Architecting on AWS
Lab Guide
Version 4.5A

AWS-100-ARC-45A-EN

Corrections or feedback on the course, please email us at:
aws-course-feedback@amazon.com.
For all other questions, contact us at:
https://aws.amazon.com/contact-us/aws-training/.

Table of Contents

Starting your labs

Starting your labs

Introduction

The labs for this course are run in the qwikLABS lab environment. qwikLABS offers a complete end-to-end cloud platform for hands-on software training lab creation, management and consumption. The labs can be delivered anywhere, anytime, on-demand.

Accessing your labs

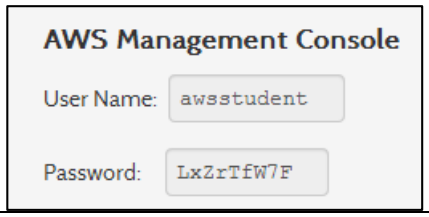
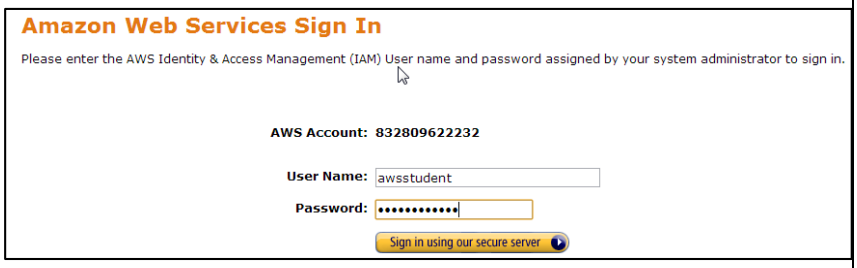
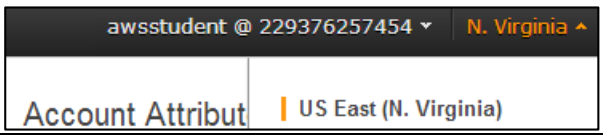
To access your lab environment in qwikLABS:

Step	Action
1	Open the qwikLABS link provided by your instructor.
2	Log in with your credentials or register for an account.
3	You should have access to the classroom for your course. The classroom will contain links to each of the labs for your class. Follow these steps to launch each of the labs in your course as directed by your instructor.
4	To the right of the lab title, click the **Start Lab** button to launch your qwikLABS. **Note**: If your lab requires setup, a status bar shows the progress of the lab environment creation process. The AWS Management Console is accessible during lab creation, but your AWS resources may not be fully available until the process is complete. 
5	On the lab details page, notice the lab properties. • **Time Remaining** – The amount of time left to complete the lab. • **Setup Time** – The estimated time to set up the lab environment. • **Duration** – The expected completion time for the lab. • **Access** – The time the lab will run before automatically shutting down.

Continued on next page

Starting your labs, Continued

Accessing your labs (continued)

Step	Action
6	In the **AWS Management Console** section of the qwikLABS page, copy the **Password** to the clipboard. 
7	Click the **Open Console** button.
8	Log into the AWS Management Console using the following steps. 1. In the **User Name** field type **awsstudent**. 2. In the **Password** field, paste the password copied from the lab details page. 3. Click **Sign in using our secure server**.  **Note**: The AWS account is automatically generated by qwikLABS. Also, the login credentials for the awsstudent account are provisioned by qwikLABS using AWS Identity Access Management.
9	In the AWS Management Console, click **EC2**.
10	Make a note of the **AWS Region** in the AWS Management Console menu bar (beside your User Name). 

Lab 1: Working with Amazon Identity and Access Management

Lab Overview

Introduction

In this lab, you will use the Amazon Identity and Access Management service to create users and roles within an AWS environment. You will then test the permissions of these users and roles to verify that they can only perform the specified actions within the AWS environment.

Allotted time

The allotted time for this lab is listed as follows:

Component	Time
Overview	5 minutes
Lab	30 minutes
Total:	**35 minutes**

Prerequisites

This lab requires:

- Access to a notebook computer with Wi-Fi on a Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat) system
  - The qwikLABS lab environment is not accessible on an iPad or tablet device, but you may use these devices to access the student guide (PDF).
- For Microsoft Windows users: Administrator access to the computer
- An Internet browser such as Chrome, Firefox, or IE 9 (previous versions of Internet Explorer are not supported)

Objectives

After completing this lab, you will be able to:

- Familiarize yourself with the Identity and Access Management (IAM) Console.
- Grant permissions to users to use a specific AWS service.
- Grant limited permissions to users in a group.
- Locate and use the IAM sign-in URL.

Creating IAM Users

Introduction

In this part of the lab, you will create two users in IAM, create and manage their passwords and then assign a specific security policy to one of them.

Task 1.1: Create users in IAM

To start, you first need to create the users:

Step	Action
1.1.1	In the **AWS Management Console**, on the **Services** menu, click **IAM**.
1.1.2	In the navigation pane, click **Users**.
1.1.3	Click **Create New Users**.
1.1.4	In the first **Enter User Names** box, type **S3TestUser**
1.1.5	In the second **Enter User Names** box, type **EC2TestUser**
1.1.6	Select the **Generate an access key for each user** option if it is not already selected.
1.1.7	Click **Create**.
1.1.8	Click **Download Credentials** to download the users' credentials.
1.1.10	Return to your browser and click **Close**.

Task 1.2: Create a password

By default, users that you create do not have access to the AWS Management Console. To grant this access, you need to create a password for the user:

Step	Action
1.2.1	Return to the **Users** menu in the IAM Management Console.
1.2.2	Select the **S3TestUser** check box.
1.2.3	In the **User Actions** drop-down list, click **Manage Password**.
1.2.4	Select the **Assign an auto-generated password** option. Leave the rest as the default. Click **Apply**. A new page confirms that IAM has generated the user's password.
1.2.5	Click **Show User Security Credentials**.
1.2.6	Open the **credentials** file you saved earlier and add a **Password** column to the right of the other two columns.
1.2.7	Paste the password generated by IAM for **S3TestUser** into the **Password** column of the **S3TestUser** entry.
1.2.8	In the IAM Management Console, click **Close**.

Continued on next page

Creating IAM Users, Continued

Task 1.2: Create a password, continued

Step	Action
1.2.9	IAM asks you to confirm the closing of the window, because you haven't downloaded the user's password. Because you copied the password to an existing file, click **Close** again to close the window.
1.2.10	Repeat steps **1.2.1** through **1.2.9** (skipping step **1.2.6**) for **EC2TestUser**.

Task 1.3: Set permissions

Grant full permissions to only Amazon S3:

Step	Action
1.3.1	Return to the **Users** menu in the IAM Management Console if you are not already there.
1.3.2	From the list of users, click the name of **S3TestUser** to open the **Details** page.
1.3.3	Scroll down to the **Permissions** section.
1.3.4	Click **Attach Policy**.
1.3.5	Scroll through the list of policies until you locate the entry marked **AmazonS3FullAccess** under the **Policy Name** column. Select the check box for this entry. This policy grants the selected user full access to all Amazon S3 functions. **Note** You can also use the **Search** box at the top of the list to locate this or any policy more easily.
1.3.6	Click **Attach Policy**. Verify that **AmazonS3FullAccess** is listed under **Policy Name** within the **Permissions** group.

Creating an IAM Group

Introduction

In this part of the lab, you are going to create a group who has full permissions (Start, Stop, Terminate, and so on) with Amazon EC2 instances. Now, instead of attaching a policy directly to each user, you will create a group that has these permissions, and then add a user to that group.

Task 1.4: Create a user group

To create a user group:

Step	Action
1.4.1	In the navigation pane, click **Groups**.
1.4.2	Click **Create New Group**.
1.4.3	In the **Group Name** box, type **EC2TestGroup**
1.4.4	Click **Next Step**.
1.4.5	Select the **AmazonEC2FullAccess** policy check box from the list. This policy grants any members of the group full access to all Amazon EC2 functions.
1.4.6	Click **Next Step**.
1.4.7	Click **Create Group**.
1.4.8	Select the **EC2TestGroup** check box.
1.4.9	In the **Group Actions** drop-down list, click **Add Users to Group**.
1.4.10	Select **EC2TestUser**, and then click **Add Users**.
1.4.11	In the **Groups** home page, click the name of the **EC2TestGroup** group to display the details of that group.
1.4.12	To verify that **EC2TestUser** has been added to the group, confirm that **EC2TestUser** is listed under **User** in the **Users** group.

Creating an IAM Role

Introduction

In this part of the lab, you will create a role within IAM. A role is an IAM entity that defines a set of permissions for making AWS service requests. IAM roles are not associated with a specific user or group. Instead, roles are ***assumed*** by trusted entities such as IAM users, applications, or AWS services such as Amazon EC2.

You are going to create an IAM role that allows anyone using that resource to have "describe" permissions to Amazon EC2 instances when it is assigned to a resource. This means that the user can list the Amazon EC2 instances that are running, but cannot start, stop, or otherwise change them.

Task 1.5: Create a role

To create an IAM role:

Step	Action
1.5.1	In the navigation pane, click **Roles**.
1.5.2	Click **Create New Role**.
1.5.3	In the **Role Name** box, type **EC2Describe**
1.5.4	Click **Next Step**.
1.5.5	Verify that the **AWS Service Roles** option is selected.
1.5.6	Locate **Amazon EC2** near the top of the list, and click **Select**.
1.5.8	Select the **AmazonEC2ReadOnlyAccess** policy from the list. This policy grants all entities that assume this role read-only access to Amazon EC2 instances associated with this account. In other words, entities that assume this role will be able to describe and list all existing Amazon EC2 instances, but will not be able to create new instances, or stop or terminate existing instances.
1.5.9	Click **Next Step**.
1.5.10	Click **Create Role**.

Testing IAM Users

Introduction

You should now have the following:

- A user who has full access to only Amazon S3 resources.
- A user who has full access to only Amazon EC2 resources.
- A role that has read access to only Amazon EC2 resources.

Next, you'll test each of these to see how they function. Before you start, obtain the URL associated with your main AWS account for this lab.

Task 1.6: Test the S3 user

You'll need the password you created for S3TestUser.

Step	Action
1.6.1	In the navigation pane, click **Dashboard**, and then copy the URL shown under **IAM users sign-in link** to a text file somewhere on your local computer.
1.6.2	Open a different web browser such as Firefox or Chrome. **Note** You can open another tab within the same browser; however, a new session will log off **awsstudent**; therefore, you will need to log in again using the password provided by qwikLABS. If you have been using Firefox, using a Chrome or Safari web browser will maintain the session for **awsstudent** while you test **S3TestUser**.
1.6.3	Navigate to the AWS Account Alias URL that you copied in step **1.6.1**.
1.6.4	In the **User Name** box, type **S3TestUser**
1.6.5	In the **Password** box, type the password that you saved in the **credentials** file.
1.6.6	Click **Sign In**. The AWS Management Console opens.
1.6.7	On the **Services** menu, click **EC2**. Because this user does not have any EC2 permissions, messages in the center pane state that you are not authorized to describe various aspects of an EC2 instance.
1.6.8	In the navigation pane, click **Instances**. This message appears: *"An error occurred fetching instance data. You are not authorized to perform this operation."* This is because the user that you have used for login has no permissions to Amazon EC2. Next, you will verify whether the user has permissions for Amazon S3.

Task 1.7: Create an S3 bucket

To create an Amazon S3 bucket:

Step	Action
1.7.1	On the **Services** menu, click **S3**.
1.7.2	Click **Create Bucket**.

Continued on next page

Testing IAM Users, Continued

Task 1.7: Create an S3 bucket, continued

Step	Action
1.7.3	In the **Create a Bucket – Select a Bucket Name and Region** dialog box: • In the **Bucket Name** box, type a unique bucket name (e.g., *awsst-lab01-1126***,** with no uppercase letters). • In the **Region** drop-down list, select a region you want to create a bucket in. **Note** The bucket name you choose must be unique across all existing bucket names in Amazon S3. One way to help ensure uniqueness is to prefix your bucket names with the name of your organization. Bucket names must comply with certain rules: • Bucket names must be at least 3 characters and no more than 63 characters long. • Bucket names can contain lowercase letters, numbers, and hyphens (Note: not uppercase letters). Each label must start and end with a lowercase letter or a number.
1.7.4	Click **Create**.
1.7.5	Click the bucket you just created. The bucket is currently empty.
1.7.6	Click **Upload**.
1.7.7	Click **Add Files**. A file selection dialog box opens.
1.7.8	Select a file from your computer that you want to upload, and then click **Open**.
1.7.9	Click **Start Upload**. The file should upload successfully. This demonstrates that **S3TestUser** has permission to create buckets and upload files to **Amazon S3**.

Task 1.8: Test the EC2 user

In this step, you will log in as the user EC2User and verify that this user has full permissions to work with Amazon EC2.

Step	Action
1.8.1	Open a new browser window or tab, and then paste the URL that you copied in step **1.6.1** in the address bar.
1.8.2	In the **User Name** box, type **EC2TestUser**
1.8.3	In the **Password** box, type the password that you saved in the **credentials** file.
1.8.4	Click **Sign In**. The AWS Management Console opens.
1.8.5	On the **Services** menu, click **EC2**.
1.8.6	In the navigation pane, click **Instances**. You should not see any error messages. This demonstrates that **EC2TestUser** has permission to work with Amazon EC2.
1.8.7	On the **Services** menu, click **S3**. Note that you cannot access any S3 resources.

