

AWS Setup Guidelines

For CSE6242 HW3, updated version of the [guidelines](#) by Diana Maclean

Important steps are highlighted in yellow.

What we will accomplish?

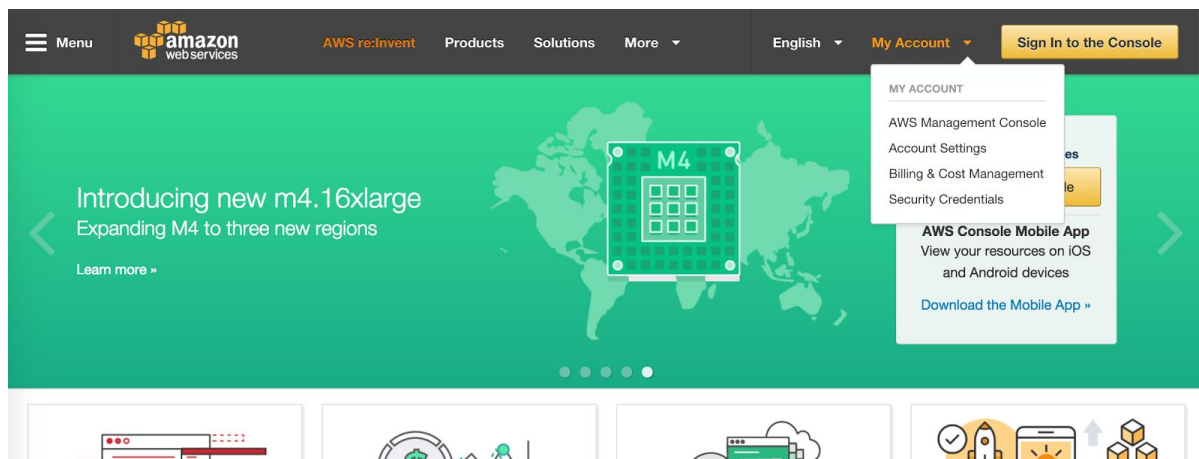
This guideline helps you get set up with the Amazon Web Services (AWS, a “cloud” platform) where you will run large-scale analysis on big data. Here are you will learn to do

1. [Create an AWS account](#) (to get access to EC2, Elastic MapReduce and S3 storage).
2. [Create storage buckets on S3](#) (to save outputs and logs of MapReduce jobs).
3. [Create a key pair](#) (required for running MapReduce jobs on EC2).
4. [Get Access Keys](#) (also required for running jobs on EC2).
5. [Redeem your free credit](#) (worth \$100).
6. [Set up a CloudWatch Usage Alert](#)
7. [Familiarize yourself with S3, EC2 and EMR](#) (by doing a sample MapReduce run).
8. [Debugging](#)

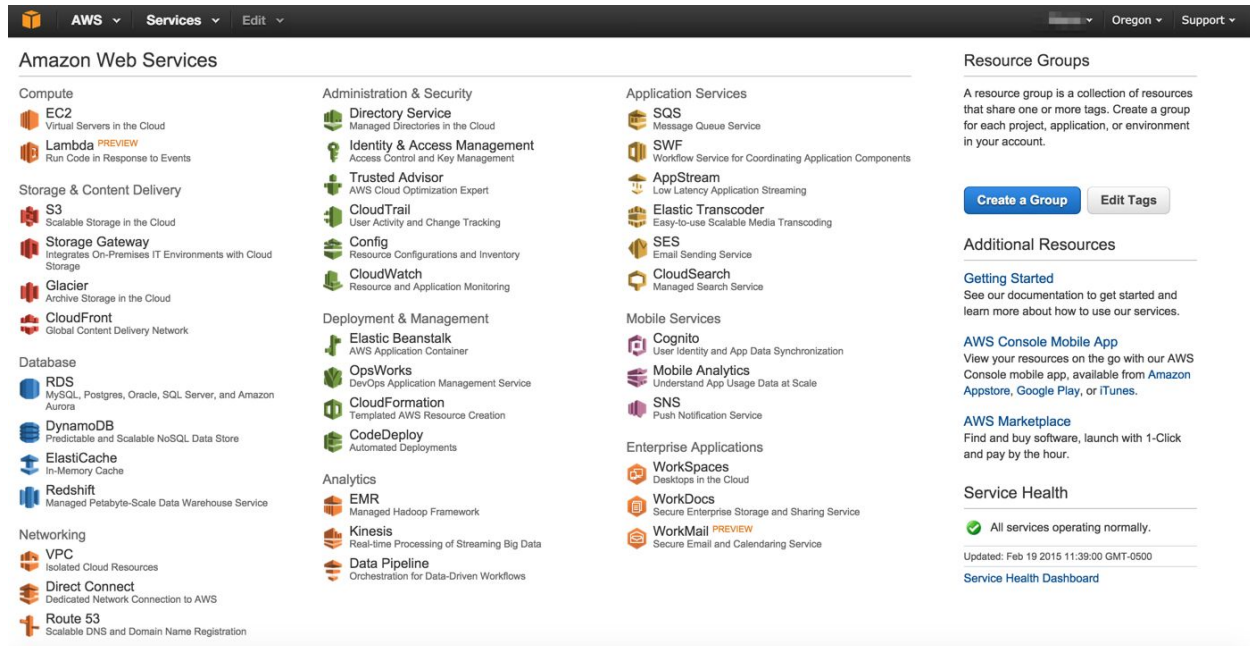
1. Create an AWS account

- Go to <http://aws.amazon.com> and sign up for an account, if you do not have one already. □
- For now, please enter the required details, including payment details (you will need a **valid credit card or debit card** to sign up). Please follow Step 5 to redeem the \$100 credits.
- Validate your account with the identity verification through your phone.

Once your account has been created and your payment method verified, you should have access to the AWS Management Console.



You AWS Management Console should look like this:



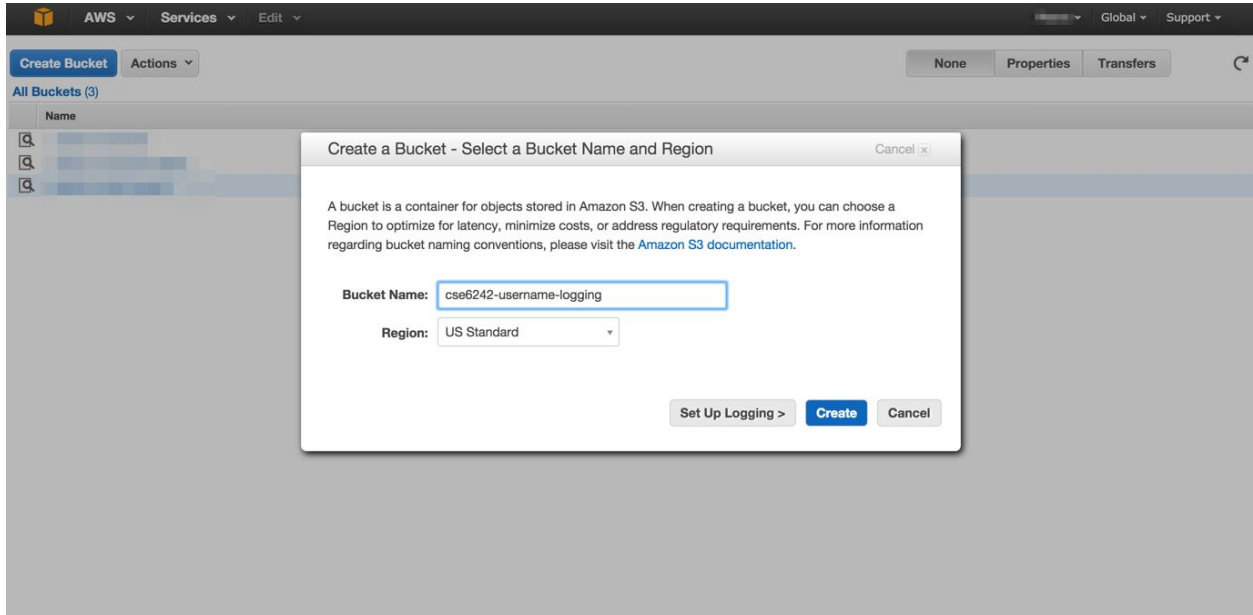
2. Create storage buckets on S3

In the AWS Management Console click on “S3” under Storage & Content Delivery. We need S3 for two reasons:

- (1) an EMR workflow requires the input data to be on S3;
- (2) EMR workflow output is always saved to S3.

