Aims

This exercise aims to get you to practice:

- AWS EC2
- AWS S3
- Hadoop MapReduce on AWS EMR

Background

AWS EC2:

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud. Using Amazon EC2 eliminates your need to invest in hardware up front, so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. See more documentation at: http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html.

AWS S3:

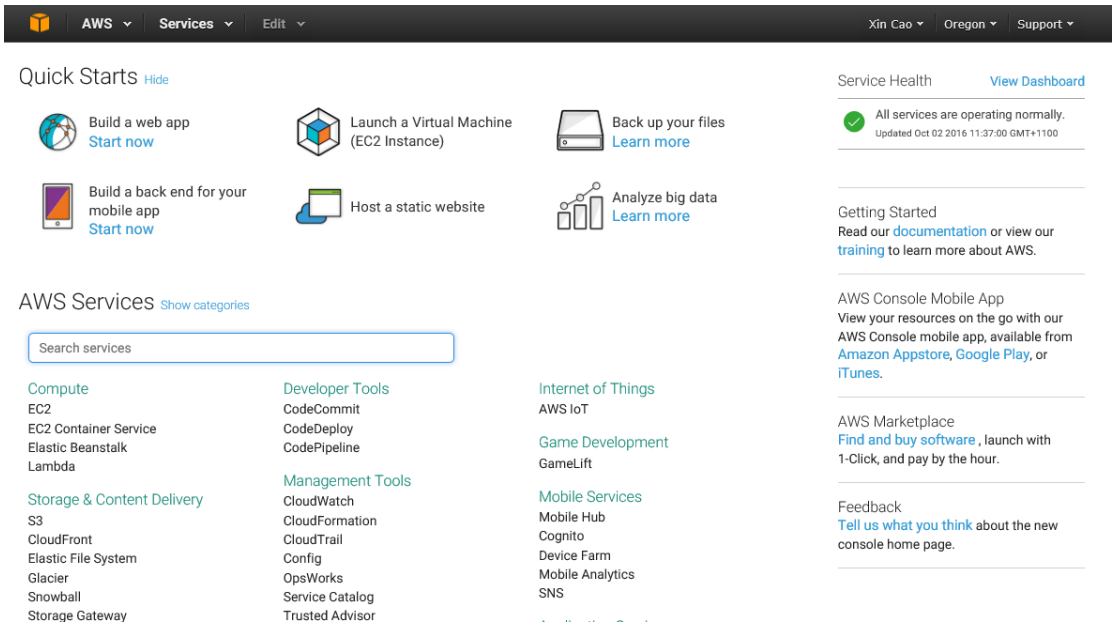
Amazon Simple Storage Service (Amazon S3) is storage for the Internet. You can use Amazon S3 to store and retrieve any amount of data at any time, from anywhere on the web. You can accomplish these tasks using the AWS Management Console, which is a simple and intuitive web interface. See more documentation at: http://docs.aws.amazon.com/AmazonS3/latest/gsg/GetStartedWithS3.html

AWS EMR:

Amazon EMR is a web service that makes it easy to quickly and cost-effectively process vast amounts of data. Amazon EMR simplifies big data processing, providing a managed Hadoop framework that makes it easy, fast, and cost-effective for you to distribute and process vast amounts of your data across dynamically scalable Amazon EC2 instances. You can also run other popular distributed frameworks such as Apache Spark in Amazon EMR, and interact with data in other AWS data stores such as Amazon S3. See more documentation at: http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-what-is-emr.html

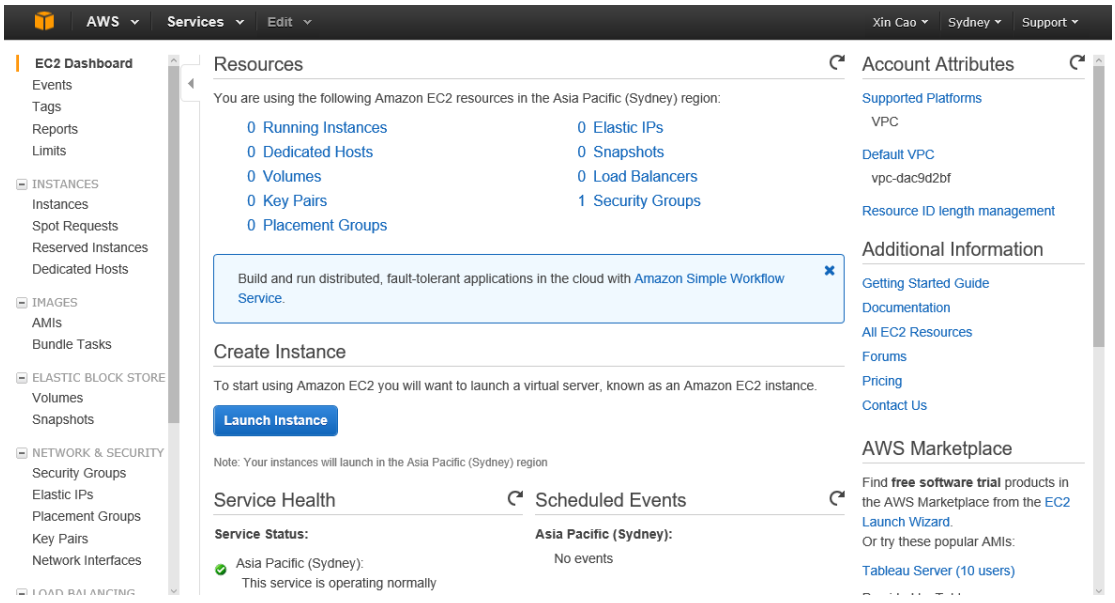
Try AWS EC2 Using Free Tier Accounts

1. Log in AWS using your own account. Once you have signed in, you will be greeted by a page like this:

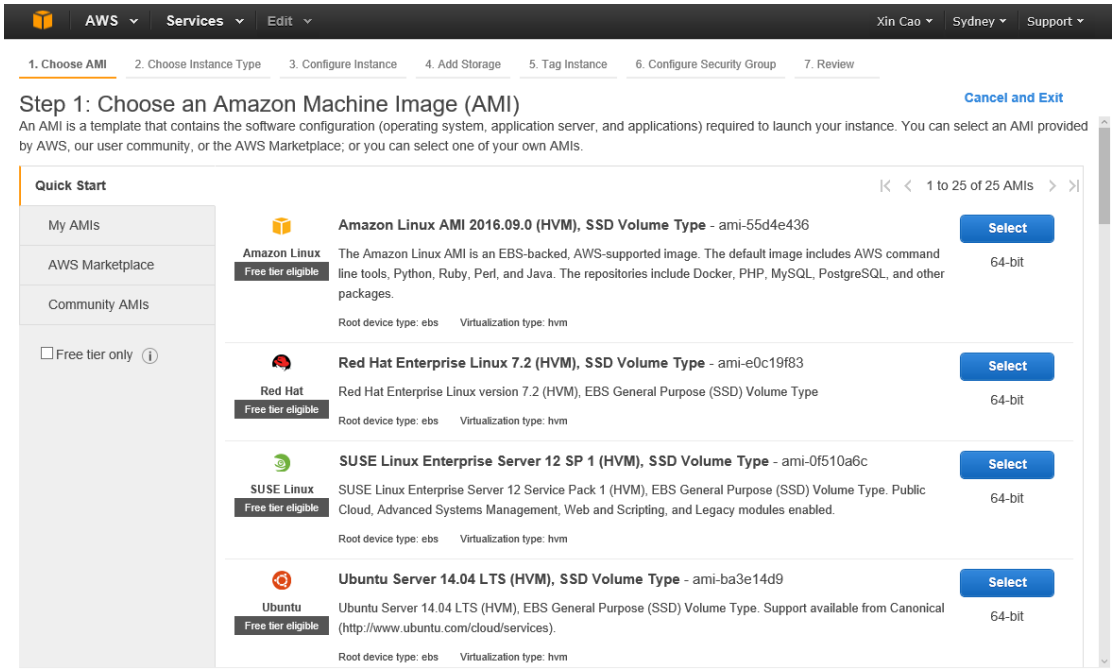


Make sure that the region information on the top right is set to “Sydney”. If it is not, change it to Sydney by selecting from the dropdown menu there.