Ending your lab

If you choose not to complete the optional portion of this lab, follow the instructions in Appendix B to end your lab.

Optional

Introduction

This part of the lab is a challenge; step-by-step instructions are not provided. Use online documentation to help you if necessary.

http://docs.aws.amazon.com/IAM/latest/UserGuide/Using_WorkingWithGroupsAndUsers.html.

Challenge 1

First, create an IAM role for Amazon EC2. Log in to the AWS Management Console as EC2TestUser, and try to launch an Amazon EC2 instance with the IAM Role set to the role that you created previously.

Discussion

Were you able to launch an instance into the new role? Why?

Instance Profile

An instance profile is a container for an IAM role. Instance profiles are used to pass role information to an Amazon EC2 instance when the instance starts.

To launch an instance with a role, the developer must have permission to launch Amazon EC2 instances and permission to pass IAM roles.

To learn more about **Instance Profile**, see the online documentation at http://docs.aws.amazon.com/IAM/latest/UserGuide/instance-profiles.html.

Continued on next page

Optional, Continued

Challenge 2

Log back in as the lab student user (awsstudent). Modify the EC2 test user's permissions in order to grant them permission to list instance profiles (`iam:ListInstanceProfiles`) as well as pass role (`iam:PassRole`). Then, try to launch an Amazon EC2 instance into the new role that you created.

Hint Refer to the online User Guide for AWS Identity and Access Management at http://docs.aws.amazon.com/IAM/latest/UserGuide/PermissionsAndPolicies.html.

To learn more about granting access to applications on Amazon EC2 instances, see the online documentation at http://docs.aws.amazon.com/IAM/latest/UserGuide/role-usecase-ec2app.html.

Ending your lab

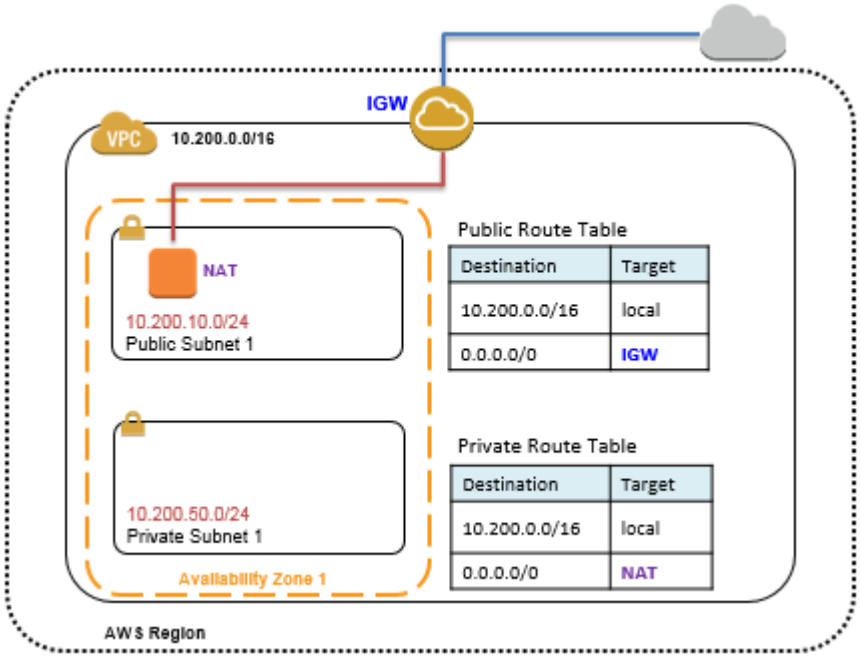
When you are done, follow the instructions in Appendix B to end your lab.

Lab 2: Creating Your First Virtual Private Cloud

Lab Overview

Introduction

In this lab, you will create a basic virtual private cloud (VPC) and extend it to produce a customized network.



Allotted time

The allotted time for this lab is listed as follows:

Component	Time
Overview	1 minute
Lab	40 minutes
Total:	**41 minutes**

Continued on next page

Lab Overview, Continued

Prerequisites

This lab requires:

- Access to a notebook computer with Wi-Fi on a Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat) system
  - The qwikLABS lab environment is not accessible on an iPad or tablet device, but you may use these devices to access the student guide (PDF)
- For Microsoft Windows users: Administrator access to the computer
- An Internet browser such as Chrome, Firefox, or IE9 (previous versions of Internet Explorer are not supported)
- An SSH client such as PuTTY

Objectives

After completing this lab, you will be able to:

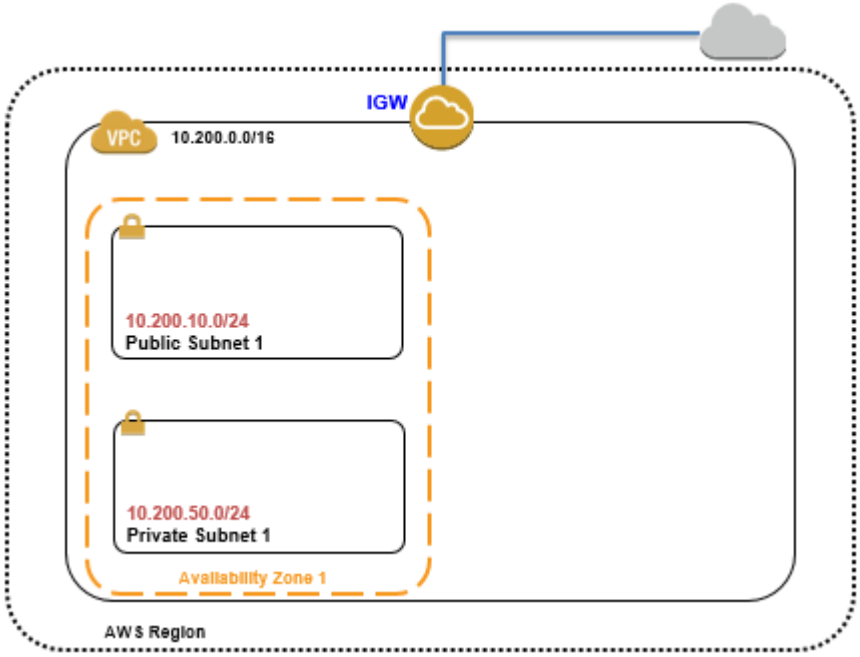
- Create a virtual private cloud (VPC).
- Create subnets within an Availability Zone.
- Create a Network Address Translation (NAT) instance
- Attach an Internet gateway (IGW) to your VPC.
- Create route tables.

Creating the Base VPC

Introduction

When you first sign in to the AWS Management Console and launch **VPC Dashboard**, you will notice that there is an existing VPC; this is the default VPC. A **default VPC** is a logically isolated virtual network in the AWS cloud that is automatically created for your AWS account the first time you provision Amazon EC2 resources. When you launch an instance without specifying a subnet ID, your instance will be launched in your default VPC.

In this part of the lab, you will create a VPC with subnets and a user-specified IP address range.



Continued on next page

Creating the Base VPC, Continued

Task 2.1: Create your VPC

To create your VPC:

Step	Action
2.1.1	In the **AWS Management Console**, on the **Services** menu, click **VPC**. **Note** You can select your desired region from the drop-down list on the navigation bar. For now, let it remain as the default.
2.1.2	In the navigation pane, click **Your VPCs**.
2.1.3	Click **Create VPC**.
2.1.4	In the **Create VPC** dialog box: • In the **Name tag** box, type **Lab VPC** • In the **CIDR block** box, type **10.200.0.0/16** • Leave the **Tenancy** value as **Default**.
2.1.5	Click **Yes, Create**. You should see a new VPC named **Lab VPC** with a **VPC ID** assigned to it (e.g., *vpc-530de336*).

Task 2.2: Attach an Internet gateway

To attach an Internet gateway to your new VPC:

Step	Action
2.2.1	In the navigation pane, click **Internet Gateways**.
2.2.2	Click **Create Internet Gateway**.
2.2.3	In the **Name tag** box, type **Lab VPC Gateway**
2.2.4	Click **Yes, Create**. **Result** At this point, the newly created **Lab VPC Gateway** is not attached to your VPC. Note the ID (e.g., *igw-912a31f3*).
2.2.5	Select the newly created **Lab VPC Gateway**, and then click **Attach to VPC**.
2.2.6	In the **Attach to VPC** dialog box, in the **VPC** drop-down list, select the **Lab VPC** that you created in Task 2.1.
2.2.7	Click **Yes, Attach**. The **State** for the **Lab VPC Gateway** should be changed to *attached*, and the VPC ID in the **VPC** column should match your **Lab VPC**.

Creating Subnets

Introduction

You have complete control over your virtual networking environment, including selection of your own IP address range and subnets. A subnet is a segment of a VPC's IP address range where you can place groups of isolated resources.

In this task, you are going to configure your VPC so that it:

- Spans two Availability Zones (AZs) so you can distribute applications across these zones to architect for application durability and availability.
- Includes two subnets within each Availability Zone (AZ). Public subnets can route directly to the Internet. Private subnets can communicate with any other subnet within the VPC, but there is no direct access between private subnets and the Internet.

Task 2.3: Create subnets in your VPC

To create subnets inside your VPC:

Step	Action
2.3.1	In the navigation pane, click **Subnets**.
2.3.2	Click **Create Subnet**.
2.3.3	In the **Create Subnet** dialog box: • In the **Name tag** box, type **Public Subnet 1** • In the **VPC** drop-down list, click **Lab VPC** (10.200.0.0/16|Lab VPC). • In the **Availability Zone** drop-down list, select the first AZ (e.g., *us-west-1a*). In the **CIDR block** box, type **10.200.10.0/24**
2.3.4	Click **Yes, Create**. You should be able to see **Public Subnet 1** listed in the table.
2.3.5	Repeat the steps to create another subnet with the following configuration: • In the **Name tag** box, type **Private Subnet 1** • In the **VPC** drop-down list, click **Lab VPC** (10.200.0.0/16|Lab VPC). • In the **Availability Zone** drop-down list, select the same AZ as for **Public Subnet 1**, which was the first AZ listed (e.g., *us-west-1a*). In the **CIDR block** box, type **10.200.50.0/24**
2.3.6	Click **Yes, Create**.

Configuring Route Tables

Introduction

A route table contains a set of rules called **routes** that are used to determine where network traffic is directed. Each subnet in your VPC must be associated with a route table; the table controls the routing for the subnet. A subnet can be associated with only one route table at a time, but you can associate multiple subnets with the same route table.

When you create a VPC, it automatically has a main route table. Initially, the main route table contains only a single route: a local route that enables communication within the VPC. If you don't explicitly associate a subnet with a route table, the subnet is implicitly associated with the main route table.

Task 2.4: Configure a route table

To create a route table that allows incoming and outgoing traffic through the Internet gateway you created earlier:

Step	Action
2.4.1	In the navigation pane, click **Route Tables**.
2.4.2	Click **Create Route Table**.
2.4.3	In the **Create Route Table** dialog box: • In the **Name tag** box, type **Public Route** • In the **VPC** drop-down list, click **Lab VPC**.
2.4.4	Click **Yes, Create**.
2.4.5	Select the **Public Route** you just created, and then click the **Routes** tab in the lower pane of the console.
2.4.6	Click **Edit**.
2.4.7	Click **Add another route**.
2.4.8	In the **Destination** box, type **0.0.0.0/0** Click in the **Target** box, and then select the **Lab VPC Gateway** that you created earlier (the ID starts with *igw-*).
2.4.9	Click **Save**.

Configuring Route Tables, Continued

Task 2.5: Associate the route table with subnets

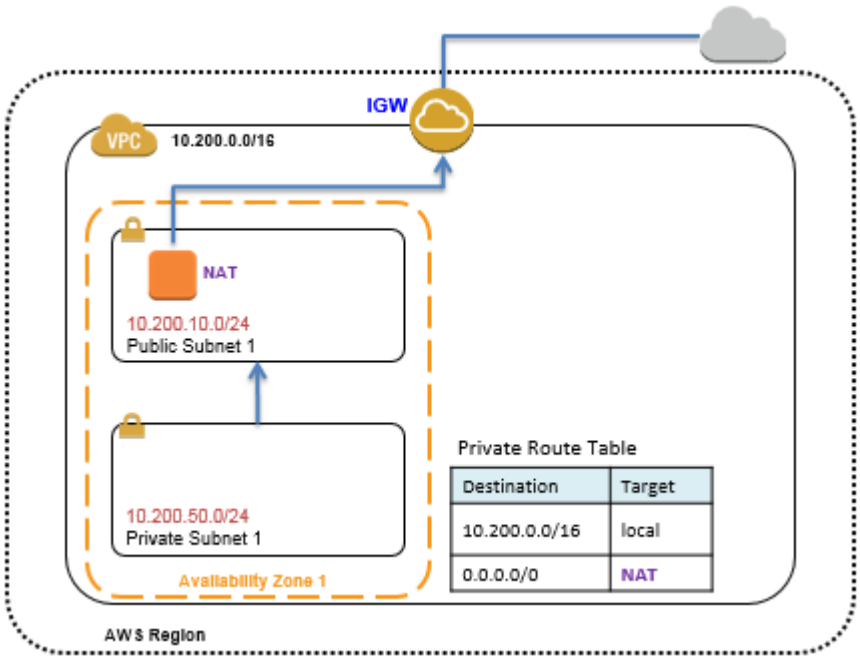
To associate your public subnet with a route table to allow incoming and outgoing data flow:

Step	Action
2.5.1	With **Public Route** selected, click the **Subnet Associations** tab.
2.5.2	Click **Edit**.
2.5.3	Select the check box for **Public Subnet 1** (**CIDR** range of **10.200.10.0/24**).
2.5.4	Click **Save**. You should now be able to see only the private subnets in a table labeled "*The following subnets have not been explicitly associated with any route tables and are therefore using the main route table*."
2.5.5	Click **Create Route Table**.
2.5.6	In the **Create Route Table** dialog box: • In the **Name tag** box, type **Private Route** • In the **VPC** drop-down list, click **Lab VPC**.
2.5.7	Click **Yes, Create**.
2.5.8	With **Private Route** selected, click the **Subnet Associations** tab if it is not already selected. **Note** Your **Private Subnet 1** is currently associated with the **Main** route table by default.
2.5.9	Click **Edit**.
2.5.10	Select the check box for **Private Subnet 1** (**CIDR** range of **10.200.50.0/24**).
2.5.11	Click **Save**.