Data (or objects) in S3 are stored in what we call “buckets”. You can think of buckets as folders. You will need to create some buckets of your own to (1) store your EMR output and (2) store your log files if you wish to debug your EMR runs. Once you have signed up, we will begin by creating the log bucket first.

- i. In the S3 console, click on “Create Bucket”.



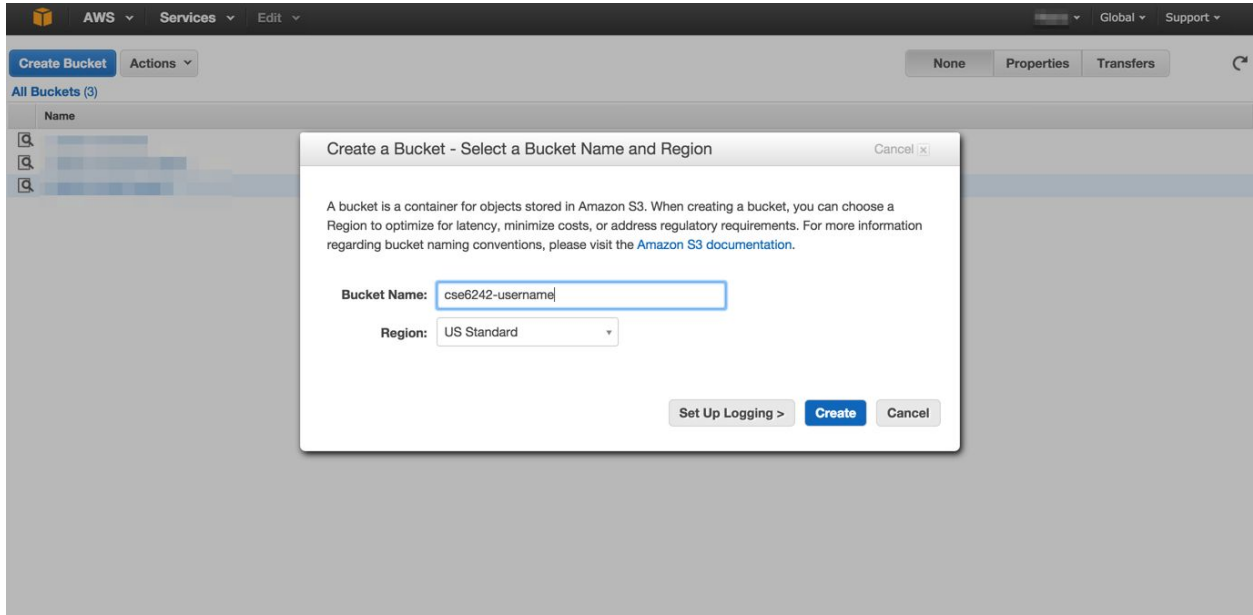
ii. All S3 buckets need to have unique names. You could name the logging bucket cse6242-`<gt;`username-logging. Important: Please select “US Standard” in the Region dropdown. Click on “Create” (not on “Set Up Logging >>”).

US Standard is important, because if you have buckets in other regions, data transfer charges would apply.

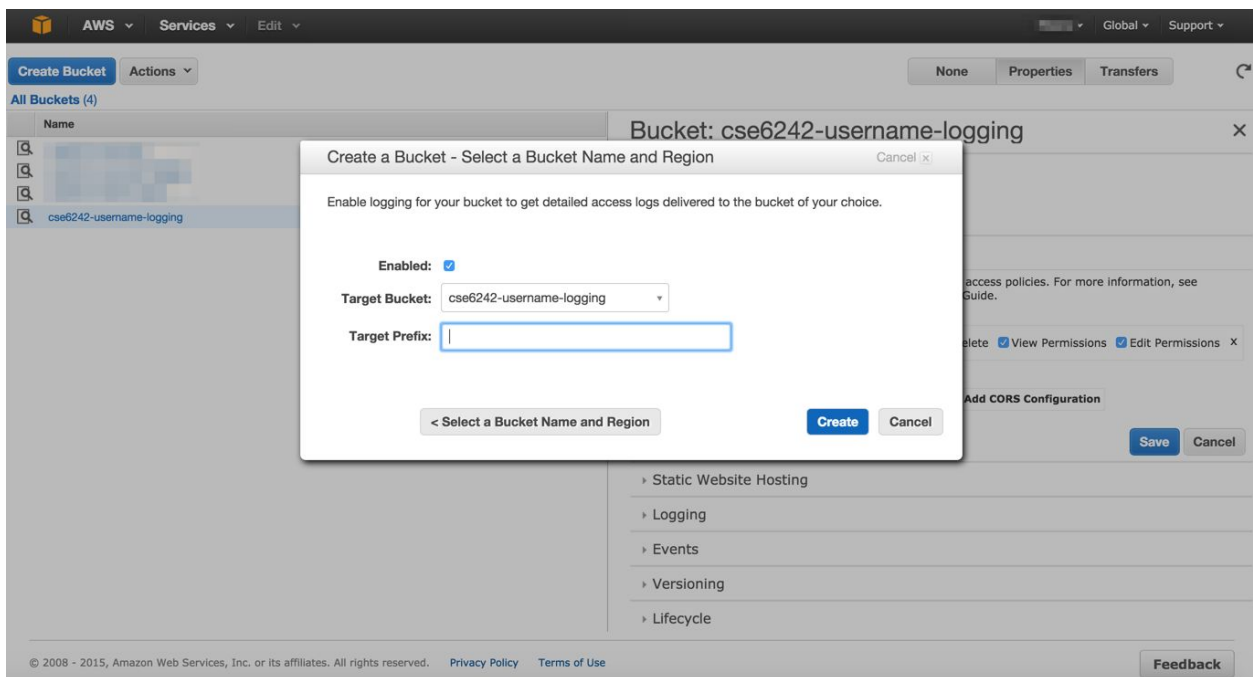
iii. Your new bucket will appear in the S3 console. Clicking on it will show you that it is empty.



iv. Now we will create our main bucket. Go back to the main screen (clicking on “All Buckets”). Again, click on “Create Bucket”. Call this one cse6242-`<gt;`username-. Again, pick “US Standard” for the Region dropdown. Since we will link this bucket to our logging bucket, the regions for the two buckets should be the same. We will link our logging bucket to the one we are creating now, so click on “Set Up Logging >”.