2. Click on the EC2 link (first link under the Compute category). You will go to a dashboard page like this:



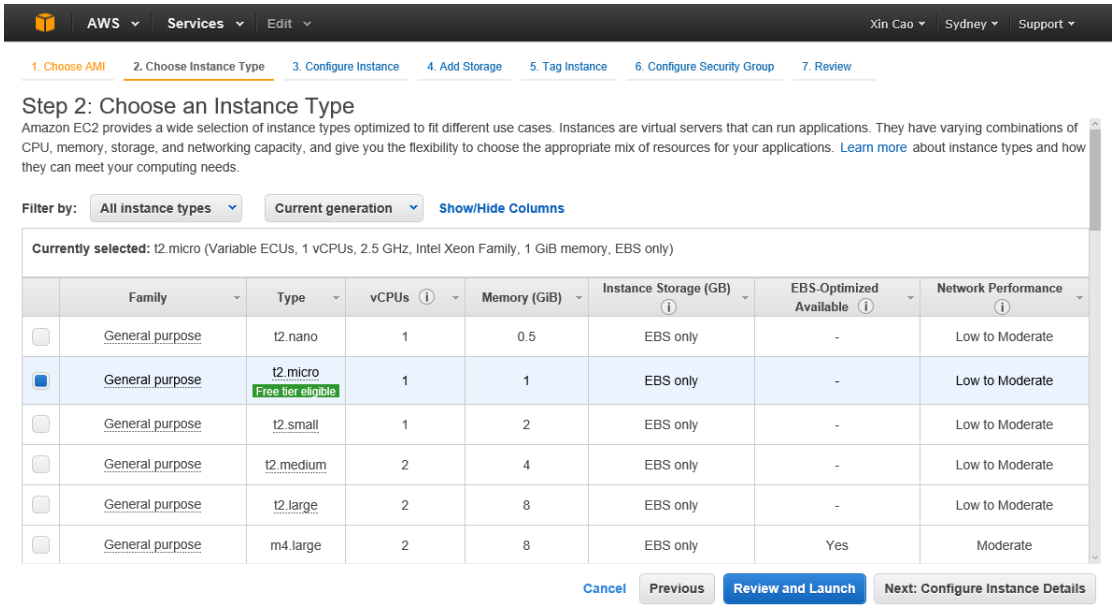
3. Click the blue “Launch Instance” button, and you will be redirected to a page like the following:



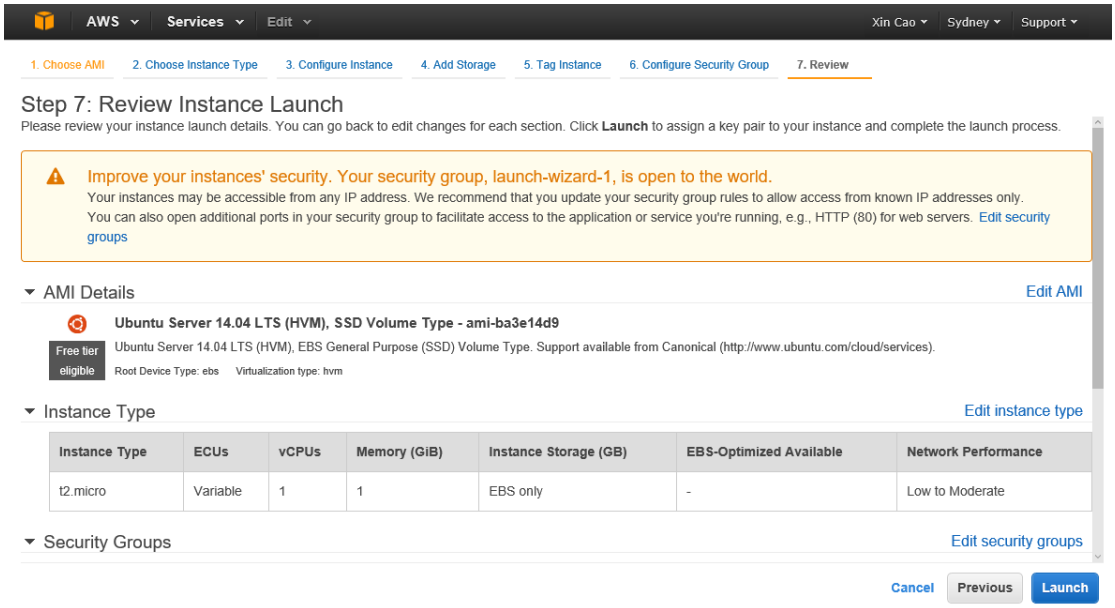
You can use many AMIs (Amazon Machine Image) to finish your task. In this lab, we will use the Ubuntu AMI, and continue to the next step to choose your instance type.

4. Choose the instance type t2.micro, and click on “Review and Launch”.

Caution: This is the only one that is free tier eligible. You will be billed if you select other instance types!

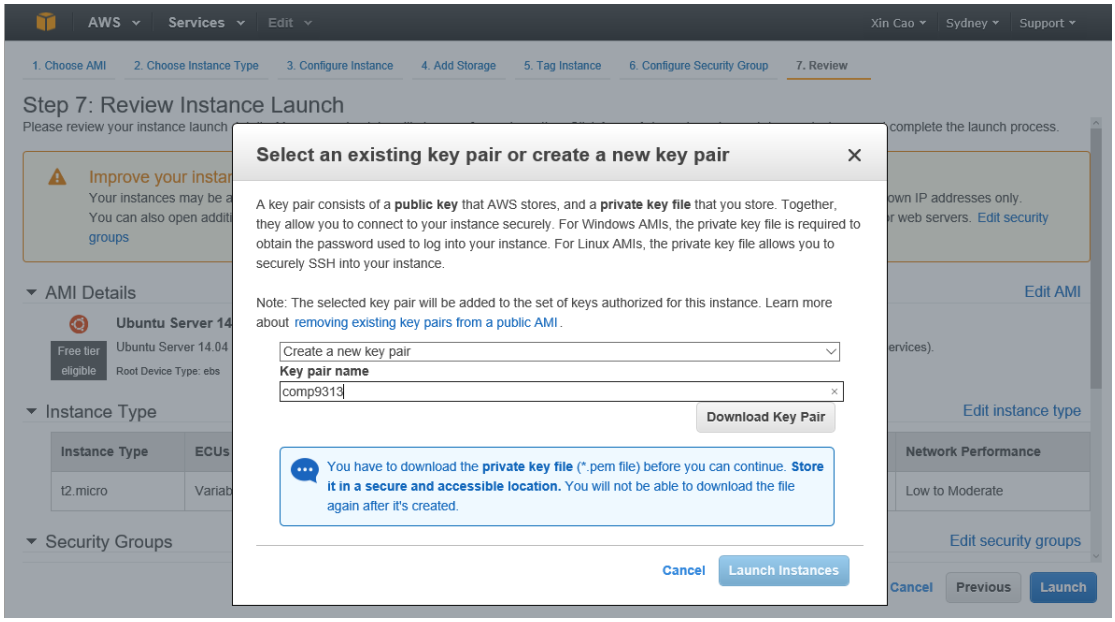


5. In the next page, click on Launch.



6. You will be then prompted to create or use an existing key-pair. Create a new one by choosing “Create a new key pair” from the drop-down menu and giving it some name of your choice (e.g., “comp9313”). You should then download the key pair, and keep it somewhere that you won’t accidentally delete. Remember that there is NO WAY to get to your instance if you lose your key.

