Amazon Web Services Tutorial
10-605: Machine Learning for Large Datasets

Abhinav Maurya
Carnegie Mellon University

www.andrew.cmu.edu/user/amaurya
AWS Login

AWS Management Console
Access and manage Amazon Web Services through a simple and intuitive web-based user interface. You can also use the AWS Console mobile app to quickly view resources on the go.

Features
Administer your AWS account
The Console facilitates cloud management for all aspects of your AWS account, including monitoring your monthly spending by service, managing security credentials, or even setting up new IAM Users.

Manage AWS resources from your mobile device
With the AWS Console mobile app, you can quickly and easily view your existing resources, including CloudWatch alarms, and perform operational tasks from your mobile device. Download our mobile app from Amazon Appstore, Google Play, or iTunes.

Resource Groups
With Resource Groups, you can view collections of resources that share common tags. Streamline your use of the console by creating a resource group for each application, service, or collection of related resources that you work with regularly.

Quickly navigate to each saved resource group using the "AWS" menu. Resource Groups are specific to each identity, so each user in an account can create unique Resource Groups for frequently accessed resources and common tasks. Users can also use a URL to share Resource Group definitions with others in the same account.

Tag Editor

http://aws.amazon.com/console
AWS Console

https://console.aws.amazon.com/console/home?region=us-east-1
AWS Services

Big data on AWS

Collect
- AWS Direct Connect
- AWS Import/Export
- Amazon Kinesis

Store
- Amazon S3
- Amazon DynamoDB
- Amazon Glacier

Process & Analyze
- Amazon EMR
- Amazon Redshift
- Amazon EC2

Automate
- AWS Data Pipeline

From Your First Big Data Application on AWS @ AWS re:Invent 2014
AWS SSH Key Pair Generation

https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#KeyPairs
**AWS SSH Key Pair Generation**

Creation will download SSH key pair `amazon-key-pair.pem` to your system.

### Amazon Key Pairs

<table>
<thead>
<tr>
<th>Key pair name</th>
<th>Fingerprint</th>
</tr>
</thead>
<tbody>
<tr>
<td>amazon-key-pair</td>
<td></td>
</tr>
</tbody>
</table>

[https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#KeyPairs](https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#KeyPairs)
Launching Amazon EC2 Instance

https://console.aws.amazon.com/ec2/v2/home?region=us-east-1
Launching Amazon EC2 Instance

https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard:
## Launching Amazon EC2 Instance

### Step 2: Choose an Instance Type

#### Filter by:
- All instance types
- All generations

Currently selected: m3.large (6.5 ECUs, 2 vCPUs, 2.5 GHz, Intel Xeon E5-2670v2, 7.5 GiB memory, 1 x 32 GiB Storage Capacity)

<table>
<thead>
<tr>
<th>Family</th>
<th>Type</th>
<th>vCPUs</th>
<th>Memory (GiB)</th>
<th>Instance Storage (GB)</th>
<th>EBS-Optimized Available</th>
<th>Network Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Micro instances</td>
<td>t1.micro</td>
<td>1</td>
<td>0.613</td>
<td>EBS only</td>
<td>-</td>
<td>Very Low</td>
</tr>
<tr>
<td>General purpose</td>
<td>t2.micro</td>
<td>1</td>
<td>1</td>
<td>EBS only</td>
<td>-</td>
<td>Low to Moderate</td>
</tr>
<tr>
<td>General purpose</td>
<td>t2.small</td>
<td>1</td>
<td>2</td>
<td>EBS only</td>
<td>-</td>
<td>Low to Moderate</td>
</tr>
<tr>
<td>General purpose</td>
<td>t2.medium</td>
<td>2</td>
<td>4</td>
<td>EBS only</td>
<td>-</td>
<td>Low to Moderate</td>
</tr>
<tr>
<td>General purpose</td>
<td>m3.medium</td>
<td>1</td>
<td>3.75</td>
<td>1 x 4 (SSD)</td>
<td>-</td>
<td>Moderate</td>
</tr>
<tr>
<td>General purpose</td>
<td>m3.large</td>
<td>2</td>
<td>7.5</td>
<td>1 x 32 (SSD)</td>
<td>-</td>
<td>Moderate</td>
</tr>
<tr>
<td>General purpose</td>
<td>m3.xlarge</td>
<td>4</td>
<td>15</td>
<td>2 x 40 (SSD)</td>
<td>Yes</td>
<td>High</td>
</tr>
<tr>
<td>General purpose</td>
<td>m3.2xlarge</td>
<td>8</td>
<td>30</td>
<td>2 x 80 (SSD)</td>
<td>Yes</td>
<td>High</td>
</tr>
</tbody>
</table>

