CS 4001/7001 Cloud Computing
Spring 2014
Lab # 2 – AWS Resource Discovery and Instance Setup

Dr. Prasad Calyam & Ronny Bazan Antequera (Contact: calyamp@missouri.edu)

1. Purpose of the Lab

Use the AWS account for the discovery, reservation and access of virtual compute/storage infrastructure instances; setup a platform along with web-application and related basic firewall configurations within your reserved infrastructure resources.

2. References to guide Lab work

- Architecture for the Cloud: Best Practices
  - http://media.amazonwebservices.com/AWS_Cloud_Best_Practices.pdf
- Amazon EC2 Linux Instances
  - http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html
- *Chapters 1 and 2, Programming Amazon EC2* – Jurg van Vliet & Flavia Paganelli (On line copy available at MU Library)
- *Chapters 3 and 4, Distributed and Cloud Computing,* Hwang, Fox & Dongarra
- Connecting to your Instance – *(ordered by priority)*
  1. Linux SSH/SCP Client - http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstancesLinux.html
  2. Windows - http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html
     - Putty - http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
     - Transfer files from Windows to AWS Instance using PSCP: http://the.earth.li/~sgtatham/putty/0.62/htmldoc/Chapter5.html#pscp

3. Lab Steps and output collection guidelines

To complete this Lab, you need to perform the following five steps:

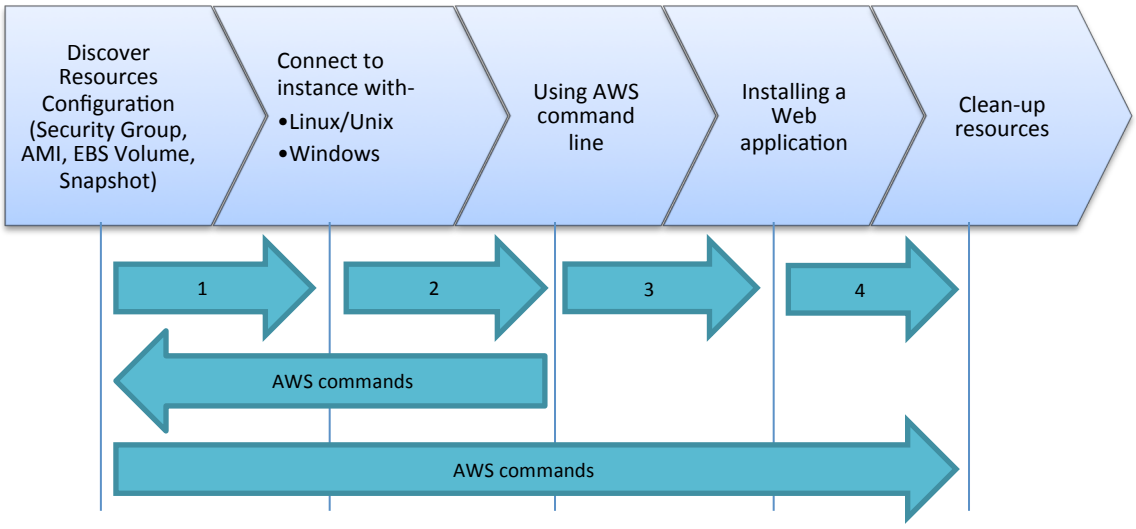


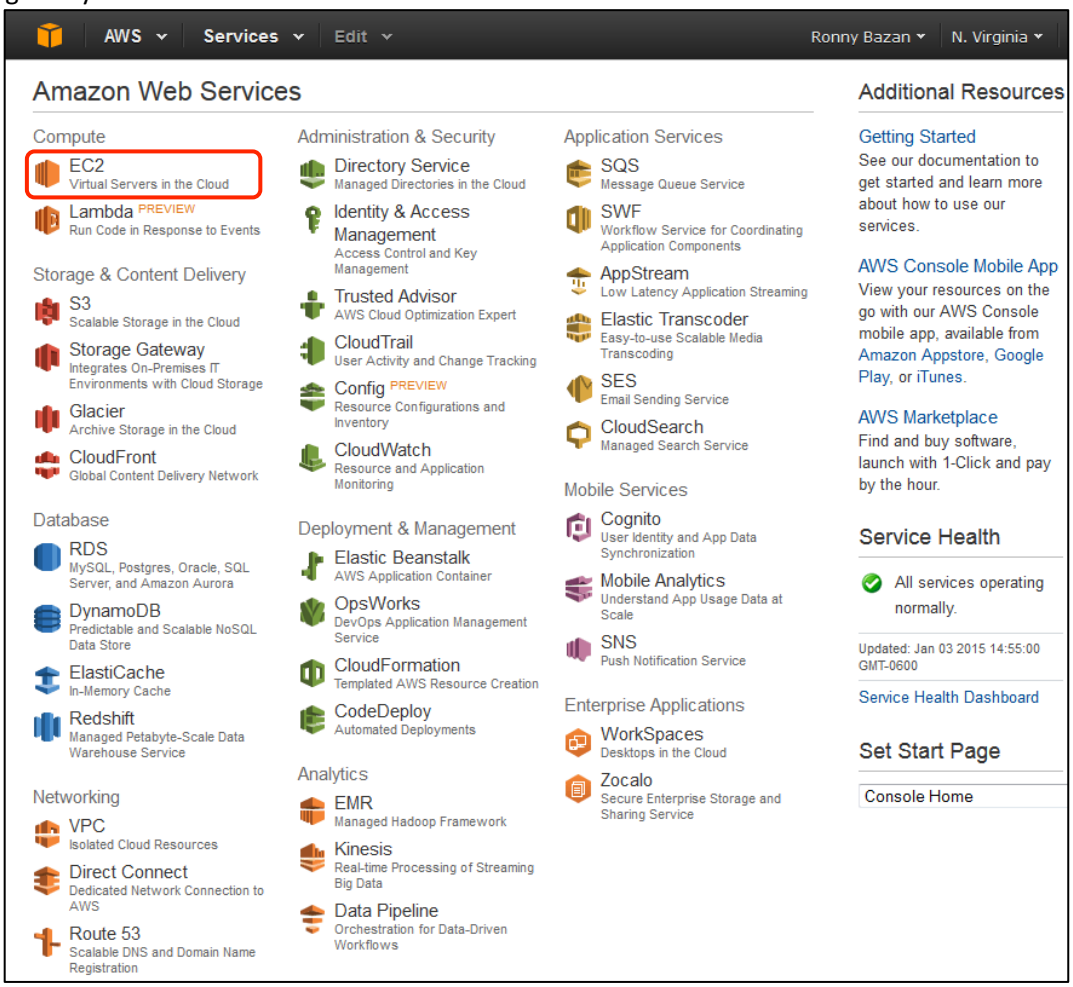
Figure 1: Lab 2 Instructions Roadmap

In the first step, you will launch the instance created in Lab 1 within your AWS account and discover the multiple options to customize your infrastructure. The second step will show you two different ways to connect to your instance through local Linux/Unix or Windows computer. The third step will focus on the configuration of your aws-cli (AWS command line interface) environment so that you can test all the aws-commands that are in the [ ] shown in Steps 1 - 5. Next, in the fourth step you will configure a web application to show your instance-specific information (metadata) on your web browser. Finally, you will clean up resources used for this Lab.