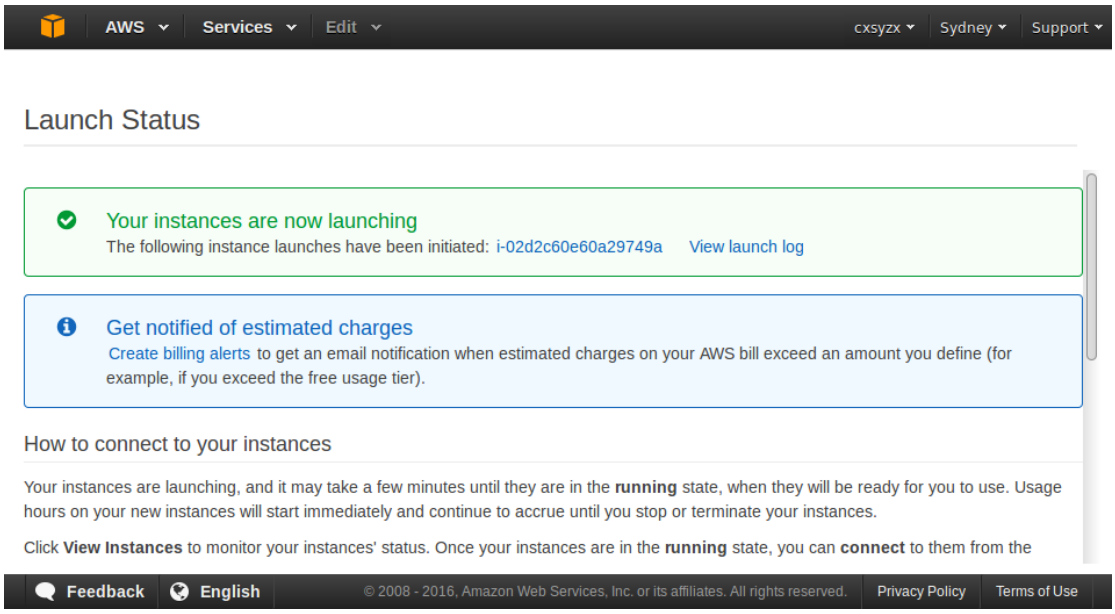
Caution: Don't select the Proceed without a key pair option. If you launch your instance without a key pair, then you can't connect to it.



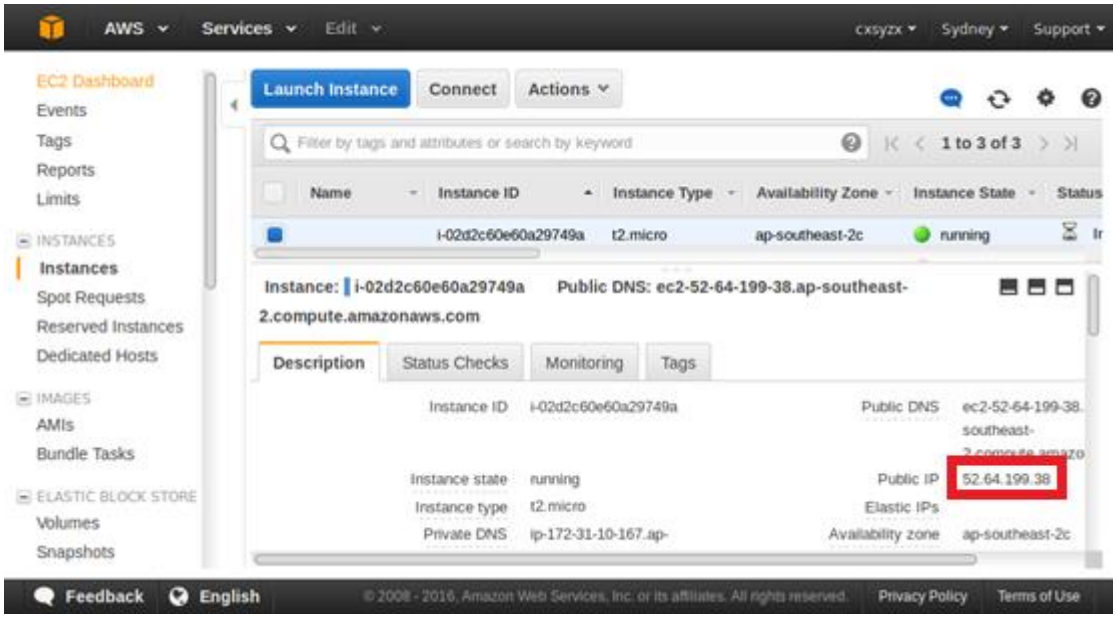
7. Once you download your key, you should change the permissions of the key to user-only RW. Move the file to your home folder, and then do:

```
$ chmod 600 comp9313.pem
```

8. After this is done, click on “Launch Instances”, and you should see a screen showing that your instances are launching:



9. Click on “View Instances” to see your instance state. It should change to “Running” and “2/2 status checks passed” as shown below within some time. You are now ready to ssh into the instance.



10. Note down the Public IP of the instance from the instance listing (in the example, it is 52.64.199.38). Then, do:

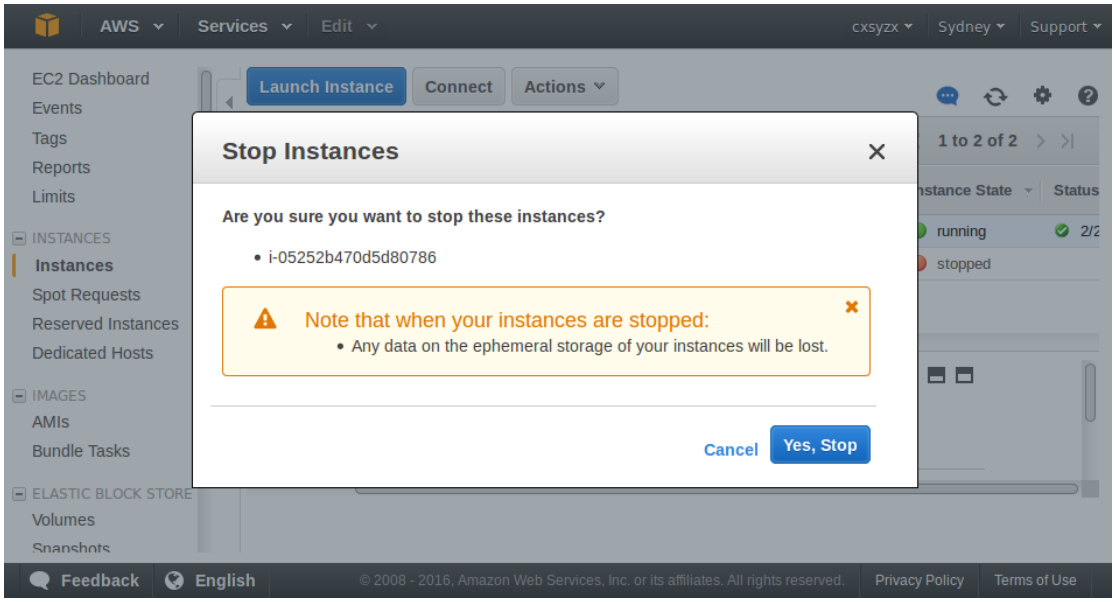
```
$ ssh -i ~/comp9313.pem ubuntu@52.64.199.38
```

Alternatively, you can also use the public DNS to connect to the instance.

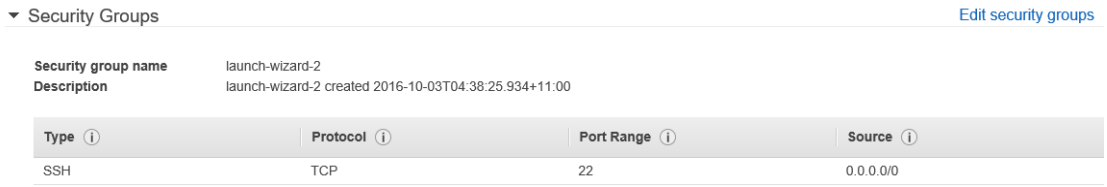
If everything works fine, you should be able to ssh to the AWS instance.

