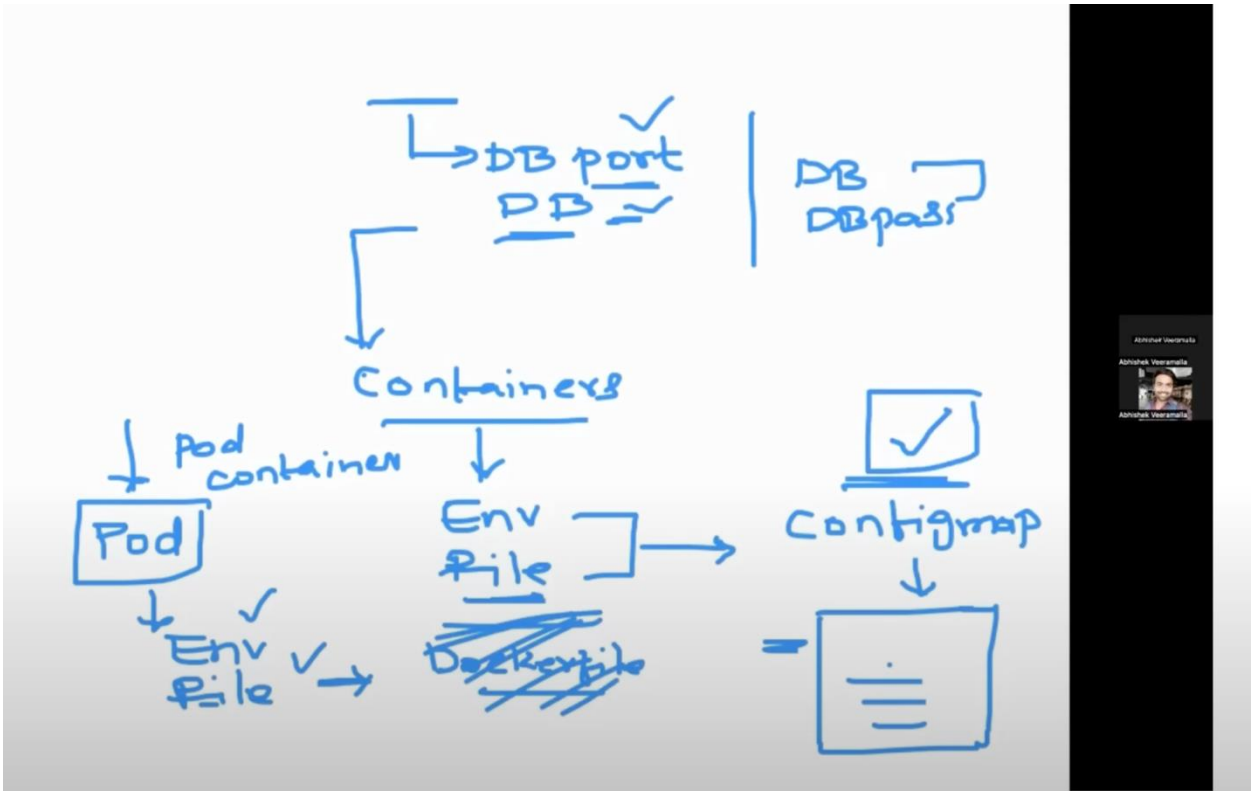
ConfigMaps and Secrets in Kubernetes.

Why is configmaps and secrets?
Difference between ConfigMaps and secrets

Configmap.

User<--------APP---------→DB.



Configmap is used to store the data. (non-sensitive)