Creating a NAT Instance and a Private Instance

Introduction

In this part of the lab, you will create a **Network Address Translation (NAT)** server that allows servers in the private subnet to initiate outbound connections to the Internet to download software and access Internet services such as Amazon S3. It does not allow systems on the Internet to initiate inbound connections to servers in the private subnet. The Public IP address assigned to the NAT server allows it to communicate with the Internet.



The route table sends the traffic from the instances in the private subnet to the NAT instance in the public subnet. The NAT instance sends the traffic to the Internet Gateway for the VPC.

Task 2.6: Create a NAT instance

To create a NAT instance in Public Subnet 1:

Step	Action
2.6.1	On the **Services** menu, click **EC2**.
2.6.2	Click **Launch Instance**.

Continued on next page

Creating a NAT Instance and a Private Instance, Continued

Task 2.6: Create a NAT instance, continued

Step	Action
2.6.3	To launch a new instance, you first need to select an Amazon Machine Image (AMI), which is a preconfigured template for an instance in the cloud. From the **Quick Start** menu, in the row for the first **Amazon Linux AMI**, click **Select**.
2.6.4	On the **Choose an Instance Type** page, you can select the family for your image, which determines how much RAM, storage, and processing speed your instance will have. To accept the default (**t2.micro**), click **Next: Configure Instance Details**.
2.6.5	On the **Configure Instance Details** page, make these selections: • In the **Network** drop-down list, select **Lab VPC**. • In the **Subnet** drop-down list, select **Public Subnet 1** (10.200.10.0/24). • In the **Auto-assign Public IP** drop-down list, select **Enable**.

Continued on next page

Creating a NAT Instance and a Private Instance, Continued

Task 2.6: Create a NAT instance, continued

Step	Action

2.6.6	Click **Advanced Details** to expand it. Copy the contents of the user data script given below from the associated command reference file for this lab, and paste the script into the **User data** box.
	
	`#!/bin/sh`
	`echo 1 > /proc/sys/net/ipv4/ip_forward`
	`echo 0 >`
	`  /proc/sys/net/ipv4/conf/eth0`
	`  /send_redirects`
	`  /sbin/iptables -t nat -A POSTROUTING`
	`  -o eth0 -s 0.0.0.0/0 -j MASQUERADE`
	`/sbin/iptables-save >`
	`  /etc/sysconfig/iptables`
	`mkdir -p /etc/sysctl.d/`
	`cat <<EOF > /etc/sysctl.d/nat.conf`
	`net.ipv4.ip_forward = 1`
	`net.ipv4.conf.eth0.send_redirects = 0`
	`EOF`
	
	This Linux shell script configures your server as a NAT server by enabling IP forwarding on the machine and by enabling IP masquerading so that the NAT server can make external requests on behalf of internal servers.

2.6.7	Click **Next: Add Storage**.
2.6.8	Accept the default, and click **Next: Tag Instance**.
2.6.9	In the **Value** box, type **NAT**
2.6.10	Click **Next: Configure Security Group**.
2.6.11	For **Assign a security group**, the **Create a new security group** option should be selected. • In the **Security group name** box, type **NAT-SG** • In the **Description** box, type **NAT Security Group**

Continued on next page

Creating a NAT Instance and a Private Instance, Continued

Task 2.6: Create a NAT instance, continued

Step	Action
2.6.12	Click **Add Rule**. • In the **Type** drop-down list, select **All traffic**. • In the **Source** drop-down list, select **Anywhere**.
2.6.13	Click **Review and Launch**. When prompted, accept the default and click **Next**.
2.6.14	Review the settings and then click **Launch**.
2.6.15	When prompted, accept the qwikLABS keypair, select the acknowledgement check box, and then click **Launch Instances**.
2.6.16	Click **View Instances**.
2.6.17	Select the **NAT** server you just created.
2.6.18	In the **Actions** drop-down list, point over **Networking**, and in the **Networking** drop-down list, click **Change Source/Dest. Check**.
2.6.19	On the **Enable Source/Destination Check** dialog box, click **Yes, Disable**.

Task 2.7: Add NAT in the Private Route table

To edit the settings of your Private Route table to send Internet-bound traffic to your NAT:

Step	Action
2.7.1	On the **Services** menu, click **VPC**.
2.7.2	In the navigation pane, click **Route Tables**.
2.7.3	Select **Private Route** from the list, and then click the **Routes** tab in the lower pane. There should be only one entry for **local**.
2.7.4	Click **Edit**.
2.7.5	Click **Add another route**. • In the **Destination** box, type **0.0.0.0/0** • In the **Target** box, type **NAT** to point to the instance that you created earlier, and then select it.
2.7.6	Click **Save**.

Task 2.8: Create a Private EC2 Instance

Step	Action
2.8.1	On the **Services** menu, click **EC2**.
2.8.2	Click **Launch Instance**.
2.8.3	From the **Quick Start** menu, in the row for the first **Amazon Linux AMI**, click **Select**.

Creating a NAT Instance and a Private Instance, Continued

Task 2.8: Create a Private EC2 Instance, continued

Step	Action
2.8.4	To accept the default (**t2.micro**), click **Next: Configure Instance Details**.
2.8.5	On the **Configure Instance Details** page, make these selections: • In the **Network** drop-down list, select **Lab VPC**. • In the **Subnet** drop-down list, select **Private Subnet 1** (10.200.50.0/24). • In the **Auto-assign Public IP** drop-down list, select **Disable**.
2.8.6	Click **Next: Add Storage**.
2.8.7	Accept the default, and click **Next: Tag Instance**.
2.8.8	In the **Value** box, type **Private Instance**
2.8.9	Click **Next: Configure Security Group**.
2.8.10	For **Assign a security group**, the **Create a new security group** option should be selected. • In the **Security group name** box, type **Private-EC2** • In the **Description** box, type **Private EC2 instance security group**
2.8.11	There should already be an **SSH** rule. • In the **Source** drop-down list, select **Custom IP**. • In the box to the right of **Custom IP**, type **sg** • A list of your security groups will appear. Select the **NAT-SG** security group from the list.
2.8.12	Click **Review and Launch**. When prompted, accept the default and click **Next**.
2.8.13	Review the settings and then click **Launch**.
2.8.14	When prompted, accept the qwikLABS key pair, select the acknowledgement check box, and then click **Launch Instances**.
2.8.15	Click **View Instances**.
2.8.16	Select the **NAT** instance.
2.8.17	From the **Description** tab, note the **Public IP** of the instance.
2.8.18	Select the instance named **Private Instance**.
2.8.19	From the **Description** tab, note the **Private IP** of the instance.

Connecting to Your NAT Instance

Introduction

In this part of the lab, you will connect to the NAT instance that you launched earlier.

Task 2.9: Download a key pair

To download the key pair file generated by qwikLABS:

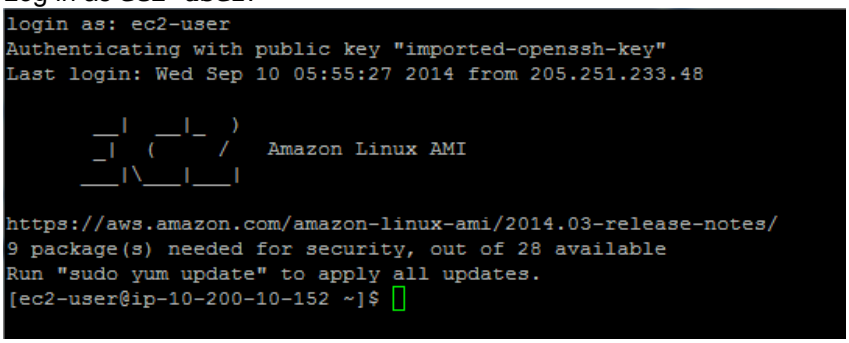
Step	Action
2.9.1	Return to the qwikLABS web page and click the **Download PEM/PPK** drop-down list. • Windows users: click **Download PPK**. • Mac/Linux users: click **Download PEM**.
2.9.2	Save the file to the directory of your choice.

Connecting to Your NAT Instance, Continued

Task 2.10: Connect to Your NAT Instance (Windows)

Note This section is for **Windows** users only. If you are running OSX or Linux, skip to **Task 2.11**.

To connect to your Amazon EC2 instance using PuTTY:

Step	Action
2.10.1	Download **PuTTY** from http://the.earth.li/~sgtatham/putty/latest/x86/putty.exe Download **Pageant** from: http://the.earth.li/~sgtatham/putty/latest/x86/pageant.exe **Note** If those links do not work, download PuTTY and Pageant from the following link: http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
2.10.2	Launch **Pageant**.
2.10.3	Click **Add Key**.
2.10.4	Select the .ppk file you downloaded in step **2.9.1**, click **OK**, and close the Pageant window.
2.10.5	Launch **PuTTY**.
2.10.6	In the **Host Name** box, enter the Public IP address that you copied in step **2.8.17**.
2.10.7	In the **Connection** list, expand **SSH**.
2.10.8	Click **Auth** (don't expand it).
2.10.9	Select **Allow agent forwarding**.
2.10.10	In the **Private key file for authentication** box, select the .ppk file you downloaded in step **2.9.1**.
2.10.11	Click **Open**
2.10.12	In the PuTTY security message, click **Yes**.
2.10.13	Log in as **`ec2-user`**. 
2.10.14	Continue to **Task 2.12: Testing Your NAT Instance.**

Continued on next page

Connecting to Your NAT Instance, Continued

Task 2.11: Connect to Your NAT Instance (Mac and Linux)

Note This section is for **OS X** and **Linux** users only.

To connect to your Amazon EC2 instance:

Step	Action
2.11.1	To add your .pem key to the authentication agent, run the following commands in Terminal: **OS X users:** `ssh-add -K <location of pem>` **Linux users:** `ssh-add -c <location of pem file from step 2.9.1>` For `<location of pem>`, substitute the path/filename to the .pem file you downloaded in step **2.9.1**.
2.11.2	To connect to your NAT instance with SSH agent forwarding enabled, run the following commands in Terminal: `ssh -A ec2-user@<Public IP>` For **<Public IP>**, substitute the public IP address you noted in step **2.8.17**.
2.11.3	Continue to **Task 2.12: Testing Your NAT Instance**.

Testing Your NAT Instance

Introduction

In this part of the lab, you will verify the connectivity of the NAT instance that you launched earlier.

Task 2.12: Test Your NAT Instance

To download the key pair file generated by qwikLABS:

Step	Action

2.12.1	To test that your NAT instance can communicate with the Internet, run this command:
	
	`ping ietf.org`
	
	You should receive a series of continuous responses that look similar to this:
	
	`PING ietf.org (4.31.198.44) 56(84) bytes of data.`
	`64 bytes from mail.ietf.org (4.31.198.44): icmp_seq=1 ttl=48 time=74.9 ms`
	
	If you don't receive the expected response, check with your instructor to determine the problem.
	
	Note this test will only work on websites which have ICMP enabled, so websites besides ietf.org may not work.

2.12.2	Once you've received the expected response, press **Ctrl+C** to end the ping command.
2.12.3	To verify that your NAT instance can connect to your private instance, run this command: `ssh ec2-user@<Private IP>` For ***<Private IP>***, substitute the private IP address you noted in step **2.8.19.**
2.12.4	When prompted for a response, type **yes** and press ENTER. You should be logged in to your private instance now. If you were unsuccessful, check with your instructor.
2.12.5	To verify that your private instance can connect to the Internet, run this command: `ping ietf.org` You should receive a series of continuous responses similar to that which was listed in step **2.12.1**. If you don't receive an appropriate response, check with your instructor to determine the problem.
2.12.6	Once you've received the expected response, press **Ctrl+C** to end the ping command.

Ending your lab

If you choose not to complete the optional portion of the lab, follow the instructions in Appendix B to end your lab.

