

EDx CC-615 Cloud Infrastructure

Week 6

Cloud Infrastructure – Lab Exercise: Migrating a Website to AWS

Objectives

In week 6, we will illustrate concepts of IaaS cloud deployments we have covered so far, by provisioning a simple virtual server environment in Amazon Web Services and migrating a simple website to it. We will:

- Set up the core infrastructure for website migration
- Access the infrastructure servers and install web server software
- Migrate existing simple website to the infrastructure

All students are encouraged to complete the labs. At the end of the week, Audit students will take the weekly exam assessing their understanding of the labs. Verified students will submit Lab Report including the material requested in the lab steps.

General Lab Instructions

- Week 6 and Week 7 labs require that you use your AWS account from edX CC 607, but you will not use any infrastructure that you previously created, and you may delete any existing infrastructure. If for some reason you terminated your AWS account, then you will need to follow these instructions from: [AWS Sign-Up Instructions](#) to create a new AWS account before proceeding with this lab.
- Note that you are responsible for any charges incurred using AWS for this lab. You were provided a credit in edX CC 607, which you may use for this course as well. If you have depleted the Amazon credits, then you will need to provide appropriate payment details.
- Each time you connect to the AWS console, you need to ensure that you are connecting to the region where your resources are created. If you do not see your resources, then you are in the wrong region and need to go to the Global menu at the top right of your screen to change the region. Example: If you created your instances in US-West, then you need to ensure that US-West is selected after you log into the console.
- Do **not** stop your EC2 instances until you have completed the labs for Weeks 6 and 7. If you do, your data and changes will be lost.

Lab Report Requirements

You are required to submit a lab report at the end of the lab. Throughout the steps, you will see indicators labeled with **LAB REPORT:** in which you should incorporate an item from that step

into your report. Examples include screenshots of your progress, URLs for running AWS instances, and summaries of your progress.

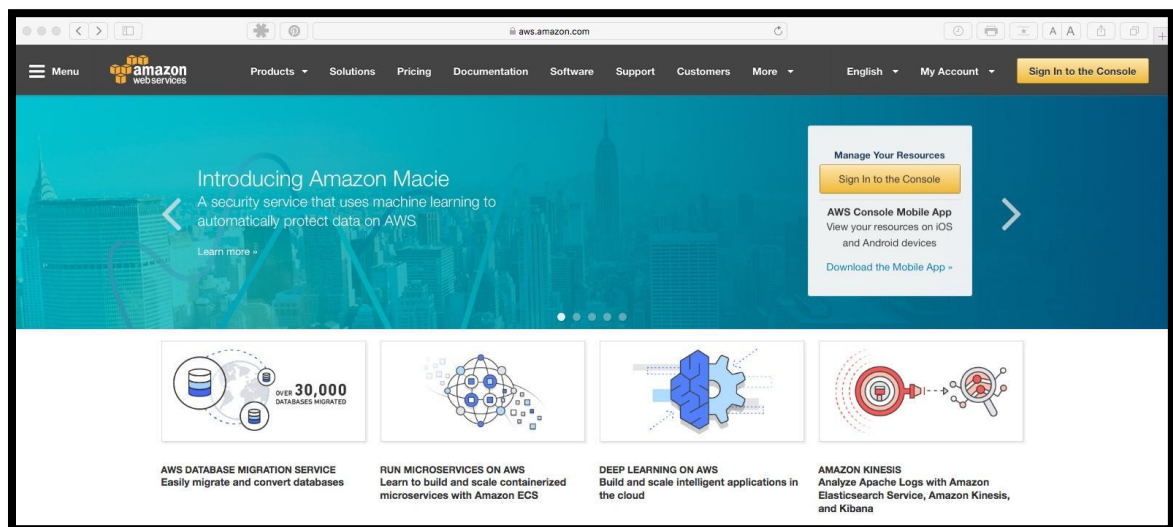
1. Learning Objective: Set up the core infrastructure for website migration.

1. Learning Topic: Set up two virtual server instances

In this part of the lab, you will set up the core infrastructure. This includes your virtual machines (or instances) in AWS Elastic Compute Cloud (EC2), Auto scaling group, and Elastic Load Balancers.

