NANYANG TECHNOLOGICAL UNIVERSITY

LABORATORY MANUAL

CE3005: Computer Networks

Understanding Cloud Computing

SEMESTER 1
2016-2017

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

UNDERSTANDING CLOUD COMPUTING

1. OBJECTIVE

To explore and understand the fundamentals of using Cloud Computing, in this laboratory session the student will deploy a web service on a public cloud service provider (Amazon AWS).

2. LABORATORY

Open Access

3. EQUIPMENT

PC and access to the Internet.
Access is provided at Hardware Lab 1(N4-1a-03)

4. DURATION

2 hours.

5. INTRODUCTION TO CLOUD COMPUTING

Over the past few years, we have seen the pervasiveness of Cloud Computing. Most companies have or are planning service deployment in the Cloud. The definition of Cloud computing is given by Wikipedia as the use of resources (hardware and software) that are delivered as a service over a network (INTERNET). Cloud computing has made significant impact to the IT and business sector. It has the advantage of agility, scalability and a new costing model in which the users pay as they use, i.e., *pay-per-use* basis.

Cloud services can be categorized in many ways. A very popular way to classify Cloud services is using the layered view as shown in Figure 1. At the most basic layer we have the Infrastructure-as-a-Service, supported by providers such as Rackspace, GoGrid, Amazon. These Cloud providers provide resources, e.g., compute, storage and network resources to the users. With regard to *platform as a service*, users are provided with a platform to develop their applications, e.g., Google Application engine. Last but not least is Software-as-a-Service where users are provided with the software to use, e.g., Saleforce.com. All these cloud services are targeted at different users, from users who want control over the virtual machines to users who just want to use an application. Moreover, there are other ways of classifying services, e.g., Data-as-a-Service.

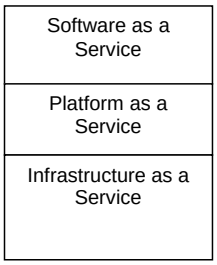

Figure 1: Cloud service model.

For IaaS Cloud service, we must select the “Instances” and set-up the environment as shown in figure 2. In the case of AWS, we need to select the instance type(CPU type, memory size). On top of this instance you would install the “images” (also known as AMI) of the operating system and software environment that you wish to run your programs. Attached to the instance are the storage type and the network and security requirement.

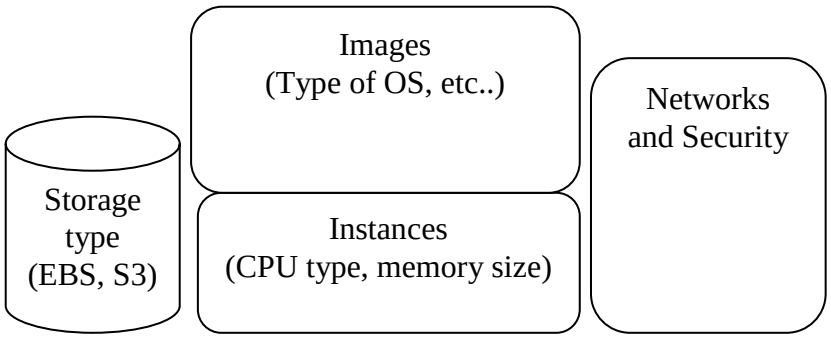

Figure 2: Components of a VM environment

In this laboratory session, we will focus on Infrastructure-as-a-Service (IaaS) and consider the cloud services provided by Amazon. The laboratory session is divided into a number of exercises. However, we can view it as taking you through the following 4 stages:

-Stage 1: Create an Amazon AWS account.
-Stage 2: Launch an Amazon EC2 Instance.
-Stage 3: Set-up environment and storage resources around the EC2 Instance
-Stage 4: Create a website using Cloud services from Amazon.

6. EXERCISE 1: REGISTERING WITH A PUBLIC CLOUD PROVIDER

To use Cloud computing resources provided by Amazon, we first need to register an account with Amazon AWS. Go to http://aws.amazon.com and request for an AWS account. After passing the identification assessment, you can then login with your AWS account.

