a dark background. The title bar at the top says "Terminal". The menu bar includes "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal content shows the following text:

```
dan@jgta:11~$ ps -aux | grep Compile
dan@jgta 17618  0.0  0.0 14428 1036 pts/0    S+   12:43   0:00 grep Compile
dan@jgta:11~$
```

The bottom of the window shows a dock with icons for "dan@jgta", "Mail - dan@jgta - Outlook", "dan@jgta", and "Terminal". The system status bar at the very bottom indicates "Wed Jul 10, 12:43".

- a.
9. Congratulations! You have officially killed the driver.