



Sandhya Babu <sandhya.bca7@gmail.com>

Day19 Challenge: Cluster Detective: Solving Kubernetes Mysteries

1 message

Sagar Utekar <getfitwithsagar2366@gmail.com>
Bcc: sandhya.bca7@gmail.com

Thu, Dec 19, 2024 at 8:00 AM

Hello Learners,

Welcome back to another thrilling episode of the **DevOps SRE Daily Challenge!** 🎉

Today, you'll step into the shoes of a **Kubernetes detective**, solving mysteries around cluster configurations, component communication, and troubleshooting. Your mission? To uncover the truth behind the Kubernetes architecture, track down elusive component configurations, and crack the case of the missing logs and ports!

What You'll Learn:

1. How to investigate the underlying mysteries of Kubernetes, from the control plane to worker node components.
2. How to decode the logs of different Kubernetes components using powerful tools like `kubectl` and `crictl`.
3. How to uncover the ports used for component communication - knowing where traffic flows is key to solving Kubernetes mysteries!

Challenge Tasks

The Kubernetes Detective Theory Challenge:

To solve the case, you must gather all the facts. Here's your first clue:

1. What is Kubernetes?

- Define Kubernetes and explain its purpose in the world of container orchestration.
- Highlight the problems Kubernetes solves (e.g., scaling, fault tolerance, declarative management).

2. Features of Kubernetes:

- List and briefly explain features like **scalability, self-healing, service discovery, storage orchestration, automated rollouts/rollbacks**, etc.

3. Kubernetes Architecture:

- Illustrate the architecture with two main sections:
 - **Control Plane Components:** API Server, etcd, Scheduler, Controller Manager.
 - **Worker Node Components:** Kubelet, Kube Proxy, Container Runtime.
- Explain the role of each component in maintaining cluster functionality.

4. Configuration Files and Logs:

- Discuss where configuration files for these components are located and why they're essential for debugging.

5. Ports Used:

- List the ports used by control plane and worker node components for communication (e.g., API server port 6443).

Practical: Solving the Cluster Mystery

Your investigation continues with Minikube — a single-node Kubernetes setup and for a multi-node Kubernetes setup using Kubeadm

Here's what to do:

- Use your existing clusters from the previous challenge.
- Access and review the configurations of the following components:
 - **API Server, Scheduler, Controller Manager, etcd**
 - **Kubelet, Kube Proxy, Container runtime**
- Check the configuration files in:

```
/etc/kubernetes/manifests/  
/var/lib/kubelet/config.yaml
```

- Retrieve logs for the components

- Identify the ports used by the components for communication:
- Use `kubectl` to inspect **pod logs**:
- Use `crictl` for **container management**:
 - List all containers
 - Inspect details of a specific container
 - Check logs of a specific container

Bonus Task

- Use the command `kubectl cluster-info dump` in both Minikube and Kubeadm setups.
- Share insights on the diagnostic information retrieved, including any unusual configurations or observations.

Submission Guidelines

- Submit the following:
 - **Explanation of Kubernetes fundamentals** based on the theory section.
 - **Screenshots of component configurations and logs** from both Minikube and Kubeadm setups.
 - **Port mapping** and explanations.
 - **Logs** from `kubectl` and `crictl` commands.
 - **The cluster detective's report**: Any insights you found while investigating the clusters.
- Post your progress with the hashtags: **#ClusterDetective**, **#SRELife**, **#DevOpsForAll**, **#ckawithsagar**

Resources to Help You:

- [Understanding Kubernetes Architecture \(YouTube\)](#)
- [Kubernetes Architecture and Components \(GitHub\)](#)

If you missed any previous challenges, you can catch up by reviewing the problem statements on [GitHub](#).

Best regards,
Sagar Utekar