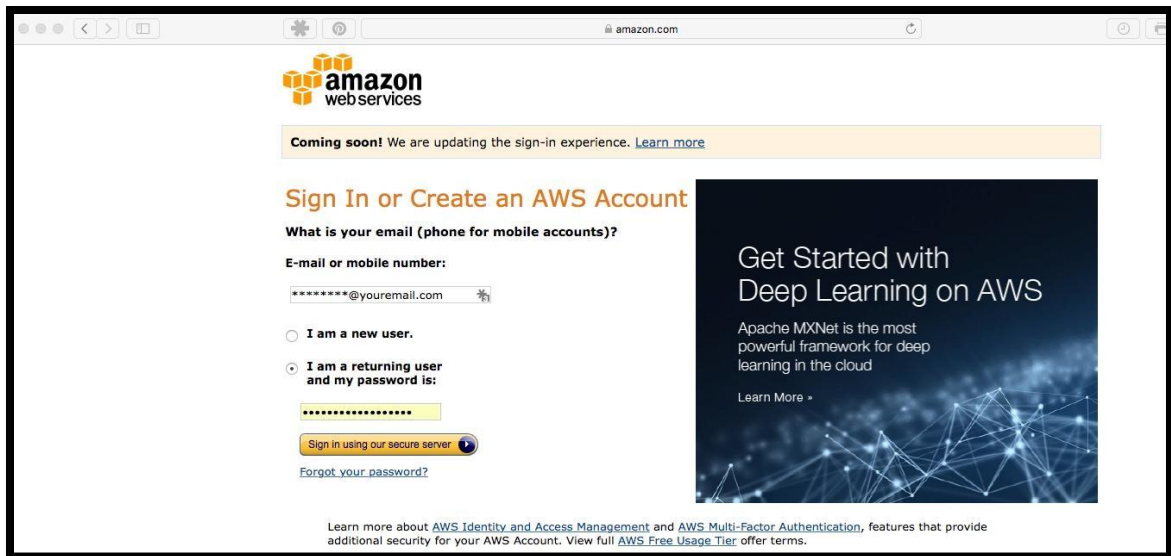
We will start with setting up two identical instances of virtual servers in AWS, using AWS Elastic Compute Cloud (EC2). See <https://aws.amazon.com/ec2/> for more information about EC2.

