AWS Technical Essentials
Lab Guide
Version 4.0

100-ESS-40-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/.

Contents

Lab 1
Build a VPC and Deploy a Web Server

Overview

In this lab session, you will use Amazon Virtual Private Cloud (VPC) to create your own VPC and add additional components to it to produce a customized network. You will create security groups for your EC2 instance. You will configure and customize the EC2 instance to run a web server and launch it into the VPC.

Objectives

After completing this lab, you will be able to:

- Create a VPC
- Create subnets
- Configure a security group
- Launch an EC2 instance into the VPC

Prerequisites

This lab requires the following:

- Access to a computer with Wi-Fi running Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat).
- The qwikLABS lab environment is not accessible using an iPad or tablet device, but you can use these devices to access the student guide.
- An Internet browser such as Chrome, Firefox, or IE9 or later (previous versions of Internet Explorer are not supported).

Duration

This lab will take approximately 45 minutes.

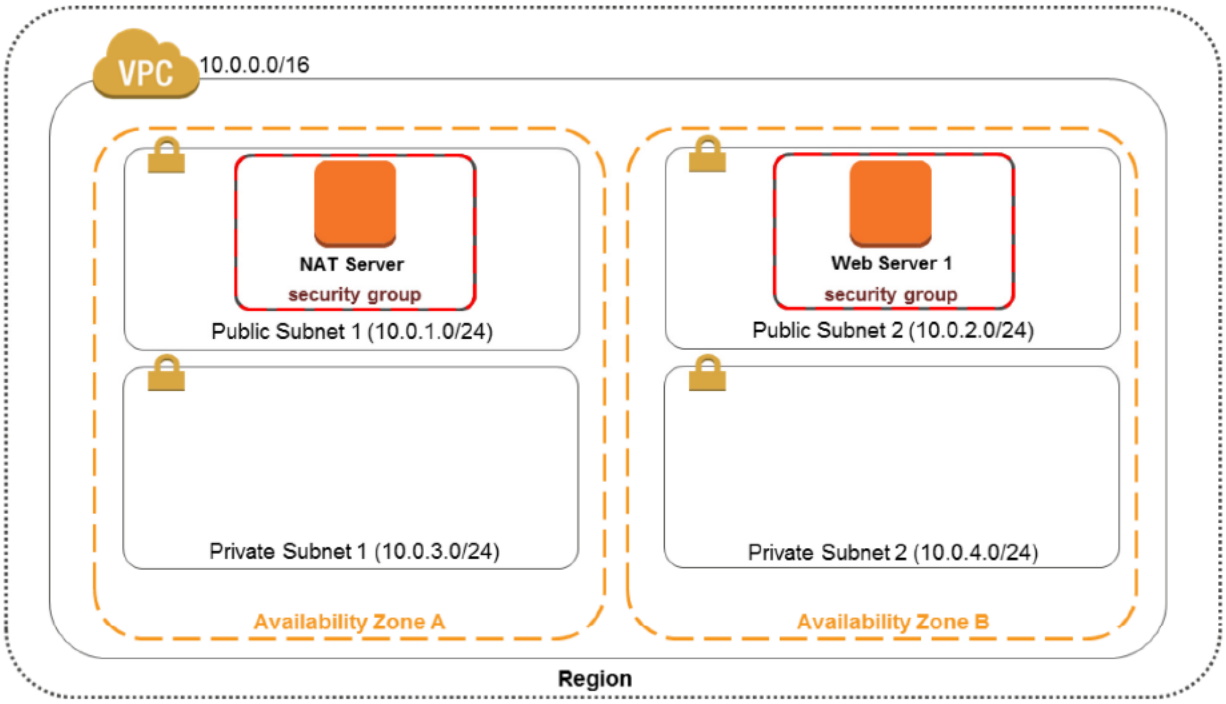
Task 1: Create Your VPC

Overview

In this section you will create your VPC.

Scenario

In this lab you will build the following infrastructure:



Task 1.1: Create Your VPC

In this task you will create a VPC with two subnets in one Availability Zone.

1.1.1 In the **AWS Management Console**, on the **Services** menu, click **VPC**.

1.1.2 Click **Start VPC Wizard**.

1.1.3 In the navigation pane, click **VPC with Public and Private Subnets**.

1.1.4 Click **Select**.

- Enter the following information:
- **IP CIDR block**: **10.0.0.0/16**
- **VPC name**: **My Lab VPC**
- **Public subnet**: **10.0.1.0/24**
- **Availability Zone**: Click an Availability Zone
- **Public subnet name**: **Public Subnet 1**
- **Private subnet**: **10.0.3.0/24**
- **Availability Zone**: Click the same Availability Zone as the Public Subnet
- **Private subnet name**: **Private Subnet 1**

1.1.5 In **Specify the details of your NAT gateway**, click **Use a NAT instance instead** on the right of the screen.

1.1.6 Select the first instance type listed in **Instance type** (example, t2.micro).

1.1.7 For **Key pair name**, select the **qwikLABS** key pair.

1.1.8 Click **Create VPC**.

1.1.9 After your VPC has been created, you will see a page stating your VPC was successfully created. Click **OK**.

Task 1.2: Create Additional Subnets

In this task you will create two additional subnets in another Availability Zone and associate the subnets with existing route tables.

