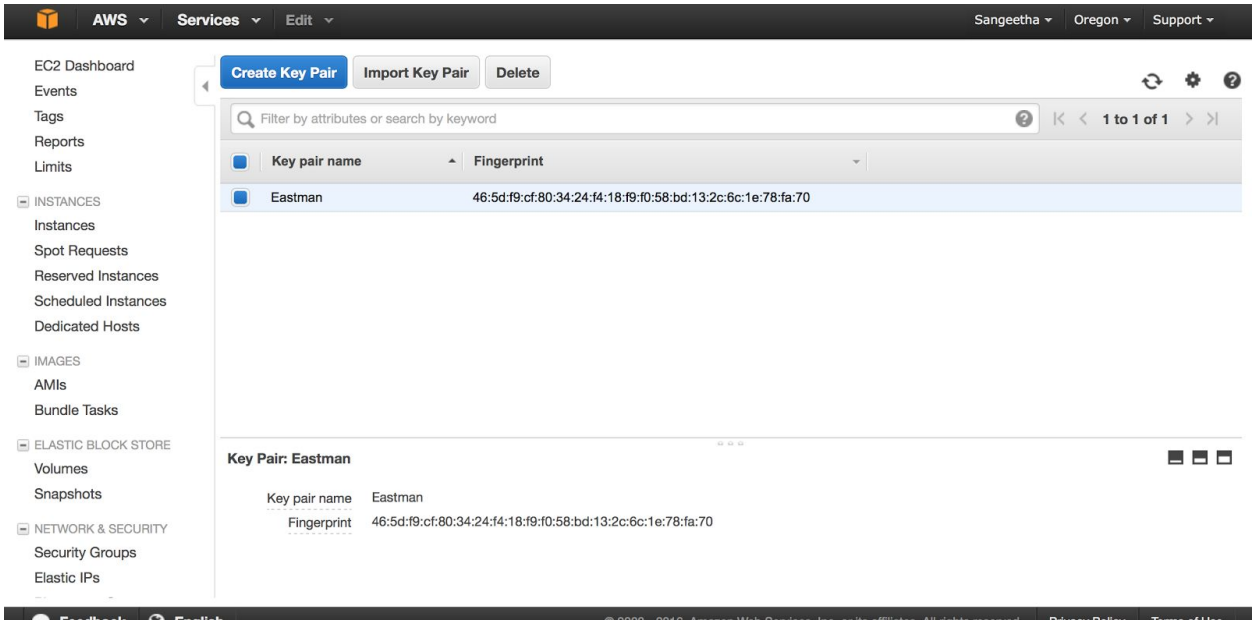
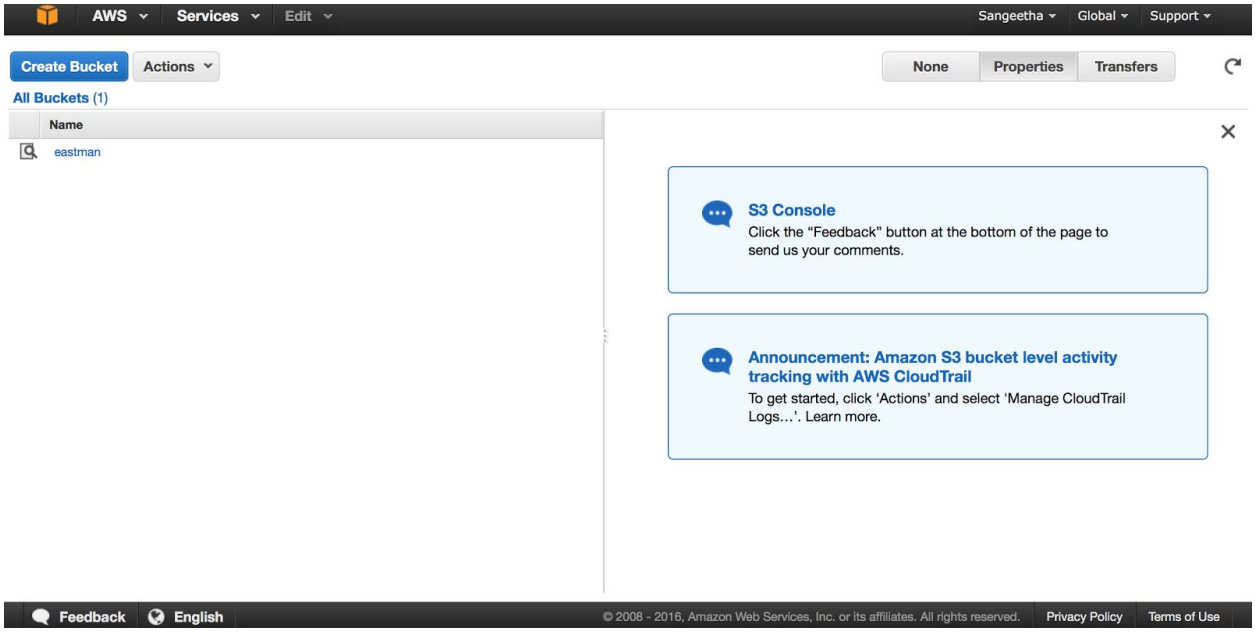
Setting up Spark on AWS

