

[version_1.0]

©2019 Amazon Web Services, Inc. and its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited.

Errors or corrections? Contact us at <https://support.aws.amazon.com/#!/contacts/aws-training>

Exercise: ELB

*The exercises are designed to be completed in your AWS account, and **will have an associated cost**. For this reason, in addition to the written instructions, this course includes video recordings of the exercises. If you intend to attempt the exercises, familiarize yourself with [AWS pricing](#), specifically [Amazon EC2 pricing](#), [Amazon S3 pricing](#), and [Amazon DynamoDB pricing](#) and the [AWS Free Tier](#).*

For this scenario, you have been tasked with setting up an Elastic Load Balancer and an AutoScaling group, so your application can horizontally scale.

In this exercise, you will create an Application Load balancer and a launch template. Then, you will set up an AutoScaling group that uses the Load Balancer and launch template you created. Finally, you will test and stress the application and watch your application scale in real time.

Lab Steps

Stage 0 - Launch an EC2 Instance

1. Search for **EC2** in the search bar at the top. Choose **EC2**.
2. Choose **Instances** under **Instances** at the left side panel.
3. Select the `employee-directory-app-lab6` instance. Which should be in the **Stopped** state.
4. Choose **Actions**. **Image and templates** and **Launch more like this**.
5. At the top, choose **3. Configure instance**.
6. Next to **Auto-assign Public IP**, choose **Enable**.
7. At the top, choose **5. Add Tags**. Remove `lab6` at the end of the **value** and append `lab7` instead.

Example:

```
employee-directory-app-lab7
```

8. Choose **Review and Launch**. Choose **Launch**.
9. Leave the `app-key-pair` selected under **Select a key pair**. Select the acknowledgement.
10. Choose **Launch Instances**.
11. Choose **View Instances**. The instance should now show up under **Instances**. Wait for the **Instance state** to change to **Running** and the **Status check** to change to **2/2 checks passed**.
12. Next to **Name**, choose the checkbox to select the `employee-directory-app-lab6` instance. Under the **Details** tab copy down the **Public IPv4 address**.

13. Paste it into a new browser tab/window. You should see a **Employee Directory** placeholder. Right now you will not be able to interact with it as it's not currently connected to the database.

Stage 1 - Create Load Balancer

1. In the EC2 console, on the left side panel under **Load Balancing**, choose **Load Balancers**.
2. Choose **Create Load Balancer**. Choose **Create** under the **Application Load Balancer**.
3. For the **Name** paste in `app-elb`. Under **Availability Zones** for the **VPC**, choose the `app-vpc`. Select both **Availability Zones**.
Example: If you are in US West (Oregon) you would choose both **us-west-2a** and **us-west-2b**.
4. Choose **Select a subnet** and choose `Public Subnet 1` for the first AZ and `Public Subnet 2` for the second AZ.
5. Choose **Next: Configure Security Groups**.
6. Again choose **Next: Configure Security Groups**. Next to **Assign a security group** choose **Create a new security group**. Next to **Security group name** paste in `load-balancer-sg`.
7. Remove the existing rule by clicking on the **X** at the right.
8. Choose **Add Rule**. For the **Type** choose **HTTP**. For **Source** choose **Anywhere**.
9. Choose **Next: Configure Routing**.
10. Under **Target group** next to **Name** paste in `app-target-group`.
11. Under **Health checks** expand **Advanced health check settings**.
12. Change the **Healthy threshold** to `2`. Change the **Unhealthy threshold** to `5`.
13. Change the **Timeout** to `30`. Change the **Interval** to `40`. Choose **Next: Register Targets**.
14. Under **Instances** select the checkbox next to the `employee-directory-app-lab7` instance and choose **Add to registered**.
15. Choose **Next: Review**. Choose **Create**. Once it's successfully created choose **Close**.
Note: Wait for the **State** to change from provisioning to **active**. Refresh the page while you wait.
16. Select the `app-elb` and copy the **DNS** name and paste it into a notepad. Edit the DNS name to include an `http://` in front of the URL, to look like the following:
`http://app-elb-0000000000.us-west-2.elb.amazonaws.com`
17. Copy the DNS name with `http://` and paste it into a new browser tab or window. You should see the employee directory application.