1.2.1 In the navigation pane, click **Subnets**.

1.2.2 Click **Create Subnet**.

1.2.3 In the **Create Subnet** dialog box, enter the following details:

- **Name tag**: **Public Subnet 2**
- **VPC**: Click **My Lab VPC**
- **Availability Zone**: Select a different Availability Zone than you selected for Private Subnet 1 and Public Subnet 1 in the previous task.
- **CIDR block**: **10.0.2.0/24**

1.2.4 Click **Yes, Create**.

1.2.5 Click **Create Subnet**.

1.2.6 In the **Create Subnet** dialog box, enter the following details:

- **Name tag**: **Private Subnet 2**
- **VPC**: Click **My Lab VPC**
- **Availability Zone**: Select the same Availability Zone that you selected for Public Subnet 2.
- **CIDR block**: **10.0.4.0/24**

1.2.7 Click **Yes, Create**.

1.2.8 Select **Public Subnet 2**, ensure all other subnets are cleared, and then click **Route Table** in the lower pane. Scroll down and verify that the **Target** for **Destination 0.0.0.0/0** contains the prefix **igw**. If it does not, click **Edit** and click the other route table in the **Change to:** list that changes the **Target** for **Destination 0.0.0.0/0** to contain the prefix **igw**. Click **Save**.

1.2.9 Select **Private Subnet 2**, ensure all other subnets are cleared, and then click **Route Table** in the lower pane. Scroll down and verify that the **Target** for **Destination 0.0.0.0/0** contains the prefix **eni**. If it does not, click **Edit** and click the other route table in the **Change to:** list that changes the **Target** for **Destination 0.0.0.0/0** to contain the prefix **eni**. Click **Save**.

Task 1.3: Create a VPC Security Group

You will create a VPC security group that permits access for web and SSH traffic.

1.3.1 In the navigation pane, click **Security Groups**.

1.3.2 Click **Create Security Group**.

1.3.3 In the **Create Security Group** dialog box, enter the following information:

- **Name tag**: **WebSecurityGroup**
- **Group name**: **WebSecurityGroup**
- **Description**: **Enable HTTP access**
- **VPC**: Click the VPC you created in Task 1.1 (**My Lab VPC**)

1.3.4 Click **Yes, Create**.

1.3.5 Select **WebSecurityGroup**.

1.3.6 Click the **Inbound Rules** tab.

1.3.7 Click **Edit**.

1.3.8 For **Type**, click **HTTP (80)**.

1.3.9 Click in the **Source** box and type **0.0.0.0/0**

1.3.10 Click **Add another rule**.

1.3.11 For **Type**, click **SSH (22)**.

1.3.12 Click in the **Source** box and type **0.0.0.0/0**

1.3.13 Click **Save**.

Task 2: Launch Your Web Server

Overview

After you create your VPC, you will launch an EC2 instance into it and bootstrap it to act as a web server.

Command Reference File

Use the command reference file when copying text provided in this lab manual. The command reference file is available by clicking the ADDL. INFO button of your lab in qwikLABS.

You should not copy and paste commands directly from this lab manual, because the manual's rich formatting may inject characters that could introduce errors to your lab experience. Download the reference file to your computer instead.

Task 2.1: Launch Your First Web Server Instance

This task walks you through launching an EC2 instance into your VPC. This instance will act as your web server.

2.1.1 On the **Services** menu, click **EC2**.

2.1.2 Click **Launch Instance**.

2.1.3 In the row for **Amazon Linux AMI**, click **Select**.

2.1.4 On **Step 2: Choose an Instance Type** page, make sure **t2.micro** is selected and click **Next: Configure Instance Details**.

2.1.5 On **Step 3: Configure Instance Details** page, enter the following information and leave all other values with their default:

- **Network**: Click the VPC that you created in Task 1.1 (**My Lab VPC**).
- **Subnet**: Click **Public Subnet 2 (10.0.2.0/24)** you created in Task 1.2.
- **Auto-assign Public IP**: Click **Enable**

2.1.6 Scroll down and expand the **Advanced Details** section.

2.1.7 Copy the following user data from the command reference file and paste it into the **User data** box, ensuring **As text** is selected:

```
#!/bin/bash -ex

yum -y update

yum -y install httpd php mysql php-mysql

chkconfig httpd on

/etc/init.d/httpd start

if [ ! -f /var/www/html/lab2-app.tar.gz ]; then

cd /var/www/html

wget https://us-west-2-aws-staging.s3.amazonaws.com/awsu-ilt/AWS-100-ESS/v4.0/lab-2-configure-website-datastore/scripts/lab2-app.tar.gz

tar xvfz lab2-app.tar.gz

chown apache:root /var/www/html/lab2-app/rds.conf.php

fi
```

2.1.8 Click **Next: Add Storage**.

2.1.9 Click **Next: Tag Instance**.

2.1.10 On **Step 5: Tag Instance** page, enter the following information:

- **Key**: **Name**
- **Value**: **Web Server 1**

2.1.11 Click **Next: Configure Security Group**.

2.1.12 On **Step 6: Configure Security Group** page, click **Select an existing security group** and then select the security group you created in Task 1.3 (**WebSecurityGroup)**.

2.1.13 Click **Review and Launch**.

2.1.14 Review the instance information and click **Launch**.

2.1.15 Click **Choose an existing key pair,** click the **qwikLABS** key pair, select the acknowledgement check box, and then click **Launch Instances**.