Optional: Making Your Environment Highly Available

Introduction

This part of the lab is optional, and step-by-step instructions are not provided. Use online documentation to help you if necessary.

http://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/VPC_Introduction.html

Challenge

You successfully created a VPC. Does your VPC's current configuration support high availability (HA)?

Make necessary changes to make your **Lab VPC** support high availability by adding subnets in another Availability Zone.

Hint

Create another public subnet: **Public Subnet 2**; and a private subnet: **Private Subnet 2** in a different Availability Zone within the region.

- CIDR range for Public Subnet 2: **10.200.15.0/24**
- CIDR range for Private Subnet 2: **10.200.55.0/24**
- Be sure to associate your new subnets with the appropriate route tables.

Ending your lab

When you are done, follow the instructions in Appendix B to end your lab.

Achievements

Having completed this lab, you are now able to:

- Create a virtual private cloud (VPC).
- Create subnets within an Availability Zone.
- Create a Network Address Translation (NAT) instance
- Attach Internet Gateway (IGW) to your VPC.
- Create route tables.

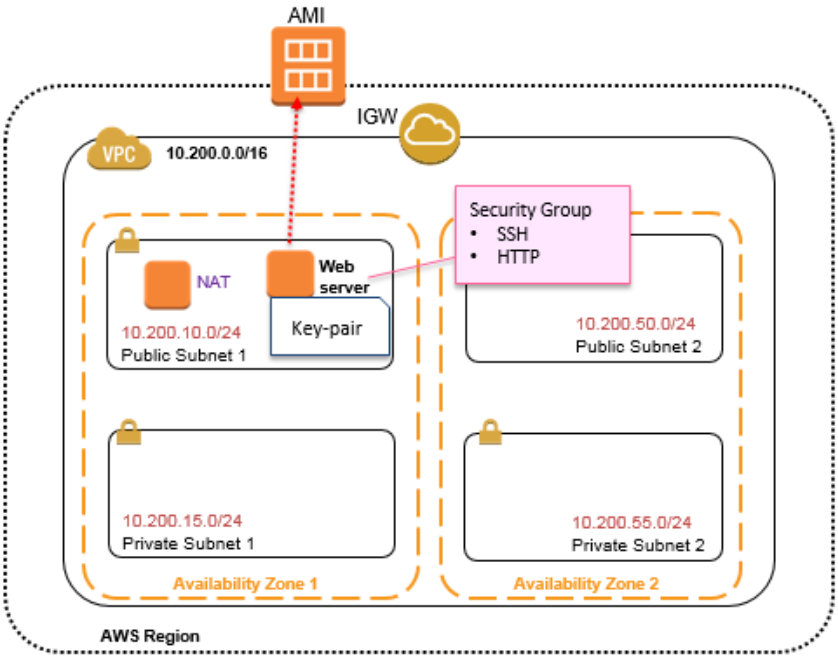
Your results are verified by your ability to complete the lab in the allotted time. When you have completed the lab, your instructor will facilitate a discussion of your results.

Lab 3: Working with Amazon EC2

Lab Overview

Introduction

In this lab, you will launch and configure an Amazon EC2 instance based on an Amazon Linux AMI. After the instance starts successfully, you will connect to a Linux instance and then install an Apache web server and a sample application.



After verifying that the installation has been completed successfully, you will create an Amazon Machine Image (AMI) based on the running instance so that you can easily start an instance with the same settings.

Allotted time

The allotted time for this lab is listed as follows:

Component	Time
Overview	1 minute
Lab	40 minutes
Total:	**41 minutes**

Continued on next page

Lab Overview, Continued

Prerequisites

This lab requires:

- Access to a notebook computer with Wi-Fi on a Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat) system
  - The qwikLABS lab environment is not accessible on an iPad or tablet device, but you may use these devices to access the student guide (PDF)
- For Microsoft Windows users: Administrator access to the computer
- An Internet browser such as Chrome, Firefox, or IE9 (previous versions of Internet Explorer are not supported)
- An SSH client such as PuTTY

Objectives

After completing this lab, you will be able to:

- Create a key pair.
- Launch a new Amazon EC2 instance.
- Connect to a running instance and install the necessary software.
- Create your own Amazon Machine Image (AMI) from the running instance.

Creating a Key Pair

Introduction

In this part of the lab, you will continue from the previous lab and create a key pair. Amazon EC2 uses public-key cryptography to encrypt and decrypt login information. Public-key cryptography uses a public key to encrypt a piece of data, such as a password, and then the recipient uses the private key to decrypt the data. The public and private keys are known as a *key pair*. To log in to your instance, you must create a key pair, specify the name of the key pair when you launch the instance, and provide the private key when you connect to the instance. Linux instances have no password, and you have to use a key pair to log in using SSH.

Task 3.1: Check the existing VPC and subnets

In the previous lab, you created a VPC and four subnets. In this lab, a similar VPC has been created for you so that you can continue to build your environment in the subnets that you created.

To check the properties of those existing subnets:

Step	Action
3.1.1	In the **AWS Management Console**, on the **Services** menu, click **VPC**.
3.1.2	In the navigation pane, click **Your VPCs**.
3.1.3	Select the **Lab VPC** check box, and make a note of the **VPC ID** (starting with *vpc-*) and the **VPC CIDR** range (e.g., *10.200.0.0/16*).
3.1.4	In the navigation pane, click **Subnets**.
3.1.5	To view only the subnets that belong to the **Lab VPC**, enter the **VPC ID** in the search bar.
3.1.6	Make a note of those subnets' **CIDR** ranges.

Continued on next page

Creating a Key Pair, Continued

Task 3.2: Create a key pair

To create a key pair:

Step	Action
3.2.1	On the **Services** menu, click **EC2**.
3.2.2	In the navigation pane, click **Key Pairs**. Notice that there is a key pair already created for you by the qwikLABS environment (the name should be similar to *qwikLABS-L465-xxxx*).
3.2.3	Click **Create Key Pair**.
3.2.4	In the **Key pair name** box, type **LabKeyPair**
3.2.5	Click **Create**.
3.2.6	At the next prompt, save the **LabKeyPair.pem** file to your local computer.

Launching a New Amazon EC2 Instance

Introduction

In this part of the lab, you will go through the steps to launch an Amazon EC2 instance based on an Amazon Linux AMI.

Amazon Elastic Compute Cloud (EC2) is a web service that provides resizable compute capacity in the cloud. It provides you with complete control over your computing resources and lets you run on Amazon's computing environment. Amazon EC2 provides developers with the tools to build failure-resilient applications and to isolate themselves from common failure scenarios.

Task 3.3: Create an instance

To create an Amazon EC2 instance:

Step	Action
3.3.1	In the navigation pane, click **Instances**.
3.3.2	Click **Launch Instance**.
3.3.3	On the **Choose an Amazon Machine Image (AMI)** page, in the row for **Amazon Linux AMI… (HVM)**, click **Select**.
3.3.4	On the **Choose an Instance Type** page, select **t2.medium**.
3.3.5	Click **Next: Configure Instance Details**.
3.3.6	On the **Configure Instance Details** page, specify the following settings: • In the **Network** drop-down list, select **Lab VPC**. • In the **Subnet** drop-down list, select **Public Subnet 1** (10.200.10.0/24). In the **Auto-assign Public IP** drop-down list, select **Enable**.
3.3.7	Click **Next: Add Storage**.
3.3.8	Accept the default and click **Next: Tag Instance**.
3.3.9	In the **Value** box, type **Lab Instance**
3.3.10	Click **Next: Configure Security Group**.
3.3.11	For **Assign a security group,** the **Create a new security group** option should be selected.
3.3.12	Click **Add Rule**. • In the **Type** drop-down list, select **HTTP**. In the **Source** drop-down list, select **Anywhere**.
3.3.13	Click **Review and Launch**.
3.3.14	Review the settings, and then click **Launch**.
3.3.15	When prompted, be sure that **LabKeyPair** is selected in the **Select a key pair** drop-down list.
3.3.16	Select the acknowledgement check box, and then click **Launch Instances**.
3.3.17	Click **View Instances**. There are two instances. One is the instance you just created, and the other one is the NAT instance you created in the previous lab.
3.3.18	Select the **Lab Instance** check box, and then make a note of the **Public IP** address on the **Description** tab of the instance.

Connecting to Your Instance

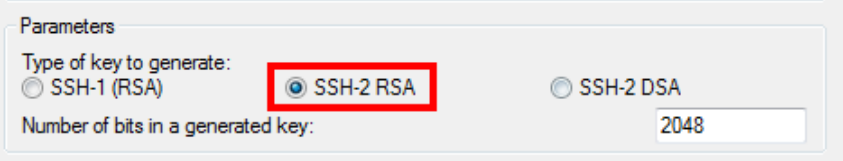
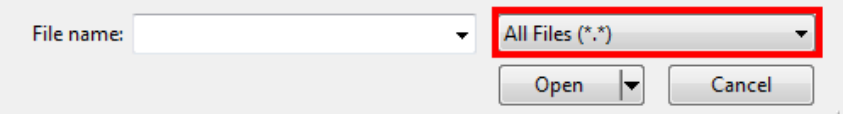
Introduction

In this part of the lab, you will connect to the Amazon EC2 instance that you just launched using the key pair that you created earlier.

Task 3.4: Connect to the EC2 instance (Windows)

Note This section is for **Windows** users only. If you are running OSX or Linux, skip to task 3.5.

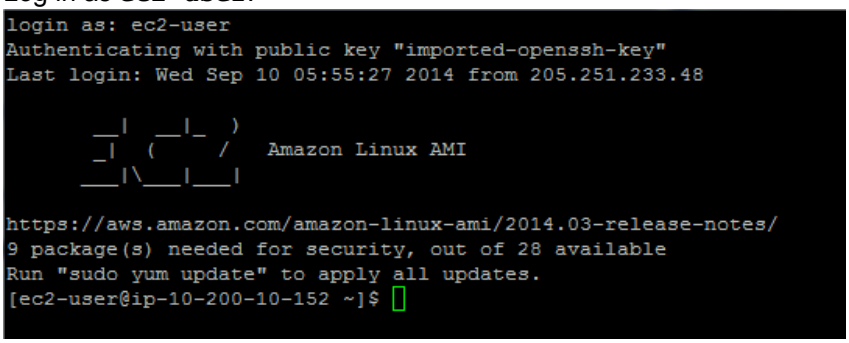
To connect to your Amazon EC2 instance using PuTTY:

Step	Action
3.4.1	Download **PuTTY** from http://the.earth.li/~sgtatham/putty/latest/x86/putty.exe Download **PuTTYgen** from http://the.earth.li/~sgtatham/putty/latest/x86/puttygen.exe **Note** If those links do not work, download PuTTY and PuTTYgen from the following link: http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
3.4.2	Run **PuTTYgen** by double-clicking the **puttygen.exe** that you downloaded.
3.4.3	In the **Parameters** section, for **Type of key to generate**, confirm that the option **SSH-2 RSA** is selected, and then click **Load**. 
3.4.4	Click **All Files (*.*)** and navigate to where you saved the **LabKeyPair.pem** file. 

Continued on next page

Connecting to Your Instance, Continued

Task 3.4: Connect to the EC2 instance (Windows), continued

Step	Action
3.4.5	Click **Open**. When prompted, click **OK** to close.
3.4.6	Click **Save private key**. In the warning message, click **Yes** to close.
3.4.7	In the **File name** box, type **LabKeyPair** Click **Save**.
3.4.8	Close **PuTTY Key Generator.**
3.4.9	Launch **PuTTY**.
3.4.10	In the **Host Name** box, enter the **Public IP** address that you copied in step **3.3.18**.
3.4.11	In the **Connection** list, expand **SSH**.
3.4.12	Click **Auth**.
3.4.13	In the **Private key file for authentication** box, browse to the **LabKeyPair.ppk** file that was generated by PuTTYgen, double-click it, and then click **Open**.
3.4.14	In the PuTTY security message, click **Yes**.
3.4.15	Log in as **`ec2-user`**. 
3.4.16	Continue to **Task 3.6: Installing Software on Your Instance**.

Connecting to Your Instance, Continued

Task 3.5: Connect to the EC2 instance (Mac and Linux)

Note This section is for **Mac** and **Linux** users only.

To connect to your Amazon EC2 instance:

Step	Action
3.5.1	Run the following commands in Terminal: **chmod 600** *<location-of-pem>* **ssh -i** *<location-of-pem>* **ec2-user@***<host-IP>* For *<location-of-pem>*, substitute the path/filename to the LabKeyPair.pem file. For *<host-IP>*, substitute the **Public IP** address you noted in step **3.3.18**.
3.5.2	Continue to **Task 3.6: Installing Software on Your Instance**.

Installing Software on Your Instance

Introduction

In this part of the lab, you will install:

- Available updates.
- An Apache web server.
- A sample PHP application.

Task 3.6: Install a web server on your instance

To install a web server on your Amazon EC2 instance:

Step	Action
3.6.1	To update your instance, execute the following command: `sudo yum –y update` This will run through a check of what updates are available for your instance, download the updates, and install them. **Note** For your convenience, a Command reference text file that contains a text version of this script to simply copy and paste is available on the qwikLABS web page.
3.6.2	To install a package that creates a web server, execute the following command: `sudo yum –y install httpd php` This command installs an Apache web server and the PHP interpreter.
3.6.3	Execute the following command: `sudo chkconfig httpd on` This configures the Apache web server to automatically start when the instance starts.