11. To shut down the instance, right click the instance and select “Instance State -> Stop”. Then confirm to stop the instance.

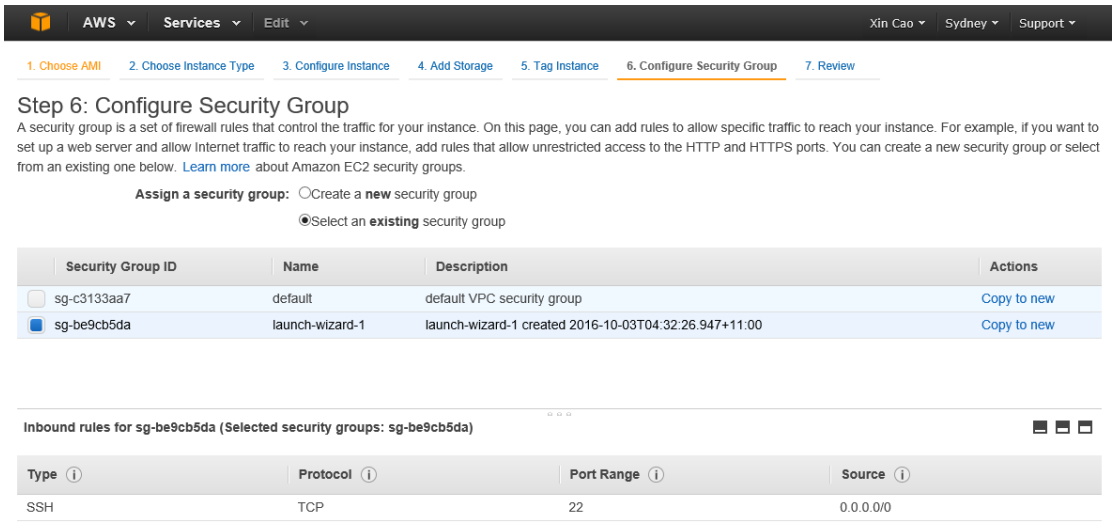
Caution: If you choose terminate, then all the files in this instance will be lost permanently, and you cannot use it again!



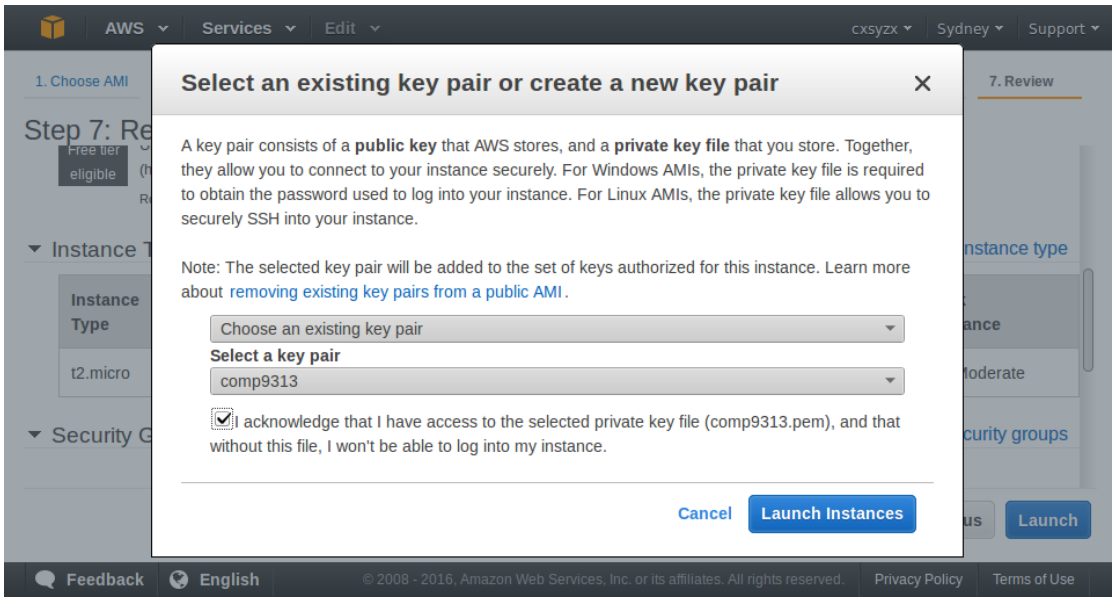
12. You can also launch another instance. This time, after the step “Review and Launch”, click “Edit security groups” (a security group is a set of firewall rules that control the traffic for your instance).



Then, choose the existing security group you created for the first instance.



Next, you can use your existing key pair to launch the instance.

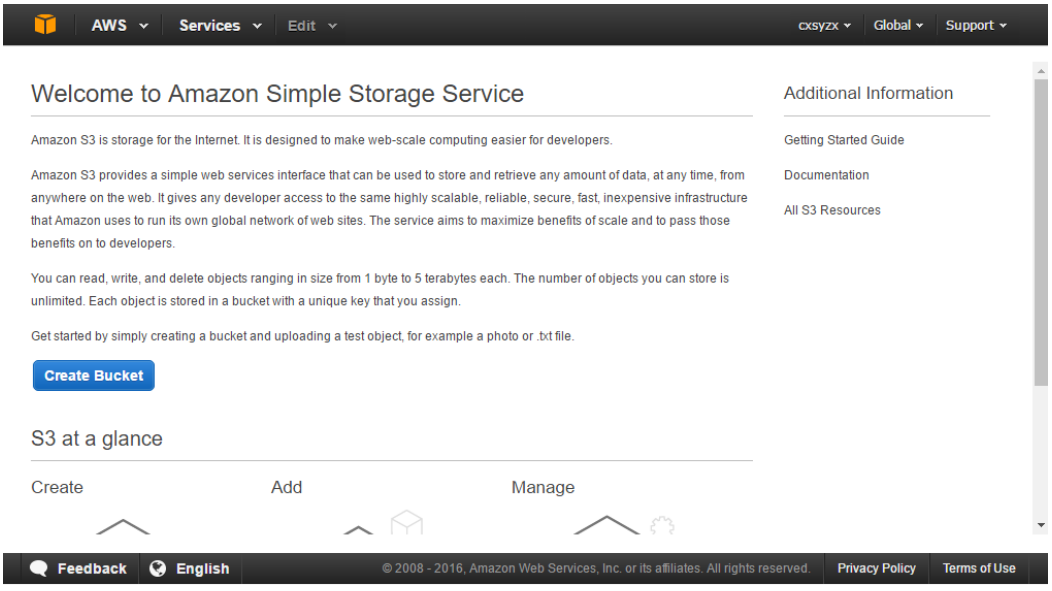


Caution: You will be billed for AWS instances as they are alive, so you will want to terminate them when they aren't in direct use! Here are the Amazon instructions. Always remember to terminate the instances if they will not be used any more. You can stop an instance if you still need to use it later.

Store Data in AWS S3

Create a Bucket in S3

1. Every object in Amazon S3 is stored in a bucket (like a folder in your local file system). Before you can store data in Amazon S3 you must create a bucket. Go back to the AWS Management Console and open the Amazon S3 console.