2.1.16 Scroll down and click **View Instances**.

2.1.17 You will see two instances – **Web Server 1** and the NAT instance launched by the VPC Wizard.

2.1.18 Wait until **Web Server 1** shows *2/2 checks passed* in the **Status Checks** column. This will take 3-5 minutes. Use the refresh icon at the top right to check for updates.

2.1.19 Select **Web Server 1** and copy the **Public DNS** value.

2.1.20 Paste the **Public DNS** value in a new web browser window or tab and press Enter. You will see the **Amazon Linux AMI Test Page**.

Lab Complete

Congratulations! You have successfully completed creating a VPC and launching an EC2 instance into it. To clean up your lab environment, do the following:

1. Log out of the **AWS Management Console** by clicking **awsstudent** in the top right corner and click **Sign Out**.
2. Return to the **qwikLABS** page where you launched your lab and click **End**.

Lab 2
Configure a Relational Data Store for Your Website

Overview

This lab builds on the previous lab. It walks you through launching an Amazon Relational Database Service (RDS) DB instance. You will configure the web server that you previously created to use Amazon RDS for its relational database management system (RDBMS) needs. This lab is designed to reinforce the concept of leveraging an AWS managed database instance for solving relational database needs.

Objectives

After completing this lab, you will be able to do the following:

- Launch an Amazon RDS DB instance with high availability
- Configure the DB instance to permit connections from your web server
- Open a web application and interact with your database

Prerequisites

This lab requires the following:

- Access to a computer with Wi-Fi running Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat)
- The qwikLABS lab environment is not accessible using an iPad or tablet device, but you can use these devices to access the student guide.
- An Internet browser such as Chrome, Firefox, or IE9 or later (previous versions of Internet Explorer are not supported)

Duration

This lab will take approximately 45 minutes.

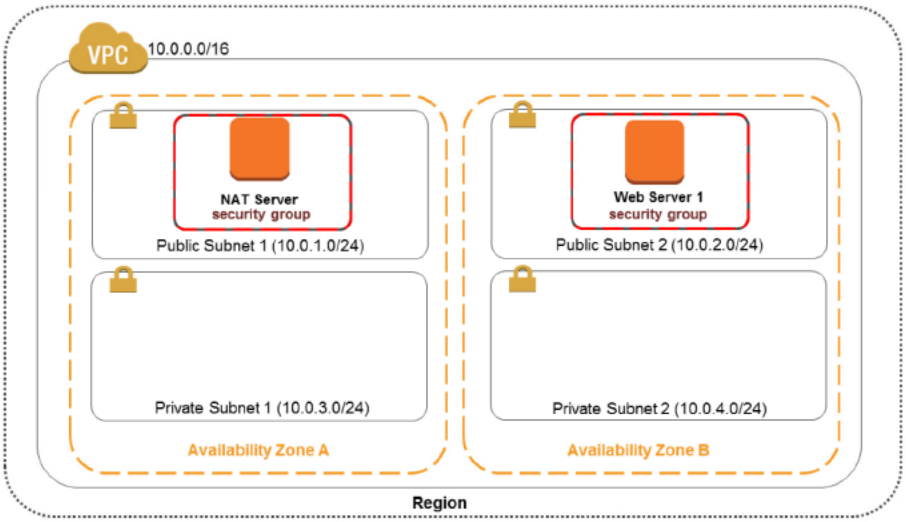
Task 1: Launch an Amazon RDS DB Instance

Overview

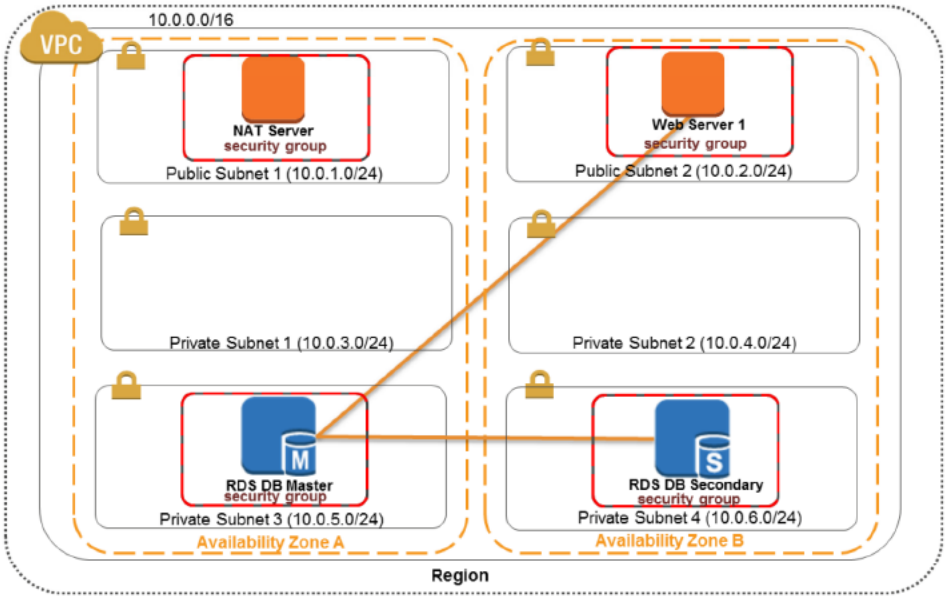
In this task, you will launch an Amazon RDS DB instance backed by MySQL.

Scenario

You will be starting with the following infrastructure:



You will build the following infrastructure:



Task 1.1: Create a VPC Security Group for the RDS DB Instance

In this task, you will create a VPC security group to allow your web server to access your RDS DB instance.