3.6.4	Execute the following command (we highly recommend that you copy this text from the Command Reference text file):
	
	`wget https://`
	`d2lrzjb0vjvpn5.cloudfront.net/architecting/v4.5`
	`/lab-3-working-with-ec2/static/phpapp.zip`
	
	This downloads a sample PHP application into the current directory.

3.6.5	Execute the following command: `sudo unzip phpapp.zip –d /var/www/html/` This extracts the PHP application into the default Apache web server directory.

Continued on next page

Installing Software on Your Instance, Continued

Task 3.6: Install a web server on your instance, continued

Step	Action
3.6.6	Execute the following command: `sudo service httpd start` This starts the Apache web server.
3.6.7	Open a new web browser window or tab, and enter the **Public IP** address for your instance in the address bar. The sample PHP application is run and the information specific to your Amazon EC2 instance is displayed.
3.6.8	Close the web browser window or tab that you opened in the previous step.
3.6.9	To end your SSH session, type: `exit` and then press ENTER.

Creating Your Amazon Machine Image (AMI)

Introduction

In this part of the lab, you will create an AMI from a running instance. An AMI provides the information required to launch an instance, which is a virtual server in the cloud.

An AMI includes the following:

- A template for the root volume for the instance.
- Launch permissions that control which AWS accounts can use the AMI to launch instances.
- A block device mapping that specifies the volumes to attach to the instance when it's launched.

Task 3.7: Create an AMI

To create an AMI based on your instance:

Step	Action
3.7.1	Return to the AWS Management Console and ensure that you are in the EC2 console.
3.7.2	In the navigation pane, click **Instances**.
3.7.3	Select the **Lab Instance** check box.
3.7.4	In the **Actions** drop-down list, point over **Image**, and then click **Create Image**.
3.7.5	In the **Create Image** dialog box, modify the following fields: • In the **Image name** box, type **My Application** • In the **Image description** box, type **Sample PHP Application** Leave the rest as the default.
3.7.6	Click **Create Image**.
3.7.7	Click the **View pending image** link provided in the confirmation screen. For a few moments, you will see the image in a *pending* state. Periodically refresh your browser to update the status, because this may take a few minutes. When the image is created, the status will change to *available*.

Creating Your Amazon Machine Image (AMI), Continued

Task 3.8: Create an instance using your AMI

To create an AMI based on your instance:

Step	Action
3.8.1	Select the check box for **My Application** (the newly created AMI), and then click **Launch**. Notice that this takes you through a menu similar to launching an EC2 instance manually. The first step that you saw previously, where you selected an AMI, is not necessary now because you have already indicated that you want to launch the AMI you just created.
3.8.2	On the **Choose an Instance Type** page, for **Instance Type**, select **t2.medium**. If **t2.medium** is not available to select, choose any other similar instance type (e.g., **t2.small** or **m3.medium**).
3.8.3	Click **Next: Configure Instance Details**.
3.8.4	On the **Configure Instance Details** page, make the following selections: • In the **Network** drop-down list, select **Lab VPC**. • In the **Subnet** drop-down list, select **Public Subnet 2** (10.200.15.0/24). • In the **Auto-assign Public IP** drop-down list, select **Enable**. Leave the rest unchanged.
3.8.5	Click **Next: Add Storage**.
3.8.6	Accept the default, and click **Next: Tag Instance**.
3.8.7	In the **Value** box, type **AMI Instance**
3.8.8	Click **Next: Configure Security Group**.
3.8.9	Leave the **Create a new security group** option selected, and then click **Add Rule**: • In the **Type** drop-down list, select **HTTP**. • In the **Source** drop-down list, select **Anywhere**.
3.8.10	Click **Review and Launch**.
3.8.11	Review the settings, and then click **Launch**.
3.8.12	When prompted, be sure that **LabKeyPair** is selected in the **Select a key pair** drop-down list.
3.8.13	Select the acknowledgement check box, and then click **Launch Instances**.

Continued on next page

Creating Your Amazon Machine Image (AMI), Continued, Continued

Task 3.8: Create an instance using your AMI, continued

Step	Action
3.8.15	Click **View Instances**. Wait until the instance state of **AMI Instance** changes to *running*. This may take a few minutes. You can click the Refresh button in the upper-right corner to refresh the status. The **Status Checks** for your new instance should show *2/2 checks passed.*

Task 3.9: Test your AMI Instance

To verify that the instance you created with your AMI image already has the PHP sample application installed:

Step	Action
3.9.1	Select **AMI Instance** from the instance list.
3.9.2	Make a note of the **Public IP** address of your new instance.
3.9.3	Open a new web browser window or tab, and enter the Public IP address. The sample PHP application runs and shows you the information specific to your Amazon EC2 instance.

Ending your lab

If you choose not to complete the optional portion of this lab, follow the instructions in Appendix B to end your lab.

Optional

Introduction

This part of the lab is optional, and step-by-step instructions are not provided. Use online documentation to help you if necessary.

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

Challenge 1

In this lab, you launched an Amazon EC2 instance, connected to it, and then installed Apache web server and a sample PHP application.

Create a bash or a shell script to **bootstrap** an Amazon EC2 instance to automatically install Apache web server, PHP interpreter, and the sample PHP application.

This would allow you to scale using dynamic configuration.

Hint Refer to the online documentation, Running Commands on Your Linux Instance at Launch, at http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/user-data.html.

Challenge 2

In this lab, you created an Amazon EBS-backed AMI. The other kind of AMI that you discussed in the class was an instance store-backed AMI.

Create an instance store-backed Linux AMI.

Hint Refer to the online documentation at http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-instance-store.html.

Ending your lab

When you are done, follow the instructions in Appendix B to end your lab.

Achievements

Having completed this lab, you are able to:

- Create a new Key Pair.
- Launch a new Amazon EC2 instance.
- Connect to a running instance and install necessary software.
- Create your own Amazon Machine Image (AMI) from the running instance.

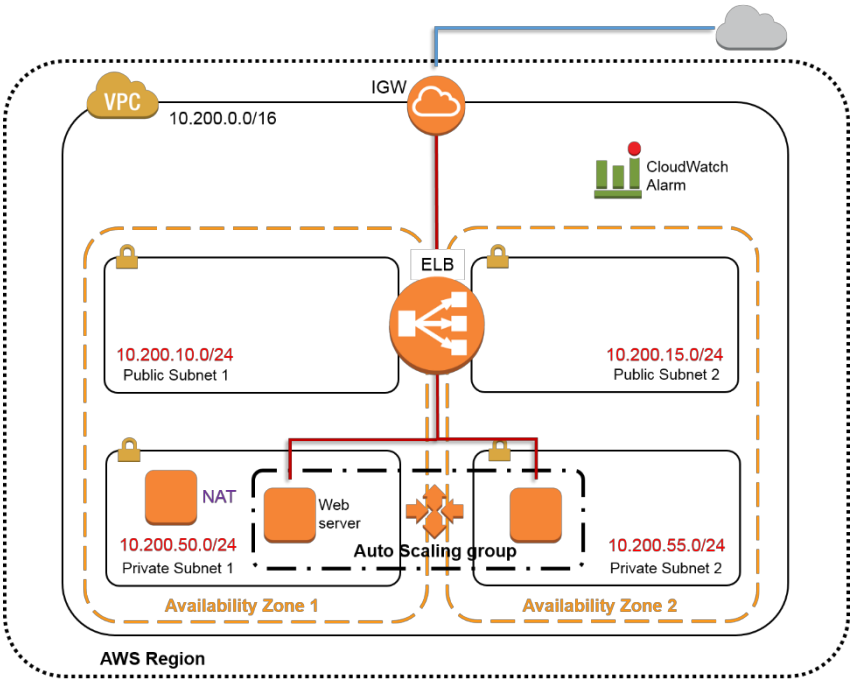
Your results are verified by your ability to complete the lab in the allotted time. When you have completed the lab, your instructor will facilitate a discussion of your results.

Lab 4: Working with Auto Scaling

Lab Overview

Introduction

In this lab, you will implement elasticity using Auto Scaling. Auto Scaling is not just adding and subtracting servers, it is also a mechanism to handle failures similar to the way that Load Balancing handles unresponsive servers. This lab demonstrates how to configure Auto Scaling to automatically launch and monitor Amazon EC2 instances and how to update an associated load balancer.



Auto Scaling allows you to automatically scale your Amazon EC2 capacity according to the conditions you define. With Auto Scaling, you can ensure that the number of Amazon EC2 instances you are using increases seamlessly during demand spikes to maintain performance, and decreases automatically during demand lulls to minimize costs. Auto Scaling is particularly well suited for applications that experience hourly, daily, or weekly variability in usage.

Continued on next page

Lab Overview, Continued

Allotted time

The allotted time for this lab is listed as follows:

Component	Time
Overview	1 minute
Lab	30 minutes
Total:	**31 minutes**

Prerequisites

This lab requires:

- Access to a notebook computer with Wi-Fi on a Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat) system
- The qwikLABS lab environment is not accessible on an iPad or tablet device, but you may use these devices to access the student guide (PDF)
- For Microsoft Windows users: Administrator access to the computer
- An Internet browser such as Chrome, Firefox, or IE9 (previous versions of Internet Explorer are not supported)
- An SSH client such as PuTTY

Objectives

After completing this lab, you will be able to:

- Create a launch configuration.
- Create a load balancer.
- Create an Auto Scaling group.
- Define Auto Scaling policies.

Creating a Security Group

Introduction

In this part of the lab, you will create a security group that will be used by your Auto Scaling group.

A security group acts as a virtual firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. To each security group you add rules that allow traffic to or from its associated instances. You can modify these rules for a security group at any time; the new rules are automatically applied to all instances that are associated with the security group. When you decide whether to allow traffic to reach an instance, all the rules from all the security groups that are associated with that instance are evaluated automatically.

Task 4.1: Check the existing VPC and subnets

In a previous lab, you created a VPC and two subnets. This lab builds on top of what you have done so far. To check the properties of those existing subnets:

Step	Action
4.1.1	In the **AWS Management Console**, on the **Services** menu, click **VPC**.
4.1.2	In the navigation pane, click **Your VPCs**.
4.1.3	Select the **Lab VPC** check box and make a note of the **VPC ID** (starting with *vpc-*).
4.1.4	In the navigation pane, click **Subnets**.
4.1.5	To view only the subnets that belong to the **Lab VPC**, enter the **VPC ID** from step **4.1.3** in the search bar.
4.1.6	Make a note of those subnets' **CIDR** ranges.

Continued on next page

Creating a Security Group, Continued

Task 4.2: Create security groups

To create a security group:

Step	Action
4.2.1	On the **Services** menu, click **EC2**.
4.2.2	In the navigation pane, click **Security Groups**.
4.2.3	Click **Create Security Group**, and then specify these settings: • In the **Security group name** box, type **ELB SG** • In the **Description** box, type **My Lab ELB Security Group** • In the **VPC** drop-down list, select **Lab VPC (10.200.0.0/16)**.
4.2.4	Under **Security group rules**, on the **Inbound** tab, click **Add Rule**. • In the **Type** drop-down list, click **HTTP**. • In the **Source** drop-down list**,** click **Anywhere**. **Note** If your browser does not provide a drop-down list for **Source,** type **0.0.0.0/0**
4.2.5	Click **Create**.
4.2.6	Note the **Group ID** that starts with *sg-* for this new security group. Store this **Group ID** to a text file, so that you can retrieve it for use in later steps.
4.2.7	Click **Create Security Group** to create another security group. • In the **Security group name** box, type **App SG** • In the **Description** box, type **My App Security Group** • In the **VPC** drop-down list, click **Lab VPC (10.200.0.0/16)**.
4.2.8	Under **Security group rules**, on the **Inbound** tab, click **Add Rule**. • In the **Type** drop-down list, click **HTTP**. • In the **Source** drop-down list, select **Custom IP**, and enter the **Group ID** of your **ELB SG** from step **4.2.6**.
4.2.9	Click **Create**.

Creating a Load Balancer

Introduction

Before you create an Auto Scaling group, you first need a load balancer. This load balancer will send requests to your Amazon EC2 instances, dynamically distributing them across Amazon EC2 instances as the Auto Scaling group increases and decreases in size.

Task 4.3: Create a load balancer

To create a load balancer:

Step	Action
4.3.1	In the navigation pane, click **Load Balancers**.
4.3.2	Click **Create Load Balancer**.
4.3.3	Specify these settings: • In the **Load Balancer name** box, type **My-LB** • In the **Create LB Inside** drop-down list, select **Lab VPC**. Leave the remaining settings with their default values.
4.3.4	In **Select Subnets**, under **Available Subnets**, select **Public Subnet 1** and **Public Subnet 2** by clicking the plus signs on the left side of their rows.
4.3.5	Click **Next: Assign Security Groups**.
4.3.6	On the **Assign Security Groups** page, select **ELB SG** from the existing security group list and clear the **default** check box. The only check box that should be selected is the one for **ELB SG**.
4.3.7	Click **Next: Configure Security Settings**.
4.3.8	Click **Next: Configure Health Check**.
4.3.9	On the **Configure Health Check** page, specify these settings: • In the **Ping Path** box, delete the default value and type **/index.php** • Change the **Response Timeout** value to **10 seconds**. • Change the **Health Check Interval** value to **15 seconds**. • In the **Unhealthy Threshold** drop-down list, select **5**. • In the **Healthy Threshold** drop-down list, select **2**. Leave the remaining settings with their default values.
4.3.10	Click **Next: Add EC2 Instances**.
4.3.11	Click **Next: Add Tags**. **Note** Do not select any of the instances.
4.3.12	Click **Review and Create**. **Note** Do not specify any tag.
4.3.13	Click **Create**.
4.3.14	Click **Close**.

