Run a Stateless Application Using a Deployment

You need to have a Kubernetes cluster, and the kubectl command-line tool must be configured to communicate with your cluster.

The GoldenAMI has already been configured to communicate the Kubernetes Cluster. You can confirm the same by login into your Instance terminal and running the below command.

$ kubectl config view *# Show Merged kubeconfig settings.*

1. Deploy an application (ex. nginx) on the kubernetes cluster

   $ kubectl run <your-app-name> --replicas=2 --image=nginx:latest --port=80

The preceding command creates a Deployment object and an associated ReplicaSet object. The ReplicaSet has Two Pods, each of which runs the **nginx** application.

2. Display information about the Deployment:

   $ kubectl get deployments <your-app-name>

   $ kubectl describe deployments <your-app-name>

3. Display information about your ReplicaSet objects:

   $ kubectl get replicasets <your-app-name>

   $ kubectl describe replicasets <your-app-name>

4. Create a **Service object** that exposes the deployment:

   $ kubectl expose deployment <your-app-name> --type=LoadBalancer --name=<your-service-name>

   Note : The --name=<your-service-name> can be set as desired.

5. Display information about the Service:

   $ kubectl get services <your-service-name>

6. Display detailed information about the Service:

   $ kubectl describe services <my-service>

7. Use the external IP address (LoadBalancer Ingress) to access the Hello World application:

   http://< load-balancer-endpoint >:<port>

   where <external-ip> is the external IP address (LoadBalancer Ingress) of your Service, and <port> is the value of Port in your Service description.

Note : Please wait for 5 minutes once the app is exposed externally before trying to access from the load-balancer (endpoint)

Cleaning up

To delete the Service, enter this command:

$ kubectl delete services my-service

To delete the Deployment, the ReplicaSet, and the Pods that are running the Hello World application, enter this command:

$ kubectl delete deployment <app-name>