1.1.1 In the **AWS Management Console,** on the **Services** menu, click **VPC**.

1.1.2 In the navigation pane, click **Security Groups**.

1.1.3 Click **Create Security Group**.

1.1.4 In the **Create Security Group** dialog box, enter the following details:

- **Name tag**: **DBSecurityGroup**
- **Group name**: **DBSecurityGroup**
- **Description**: **DB Instance Security Group**
- **VPC**: Click **My Lab VPC**

1.1.5 Click **Yes, Create**.

1.1.6 Select **DBSecurityGroup** you just created and ensure that all other security groups are cleared.

1.1.7 Click the **Inbound Rules** tab, and then click **Edit**.

1.1.8 Create an inbound rule with the following details:

- **Type**: **MySQL/Aurora (3306)**
- **Protocol**: **TCP(6)**
- **Source**: Click **WebSecurityGroup**

1.1.9 Click **Save**.

Task 1.2: Create Private Subnets for Your Amazon RDS Instances

In this task you will create two private subnets for your Amazon RDS instances.

1.2.1 In the navigation pane, click **Subnets**.

1.2.2 Select **Public Subnet 1**, clear all other subnets, and scroll down to the **Summary** tab in the lower pane. Take note of the **Availability Zone** for this subnet.

1.2.3 Select **Public Subnet 2**, clear all other subnets, and scroll down to the **Summary** tab in the lower pane. Take note of the **Availability Zone** for this subnet.

1.2.4 Click **Create Subnet**.

1.2.5 In **Create Subnet** dialog box, enter the following details:

- **Name tag**: **Private Subnet 3**
- **VPC**: Select **My Lab VPC**
- **Availability Zone**: Click the same Availability Zone that you noted for Public Subnet 1 previously
- **CIDR block**: **10.0.5.0/24**

1.2.6 Click **Yes, Create**.

1.2.7 Click **Create Subnet**.

1.2.8 In **Create Subnet** dialog box, enter the following details:

- **Name tag**: **Private Subnet 4**
- **VPC**: Click **My Lab VPC**
- **Availability Zone**: Click the same Availability Zone that you noted for Public Subnet 2 previously
- **CIDR block**: **10.0.6.0/24**

1.2.9 Click **Yes, Create**.

1.2.10 Select **Private Subnet 3**, ensure all other subnets are cleared, and then click **Route Table** in the lower pane. Scroll down and verify that the **Target** for **Destination 0.0.0.0/0** contains the prefix **eni**. If it does not, or there is no **Destination 0.0.0.0/0**, click **Edit** and click the **Private Route Table** in the **Change to:** drop-down list that changes the **Target** for **Destination 0.0.0.0/0** to contain the prefix **eni**. Click **Save**.

1.2.11 Repeat the previous step for **Private Subnet 4**.

Task 1.3: Create a DB Subnet Group

In this task, you will create a DB subnet group. Each DB subnet group should have subnets in at least two Availability Zones in a given region.

1.3.1 On the **Services** menu, click **RDS**.

1.3.2 In the navigation pane, click **Subnet Groups**.

1.3.3 Click **Create DB Subnet Group**.

1.3.4 On the **Create DB Subnet Group** page, enter the following details:

- **Name**: **dbsubnetgroup**
- **Description**: **Lab DB Subnet Group**
- **VPC ID**: Click **My Lab VPC**

1.3.5 For **Availability Zone**, click the Availability Zone you selected for **Private Subnet 3**.

1.3.6 For **Subnet ID**, click **10.0.5.0/24**, then click **Add**.

1.3.7 For **Availability Zone**, click the Availability Zone you selected for **Private Subnet 4**.

1.3.8 For **Subnet ID**, click **10.0.6.0/24**, then click **Add**.

1.3.9 Click **Create**.

1.3.10 If you do not see your new subnet group, click the refresh icon in the upper-right corner of the console.

Task 1.4: Create an RDS DB Instance

In this task you will configure and launch your MySQL-backed Amazon RDS DB instance.

1.4.1 On the **Services** menu, click **RDS**.

1.4.2 Click **Get Started Now**.

1.4.3 Click **MySQL** > **Select**.

1.4.4 Under **Production**, click **MySQL**.

1.4.5 Click **Next Step**.

1.4.6 On the **Specify DB Details** page, enter the following details:

- **DB Instance Class**: Click the first option in the list
- **Multi-AZ Deployment**: Click **Yes**
- **DB Instance Identifier**: **labdbinstance**
- **Master Username**: **labuser**
- **Master Password**: **labpassword**
- **Confirm Password**: **labpassword**

1.4.7 Click **Next Step**.

1.4.8 On the **Configure Advanced Settings** page, enter the following details and leave all other values with their default:

- **VPC**: **My Lab VPC**
- **Subnet Group**: **dbsubnetgroup**
- **Publicly Accessible**: **No**
- **VPC Security Group(s)**: **DBSecurityGroup (VPC)**
- **Database Name**: **sampledb**

1.4.9 Click **Launch DB Instance**.

1.4.10 Click **View Your DB Instances**.

1.4.11 Select **labdbinstance** and wait until the **Endpoint** is *available* or *modifying* – this may take up to 10 minutes. Use the refresh icon in the top right corner to check for updates.