Creating a Launch Configuration

Introduction

Your first step in creating an Auto Scaling group is to generate a launch configuration. A launch configuration specifies details such as the AMI to be used when launching new instances, the instance type, and the configuration scripts.

Task 4.4: Create a launch configuration

To create a launch configuration:

Step	Action
4.4.1	In the navigation pane, click **Launch Configurations**.
4.4.2	Click **Create Auto Scaling group**.
4.4.3	Click **Create launch configuration**.
4.4.4	In the row for **Amazon Linux AMI… (HVM)**, click **Select**.
4.4.5	Select the **t2.small** instance type.
4.4.6	Click **Next: Configure details**.
4.4.7	In the **Name** box, type **Lab LC**
4.4.8	For **Monitoring**, select the **Enable CloudWatch detailed monitoring** option.
4.4.9	Expand the **Advanced Details** section and then add the following to the **User data** box: **#!/bin/bash** **yum -y update** **yum -y install httpd-2.2.27 php-5.3.29** **chkconfig httpd on** **wget https://d2lrzjb0vjvpn5.cloudfront.net/architecting/v4.5/lab-4-working-with-autoscaling/static/phpapp.zip** **unzip phpapp.zip -d /var/www/html/** **service httpd start** **yum -y install stress** **sleep 90** **stress --cpu 8 --io 8 --vm 6 --hdd 8 -t 400** **Note** For your convenience, a Command reference text file that contains a text version of this script is available on the qwikLABS web page. Copy and paste the script from that text file.
4.4.10	Leave the remaining settings in their default values and click **Next: Add Storage**.
4.4.11	Click **Next: Configure Security Group**.
4.4.12	For **Assign a security group**, click the **Select an existing security group** option and then select **App SG**.

Continued on next page

Creating a Launch Configuration, Continued

Task 4.4: Create a launch configuration, continued

Step	Action
4.4.13	Click **Review**. In the warning message, click **Continue**. **Note** In this lab, you will not be logging in to the instance via SSH; therefore, that port is not opened in the **App SG** security group.
4.4.14	Review the settings and then click **Create launch configuration**.
4.4.15	When prompted, accept the key pair generated by qwikLABS.
4.4.16	Select the acknowledgement check box, and then click **Create launch configuration**.

Creating an Auto Scaling Group

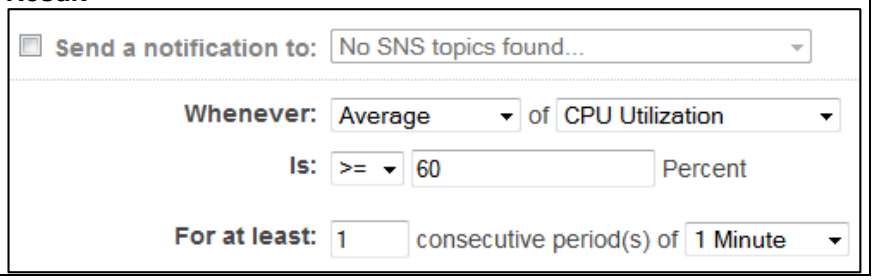
Introduction

The launch configuration controls *which* instances are launched and how they are configured, and the Auto Scaling group controls *when* instances are launched or terminated and what criteria trigger an auto scaling action.

Task 4.5: Create an Auto Scaling group

After creating your **Launch Configuration** in the last procedure, you should be automatically redirected to the **Create an Auto Scaling Group** page.

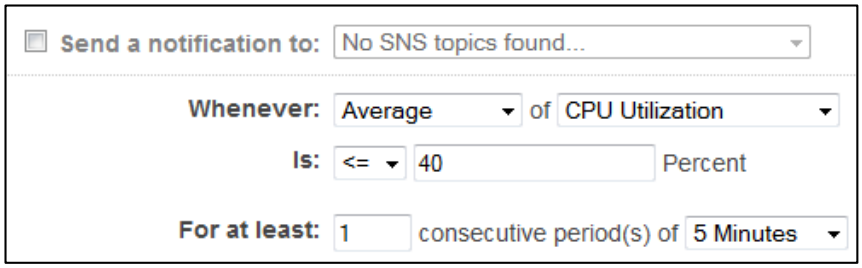
To create an Auto Scaling group:

Step	Action
4.5.1	On the **Create Auto Scaling Group** page, specify the following settings: • In the **Group name** box, type **Lab-ASG** • In the **Group size** box, type **2** • In the **Network** drop-down list, select **10.200.0.0/16 l Lab VPC**. • Click in the **Subnet** box, and then select **Private Subnet 1** and **Private Subnet 2**.
4.5.2	Expand the **Advanced Details** section: • For **Load Balancing**, select the **Receive traffic from Elastic Load Balancer(s)** check box, and then select **My-LB** in the box (the load balancer that you created earlier).
4.5.3	Click **Next: Configure scaling policies**.
4.5.4	Select the **Use scaling policies to adjust the capacity of this group** option, and change the policy to “Scale between **2** and **6** instances.”
4.5.5	Under **Increase Group Size**, click the **Add new alarm** link for the **Execute policy when** section.
4.5.6	In the **Create Alarm** dialog box: • Clear the **Send a notification to** check box. • In the **Percent** box, type **60** • Change **consecutive period(s) of** 5 Minutes to **1 Minute**. **Result** 

Continued on next page

Creating an Auto Scaling Group, Continued

Task 4.5: Create an Auto Scaling group, continued

Step	Action
4.5.7	Click **Create Alarm**.
4.5.8	In the **Increase Group Size** section, specify the following settings: • Change the **Take the action** value to "Add **2** instances." • Change the **And then wait** value to "**90** seconds before allowing another scaling activity."
4.5.9	Under **Decrease Group Size**, click the **Add new alarm** link for the **Execute policy when** section.
4.5.10	In the **Create Alarm** dialog box: • Clear the **Send a notification to** check box. • Change "**>=**" to "**<=**". • In the **Percent** box, type **40** **Result** 
4.5.11	Click **Create Alarm**. **Note** Ensure that the two alarms do not have identical names. In rare circumstances, the names created by the tool will be identical and cause conflicts. If they are identical, click "Edit" next to one of the policies and give it a unique name instead.
4.5.12	Still in the **Decrease Group Size** section, modify the following: • Change the **Take the action** value to "Remove **2** instances." • Change the **And then wait** value to "**90** seconds before allowing another scaling activity."
4.5.13	Click **Review**. **Note** You will not configure notifications in this lab.
4.5.14	Click **Create Auto Scaling group**.
4.5.15	Click **Close**.

Continued on next page

Creating an Auto Scaling Group, Continued

Task 4.5: Create an Auto Scaling group, continued

Step	Action
4.5.16	With the newly created **Lab-ASG** auto scaling group selected, click the **Scaling History** tab. The Auto Scaling group should launch two instances because the group size was set to 2. From the EC2 Dashboard, click **Instances**. Wait until the instances are fully started and the **Status Checks** shows *2/2 checks passed*. In Step 4.4.9, you defined the launch configuration with user data that emulates resource consumption: … yum -y install stress sleep 90 stress --cpu 4 --io 4 --vm 2 --hdd 4 -t 400 sleep 300 stress --cpu 6 --io 6 --vm 4 -hdd 6 -t 400 sleep 300 stress --cpu 8 --io 8 --vm 6 --hdd 8 -t 400 Refresh your **Scaling History** tab regularly, and within about 5 minutes your Auto Scaling Group should add two more instances in response to the high stress being placed on the instances' CPUs. Within another 3 minutes, two more instances should be started. Finally, 3 minutes after that, your Auto Scaling Group should detect that the stress functions have completely timed out and it will subsequently terminate your two oldest instances.

Ending your lab

When you are done, follow the instructions in Appendix B to end your lab.

Achievements

Having completed this lab, you are able to:

- Create a launch configuration.
- Create a load balancer.
- Create an Auto Scaling Group.
- Define Auto Scaling policies.

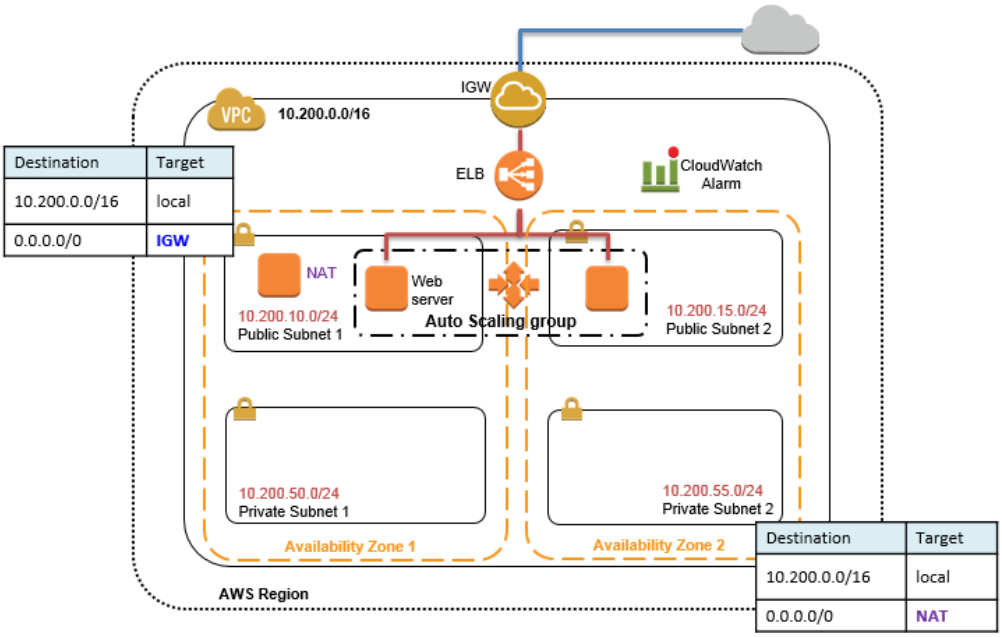
Your results are verified by your ability to complete the lab in the allotted time. When you have completed the lab, your instructor will facilitate a discussion of your results.

Lab 5: Exploring AWS CloudFormation

Lab Overview

Introduction

In this lab, you will use AWS CloudFormation to automate the creation of all AWS resources that you manually created in the previous labs, such as a VPC, subnets, a NAT server, and an Auto Scaling group. Using an AWS CloudFormation template, you can manage and configure your AWS cloud more quickly.



Allotted time

The allotted time for this lab is listed as follows:

Component	Time
Overview	1 minute
Lab	30 minutes
Total:	**31 minutes**

Continued on next page

Lab Overview, Continued

Prerequisites

This lab requires:

- Access to a notebook computer with Wi-Fi on a Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat) system.

  (The qwikLABS lab environment is not accessible on an iPad or tablet device, but you may use these devices to access the student guide PDF.)

- For Microsoft Windows users: Administrator access to the computer
- An Internet browser such as Chrome, Firefox, or IE9 (previous versions of Internet Explorer are not supported)

Objectives

After completing this lab, you will be able to:

- Navigate through the AWS CloudFormation Dashboard and inspect the AWS resources that were created.
- Select a region and create an AWS CloudFormation stack.
- Examine the execution of the AWS CloudFormation template.

Exploring the AWS CloudFormation Dashboard

Introduction

AWS CloudFormation gives developers and systems administrators an easy way to create and manage a collection of related AWS resources, provisioning and updating them in an orderly and predictable fashion.

Task 5.1: Inspect the existing stack

Your lab environment contains a pre-defined Amazon VPC. It contains four subnets, and is nearly identical to the VPC that you created in the previous lab. To check the properties of those subnets:

Step	Action
5.1.1	In the **AWS Management Console**, on the **Services** menu, click **CloudFormation**. You should see a stack with *CREATE_IN_PROGRESS* status. Wait until the status changes to *CREATE_COMPLETE*.
5.1.2	Select the check box for the existing stack (the stack name starts with *qlstack2-labinstance-*). Information about this stack is displayed on the **Overview** tab in the lower pane. Notice that the **Description** field provides a brief explanation about this stack.
5.1.3	Click the **Resources** tab. Review each of the resources created for you by the template. Notice that the resource Type and its ID, such as **AWS::EC2::VPC,** has a **Physical ID** that starts with *vpc-*, which you observed in previous labs.
5.1.4	Click the **Events** tab. The Events tab provides point-in-time event information, starting at template initialization and ending with the completed template. This information can help you debug your AWS CloudFormation template.
5.1.5	Click the **Parameters** tab. The Parameters tab displays parameter values that were set by the AWS CloudFormation template.

