

Scale and Load Balance Your Architecture

AWS Lab 6

Evan Choi | Cisco CCNP | 4/28/2022

**Background Information**

Elastic Load Balancing distributes incoming traffic from applications across multiple Amazon EC2 instances automatically. It monitors the health of its targets, and routes traffic to only healthy targets. This enables you to have fault tolerance in your applications.

Auto Scaling can help you maintain your applications availability and allows you to scale your Amazon EC2 capacity in or out automatically based on conditions and rules that you define.

**Configuration Steps**

Accessing the AWS Management Console

1. Click the start lab button until you get the **“Lab status ready”** message

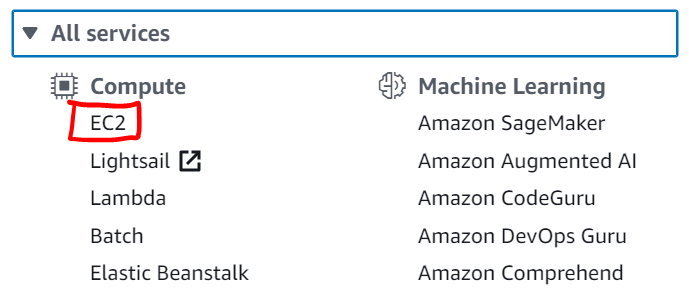


1. Open the AWS Management Console by clicking the AWS button

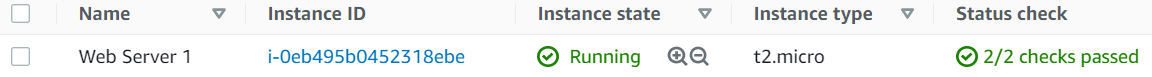


Creating an AMI for Auto Scaling

1. Click the **Services** box, then click **All Services,** and click on **EC2**



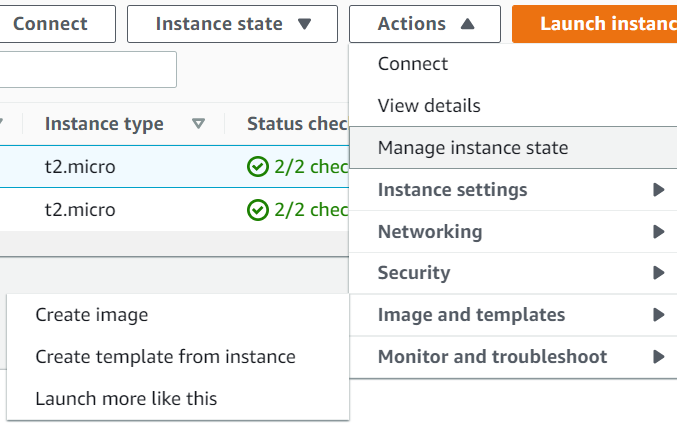
1. On the left menu, click **Instances**
2. Wait for the **Status Checks** for **Web Server 1** to show that 2/2 checks passed



1. Select **Web Server 1.** Click the **Actions** drop-down, click **Image and templates** then click **Create image.** Configure the following:

**Image name:** WebServerAMI

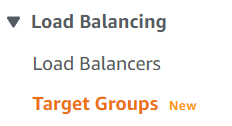
**Image description:** lab AMI for Web Server



1. Click the orange **Create image** button

Creating a Load Balancer

1. On the left menu, select **Target Groups**



1. Click the orange **Create target group** button and configure the following:

**Target Type:** Instances

**Target group name:** LabGroup

**VPC:** Select Lab VPC

1. Click the orange **Next** button
2. Scroll down and click **Create target group**
3. On the left menu, click **Load Balancers**
4. In the top left, click the blue **Create Load Balancer** button

