**Kubernetes (Kops and EKS (Elastic Kubernetes Service) )**

**kOps, also known as Kubernetes operations, is an open-source project which helps you create, destroy, upgrade, and maintain a highly available, production-grade Kubernetes cluster. Depending on the requirement, kOps can also provision cloud infrastructure.**

**Setting up the Cluster by using the Kops Method: Reference(https://www.fosstechnix.com/setup-kubernetes-on-aws-using-kops/)**

**Step #1: Install AWS CLI on Ubuntu (Get one ec2 instance)**

sudo apt-get update

sudo apt-get install awscli -y

**Step #2: Install Kubectl Binary with CURL on Ubuntu**

🡪 **Download kubectl binary with curl on Ubuntu using below command**

sudo curl -LO https://storage.googleapis.com/kubernetes-release/release/$(curl -s <https://storage.googleapis.com/kubernetes-release/release/stable.txt)/bin/linux/amd64/kubectl>

**🡪 Make the kubectl binary executable**

sudo chmod +x ./kubectl

🡪 **Move kubectl to /usr/local/bin/kubectl directory**

sudo mv ./kubectl /usr/local/bin/kubectl

**Use this if you want to run in one command**

**sudo curl -LO https://storage.googleapis.com/kubernetes-release/release/$(curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt)/bin/linux/amd64/kubectl**

**sudo chmod +x ./kubectl**

**sudo mv ./kubectl /usr/local/bin/kubectl**

**Step #3: Install KOPS on Ubuntu Instance**

🡪 **Download the KOPS setup on Ubuntu using curl**

curl -LO https://github.com/kubernetes/kops/releases/download/$(curl -s https://api.github.com/repos/kubernetes/kops/releases/latest | grep tag\_name | cut -d '"' -f 4)/kops-linux-amd64

🡪 **set the execution permission**

sudo chmod +x kops-linux-amd64

🡪 **move the kops to /usr/local/bin directory**

sudo mv kops-linux-amd64 /usr/local/bin/kops

**Use this if you want to run in one command**

**curl -LO https://github.com/kubernetes/kops/releases/download/$(curl -s https://api.github.com/repos/kubernetes/kops/releases/latest | grep tag\_name | cut -d '"' -f 4)/kops-linux-amd64**

**sudo chmod +x kops-linux-amd64**

**sudo mv kops-linux-amd64 /usr/local/bin/kops**

**Step #4: Creating Domain and Hosted Zone in AWS**

🡪Get any Domain name from and third party domain name provider (ex- godaddy etc)

🡪 Go the route(R53 Service in AWS and create the domain configuration (Create r53 service and get the namespace id ) (You can create either public or private hosted zone)

🡪 Take the name space service and configure in the domain provider DNS setting

**Step #5: Create and configure IAM User in AWS**

**Create IAM user in AWS using login console -> IAM -> ADD User -> Username -> Select Programmatic access -> then you will get Access Key ID and Secret Access Key.**

**Configure AWS CLI with your Access Key ID,  Secret Access  key and region**

**🡪User command to access**

aws configure

**Step #6: Create IAM User with full S3, EC2, Route53 and VPC access in AWS**

**Now create IAM user with full S3, EC2, Route 53 and VPC access in your AWS account as shown below**

1. AmazonS3FullAccess
2. AmazonEC2FullAccess
3. AmazonRoute53FullAccess
4. IAMFullAccess
5. AmazonVPCFullAccess
6. Event bridge full access

**Step #7: Creating S3 Bucket using command line**

🡪 **Create the S3 bucket to store Kubernetes cluster states**

aws s3 mb s3://k8.batabittu.fun

**🡪 Enable versioning on S3 bucket**

aws s3api put-bucket-versioning --bucket k8.batabittu.fun --versioning-configuration Status=Enabled

🡪 **Export kops state**

export KOPS\_STATE\_STORE=s3://k8.batabittu.fun

**Step #8: Create SSH Keys**

**🡪 Create ssh keys on Ubuntu instance to exchange kubernetes cluster and connect**

ssh-keygen

**Step #9: Setup Kubernetes cluster on AWS using KOPS**

🡪 **Create Kubernetes on AWS using Kops using below command**

kops create cluster --cloud=aws --zones=ap-south-1a --name=k8.batabittu.fun --dns-zone=k8.batabittu.fun --dns public

**🡪 if you want to install calico networking along with setup of Kubernetes KOPS add –networking with calico as shown below (**This you can user where you will get networking capailaplity)

kops create cluster --cloud=aws --zones=ap-south-1a --networking calico --name=k8.batabittu.fun --dns-zone=k8.batabittu.fun --dns public

🡪 **If you want to create K8s cluster using multimaster and different availability zone use below command (**This option will give the access to create no of nodes (master and worker nodes**)**

kops create cluster --cloud=aws --zones=ap-south-1a,ap-south-1b --networking calico --master-size t3.medium --master-count 3 --node-size t3.xlarge --node-count 3 --name=k8.batabittu.fun --dns-zone=k8.batabittu.fun --dns public

🡪 **Configure the k8s kops cluster using below command,**

kops update cluster k8.batabittu.fun --yes --admin

🡪 **Validate the Kubernetes KOPS cluster**

kops validate cluster

***==Commands==***

🡪 **To list the clusters**

kops get cluster

🡪 **To view Cluster information**

kubectl cluster-info

or

**To View in detail (This will help for the debug and diagnose cluster problems)**

kubectl cluster-indo dump

**🡪 To delete Kubernetes KOPS cluster**

kops delete cluster --name=k8s.fosstechnix.info -y

🡪 **How to add or change node size/type in Kops Cluster**

kops get ig (This will list the cluster nodes details )

**Output**

Using cluster from kubectl context: k8s.fosstechnix.info

NAME ROLE MACHINETYPE MIN MAX ZONES

master-ap-south-1a Master t3.medium 1 1 ap-south-1a

nodes-ap-south-1a Node t3.medium 1 1 ap-south-1a

🡪 **To edit the nodes**

kops edit ig master-ap-south-1a

**Conclusion**

**In the above steps where all followed steps to create the cluster with 1 maser and 1 worker node**

**Note: create the one ec2 instances and in that installed the kops configuration and when every u want create the cluster you can create and if you want to delete the cluster you can delete the cluster by below command**

**kops delete cluster --name=<Cluster name> -y**

**And again you want to create the cluster clear with the following steps**

**again execute the 9 step and the (creating cluster , updating and validate)**

=========================================================

***Pods***

🡪 **To create the pod with using normal command**

Ex- create the pod with nginx pod with using the command line

kubectl run <new pod name> --image=<images name>