Troubleshooting a Broken Kubernetes Cluster

00:54:44 Exit Lab Complete Lab

1 hour duration Practitioner Rate this lab

VIDEOS GUIDE

Troubleshooting a Broken Kubernetes Cluster

Introduction

Every Kubernetes administrator will likely need to deal with a broken cluster at some point, whether a single node or the entire cluster is down. In this lab, you will be able to practice your troubleshooting skills. You will be presented with a broken Kubernetes cluster and asked to use your investigative skills to identify the problem and fix it.

Solution

Log in to the control plane node server using the credentials provided:

```
ssh cloud_user@<PUBLIC_IP_ADDRESS>
```

Determine What is Wrong with the Cluster

1. Find out which node is having a problem by using `kubectl get nodes`. Identify if a node is in the *NotReady* state.
2. Get more information on the node by using `kubectl describe node <NODE_NAME>.`
3. Look for the *Conditions* section of the *Node Information* and find out what is affecting the node's status, causing it to fail.
4. Log in to the worker 2 node server using the credentials provided:

   ```
   ssh cloud_user@<PUBLIC_IP_ADDRESS>
   ```

5. Look at the kubelet logs of the worker 2 node by using `sudo journalctl -u kubelet`.

6. Go to the end of the log by pressing `Shift + G` and see the error messages stating that kubelet has stopped.

7. Look at the status of the kubelet status by using `sudo systemctl status kubelet`, and note whether the kubelet service is running or not.

Fix the Problem

1. In order to fix the problem, we need to not only start the server but also enable kubelet to ensure that it continues to work if the server restarts in the future. Use `clear` to clear the service status, and then start and enable kubelet by using `sudo systemctl enable kubelet`, followed by `sudo systemctl start kubelet`.

2. Check if kubelet is active by using `sudo systemctl status kubelet`, and note if the service is listed as *active (running)*.

3. Return to the control plane node.

4. Check if all nodes are now in a *Ready* status by using `kubectl get nodes`. You may have to wait and try the command a few times before all nodes appear as *Ready*.

Conclusion

Tools

Lab Diagram | Instant Terminal

Credentials

How do I connect?

Cloud Server Control Plane Node

Username

cloud_user

Password

dI*4)O&g

Control Plane Node Private IP

10.0.1.101

Control Plane Node Public IP

3.81.55.161

Launch Instant Terminal

How do I connect?

Additional Resources

Your company, BeeBox, has a new Kubernetes cluster that was just built by an outside contractor. Last night, someone restarted the servers used to run this cluster. Ever since the restart, some of your team members are reporting issues with one of the worker nodes. Unfortunately, the contractor is no longer working with the company, so you will need to find and fix the problem.

Explore the cluster and determine what is causing the issues. Then take steps to fix the problem and ensure that it does not happen again.

Learning Objectives

0 of 2 completed

- [ ] **Determine What is Wrong with the Cluster**
- [ ] **Fix the Problem**