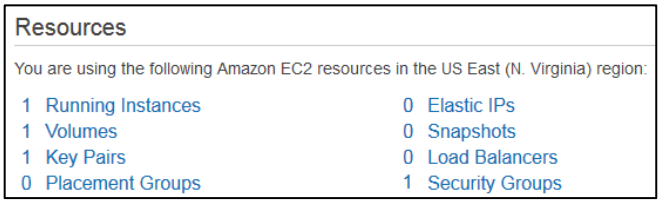
Let's get started!

3.1 Discovery Resources.

3.1.1 Login to your AWS account and select the EC2 service.

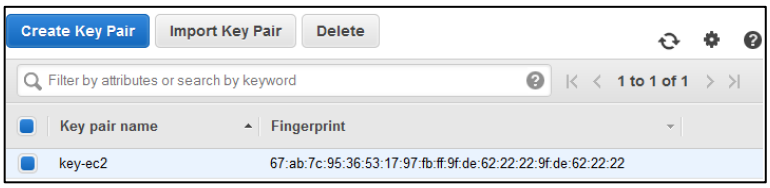


- You will see your resources for your zone (N. Virginia).



Important: Most of the configuration done in Lab 1 can also be performed through the AWS command line interface aws-cli. This configuration will be setup in Step 3.3. So, once you reach that step you can come back and test commands that are in this kind of boxes.

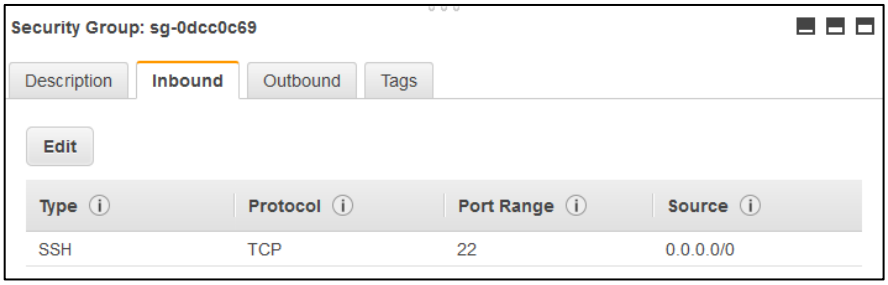
3.1.2 Verify that your Key Pair is created, click on 'Key Pair' option in the lef menu.



Create a key pair
#ec2-add-keypair key-ecc1

Get information of your key-pairs
#ec2-describe-keypairs

3.1.3 Add a 'HTTP' to your 'ec2-sg' security group.



Create a security group called 'ec2-sg2'

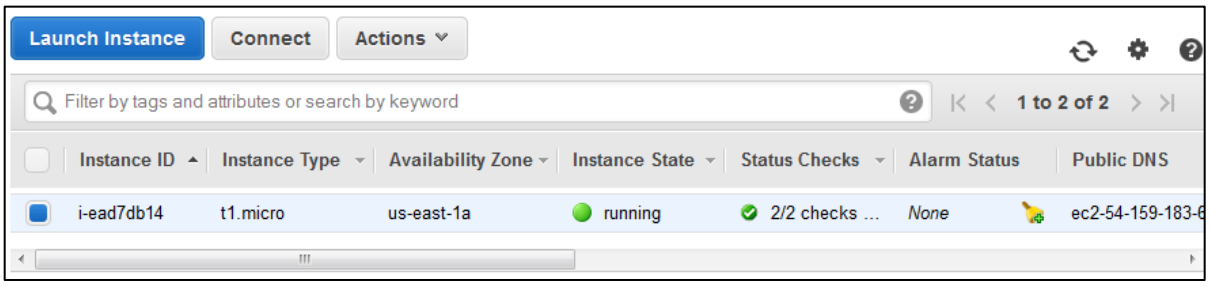
#ec2-add-group ec2-sg2 -d 'HTTP-SSH' – [VPC_ID]
#ec2-authorize ec2-sg2 -P tcp -p 22 -s 0.0.0.0/0
#ec2-authorize ec2-sg2 -P tcp -p 80 -s 0.0.0.0/0

Ref: d Description, P Protocol, p Port, s Source
VPC_ID can be found in VPC service.

Get information of Security Group
#ec2-describe-group

Remove a Security Group
#ec2-delete-group ec2-g2

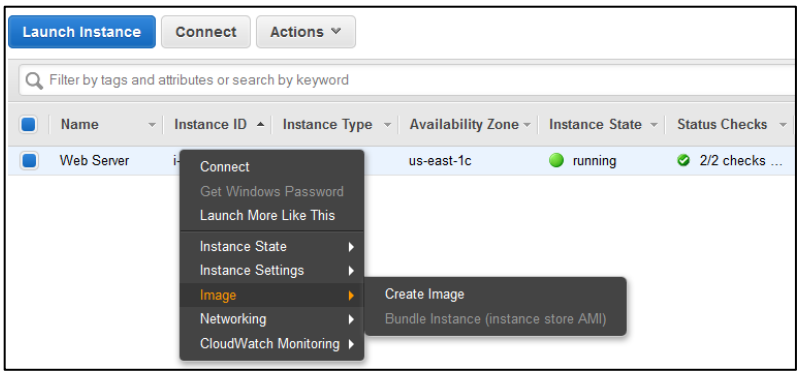
3.1.4 Start the instance created in Lab 1. (Right click over the instance and select 'Start')



