Technologies Used

In the process of deploying a Java HTTP server to an Azure Kubernetes Service (AKS) cluster using Terraform, you will have the opportunity to work with several DevOps technologies and tools. Here's an overview of the technologies and tools you are likely to learn and use in this project:

1. **Version Control System (VCS) - Git**: Git is a distributed version control system used to track changes in code and collaborate with a team. You will use Git to manage the source code of your Java HTTP server, including creating branches, committing changes, and collaborating with other team members.
2. **Java Development Kit (JDK)**: The Java Development Kit is a software development environment that includes the tools needed to compile, debug, and run Java applications. You will need JDK to compile the `HelloWorldHttpServer.java` source code into class files.
3. **Docker**: Docker is a platform that enables developers to package applications and their dependencies into containers, ensuring consistency across different environments. You will use Docker to create a container image for your Java HTTP server, making it easily deployable and portable.
4. **Dockerfile**: The Dockerfile is a script that defines the instructions to build a Docker image. You will write a Dockerfile to specify the base image, copy the compiled Java class files, and set up the necessary environment for running the Java HTTP server in a Docker container.
5. **Container Registry**: A container registry is a repository that stores Docker images. You will choose a container registry (e.g., Docker Hub, Azure Container Registry) to push the Docker image of your Java HTTP server.
6. **Terraform**: Terraform is an infrastructure-as-code tool that allows you to define and manage cloud resources using configuration files. You will use Terraform to define the infrastructure for the AKS cluster, including the desired node pool size, networking, and Kubernetes version.
7. **Azure Kubernetes Service (AKS)**: AKS is a managed Kubernetes service provided by Microsoft Azure. You will use AKS to create a Kubernetes cluster to host and manage the Docker containers.
8. **Kubernetes**: Kubernetes is an open-source container orchestration platform for automating the deployment, scaling, and management of containerized applications. You will interact with Kubernetes using `kubectl` to deploy your Java HTTP server as a Kubernetes Pod or Deployment.
9. **Infrastructure as Code (IaC)**: IaC is an approach where infrastructure is managed and provisioned using code and version control. You will use Terraform as an IaC tool to define the AKS cluster infrastructure.
10. **Networking and Load Balancing**: You will configure networking for the AKS cluster and set up load balancing to expose the Java HTTP server to the internet.

Through this project, you will gain hands-on experience in working with these DevOps technologies and tools, empowering you to deploy applications in a scalable, automated, and efficient manner.

Revision #2
Created 27 July 2023 10:01:05 by Gaurav Pandey
Updated 27 July 2023 10:03:08 by Gaurav Pandey