After you have login into the AWS account, you will see the AWS Management Console EC2 dashboard as shown in Figure 3.

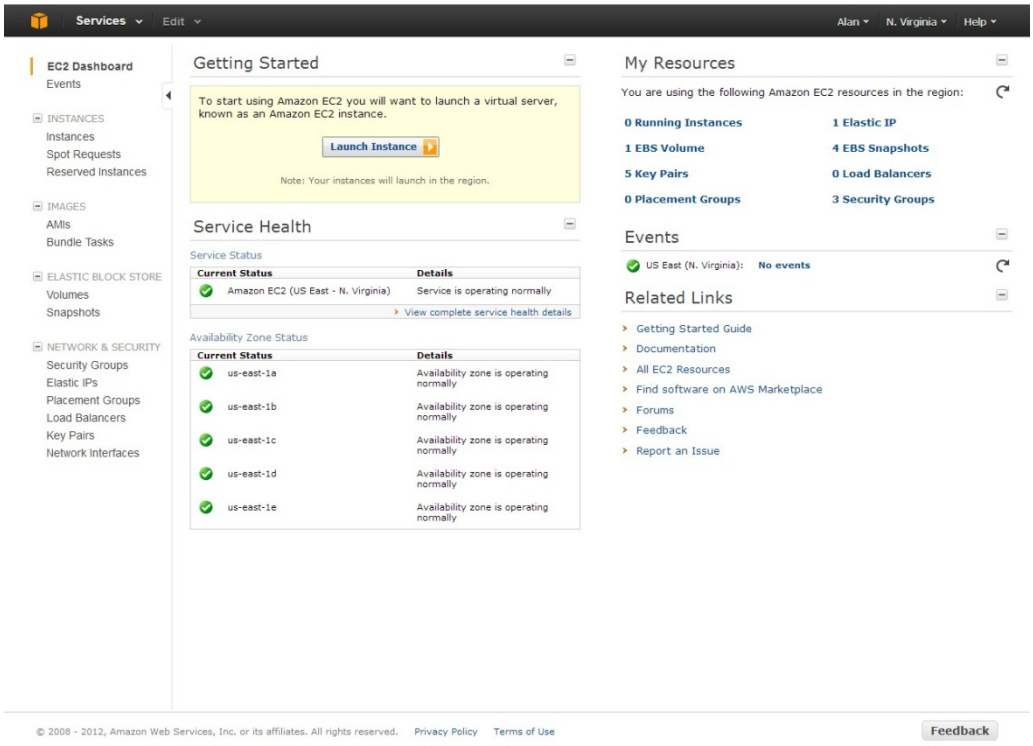

Figure 3: AWS management console EC2 dashboard.

In the left-side column of the management console, you would see the available "resources", e.g., instances, images, storages, and network and security.

7. EXERCISE 2: CREATING AN SSH KEY PAIR

Communication with Amazon EC2 instances is through SSH protocol. Thus, we must first create the SSH key pair as follows:

- Login to http://console.aws.amazon.com
- Click "**EC2**" to go to "**EC2 Dashboard**"
- Select region as "**Asia Pacific (Singapore)**" (in upper right quarter of the page, see Figure 4).
- Click on "**Key Pairs**" tab (under "**NETWORK & SECURITY**" menu).
- Click "**Create Key Pair**" button.
- Enter a name in the pop-up dialog and proceed to create the Key Pair by clicking "**Create**". The AWS console will initiate a file to be downloaded.
- Save the "**.pem**" file in your machine.

To create an SSH client on your PC, if you do not have an SSH client, you can download the software from **PuTTY** on Windows (http://www.putty.org). Do the following:

- Launch the **PuTTYgen**, and click on the "**Conversions**" tab,
- Choose "**Import key**" and specify the location of the "**.pem**" file you save previously. Do NOT put a passphrase.
- Save the private key by going to the "**File**" tab, choose "**Save private key**".

Figure 4: Region selection menu

8. EXERCISE 3: LAUNCH AN INSTANCE

In the previous exercise we have learnt about Amazon Web Services and set-up the SSH software on your client machine. In this exercise we will launch a Linux instance using the AWS Management Console.

