Deploy MSQL On Kubernetes

A new MySQL server needs to be deployed on Kubernetes cluster. The Nautilus DevOps team was working on to gather the requirements. Recently they were able to finalize the requirements and shared them with the team members to start working on it. Below you can find the details:

1.) Create a PersistentVolume mysql-pv, its capacity should be 250Mi, set other parameters as per your preference.

2.) Create a PersistentVolumeClaim to request this PersistentVolume storage. Name it as mysql-pv-claim and request a 250Mi of storage. Set other parameters as per your preference.

3.) Create a deployment named mysql-deployment, use any mysql image as per your preference. Mount the PersistentVolume at mount path /var/lib/mysql.

4.) Create a NodePort type service named mysql and set nodePort to 30007.

5.) Create a secret named mysql-root-pass having a key pair value, where key is password and its value is YUIidhb667, create another secret named mysql-user-pass having some key pair values, where frist key is username and its value is kodekloud\_tim, second key is password and value is TmPcZjtRQx, create one more secret named mysql-db-url, key name is database and value is kodekloud\_db1

6.) Define some Environment variables within the container:

a) name: MYSQL\_ROOT\_PASSWORD, should pick value from secretKeyRef name: mysql-root-pass and key: password

b) name: MYSQL\_DATABASE, should pick value from secretKeyRef name: mysql-db-url and key: database

c) name: MYSQL\_USER, should pick value from secretKeyRef name: mysql-user-pass key key: username

d) name: MYSQL\_PASSWORD, should pick value from secretKeyRef name: mysql-user-pass and key: password

Note: The kubectl utility on jump\_host has been configured to work with the kubernetes cluster.

**SOLUTION**

**---> apiVersion: v1**

**kind: PersistentVolume**

**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**

**---->**

apiVersion: apps/v1

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