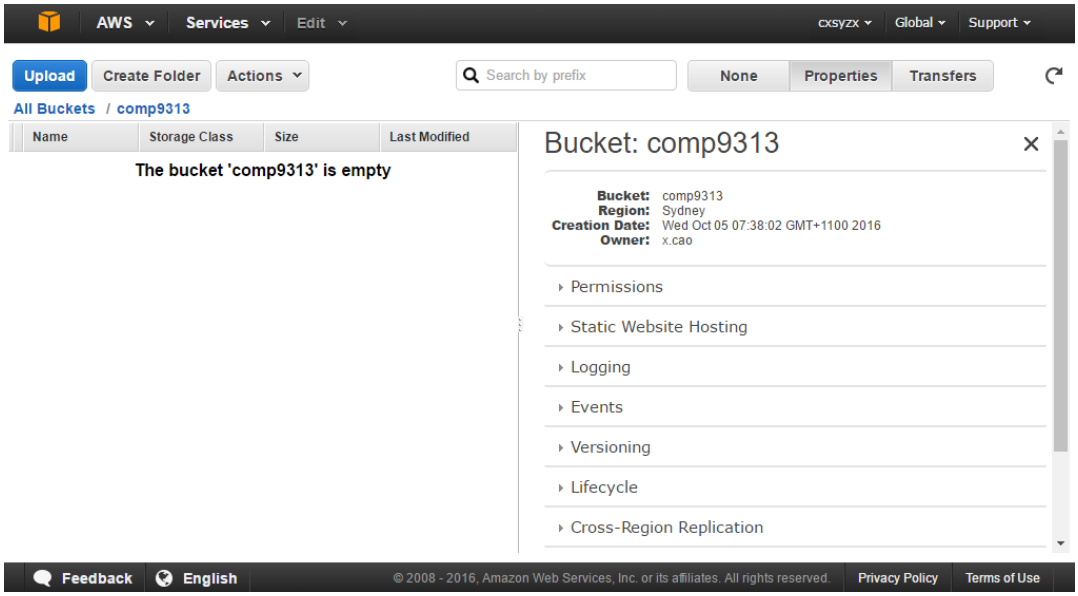
2. Click Create Bucket. The Create a Bucket dialog box appears. Enter a bucket name in the Bucket Name field. The bucket name you choose must be unique across all existing bucket names in Amazon S3. For example, the tutorial names the bucket as “comp9313”.

Bucket names must comply with the following requirements:

- Can contain lowercase letters, numbers, periods (.) and dashes (-)
- Must start with a number or letter
- Must be between 3 and 255 characters long
- Must not be formatted as an IP address (e.g., 265.255.5.4)

Caution: Because S3 allows your bucket to be used as a URL that can be accessed publicly, the bucket name that you choose must be globally unique. If some other account has already created a bucket with the name that you chose, you must use another name. Therefore, it is recommended to name your bucket as your student ID.

In the Region drop-down list box, select region “Sydney”, and click “Create”.



Add and Manage Files in a Bucket:

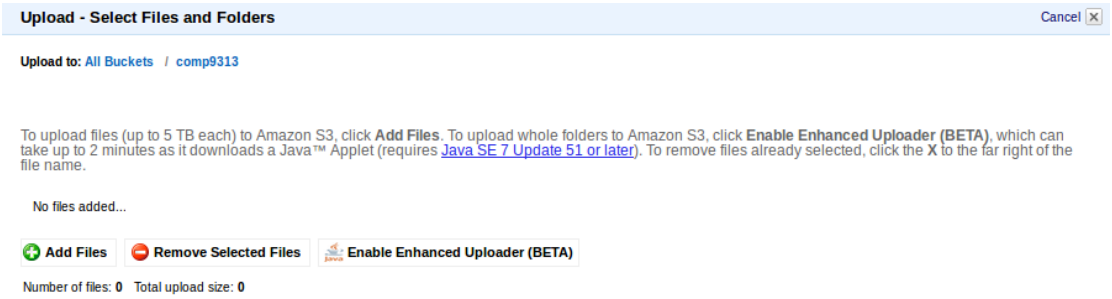
Now that you've created a bucket, you're ready to add an object to it. An object can be any kind of file: a text file, a photo, a video and so forth. When you add a file to Amazon S3, you have the option of including metadata with the file and setting permissions to control access to the file.

In the Amazon S3 console click the bucket you want to upload an object into and then click “Upload” in the Objects and Folders panel. The Upload -

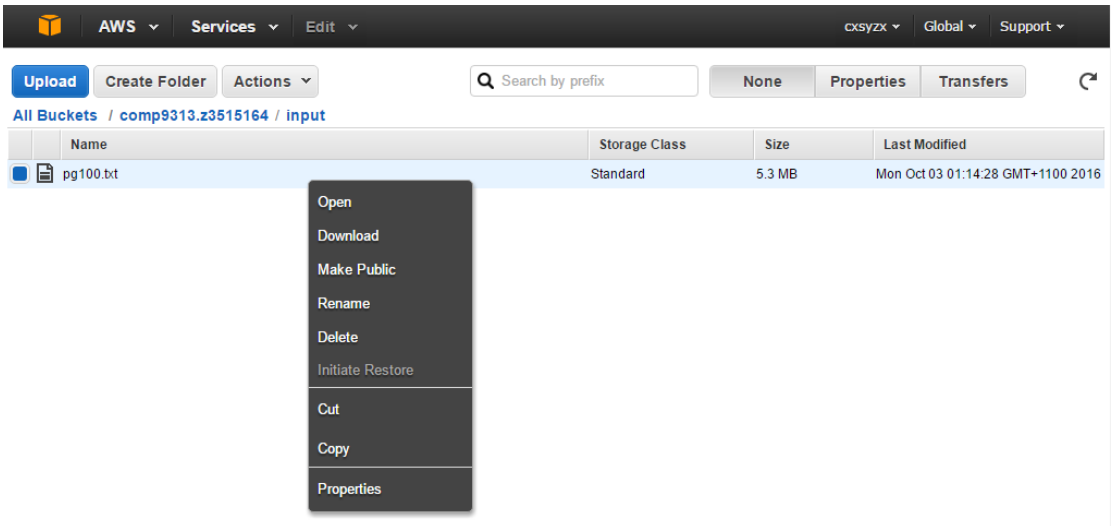
Select Files wizard opens (appearance may differ slightly in different browsers). Download the pg100.txt file, create a folder “input” in your bucket, and upload it into the folder.

Caution: The free tier account only has 5GB S3 storage. If your files exceed this space limit, you will be billed for the service!!!

If you want to upload a folder you must click Enable Enhanced Uploader for the Java applet. After you download the Java applet, the “Enable Enhanced Uploader” link disappears from the wizard. You only need to do this once per console session and you can transfer entire folders. You can cancel this operation if it cannot be finished for several minutes.



You can do various actions on the files in your bucket. Select the file to be managed, then click “Actions”, in the menu you can see all the actions you can do, such as Rename, Cut, and Copy. You can also view the properties of the file.



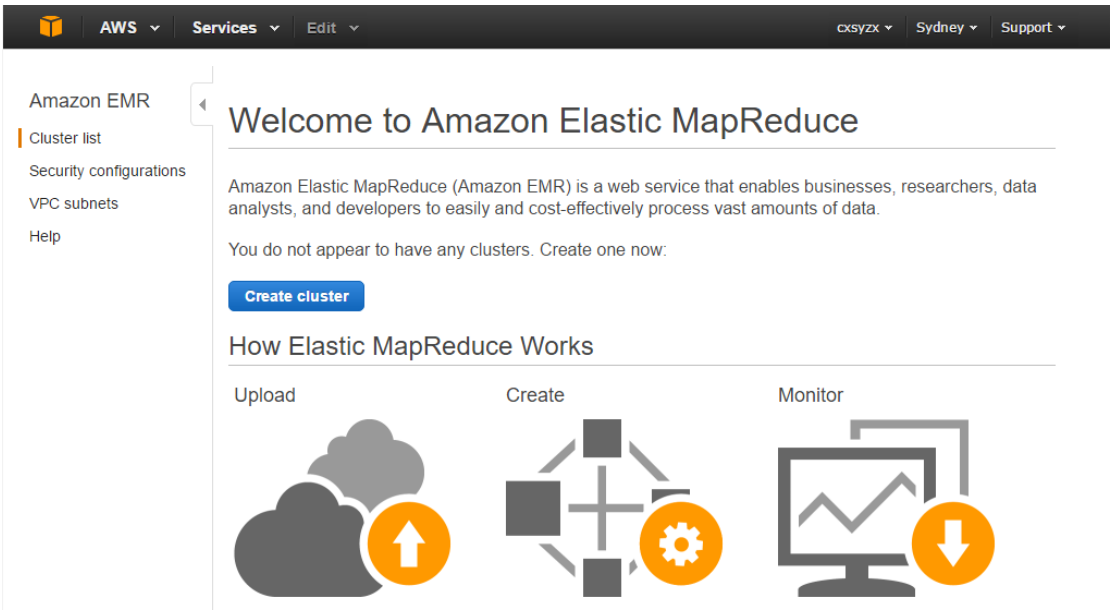
Finally, prepare a WordCount jar file, and upload it to AWS S3.

a) Download the WordCount.java used in Lab 3 from the course home page. Set the number of Reducers as 3. Compile the file and package the MapReduce program as a jar file wc.jar.

c) Test the jar file in your local machine first before uploading to S3.

