CS-695 Assignment 4 (Mini Project)

Title : FaaS + K8s = faast!

A Report Submitted by

Ajith Pasuvula (22m0821)

Atluri Sai Kumar Reddy (22m0745)

Academic Year (2023-2024)

Implementing FaaS Platform with Kubernetes Integration

Introduction:

The aim of this project, titled "FaaS + K8s = faast!", was to develop a Function as a Service (FaaS) platform as a wrapper around Kubernetes (K8s). The project focused on building a system capable of function registration, trigger registration, trigger dispatch, and supporting metrics within a Kubernetes environment. The primary objective was to demonstrate the correctness and feature set of the FaaS platform while also conducting performance benchmarking.

Implementation Overview:

The implementation of the FaaS platform involved several key steps:

- **Server Creation with Node.js:**
  - A server was created using Node.js to handle the orchestration of function deployment and execution.

- **Bash Script for Compression and Deployment:**
  - A Bash script was developed to compress files from a specified directory and deploy them to a Kubernetes cluster.
  - The script is responsible for compressing all files within the home directory and sending the compressed file to a container running in a Kubernetes cluster via Minikube.

- **Container Deployment Workflow:**
    - Upon execution of the Bash script, the compressed file is sent to a designated container within the Kubernetes cluster.
    - The container then decompresses the file, downloads any necessary dependencies, and creates a new container to execute the function code.
    - The initial container serves as the function registration entity, while the second container executes the registered function.

- **Function Invocation and Response:**
    - Once the second container successfully creates and runs the function, the registration container responds with a success message and provides a link to access the newly created container running the function.
    - Function invocation can be done using the provided link, enabling users to execute their functions within the Kubernetes environment.

What is MiniKube ?

Minikube is a tool that enables you to run Kubernetes locally on your machine. It's designed to simplify the setup and operation of Kubernetes clusters for development and testing purposes. Minikube runs a single-node Kubernetes cluster inside a virtual machine (VM) on your local system, allowing you to experiment with Kubernetes features and deploy containerized applications without needing a full-scale production environment.

Minikube provides a lightweight and easy-to-use solution for developers who want to learn Kubernetes concepts, develop Kubernetes-based applications, or test their applications in a Kubernetes-like environment before deploying them to a production Kubernetes cluster. It abstracts away much of the complexity of setting up and managing a Kubernetes cluster, making it accessible to developers with varying levels of Kubernetes expertise.

Conclusion:

In conclusion, the project successfully implemented a FaaS platform integrated with Kubernetes, allowing for the deployment and execution of functions within a containerized environment. The developed system demonstrated functionality for function registration, deployment, and invocation, showcasing the potential of combining FaaS and Kubernetes technologies for scalable and efficient application deployment.

Future Work:

Future iterations of the project could focus on enhancing security measures, improving scalability, and optimizing performance for large-scale deployments. Additionally, integrating monitoring and logging functionalities could provide valuable insights into function execution and resource utilization within the Kubernetes cluster.