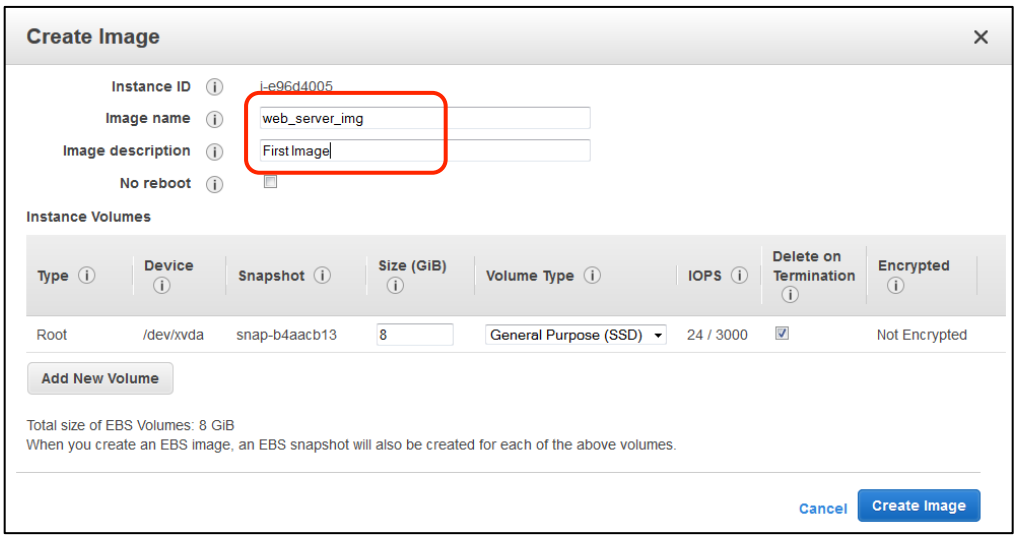
Get group id attribute information
 # ec2-describe-instance-attribute i-XXXXXX --instance-type

Get instances information
#ec2-describe-instances
#ec2-describe-instance-status

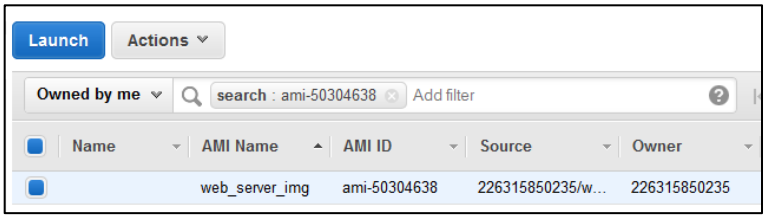
3.1.5 Create an Amazon Machine Image from a running instance. Select your instance, and in 'Actions' button select 'Create Image (EBS AMI)'.



- Give Image Name: 'web_server_img' and Image Description: 'First Image'.



- Click on 'AMIs' in the left menu under IMAGES to see an available AMI for future deployment.



- The EBS AMI will include any volume attached to your instance.

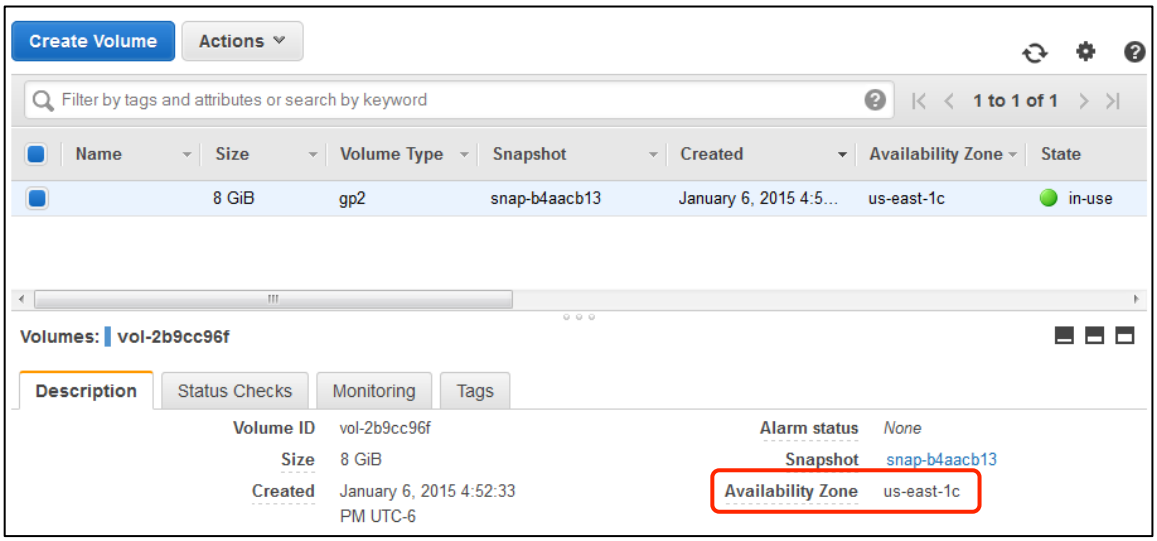
- Create a new instance.

Launch new instances from an AMI Image

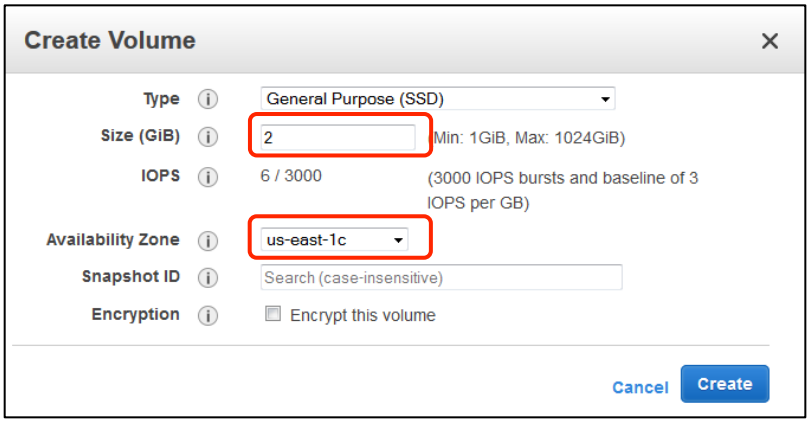
```
#ec2-run-instances ami-XXXXXXX      \
    --instance-type t1.micro        \
    --region us-east-1              \
    --instance-count 1              \
    --key key-ec2                   \
    --group SG_EC2
```

Note: You can execute a command with multiple lines adding "\" at the end of each line.

3.1.6 Volumes. In left menu "Elastic Block Store", select 'Volumes' and you will see the default volume where your instance is stored.



- Click in 'Create Volume' for creating a new **2GB** volume **and Volume Type 'General Purpose (SSD)'**. Be sure that the '**Availability Zone**' in the new volume is the same as the 'Zone' of the first volume (For this specific case us-east-1c)

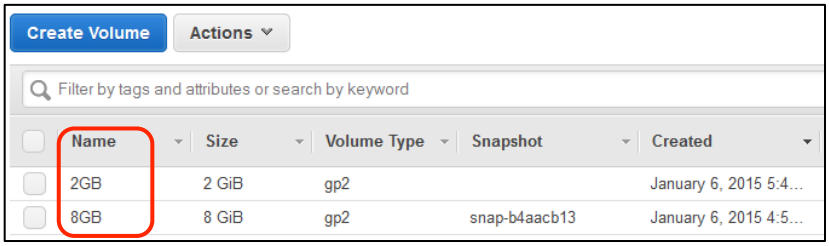


Create a new volume of 2 GB in us-east zone
#ec2-create-volume --size 2 --availability-zone us-east-1x

- Once the new volume is 'available', right click on it and select 'Attach volume', select your created instance and attach it. Note the path will be established in the 'Device' option.

