ORACLE

2023 December Workshop OKE on OCI

Overall Flow

1. We will be using the OKE module to spin up an OKE enhanced cluster in OCI.
2. We will explore the OKE cluster, and its resources.
3. We deploy a sample application and access the app from the internet.

Step 1 Setting up the OKE Cluster

1. Login to your OCI Console.
2. Launch Cloud Shell in the OCI Console.

3. Generate your SSH keys

```
$ ssh-keygen -t ecdsa -b 384
```

This step creates two files in the .ssh directory under your home directory.

4. Run the following command to copy the OCI config file from its default location.

```
$ mkdir -p /home/<your user name>/.oci
$ cp /etc/oci/config /home/<some user name>/.oci/config
```

5. Run the following command to clone the terraform scripts

```
$ git clone https://github.com/skripted/202312workshopCA.git
```

6. This creates a new directory “202312workshopCA” in your current path.
7. Change your working directory to **202312workshopCA**.
8. Rename the file workshop_auto_tfvars to workshop.auto.tfvars.
9. Edit the contents of the workshop.auto.tfvars file.

config_file_profile	“DEFAULT”
home_region	Your OCI Home Region
region	the same as above
tenancy_id	Tenancy OCID (Identifier)

compartment_id	Compartment OCID (identifier) keep this same as above
ssh_public_key_path	"/home/<some user name>/.ssh/id_ecdsa.pub"
ssh_private_key_path	"/home/<some user name>/.ssh/id_ecdsa"

10. Run Terraform

```
$ terraform init
$ terraform plan
$ terraform apply
```

Sit back and relax, while the cluster resources are being provisioned. We will walk through OKE concepts during the provisioning of the cluster resources.

References:

1. https://oracle-terraform-modules.github.io/terraform-oci-oke/gs/intro.html
2. https://github.com/oracle-terraform-modules/terraform-oci-oke
3. https://medium.com/oracledevs/announcing-release-of-terraform-oke-kubernetes-module-5-0-part-1-59205d1b9d8d

Step 2 Exploring the OKE Cluster

1. Review https://docs.oracle.com/en-us/iaas/Content/ContEng/Tasks/contengaccessingclusterkubectl.htm#cloudshellkubectl
2. We will access the OKE cluster through Cloud Shell
3. Follow the instructor on various kubectl commands to review various resources in the cluster.

Step 3 Deploy Sample App in OKE Cluster

1. We will deploy a stateless application in the K8 Cluster (https://kubernetes.io/docs/tutorials/stateless-application/expose-external-ip-address/)
2. We will apply the manifest that contains the Deployment.

```
$ kubectl apply -f https://k8s.io/examples/service/load-balancer-example.yaml
```

3. The preceding command creates a Deployment and an associated ReplicaSet. The ReplicaSet has five Pods each of which runs the Hello World application.
4. Follow the instructor to run some commands to get information about this application.
5. Expose the application through a service of type Loadbalancer.

```
$ kubectl expose deployment hello-world --type=LoadBalancer --name=my-service
```

6. Follow the instructor to run additional commands to review the environment and the service.
7. Access the application! You have successfully deployed an app on OKE!

ORACLE

Connect with us

Call +**1.800.ORACLE1** or visit **oracle.com**. Outside North America, find your local office at: **oracle.com/contact**.

blogs.oracle.com facebook.com/oracle twitter.com/oracle