[Next: Configure Instance Details]
Launching Amazon EC2 Instance

Step 3: Configure Instance Details
Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

- **Number of instances**: 1
- **Purchasing option**: Request Spot Instances
- **Network**: vpc-e8a8308c (172.31.0.0/16) (default)
- **Subnet**: No preference (default subnet in any Availability)
- **Auto-assign Public IP**: Use subnet setting (Enable)
- **IAM role**: None
- **Shutdown behavior**: Stop
- **Enable termination protection**: Protect against accidental termination
- **Monitoring**: Enable CloudWatch detailed monitoring
  
  Additional charges apply.
- **Tenancy**: Shared tenancy (multi-tenant hardware)
  
  Additional charges will apply for dedicated tenancy.

Next: Add Storage
Launching Amazon EC2 Instance

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. Learn more about storage options in Amazon EC2.

<table>
<thead>
<tr>
<th>Type</th>
<th>Device</th>
<th>Snapshot</th>
<th>Size (GiB)</th>
<th>Volume Type</th>
<th>IOPS</th>
<th>Delete on Termination</th>
<th>Encrypted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root</td>
<td>/dev/sda1</td>
<td>snap-0c975e6b</td>
<td>15</td>
<td>Magnetic</td>
<td>$</td>
<td>✓</td>
<td>Not Encrypted</td>
</tr>
<tr>
<td>EBS</td>
<td>/dev/sdf</td>
<td>snap-08975e6f</td>
<td>20</td>
<td>Magnetic</td>
<td>$</td>
<td>✓</td>
<td>Not Encrypted</td>
</tr>
<tr>
<td>Instance Store 0</td>
<td>/dev/sdb</td>
<td>N/A</td>
<td>N/A</td>
<td>Magnetic</td>
<td>N/A</td>
<td>N/A</td>
<td>Not Encrypted</td>
</tr>
</tbody>
</table>

Add New Volume

General Purpose (SSD) volumes provide the ability to burst to 3,000 IOPS per volume, independent of volume size, to meet the performance needs of most applications and also deliver a consistent baseline of 3 IOPS/GiB. Set my root volume to General Purpose (SSD).

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Learn more about free usage tier eligibility and usage restrictions.

[Review and Launch]
Launching Amazon EC2 Instance

Step 7: Review Instance Launch

AMI Details

```
cloudera-maverick-64 - ami-dfa7fbd6
Hbase Hive Hadoop on Maverick 64 bit
Root Device Type: ebs  Virtualization type: paravirtual
```

Instance Type

<table>
<thead>
<tr>
<th>Instance Type</th>
<th>ECUs</th>
<th>vCPUs</th>
<th>Memory (GiB)</th>
<th>Instance Storage (GB)</th>
<th>EBS-Optimized Available</th>
<th>Network Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>m3.large</td>
<td>6.5</td>
<td>2</td>
<td>7.5</td>
<td>1 x 32</td>
<td>-</td>
<td>Moderate</td>
</tr>
</tbody>
</table>

Security Groups

- Security group name: launch-wizard-1
  - Description: launch-wizard-1 created 2015-02-03T00:49:50.721-05:00

<table>
<thead>
<tr>
<th>Type</th>
<th>Protocol</th>
<th>Port Range</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSH</td>
<td>TCP</td>
<td>22</td>
<td>0.0.0.0/0</td>
</tr>
</tbody>
</table>
Launching Amazon EC2 Instance

Step 7: Review Instance Launch

- AMI Details
  - cloudera-maverick-64 - apache Hbase Hive Hadoop on Maverick
  - Root Device Type: ebs

- Instance Type
  - Instance Type: m3.large
  - ECUs: 6.5

- Security Groups
  - Security Group Name: launch-wizard-1
    - Description: launched with wizard
    - Type: SSH

- Instance Details

Select an existing key pair or create a new key pair

A key pair consists of a public key that AWS stores, and a private key file that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

- Choose an existing key pair
  - Select a key pair
    - amazon-key-pair

- I acknowledge that I have access to the selected private key file (amazon-key-pair.pem), and that without this file, I won't be able to log into my instance.

Cancel  Launch Instances

© 2008 - 2015, Amazon Web Services, Inc. or its affiliates. All rights reserved.  Privacy Policy  Terms of Use

Carnegie Mellon University
Launching Amazon EC2 Instance

Launch Status

Your instances are now launching
The following instance launches have been initiated: i-33ef82c2
View launch log

Get notified of estimated charges
Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the running state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click View Instances to monitor your instances’ status. Once your instances are in the running state, you can connect to them from the Instances screen. Find out how to connect to your instances.