1. In a web browser, open the AWS Console: <https://aws.amazon.com>



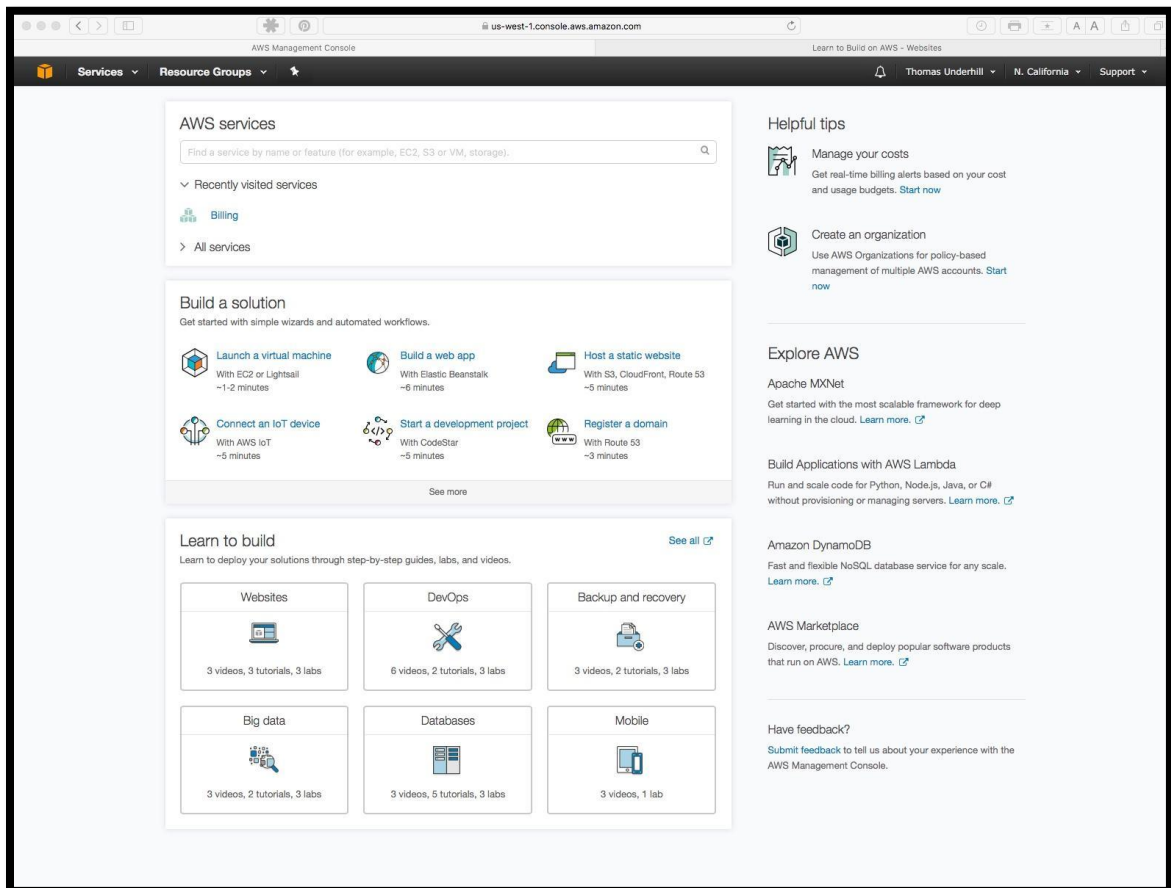
Source: AWS

2. Click the "Sign into the Console" button in the upper right of the screen.
3. Enter your AWS login credentials into the boxes provided.



Source: AWS

4. After you are logged in, you will see an AWS Management Console screen similar to this one.



Source: AWS

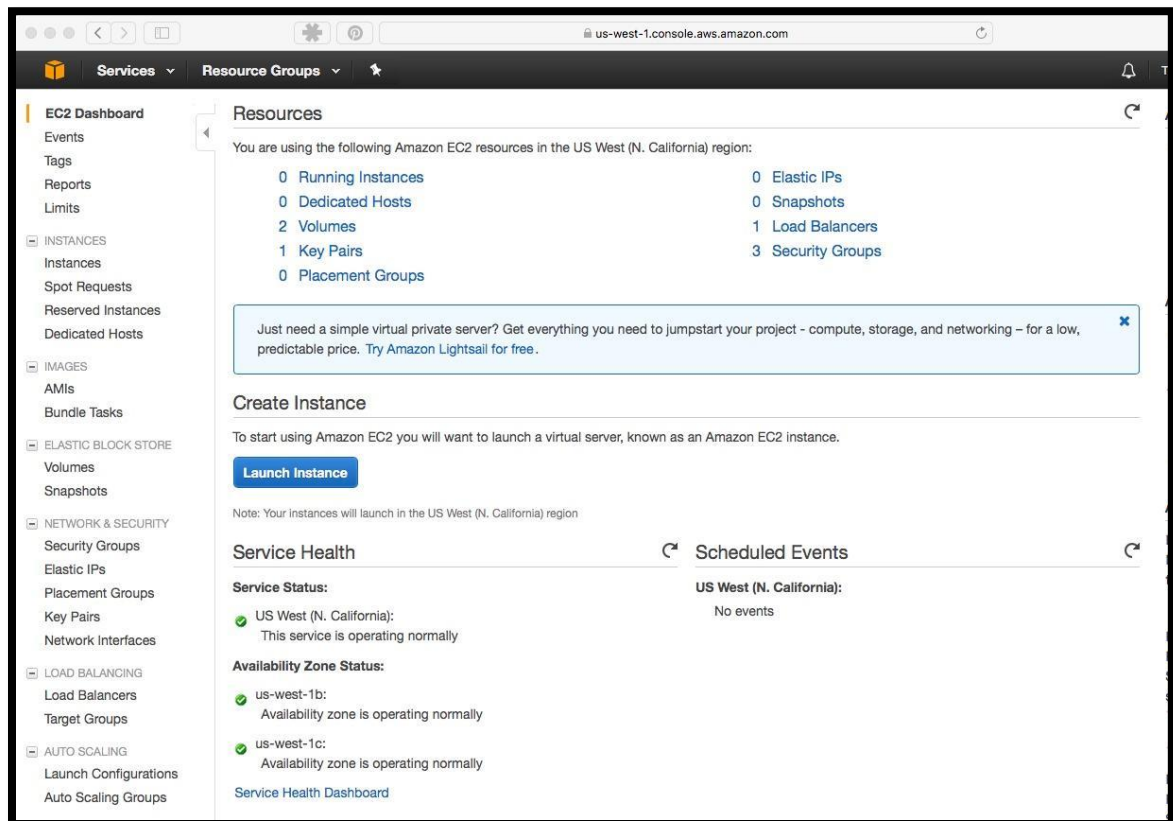
Clicking on the orange cube in the upper left corner will always bring you to the AWS Console home screen.