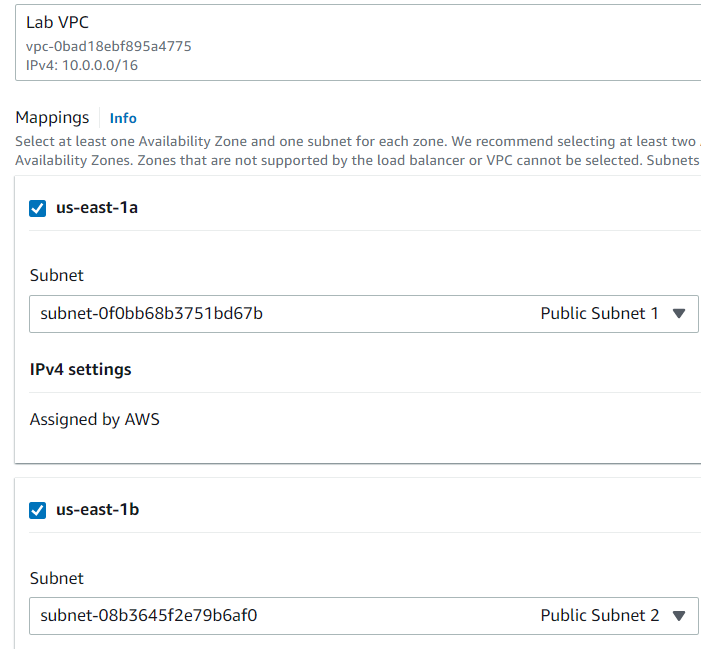


1. Under **Application Load Balancer,** click the **Create** button
2. For **Load balancer name,** type: LabELB
3. Scroll to the **Network mapping** section and do the following:

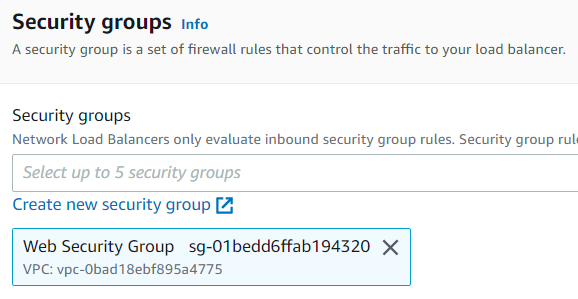
Under VPC, select: **Lab VPC**

Click the **first Availability Zone,** and select **Public Subnet 1**

Click the **second Availability Zone,** and select **Public Subnet 2**



1. Under **Security groups** select the drop-down for **Security groups** and choose **Web Security Group.** Click the **X** next to the default security group to delete it so only **Web Security Group** appears



1. Under **Listener HTTP: 80,** change the **Default action** to **LabGroup**
2. Scroll down and click the orange **Create load balancer** button
3. Click the orange **View load balancer** button

Creating a Launch Configuration and an Auto Scaling Group

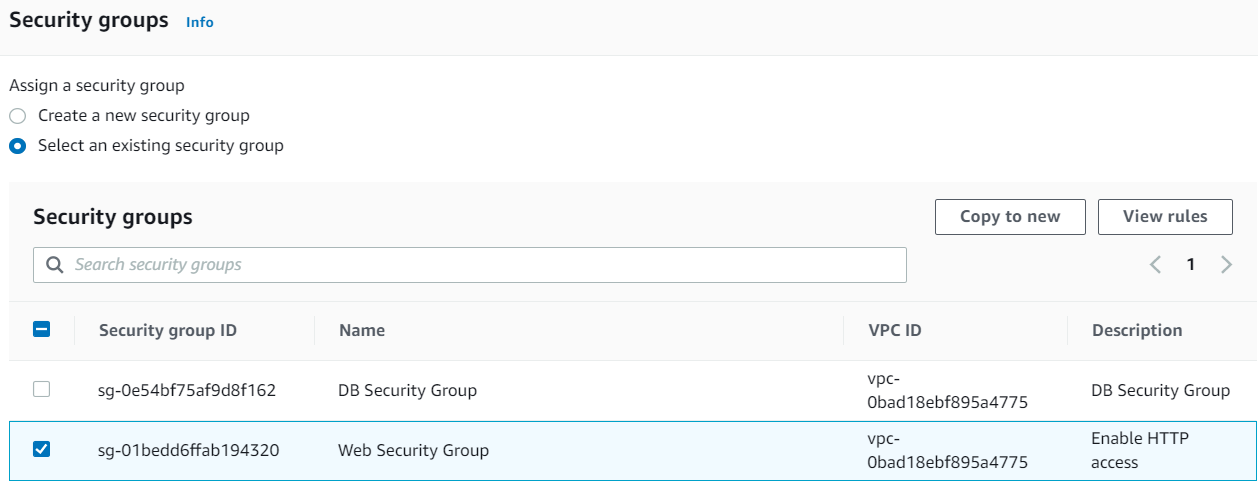
1. On the left menu, click, **Launch Configurations**
2. Click the orange **Create launch configuration** button
3. Configure the following:

