SSH to the **haproxyVM on Azure or any of the master NODE** of the 3 AWS Clusters

## **1. Objectives**

* Create a PersistentVolume referencing a disk in your environment.
* Create a MySQL Deployment.
* Expose MySQL to other pods in the cluster at a known DNS name.

SSH to the haproxy VM and run the below commands.

## **2. Deploy MySQL**

|  |
| --- |
| $ sudo su  # vim mysql-deployment.yaml |

Paste the below script

|  |
| --- |
| apiVersion: v1 kind: Service metadata:  name: mysql spec:  ports:  - port: 3306  selector:  app: mysql  clusterIP: None --- apiVersion: apps/v1 # for versions before 1.9.0 use apps/v1beta2 kind: Deployment metadata:  name: mysql spec:  selector:  matchLabels:  app: mysql  strategy:  type: Recreate  template:  metadata:  labels:  app: mysql  spec:  containers:  - image: mysql:5.6  name: mysql  env:  # Use secret in real usage  - name: MYSQL\_ROOT\_PASSWORD  value: password  ports:  - containerPort: 3306  name: mysql  volumeMounts:  - name: mysql-persistent-storage  mountPath: /var/lib/mysql  volumes:  - name: mysql-persistent-storage  persistentVolumeClaim:  claimName: mysql-pv-claim |

|  |
| --- |
| # vim mysql-pv.yaml |

Paste the below script

|  |
| --- |
| kind: PersistentVolume apiVersion: v1 metadata:  name: mysql-pv-volume  labels:  type: local spec:  storageClassName: manual  capacity:  storage: 20Gi  accessModes:  - ReadWriteOnce  hostPath:  path: "/mnt/data" --- apiVersion: v1 kind: PersistentVolumeClaim metadata:  name: mysql-pv-claim spec:  storageClassName: manual  accessModes:  - ReadWriteOnce  resources:  requests:  storage: 20Gi |

1. **Deploy the PV and PVC of the YAML file:**

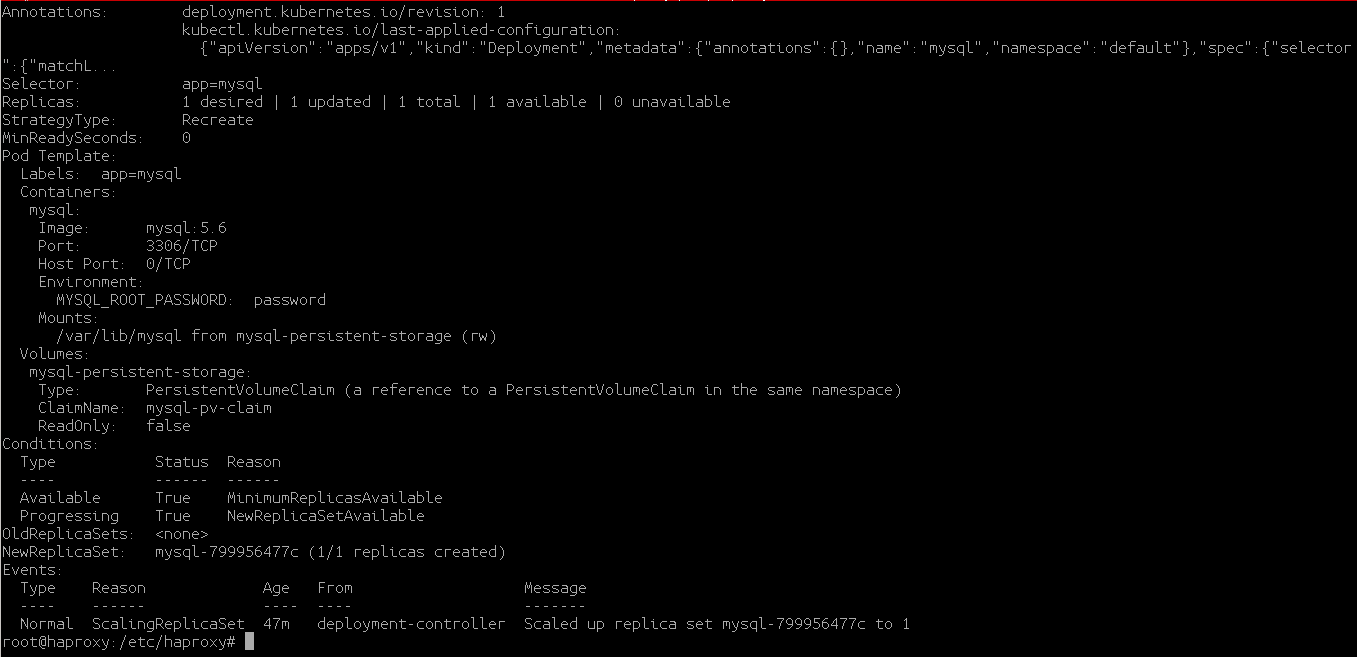
|  |
| --- |
| # kubectl apply -f mysql-pv.yaml |