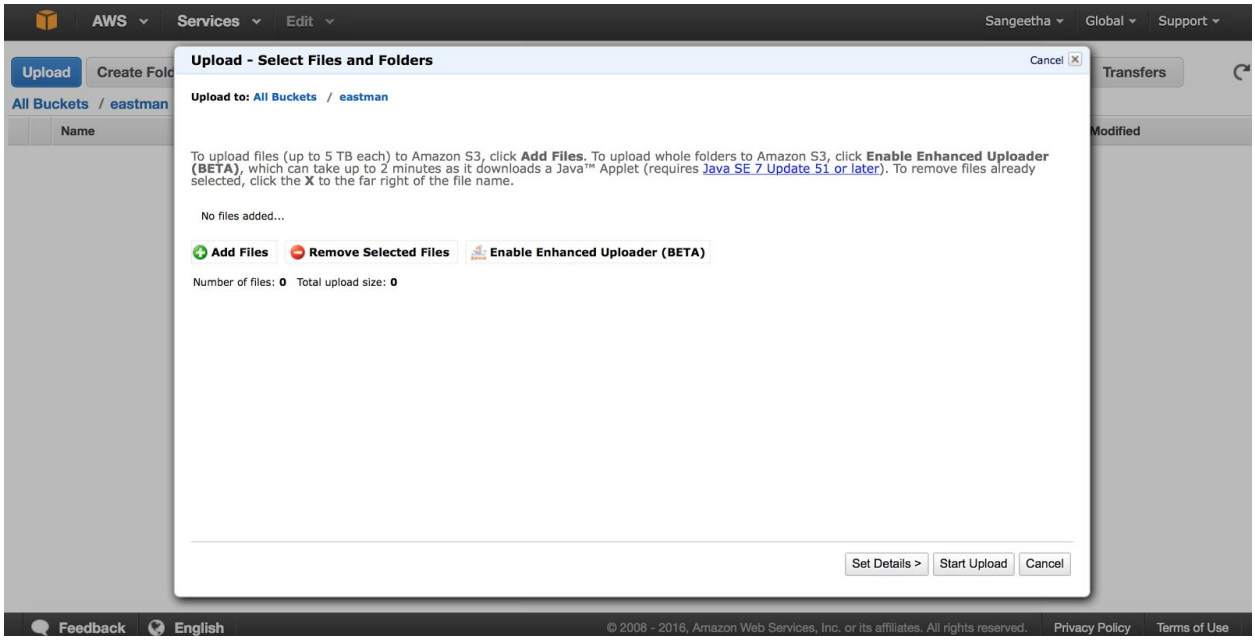
Below are the steps for setting up Spark on AWS:

1. Go to EC2 console. Choose Key Pairs and Click on Create Key Pair. Save the .pem file in your home directory.



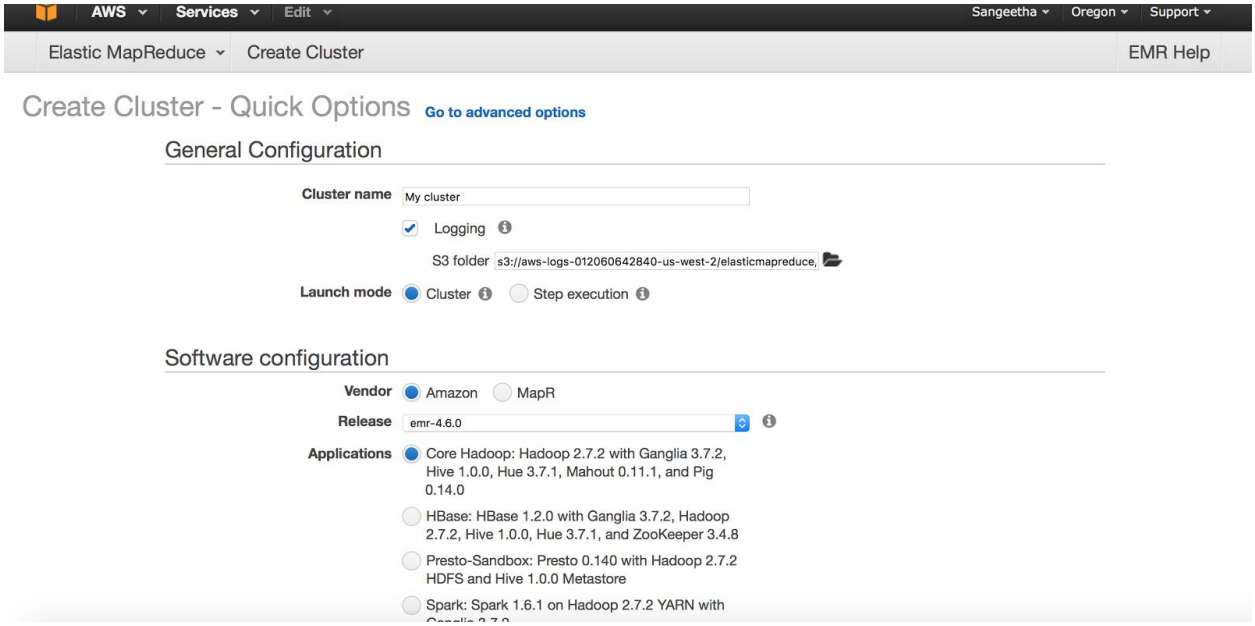
2. You can use either S3 or HDFS to store your files. HDFS is local to the instance that you get and will thus require you to upload the data files each time. S3 on the other hand stores files till you delete them and can be accessed from any cluster.
    a. S3 has a simple interface using which you can upload your data files. Go to S3. Create a Bucket and upload your files for easy access. They can be accessed as s3://<bucket-name>.