5. In the AWS services search box, you should type the following and hit Return or Enter on the keyboard.

ec2

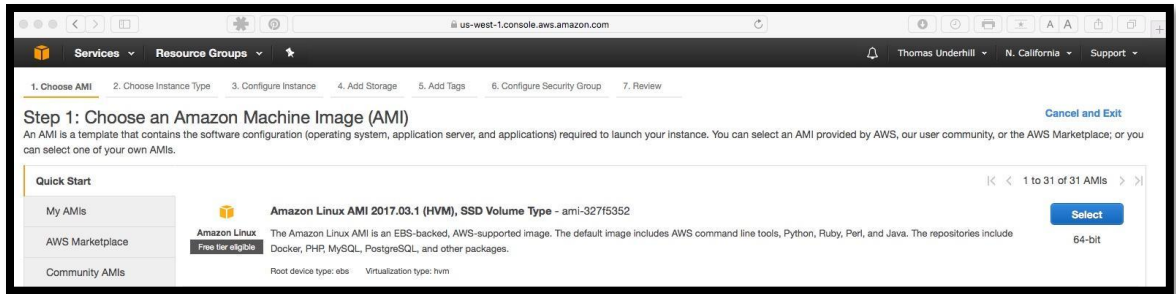
6. You will arrive at the EC2 Management Console (or Dashboard) screen which on the left-and side contains links to the EC2 resources of various types. You can click on these links and inspect the EC2 resources you have.

At this time, click "Launch instance."