**2. Deploy the contents of the YAML file:**

|  |
| --- |
| # kubectl apply -f https://k8s.io/examples/application/mysql/mysql-deployment.yaml |

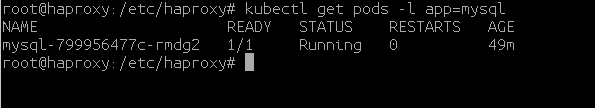
**3. Display information about the Deployment:**

|  |
| --- |
| # kubectl describe deployment mysql |

****

**4. List the pods created by the Deployment:**

|  |
| --- |
| # kubectl get pods -l app=mysql |



**5. Inspect the PersistentVolumeClaim:**

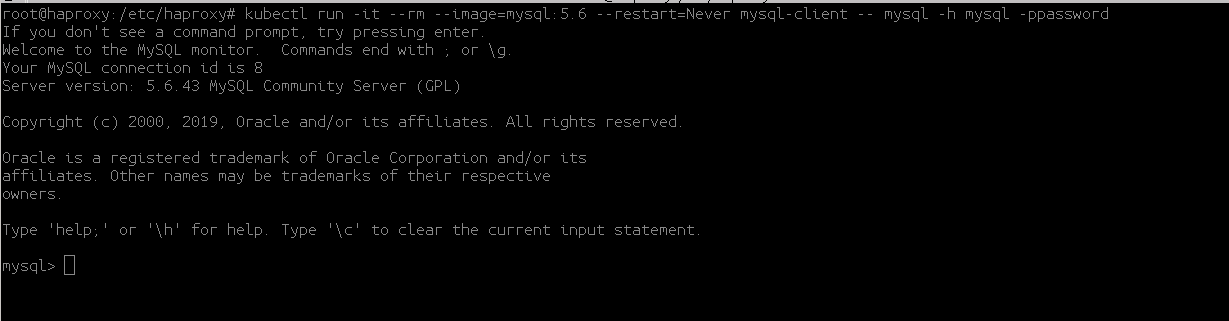
|  |
| --- |
| kubectl describe pvc mysql-pv-claim |