Run MapReduce Tasks on AWS EMR (Part 1)

1. Go back to the AWS Management Console and open the Amazon EMR console.



2. Choose Create cluster. On the Create Cluster page, you need to do the following:

In General Configuration section:

a) Cluster name: comp9313.lab8

b) Logging: Select

By default, clusters created using the console have logging enabled. This option determines whether Amazon EMR writes detailed log data to Amazon S3.

When this value is set, Amazon EMR copies the log files from the EC2 instances in the cluster to Amazon S3. Logging to Amazon S3 can only be enabled when the cluster is created.

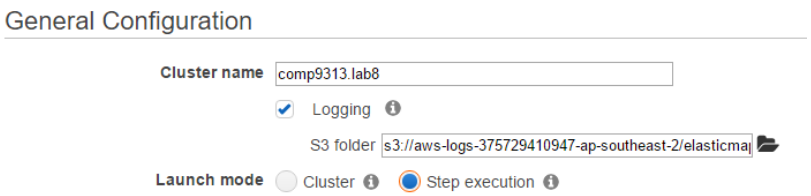
Logging to Amazon S3 prevents the log files from being lost when the cluster ends and the EC2 instances hosting the cluster are terminated. These logs are useful for troubleshooting purposes.

c) S3 folder: use default. The folder is used to store the logs.

You can also type or browse to your Amazon S3 bucket to store the Amazon EMR logs; for example, s3://YOUR_BUCKET/logs, or you can allow Amazon EMR to generate an Amazon S3 path for you. If you type the name of a folder that does not exist in the bucket, it is created for you.

d) Launch mode: select “Step execution. “

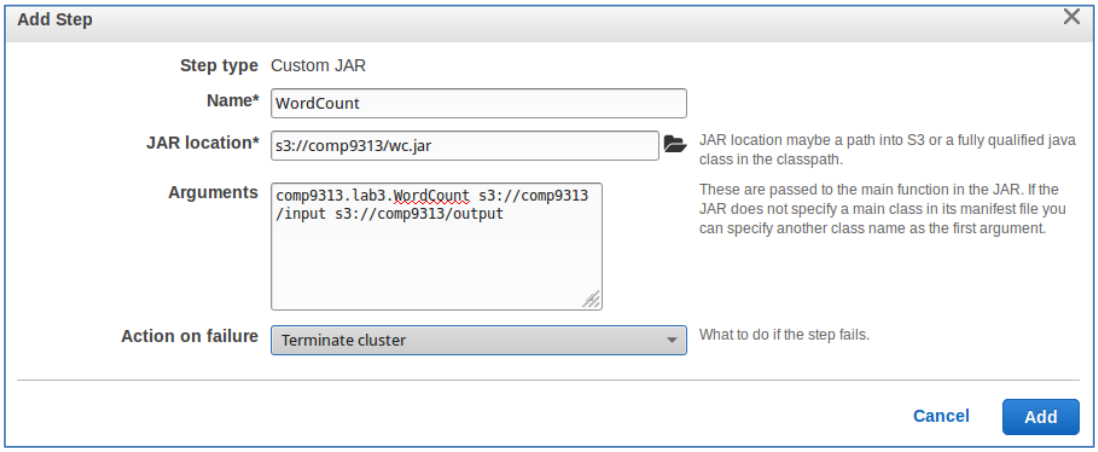
If you select “Cluster”, the instances will keep running after your MapReduce task is finished. However, you can do more jobs without creating a new cluster. By selecting “Step execution”, the instances will be terminated after the task is completed.



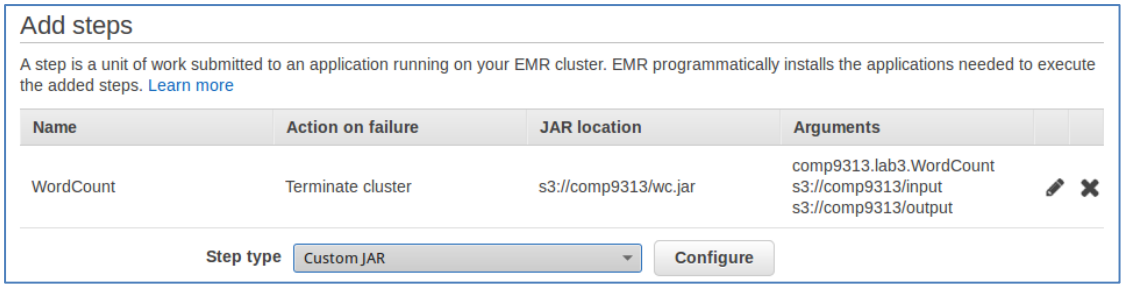
In Add steps section:

a) Set the step type as Custom JAR

b) Click “Configure”, set Name as “WordCount”, set JAR location as “s3://comp9313/wc.jar”, set Arguments as “comp9313.lab3.WordCount s3://comp9313/input s3://comp9313/output”, select “Terminate cluster” for Action on Failure, and finally click Add.



Then, in the Add steps section, you will see:



In the Software Configuration section:

a) Vendor: select Amazon

b) Release: select emr-5.0.0

In the Hardware Configuration section:

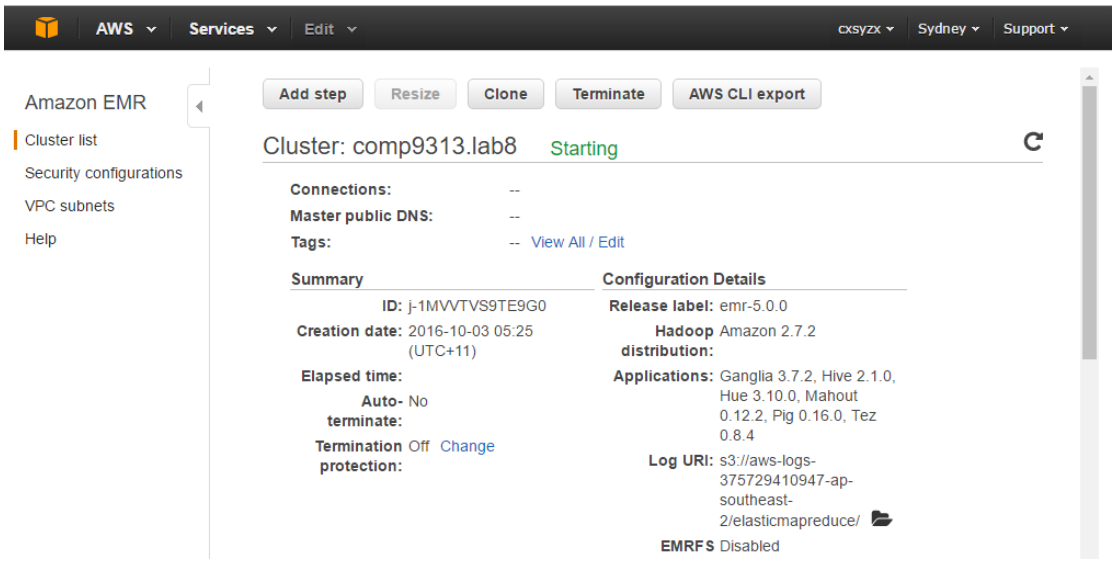
a) Instance type: use m4.large (much cheaper than the default m3.xlarge)

b) Number of instances: enter 3

In the Security and Access section:

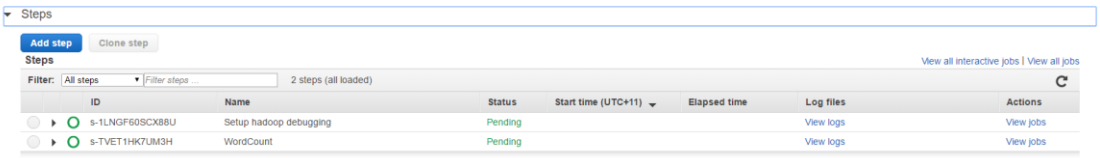
Accept the remaining default options.

