**Lab Exercise 14- Creating a ConfigMap from a Folder**

In this lab exercise, you will learn how to create a ConfigMap in Kubernetes from a folder containing multiple files. This allows you to manage multiple configuration files as a single ConfigMap.

**Step 1: Set Up Kubernetes Cluster**

Ensure you have access to a Kubernetes cluster. You can use a local setup with Minikube, kind, or use a cloud-based Kubernetes service.

**Step 2: Prepare the Folder with Multiple Files**

Create a directory named configmap-folder:

mkdir configmap-folder

cd configmap-folder/

Create multiple configuration files inside the directory:

**config1.txt** with content:

key1=value1

**config2.txt** with content:

key2=value2

**config3.txt** with content:

key3=value3

Verify the directory structure:

ls configmap-folder/

cd ..

You should see the files config1.txt, config2.txt, and config3.txt.

**Step 3: Create a ConfigMap from the Folder**

Create a ConfigMap named folder-configmap from the folder:

kubectl create configmap folder-configmap --from-file=configmap-folder/

Verify the ConfigMap:

kubectl get configmap folder-configmap -o yaml

You should see the folder-configmap with the contents of all the files in the configmap-folder.

**Step 4: Use the ConfigMap in a Pod**

Create a file named **pod-folder-configmap.yaml** with the following content:

apiVersion: v1

kind: Pod

metadata:

name: configmap-pod

spec:

containers:

- name: nginx

image: nginx

envFrom:

- configMapRef:

name: folder-configmap

In this manifest:

* The Pod mounts the folder-configmap at /etc/config.
* The busybox container concatenates and prints the contents of the files.

Apply the manifest to create the Pod:

kubectl apply -f pod-folder-configmap.yaml

Check the status of the Pod:

kubectl get pods

View the Pod logs to see the content of the mounted files:

kubectl exec -it configmap-pod -- printenv

You should see the contents of config1.txt, config2.txt, and config3.txt printed.

**Step 5: Clean Up**

After completing the exercise, clean up the resources created:

kubectl delete pod folder-configmap-pod

kubectl delete configmap folder-configmap

rm -r configmap-folder