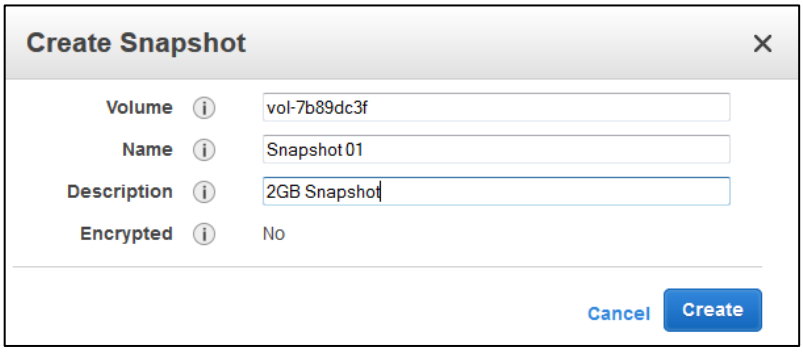
Attach volume
#ec2-attach-volume vol-c7712683 --instance i-db9eb337 --device /dev/sdf

- You will end up with a new 2 GB SDD storage drive attached to your instance.
- It is good practice to name your volumes.



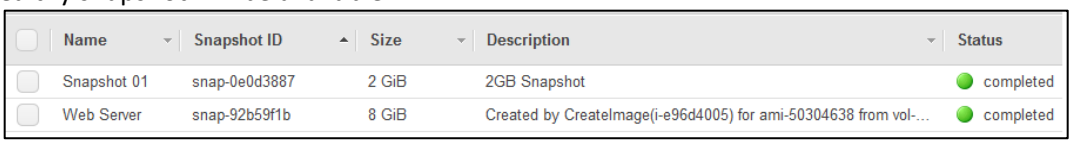
3.1.7 Snapshots. Under 'Elastic Block Store' options click on 'Snapshots' to create one, this will enable a recover point for restoring your instance.

- Select the 2GB volume created previously and create the Snapshot.



Create a snapshot
#ec2-create-snapshot vol-xxxxxxxx --description "Snapshot 2GB volume"

- A healthy snapshot will be available.



Name	Snapshot ID	Size	Description	Status
Snapshot 01	snap-0e0d3887	2 GiB	2GB Snapshot	completed
Web Server	snap-92b59f1b	8 GiB	Created by CreateImage(i-e96d4005) for ami-50304638 from vol-...	completed

Get snapshot information
#ec2-describe-snapshots

3.2 Connection to the instance using SSH

3.2.1 OPTION 1. Connect to the Instance through Linux/Unix. (Make sure to use an account with adequate root/admin privilege or use 'sudo' command before to execute a command whenever you need it).

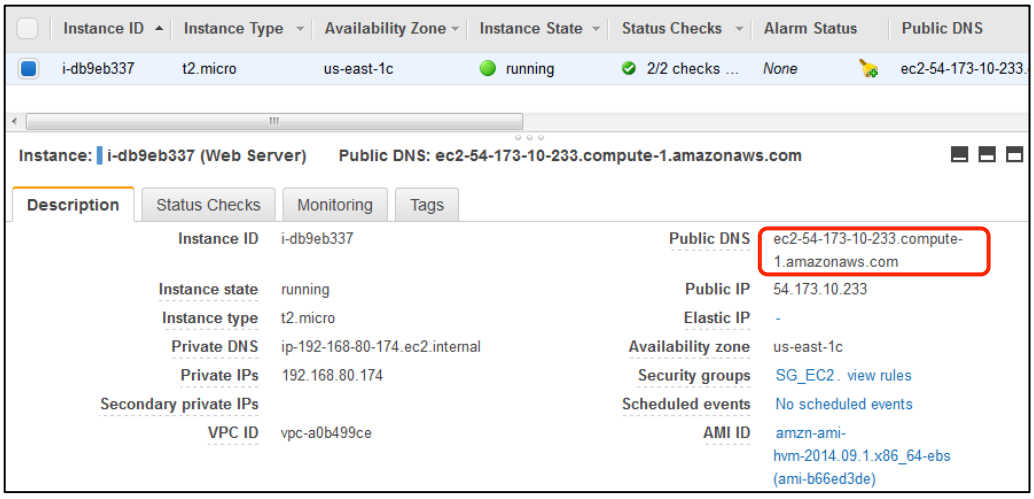
- Create a work folder /opt/aws
  #mkdir -p /opt/aws

- Copy the key 'key-ec2.pem' downloaded in AWS-Lab1 Step 3.5.2 inside /opt/aws directory.

```
root@ubuntu:/# ls -l key-ec2.pem
-rwxr--r-- 1 root root 1692 Jul 24 12:32 key-ec2.pem
root@ubuntu:/# chmod 400 key-ec2.pem
root@ubuntu:/# ls -l key-ec2.pem
-r-------- 1 root root 1692 Jul 24 12:32 key-ec2.pem
root@ubuntu:/#
```

- ssh to the instance with the 'ec2-user' default user and <u>your</u> [public dns].

```
$ ssh -i key-ec2.pem ec2-user@ec2-54-173-10-233.compute-1.amazonaws.com
```



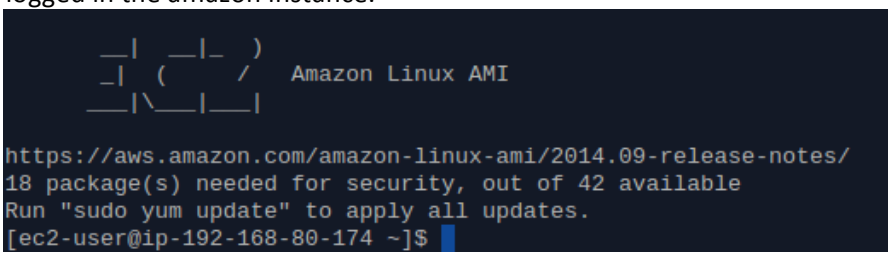
Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS
i-db9eb337	t2.micro	us-east-1c	running	2/2 checks ...	None	ec2-54-173-10-233.

