Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile

1.Created a namespace

```
C:\Windows\System32\cmd.e: × + v
F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl create namespace emqx
namespace/emqx created
F:\Repo\KIII\Kubernetes
                       - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl get ns
                           6d4h
default
                  Active
                  Active
emqx
kube-node-lease
                  Active
kube-public
                  Active
                           6d4h
                  Active
kube-system
F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>
```

2.ConfigMap for emqx

```
F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>vim emqx-configmap.yaml

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl apply -f emqx-configmap.yaml

configmap/emqx-config created

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl get configmap -n emqx

NAME DATA AGE

emqx-config 3 14s

kube-root-ca.crt 1 14m

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>
```

3.Headless service for emqx

4.emqx statefulset

```
F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl apply -f emqx-configmap.yaml

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl apply -f emqx-configmap.yaml

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl apply -f emqx-statefulset.yaml

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl apply -f emqx-statefulset.yaml

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl delete statefulset emqx -n emqx

'\ubectl' is not recognized as an internal or external command,

operable program or batch file.

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl delete statefulset emqx -n emqx

statefulset.apps "emqx" deleted

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl delete statefulset.yaml

statefulset.apps/emqx configured

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl apply -f emqx-statefulset.yaml

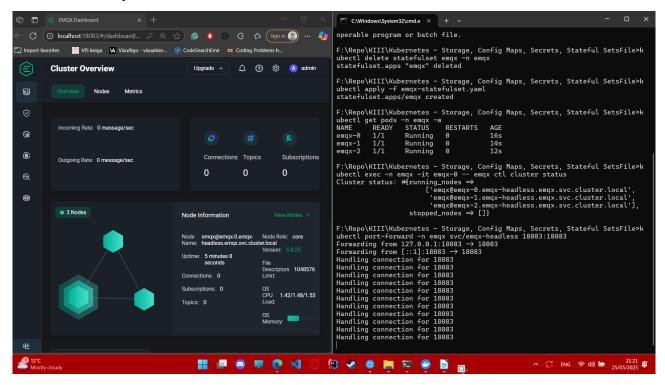
statefulset.apps/emqx configured

F:\Repo\KIII\Kubernetes - Storage, Config Maps, Secrets, Stateful SetsFile>kubectl apply -f emqx-statefulset.yaml

statefulset.apps/emqx configured
```

5. Validate emqx cluster

6.Dashboard emqx



Беќир Фазли 191045

7. The code:

```
emqx-configmap.yaml
apiVersion: v1
```

kind: ConfigMap metadata:

name: emqx-config namespace: emqx

data

EMQX_CLUSTER__STATIC__SEEDS: "emqx@emqx-0.emqx-headless.emqx.svc.cluster.local,emqx@emqx-1.emqx-headless.emqx.svc.cluster.local,emqx@emqx-2.emqx-headless.emqx.svc.cluster.local"

emqx-headless-service.yaml

apiversion: v1 kind: Service metadata:

name: emqx-headless namespace: emqx

spec:

clusterIP: None
selector:
 app: emqx
ports:

- name: mqtt

port: 1883
targetPort: 1883
- name: dashboard
port: 18083

targetPort: 18083

storage: 1Gi

```
emgx-statefulset.vaml
apiversion: apps/v1
kind: StatefulSet
metadata:
  name: emqx
  namespace: emqx
spec:
  serviceName: emqx-headless
  replicas: 3
  selector:
    matchLabels:
      app: emqx
  template:
    metadata:
      labels:
        app: emqx
    spec:
      containers:
      - name: emqx
        image: emqx/emqx:5.0.25
        env:
        - name: POD_NAME
          valueFrom:
            fieldRef:
              fieldPath: metadata.name
        - name: EMQX_NODE__NAME
          value: "emgx@$(POD_NAME).emgx-headless.emgx.svc.cluster.local"
        - name: EMQX_CLUSTER__DISCOVERY_STRATEGY
          value: "static"
        - name: EMQX_CLUSTER__STATIC__SEEDS
          valueFrom:
            configMapKeyRef:
              name: emqx-config
              key: EMQX_CLUSTER__STATIC__SEEDS
        - name: EMQX_NODE__COOKIE
          value: "secretcookie"
        ports:
        - containerPort: 1883
        - containerPort: 18083
        volumeMounts:
        - name: emqx-data
          mountPath: /opt/emqx/data
      volumes:
      - name: emqx-data
        persistentVolumeClaim:
          claimName: emqx-pvc
  volumeClaimTemplates:
  - metadata:
      name: emqx-data
    spec:
      accessModes: [ "ReadWriteOnce" ]
      resources:
        requests:
```