6. Choose Create cluster. You should see:



Later, you will see the information for Connections and Master public DNS is updated, since the cluster is already started.

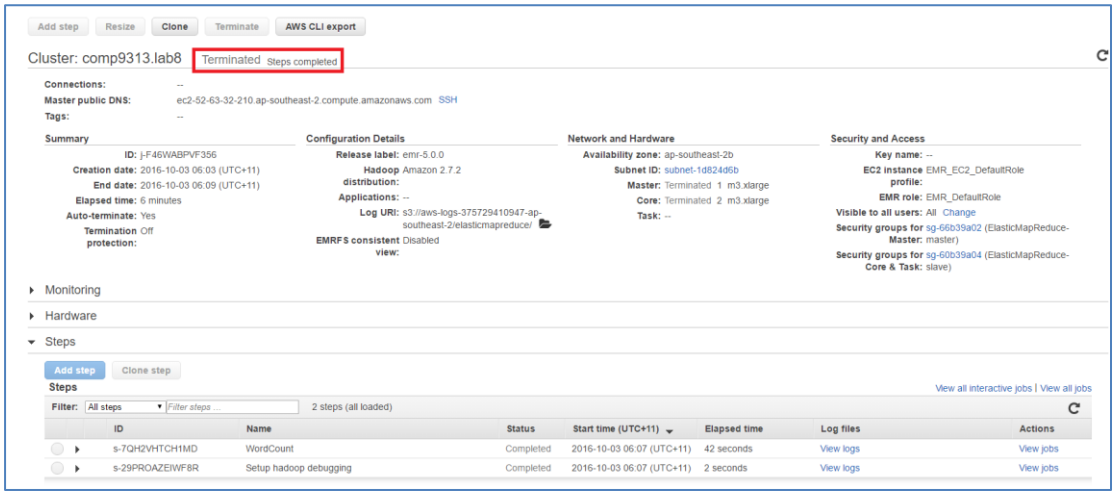
Click “Steps”, and you should see two jobs listed.



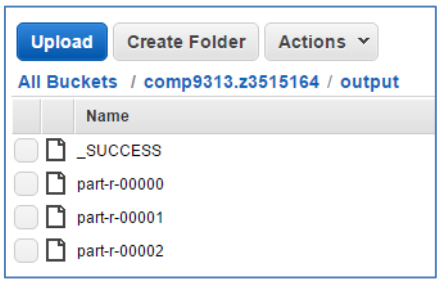
7. Wait until the WordCount task is finished. Note that this may take several minutes.

In the meantime, you can begin working on the next section, and go back to check the results later.

8. If the task is completed, you should see:



Go to your S3 bucket, the results should be stored there.

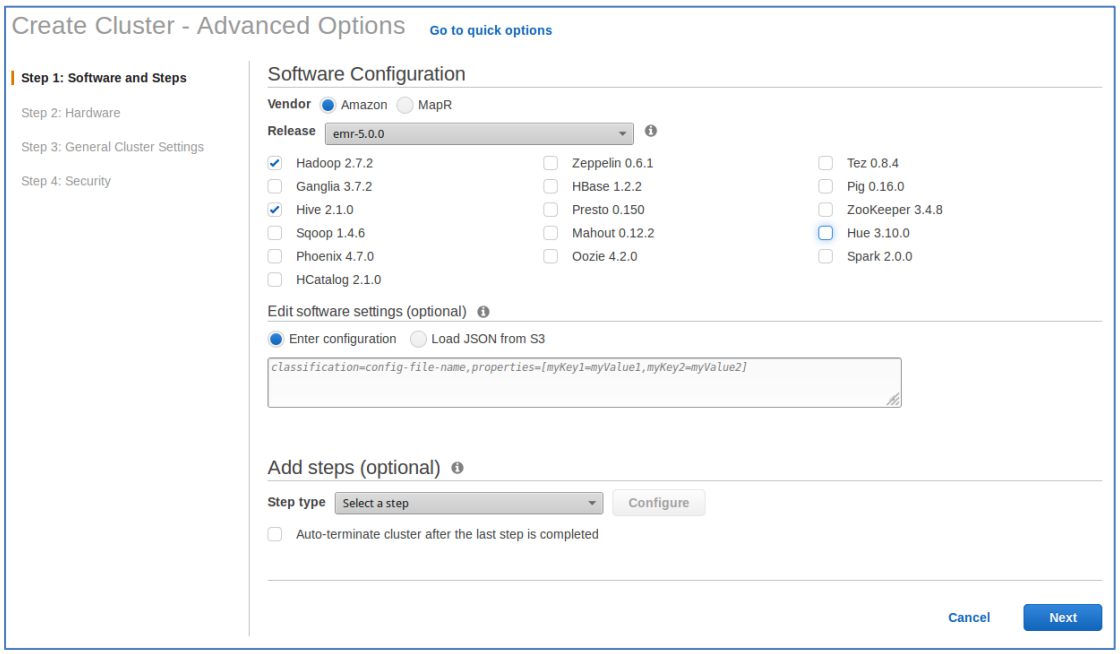


Run MapReduce Tasks on AWS EMR (Part 2)

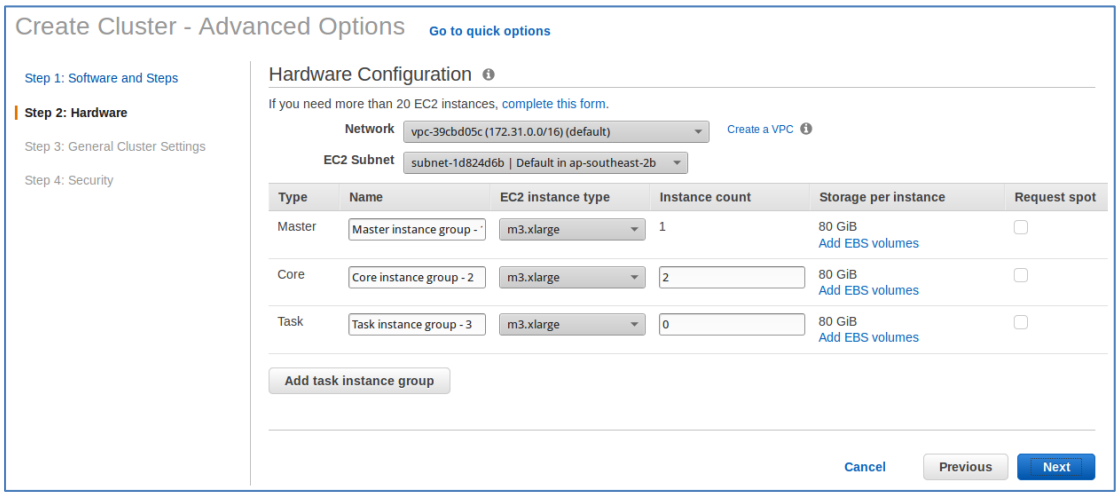
In the previous section, we add a step to the cluster, and wait for the completion of the job. In this section, we will ssh to the cluster to do a MapReduce job.