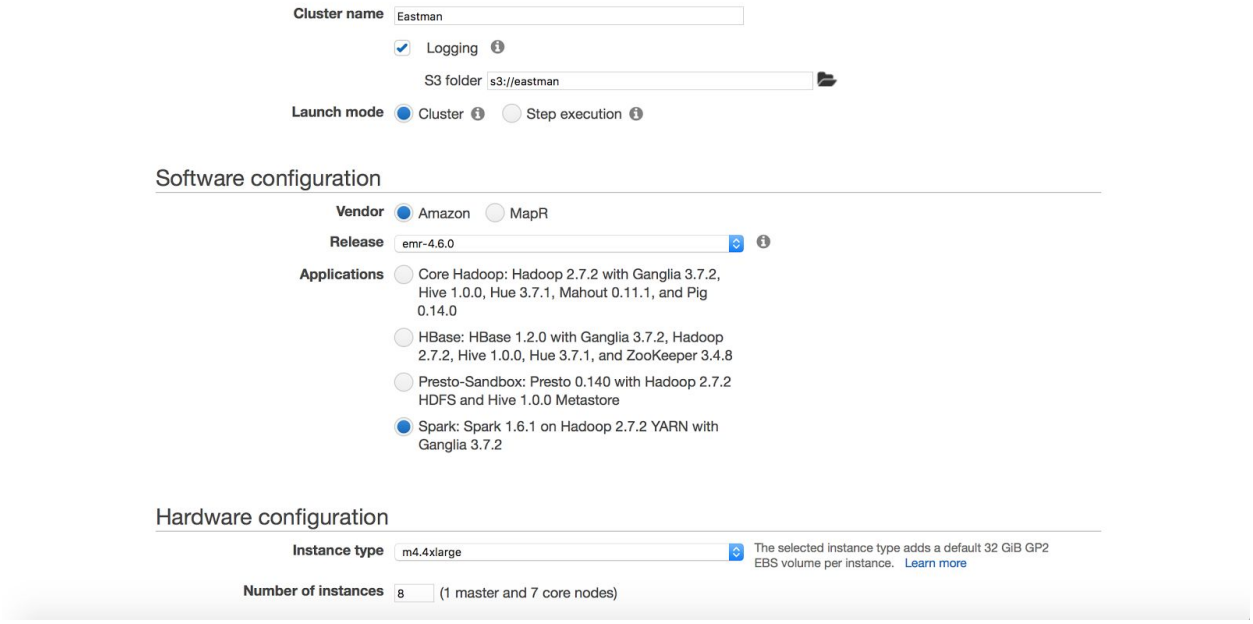


   b. For HDFS, once the cluster has been created, you can execute **hadoop fs -copyFromLocal filename <path-on-hdfs>.**