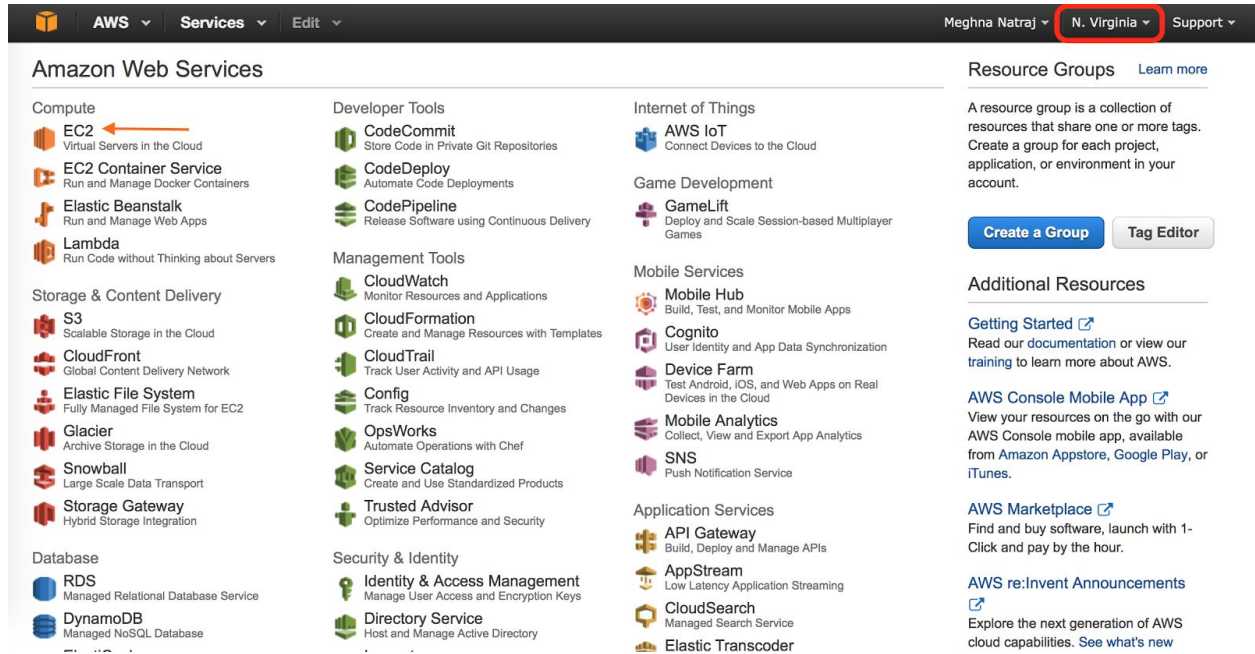
v. Click on “Enabled” to enable logging, and start typing in the name of your logging bucket. It should appear in the drop down menu, select it. Clear the “Target Prefix” field and click “Create”.



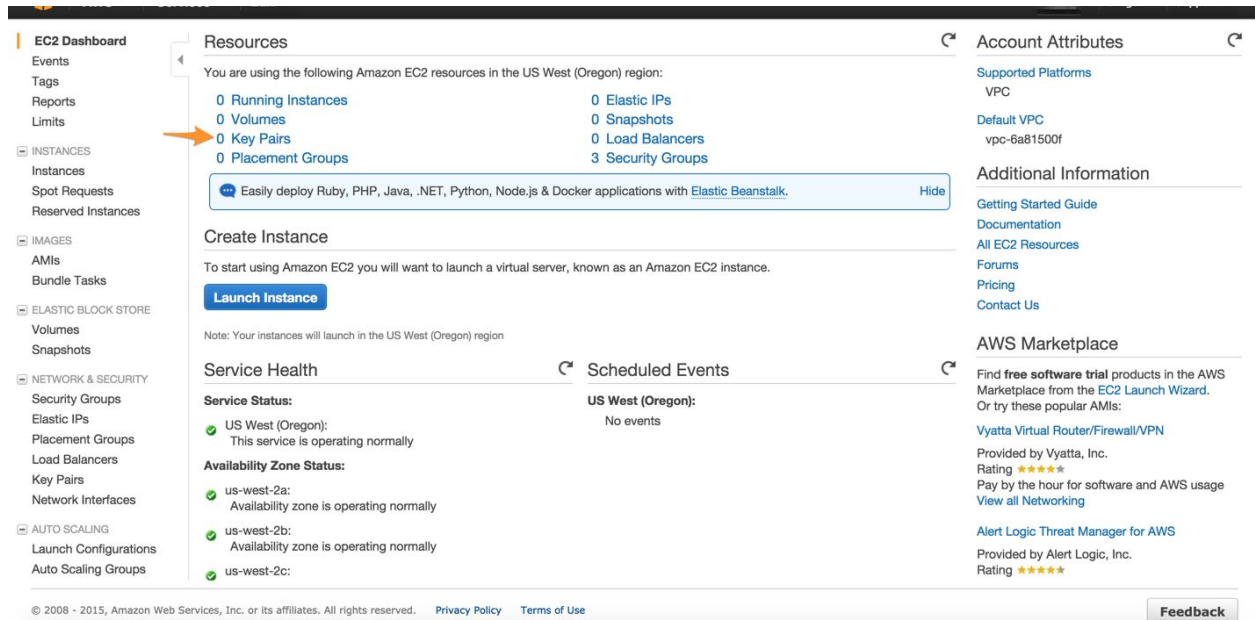
We are done creating buckets at this point.

3. Create a key pair

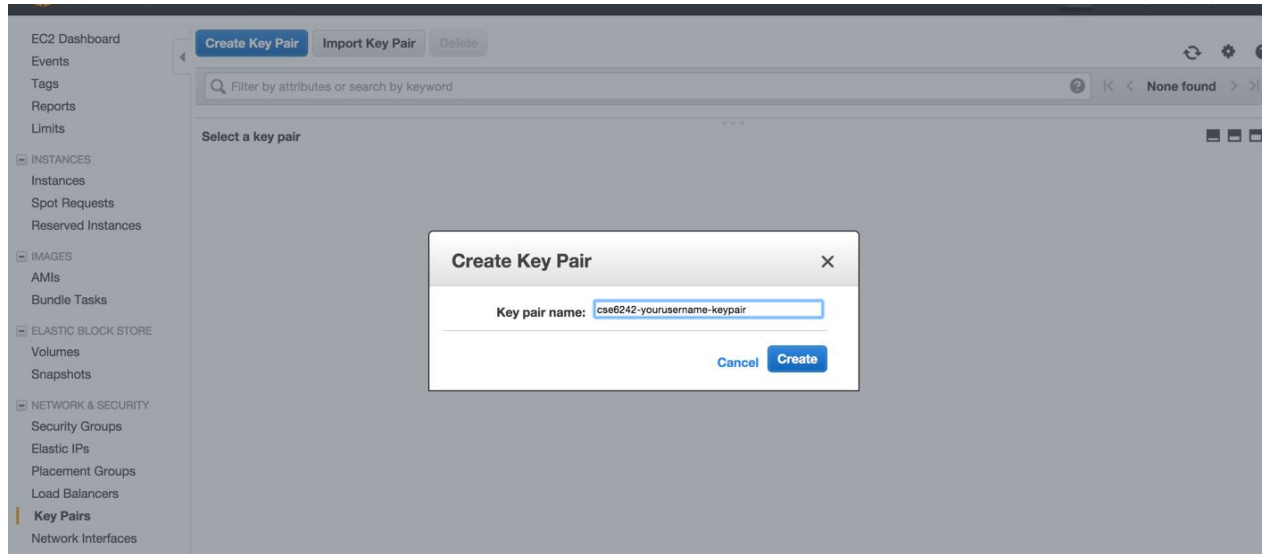
Select the region on the top right as US East (N. Virginia) since the data bucket is stored in this region. When you run jobs on EMR, you will need to have a valid public/private key pair. To create your first key pair, click on “EC2” under Compute in the AWS Management Console.



