Introduction

Performing end-to-end tests to make sure everything is working as you think it should is very important. Although this may be an automated procedure, often a quick sanity test by other individuals and/or groups directly from the AWS Console is also helpful. This lab step will point out a few ways to test that your Launch Configuration is working in conjunction with the Auto Scaling group and CloudWatch Alarm (which uses AWS Simple Notification Service (SNS)).

Instructions

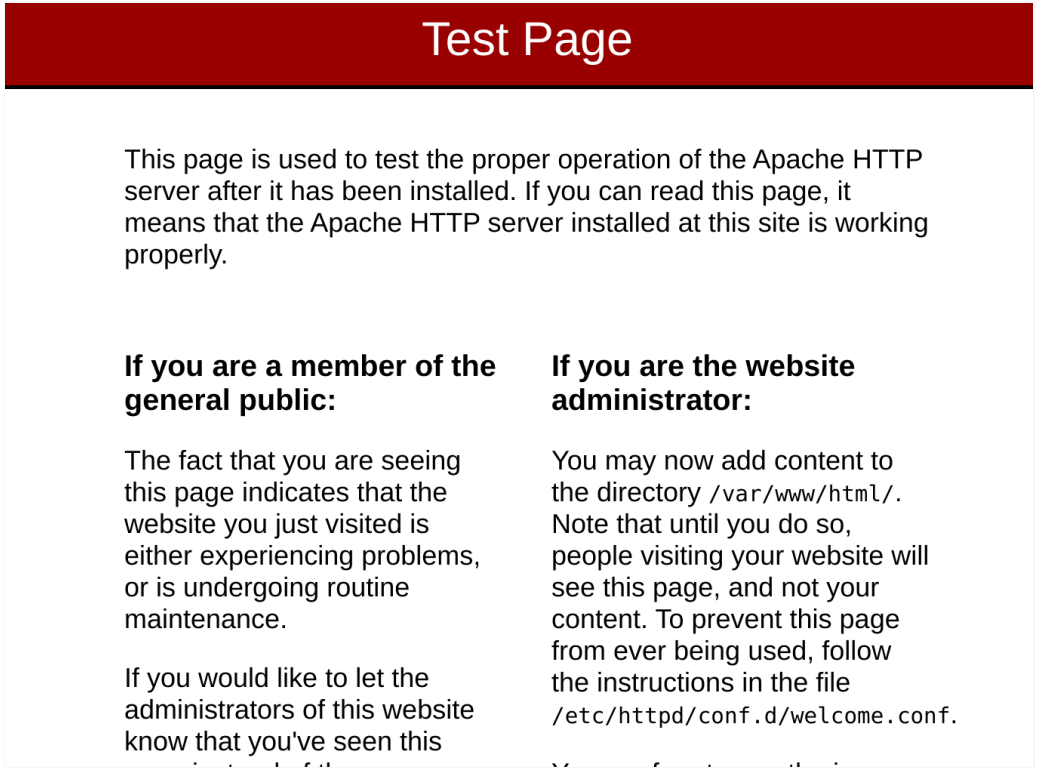
1. In the left-hand menu, under **Load Balancing**, click **Load Balancers**:

You will see a list of load balancers with one item selected, the load balancer you created previously.

2. In the **Description** tab, look for the **DNS name** field, and copy the value (you can click the copy button at the end of the field).

3. Open a new browser tab and paste the DNS name you just copied into the address bar.

You will see the Apache webserver test page that is served by default upon a fresh unconfigured web server installation:



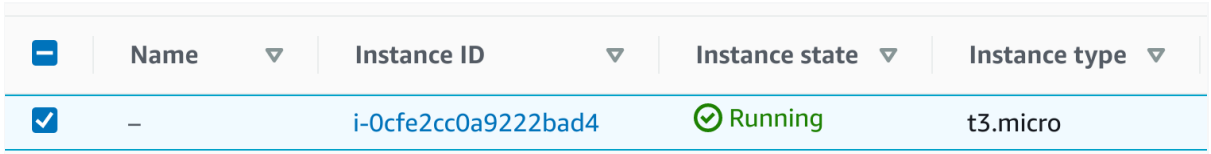
If the page is not displayed, there are several places you can check to troubleshoot the issue starting with the following:

- Ensure the security groups of the load balancer and the instances allows HTTP ingress traffic
- Ensure the user data script in the launch template correctly installs and runs the Apache webserver
- Ensure the Auto Scaling group is configured to add its instances to the load balancer's target group
- Ensure the health checks are configured for TCP port 80 otherwise, the instances will never reach a healthy status and will be terminated and then replaced with a new instance by the Auto Scaling group. The new instance will subsequently never reach a healthy status and be replaced, and the process repeats.
    - To allow for you to debug the instances without having them be replaced, you can block instance termination by performing the following steps:
        - Navigate to **Auto Scaling Groups (and check it)** > **Edit**
        - Set the **Suspended Processes** to **Terminate** (This will prevent instances in your group from getting terminated. Don't forget to remove the configuration once the issue is resolved.)

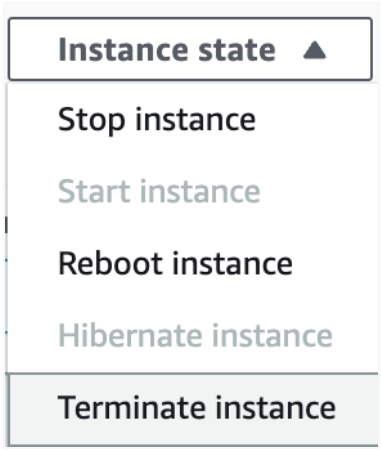
4. To list EC2 instances, in the left-hand menu, click **Instances**:

You will see a list of instances with one running instance, this instance is created by your auto-scaling group.

5. To select your instance, click the check-box on the left-hand side:

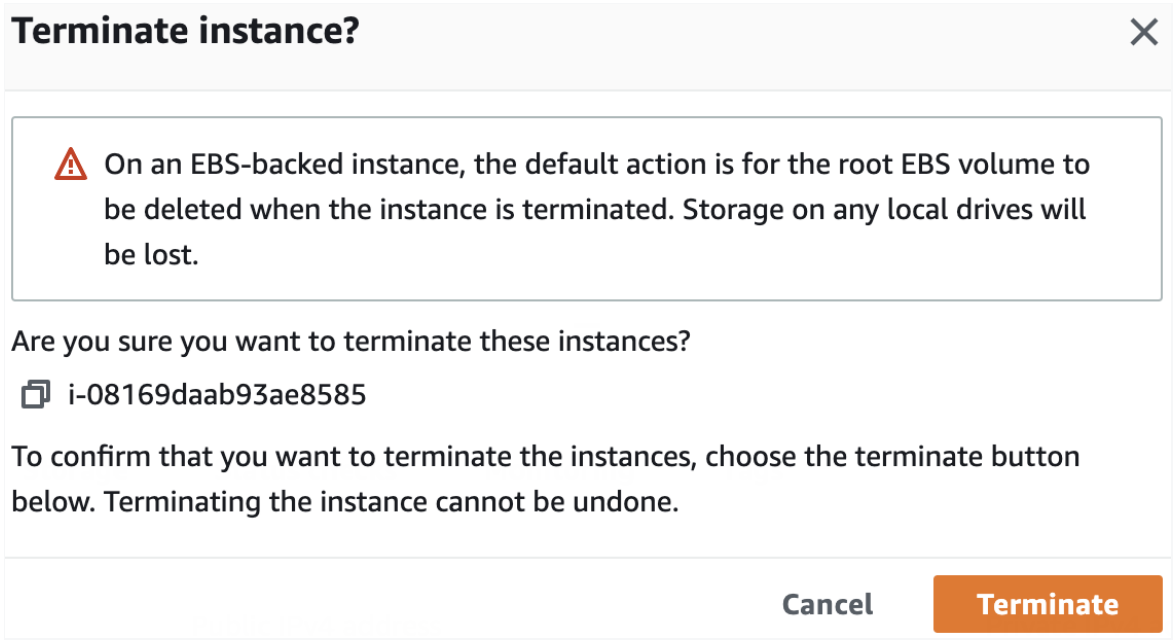


6. To terminate the running instance, click **Instance state** > **Terminate instance**



The note on EBS-backed instances will appear. You can ignore this warning.