**Launch configuration name:** LabConfig

**Amazon Machine Image (AMI):** WebServerAMI

**Instance type:** Click the **Choose instance type** button, select **t3.micro,** and click the orange **Choose** button. If you are in the **us-east-1** region, select **t2.micro** instead

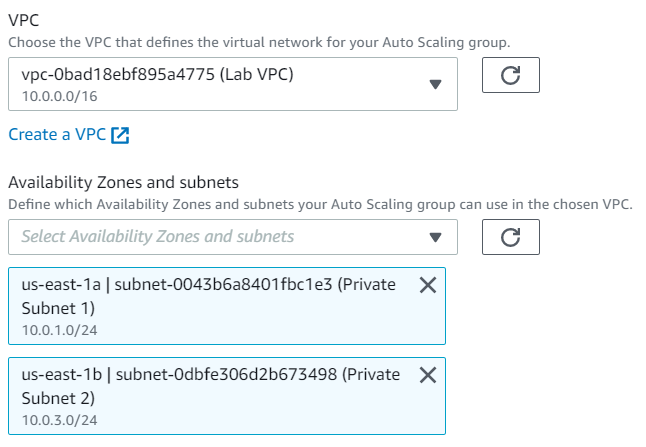
1. Under the **Additional configuration** section, and **Monitoring** area, select **Enable EC2 instance detailed monitoring within CloudWatch**
2. Under the **Security groups** section click **Select an existing security group,** and select **Web Security Group**



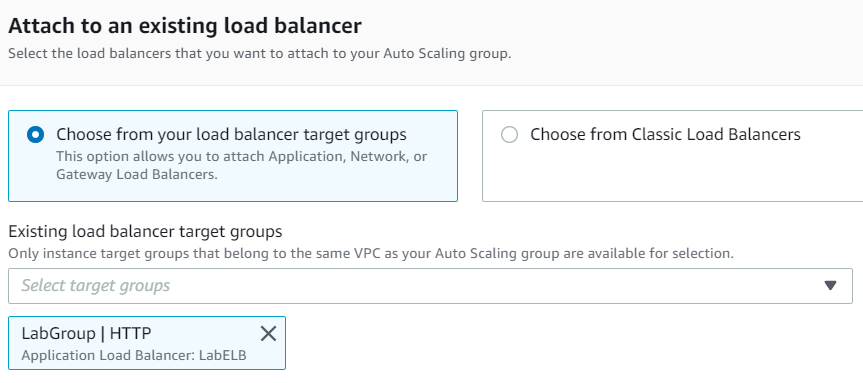
1. In the **Key pair** section, under the **Key pair options** area click **Choose an existing key pair,** and under the **Existing key pair** area click **vockey**
2. Click the orange **Create launch configuration** button
3. Click the box next to **LabConfig,** click the **Actions** drop-down, and choose **Create Auto Scaling Group**
4. Under the **Auto Scaling Group name** area, enter**:** Lab Auto Scaling Group
5. Click the orange **Next** button
6. In the **Network** section configure the following:

**Network:** Lab VPC

**Subnet:** Private Subnet 1 and Private Subnet 2



1. Click the orange **Next** button
2. In the **Load balancing – optional** section, select **Attach to an existing load balancer**
3. In the **Attach to an existing load balancer** section, click the drop-down and select **LabGroup**

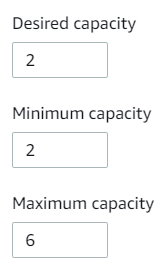


1. In the **Additional settings – options** menu, select **Enable group metrics collection within CloudWatch** and click the orange **Next** button
2. In the **Group size** menu, configure the following:

**Desired capacity:** 2

**Minimum capacity:** 2

**Maximum capacity:** 6



1. In the **Scaling policies** section, click **Target tracking scaling policy** and configure the following:

**Scaling policy name:** LabScalingPolicy

**Metric type:** Average CPU Utilization

**Target value:** 60

1. Click the orange **Next** button. Click the orange **Next** button that appears again
2. Click the **Add tag** button and configure the following:

**Key:** Name

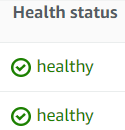
**Value:** Lab Instance



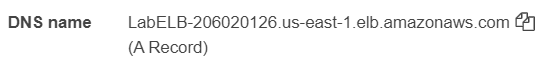
1. Click the orange **Next** button
2. Click the orange **Create Auto Scaling group** button

Verifying that Load Balancing is Working

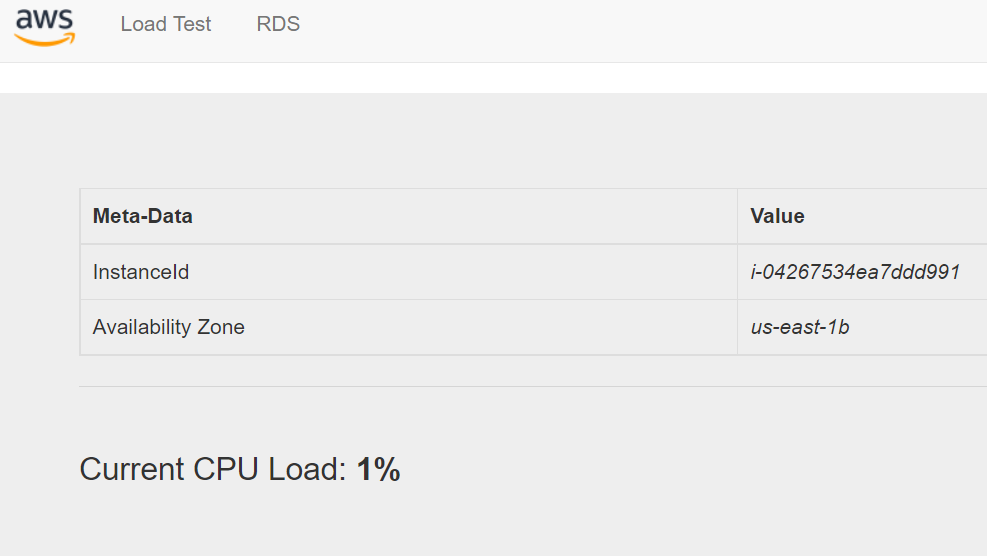
1. On the left menu, click **Instances.** There should be two new instances named **Lab Instance**
2. To confirm if they passed their Health Check select **Target Groups** in the left menu. Select the **LabGroup** box and go to the **Targets** tab. There should be two **Lab Instance** targets
3. Wait for the **Status** of both of the instances to transition to **healthy** which means that the Load Balancer will send traffic to the instance



1. On the left menu, click **Load Balancers**
2. Scroll down and copy the **DNS name**. Delete the part that says **“A Record”**