Instance: i-db9eb337 (Web Server) Public DNS: ec2-54-173-10-233.compute-1.amazonaws.com

Description | Status Checks | Monitoring | Tags

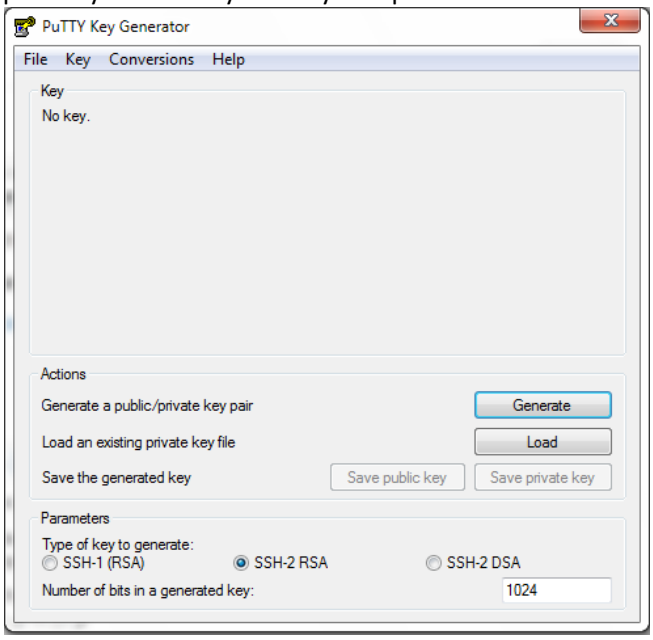
Instance ID	i-db9eb337	Public DNS	ec2-54-173-10-233.compute-1.amazonaws.com
Instance state	running	Public IP	54.173.10.233
Instance type	t2.micro	Elastic IP	-
Private DNS	ip-192-168-80-174.ec2.internal	Availability zone	us-east-1c
Private IPs	192.168.80.174	Security groups	SG_EC2. view rules
Secondary private IPs		Scheduled events	No scheduled events
VPC ID	vpc-a0b499ce	AMI ID	amzn-ami-hvm-2014.09.1.x86_64-ebs (ami-b66ed3de)

- You will be logged in the amazon instance.

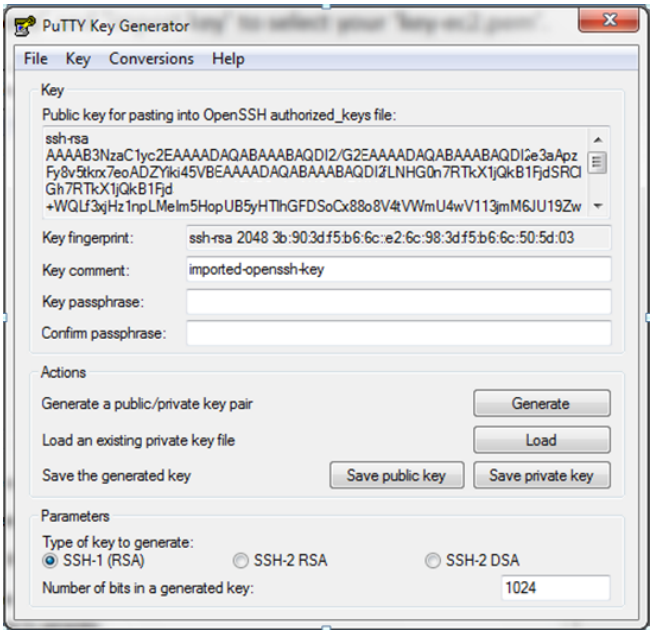


3.2.2 OPTION 2. Connect to the Instance through Windows

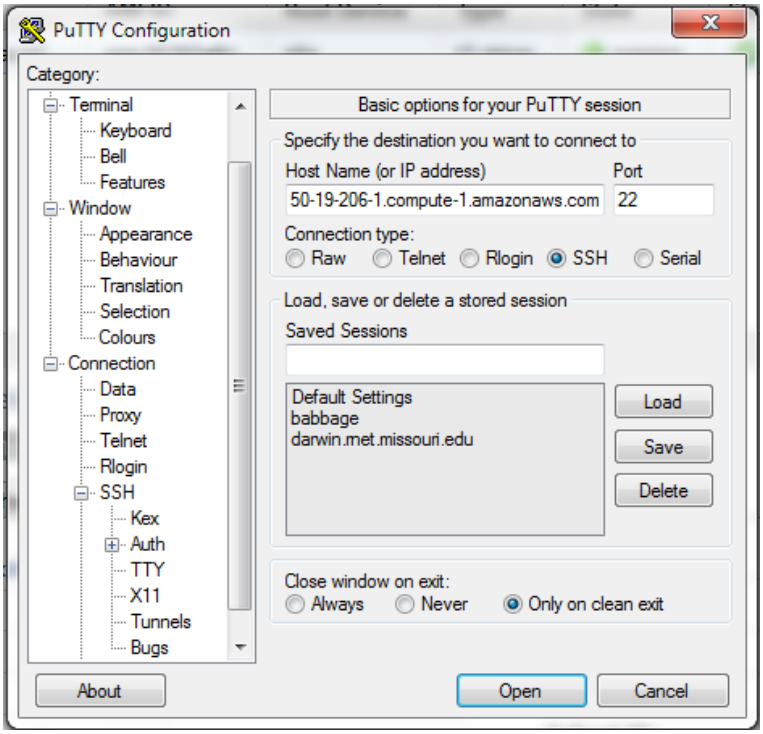
- Download 'PuTTY Key Generator' to convert your key to PuTTY compatible format. Click in 'conversions' and 'import key' to select your 'key-ec2.pem'.



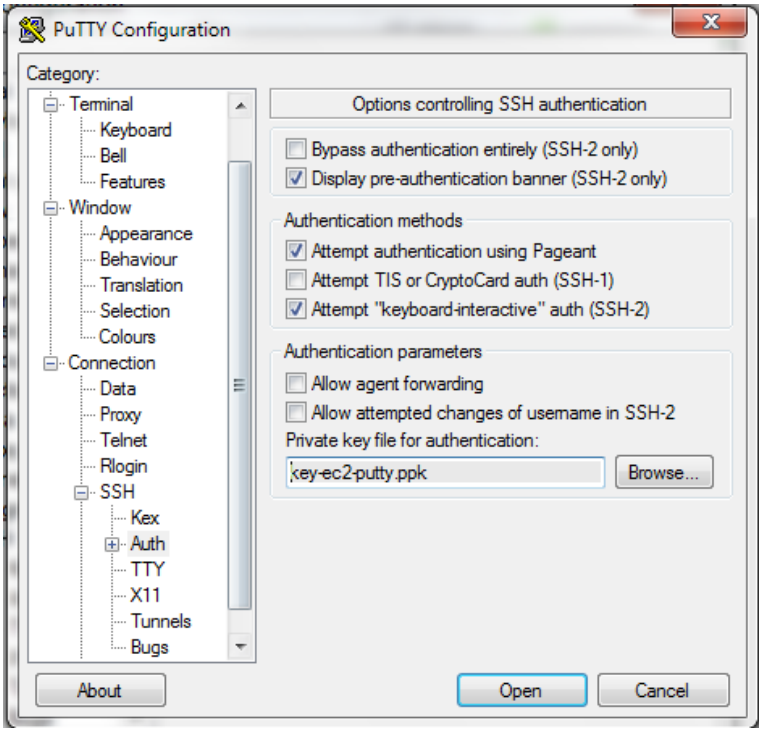
- Select the 'SSH-1 (RSA)' check box and click on 'Save private key' for storing. Name it 'key-ec2-putty'



