Linux Academy

Hands-on Lab

Challenge Lab: Troubleshooting Connectivity Issues #3

Contents

Lab Connection Information

- Labs may take up to five minutes to build
- Access to an AWS Console is provided on the Hands-on Lab page, along with your login credentials
- Ensure you are using the N. Virginia region
- Labs will automatically end once the alloted amount of time finishes

Related Courses

AWS Certified Solutions Architect - Associate

Related Videos

ELB Troubleshooting Scenarios

Need Help?

Linux Academy Community

... and you can always send in a support ticket on our website to talk to an instructor!

For this challenge lab, we provide you with a broken environment that is in need of fixing. Attempt to troubleshoot the environment on your own, after reading the parameters in the scenario, then compare your answer with the solution.

The Challenge Scenario

Log in to the AWS Console using the credentials provided on the Hands-on Lab page.

In this challenge lab, you need to fix the connectivity issue within the AWS environment to allow you to view the Apache test located on our EC2 instances via the Elastic Load Balancer.

The lab provides two EC2 instances, *Web 1* and *Web 2*. If we were to copy the public IP of either of these instances into our browser, we would see the Apache test page; however, if we were to use the Elastic Load Balancer DNS name (which can be found on the ELB Dashboard), this would fail. This is what must be fixed.

The Solution

We first want to consider how our environment should be set up. We have a VPC with an internet gateway, two public subnets with two EC2 instances (one in each), and an Elastic Load Balancer to distribute traffic between them.

Because we where able to access the Apache test page for both EC2 instances, we know that instances themselves are configured properly. Since the error was in relation to accessing the servers through the Elastic Load Balancer, we know that the problem most likely lies in the configuration of the ELB.

Go to the **EC2 Dashboard** and select **Load Balancers**. Select the available load balancer, then view the associated security group, under the **Description** tab. We want to make sure access to port 80 is available. Click on the security group, then click of the **Inbound Rules** tab. As we can see, we only have an inbound rule for SSH, not HTTP. Press **Edit**, then change the *SSH* rule to *HTTP*, because we do not need SSH access to our load balancer. **Save**.

Copy the ELB's DNS name and try to view it in your browser again. It still fails.

Return to the **Load Balancers** page and select the load balancer. View the **Instances** tab. Our two instances are listed with a status of *OutOfService*. This could be due to an issue in configuration or because the health checks have yet to be run. If we view the **Health Checks** tab, we can see that these checks are run at short intervals; however, an issue lies with the health check itself. The **Ping Target** is set to 8000, which is not a port to which we can ping; instead, this needs to be set to *80*. To fix this, press **Edit Health Check** and adjust the **Ping Port**. **Save**.

Return to the **Instances** tab. The instances are already listed as InService.

Try to access the ELB over your browser using the DNS name again. This time, it works! This lab is now complete.