Continued on next page

Exploring the AWS CloudFormation Dashboard, Continued

Task 5.2: Review the existing launch configuration

To review the existing launch configuration:

Step	Action
5.2.1	On the **Services** menu, click **EC2**.
5.2.2	In the navigation pane, click **Instances**. Notice that there are three instances: two **WebApplicationServer** instances and one **NAT** instance.
5.2.3	In the navigation pane, click **Launch Configurations**.
5.2.4	Select the existing launch configuration. The **Details** tab displays the settings for the launch configuration, such as **AMI ID** and **Instance Type**.
5.2.5	Click the **View User data** link to see the bootstrapping script. **Note** In the previous lab, you configured these properties manually from the AWS Management Console.
5.2.6	Click **Close**.
5.2.7	In the navigation pane, click **Auto Scaling Groups**.
5.2.8	If it is not already selected, select the check box for the existing Auto Scaling group. In the lower pane, the **Details** tab displays the configuration for this Auto Scaling group.
5.2.9	Click the **Scaling History** tab to see that two instances were launched by this Auto Scaling group because the policy is set to run at least two instances (min).

Creating a Stack

Introduction

In this part of the lab, you will deploy a template and its associated collection of resources (called a *stack*) by using the AWS CloudFormation Dashboard.

Task 5.3: Create a stack

To create a stack in a different region:

Step	Action
5.3.1	On the **Services** menu, click **CloudFormation**.
5.3.2	In the navigation bar, select a different region from the drop-down list. 
5.3.3	Click **Create New Stack**.
5.3.4	In the **Name** box, type **MyLabTemplate**
5.3.5	For **Source**, select the **Specify an Amazon S3 template URL** option.
5.3.6	Type the following Amazon S3 URL: `https://us-west-2-aws-training.s3.amazonaws.com/awsu-ilt/architecting/v4.5/lab-5-exploring-cloudformation/static/lab-5-exploring-cloudformation-student.template` **Note** If you prefer, you can copy and paste the URL from the **Cloudformation Template for Lab Instructions** tab of the qwikLABS web page.
5.3.7	Click **Next**.
5.3.8	Click **Next**. You don't need to change any of the parameter values.
5.3.9	In the **Tags** section, enter the following values: • In the **Key** box, type **Environment** • In the **Value** box, type **Lab**
5.3.10	Click **Next**.
5.3.11	On the **Review** page, click the **Template URL** link to open the provided AWS CloudFormation template in another tab. You can examine the template to see what it creates. Notice that **Mappings** specifies the AMI ID to be used for each region. Locate **WebApplicationLC** to see that user data is also defined within the block. You launched an Amazon EC2 instance and installed available updates and an Apache server manually in Lab 3. The same steps can be automated using an AWS CloudFormation template. Take your time to review the Cloud Formation template before proceeding.

Continued on next page

Creating a Stack, Continued

Task 5.3: Create a stack, continued

Step	Action
5.3.12	Click **Create**. The status should be *CREATE_IN_PROGRESS*. Click the **Events** tab in the lower pane to track the progress. As the resources are created, you can see them in their respective sections of your AWS Management Console, such as in the EC2 Dashboard and VPC Dashboard. When all the resources are created, the status changes to *CREATE_COMPLETE*.

Ending your lab

When you are done, follow the instructions in Appendix B to end your lab.

Conclusion

Achievements

Having completed this lab, you are able to:

- Navigate through the AWS CloudFormation Dashboard and inspect the AWS resources that were created.
- Select a region and create an AWS CloudFormation stack.
- Examine the execution of the AWS CloudFormation template.

Your results are verified by your ability to complete the lab in the allotted time. When you have completed the lab, your instructor will facilitate a discussion of your results.

Lab 6: Creating a Batch Processing Cluster

Lab Overview

Introduction

In this lab, you will use the AWS Management Console to build a basic batch processing cluster.

You will:

- Launch and configure an Amazon EC2 instance that will serve as the template for future worker nodes in your batch processing cluster.
- Create an Amazon Machine Image (AMI) from that instance.
- Use Amazon SQS to create task queues for passing messages to your instances.
- Launch an Auto Scaling group of instances based on your AMI.
- Schedule work via your task queue.
- Observe the output queue.

The worker nodes in your cluster have a simple job: to convert some number of individual images into a single montage image. A worker node will download images from a list that you provide and will then stitch them into a composite montage using the ImageMagick tool. While this is not the most CPU-intensive job, it does require some cycles; the larger the size and number of images you provide for each job, the more work each node will have.

For this lab, you will provide a newline-delimited list of image URLs. An Amazon EC2 worker node will download each image and produce output such as:

Continued on next page

Lab Overview, Continued

Allotted time

The allotted time for this lab is listed as follows:

Component	Time
Overview	1 minute
Lab	40 minutes
Total:	**41 minutes**

Prerequisites

This lab requires:

- Access to a notebook computer with Wi-Fi on a Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat) system
  - The qwikLABS lab environment is not accessible on an iPad or tablet device, but you may use these devices to access the student guide (PDF)
- For Microsoft Windows users: Administrator access to the computer
- An Internet browser such as Chrome, Firefox, or IE9 (previous versions of Internet Explorer are not supported)
- An SSH client such as PuTTY

Objectives

After completing this lab, you will be able to:

- Bootstrap an EC2 instance using user data.
- Create an AMI from a running instance.
- Use the AWS Management Console to create an Amazon SQS queue.
- Create an Auto Scaling group with scaling policies based on an Amazon SQS queue.
- Use the AWS Management Console to pass messages to, and read messages from, an Amazon SQS queue.

Creating an IAM Role

Introduction

Your batch processing nodes will communicate with Amazon SQS to receive processing instructions and will then store results in Amazon S3. Start by creating an IAM role that grants access to both Amazon SQS and Amazon S3. This role will be assigned to your Amazon EC2 instance.

Task 6.1: Create a role

To create an IAM role:

Step	Action
6.1.1	In the **AWS Management Console**, on the **Services** menu, click **IAM**.
6.1.2	In the navigation pane, click **Roles**.
6.1.3	Click **Create New Role**.
6.1.4	In the **Role Name** box, type **BatchProcessing**
6.1.5	Click **Next Step**.
6.1.6	On the **Select Role Type** page, ensure **AWS Service Roles** is selected.
6.1.7	In the **AWS Service Roles** menu, in the row for **Amazon EC2,** click **Select.**
6.1.8	On the **Attach Policy** page, select the **AmazonSQSFullAccess** policy from the list.
6.1.9	Click **Next Step**.
6.1.10	Click **Create Role**.
6.1.11	Click the name of the newly created **BatchProcessing** role to display the details view.
6.1.12	To add an additional role policy, click **Attach Policy**.
6.1.13	On the **Attach Policy** page, select the **AmazonS3FullAccess** policy from the list.
6.1.14	Click **Attach Policy**.

Creating Two Amazon SQS Task Queues

Introduction

In this section, you will use the AWS Management Console to create two Amazon Simple Queue Service (SQS) queues to hold input and output tasks. You will then send a message to your input queue. You will eventually dispatch work via the input queue and view the results provided by your worker nodes in the output queue.

Task 6.2: Create queues

To create queues named **input** and **output** in Amazon SQS:

Step	Action
6.2.1	On the **Services** menu, click **SQS**.
6.2.2	Click **Create New Queue**.
6.2.3	In the **Create New Queue** dialog box, specify the following settings: • In the **Queue Name** box, type **input** • Change the **Default Visibility Timeout** to **90 seconds**. Leave the remaining settings in their default values.
6.2.4	Click **Create Queue**.
6.2.5	Repeat these steps to create another queue named **output** with a **Default Visibility Timeout** value of **90 seconds**.
6.2.6	Select your **input** queue. Ensure that the **output** queue is not selected.
6.2.7	In the **Queue Actions** drop-down list, click **Send a Message**.
6.2.8	Copy the list of image URLs from the lab's provided Command Reference File and paste it in the **Enter the text of a message you want to send** dialog box.
6.2.9	Click **Send Message**.
6.2.10	Click **Close**.

Creating an Amazon S3 bucket

Introduction

In this section, you will create an Amazon S3 bucket to hold the output from your worker nodes.

Task 6.3: Create an Amazon S3 bucket

To create an Amazon S3 bucket to hold your image files:

Step	Action
6.3.1	On the **Services** menu, click **S3**.
6.3.2	Click **Create Bucket**.
6.3.3	In the **Bucket Name** box, type **image-bucket-**<*number*> **Note** Your bucket name must be unique; so, change the number to generate a unique bucket name, such as *image-bucket-1029.*
6.3.4	To accept the default region and create your S3 bucket, click **Create**. You are now ready to launch worker nodes within an Auto Scaling group.

Creating a Master Amazon EC2 Instance

Introduction

You will now launch an Amazon EC2 instance with a configuration script that loads ImageMagick and the batch processing software. This master instance will then be used to create an Amazon Machine Image (AMI).

Task 6.4: Launch an instance

To create a master instance:

Step	Action
6.4.1	On the **Services** menu, click **EC2**.
6.4.2	In the navigation pane, click **Instances**.
6.4.3	Click **Launch Instance**.
6.4.4	On the **Choose an Amazon Machine Image (AMI)** page, in the row for **Amazon Linux AMI… (HVM),** click **Select**.
6.4.5	On the **Choose an Instance Type** page, leave the default instance type of **t2.micro** and click **Next: Configure Instance Details**.
6.4.6	On the **Step 3: Configure Instance Details** page, use the following values: • **Network: Lab VPC** • **Subnet:** Select either **Public Subnet 1** or **Public Subnet 2** • **Auto-assign Public IP: Enable** • **IAM role**: **BatchProcessing**
6.4.7	Click **Advanced Details.**
6.4.8	Next, you will supply a user data field to the Amazon EC2 instance so that it is configured to load ImageMagick and the batch processing software. In the **User data** section: • Verify that the **As text** option is selected. • Using the command reference file you can find on the qwikLABS page you used to access this lab, copy the script for this procedure into the **User data** box.
6.4.9	Click **Next: Add Storage**.
6.4.10	Accept the default and click **Next: Tag Instance**.
6.4.11	In the **Value** box, type **Master**
6.4.12	Click **Next: Configure Security Group**.
6.4.13	For **Assign a security group**, the **Create a new security group** option should be selected. Specify the following settings: • In the **Security group name** box, type **BatchProcessing** • In the **Description** box, type **Batch Processing Security Group**

Continued on next page

Creating a Master Amazon EC2 Instance, Continued

Task 6.4: Launch an instance, continued

Step	Action
6.4.14	To accept the default traffic rule settings, click **Review and Launch**. **Note** For the purposes of this lab, you are creating a security group that is open to the world via SSH. This is because a more specific security group may not work on the network where you are performing this lab. In practice, you would specify a security group locked down to a specific public IP range allocated to your organization.
6.4.15	Review the settings, and then click **Launch**.
6.4.16	To accept the key pair generated by qwikLABS, select the acknowledgement check box.
6.4.17	Click **Launch Instances**.
6.4.18	Click **View Instances**. Wait until the **Instance State** of the **Master** Instance changes to *running* and the **Status Checks** shows *2/2 checks passed*.

Connecting to Your Instance

Introduction

You will use your **Master** instance as the basis for an AMI. Before doing so, you will connect to the instance via SSH to ensure that the necessary files were loaded via the User data script.

Task 6.5: Download a key pair

To download the key pair file generated by qwikLABS:

Step	Action
6.5.1	Return to the qwikLABS web page and click the **Download PEM/PPK** drop-down list. • Windows users: click **Download PPK**. • OS X/Linux users: click **Download PEM**.
6.5.2	Save the file to the directory of your choice.

Task 6.6: Connecting to your instance with PuTTY (Windows)

Note This section is for **Windows** users only. **OSX** or **Linux** users, skip to Task 6.7.

To connect to the newly created Master instance:

Step	Action
6.6.1	Launch **PuTTY**.
6.6.2	Copy the **Public IP** of your **Master** instance from the EC2 Dashboard and paste it in the **Host Name** box.
6.6.3	In the **Connection** list, expand **SSH** and then click **Auth**.
6.6.4	In the **Private key file for authentication** box, browse to the **PPK** file that you downloaded in the previous step.
6.6.5	Click **Open** to initiate the connection, in the PuTTY security message, click **Yes** and log in as `ec2-user`.
6.6.6	Run the `ls` command to view the contents of your user directory. If you see entries named `image_processor.py`, `MessageProducer.py` and `jobs`, your instance is correctly configured.
6.6.7	To end your SSH session, type **exit**
6.6.8	Proceed to **Task 6.8: Create a launch configuration.**

Task 6.7: Connecting to your instance with Terminal (OS X and Linux users)

Note This section is for **OS X** and **Linux** users only.

To connect to your Amazon EC2 instance using PEM:

Step	Action
6.7.1	Copy the **Public DNS** value of your **Master** instance from the EC2 Dashboard.
6.7.2	Run the following commands in Terminal to connect to the Amazon EC2 instance. **chmod 600** *<location-of-pem>* **ssh -i** *<location-of-pem>* **ec2-user@***<host-IP>* Substitute the Public DNS value for *<host-IP>*.
6.7.3	Run the **ls** command to view the contents of your user directory. If you see entries named **image_processor.py**, **MessageProducer.py** and **jobs**, your instance is correctly configured.
6.7.4	To end your SSH session, type **exit**
6.7.5	Proceed to **Task 6.8: Create a launch configuration**.

