**Lesson 4 Demo 11**

**Understanding Config Maps and Secrets**

**Objective:** Understanding Config Maps and Secrets

**Tools required:** kubeadm, kubectl, kubelet, and etcd

**Prerequisites:** kubeadm, kubectl, kubelet, and etcd should be installed

Steps to be followed:

1. Adding a Config Map entry to the Pod
2. Creating secrets using kubectl

**Step 1: Adding a Config Map entry to the Pod**

1. Run the following command:

**vi configmap.yaml**

1. Add the following code to the configmap.yaml file:

**kind: ConfigMap**

**apiVersion: v1**

**metadata:**

**name: example-configmap**

**data:**

**# Configuration values can be set as key-value properties**

**database: mongodb**

**database\_uri: mongodb://localhost:27017**

Text

Description automatically generated

1.3 Create a Config Map using the configmap.yaml file:

**kubectl create -f configmap.yaml**

**kubectl get configmap**

Text

Description automatically generated

1.4 Run the following command:

**vi configpod.yaml**

1.5 Write the following code in the configpod.yaml file:

**kind: Pod**

**apiVersion: v1**

**metadata:**

**name: pod-env-var**

**spec:**

**containers:**

**- name: env-var-configmap**

**image: nginx:1.7.9**

**envFrom:**

**- configMapRef:**

**name: example-configmap**

Text

Description automatically generated with low confidence

1.6 Create a Pod by using the following command:

**kubectl create -f configpod.yaml**

Text

Description automatically generated

**Step 2: Creating secrets using kubectl**

2.1 Run the following command:

**vi config-svc.yaml**

2.2 Write a code in the config-svc.yaml file to create a secret for Ngnix Pod:

**kind: Pod**

**apiVersion: v1**

**metadata:**

**name: pod-env12**

**spec:**

**containers:**

**- name: env-var-configmap**

**image: nginx:1.7.9**

**env:**

**- name: testenv**

**valueFrom:**

**configMapKeyRef:**

**name: example-configmap**

Text

Description automatically generated **key: database**

2.3 Create a Pod with service by using the following commands:

**kubectl create -f config-svc**

**kubectl get pods**

Graphical user interface, text

Description automatically generated

2.4 Access the container and verify the database by using the following commands:

**kubectl exec -it pod-env12 bash**

**env**

**env | grep database**

Text

Description automatically generated

2.5 Run the following command:

**vi configfile.yaml**

2.6 Create a Pod with the volume configfile.yaml:

**apiVersion: v1**

**kind: Pod**

**metadata:**

**name: testconfig**

**spec:**

**containers:**

**- name: test**

**image: docker.io/httpd**

**volumeMounts:**

**- name: config-volume**

**mountPath: /tmp/myenvs/**

**volumes:**

**- name: config-volume**

**configMap:**

**name: example-configmap**

**restartPolicy: Never**

Text

Description automatically generated

2.7 Create a Pod using the following command:

**kubectl create -f configfile.yaml**

A screenshot of a computer

Description automatically generated

2.8 Access the Pod by using the following command:

**kubectl exec -it testconfig bash**