- Login to https://console.aws.amazon.com/ec2, using the email address and password you have specified when you signed up for AWS.
- Select the same region (Singapore) as you done in Exercise 2.
- Click "**Launch Instance**" button. You are prompted to select the method of creating the Instance. Select "**Quick Start**".
    - A list of valid AMIs will be displayed for you to select. Select the latest version of "**Ubuntu Server**".
- In the next prompted screen, click "**Review and Launch**".
- In the review page, click "**Edit security groups**" -> "**Select an existing security group**" -> "**default**" to use the "**default**" security group and click "**Review and Launch**" -> "**Launch**". And choose the Key Pair you wish to use to access the instance in "**Select a key pair**" field and "**tick**" the agreement. DO NOT select NONE, as this means you cannot connect to the Instance.
- **Click "View Instances" to view the existing instances you have created already.**
- **C**lick "Actions" to select the instance state, "start", "stop", "reboot", and "terminate" (Note that you are billed from this point onwards till you terminate the instance.)
- View the status of the Instance, by selecting it from the list. (An example of the screen is as shown in Figure 5.)

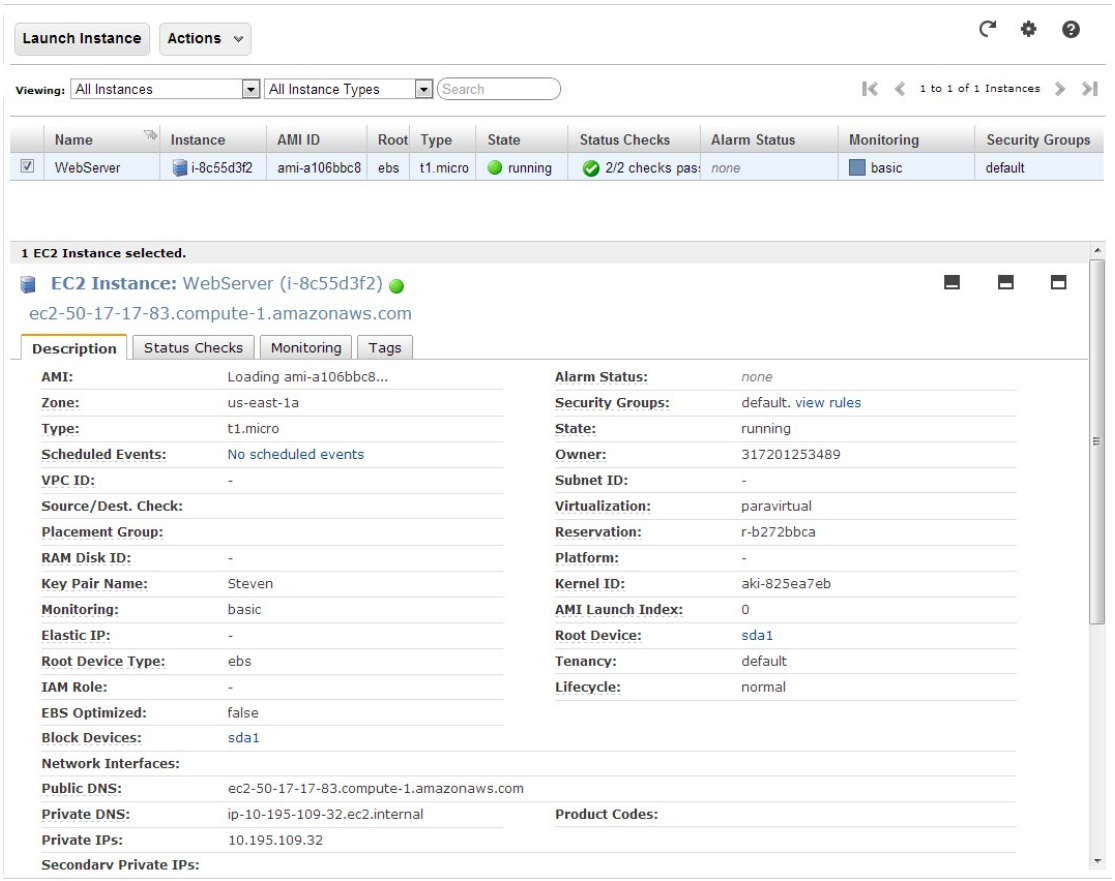

Figure 5: Sample image of the screen showing status of Instances

9. EXERCISE 4: CONFIGURING FIREWALL

By default, the "default" security group only allows intercommunications within the same group, meaning that we cannot access remote instances in this group from your own local machines through SSH protocol without any configuration. Thus, there is a need to enable SSH connection by adding a new rule to the "default" security group as follows:

- Go to EC2 Dashboard and select "**NETWORK & SECURITY**" -> "**Security Groups**".
- Choose the "**default**" security group, and click "**Inbound**" tab. **Click the "Edit" button, then click "Add Rule", select "SSH" with "Anywhere" source (0.0.0.0/0), and click "Save"**

Similarly, to allow access through HTTP protocol, which we will use to access our website in Exercise 8, we need to enable connections using HTTP protocol as well. Do the following:

- Choose the "default" security group, and click **the "Edit" button, then click "Add Rule", select "HTTP" with "Anywhere" source (0.0.0.0/0), and click "Save"**

So far, we have completed the security settings, as shown in Figure 6. Instances inside the "default" security group can be access using both SSH and HTTP (port "80" only) protocols.

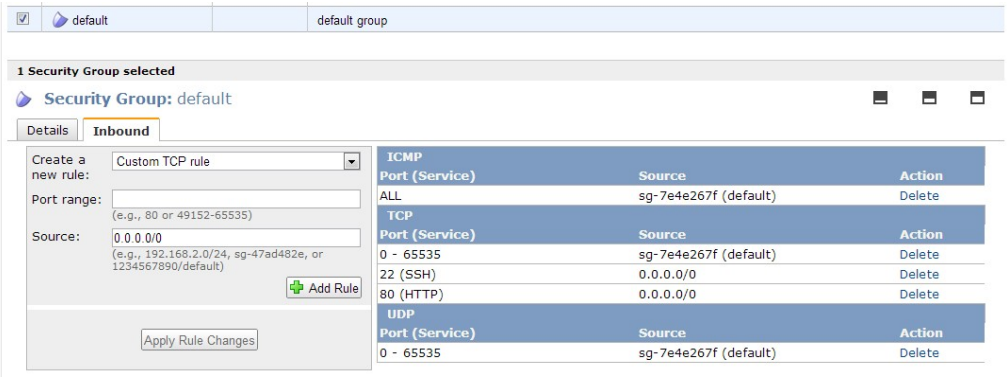

Figure 6: Settings of Security Group: default.

10. EXERCISE 5: CONNECT TO EC2 INSTANCE

You can connect to the instance using the "Key Pair" you have generated previously. The address of the new instance is given by the Public DNS name that you can get from Instance Description. Using **PuTTY** on Windows:

- Launch **PuTTY** and paste the Public DNS name into Host Name (or IP address) field with port number "22" as shown in Figure 7.
- Select the "**Auth**" tree item (nested inside the "**SSH**" tab) shown in Figure 8 and use the "**Browse**" button to locate the private key you have saved previously in Exercise 2.
- Select the "**Open**" button and connect to the instance.
- Click "**Yes**" in the promoted **PuTTY Security Alert** and login as "**ubuntu**".

You are now connected to your instance, and you will see as screen similar to Figure 9.

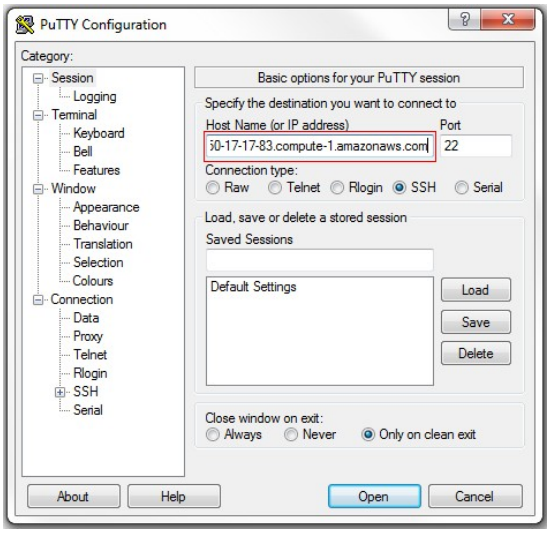

Figure 7: Overview of Connecting Amazon EC2 instance from PuTTY on Windows.