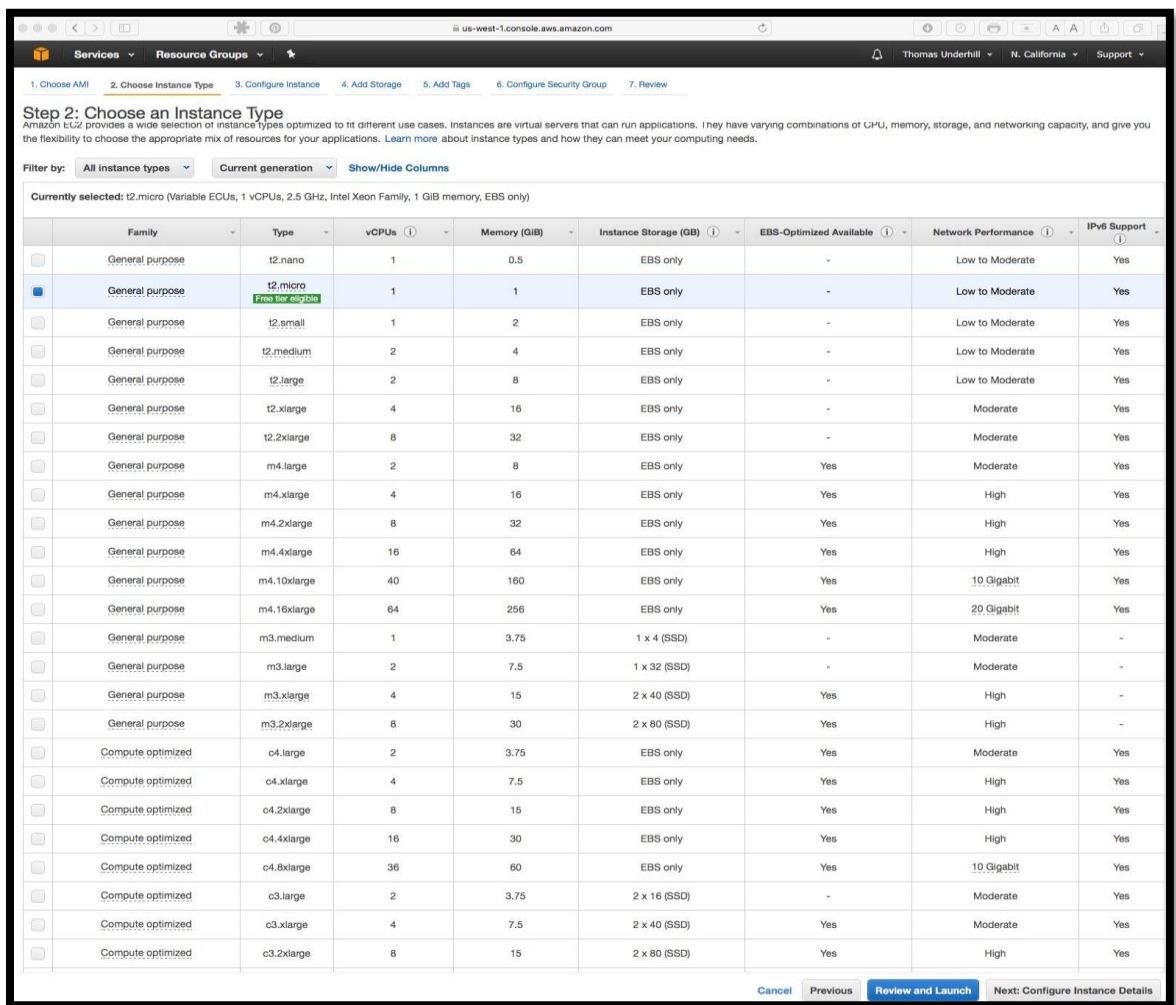
Source: AWS

7. Choose an Amazon Machine Image (AMI). You should choose the Amazon Linux AMI located at the top of the screen.



Source: AWS

8. Choose an Instance Type. Be sure that you are selecting the **t2.micro**, which is Free-tier eligible.



Source: AWS

9. Click on the “Next: Configure Instance Details” button.

10. On the “Configure Instance Details” screen, change the following values from their defaults:

Number of instances: 2

11. Click on the “Launch into Auto Scaling Group” button, and then click “Create Launch Configuration” in the pop-up window that appears.

12. On the “Create Launch Configuration” screen, enter the following information:

Name: CC615 Lab

13. Click “Skip” to review.

14. On the review screen, you will see a warning about security groups and the default making your instances open to any Internet traffic. You do not need to and should not change this security setting for the purposes of this course. *Note: If you are using AWS outside of the scope of this course, then you probably would want to configure the security groups.*

15. On the “Create Launch Configuration” screen, click “Create Launch Configuration.”

16. In the “Select an Existing Key Pair or Create a New Key Pair” box, select “Create a new key pair.” You will use the key pair rather than a password to access your instances. For the key pair name, enter:

cc615-key

17. Click the “Download Key Pair” button.

IMPORTANT: Save this private key file in a secure and accessible location on your computer. You will not be able to download it again from Amazon. You will need this key for future lab activities in CC615. Do not share this key file with anyone else since it will permit that person to SSH into your Amazon instances.

Save the key as the following file name on your computer:

cc615-key.pem

2. Learning Topic: Create auto scaling group

Next, we will create the auto scaling group for our servers. In week 2, we introduced the Compute auto scaling feature, which allows to horizontally scale a group of servers up or down depending on a scaling policy defined by the customer.

See <https://aws.amazon.com/autoscaling/> for more information.

We will keep our auto scaling group at the initial size, and attach a load balancer to it in the next learning topic.

Keeping the auto scale group at fixed size will trigger re-launching a new instance whenever an existing instance is stopped or terminated.

18. On the “Create Auto Scaling Group” screen, specify the following:

Group Name: CC615-Lab-Scaling-Group
Group Size: 2 instances
Network: Leave as the default
Subnet: Select the subnets available to you in your region
Under Advanced Details, leave the defaults

19. Click “Next: Configure Scaling Policies.”

20. On the “Create Auto Scaling Group” screen, select “Keep this group at its initial size.”

21. Click “Review.”

22. Click “Create Auto Scaling Group.”

23. Once created, you can click on “View Your Auto Scaling Groups” to view them.

24. On the “Welcome to Auto Scaling” screen, click on “Auto Scaling Group: 1

LAB REPORT”: Take a screenshot of your “Auto Scaling groups” and insert it into your lab report document.

