# Homework #7 Amazon Elastic Compute Cloud Web Services

**This semester we are allowing all students to explore cloud computing as offered by Amazon's Web Services. Using the instructions below one can establish a website at AWS. Once established, you will be able to move your PHP program developed for Assignment #6 to your AWS website and have it execute there.**

## 1. Sign up

To sign up for AWS, you need a credit card. If you do not have one, buy a $25 American Express Gift card at Ralphs or other grocery store.

To sign up go to:

http://aws.amazon.com

and click on **Sign Up**. Follow the instructions to create your account using the "AWS Free Usage Tier".

Please note that many of the URLs listed from now on will only be available if you are signed up to AWS.

## 2. Apply the $100 credit

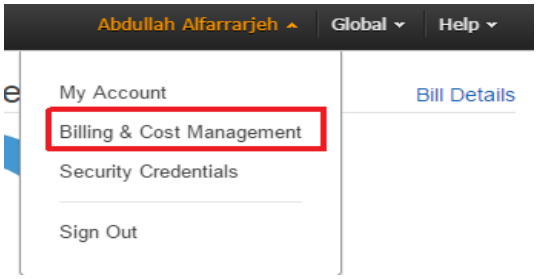
Once you are signed up, login to your AWS account. Go to the URL:

http://aws.amazon.com/awscredits/

Enter the code that we will provide to you and click on **Redeem Credit**.

## 3. Provide Credit Card or Gift Card Information

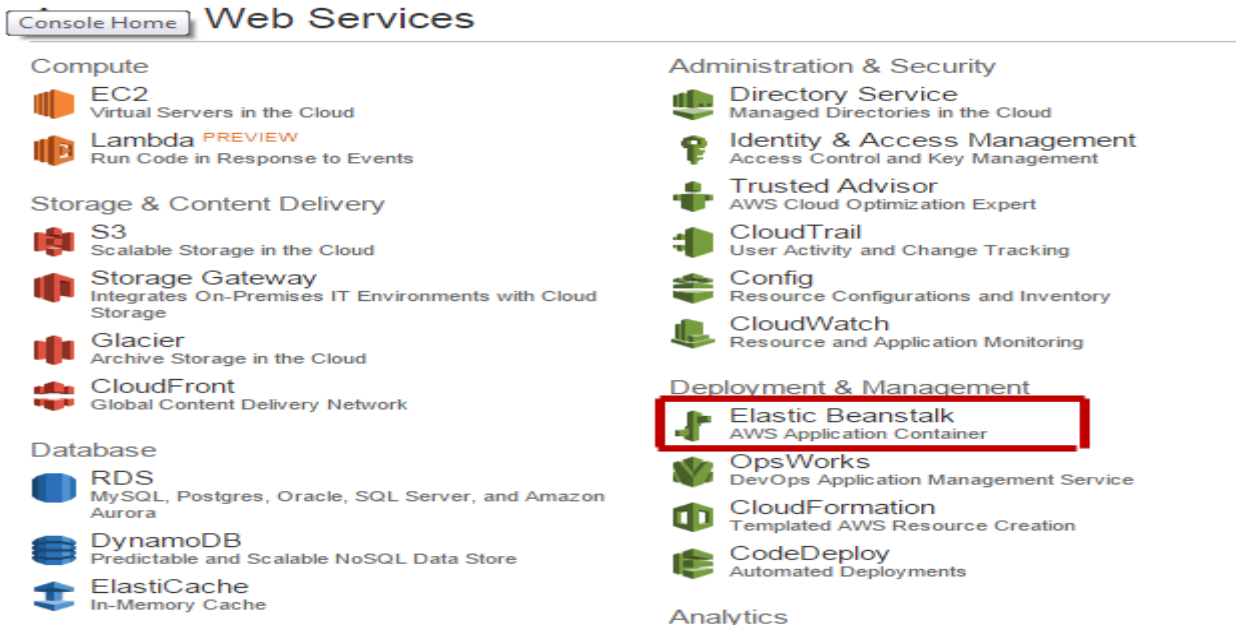
- In the top menu, click on your name and select **Billing and Cost Management** in the drop down menu.