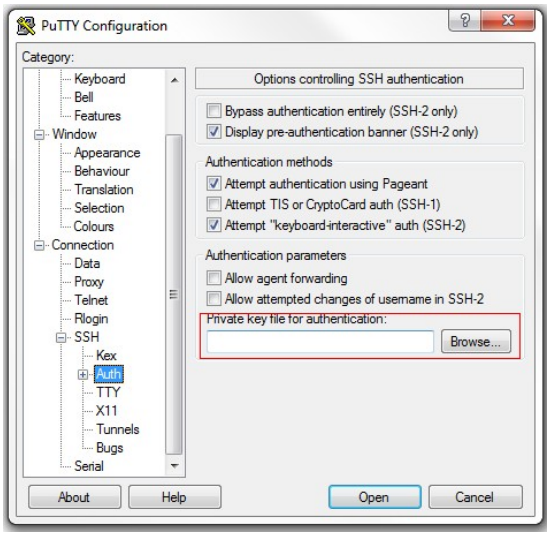

Figure 8: Input Private Key file for authentication in PuTTY on Windows.

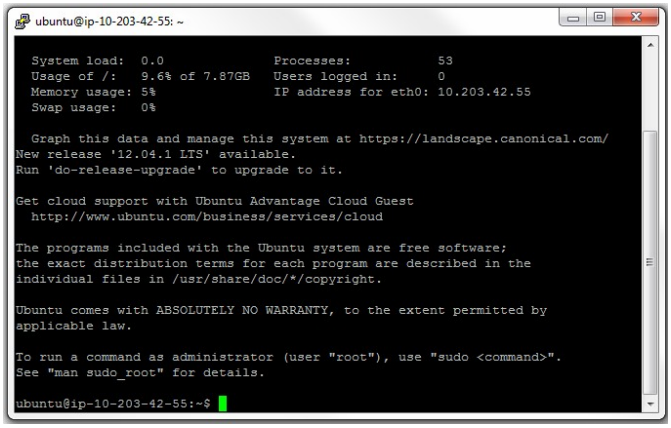

Figure 9: Login home using PuTTY on Windows.

11. EXERCISE 6: MOUNTING AMAZON S3TO EC2 INSTANCE

In this exercise, we will create an Amazon Simple Storage Service (S3) bucket and mount it to our EC2 instance. And we will storage our data in Amazon S3. Create an Amazon S3 bucket as below:

- Go to https://console.aws.amazon.com to login with your AWS account if necessary.
- Select "**S3**" service from Amazon Web Services panel to entry S3 dashboard screen as shown in Figure 10.
- Click "**Create Bucket**" button, enter your bucket name (Note: bucket name must be globally unique, you may try multiple times to find out an available name), and choose the region as "**Singapore**", then click "**Create**".

You now have created an Amazon S3 bucket with default security settings that only allow you to access this bucket. You can upload data to and download data to/from your bucket. Do NOT change the permissions of your bucket to "Public"; otherwise everyone can access your bucket.