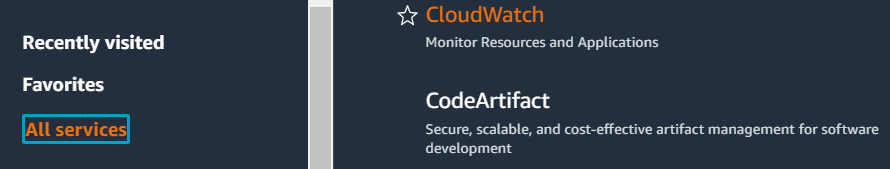


1. Paste the **DNS Name** into a new web browser. A page like this should appear. This means that the Load Balancer received the request, sent it to an EC2 instance, and passed back the result.

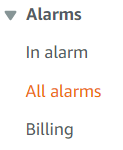


Testing Auto Scaling

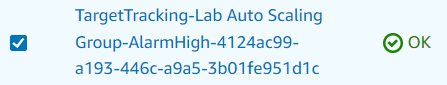
1. Click the **Services** box in the top left, then click **All Services**, and click **CloudWatch**



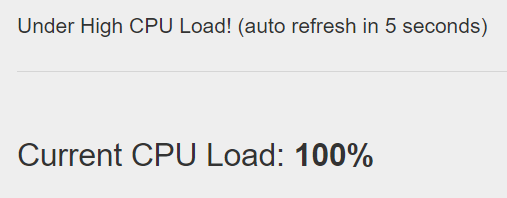
1. In the left menu, click **Alarms**, then click **All Alarms.** Two alarms should be displayed



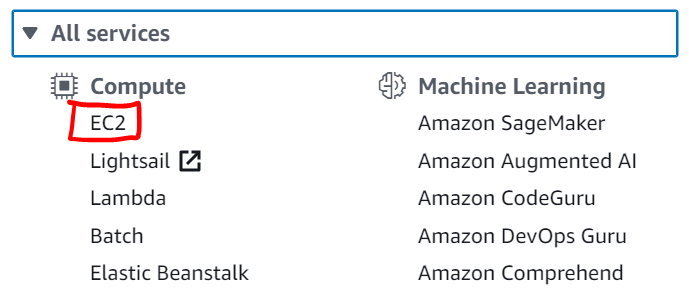
1. Click the alarm in the **OK** state. It has **AlarmHigh** in its name



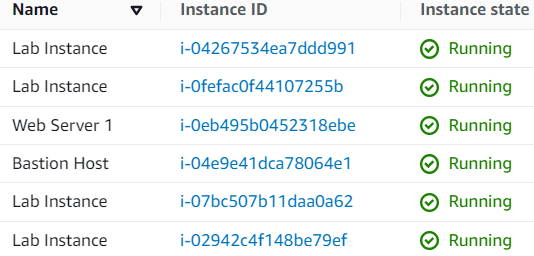
1. Go back to the web browser and click the **Load Test** button at the top of the page. This should generate high loads, making the CPU Load go to 100%

1. Go back to the **CloudWatch** console
2. Wait for the state of **AlarmLow** to change to **OK** and the state of **AlarmHigh** to change to **In alarm.**
3. Click the **Services** box, then click **All Services,** and click on **EC2**

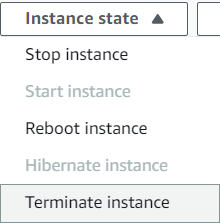


1. In the left menu, click **Instances.** There should be more than two instances labeled **Lab Instance** running now. They were created by Auto Scaling as a result of the Alarm



Terminating Web Server 1

1. Select **Web Server 1**
2. Click the **Instance state** drop-down and click **Terminate Instance**



1. Click the orange **Terminate** button

You have finished this lab!

Click the **End Lab** button and select the blue **Yes** button.



**Conclusion**

In this lab, you will create an Amazon Machine Image from a running instance, create a load balancer, create a launch configuration and an Auto Scaling group, scale new instances within a private subnet automatically, and create Amazon CloudWatch alarms and monitor performance of your infrastructure.