25. Click “Close.”

26. Click on the Instances link on the left side of the screen.

27. Select the two instances that you created, and go to the “Actions” menu at the top.

LAB REPORT: Take a screenshot of your Instances and insert into your lab report document.

3. *Learning Topic: Create load balancer*

We will now associate a load balancer with the auto scaling group we've created. As discussed in Week 2, the load balancer will distribute incoming requests among members of the group.

28. Go to the AWS console: <https://aws.amazon.com>.

29. Click on "EC2."

30. Click on "Load Balancers" under Load Balancing on the left menu.

31. Click "Create Load Balancer."

32. Select "Classic Load Balancer."

33. Click "Create."

34. On *Step 1: Define Load Balancer Screen*, enter the following:

Load Balancer name: CC615-LB

35. Leave all of the other settings as defaults (indicated below):

Create LB inside DEFAULT SETTING Create an internal load balancer: Unchecked Enable advanced VPC configuration: Unchecked Load balancer protocol: HTTP Load balancer port: 80 Instance Protocol: HTTP Instance port: 80

36. Click "Next: Assign Security Groups."

37. Select the "Autoscaling-Security-Group-1" and unselect any others.

38. Click “Next: Configure Security Settings.”

NOTE: Ignore the security warning; we are **not** using SSL in this course.

39. Click “Next: Configure Health Check.”

NOTE: Ignore the security warning; we are **not** using SSL in this course.

40. On *Step 4: Configure Health Check* screen, enter the following:

Ping Protocol: HTTP Ping Port: 80 Ping Path: /index.html Response Timeout: 5 seconds Interval: 30 seconds Unhealthy threshold: 2 Healthy threshold: 10
--

41. Click “Next: Add EC2 Instances.”

42. Don’t add any instances at this time – make sure that none are selected.

43. Click “Next: Add Tags.”

44. On “Step 6: Add Tags” screen, enter the following:

Key: Name Value: Webserver

45. Click “Review and Create.”

46. On Step 7: Review screen, click “Create.”

LAB REPORT: Take a screenshot of Step 7: Review screen and insert it into your lab report.

47. On the Load Balancer Creation Status screen, click “Close.”

48. Go back to the Auto Scaling Groups screen in the EC2 Dashboard and select the “CC615-Lab-Scaling-Group” auto scaling group you have created in previous steps.

49. On the “Details” tab, click Edit, and then click on the “Load Balancer” box, and select the *CC615-LB* load balancer. It will put the load balancer in charge of balancing your auto scale group.

50. Click “Save.”

51. Go back to the *Load Balancer* screen in the EC2 Dashboard and look for the *DNS name for your new load balancer*.

EXAMPLE: *CC615-LB-1234567890.us-wes-1.elb.amazonaws.com*

LAB REPORT: Record the DNS name for your load balancer in your lab report as both a DNS name and a web service URL as indicated below:

Load Balancer DNS Name: CC615-LB-1234567890.us-wes-1.elb.amazonaws.com

BallotOnline New Web Service URL: <http://CC615-LB-1234567890.us-wes-1.elb.amazonaws.com>

52. Go to the “Security Groups” menu in the EC2 Dashboard, and click on it.

53. Under “Security Groups,” select “Autoscaling-Security-Group-1,” click on the “Inbound” tab, and click “Edit.”

54. Add a new rule for *HTTP* with the following settings:

Protocol: TCP
Port Range: 80
Source: Anywhere

55. Click “Save.”

56. Go to the Instances menu, in the EC2 Dashboard, and click on “Instances.”

57. Under *Instances*, locate the *Public IP Address* for your instances

Example: 64.23.23.43

LAB REPORT: Record the Public IP addresses for each of your instances in your lab report as indicated below:

Public IP Address for Instance 1: 66.23.34.33
Public IP Address for Instance 2: 63.23.34.33

LAB REPORT: Write a two- to three-paragraph summary of your lessons learned while completing this portion of the lab.