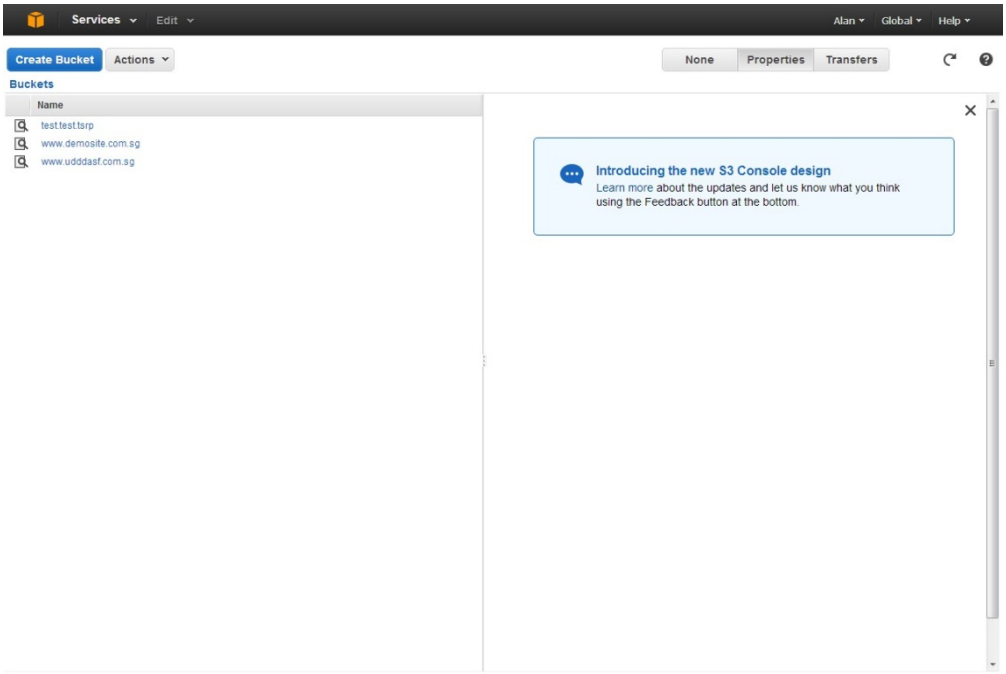

Figure 10: AWS Management Console S3 Dashboard.

Second, we will mount aforementioned Amazon S3 bucket to Amazon EC2 file system:

- Login to the EC2 instance you previously launched and download a tool for S3: "*sudo wget http://s3fs.googlecode.com/files/s3fs-1.61.tar.gz*".
- Unpack the downloaded package: "*tar zxvf s3fs-1.61.tar.gz*".
- Update Ubuntu's repository: "*sudo apt-get update*".
- Install GCC compiler: "*sudo apt-get install gcc*".
- Install dependencies for building the tool: "*sudo apt-get install build-essential libxml2-dev libfuse-dev libcurl4-openssl-dev*".
- Configure, build, and install the S3 tool: (1) change directory: "*cd s3fs-1.61*"; (2) configure the code: "*./configure*"; (3) compile and install the code: "*sudo make install*".
- New a configuration file for storing your keys (you can check out your "**Access Key ID**" and "**Secret Access Key**" associated with your AWS account under **AWS Menu**(*in upper right quarter of the page*)->**Your Account**->"**Security Credentials**", see Figure 11. NOTE: Please keep this information safe): (1) "*touch passwd-s3fs*"; (2) "*chmod 640 passwd-s3fs*"; (3) "*echo 'Your-Access-Key-ID':'Your-Secret-Access-Key' > passwd-s3fs*"; (4) "*sudo mv passwd-s3fs /etc/passwd-s3fs*". (Note: in (3), replace '*Your-Access-Key-ID*' and '*Your-Secret-Access-Key*' with your "**Access Key ID**" and "**Secret Access Key**" respectively, and omit the single quotation marks. Using colon as separator is necessary.)
- Create a mount point: "*sudo mkdir /mnt/s3*".
- Mount the bucket just created to EC2: "*sudo s3fs –o allow_other 'The-Bucket-Name' /mnt/s3*". (Replace '*The-Bucket-Name*' with your real bucket name and omit single quotation marks.)
- ***Set up mounting S3 bucket at boot time to file "**/etc/fstab**". To edit this file, *root* privilege is required. To do so, follow steps below:
  - "*sudo vi /etc/fstab*"
  - Enter "**i**" to change to "**edit**" mode.
  - Type "*s3fs#'Your-Bucket-Name' /mnt/s3 fuse allow other 0 0*"
  - Press **Esc** to exit typing.
  - Enter **Shift**+"**:**" and type "**wq**" to exit editing and save the file.
- You can see the contents of your bucket: "*sudo ls –l /mnt/s3*".

Now, you have succeeded in mounting an Amazon S3 bucket to EC2 file system, and you can simply store you data to Amazon S3 by putting them into directory "**/mnt/s3**".