Stage 2 - Create launch template

Now that you can access your application from a singular DNS name, it's time to horizontally scale! To do this, the first thing you need is a launch template.

1. Back in the AWS Management Console, search for **EC2** in the service search bar. Then, choose EC2. On the left side panel, choose **Launch Templates** under the **Instances** header.
2. Choose **Create launch template**. Under **Launch template name** paste in `app-launch-template`.

3. Under **Template version description** paste in `A web server for the employee directory application`.
4. Under **Auto Scaling guidance** check the box for **Provide guidance to help me set up a template that I can use with EC2 Auto Scaling**.
5. Under **AMI** choose the **Amazon Linux 2 AMI (HVM), SSD Volume Type**.
6. Under **Instance type** choose the **t2.micro**.
7. Under **Key pair name** choose the `app-key-pair`.
8. Under **Network settings** and **Security group** choose the `web-security-group`.
9. Expand **Advanced details** and under **IAM instance profile**. Choose `S3DynamoDBFullAccessRole`.
10. Scroll down to **User data** and paste in the following information:

```
#!/bin/bash -ex
wget https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/DEV-AWS-MO-
GCNv2/FlaskApp.zip
unzip FlaskApp.zip
cd FlaskApp/
yum -y install python3 mysql
pip3 install -r requirements.txt
amazon-linux-extras install epel
yum -y install stress
export PHOTOS_BUCKET=${SUB_PHOTOS_BUCKET}
export AWS_DEFAULT_REGION=<INSERT REGION HERE>
export DYNAMO_MODE=on
FLASK_APP=application.py /usr/local/bin/flask run --host=0.0.0.0 --port=80
```

Change the following line to match your region:

Note: You can find this at the top right next to your user name.

```
export AWS_DEFAULT_REGION=<INSERT REGION HERE>
```

Example:

Note: US West (Oregon)

```
export AWS_DEFAULT_REGION=us-west-2
```

Also change the bucket variable to match your bucket:

```
export PHOTOS_BUCKET=<INSERT_BUCKET_NAME>
```

Example:

```
export PHOTOS_BUCKET=employee-photo-bucket-al-007
```

11. Finally choose **Create launch template**.

Stage 3 - Create ASG

1. Choose **View Launch templates**. On the left side panel, under the **Auto Scaling** header, choose **Auto Scaling Groups**.
2. Choose **Create Auto Scaling group**. Under **Auto Scaling group name** paste in `app-asg`.
3. Under **Launch template** choose the `app-launch-template`. Choose **Next**.
4. Under **Network and VPC** choose the `app-vpc`. Choose `Public Subnet 1` and `Public Subnet 2`. Choose **Next**.
5. Under **Load balancing** choose **Attach to an existing load balancer**. Under **Attach to an existing load balancer** choose **Choose from you load balancer target groups**.
6. Under **Existing load balancer target groups** choose **Select target groups**. Choose the `app-target-group`.
7. Under **Health checks** choose **ELB**. Choose **Next**.
8. Under **Group size** change the **Desired capacity** to `2`.
9. Change the **Minimum capacity** to `2`.
10. Change the **Maximum capacity** to `4`.
11. Under **Scaling policies** choose **Target tracking scaling policy**.
12. Change the **Target value** to `60`. Change the **Instances need to** `300`. Choose **Next**.
13. Under **Add notifications** choose **Add notification**.
14. Under **SNS Topic** choose **Create a topic**. Under **Send a notification to:** paste in `app-sns-topic`. Under **With these recipients** paste in your email address. Choose **Next**.
15. Choose **Next**. Choose **Create Auto Scaling group**.
16. You should have received an **AWS Notification - Subscription Confirmation**. In the email, choose **Confirm subscription**.