1.4.12 Copy and save the **Endpoint**, making sure to not copy the :3306 - your **Endpoint** should look similar to the following example: qr7g2qco3oeq5h.cze6p5rivinc.us-west-2.rds.amazon.com

Task 2: Interact with Your Database

Overview

In this task you will interact with your database through a PHP web application that was deployed to the web server you created in the previous lab.

Task 2.1: Access the Database Web Application

You will open a web application running on your web server.

2.1.1 On the **Services** menu, click **EC2**.

2.1.2 In the navigation pane, click **Instances**.

2.1.3 Select **Web Server 1**, ensure that all other instances are cleared, and scroll down to view the **Description** tab in the lower pane.

2.1.4 Copy the **Public IP** address of **Web Server 1**.

2.1.5 Paste the IP address in a new browser tab or window. A web application will be displayed with the web server's instance meta-data.

2.1.6 Click the **RDS** link under the AWS logo.

2.1.7 Enter the following details:

- **Endpoint**: Paste the endpoint you copied previously, making sure to omit the :3306
- **Database**: **sampledb**
- **Username**: **labuser**
- **Password**: **labpassword**

2.1.8 Click **Submit**. The connection string will be displayed and then the page will be redirected. Two new records will be added to the address table and displayed.

2.1.9 To add another contact, click **Add Contact** and enter a **Name**, **Phone**, and **Email** and then click **Submit**.

2.1.10 To edit a contact, click **Edit**, modifying one of the fields, and then click **Submit**.

2.1.11 To remove a record, click **Remove**.

2.1.12 You can now close this browser tab or window.

Lab Complete

Congratulations! You have successfully completed configuring a relational data store for your website. To clean up your lab environment, do the following:

1. Log out of the **AWS Management Console** by clicking **awsstudent** in the top right corner and click **Sign Out**.
2. Return to the **qwikLABS** page where you launched your lab and click **End**.

Lab 3
Manage Your Infrastructure

Overview

This lab builds on the previous lab and walks you through using the Elastic Load Balancing (ELB) and Auto Scaling services to load balance and auto scale your infrastructure.

Objectives

After completing this lab, you will be able to:

- Create an Amazon Machine Image (AMI) from a running instance
- Add a load balancer
- Create a launch configuration
- Create an Auto Scaling group
- Auto scale new instances within a private subnet
- Create Amazon CloudWatch alarms
- Monitor performance of your infrastructure

Prerequisites

This lab requires the following:

- Access to a computer with Wi-Fi running Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat)
- The qwikLABS lab environment is not accessible using an iPad or tablet device, but you can use these devices to access the student guide.
- An Internet browser such as Chrome, Firefox, or IE9 or later (previous versions of Internet Explorer are not supported)

Duration

This lab will take approximately 45 minutes.