***Note:** if you cannot set up mounting S3 at boot time correctly, most probably the instance will also not be accessible. In such case, you are required to terminate it and launch a new S3 bucket. Thus, be careful when editing "**fstab**" file.

WARNING: Amazon S3 charges for storage costs, as well as Request and Data Transfer costs. Please check out the pricing scheme and your free tier at http://aws.amazon.com/s3/pricing
Be careful of your budget while using Amazon S3.

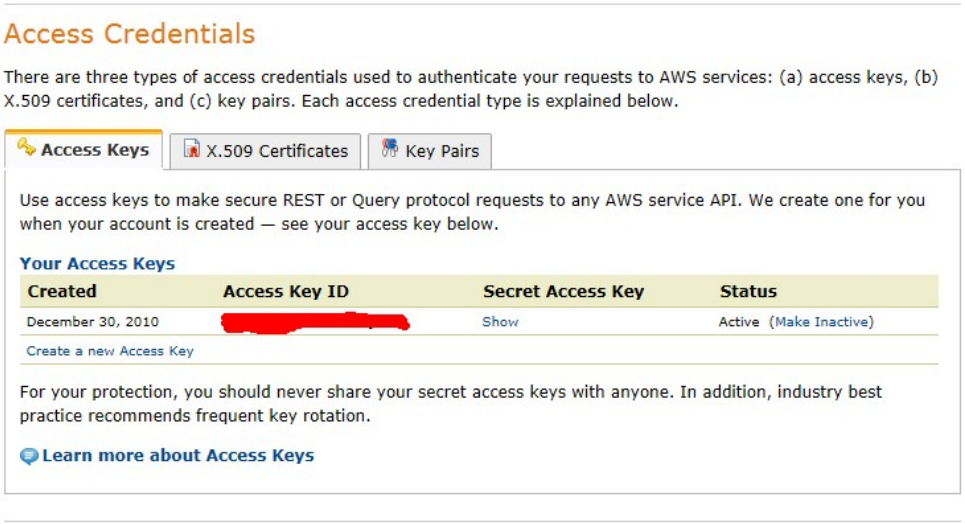

Figure 11: Access credentials associated with your AWS account.

12. EXERCISE 7: RESERVE PUBLIC ELASTIC IP ADDRESS

Each time you stop an EC2 instance and restart it again, its public IP address will change which is inconvenient for a web server since we need to access our website via the public IP address. To associate a static IP address to our web server, we are required to request an elastic IP as below:

- Go to Amazon Management Console EC2 Dashboard.
- Click "**Elastic IPs**" on the left-side menu and then click "**Allocate New Address**" button. In the pop-up dialog, select default setting (EIP used in EC2) and click "**Yes, Allocate**" button. And you can see a new IP address has been allocated to you as shown in Figure 12.
- Right-click the allocated IP address and select "**Associate**", then choose the instance to which you wish to associate with the IP address. Next, simply click "**Yes, Associate**".

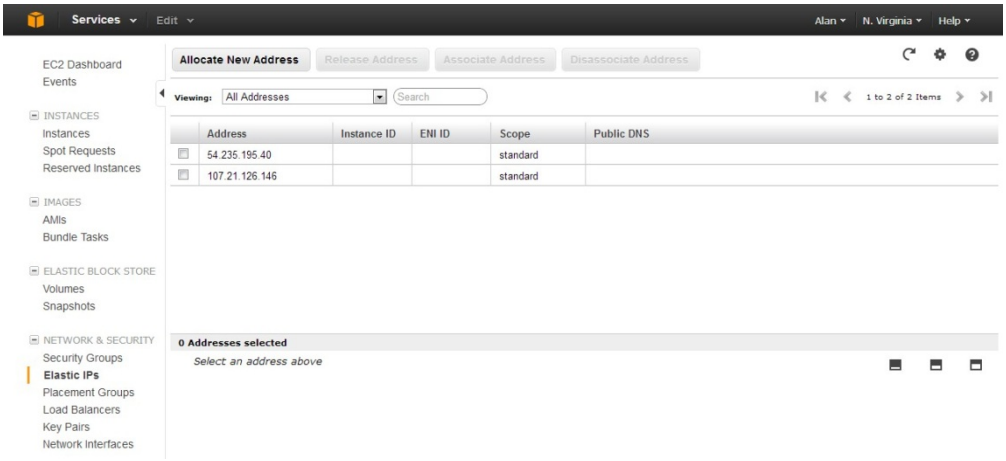

Figure 12: Elastic IPs management window.