1. Choose Create cluster. On the Create Cluster page, click “Go to advanced options”.

2. In Step 1, select “Amazon” for Vendor, emr-5.0.0 for Release, and only use “Hadoop 2.7.2” and “Hive 2.1.0” in the cluster. Accept the other default configurations, and click “Next”.

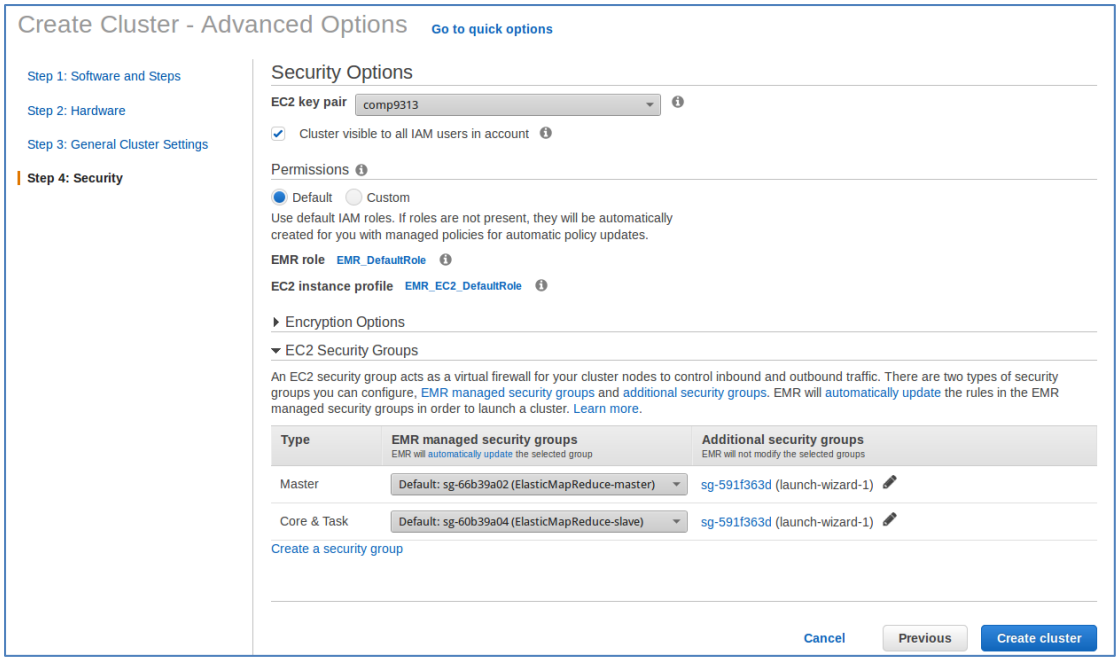


3. In Step 2, select the default m3.xlarge as the instance type for both Master and Core. Click “Next”



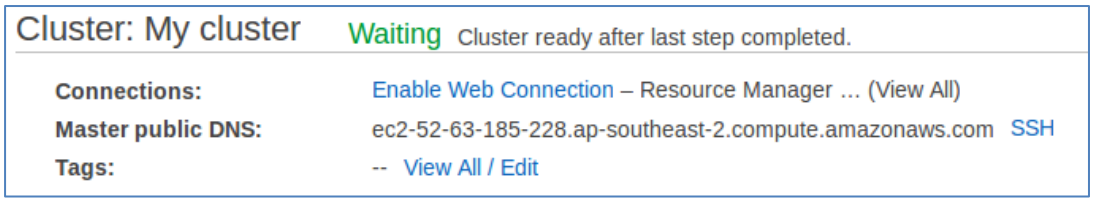
4. In Step 3, accept all default configurations and click “Next”.

5. In Step 4, use your key pair for the cluster. Click “EC2 Security Groups”, configure the security groups for both Master and Core as “launch-wizard-1”. Finally, click “Create Cluster”.

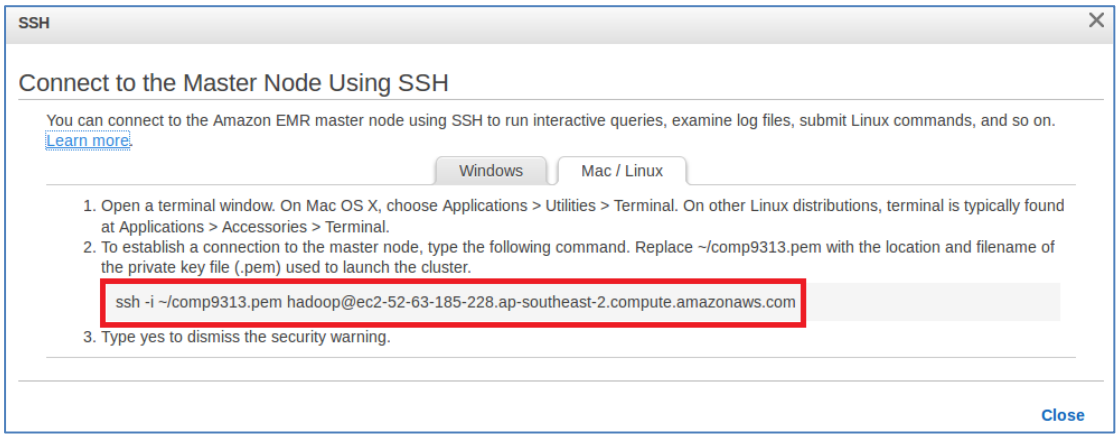


6. Waiting for the starting of the cluster. You can go back to check the results of your first cluster.

Once the information for “Connection” and “Master public DNS” is updated, your cluster is started, and you can ssh to the master node now.



Click SSH in the line of “Master public DNS:”, you will see:



SSH to the master node by copying the command as shown in the dialog:

```
$ ssh -i ~/comp9313.pem hadoop@YOUR_INSTANCE
```

```
comp9313@comp9313-VirtualBox:~$ ssh -i ~/comp9313.pem hadoop@ec2-52-63-185-228.a
p-southeast-2.compute.amazonaws.com
The authenticity of host 'ec2-52-63-185-228.ap-southeast-2.compute.amazonaws.com
 (52.63.185.228)' can't be established.
ECDSA key fingerprint is 1f:7b:ac:f4:d7:fa:d7:68:32:be:ac:b9:c7:41:78:17.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-52-63-185-228.ap-southeast-2.compute.amazonaws.c
om,52.63.185.228' (ECDSA) to the list of known hosts.
Last login: Tue Oct  4 21:17:21 2016

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|
```

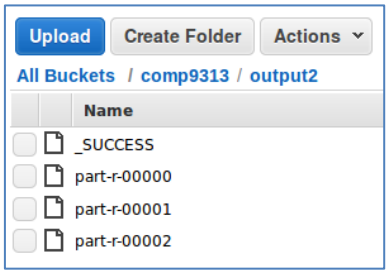
7. Download the jar file from S3 by the following command:

```
$ hadoop fs -get s3://comp9313/wc.jar
```

8. Run the MapReduce task. Generate the results in a different folder!

```
$ hadoop jar wc.jar comp9313.lab3.WordCount s3://comp9313/input
s3://comp9313/output2
```

9. Wait for the completion of the task, and check the results in your S3 bucket. You should see:



10. You can also download “pg100.txt” from S3, and put the file to HDFS, and run the MapReduce task by reading/writing files from/to HDFS instead of S3.

```
$ hdfs dfs -mkdir input

$ hdfs dfs -put pg100.txt input

$ hadoop jar wc.jar comp9313.lab3.WordCount input output
```

Caution: The I/O between the cluster and S3 is also billed if your transfer exceeds the free tier limit!!!

11. You can also add a new step to this cluster to run a MapReduce task. Try it by yourself.

12. Caution: Do not forget to terminate the cluster after you finish all labs!!! (click “Terminate” and turn termination protection off)