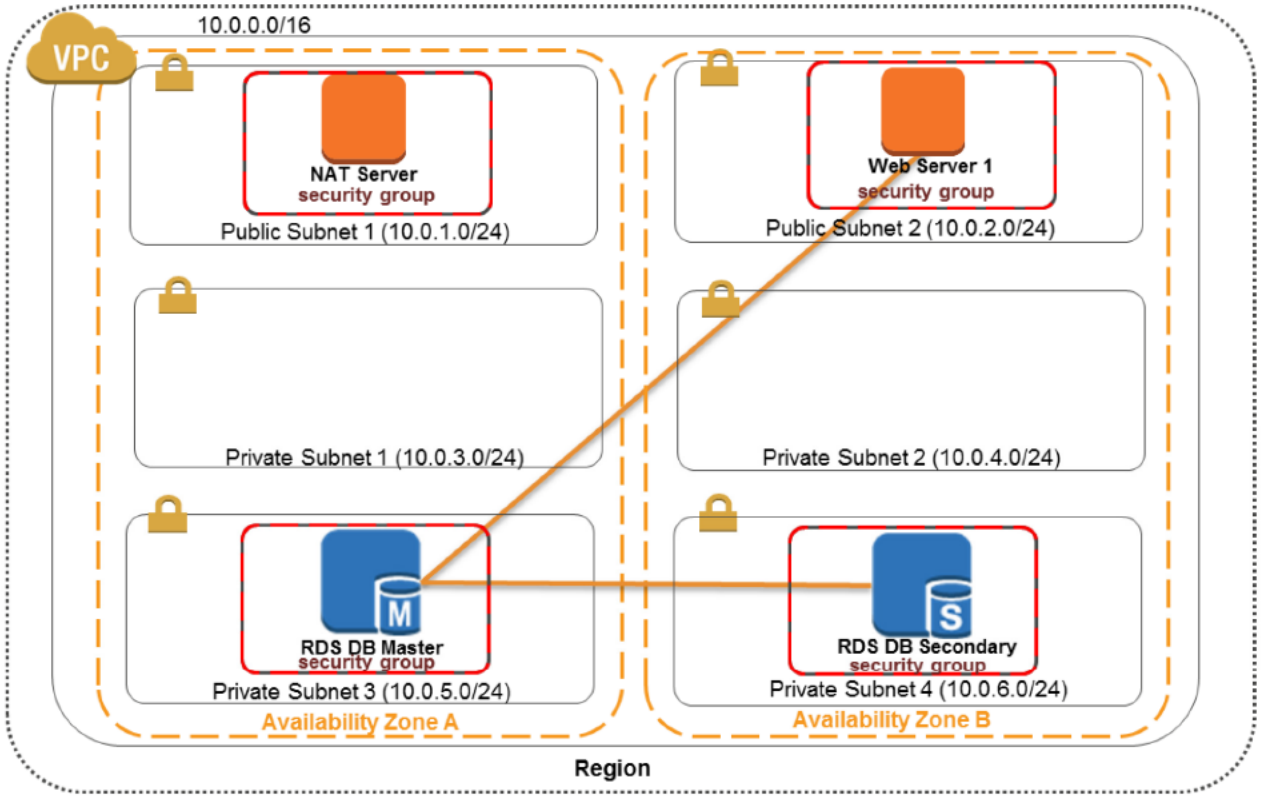
Task 1: Auto Scaling

Overview

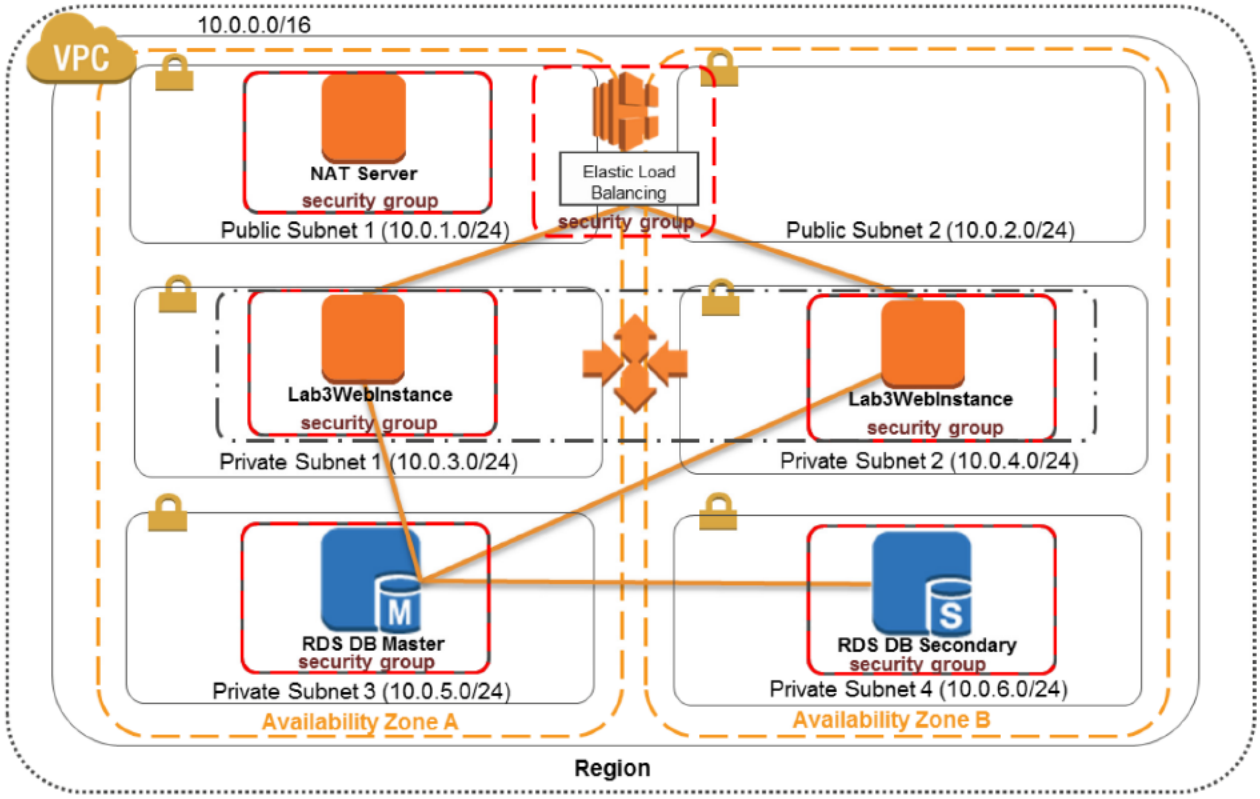
In this task you will create and auto scale your infrastructure.

Scenario

You will be starting with the following infrastructure:



You will build the following infrastructure:



Task 1.1: Create an AMI For Auto Scaling

In this task you will create an AMI as the starting point for launching new instances to use with Auto Scaling.

1.1.1 In the **AWS Management Console**, on the **Services** menu, click **EC2**.

1.1.2 In the navigation pane, click **Instances**.

1.1.3 Verify that the **Status Checks** for **Web Server 1** displays *2/2 checks passed*. If it doesn't, wait until it does before proceeding to the next step. Use the refresh button in the top right corner to check for updates.

1.1.4 Right click on **Web Server 1**, click **Image** > **Create Image**.

1.1.5 Enter the following details, leaving the other values with their default:

- **Image name**: **Web Server AMI**
- **Image description**: **Lab 3 AMI for Web Server**

1.1.6 Click **Create Image**.

1.1.7 The confirmation screen will display the **AMI ID** for your new AMI.

1.1.8 Click **Close**.

Task 1.2: Add a Load Balancer

In this task you will create a load balancer to balance the load of traffic across several EC2 instances in two Availability Zones.

1.2.1 In the navigation pane, click **Load Balancers**.

1.2.2 Click **Create Load Balancer**.

1.2.3 Enter the following details, leaving the remaining values with their default:

- **Load Balancer name**: **Lab3ELB**
- **Create LB inside**: Select **My Lab VPC**
- **Select Subnets**: Click the + to select **Public Subnet 1** and **Public Subnet 2**

1.2.4 Click **Next: Assign Security Groups**.

1.2.5 Clear the **default** security group and select the security group that contains **WebSecurityGroup** in the name and a **Description** of **Enable HTTP access**.

