**Setting up the instance**

I spun up three Amazon Linux 2 AMI (HVM), SSD Volume Type EC2 instances in my personal AWS account, all associated to the same private key file (aws\_key.pem) that resides on my hard drive. In order to share access with the team, I will have to share this key file. I don’t think this is technically best practice (for security reasons) but it should work for our purposes. All of the default free tier settings were maintained. We are not currently being charged for any AWS services, although I did set up automatic emails to alert us if we were moving beyond the limits of the free tier.

**Accessing the instances from the terminal**

First, cd to the location of the key file.

* Boss: ssh -i aws\_key.pem ec2-user@ec2-52-14-142-88.us-east-2.compute.amazonaws.com
* Minion 1: ssh -i aws\_key.pem ec2-user@ec2-3-22-168-13.us-east-2.compute.amazonaws.com
* Minion 2: ssh -i aws\_key.pem ec2-user@ec2-18-221-18-205.us-east-2.compute.amazonaws.com

**Installing software onto the instances**

The instances come basically empty, so I ran the following commands to get everything we need:

* Install python3: sudo amazon-linux-extras install python3
* Reset the default python version: sudo update-alternatives --install /usr/bin/python python /usr/bin/python3.6 10
* Install pip:
  + cd /tmp
  + curl -O <https://bootstrap.pypa.io/get-pip.py>
  + python3 get-pip.py --user
* Install python packages:
  + pip install bottle
  + pip install pandas
  + pip install bs4
  + pip install requests

**Code changes**

There are a few small but key changes that need to be made in the code in order for it to run in the distributed system.

* Boss script:
  + Comment out host = localhost. Instead, set host = '172.31.47.154'. This is the private IP address of the boss node
  + Comment out review\_port, book\_port = 8080, 80. Instead, set review\_port, book\_port = 6000, 7000
* Minion script:
  + Comment out host = localhost. Instead, set host = '52.14.142.88'. This is the public IP address of the boss node
  + Comment out review\_port, book\_port = 8080, 80. Instead, set review\_port, book\_port = 6000, 7000

**Transferring files to instances**

I had the master branch of the git repository on my Desktop, and copied it to the home directory each of the instances. This only needs to be done if there are changes to the code. Run this command from your local:

* Boss: scp -r -i /file/path/to/aws\_key.pem /local/file/path/to/directory/Reading-Trends-master/distributed\_data\_collection ec2-user@ec2-52-14-142-88.us-east-2.compute.amazonaws.com:/home/ec2-user
* Minion 1: scp -r -i /file/path/to/aws\_key.pem /local/file/path/to/directory/Reading-Trends-master/distributed\_data\_collection ec2-user@ec2-3-22-168-13.us-east-2.compute.amazonaws.com:/home/ec2-user
* Minion 2: scp -r -i /file/path/to/aws\_key.pem /local/file/path/to/directory/Reading-Trends-master/distributed\_data\_collection ec2-user@ec2-18-221-18-205.us-east-2.compute.amazonaws.com:/home/ec2-user

**Setting up security rules**

In order for the instances to be able to connect to each other, it’s necessary to set up security rules around the boss instance. This only needs to be set up once and is already done.

* Navigate to instances in the AWS console
* Click on Instance ID i-095f53f41f9c73b76
* Click Security
* Under Security Groups, click on sg-070e8f9c542c27fb0 - launch-wizard-1
* Click Add Rule
* Under Type select “All Traffic” and under Source select “Anywhere”

This is probably not advisable from a security standpoint and will be fine tuned in the future.

**Running the code**

* SSH into all 3 instances
* cd distributed\_data\_collection
* Run python boss\_script.py on the boss node. Wait for everything to get up and running (i.e. make sure it’s listening on both ports)
* Run python minion\_script.py on the minion nodes
* The data is updated in the CSVs on the boss node