Stage 4 - Test the application

1. Under **Load Balancing** on the left side panel, choose **Target Groups**. Select the name of the `app-target-group`. Choose the **Targets** tab. You should see two additional instances launching. Wait until the **Status** shows as healthy.
2. Under **Load Balancing** on the left side panel, select **Load Balancers** and select the `app-elb`. Again, copy the **DNS** name and append `http://` to the beginning of it.
3. Paste it into a new browser tab or window. Append `/info` to the URL.

Example:

```
http://app-elb-0000000000.us-west-2.elb.amazonaws.com/info
```

You should be presented with **Instance Info** which will show you which **instance_id** and **availability_zone** you are being routed to. Refresh the page a few times and you will notice you are being routed to a different **instance_id** and **availability_zone**.

4. Now, you will need to stress the CPU of the instance in order to test Auto Scaling.
5. Next to **Stress cpu:** choose **10 min**. At the top you will notice it now says **Stressing CPU**.
Note: Go grab a beverage as you wait :)
6. After the 10 minutes are up, go back in the EC2 Console. Under **Load Balancing** on the left side panel, choose **Target Groups**. Choose the `app-target-group`. Choose the **Targets** tab.
7. You should see additional instances being spun up due to the stress test. You should also see a notification email as well.

Stage 5 - Delete Course Resources

1. To ensure that you don't incur additional costs, please ensure that you delete all resources in the AWS account you created.
2. In the EC2 console, on the left side panel, choose **Auto Scaling groups**. Select the app-asg and select delete. Type **delete** in the text box to confirm your deletion.
3. In the EC2 console, on the left side panel, choose **Load Balancers**. Select the `app-elb` and select **Actions** and then **delete**. Choose **Yes, Delete**.
4. In the EC2 console, on the left side panel, choose Target Groups. Select the `app-target-group` and select **Actions** and then **delete**. Choose **Yes, Delete**.
5. In the EC2 console, on the left side panel, choose Instances. Terminate all EC2 instances you created in this course. Select the checkbox next to the instances you want to delete (all instances starting with `employee-directory-app`) and then select **Instance State**. Then, choose **Terminate Instance**.
6. In the DynamoDB console, on the left side panel, click on **Tables**. Then, click the circle next to the `Employees` table. Choose **Delete Table**. Type in the word `delete` and select **Delete**.
7. In the S3 console, find the employee-photo-bucket. Select the circle next to the bucket you created. Then, choose **Empty**. Type in `permanently delete` in the text field. Select **Empty**. Choose **Exit**. Now, choose **Delete**. Type in the name of your bucket in the text field and choose **Delete bucket**.
8. In the VPC console, on the left side panel, click on Route Tables. Select the checkbox next to **app-route-table-public** and click on the **Subnet Associations** tab. Select **Edit subnet associations**. Deselect the boxes next to both Public Subnets and click **save**. Select **Actions** and then **Delete**. Then, choose Delete Route Table again. Follow the same steps for **app-route-table-private**.
9. In the VPC console, on the left side panel, click on **Internet Gateways**. Select the app-igw and choose **Detach from VPC** and choose **Detach Internet Gateway**. Then, click **Actions** and select **Delete Internet Gateway**. Type in delete in the text field and select **Delete Internet Gateway**.
10. In the VPC console, on the left side panel, select **Subnets**. Select the boxes next to Public Subnet 1, Public Subnet 2, Private Subnet 1, and Private Subnet 2. Select **Actions** and then **delete subnet**. Type `delete` in the text field. Then, choose **Delete**.
11. In the VPC console, on the left side panel, select **Your VPCs**. Select the checkbox next to app-vpc. Select **Actions** and then **Delete VPC**. Type `delete` in the text field and then choose **Delete**.

Lab Complete

Congratulations! You have completed the lab.

For feedback, suggestions, or corrections, please contact us at: <https://support.aws.amazon.com/#/contacts/aws-training>