You should see a link stating “0 Key Pairs” under Resources. Click on this.



You will be given an option to “Create Key Pair”. Name your key pair as you wish. Upon providing a name and clicking on “Create”, your private key (a .pem file) will automatically download. **Save it in a safe place where you will be able to find it again (IMPORTANT, do not lose this file).**



If you need to access your public key, you will be able to find it in the same place where you found your account credentials. Amazon keeps no record of your private key, and if you lose it, you will need to generate a new set.

If your computer runs **Windows**, use the steps in the following link to convert your .pem file to a .ppk file for use with PuTTY.

Read the section titled Converting Your Private Key Using PuTTYgen in the link below:

<http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html>

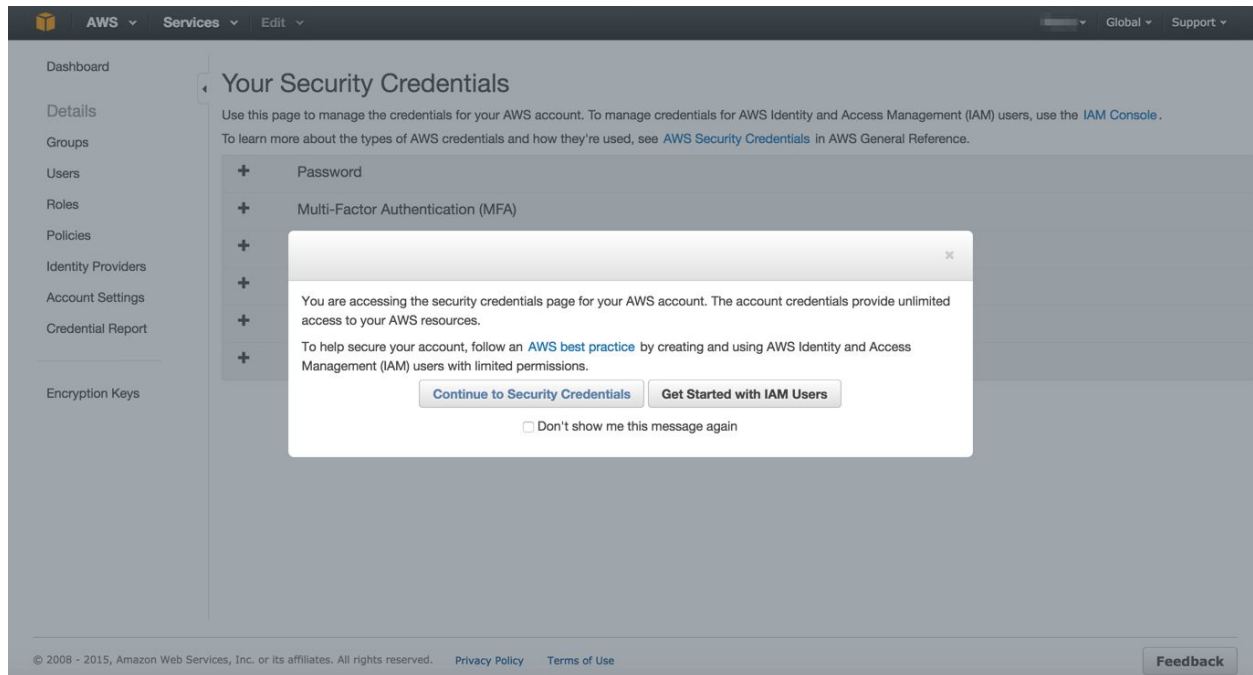
Note: If you use the AWS Management Console, you would typically not be required to access your private key. However, you will be asked to name your access key pair and the private key each time you run an EMR job.

If you wish to log into the master node running your MapReduce job, you will need your .pem file (you will need this in case you wish to run an interactive HIVE/PIG job flow). To log on to the master node (you can find the address of the master node from the MapReduce dashboard), you will need to do the following: **(do not copy paste the command from this pdf as your command may fail due to the presence of special characters)**

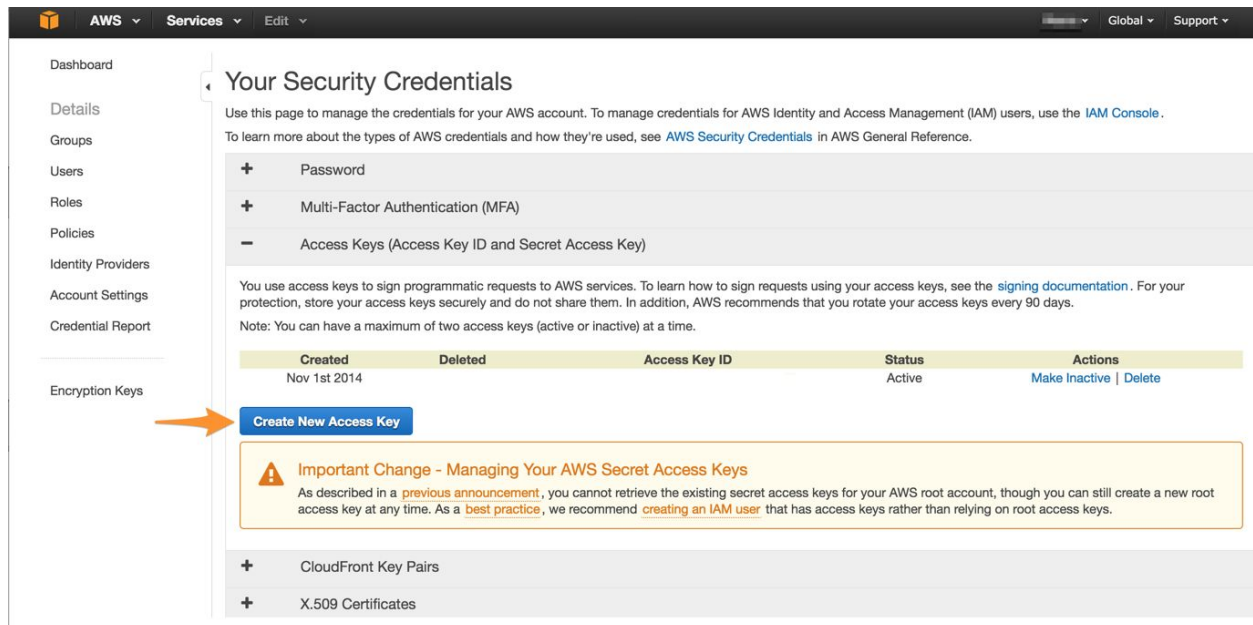
```
$ ssh hadoop@<master-node-address> -i <path-to-pem-file>/<pem-file-name>.pem
```

4. Get Access Keys (new site)

Click on “Security Credentials” under your username (top right). Click on “Continue ...”



Click on the **Create a new Access Key** link (under Access Keys), and download the Access Key file (**do not lose this file**). Now you are ready to run a MapReduce job.



5. Redeem your free credit

In order to add the credit to your account, you will need your unique Credit Code obtained after applying for the AWS Educate program for Students (follow steps listed at the start of HW3) . Once you have your code, go to your account page (<http://aws.amazon.com/account>)

The screenshot shows the AWS Management Console home page. The 'My Account' dropdown menu is open, showing options like 'Billing & Cost Management', 'Security Credentials', and 'Sign Out'. An orange arrow points to the 'My Account' dropdown.

