The Kubenetes app deployment contains Mongodb and mongo-express container **i**n minikube cluster.

Mongo-express: Mongo Express is an interactive lightweight Web-Based Administrative Tool to effectively manage MongoDB Databases

MongoDB: MongoDB is an open-source document database and leading NoSQL database.

**Deployment Process in minikube cluster:**

**[STEP 1]: Create Persistent volume and persistent-volume-claim to mount volume in mongodb app**

Manifest file :

1. pv.yaml
2. pvc.yaml

Screenshot:

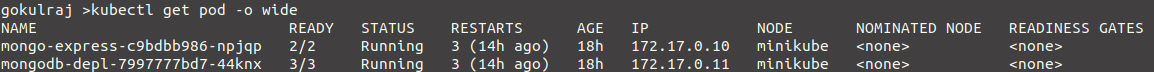


**[STEP 2]: Create pods/services for the Application**

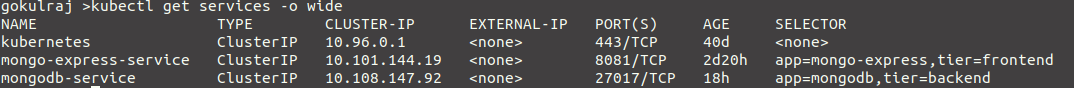
Manifest file for Mongodb and mongo-express pod and services :

1. mongo-express-depl.yaml
2. mongodb-depl.yml

PODS:



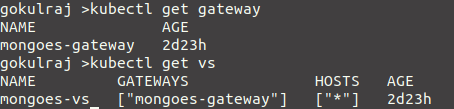
Services: (Cluster-IP)



**[STEP 3]: Create Ingress gateway using istio service mesh to access mongo-express from outside cluster**

1. Install istio using istioctl command-line tool
2. Create gateway and Virtual service for ingress gateway

Manifest file for Gateway and virtual-service to access mongo-express **“mongo-express-gateway.yaml”**

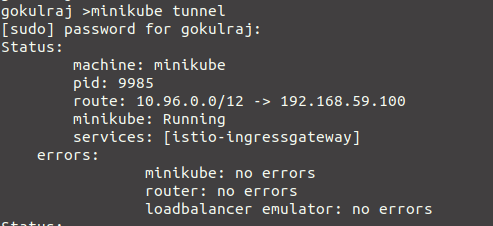


**[STEP 4]: Access the mongo-express from outside cluster:**

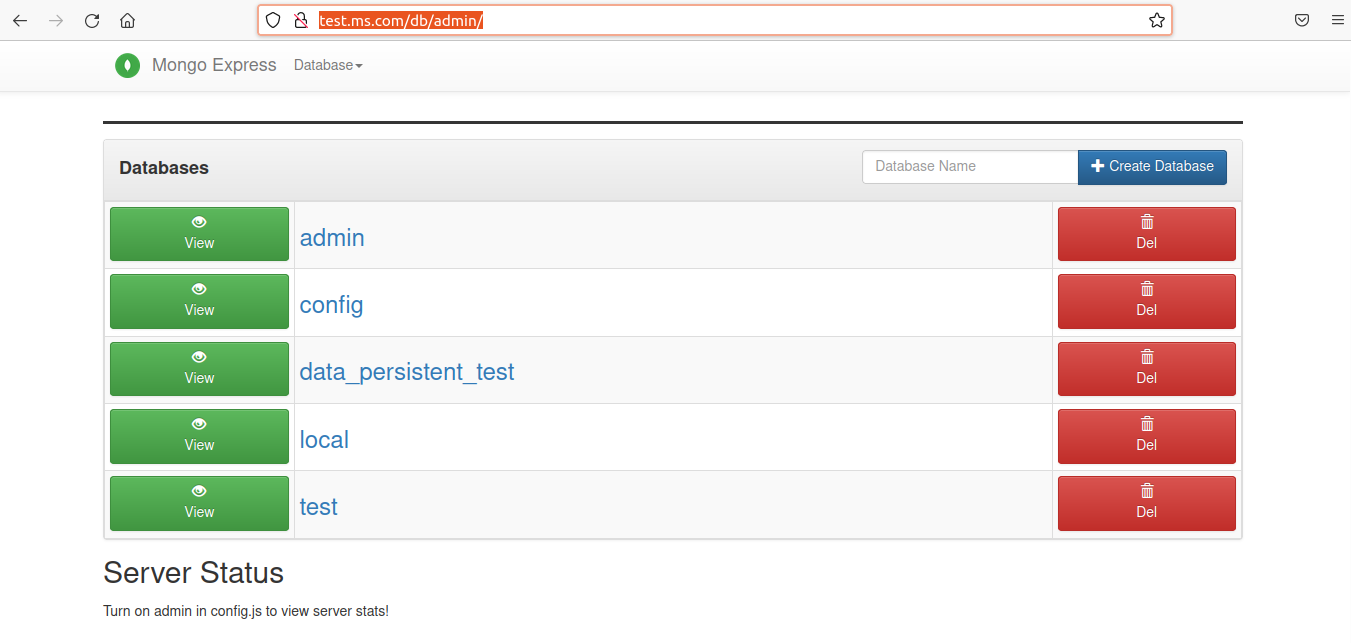
1. Make an entry with the istio ingress gateway IP with hostname in /etc/hosts file in the PC.



1. Enable the minikube tunnel

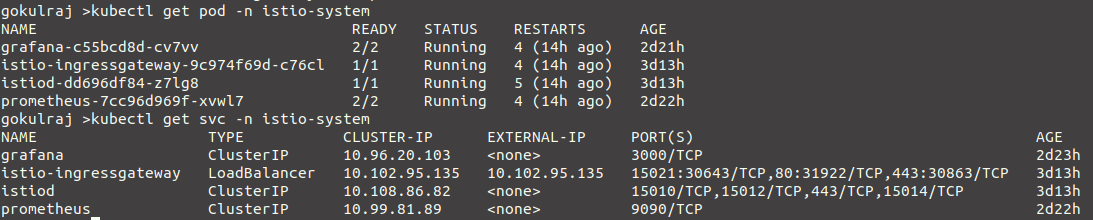


1. Access the app in browser using the hostname configured with the path configured in istio virtual-service



**[STEP 5]: Enabling metrix and Vistualize it using prometheus and grafana**

1. Run the prometheus and grafana pod on istio-system namespace.



1. The mongodb-exporter container is already running on mongodb pod as the mongodb-exporter specs are mentioned in mongodb-depl.yml
2. Create Virtual-service to access grafana using istio ingress gatway

Manifest file : grafana-vs.yaml



1. Access Grafana using the path /grafana with the hosname.