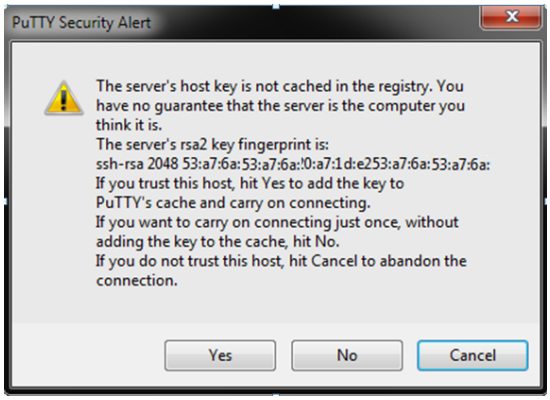
- Now with PuTTY. Paste [public dns] in 'Host Name (or IP address)'

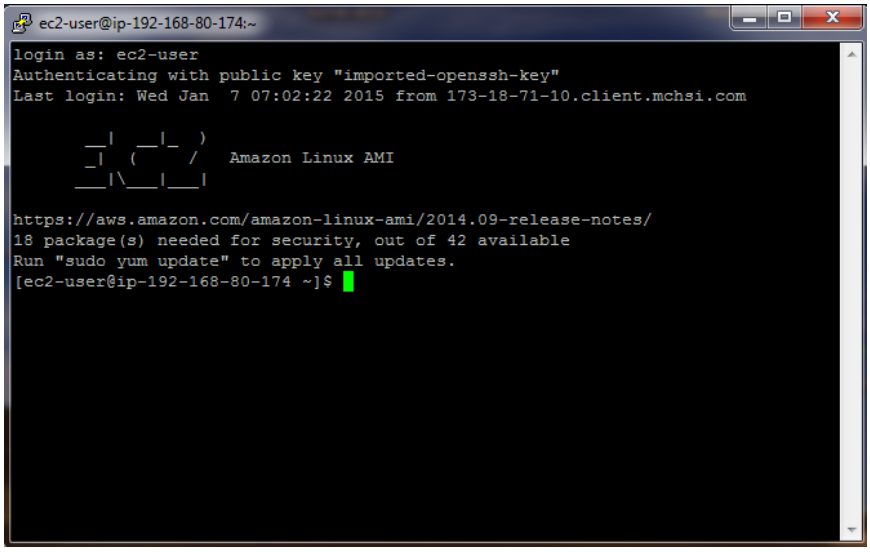


- Browse the 'key-ec2-putty' in Connection/SSH/Auth then click Browse.



- 'Yes' in the PuTTY alert. Then Login as 'ec2-user'





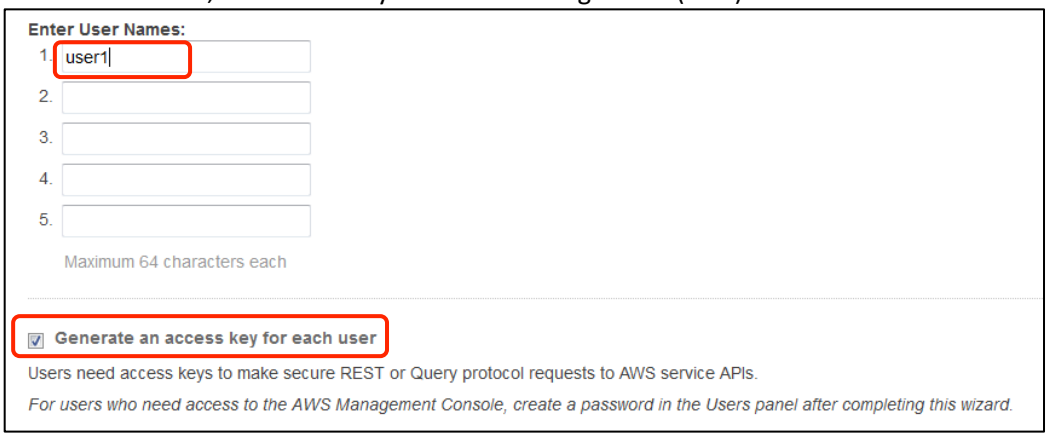
You can also use this new 'key-ec2-putty' key to copy files from windows to the Linux instance, using pscp

e.g. c:\>pscp –i key-ec2-putty.ppk any-file-name ec2-user@[public dns]:/var/www/html

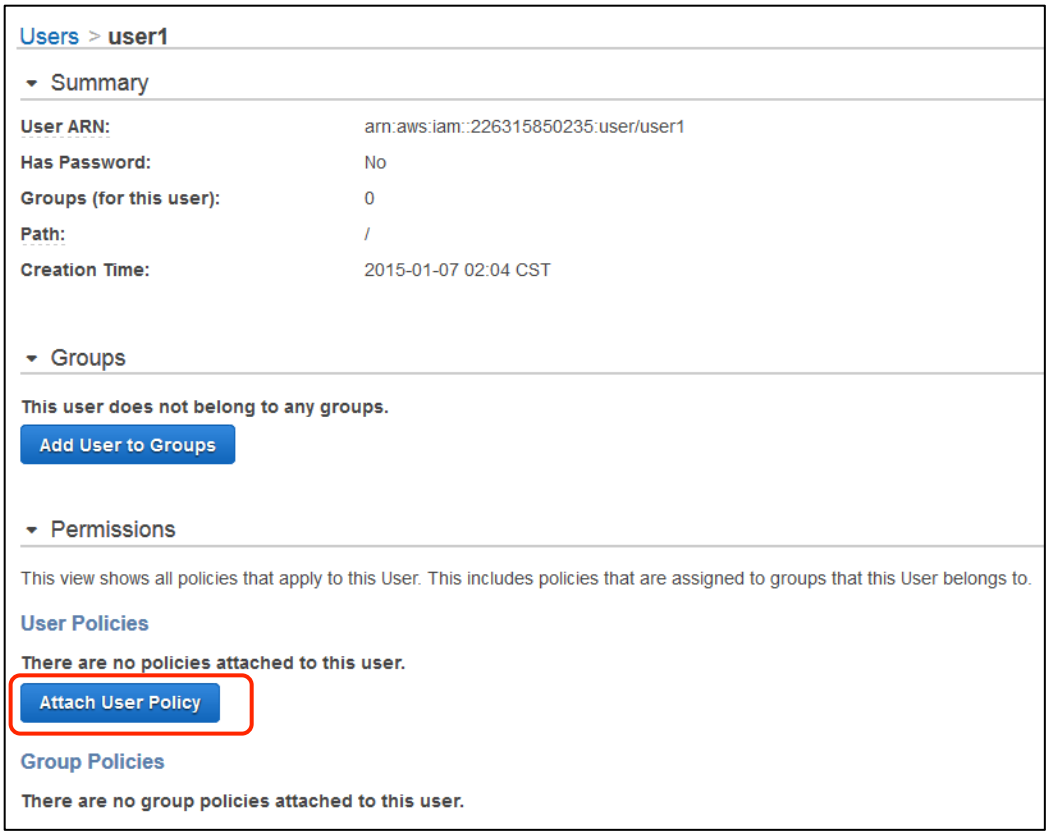
3.3 Amazon Command Line Configuration in your local machine.

