1. AWS Account.

2. Create VPC & Deploy t2.micro Ubuntu Instance/Server for Management.

3. Create AWS S3 Bucket - k8sb04kopsbucket

4. Domain Name (Can be purchased from Godaddy) - teluguk8s.xyz

5. Create AWS Route53 Zone using the domain from step 4.

6. Create a AWS User with Access Key & Secret Key or assign role.

7. Login to Mgmt Server, sudo to root & Generate SSH-Keys.

8. Create a folder called .aws under home and create credetails file. Add

AWS Access and Secret Key in the credentails file.

9. Download kops binary to Mgmt Server & provide execution permissions.

10. Download kubectl binary to Mgmt Server & provide execution permissions.

11. Deploy Kubernetes Server with KOPS.

export NAME=sreeharshak8s.xyz

export KOPS\_STATE\_STORE=s3://sreeharshak8s.xyz

export AWS\_REGION=us-east-1

export CLUSTER\_NAME=sreeharshak8s.xyz

export EDITOR='/usr/bin/nano'

#export K8S\_VERSION=1.6.4

Multi-Master Dry-run:

kops create cluster --name=sreeharshak8s.xyz \

--state=s3://sreeharshak8s.xyz --zones=us-east-1a,us-east-1b,us-east-1c \

--node-count=3 --master-count=3 --node-size=t3.medium --master-size=t3.medium \

--master-zones=us-east-1a,us-east-1b,us-east-1c --master-volume-size 10 --node-volume-size 10 \

--dns-zone=sreeharshak8s.xyz --dry-run --output yaml

Multi-Master Deploy:

kops create cluster --name=sreeharshak8s.xyz \

--state=s3://sreeharshak8s.xyz --zones=us-east-1a,us-east-1b,us-east-1c \

--node-count=3 --master-count=3 --node-size=t3.medium --master-size=t3.medium \

--master-zones=us-east-1a,us-east-1b,us-east-1c --master-volume-size 10 --node-volume-size 10 \

--dns-zone=sreeharshak8s.xyz --yes

kops create cluster --name=devopsk8s.xyz --state=s3://devopsk8s.xyz --zones=us-east-1a --node-count=2 --node-size=t2.micro --master-size=t2.small --master-volume-size 20 --node-volume-size 10 --dns-zone=devopsk8s.xyz --yes

kops create cluster --name=trainingk8s.xyz \

--state=s3://sreek8s --zones=us-east-1a,us-east-1b,us-east-1c \

--node-count=2 --node-size=t3.medium --master-size=t3.medium \

--master-volume-size 10 --node-volume-size 10 \

--dns-zone=trainingk8s.xyz --yes

kops create cluster --name=trainingk8s.xyz \

--state=s3://sreek8s --zones=us-east-1a,us-east-1b,us-east-1c \

--node-count=3 --node-size=t3.medium --master-size=t3.medium \

--master-volume-size 10 --node-volume-size 10 \

--dns-zone=trainingk8s.xyz --yes

**Kubectl bash completion:**

echo 'source <(kubectl completion bash)' >>~/.bashrc

echo 'alias ku=kubectl' >>~/.bashrc

echo 'complete -F \_\_start\_kubectl ku' >>~/.bashrc

kops get cluster --state s3://devopsk8s.xyz

kops get ig --name devopsk8s.xyz --state s3://devopsk8s.xyz

kops edit ig --name=devopsk8s.xyz master-us-east-1a --state s3://devopsk8s.xyz

kops edit ig --name=devopsk8s.xyz nodes --state s3://devopsk8s.xyz

kops update cluster --name devopsk8s.xyz --yes --state s3://devopsk8s.xyz

kops rolling-update cluster --name devopsk8s.xyz --yes --state s3://devopsk8s.xyz

kops delete cluster --name=devopsk8s.xyz --state s3://devopsk8s.xyz --yes

**Cluster Smoke Testing:**

kubectl version --short

kubectl get nodes

kubectl cluster-info

kubectl get ns

kubectl get pods

kubectl get pods -n kube-system

kubectl get service -A

kubectl run pod01 --image=sreeharshav/rollingupdate:v5

kubectl expose pod pod01 --port=8000 --target-port=80 --type=NodePort

kubectl get pods -o wide

kubectl get pods -o wide --no-headers

kubectl create deployment deploy01 --image=sreeharshav/rollingupdate:v5 --dry-run -o yaml

**MINIKUBE:**

sudo apt-get update && sudo apt-get install -y apt-transport-https

curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -

echo "deb https://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee -a /etc/apt/sources.list.d/kubernetes.list

sudo apt-get update

sudo apt-get install -y kubectl conntrack

curl https://get.docker.com | bash

curl -Lo minikube https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64 && chmod +x minikube

sudo mv minikube /usr/local/bin

set NO\_PROXY=localhost,127.0.0.1,10.96.0.0/12,192.168.99.1/24,192.168.39.0/24

minikube start --vm-driver=none