Click on “Credits”. Enter the Code into the Promo Code text box, and click Redeem.

The screenshot shows the AWS 'Credits' page. The 'Credits' link in the left navigation menu is highlighted with an orange arrow. The main content area shows a 'Credits' section with a 'Promo Code' input field and a 'Redeem' button. Below this is a table of redeemed credits and a 'Total Amount of Credits Remaining' section.

Expiration Date	Credit Name	Credits Used	Credits Remaining	Applicable Products
				See complete list

Total Amount of Credits Remaining:

Please contact the CSE6242 instructors immediately if this does not work. You can check the credit remaining by clicking on the “Account Activity” link from your account page or by returning to this page. Sometimes this can take a while to update, so don’t be surprised if recent changes are not immediately apparent. We will set up a monitor in the next step which is triggered when you utilize half of the credit.

6. Set up a CloudWatch Usage Alert

Make sure your region (in the upper right corner of the screen) is set to: US East (US Standard). Test whether this email alert is working before scheduling in practice. That is, out of 100\$, when your credit balance goes below say 95\$, schedule a test alert and make sure it works. Remember this alert works only once. So once you got an alert for 95\$, you schedule the next alert for 70\$ and the next one for 60\$ and so on.

<http://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/free-tier-alarms.html>.

Now we will turn on alerts.

1. Go to the “Billing and Cost Management” page. (Log In using your AWS credentials if necessary)

The screenshot shows the Amazon Web Services console interface. The top navigation bar includes the 'My Account' dropdown menu, which is open and shows the following options: 'Billing & Cost Management', 'Security Credentials', and 'Sign Out'. An orange arrow points to the 'Billing & Cost Management' option. Below the navigation bar, the console is organized into several columns of service categories:

- Compute:** EC2 (Virtual Servers in the Cloud), Lambda (Run Code in Response to Events).
- Storage & Content Delivery:** S3 (Scalable Storage in the Cloud), Storage Gateway (Integrates On-Premises IT Environments with Cloud Storage), Glacier (Archive Storage in the Cloud), CloudFront (Global Content Delivery Network).
- Database:** RDS (MySQL, Postgres, Oracle, SQL Server, and Amazon Aurora), DynamoDB (Predictable and Scalable NoSQL Data Store), ElastiCache (In-Memory Cache), Redshift (Managed Petabyte-Scale Data Warehouse Service).
- Networking:** VPC (Isolated Cloud Resources), Direct Connect (Dedicated Network Connection to AWS), Route 53 (Scalable DNS and Domain Name Registration).
- Administration & Security:** Directory Service (Managed Directories in the Cloud), Identity & Access Management (Access Control and Key Management), Trusted Advisor (AWS Cloud Optimization Expert), CloudTrail (User Activity and Change Tracking), Config (Resource Configurations and Inventory), CloudWatch (Resource and Application Monitoring).
- Deployment & Management:** Elastic Beanstalk (AWS Application Container), OpsWorks (DevOps Application Management Service), CloudFormation (Templated AWS Resource Creation), CodeDeploy (Automated Deployments).
- Analytics:** EMR (Managed Hadoop Framework), Kinesis (Real-time Processing of Streaming Big Data), Data Pipeline (Orchestration for Data-Driven Workflows).
- Application Services:** SQS (Message Queue Service), SWF (Workflow Service for Coordinating Application Components), AppStream (Low Latency Application Streaming), Elastic Transcoder (Easy-to-use Scalable Media Transcoding), SES (Email Sending Service), CloudSearch (Managed Search Service).
- Mobile Services:** Cognito (User Identity and App Data Synchronization), Mobile Analytics (Understand App Usage Data at Scale), SNS (Push Notification Service).
- Enterprise Applications:** WorkSpaces (Desktops in the Cloud), WorkDocs (Secure Enterprise Storage and Sharing Service), WorkMail (Secure Email and Calendaring Service).

On the right side of the console, there are sections for 'Additional Resources' (Getting Started, AWS Console Mobile App, AWS Marketplace) and 'Service Health' (All services operating normally).

2. Under Preferences, check the box labeled **Receive Billing Alerts**

Dashboard
Bills
Cost Explorer
Payment Methods
Payment History
Consolidated Billing
Account Settings
Reports
Preferences
Credits
Tax Settings
DevPay

Preferences

Receive PDF Invoice By Email
Turn on this feature to receive a PDF version of your invoice by email. Invoices are generally available within the first three days of the month.

Receive Billing Alerts
Turn on this feature to monitor your AWS usage charges and recurring fees automatically, making it easier to track and manage your spending on AWS. You can set up billing alerts to receive email notifications when your charges reach a specified threshold. Once enabled, this preference cannot be disabled. [Manage Billing Alerts](#)

Receive Billing Reports
Turn on this feature to receive ongoing reports of your AWS charges once or more daily. AWS delivers these reports to the Amazon S3 bucket that you specify where indicated below. For consolidated billing customers, AWS generates reports only for paying accounts. Linked accounts cannot sign up for billing reports.

Save to S3 Bucket:

Now we need to create a custom alarm so that it tells you when you have spent money.

1. Click **CloudWatch** in the AWS Management Console.

Amazon Web Services

- Compute
 - EC2: Virtual Servers in the Cloud
 - Lambda **PREVIEW**: Run Code in Response to Events
- Storage & Content Delivery
 - S3: Scalable Storage in the Cloud
 - Storage Gateway: Integrates On-Premises IT Environments with Cloud Storage
 - Glacier: Archive Storage in the Cloud
 - CloudFront: Global Content Delivery Network
- Databases
 - RDS: MySQL, Postgres, Oracle, SQL Server, and Amazon Aurora
 - DynamoDB: Predictable and Scalable NoSQL Data Store
 - ElasticCache: In-Memory Cache
 - Redshift: Managed Petabyte-Scale Data Warehouse Service
- Networking
 - VPC: Isolated Cloud Resources
 - Direct Connect: Dedicated Network Connection to AWS
 - Route 53: Scalable DNS and Domain Name Registration
- Administration & Security
 - Directory Service: Managed Directories in the Cloud
 - Identity & Access Management: Access Control and Key Management
 - Trusted Advisor: AWS Cloud Optimization Expert
 - CloudTrail: User Activity and Change Tracking
 - Config: Resource Configurations and Inventory
 - CloudWatch: Resource and Application Monitoring**
- Deployment & Management
 - Elastic Beanstalk: AWS Application Container
 - OpsWorks: DevOps Application Management Service
 - CloudFormation: Templated AWS Resource Creation
 - CodeDeploy: Automated Deployments
- Analytics
 - EMR: Managed Hadoop Framework
 - Kinesis: Real-time Processing of Streaming Big Data
 - Data Pipeline: Orchestration for Data-Driven Workflows
- Application Services
 - SQS: Message Queue Service
 - SWF: Workflow Service for Coordinating Application Components
 - AppStream: Low Latency Application Streaming
 - Elastic Transcoder: Easy-to-use Scalable Media Transcoding
 - SES: Email Sending Service
 - CloudSearch: Managed Search Service
- Mobile Services
 - Cognito: User Identity and App Data Synchronization
 - Mobile Analytics: Understand App Usage Data at Scale
 - SNS: Push Notification Service
- Enterprise Applications
 - WorkSpaces: Desktops in the Cloud
 - WorkDocs: Secure Enterprise Storage and Sharing Service
 - WorkMail **PREVIEW**: Secure Email and Calendaring Service

Resource Groups

A resource group is a collection of resources that share one or more tags. Create a group for each project, application, or environment in your account.