3.3.1 OPTION 1.aws-cli Linux/Unix.

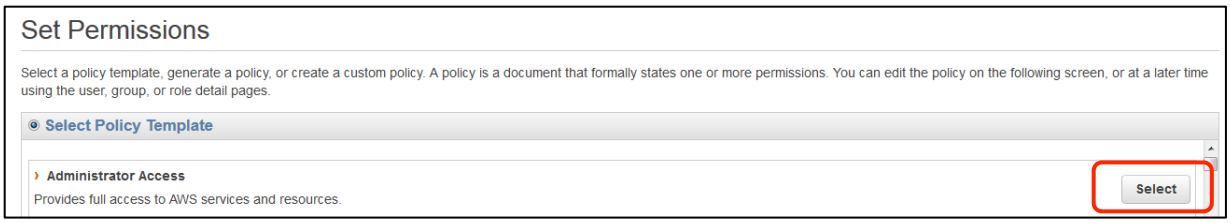
- In AWS console home, select Identity & Access Management (IAM) service and create a user 'user'1



- Once you create the user, download the 'credentials.cvs file that contain an 'Access Key Id' and 'Secret Access Key' and store it in /opt/aws directory on your **LOCAL** system.
- In IAM service, select the user 'user1' and add permissions.



- Grant Administrator Access to the user 'user1'.



- Install required dependencies, tools and amazon command line interface.

Note: Use easy_install pip if necessary on Mac OS.

```
#sudo su (optional if you are not using a root user)
#cd /opt/aws
#apt-get install openjdk-7-jdk
#java -version
#apt-get install curl
#curl http://python-distribute.org/distribute_setup.py | python
#curl https://raw.github.com/pypa/pip/master/contrib/get-pip.py | python
#pip install awscli
#apt-get install ec2-api-tools
```

PATH and environment variables.

It is recommended that you make these modifications in a global configuration file so that they are set every time you login to your host.

For Ubuntu, Xubuntu, open the bash file:

#nano /etc/bash.bashrc

At the end of the file add:

```
export JAVA_HOME=/usr/
export AWS_DEFAULT_REGION=us-east-1
export AWS_ACCESS_KEY=[copy-your-aws-access-key-Is-inside-rootkey.csv-file-without-brakets]
export AWS_SECRET_KEY=[copy-your-aws-secret-key-Is-inside-rootkey.csv-file-without-brakets]
```

For the changes to take effect **open a new terminal** or in your current terminal run:
#source /etc/bash.bashrc

Describe available regions

#ec2-describe-regions

```
root@ubuntu:/opt/aws# ec2-describe-regions
REGION  eu-west-1       ec2.eu-west-1.amazonaws.com
REGION  sa-east-1       ec2.sa-east-1.amazonaws.com
REGION  us-east-1       ec2.us-east-1.amazonaws.com
REGION  ap-northeast-1  ec2.ap-northeast-1.amazonaws.com
REGION  us-west-2       ec2.us-west-2.amazonaws.com
REGION  us-west-1       ec2.us-west-1.amazonaws.com
REGION  ap-southeast-1  ec2.ap-southeast-1.amazonaws.com
REGION  ap-southeast-2  ec2.ap-southeast-2.amazonaws.com
```

If you cannot see the previous output, open a new terminal and double check the information added in /etc/bash.bashrc.

A list of command can be found in the following link:
http://docs.aws.amazon.com/AWSEC2/latest/CommandLineReference/command-reference.html

Once you will be able to display the regions, you can execute the aws-cli command inside the BOXES from step **3.1.1**

3.3.2 OPTION 2.aws-cli Windows.

For AWS command line interface installation in your Windows environment, follow the steps at -
http://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/InstallEC2CommandLineTools.html

3.4 *Web Application - LAMP*

3.4.1 Login into your instance using your [public dns] and install http, php and mysql

#sudo yum -y install httpd php mysql php-mysql

```
[ec2-user@ip-10-164-28-156 ~]$ sudo yum -y install httpd php mysql php-mysql
Loaded plugins: priorities, security, update-motd, upgrade-helper
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.2.25-1.0.amzn1 will be installed
--> Processing Dependency: httpd-tools = 2.2.25-1.0.amzn1 for package: httpd-2.2
.25-1.0.amzn1.x86_64
```

…

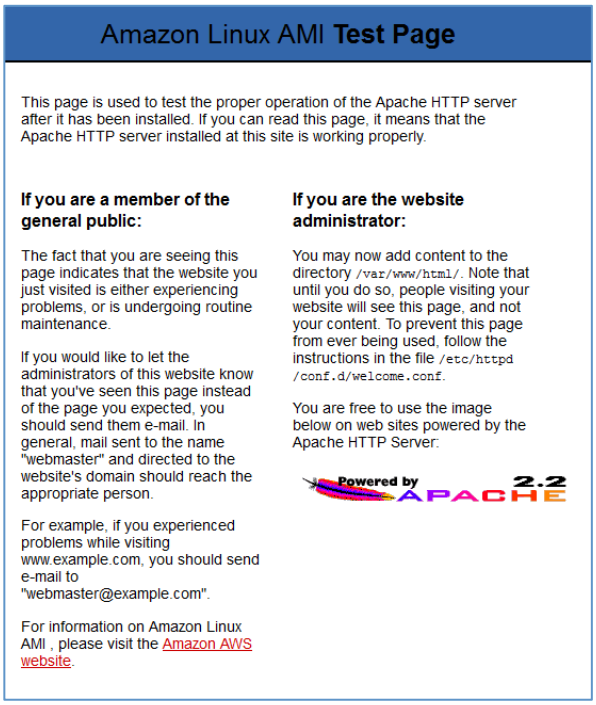
```
  mysql55-libs.x86_64 0:5.5.31-1.32.amzn1
  php-cli.x86_64 0:5.3.27-1.0.amzn1
  php-common.x86_64 0:5.3.27-1.0.amzn1
  php-pdo.x86_64 0:5.3.27-1.0.amzn1

Complete!
```

#sudo chkconfig httpd on
#sudo /etc/init.d/httpd start

```
[ec2-user@ip-10-164-28-156 ~]$ sudo chkconfig httpd on
[ec2-user@ip-10-164-28-156 ~]$ sudo /etc/init.d/httpd start
Starting httpd:                                            [  OK  ]
```

Add a new HTTP rule in your Security Group, similar to step 3.5.3 in AWS Lab1.