Congratulations! You have completed this portion of the lab. You may now go to the EC2 dashboard, click on “Instances, Auto Scale Groups, Load Balancers, and Security Groups” in turn in the list on the left, and inspect your resources.

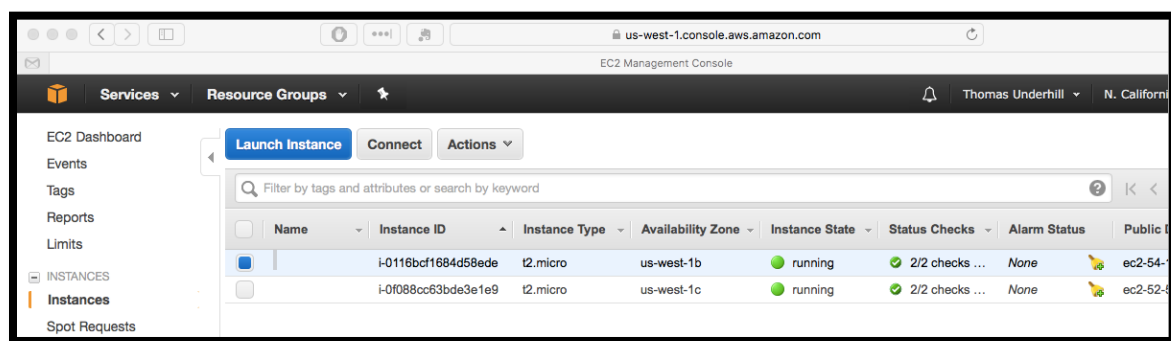
If you click on Volumes under Elastic Block Storage, you will also see two volumes AWS created for your instances.

Do **not** stop your EC2 instances until you have completed the labs for Weeks 6 and 7. If you do, your data and changes will be lost.

2. Learning Objective: Access the infrastructure servers and install web server software

In this part of the lab, you will access the provisioned virtual servers and install the nginx web server which will serve the new BallotOnline website.

1. Open the “AWS Console: <https://aws.amazon.com>”
2. Click on “EC2.”
3. Click on the “Instances” link on the left side of the screen.
4. Click the checkbox to the left of your first instance and then click the “Connect button above your instance list.”



5. You will see a pop-up screen that enables you to connect to your instance either using a Java SSH client or a standalone SSH client. For this step, you will follow the instructions provided by Amazon for your specific instance. Once you have completed the steps provided by Amazon, then you will move to the next step in the lab instructions.

On Windows machine, you can use either the standalone SSH client for Windows called PuTTY (follow the directions from the pop-up screen link), or the Java SSH client from your browser. On a Mac, you can use the ssh command in the Terminal. Each asks for a different way of making the private key file cc615-key.pem available. Amazon will provide very specific instructions for you to follow, and they will be specific to your operating system and instances. Also, see this document for detailed instructions how to use PuTTY and the ssh command: [Connecting to Your EC2 Instance](#).

Once you have successfully logged into your first instance, you should see something like this:

```
nx-01:~ porthos$ ssh -i "cc615-key.pem" ec2-user@ec2-99-99-99-89.us-west-1.compute.amazonaws.com

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

https://aws.amazon.com/amazon-linux-ami/2017.03-release-notes/
 4 package(s) needed for security, out of 8 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-16-0-5 ~]$
```

LAB REPORT: Take a screenshot of the login screen from each of your instances and paste them into your lab report.

6. If you get prompted to install security updates, do so with:

```
sudo yum update
```

7. Install the **nginx web server** on the instance:

```
sudo yum install nginx
```

You will be prompted with:

```
Is this ok [y/d/N]:
```

Enter:

```
y
```

Hit **Return** or **Enter** on the keyboard.

The output on your screen should show the successful installation of nginx.

LAB REPORT: Paste a screenshot of your successful nginx installation into your lab report.

3. Learning Objective: Migrate existing simple website to the infrastructure

Finally, you will migrate the BallotOnline website to the cloud environment you've created, and configure the nginx web server to serve it.

After you complete this lab, you will have two web server instances behind a load balancer with auto-scaling enabled. This will permit the BallotOnline website to remain available even if one of your instances is down for maintenance.