1.2.6 Click **Next: Configure Security Settings**.

1.2.7 You will not be configuring a security listener in this lab, so click **Next: Configure Health Check**.

1.2.8 Enter the following details, leaving the remaining values with their default:

- **Ping Path**: **/index.php** (Note this is different than the default value)
- **Health Check Interval**: **6**
- **Healthy Threshold**: Select **2**

1.2.9 Click **Next: Add EC2 Instances**.

1.2.10 You will add EC2 instances to the load balancer in a subsequent task. Click **Next: Add Tags**.

1.2.11 Click **Review and Create**.

1.2.12 Review the configuration of your load balancer and click **Create**.

1.2.13 Click **Close**.

1.2.14 Select **Lab3ELB** and on the **Description** tab in the lower pane, make note of the **DNS Name** of your load balancer, making sure to omit (A Record).

Task 1.3: Create a Launch Configuration and an Auto Scaling Group

In this task you will create a launch configuration for your Auto Scaling group.

1.3.1 In the navigation pane, click **Launch Configurations**.

1.3.2 Click **Create Auto Scaling group** > **Create launch configuration**.

1.3.3 In the navigation pane, click **My AMIs**.

1.3.4 To select the **Web Server AMI** you created earlier, click **Select**.

1.3.5 Accept the **t2.micro** selection and click **Next: Configure details**.

1.3.6 Enter the following details, leaving the remaining values with their default:

- **Name**: **Lab3Config**
- **Monitoring**: Select **Enable CloudWatch detailed monitoring**

1.3.7 Click **Next: Add Storage**.

1.3.8 Click **Next: Configure Security Group**.

1.3.9 Click **Select an existing security group** and select the security group that contains **WebSecurityGroup** in the name and a **Description** of **Enable HTTP access**.

1.3.10 Click **Review**.

1.3.11 Review the details of your launch configuration and click **Create launch configuration**.

1.3.12 Click **Choose an existing key pair**, select the **qwikLABS key pair**, select the acknowledgement box, and click **Create launch configuration**.

1.3.13 Enter the following details for your auto scaling group, leaving the remaining values with their default:

- **Group name**: **Lab3ASGroup**
- **Group size**: Start with **2** instances
- **Network**: Select **My Lab VPC**
- **Subnet**: Select **Private Subnet 1 (10.0.3.0/24)** and **Private Subnet 2 (10.0.4.0/24)**

1.3.14 Scroll down, expand **Advanced Details**, and select **Receive traffic from Elastic Load Balancer(s)**.

1.3.15 Click in the **Load Balancing** text box and then click **Lab3ELB**.

1.3.16 Enter the following details, leaving the remaining values with their default:

- **Health Check Type**: Select **ELB**
- **Monitoring**: Select **Enable CloudWatch detailed monitoring**

1.3.17 Click **Next: Configure scaling policies**.

1.3.18 Click **Use scaling policies to adjust the capacity of this group**.

1.3.19 Modify the **Scale between** textbox to scale between **2** and **6** instances.

1.3.20 In **Increase Group Size**, for **Execute policy when**, click **Add New Alarm**.

1.3.21 Verify that **Send a notification to:** is selected, then click **create topic** (creating an email notification is optional and you may skip the applicable steps marked with an * for the remainder of the lab – you must clear **Send a notification to:** if you do not want to receive an email notification).

1.3.22 Enter the following details, leaving the remaining with their default values:

- ***Send a notification to**: **ASTopic**
- ***With these recipients**: Enter an email address you have access to
- **Whenever**: **Average of CPU Utilization**
- **Is >= 65 Percent**
- **For at leas**t: **1** consecutive period(s) of **1 minute**
- **Name of alarm**: **HighCPUUtilization**

1.3.23 Click **Create Alarm**.

1.3.24 Remaining in **Increase Group Size**, enter the following details:

- **Take the action**: select **Add**, type **1**, select **instances**, type **65**
- **Instances need**: **60** seconds to warm up after each step

1.3.25 In **Decrease Group Size**, for **Execute policy when**, click **Add New Alarm**.

1.3.26 *Verify that **Send a notification to:** is selected and select the **ASTopic (<your email address>)** – clear if you do not wish to receive an email notification.

1.3.27 Enter the following details, leaving the remaining with their default values:

- **Whenever**: **Average of CPU Utilization**
- **Is <= 20 Percent**
- **For at least**: **1** consecutive period(s) of **1 minute**
- **Name of alarm**: **LowCPUUtilization**

1.3.28 Click **Create Alarm**.

1.3.29 Remaining in **Decrease Group Size**, enter the following details:

- **Take the action**: select **Remove**, type **1**, select **instances**, type **20**

1.3.30 Click **Next: Configure Notifications**.

1.3.31 Click **Next: Configure Tags**.

1.3.32 Enter the following details, leaving the other values with their default:

- **Key**: **Name**
- **Value**: **Lab3WebInstance**

1.3.33 Click **Review**.

1.3.34 Review the details of your Auto Scaling group, then click **Create Auto Scaling group**.

1.3.35 Click **Close** when your Auto Scaling group has been created.

1.3.36 * You will receive an email to confirm your subscription to notifications about the Auto Scaling group. Open that email and click the **Confirm subscription link**.