Additional Resources

Getting Started
See our documentation to get started and learn more about how to use our services.

AWS Console Mobile App
View your resources on the go with our AWS Console mobile app, available from [Amazon Appstore](#), [Google Play](#), or [iTunes](#).

AWS Marketplace
Find and buy software, launch with 1-Click and pay by the hour.

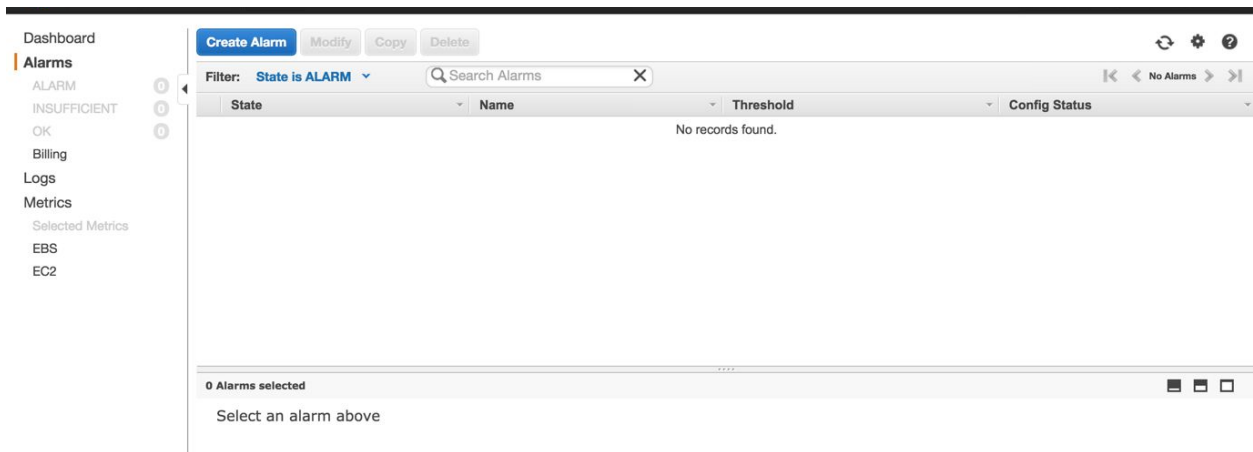
Service Health

All services operating normally.

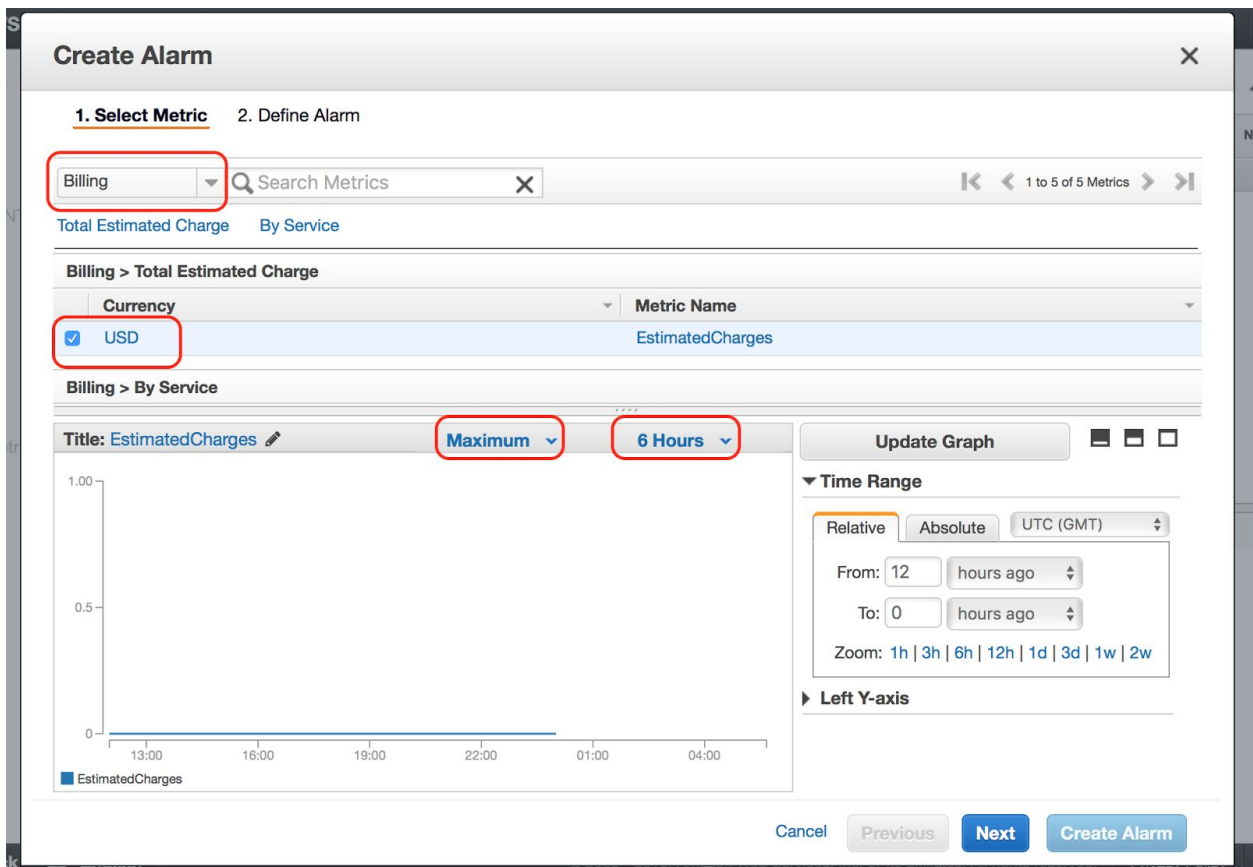
Updated: Feb 22 2015 03:12:00 GMT-0500

[Service Health Dashboard](#)

2. In the navigation pane on the left, click **Alarms**, and then in the **Alarms** pane, click **Create Alarm**.



3. In the **CloudWatch Metrics by Category** pane, under **1. Select Metric**, in the dropdown choose **Billing** and check currency as **USD**. Select **“Maximum”** and **“6 Hours”** in the dropdown as shown in the image below. Click **Next**.



4. Fill out the alarm details and click **New List** next to “Send notification to”:

Alarm Threshold

Provide the details and threshold for your alarm. Use the graph on the right to help set the appropriate threshold.

Name: ←

Description: ←

Whenever charges for: EstimatedCharges

is: ←

Actions

Define what actions are taken when your alarm changes state.

Notification Delete

Whenever this alarm: ↓

Send notification to: ↓ New list

+ Notification + AutoScaling Action + EC2 Action

Enter your name and email.

Actions

Define what actions are taken when your alarm changes state.

Notification Delete

Whenever this alarm: ↓

Send notification to:

Email list:

+ Notification + AutoScaling Action + EC2 Action

You have now created an alert that will bother you when you pass \$50. Consider making another alert which is activated when you use up \$90 so that you do not get charged!

7. Familiarize yourself with S3, EC2 and EMR

We will now run a sample application. We will begin by clicking on the Elastic MapReduce(EMR) link in the Analytics section of the AWS Management Console. This will take you to the EMR Job Flows page. Click on the “Create Cluster” → “Go to the advanced options”. You will be directed to the following steps.

Note:

- Ensure that you first test your code on the smaller dataset. (not larger)
 - Each time you run the code, it may take a couple of hours to terminate.
 - To test and debug your code step by step, refer to the Debugging section at the end of the document. This is highly recommended if you are not familiar with Pig.
1. Under **Step 1 : Software and Steps** , Select only “Hadoop” and “Pig” from the options and unselect others in the Software Configuration options menu. In the “Add Steps”, choose Step Type “Pig Program” and then click “Configure”.