You should be able to see your default Apache web page if you paste your [public dns] in a browser.



3.4.2 Display Instance information

Move to the default web directory and create an index.php file.

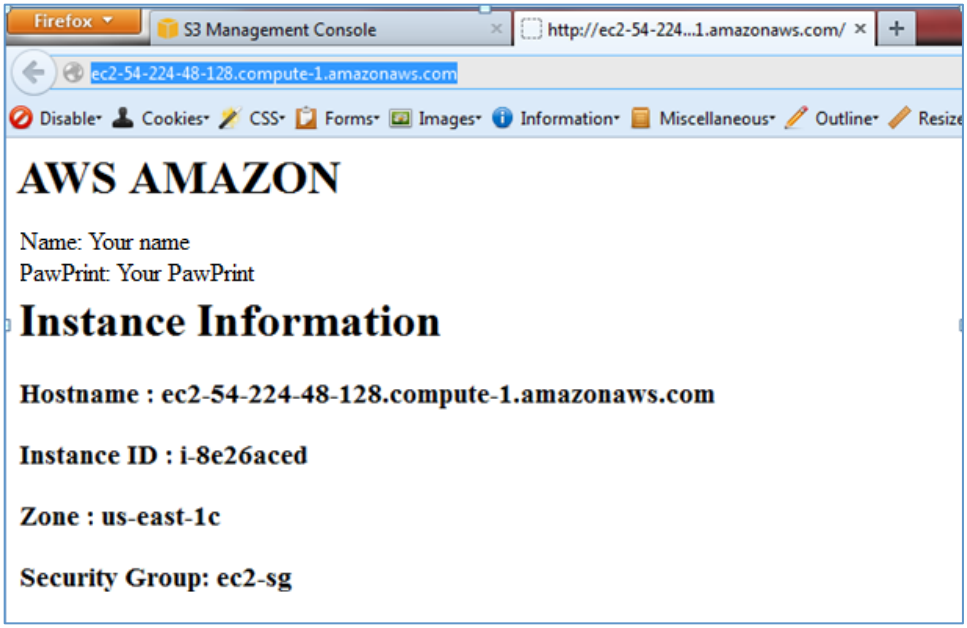
#cd /var/www/html

#sudo vi index.php

- Read Hostname, Instance ID, Zone and Security Group from Instances Data from metadata set and show it on the web page. **(Type again the double-quotes in your code).**

```
<html>
<head>
<h1>AWS AMAZON</h1>
Name: Your name</br>
PawPrint: Your PawPrint
</head>
<body>
<h1>AWS AMAZON</h1>
<?php
    $hostname = file_get_contents("http://169.254.169.254/latest/meta-data/public-hostname");
    $instance_id = file_get_contents("http://169.254.169.254/latest/meta-data/instance-id");
    $zone = file_get_contents("http://169.254.169.254/latest/meta-data/placement/availability-zone");
    $secgroup = file_get_contents("http://169.254.169.254/latest/meta-data/security-groups");
?>
<h2>Instance Information</h2>
<h3>Hostname          : <?php echo $hostname; ?></h3>
<h3>Instance ID       : <?php echo $instance_id; ?></h3>
<h3>Zone              : <?php echo $zone; ?></h3>
<h3>Security Group    : <?php echo $secgroup; ?></h3>
</body>
</html>
```

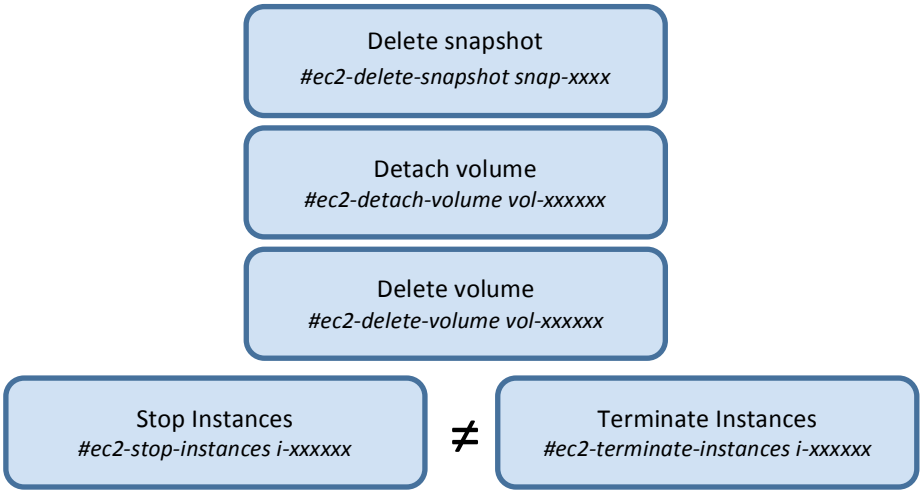
- In your browser and using your [public dns], you should be able to see similar Metadata information. Do not forget to add your name and PawPrint. **If you get an error, retype all the double quotes ("") manually in your index.php file.**



Take a screenshot of this output for grading purpose (See section 4 of manual).

3.5 Clean-up resources

- First of all detach and remove the extra volume added in Step 3.1.7 by selecting the 2GB volume and applying 'delete volume' in actions menu. Also remove the snapshot created in Step 3.1.8. Instead, you can perform the aws-cli commands.



4. What to turn in for Grading?

1. Create and delete a key pair called ‘cloud-key’ using aws-cli commands. Describe the commands used.

2. Create and delete a security group called ‘cloud-group’ in the default VPC, add ‘Open ports’ as a description and open SSH, HTTP, and HTTPS ports, use default IP source. Describe the commands used.

3. Enter the commands below and describe in detail the results; you will need an AMI available. Once the command is executed, using aws-cli commands, terminate the instances, delete ‘cloud-key’ and ‘cloud-group’. Include screenshot.

   #ec2-run-instances ami-xxxxxx \
   --instance-type t1.micro \
   --instance-count 2 \
   --key cloud-key \
   --group cloud-group \
   --region us-east-1

4. Get status information of all instances using aws-cli commands, include screenshot.

5. Create and remove a snapshot of your running instance volume, add the description ‘Backup’. Describe the commands used.

6. Use aws-cli commands to add and attach a new 3GB volume to your instance, describe commands. Include screenshot.

7. Provide a screenshot taken in Step 3.4.2 showing your name, PawPrint, and the metadata associated with your running instance. Also, clearly include the [public dns] link to that running instance to verify your job. After you receive your grade for this lab assignment, you can stop your instance.

8. Briefly explain the 6 AWS best practices described by Amazon AWS.