Task 1.4: Verify Auto Scaling is Working and Add Instances to Load Balancer

In this task you will verify Auto Scaling is working correctly.

1.4.1 In the navigation pane, click **Instances**.

1.4.2 You will see four instances: **Web Server 1**, **NAT Server**, and two new instances labeled as **Lab3WebInstance.**

1.4.3 In the navigation pane, click **Load Balancers**.

1.4.4 Select **Lab3ELB**, scroll down and click the **Instances** tab. You will see your **Lab3WebInstance** listed for this load balancer.

1.4.5 Wait until the instance displays a **Status** of *InService* in the **Instances** tab for **Lab3ELB**. Use the refresh button in the top right corner to check for updates.

1.4.6 Your load balancer will display **Yes** under the **Healthy?** field for the Availability Zone the instance is running in.

Task 2: Monitor Your Infrastructure

Overview

You have created an Auto Scaling group with a minimum of two instances and a maximum of six instances. You created Auto Scaling policies to increase and decrease the group by one instance. You created Amazon CloudWatch alarms to trigger these policies when the aggregate average CPU of the group is >= 65% and <= 20% respectively. Currently two instances are running because the minimum size is two and the group is currently not under any load. You will now monitor this infrastructure using the CloudWatch alarms that you created.

Task 2.1: Test Auto Scaling

In this task you will test the Auto Scaling configuration you have implemented.

2.1.1 On the **Services** menu, click **CloudWatch**.

2.1.2 In the navigation pane, click **Alarms** (not **ALARM**).

2.1.3 You will see the two alarms **HighCPUUtilization** and **LowCPUUtilization**. **LowCPUUtilization** should have a **State** of **Alarm** and **HighCPUUtilization** should have a **State** of **OK**. This is because the current group CPU Utilization is < 20%. Auto Scaling is not removing any instances because the group size is currently at its minimum (2).

2.1.4 Paste the load balancer's DNS name that you copied in step 1.2.13 in a new browser window or tab.

2.1.5 Click **LOAD TEST** under the AWS logo. The application will load test your instances and auto-refresh in 5 seconds. You will see the Current CPU Load jump to 100%. The **Load Test** link triggers a simple background process.

2.1.6 On the **Services** menu, click **CloudWatch**.

In less than 5 minutes, you should see the **Low CPU** alarm status change to **OK** and the **High CPU** alarm status change to **ALARM**.

2.1.7 On the **Services** menu, click **EC2**.

2.1.8 In the navigation pane, click **Instances**.

2.1.9 You will now see more than two instances labeled **Lab3WebInstance** running.

2.1.10 Close the browser tab or window you opened in step 2.1.3.

Task 2.2: Optional: Terminate Web Server 1

In this task you will terminate Web Server 1 in Public Subnet 2. Your auto scaling group launched instances into private subnets and the original publically accessible web server is no longer needed.

2.2.1 On the **Services** menu, click **EC2**.

2.2.2 In the navigation pane, click **Instances**.

2.2.3 Rick click **Web Server 1** and click **Instance State** > **Terminate**.

Lab Complete

Congratulations! You have successfully completed managing your infrastructure using Auto Scaling and Elastic Load Balancing. To clean up your lab environment, do the following:

1. Log out of the **AWS Management Console** by clicking **awsstudent** in the top right corner and click **Sign Out**.
2. Return to the **qwikLABS** page where you launched your lab from and click **End**.

Appendix A
Logging in to the AWS Management Console

Introduction

In this appendix, you will learn how to log in to the student account created for you as part of this course.

About Student Accounts

Each lab in this course has a corresponding lab environment that is launched from the qwikLABS page. (Your instructor should have already supplied you with instructions for creating a qwikLABS account.) Whenever you launch a new lab, the qwikLABS environment creates a new AWS account for you. Within this AWS account, it creates an IAM user named **awsstudent**. When you terminate your lab, this account is recycled, and all resources associated with it are terminated.

Each time you start a new lab in this course, you will need to log in to your new lab environment as the user **awsstudent**, using the automatically-generated password provided for you on the qwikLABS page for that specific lab.

Task 1.1: Logging In

These instructions walk you through logging in to the AWS Management Console.

1.1.1 From the **Class Details** page in qwiklabs, find the current lab, and click **Select**.

1.1.2 Click **Start Lab**.

1.1.3 On the lab page, wait until the text Create in Progress... disappears from the screen. For some labs, this may happen instantly; for other labs, it may take anywhere from five to 10 minutes for your lab to initialize.

Note Make sure to wait until the lab creation process has completed before you move on to the next step.

1.1.4 Under **AWS Management Console**, you will see the fields **User Name** and **Password**. These are your AWS account credentials. Select and copy the **Password** field.

1.1.5 Click **Open Console**. This will open the AWS Management Console, pre-populating it with the AWS account ID created for you by qwikLABS.

Note You can right-click on this button and use your Web browser's "open in new tab" function to prevent this page from opening in a separate window.

1.1.6 On the new window or tab containing the AWS Management Console, you should see the Account ID already filled in. In the Username field, type **awsstudent**. In the Password field, paste the password that you copied from Step 3. Finally, click the **Sign In** button.

Note On rare occasions, the Account ID on your signing page may be blank. Consult your instructor for assistance on how to locate your qwikLABS account ID.