1. Create a directory for the BallotOnline website data:

```
sudo mkdir -p /data/www
```

2. Next, you will download the website files from the legacy webserver. The files has been compressed into website.zip and the URL is www.ballotonline.biz/website.zip

```
cd /data/www  
sudo wget www.ballotonline.biz/website.zip
```

3. Now, you will unzip the file into the /data/www/BallotOnline directory.

```
sudo unzip website.zip -d /data/www/BallotOnline
```

4. Check to make sure that the index.html file was downloaded to the /data/www directory:

```
cat index.html
```

You should see text starting with:

```
<!DOCTYPE html>
```

5. Change the permissions on the `index.html` file to make it less restrictive:

```
sudo chmod 755 /data/www/index.html
```

6. Replace the default nginx configuration file (located in `/etc/nginx/nginx.conf`) with the one that is provided within the zip file that you have downloaded and unzipped.

```
sudo mv /etc/nginx/nginx.conf /etc/nginx/nginx.conf.old  
  
cat /etc/nginx/nginx.conf
```

Note: You should receive an error "No such file or directory"

7. For the next line, you will copy the *nginx.conf* that we have downloaded as part of the `website.zip` file to the correct location.

```
sudo cp /data/www/BallotOnline/nginx.conf /etc/nginx/
```

You should now have a file named *nginx.conf* in the `/etc/nginx` directory.

8. To double-check, type:

```
cat /etc/nginx/nginx.conf
```

You should see the content of *nginx.conf* from the `/etc/nginx` directory.

9. Start *nginx*.

```
sudo nginx
```

NOTE: If you already started `nginx` for some reason, then you reload the configuration with: `sudo nginx -s reload`.

10. From your SSH session, test the web server:

```
curl http://localhost
```

You should see the HTML file for BallotOnline.

11. Log out from the instance:

```
logout
```

12. You have now confirmed that your basic nginx web server configuration is correct for your first instance. Go to the *Public IP Address for Instance 1* that is in your lab report via a web browser to see if you can access the index.html file:

<http://NN.NN.NN.NN/index.html>

13. You will now configure your second instance. **Repeat Steps 4 through 7 from the previous Learning Objective to install nginx, and Steps 1 through 8 15 from this Learning Objective to migrate the BallotOnline legacy website and configure nginx, replacing references to the first instance with the second instance.**

14. After you have configured the second instance and confirmed that it is working properly, then you will test your load balancer.

Pull the load balancer name from your Step 5 lab report:

<http://LOADBALANCERNAME>

Example:

<http://cc615-lb-123456.us-west-1.elb.amazonaws.com/index.html>

LAB REPORT: Paste the Load Balancer URL and a screenshot of the new BallotOnline website into your lab report. Write a one- to two-page summary of your experiences completing this lab.

Congratulations! You have successfully performed a test migration of BallotOnline's website to AWS and completed this lab.

IMPORTANT: Do **not** stop or terminate your EC2 instances since you will need them for the Week 7 project.

OPTIONAL: Because we have assigned our load balancer to the auto scale group, every time an instance in the group goes down the auto-scale mechanism will stand up another instance and add it to the load balancer.

However, that instance will not have the nginx web server installed and running, nor the BallotOnline website migrated to it. To resolve this, you would have to create the private Amazon Machine Image off the fully configured instance, and then re-create the load balancer and auto scale group using that image for provisioning virtual servers launched into auto scale group. You may try to do this as an optional activity, which will be not graded.

Week 6 Lab Report

Please include information requested under the LAB REPORT: labels in the lab steps into Week 6 Lab Report.

WEEK 6 SLIDES:

In addition to the overview slide, just a couple of slides with this content:

In Week 6 labs you will accomplish the following:

- Provision 2 AWS instances, using Linux Amazon Machine Image.
- Create Launch Configuration, and create a key pair for SSH access. The instances will be launched to an auto scale group.
- Create the auto scale group of fixed size, with Security Group
- Configure Security Group for HTTP access
- Create the load balancer, and assign it to the auto scale group
- Access the instances, and install the open source nginx web server on them
- Migrate the ballotonline.biz website to both instances, and configure the web server to serve it.
- Test the workings of individual instances and the load balancer.