Now, you have assigned a static IP address to your instance. **Note** that each time you restart the instance; you need to associate the static IP address to it again.

13. EXERCISE 8: SETTING UP AN WEBSITE

We will next setup a LAMP (Linux, Apache, MySQL and PHP) web site on the instance by following steps below:

- Install Apache Http Server: "*sudo apt-get install apache2*".
- Test for Apache Server installation: enter the Public DNS in your browser to check whether you can see "*It works…*". If so, installation of Apache Server is successful.
- Install MySQL Server: "*sudo apt-get install mysql-server*", enter password for root as it prompts.
- Install PHP 5 and Apache PHP5 Module: "*sudo apt-get install php5 libapache2-mod-php5*", and restart Apache Server by entering command "*sudo /etc/init.d/apache2 restart*".
- Test for PHP installation: new a file "*test.php*" under directory "/var/www/html/" with contents "<?php phpinfo(); ?>", and then type "*Public DNS*"+"/test.php" in your browser to check whether you can see PHP information page.
- Configure Apache and PHP to support MySQL via following commands: (1) *"sudo apt-get install libapache2-mod-auth-mysql;* (2) "*sudo apt-get install php5-mysql*"; (3) "*sudo /etc/init.d/apache2 restart*".

You now have setup your own web server. Create your website and upload to "**/var/www**" and you can access the website via entering Public DNS in your browser.

To upload files to Amazon EC2 Instance, you need to install **WinSCP** on Windows (http://winscp.net/eng/index.php). To connect to you EC2 instances through **WinSCP**, it is similar to connecting using **PuTTY**. You will use the private key you previously generated from **PuTTY**.

14. EXERCISE 9: SHUT DOWN

After completing your exercise, remember to terminate the instance you launched. Amazon will continue to charge you for the instances as long as you have not terminated the instances. To terminate an instance, do the following steps:

- Sign in to the AWS Management Console and open the Amazon EC2 console.
- Locate your instance in the list of instances on the Instance page.
- ***Right-click the instance, and then click **"Terminate"**.
- Click **"Yes"**, "**Terminate**" when prompted for confirmation.

NOTE: once you terminate an instance, everything inside the instance will disappear and cannot be restored any more.

***Alternatively, you can "**Stop**" the instance instead of terminating it, if you want to reuse the instance in the near future. To start a stopped instance, right-click the instance and click "**Start**".

WARN: DO NOTE that even though Amazon does not charge compute hours for **stopped** instances, you are still charged for EBS storage and S3 storage (if any) attached with those stopped instances. Please check out the Amazon EC2 pricing and your free quotas at
http://aws.amazon.com/ec2/pricing.

For new users, we **recommend** that you should terminate those running instances after use in case of overdrawing your free tier entitlements.

15. CONCLUSIONS

We have gone through the entire process of creating an account and setting up a web site on the public cloud: Amazon EC2. It is hoped that it has introduce, in a very small way, you to the Cloud computing. Do note that there are other Cloud service providers who have a slightly different way of setting up a service.

References

1. "Host Your web site in the Cloud. Amazon web service made easy"ed. Jeff Barr. Publisher Sitepoint Pte. Ltd.
2. "*User Guide of Amazon Elastic Compute Cloud*" http://awsdocs.s3.amazonaws.com/EC2/latest/ec2-ug.pdf
3. "Cloud Computing : A Practical Guide"ed. Anthony T. Velte, Toby J. Velte, Robert Elsenpeter. Publisher Mc. Graw Hill

Acknowledgement

Special thanks to Changbing Chen, Tang Shanjiang, Liu Yi and Lim-Tan Lay Choo for their help and assistance in creating, refining and testing this laboratory session.