Create Cluster - Advanced Options [Go to quick options](#)

Step 1: Software and Steps
Step 2: Hardware
Step 3: General Cluster Settings
Step 4: Security

Software Configuration

Vendor Amazon MapR

Release

<input checked="" type="checkbox"/> Hadoop 2.7.2	<input type="checkbox"/> Zeppelin 0.6.1	<input type="checkbox"/> Tez 0.8.4
<input type="checkbox"/> Ganglia 3.7.2	<input type="checkbox"/> HBase 1.2.2	<input checked="" type="checkbox"/> Pig 0.16.0
<input type="checkbox"/> Hive 2.1.0	<input type="checkbox"/> Presto 0.150	<input type="checkbox"/> ZooKeeper 3.4.8
<input type="checkbox"/> Sqoop 1.4.6	<input type="checkbox"/> Mahout 0.12.2	<input type="checkbox"/> Hue 3.10.0
<input type="checkbox"/> Phoenix 4.7.0	<input type="checkbox"/> Oozie 4.2.0	<input type="checkbox"/> Spark 2.0.0
<input type="checkbox"/> HCatalog 2.1.0		

Edit software settings (optional)

Enter configuration Load JSON from S3

Add steps (optional)

Step type

Auto-terminate cluster after the last step is completed

1a. Fill the form with details as provided in the box and image below.

Name : (any name)

Script S3 Location : s3://cse6242-<your-username>/pig.txt (must upload the script here)

Input S3 Location: s3://cse6242-2016fall-bigrams-small/* (or big - as provided in the HW)

Output S3 Location : s3://cse6242-<your-username>/output (must be unique)

Action on failure : Terminate (else you may be charged even if the task fails)

Upload your script to an S3 location and select the location of your script from the list of items available at “Script S3 Location”. For the S3 output Location you should specify the bucket and an **additional unique folder for each new run**. It will help with organization. Now, click **Save**.

Add Step

Step type: Pig program

Name: ngram-pig

Script S3 location*: s3://cse6242-<your-username>/pig-small.txt
s3://<bucket-name>/<path-to-file> S3 location of your Pig script.

Input S3 location: s3://cse6242-2016fall-bigrams-small/*
s3://<bucket-name>/<folder>/ S3 location of your Pig input files.

Output S3 location: s3://cse6242-<your-username>/output/
s3://<bucket-name>/<folder>/ S3 location of your Pig output files.

Arguments: Specify optional arguments for your script.

Action on failure: Terminate cluster What to do if the step fails.

[Cancel](#) [Save](#)

2. For **Step 2 : Hardware** configuration, you may see one of the following two views. Modify the EC2 instances as per your needs and select **Next**. (One Master instance and 1-15 Core instances should be sufficient. You may face Bootstrapping errors if you exceed a certain limit of core instances)

View 1 : Using VPC (Virtual Private Cloud)

Create Cluster - Advanced Options [Go to quick options](#)

Step 1: Software and Steps
Step 2: Hardware
 Step 3: General Cluster Settings
 Step 4: Security

Hardware Configuration ⓘ

If you need more than 20 EC2 instances, complete this form.

Network: vpc-bd62d0d9 (172.31.0.0/16) (default) [Create a VPC ⓘ](#)

EC2 Subnet: subnet-42f30634 | Default in us-east-1b

Type	Name	EC2 instance type	Instance count	Storage per instance	Request spot	Bid price
Master	Master instance group - 1	m3.xlarge	1	80 GiB Add EBS volumes	<input type="checkbox"/>	ⓘ
Core	Core instance group - 2	m3.xlarge	2	80 GiB Add EBS volumes	<input type="checkbox"/>	ⓘ
Task	Task instance group - 3	m3.xlarge	0	80 GiB Add EBS volumes	<input type="checkbox"/>	ⓘ

[Add task instance group](#)

[Cancel](#) [Previous](#) [Next](#)

View 2 : Using EC2 - Classic

Hardware Configuration

Specify the [networking](#) and [hardware](#) configuration for your cluster. If you need more than 20 EC2 instances, [complete this form](#).
Request Spot instances (unused EC2 capacity) to save money.

EC2-Classical is fine **Network** Launch into EC2-Classical Use a Virtual Private Cloud (VPC) to process sensitive data or connect to a private network. [Create a VPC](#)

To create a cluster in a VPC, you must first create a VPC. For more information, [click here](#).

You don't need to change this. No preference Launch the cluster in a specific EC2 Availability Zone.

m1.small or m1.medium should be enough

	EC2 instance type	Count	Request spot	
Master	m1.small	1	<input type="checkbox"/>	The Master instance assigns Hadoop tasks to core and task nodes, and monitors their status.
Core	m1.small	2	<input type="checkbox"/>	Core instances run Hadoop tasks and store data using the Hadoop Distributed File System (HDFS).
Task	m1.small	0	<input type="checkbox"/>	Task instances run Hadoop tasks.

You will modify this to speed up computation, you can't use more than 19 Cores

Security and Access

EC2 key pair Proceed without an EC2 key pair Use an existing key pair to SSH into the master node of the Amazon EC2 cluster as the user "hadoop". [Learn more](#)

IAM user access All other IAM users No other IAM users Control the visibility of this cluster to other IAM users. [Learn more](#)

You don't need to change these. IAM role No roles found Control permissions for applications on the cluster. [Learn more](#)

Select your key pair here

Bootstrap Actions

You don't need to change these.

Bootstrap actions are scripts that are executed during setup before Hadoop starts on every cluster node. You can use them to install additional software and customize your applications. [Learn more](#)

Bootstrap action type	Name	S3 location	Optional arguments
Add bootstrap action	Select a bootstrap action		
	Configure and add		

Note: If your account supports only EC2-VPC, you can select the default VPC from the Network list i.e. you will not see "EC2-Classical".

The costs listed in [pricing](#) are charged on an hourly rate, based on the number and type of nodes in your cluster.

3. For **Step 3 : General Cluster Settings**, type a cluster name of your choice, and add the correct path to the logging folder (created in Step 2). Check Logging, Debugging and Termination protection as shown in the image below. Click “Next”.

Create Cluster - Advanced Options [Go to quick options](#)

Step 1: Software and Steps
Step 2: Hardware
Step 3: General Cluster Settings
Step 4: Security

General Options

Cluster name

Logging ⓘ
S3 folder

Debugging ⓘ
 Termination protection ⓘ

Tags ⓘ

Key	Value (optional)
<input type="text" value="Add a key to create a tag"/>	<input type="text"/>

Additional Options

EMRFS consistent view ⓘ

▶ Bootstrap Actions

4. For **Step 4 : Security**, select your keypair and click “Create Cluster” to run the application.

Create Cluster - Advanced Options [Go to quick options](#)

Step 1: Software and Steps
Step 2: Hardware
Step 3: General Cluster Settings
Step 4: Security

Security Options

EC2 key pair

Cluster visible to all IAM users in account ⓘ

Permissions ⓘ

Default Custom
Use default IAM roles. If roles are not present, they will be automatically created for you with managed policies for automatic policy updates.