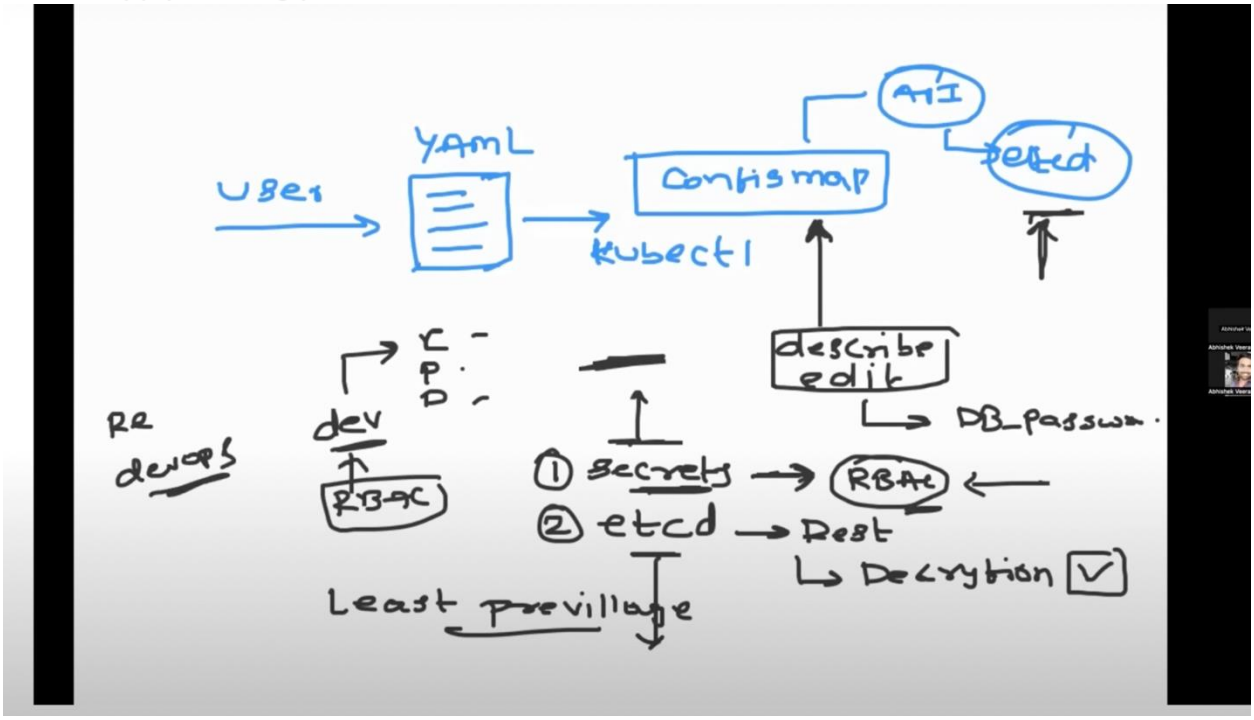
Secrets deals with sensitive data. Kubernets will encrypt the data before storing the etcd.

In etcd data is saved as objects.

We will encrypt the data. Before the storing the data in etcd it will encrypt.

User
Config.yaml

Kubectl apply -f config.yaml



Step 1:

Create the cm.yml file.

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: test-cm
data:
  db-port: "3306"
```

save it.

Kubectl apply -f cm.yml

kubectl describe cm test-cm

kubectl apply -f deploymet.yaml

kubectl exec -it nameofThePod -- /bin/bash

env | grep db

inside the deployment.yam,

add this

```
env:
 - Name: DB-PORt
   valueFrom:
    configMapKeyRef:
     name: test-cm
     key: db-port
```

```
env | grep -i db
```

```
kubectl apply -f deployment.yaml
```

```
kubectl exec -it podname -- /bin/bash.
```

Use the VolumeMounts
Use them as files.
Open deployment.yaml

Delete the env

Create the volume mount.

Deployment.yaml(this for env varibel)

```
apiVersion: apps/v1
kind: Deployment
```

```
metadata:
  name: sample-python-app
  labels:
    app: sample-python-app
spec:
  replicas: 2
  selector:
    matchLabels:
      app: sample-python-app
  template:
    metadata:
      labels:
        app: sample-python-app
    spec:
      containers:
      - name: python-app
        image: docker.io/akhil626/python:v1
        env:
         - name: DB-PORT
           valueFrom:
            configMapKeyRef:
             name: test-cm
```

```
            key: db-port
      ports:
         - containerPort: 8000
```

Changing the env variable is not allowed in Kubernetes. So use the volume mounts.

Deployment.yaml (for volume mounts)

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: sample-python-app
  labels:
    app: sample-python-app
spec:
  replicas: 2
  selector:
    matchLabels:
      app: sample-python-app
```

```
  template:
    metadata:
      labels:
        app: sample-python-app
    spec:
      containers:
      - name: python-app
        image: docker.io/akhil626/python:v1
        volumeMounts:
        - name: db-connection
          mountPath: /opt
        ports:
        - containerPort: 8000
      volumes:
      - name: db-connection
        configMap:
          name: test-cm
```

other way of creating the secrets

```
kubectl create secret generic test-secret --from-literal=db-port="3306"
```

```
kubectl describe secret test-secret¥


echo MzMwNg== | base64 --decode
```