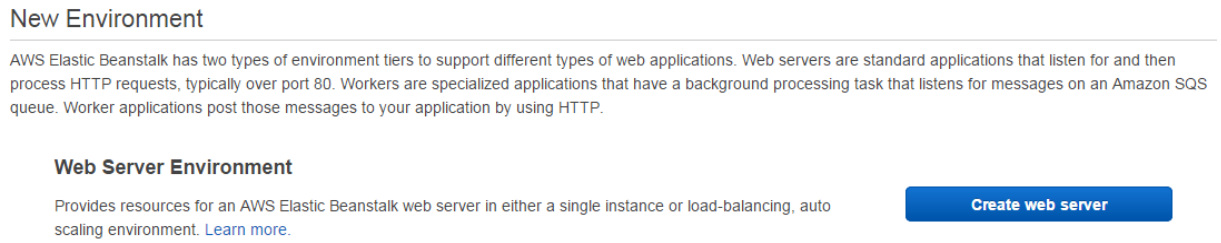

- In the left menu select **Payment Methods**
- Click on **Add a Card** button
- Provide your card information (Credit Card Number, Name of Cardholder, Expiration Data)

## 4. Set up the Default Elastic Beanstalk Application

- From the **Services** drop down in the top left select **Console Home**.
- From the list of Amazon Web Services, select **Elastic Beanstalk,** under **Deployment & Management**.

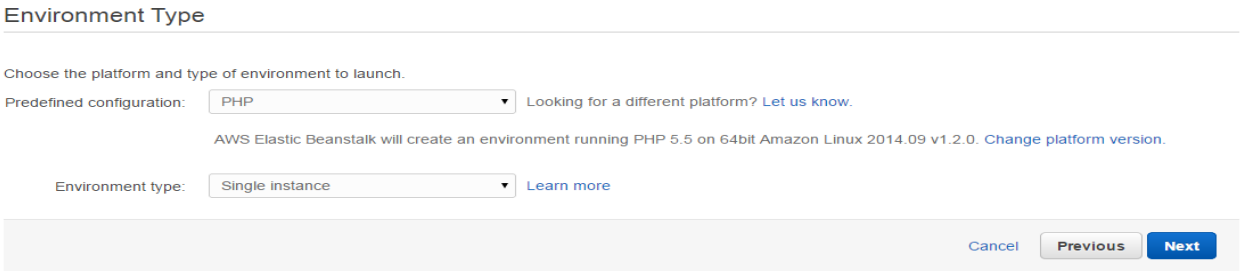



- Select **Create a New Application** in the top left menu and follow the Wizard.
- In the **Application Info** section, select a name for your application. Click Next.
- In the **New Environment** section: click on the **Create Web Server** button

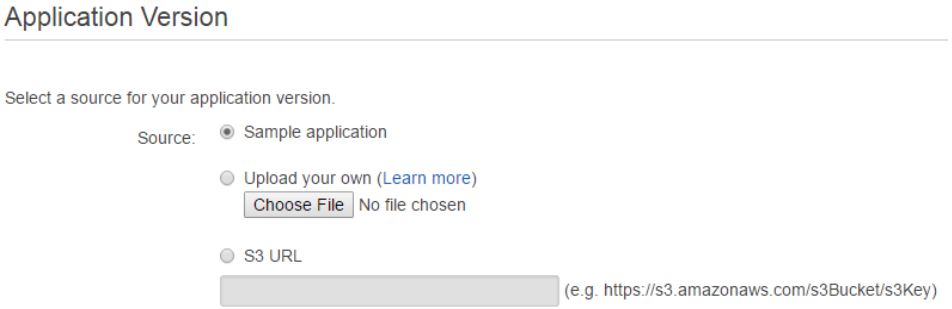



- In the Permissions pop-up window, select “Create an IAM role and instance profile”. Then click on the **next** button.
- In the Environment type section: choose the following options in the drop-down list
  - Predefined Configuration: **PHP**
  - Environment Type: **Single Instance**.

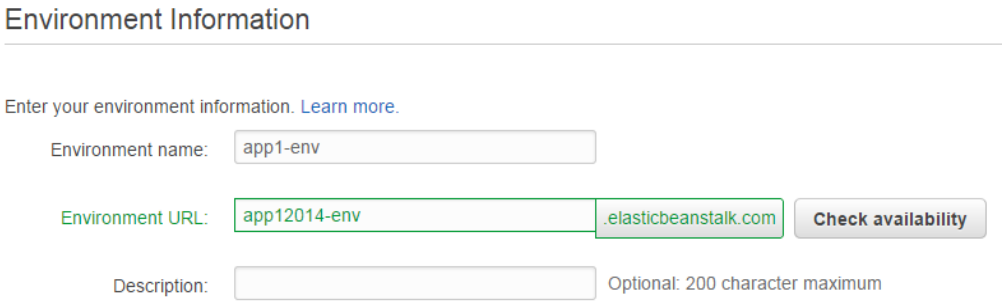
Then click on Next



- In the **Application Version** section, select **Sample Application.** Click Next