3. In order to create a cluster, go to EMR, click on Create Cluster.

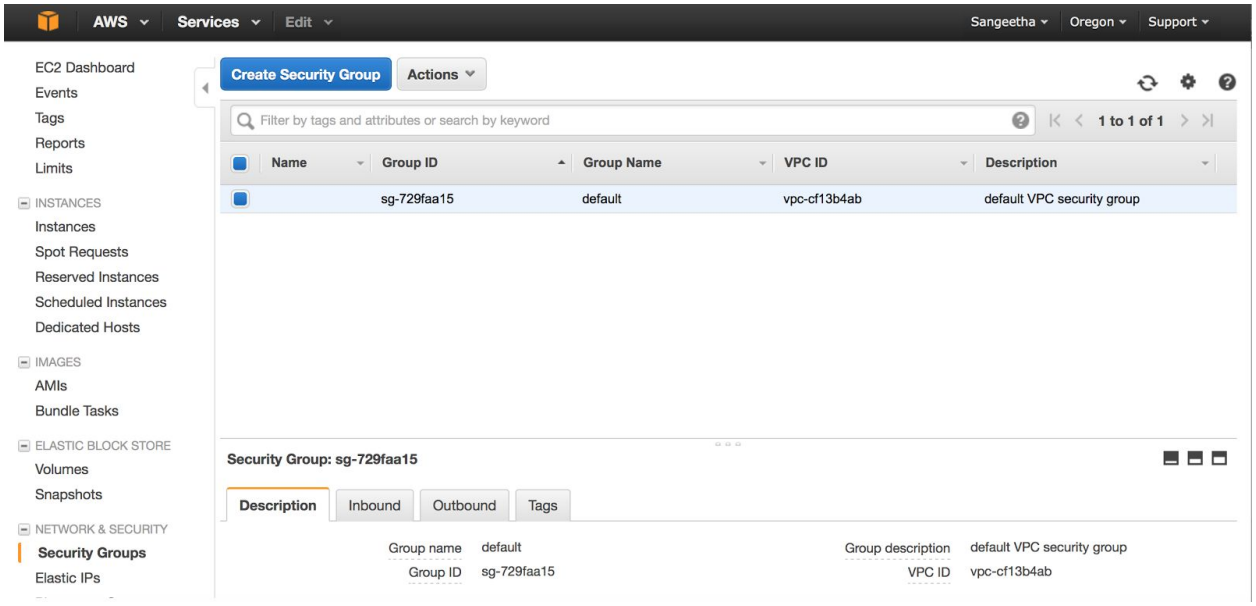


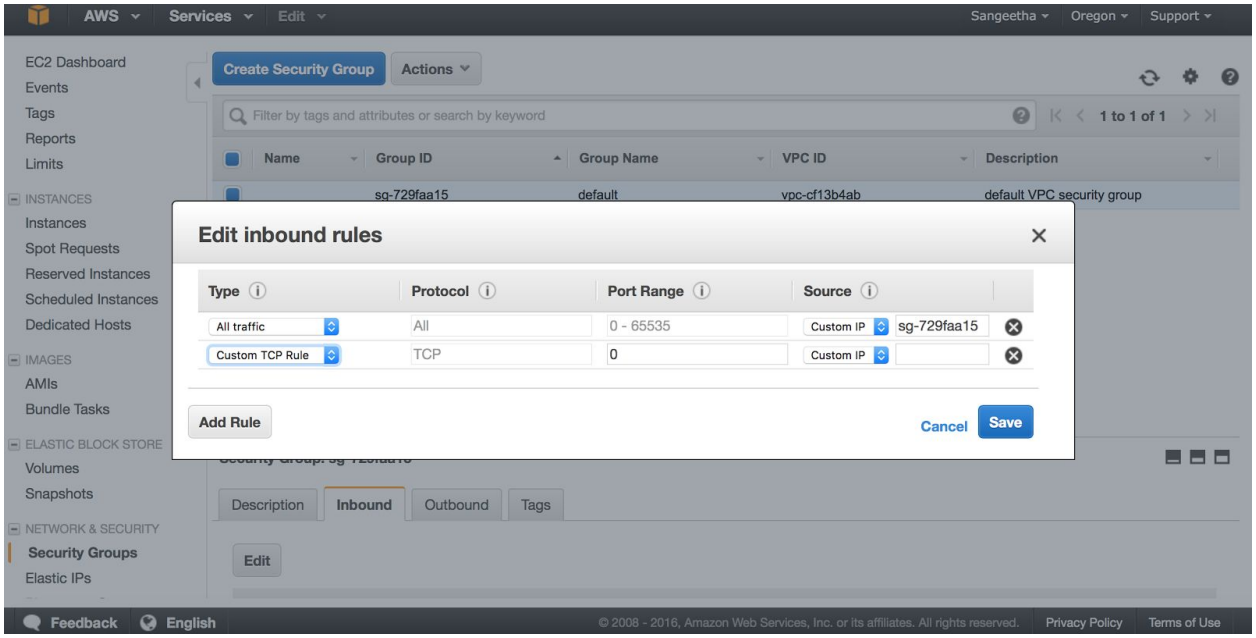
a. Choose a name for your cluster.



b. Choose Spark 1.6.1.
c. Choose your instance type and then select the total number of instances you need. There will always be atleast one master and one worker at all times. If you delete the worker the cluster gets terminated.
d. Next choose the EC2 pair that you just created.
e. Click on Create Cluster.
f. You can choose Advanced options if you would like to configure some specific settings or run steps on startup. https://docs.aws.amazon.com/ElasticMapReduce/latest/ReleaseGuide/emr-spark.html

4. You may need to modify the security groups that are created by default in order to be able to login and launch jobs. To do this go to EC2 dashboard. Choose Security Groups. Click on Inbound and click Edit. Configure to allow your traffic from your IP.





5. Once the cluster is up and running, ssh into the master node using the link that is shown from the EMR cluster page.
6. After logging in, you may want to install git to pull the source code.
    a. Sudo yum install git
7. After this you can submit spark jobs by just using spark-submit.
    a. `spark-submit --executor-cores 16 --executor-memory 20G --driver-cores 16 Narrow/2008/correlations-narrow.py`
8. You can enable spark history server by installing FoxyProxy on Chrome or Mozilla.

a. http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-connect-master-node-proxy.html
b. Once this is done you can access spark history server, Hadoop Resource Manager, Hadoop Name Node and Ganglia.
c. Ganglia is a nice visualization tool with which you can monitor the cluster. You can check the load of the cluster at different points in time as well as get information about CPU, Memory and IO usage.