EMR role

EC2 instance profile

▶ Encryption Options

▶ EC2 Security Groups

5. The cluster must start running as follows,

Amazon EMR

Cluster list
Security configurations
VPC subnets
Help

Cluster: ngram **Starting**

Connections: --
Master public DNS: --
Tags: -- [View All / Edit](#)

Summary	Configuration Details	Network and Hardware	Security and Access
ID: j-2M320H9DWE6H Creation date: 2016-10-17 02:41 (UTC-4) Elapsed time: 3 minutes Auto-terminate: Yes Termination On protection: Change	Release label: emr-5.0.0 Hadoop: Amazon 2.7.2 distribution: Applications: Pig 0.16.0 Log URI: s3://cse6242-mnatraj3-logging-task4/ EMRFS Disabled consistent view:	Availability zone: -- Subnet ID: subnet-42f06834 Master: Provisioning 1 m3.xlarge Core: Provisioning 3 m3.xlarge Task: --	Key name: cse6242-mnatraj3-keypair EC2 instance profile: EMR_EC2_DefaultRole EMR role: EMR_DefaultRole Visible to all users: Change Security groups for Master: sg-ea868890 (ElasticMapReduce-master) Security groups for Core & Task: sg-eb868891 (ElasticMapReduce-slave)

Monitoring
Hardware
Steps
Configurations
Bootstrap Actions

You now can view the status of your application in this “Cluster Details” screen. It takes several minutes for the whole process to run.

Provisioning - Amazon locates resources for your application

Bootstrapping - Amazon sets up and configures the nodes to run your application

Running - Runs and writes to your output bucket.

Terminating - Amazon deconstructs the setups you used for the application

You can track its progress once it’s been created.

After the application terminates, you could go back to the S3 output bucket you chose. The results will be written to the output folder. You should have several partxxxx files in the output folder. These are texts of the output! You have just successfully completed a MapReduce job flow on AWS and are ready for large scale data analytics.

8. Debugging

A very important part of running Pig Scripts on AWS is the ability to also run your code directly on the master node. You can run your script step by step and identify the exact step where an error occurred. The steps to debug are given below.

1. You must repeat all the steps in Section 7, except with three modifications:
 - a. Ensure that you verify the script location, its input and output path. Do this each time you create/clone a cluster (many students make a mistake here)
 - b. Modify the action on failure option to “Continue”
 - c. Uncheck the “Auto-terminate cluster after....” option.

Warning : You must revert back these changes after debugging else you may leave the clusters running forever and you will be charged for this.

Add Step
✕

Step type Pig program

Name

Script S3 location* S3 location of your Pig script.
s3://<bucket-name>/<path-to-file>

Input S3 location S3 location of your Pig input files.
s3://<bucket-name>/<folder>/

Output S3 location S3 location of your Pig output files.
s3://<bucket-name>/<folder>/

Arguments Specify optional arguments for your script.

Action on failure Continue What to do if the step fails.

Cancel Save

Create Cluster - Advanced Options [Go to quick options](#)

Step 1: Software and Steps

- Step 2: Hardware
- Step 3: General Cluster Settings
- Step 4: Security

Software Configuration

Vendor Amazon MapR

Release

Hadoop 2.7.2

Zeppelin 0.6.1

Tez 0.8.4

Ganglia 3.7.2

HBase 1.2.2

Pig 0.16.0

Hive 2.1.0

Presto 0.150

ZooKeeper 3.4.8

Sqoop 1.4.6

Mahout 0.12.2

Hue 3.10.0

Phoenix 4.7.0

Oozie 4.2.0

Spark 2.0.0

HCatalog 2.1.0

Edit software settings (optional)

Enter configuration Load JSON from S3

Add steps (optional)

Name	Action on failure	JAR location	Arguments
ngram-pig	Continue	command-runner.jar	<pre> pig-script --run-pig-script -- pig-versions 0.16.0 --args -f s3://cse6242-2016fall- bigrams-small/pig/pig- small.txt -p INPUT=s3://cse6242- 2016fall-bigrams-small/* -p OUTPUT=s3://cse6242- 2016fall-bigrams-small- pig/output/ </pre>

Step type Select a step Configure

Auto-terminate cluster after the last step is completed

Cancel Next

2. Once the cluster is running, you can open the TCP Port of your Master node to allow SSH connections. Click on the security group of your master node.

The screenshot shows the Amazon EMR console for a cluster named 'ngram' in a 'Terminated' state. The 'Security and Access' section is highlighted, showing the 'Security sg-ea868890 groups for (ElasticMapReduce-Master: master)' entry circled in red. Other details include the cluster ID, creation and end dates, and various configuration parameters like Hadoop distribution and applications.

Add an entry for SSH in the inbound tab of your master node with the exact details as follows.

The screenshot shows the 'Create Security Group' dialog in the Amazon EC2 console. The 'Inbound' tab is selected, and a table of rules is displayed. The 'SSH' rule is circled in red, showing its configuration: Type: SSH, Protocol: TCP, Port Range: 22, and Source: 0.0.0.0/0. The table below shows the full list of rules for the security group.

Type	Protocol	Port Range	Source
All TCP	TCP	0 - 65535	sg-ea868890 (ElasticMapReduce-master)
All TCP	TCP	0 - 65535	sg-eb868891 (ElasticMapReduce-slave)
SSH	TCP	22	0.0.0.0/0
Custom TCP Rule	TCP	8443	207.171.112.0/32
Custom TCP Rule	TCP	8443	54.239.98.0/24
Custom TCP Rule	TCP	8443	54.240.217.8/29
Custom TCP Rule	TCP	8443	207.171.167.26/32
Custom TCP Rule	TCP	8443	72.21.198.64/29
Custom TCP Rule	TCP	8443	207.171.167.101/32
Custom TCP Rule	TCP	8443	72.21.196.64/29
Custom TCP Rule	TCP	8443	54.240.217.80/29
Custom TCP Rule	TCP	8443	54.240.217.16/29
Custom TCP Rule	TCP	8443	207.171.167.25/32
Custom TCP Rule	TCP	8443	72.21.217.0/24
Custom TCP Rule	TCP	8443	54.240.217.64/28
All UDP	UDP	0 - 65535	sg-ea868890 (ElasticMapReduce-master)
All UDP	UDP	0 - 65535	sg-eb868891 (ElasticMapReduce-slave)
All ICMP	All	N/A	sg-ea868890 (ElasticMapReduce-master)
All ICMP	All	N/A	sg-eb868891 (ElasticMapReduce-slave)

You can now SSH into your master node.

3. To SSH, first copy the command as follows.

The screenshot shows the Amazon EMR console interface. On the left is a navigation menu with 'Cluster list' selected. The main content area displays details for a cluster named 'ngram' which is in a 'Terminated' state. The 'Connections' section shows 'Master public DNS' as 'ec2-54-208-56-158.compute-1.amazonaws.com' with an 'SSH' button next to it. Below this are sections for 'Summary', 'Configuration Details', and 'Network and Hardware'. The 'Summary' section includes ID, creation and end dates, and elapsed time. 'Configuration Details' lists release label, Hadoop distribution, and applications. 'Network and Hardware' shows availability zone and instance types. A 'Security and Access' section at the bottom provides key name, EC2 instance profile, and EMR role information.

Below the console screenshot is a 'SSH' dialog box titled 'Connect to the Master Node Using SSH'. It provides instructions on how to connect to the master node. A terminal window is shown with the following command:

```
ssh -i ~/cse6242-mnatraj3-keypair.pem hadoop@ec2-54-208-56-158.compute-1.amazonaws.com
```

Modify the path to your .pem file and run the command on your terminal. (ensure that the file permissions of your .pem file is set to 400).

4. You will now be logged into the master node. Type **pig** to be able to run commands on the pig shell.

5. Run your code line by line and spot the errors!