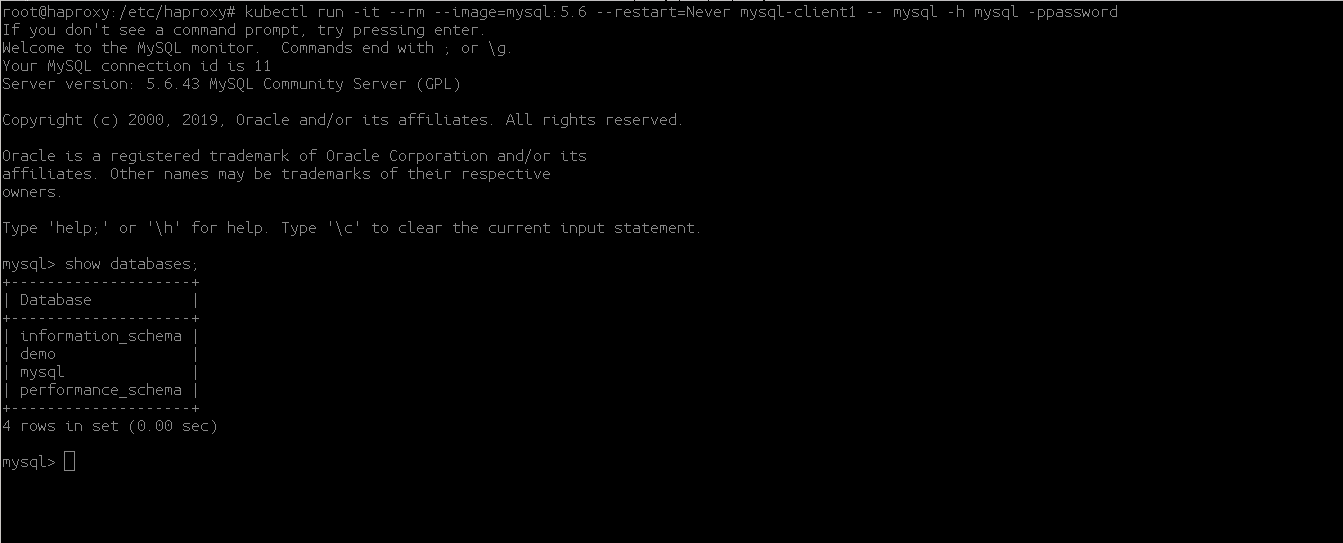
**Run a MySQL client to connect to the server:**

|  |
| --- |
| # kubectl run -it --rm --image=mysql:5.6 --restart=Never mysql-client -- mysql -h mysql -ppassword |

This command creates a new Pod in the cluster running a MySQL client and connects it to the server through the Service. If it connects, you know your stateful MySQL database is up and running.

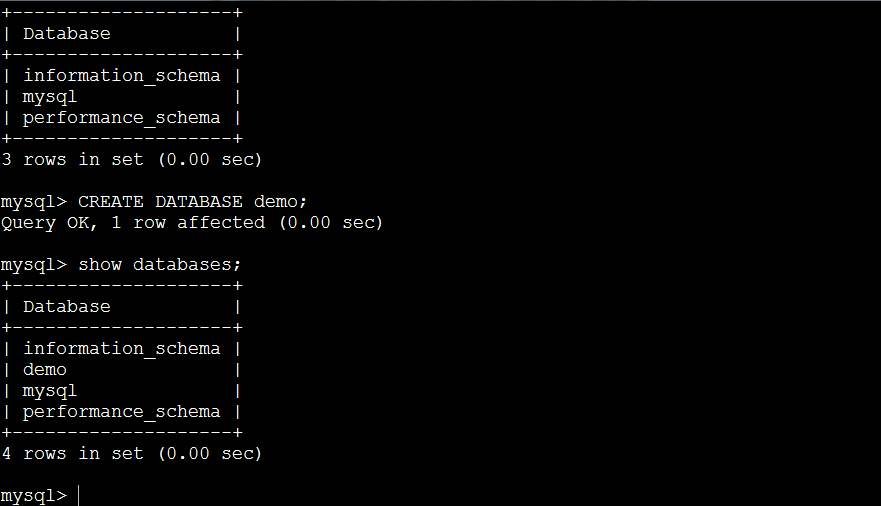


**Get the databases running on mysql.**



**Create a sample database**

**CREATE DATABASE <dbname>;**



**Delete the Deployment and Services.**

|  |
| --- |
| # kubectl delete deployment,svc mysql # kubectl delete pvc mysql-pv-claim # kubectl delete pv mysql-pv-volume |