7. In the **Terminate instance?** form that appears, click **Terminate:**



In the instances list, you will see the **Instance state** change from **Running** to **Shutting-down** and then **Terminated**.

You have terminated the running instance.

The Auto Scaling group will detect this change and relaunch an instance automatically to meet the minimum desired capacity of one.

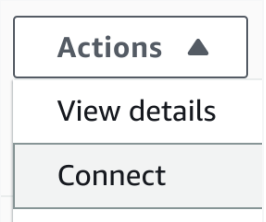
8. Wait two or three minutes and click the refresh icon:

You will see the new instance launch and settle into a running state:

Name	Instance ID	Instance state	Instance type	Status check
–	i-07bfece2cb72063bc	Terminated	t2.micro	–
–	i-07314fa1ddd7ff708	Running	t2.micro	–

9. Connect to the running instance using the PEM (macOS/Linux) or PPK (Windows) key file in the **Your lab data** of this Lab.

Tip: To remind yourself how to connect, select the running instance, click **Actions**, and click **Connect**:



Cloud Academy recommends connecting using EC2 Instance Connect. You can also connect using an SSH client and PEM file in the credentials section of this lab. Connecting using session manager is not supported in this lab.

10. Enter the following command at the command line to run `stress` causing the CPU utilization to increase for five minutes:

Copy code

```
stress --cpu 2 --io 1 --vm 1 --vm-bytes 128M --timeout 5m
```

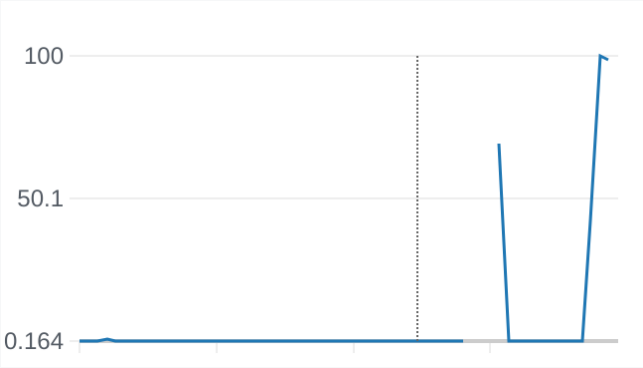
```
stress: info: [3933] dispatching hogs: 2 cpu, 1 io, 1 vm, 0 hdd
```

Enter `man stress` for more information about `stress`.

11. Navigate to the Auto Scaling group's Monitoring tab in the EC2 Console and click the **EC2** tab.

Wait a minute or two and click the refresh button.

You will see the CPU utilization metric increase to near one hundred percent:



The auto-scaling group will detect the increase in utilization and launch a second instance in response.

Note that it will take a few minutes for the metrics and graph to update.

12. To see auto-scaling events, click the **Activity** tab:

Look at the **Activity history** section.

The exact events you will see may differ depending on how long the auto-scaling group and stress command has been running:

WaitingForInstanceWarmup	Launching a new EC2 instance: i-06b0d8245ca4bcc99
Successful	Launching a new EC2 instance: i-077546ed485311f84
Successful	Terminating EC2 instance: i-08169daab93ae8585

You will see events detailing your earlier deletion of the initial instance.

If you don't see a new instance launching wait a minute or two and click the refresh button.

Eventually, because the stress command stops after five minutes, the number of instances will drop down to the minimum of one.

13. Navigate to the Instances section of the EC2 Console.

You will see two running instances, or if your auto-scaling group has already initiated a scale-in (because the stress command finished and CPU utilization dropped), you will see two terminated instances including the one you manually terminated, and one running instance.

Summary

In this lab step, you performed several tests of the Auto Scaling group, launch template, and Network Load Balancer system. You also learned where to look when something doesn't work as expected with Auto Scaling groups.