Here are some helpful resources to get you started

- How to connect to your Linux instance
- Learn about AWS Free Usage Tier
- Amazon EC2: User Guide
- Amazon EC2: Discussion Forum

While your Instances are launching you can also

Create status check alarms to be notified when these instances fail status checks. (Additional charges may apply)
Create and attach additional EBS volumes (Additional charges may apply)
Manage security groups

© 2008 - 2015, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Amazon EC2 Instances Dashboard
Use Ubuntu instead of root
Connect to Amazon EC2 Instance using ssh

```bash
$ cd aws
$ ssh -i amazon-key-pair.pem ubuntu@54.165.125.249
$ which hadoop
$hadoop
```
AWS IAM User Creation
AWS IAM Group Creation

https://console.aws.amazon.com/iam/home?#groups
## AWS Access Key Generation

### AWS Console Screen

- **Dashboard**
- **Services**
- **Create New Users**
- **User Actions**

<table>
<thead>
<tr>
<th>User Name</th>
<th>Groups</th>
<th>Password</th>
<th>Password Last Used</th>
<th>Access Keys</th>
<th>Creation Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>ahmaurya</td>
<td>1</td>
<td>N/A</td>
<td>N/A</td>
<td>1 active</td>
<td>2015-02-03 02:19 EST</td>
</tr>
</tbody>
</table>

### Manage Access Keys

[![Manage Access Keys](https://i.imgur.com/3G6zG.png)](https://i.imgur.com/3G6zG.png)

- **Created**: 2015-02-03 02:19 EST
- **Access Key ID**: [Redacted]
- **Status**: Active (Make Inactive | Delete)

**Note:**
- For your protection, you should never share your secret keys with anyone. In addition, industry best practice recommends frequent key rotation.
- Learn more about Access Keys

[https://console.aws.amazon.com/iam/home?#users](https://console.aws.amazon.com/iam/home?#users)
$sudo pip install awscli
$aws configure
>>>AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
>>>AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
>>>Default region name [None]: us-east-1
>>>Default output format [None]: json

AWS CLI: Create Security Group

```bash
$aws ec2 create-security-group --group-name devenv-sg --description "security group for development environment in EC2"
{
    "GroupId": "sg-b018ced5"
}
$aws ec2 authorize-security-group-ingress --group-name devenv-sg --protocol tcp --port 22 --cidr 0.0.0.0/0
$aws ec2 describe-security-groups
```
AWS CLI: Create SSH Key Pair

```bash
$aws ec2 create-key-pair --key-name devenv-key --query 'KeyMaterial' --output text > devenv-key.pem
$chmod 400 devenv-key.pem
$aws ec2 describe-key-pairs
```
$aws ec2 run-instances --image-id ami-076e6542 --count 1 --instance-type t2.micro --key-name devenv-key --security-groups devenv-sg --query 'Instances[0].InstanceId' "i-ec3e1e2k"

$aws ec2 describe-instances --instance-ids i-ec3e1e2k --query 'Reservations[0].Instances[0].PublicIpAddress' "54.183.22.255"

$ssh -i devenv-key.pem ubuntu@54.183.22.255
AWS EC2 CLI Tools: Installation
(Alternative to AWS CLI Tools)

```bash
$ # download ec2-api-tools.zip
$ curl -O http://s3.amazonaws.com/ec2-downloads/ec2-api-tools.zip
$ # unzip ec2-api-tools.zip
$ sudo mkdir /usr/local/ec2
$ sudo unzip ec2-api-tools.zip -d /usr/local/ec2
$ # setup ec2 api tools in your $PATH
$ export EC2_HOME=/usr/local/ec2/ec2-api-tools-1.7.0.0
$ export PATH=$PATH:EC2_HOME/bin
$ # test if the tools work
$ ec2-describe-regions
>>>REGION us-east-1 ec2.us-east-1.amazonaws.com
>>>REGION us-west-1 ec2.us-west-1.amazonaws.com
>>>REGION us-west-2 ec2.us-west-2.amazonaws.com
```
https://console.aws.amazon.com/s3/
AWS S3

Accessed using
s3://ahmaurya-emr-bucket/output

https://console.aws.amazon.com/s3/
AWS Elastic MapReduce

https://console.aws.amazon.com/elasticmapreduce/
AWS Elastic MapReduce

https://console.aws.amazon.com/elasticmapreduce/