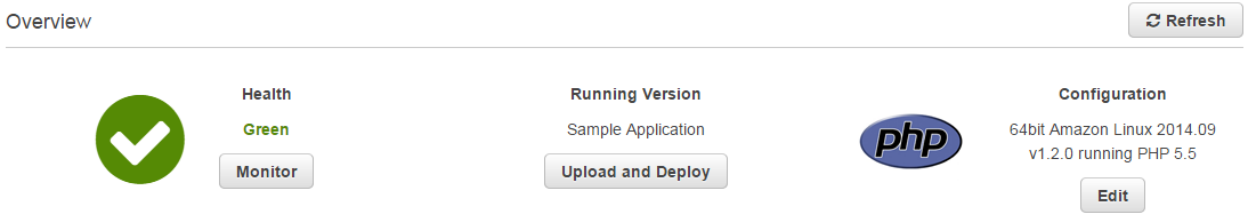



- In the **Environment Information** section, select an **Environment URL** (use the default or check availability of your own subdomain of elasticbeanstalk.com). Click on “Check Availability” button. Your URL should be green. Otherwise you should change the environment URL. Click Next.




- In the **Additional Resources** section leave all boxes **unchecked**. Click Next.
- In the **Configuration Details** section. Accept default configuration (e.g., instance type is t1.micro) and click Next.
- In the **Environment Tags** section click Next.
- In the **Review** section click **Launch**, The default "My First Elastic Beanstalk Application will be created.

You will need to wait for several minutes as your Linux + Apache + PHP-5 instance is created and launched. You will see a *rotating wheel* next to **the “Monitor” button**. Once creation and launch are completed, you will see the wheel turn into a green round circle with a check mark in the middle.




Beside “Default-Environment” subtitle there is a link such as *YourAppName-env*.elasticbeanstalk.com, click on it. You should see the "*Congratulations*" page. If you see it so your application and environment have been created properly. Then go back to Elastic Beanstalk console.


## 5. Upload your PHP application

Develop your PHP server application, and make sure that you name the file **index.php**. Compress the file with ZIP so that the resulting file is named **index.php.zip**. On a Mac you can use zip or gzip. On Windows there are several free programs 7-Zip or FreeZip, etc. that you can use.

From the Elastic Beanstalk console click on the **Upload and Deploy** button.

The **Upload and Deploy** popup will display. Enter a Version **label** (e.g., version1.0). Click on the **Choose File** button and select the **index.php.zip** file. Then click on Deploy button. Again wait several minutes for the *rotating wheel* to finish and the green circle with checkmark to appear. Click again on **the link "*YourAppName-env*.elasticbeanstalk.com"**. Check that your PHP app is running correctly.

- **Important Note:** in the future if you want to upload an updated version of *index.php*, you should enter a different version label. Otherwise, you will get an error.

To browse how your web service and application has been set up, go to the **Services** console (which is located in the top menu of Elastic Beanstalk console) and select the **EC2** Service. You should see that you have 1 **Running Instance (**the 64-bit Linux service), 1 **Volume** (the 8GB disk hosting your instance), and 1 or no **Load Balancer** (managing port 80).

## 6. Set up Exploring Your Instance (Optional)

If you want to explore your Instance and create your own domain based URL with SSH control, you can add the following steps.

### 6.1 Get and Setup SSH

Once the PHP app with SSH-enabled environment is running, you can get access using SSH. You can use ssh on a Mac running OS X, or Putty when running on Windows.

On a Mac, SSH is built into OSX X and can be accessed through the **Terminal** app and there is no additional setup needed.

On a Windows PC, you will need to download the complete PuTTY distribution at:

http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

You should download the file putty.zip that contains all the binaries, including **PuTTYgen** at:

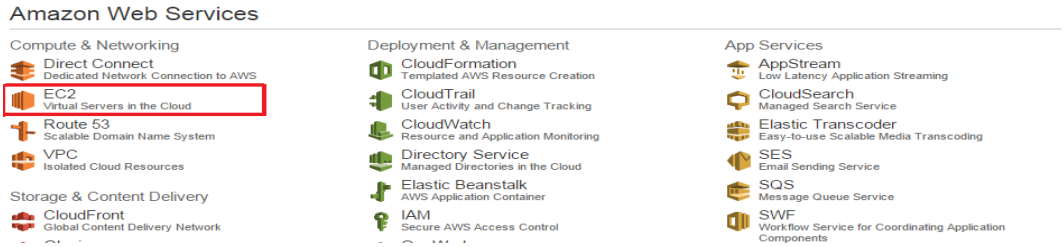
http://tartarus.org/~simon/putty-snapshots/x86/putty.zip

PuTTY needs additional setup as it needs to use a converted version of the private key. The instructions on how to perform such conversion are available here:

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

### 6.2 Create a Key Pair

- From the **Services** drop down in the top right select **Console Home**.
- From **Services**, select the **EC2** dashboard.




