**Lab Name: Creating a GKE Cluster Via Cloud**

**Deploy GKE clusters**

set the environment variable for the zone and cluster name.

export my\_zone=us-central1-a export my\_cluster=standard-cluster-1

create a Kubernetes cluster.

gcloud container clusters create $my\_cluster --num-nodes 3 --zone $my\_zone --enable-ip-alias

**Modify GKE Clusters**

to modify standard-cluster-1 to have four nodes

gcloud container clusters resize $my\_cluster --zone $my\_zone --size=4

press y to confirm.

**Connect to a GKE Clusters**

to allow communicating with that cluster through the kubectl command-line tool

*gcloud container clusters get-credentials $my\_cluster --zone $my\_zone*

This command creates a .kube directory in your home directory if it doesn't already exist. In the .kube directory, the command creates a file named config if it doesn't already exist, which is used to store the authentication and configuration information. The config file is typically called the kubeconfig file.

**Note:** The kubeconfig file can contain information for many clusters. The currently active context (the cluster that kubectl commands manipulate) is indicated by the current-context property.

You don't have to run the gcloud container clusters get-credentials command to populate the kubeconfig file for clusters that you created in the same context (the same user in the same environment), because those clusters already have their details populated when the cluster is created. But you have to run the command to connect to a cluster created by another user or in another environment. The command is also an easy way to switch the active context to a different cluster.

Gcloud Documentation Page

**Use kubectl to inspect a GKE cluster**

After the kubeconfig file is populated and the active context is set to a particular cluster, you can use the kubectl command-line tool to execute commands against the cluster. Most such commands ultimately trigger a REST API call against the master API server, which triggers the associated action.

You can use the following command to view the content of the kubeconfig file

*kubectl config view*

You can get the cluster information for the active context by running the following command

*kubectl cluster-info*

command to print out the active context

*kubectl config current-context*

command to print out some details for all the cluster contexts in the kubeconfig file.

*kubectl config get-contexts*

command to change the active context

*kubectl config use-context gke\_${GOOGLE\_CLOUD\_PROJECT}\_us-central1-a\_standard-cluster-1*

You may have more than one cluster in a project. You can use this approach to switching the active context when your *kubeconfig* file has the credentials and configuration for several clusters already populated.

This approach requires the full name of the cluster, which includes the gke prefix, the project ID, the location, and the display name, all concatenated with underscores.

command to view the resource usage across the nodes of the cluster