Launching Worker Nodes

Introduction

In this section, you will create an Auto Scaling group of worker nodes to process your work. After you successfully test the initial node, you will add Scaling Policies to automatically expand the size of the Auto Scaling group.

Task 6.8: Create a launch configuration

To create an Auto Scaling group that responds to a need to scale out by launching new instances that use your AMI:

Step	Action
6.8.1	On the **Services** menu, click **EC2**.
6.8.2	In the navigation pane, click **Launch Configurations**.
6.8.3	Click **Create Auto Scaling group**.
6.8.4	Click **Create launch configuration**.
6.8.5	On the **Choose an Amazon Machine Image (AMI)** page, in the row for **Amazon Linux AMI… (HVM)** , click **Select**.
6.8.6	To accept the default instance type, click **Next: Configure details**.
6.8.7	On the **Create Launch Configuration** page, specify the following settings: • In the **Name** box, type **Workers** • In the **IAM role** drop-down list, select **BatchProcessing**.
6.8.8	Click **Advanced Details** to expand it.
6.8.9	Next, you will supply a user data field to the Amazon EC2 Launch Configuration so that the instances it creates are configured to automatically run the image conversion process. Because these instances will be generated from the **AMI** you created earlier, they will already have the ImageMagick and image processing software installed on them when they are launched. In the **User data** box, verify that the **As text** option is selected. Using the command reference file that you used earlier, copy the user data text for this procedure into the **User data** box.
6.8.10	For **IP Address Type**, select the **Assign a public IP address to every instance** option.
6.8.11	Click **Next: Add Storage**.
6.8.12	Click **Next: Configure Security Group** to accept the default storage setting.
6.8.13	For **Assign a security group**, select the **Select an existing security group** option. From the list of security groups, select **Batch Processing Security Group**.
6.8.14	Click **Review**.
6.8.15	Click **Create launch configuration**.
6.8.16	To accept the key pair generated by qwikLABS, select the acknowledgement check box and then click **Create launch configuration**.

Launching Worker Nodes, Continued

Task 6.9: Create an auto scaling group

To create an Auto Scaling group:

Step	Action
6.9.1	On the **Configure Auto Scaling group details** page, specify the following settings: • In the **Group name** box, type **worker-group** • In the **Network** drop-down list, click **Lab VPC**. • Click in the **Subnet** box, and select either **Public Subnet 1** or **Public Subnet 2**.
6.9.2	Click **Next: Configure scaling policies**.
6.9.3	Select the **Use scaling policies to adjust the capacity of this group** option and change the policy to "Scale between **1** and **4** instances."
6.9.4	In the **Increase Group Size** section, change **Take the action** to Add **1** instances. **And then wait** to **180** seconds before allowing another scaling activity. **Note** You've specified the scaling policies but the alarm you'll need to create can't be created here. You'll create the alarms in a later task, using the **SQS Management Console**.
6.9.5	Click **Review**.
6.9.6	Click **Create Auto Scaling group** and then click **Close**. Your Auto Scaling group has been configured to run only a single instance now, and no alarms have been attached to its scaling policies. After you verify that the worker node is functioning correctly, you can create alarms to automatically adjust the number of worker nodes.
6.9.7	In the navigation pane, click **Instances**. A new instance will start now. Wait until the **Instance State** changes to *running* and the **Status Checks** shows *2/2 checks passed.*

Dispatching Work and Viewing Results

Introduction

In this section, you will use the **SQS Dashboard** to view the output of your last message, and use a message producer script to add more messages to your Amazon SQS input queue in order to use CloudWatch SQS metrics in the next section.

Task 6.10: Check your queue for messages

To verify that the message you sent to the input queue earlier has been processed and moved to the output queue:

Step	Action
6.10.1	On the **Services** menu, click **SQS**. Confirm that there is **1 Message Available** in your **output** queue. If the message is still in your input queue, click **Refresh** periodically. It may take a few minutes after your worker instances were created for the message to move to the **output** queue from the **input** queue. **Note** If there is no message in your **output** queue: • Ensure that your queues are named **input** and **output** (in lowercase). • Ensure that your **BatchProcessing** role has granted full permissions for Amazon SQS and Amazon S3. • Ensure that your worker node is using the **BatchProcessing** IAM role (defined in the Launch Configuration). • Ensure that your worker node is running. Ask your instructor for assistance in successfully running your worker node.
6.10.2	Select the **output** queue. Ensure that the **input** queue is not selected.
6.10.3	In the **Queue Actions** drop-down list, click **View/Delete Messages**.
6.10.4	Click **Start Polling for Messages**.
6.10.5	Find your message and click **More Details** to view the message body. The message contains the output. To view the image, select and copy the URL portion of the message. Then paste the link into a new browser window. You should see a composite image of the five images that were linked individually in the message you sent to the **input** queue in **Step 6.2.9**.
6.10.6	Click **Close** to close the **Message Details** window.
6.10.7	Click **Close** to close the **View/Delete Messages in output** window.

Dispatching Work and Viewing Results, Continued

Task 6.11: Connect to your worker instance and run the message production script (Windows)

Note This section is for **Windows** users only. **OS X** or **Linux** users should skip to Task 6.12.

To run the message production script using Windows:

Step	Action
6.11.1	Launch **PuTTY**.
6.11.2	On the **Services** menu, click **EC2**.
6.11.3	In the navigation pane, click **Instances**.
6.11.4	Select your worker instance. **Note** Your worker instance will be unnamed; however, it will be the only other instance available other than your **Master** instance.
6.11.5	Copy the **Public DNS** name of your worker instance from the EC2 Dashboard and paste it into the **Host Name** box in your **PuTTY** client.
6.11.6	In the **Connection** list, expand **SSH**, and then click **Auth**.
6.11.7	In the **Private key file for authentication** box, browse to the **PPK** file that you downloaded in **Task 6.3**.
6.11.8	Click **Open** to initiate the connection, in the PuTTY security message, click **Yes** and then log in as **`ec2-user`**.
6.11.9	Run the following command: `sudo python MessageProducer.py` **Note** You should receive a message that confirms that the message was sent to your input queue. If you do not receive this confirmation, notify your instructor. If attempts to run the script continue to fail, you can emulate this script's behavior by using steps 6.2.6–6.2.9 to send at least 35 messages from your input queue. The purpose of this script is to generate a long backup of messages in your input queue, so that the Amazon CloudWatch metric associated with visible queue messages will be available in the next task.
6.11.10	Proceed to **Task 6.13: Monitoring the Cluster**.

Dispatching Work and Viewing Results, Continued

Task 6.12: Connect to your worker instance and run the message production script (OS X and Linux)

Note This section is for **OS X** or **Linux** users only.

To run the message production script using a OS X or Linux machine:

Step	Action
6.12.1	On the **Services** menu, click **EC2**.
6.12.2	In the navigation pane, click **Instances**.
6.12.3	Select your worker instance. **Note** Your worker instance will be unnamed; however, it will be the only other instance available other than your **Master** instance.
6.12.4	Copy the **Public DNS** value of your worker instance from the EC2 Dashboard.
6.12.5	Run the following command to connect to the Amazon EC2 instance: `ssh -i <location-of-pem> ec2-user@<host-IP>` Substitute the Public DNS value for *<host-IP>*.
6.12.6	Run the following command: `sudo python MessageProducer.py` **Note** You should receive a message that confirms that the message was sent to your input queue. If you do not receive this confirmation, notify your instructor. If attempts to run the script continue to fail, you can emulate this script's behavior by using steps 6.2.6–6.2.9 to send at least 35 messages from your input queue. The purpose of this script is to generate a long backup of messages in your input queue, so that the Amazon CloudWatch metric associated with visible queue messages will be available in the next task.
6.12.7	Proceed to **Task 6.13: Monitoring the Cluster**.

Monitoring the Cluster

Introduction

You can now use CloudWatch to monitor your cluster. You will define a CloudWatch alarm for use with Auto Scaling policies.

Task 6.13: Create an alarm

To create an alarm that triggers your Auto Scaling group:

Step	Action
6.13.1	On the **Services** menu, click **CloudWatch**.
6.13.2	Click **Browse Metrics**.
6.13.3	To expand the drop-down list, click **Browse Metrics** again.
6.13.4	In the **Browse Metrics** drop-down list, click **SQS**. **Note** If the **SQS Metrics** header is not visible, return to your **input** queue and ensure that there are messages queued for processing. This will trigger metrics to be sent to CloudWatch after a few minutes. The **Visible** count may take a bit longer to appear. If the list of messages waiting in your input queue is lower than 25, run the message producer script again.
6.13.5	Select the check box next to the **ApproximateNumberOfMessagesVisible** metric for the **input** queue.
6.13.6	On the **Tools** menu in the lower-right corner, click **Create Alarm**.
6.13.7	On the **Alarm Threshold** page, specify the following settings: • In the **Name** box, type **long-queue** • In the **Description** box, type **Queue too long** • Enter **10** so that the threshold is: **Whenever:** ApproximateNumberOfMessagesVisible **is:** >= `10` **for: 1** consecutive period(s)
6.13.8	Under the **Actions** section, click **Delete** on the existing Notification Action.
6.13.9	Click **+ AutoScaling Action**, and specify the following settings: • In the **From the group** drop-down list, select **worker-group**. • In the **Take this action** drop-down list, select **Increase Group Size – Add 1 instance**.
6.13.10	Click **Create Alarm**. **Note** These configuration settings will automatically **add 1 instance** to your Auto Scaling group called **worker-group** whenever the **input** queue has **10 or more messages visible** within a **1 minute period**.

Monitoring the Cluster, Continued

Task 6.14: Test your Auto Scaling group

To test and verify that your Auto Scaling group adds and removes instances:

Step	Action
6.14.1	Switch to your **CloudWatch Dashboard**. Your **long-queue** alarm should appear within the **Alarm Summary** section.
6.14.2	Refresh the **Alarm Summary** section until it displays **"You have 1 alarm in ALARM state…"**
6.14.3	Switch back to the **EC2 Dashboard.** In the navigation pane, click **Instances**. Within a few minutes, you should be able to see the additional instances that have been launched by your Auto Scaling group. As a result, the messages in your queue will be processed more quickly.
6.14.4	In the navigation pane, click **Auto Scaling Groups** and select your **worker-group**. On the **Scaling History** tab, you can verify that Auto Scaling has launched new instances.

Ending your lab

When you are done, follow the instructions in Appendix B to end your lab.

Optional

Introduction

This part of the lab is a challenge wherein step-by-step instructions are not given.

Challenge

Currently, your Auto Scaling group only scales out. Any instances that the group launches in order to handle a surplus of messages will continue to run even after those messages have been processed.

Modify your environment's settings so that your Auto Scaling group scales in once the number of messages in your input queue drops below 10. You can use the message producer script to load up your input queue again, and verify that your environment's new settings successfully scale in your instances.

Ending your lab

When you are done, follow the instructions in **Appendix B** to end your lab.

Conclusion

Achievements

Having completed this lab, you are able to:

- Bootstrap an EC2 instance using User Data.
- Create an AMI from a running instance.
- Use the AWS Management Console to create an SQS queue.
- Create an Auto Scaling group with Scaling Policies based on an SQS queue.
- Use the AWS Management Console to pass messages to, and read messages from, an SQS queue.

Your results are verified by your ability to complete the lab in the allotted time. When you have completed the lab, your instructor will facilitate a discussion of your results.

Appendix A: Downloading your key pair or remote desktop shortcut

Introduction

Access to instances used in your labs requires a secure connection using either an SSH client (for Linux instances) or Remote Desktop Connection (for Windows instances). The qwikLABS page for your lab provides access to the key pair or remote desktop shortcut used to access these instances.

Downloading your key or shortcut

To download your key pair or remote desktop shortcut:

Step	Action
1	From the qwikLABS page in your browser, in the **Connection Details** section, click the **Download PEM/PPK** drop-down and choose either the PPK file (for PuTTY clients on Windows) or the PEM file (for Macintosh clients).
2	Save the file to your `\Downloads` folder or an alternate location.
3	If your lab requires access to a Windows instance, in the **RDP** section, click the **Download RDP File** option.
4	Save the file to your `\Downloads` folder or an alternate location.
5	In the **RDP** section of the page, make a note of the RDP **Password**. You will need it to connect to your Windows instance using the RDP shortcut.

Appendix B: Terminating the qwikLABS lab environment

Ending your labs

Ending your lab

When you are finished with your lab, terminate the lab environment using the following steps.

Step	Action
1	To log out of the AWS Management Console, from the menu, click **awsstudent @ [YourAccountNumber]** and choose **Sign out** (where [YourAccountNumber] is the AWS account generated by qwikLABS).
2	Close any active SSH client sessions or remote desktop sessions.
3	Click the **End Lab** button on the qwikLABS lab details page in your browser.
4	When prompted for confirmation, click **OK**.
5	For **My Rating**, rate the lab (using the applicable number of stars), optionally type a **Comment**, and click **Submit**. **Note**: The number of stars indicates the following: 1 star = very dissatisfied, 2 stars = dissatisfied, 3 stars = neutral, 4 stars = satisfied, and 5 stars = very satisfied. Also, you may close the dialog if you do not wish to provide feedback.