- Under **NETWORK AND SECURITY** select **Key Pairs**.
- Click on the button **Create Key Pair**.
- Enter a name like ***phphosts*** **(you must have your own random name!)** and click on **Yes**.
- A download of your private key should start. Save the key, like **phphosts.pem**, in an appropriate location.

### 6.3 Access your Linux Instance with SSH

- To see how to launch your SSH client go to **Services** and select **EC2**.
- Select **Instances** in the navigation pane at left.
- Select your instance in the table and select **Connect** button.
- "Connect To Your Instance" popup will display. Click on **Connect with a standalone SSH client**.

#### 6.4.1 Mac running OS X / ssh

On a Mac you will need to enter a command like this one:

```
ssh -i phphosts.pem ec2-user@ec2-54-235-60-138.compute-
1.amazonaws.com
```

type **yes**, when asked. Make sure that you are executing the ssh command in the same folder that contains the key. You should see output similar to this one:

```
$ ssh -i phphosts.pem ec2-user@ec2-54-235-60-138.compute-
1.amazonaws.com
The authenticity of host 'ec2-54-235-60-138.compute-
1.amazonaws.com (54.235.60.138)' can't be established.
RSA key fingerprint is
47:f7:b3:2f:88:b5:62:d9:6c:7c:60:19:53:cd:1a:0c.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-54-235-60-138.compute-
1.amazonaws.com,54.235.60.138' (RSA) to the list of known hosts.

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

https://aws.amazon.com/amazon-linux-ami/2012.09-release-notes/
There are 15 security update(s) out of 56 total update(s)
available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-158-56-168 ~]$
```

You can find more info here:

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstancesLinux.html - AccessingInstancesLinuxSSHClient

### 6.4.2 PC running Windows / PuTTY

In the popup windows titled **Connect To Your Instance**, click on **Connect using PuTTY**. You will be redirected to the URL.

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

### 6.5 Explore

You can now explore your Instance. When you login with SSH, your account home directory will be located at:

/home/ec2-user

That folder is empty, and is not where your apache / PHP files are. Run 'ps -ax', and you should see several instances of **httpd**:

```
 1940 ?        S      0:00 /usr/sbin/httpd -D FOREGROUND
```

```
 1941 ?        S      0:00 /usr/sbin/httpd -D FOREGROUND
 1942 ?        S      0:00 /usr/sbin/httpd -D FOREGROUND
 1944 ?        S      0:00 /usr/sbin/httpd -D FOREGROUND
 1945 ?        S      0:00 /usr/sbin/httpd -D FOREGROUND
```

To see your mounted volumes, run 'df -h':

```
[ec2-user@ip-10-158-56-168 sbin]$ df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/xvda1            7.9G  1.3G  6.6G  17% /
tmpfs                 298M     0  298M   0% /dev/shm
[ec2-user@ip-10-158-56-168 sbin]$
```

To see your document root, run 'ls  /var/www/html/':

```
[ec2-user@ip-10-158-56-168 html]$ ls -l /var/www/html
lrwxrwxrwx 1 root root 16 Mar 22 20:38 /var/www/html ->
/var/app/current
[ec2-user@ip-10-158-56-168 html]$
```

To see your uploaded index.php file:

```
[ec2-user@ip-10-158-56-168 sbin]$ cd /var/www/html
[ec2-user@ip-10-158-56-168 html]$ ls -l
total 4
-rw-r--r-- 1 webapp webapp 3723 Feb  5 02:09 index.php
[ec2-user@ip-10-158-56-168 html]$
```

To see your php.ini file, 'ls -l /etc/php.ini':

```
[ec2-user@ip-10-158-56-168 html]$ ls -l /etc/php.ini
-rw-r--r-- 1 root root 65782 Mar 22 20:38 /etc/php.ini
[ec2-user@ip-10-158-56-168 html]$
```

**Have fun exploring AWS!!**