KOPS installation in AWS::  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
KOPS - Kubernetes Operations

1. Launch one Ubuntu instance and execute below steps to install kops.

2. kops binary download

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  
chmod +x kops-linux-amd64  
sudo mv kops-linux-amd64 /usr/local/bin/kops

3. aws cli setup to enable ubuntu to interact with aws.  
apt-get update  
apt-get install -y python-pip   
sudo pip install awscli

aws –version

4.  
- Create an IAM role and give access to the ec2 workstaiton by attaching that role  
- Create S3 bucket and enable versioning.

5. kubectl installation (K8s cli)

snap install kubectl --classic  
kubectl version

ssh-keygen -f .ssh/id\_rsa

6. Environment variables setup -- Remember cluster name should ends with k8s.local

updated these two vars in .bashrc & .profile in ~ dir.

export KOPS\_CLUSTER\_NAME=lab1.k8s.local  
export KOPS\_STATE\_STORE=s3://kops-state-lab1-bucket

7. Create cluster:: -- This will actually prepare the configuration files.  
kops create cluster \  
--node-count=2 \  
--node-size=t2.micro \  
--master-size=t2.micro \  
--zones=us-east-1b \  
--name=${KOPS\_CLUSTER\_NAME}

(optional)if you wanted to review & edit the cluster configuration:

kops edit cluster --name ${KOPS\_CLUSTER\_NAME}

RUN if you're okay withe the configuration run the command with --yes as like below:

kops update cluster --name ${KOPS\_CLUSTER\_NAME} –yes

Output shows like below..::  
 Cluster is starting. It should be ready in a few minutes.

Suggestions:  
 \* validate cluster: kops validate cluster  
 \* list nodes: kubectl get nodes --show-labels  
 \* ssh to the master: ssh -i ~/.ssh/id\_rsa admin@api.lab.k8s.local  
 \* the admin user is specific to Debian. If not using Debian please use the appropriate user based on your OS.  
 \* read about installing addons at: <https://github.com/kubernetes/kops/blob/master/docs/addons.md>.

To validate the cluster::

kops validate cluster

Validating cluster lab1.k8s.local

INSTANCE GROUPS  
 NAME ROLE MACHINETYPE MIN MAX SUBNETS  
 master-us-east-1a Master m3.medium 1 1 us-east-1a  
 nodes Node t2.medium 1 1 us-east-1a

NODE STATUS  
 NAME ROLE READY  
 ip-172-20-52-91.ec2.internal node True  
 ip-172-20-54-252.ec2.internal master True

Your cluster lab.k8s.local is ready

8. deploying dashboard feature::

kubectl apply -f <https://raw.githubusercontent.com/kubernetes/dashboard/master/src/deploy/recommended/kubernetes-dashboard.yaml>

Edit master's security group:

- Make sure 443 port is allowed from ANYWHERE in aws security group.

To get admin user's password::

root@ip-172-31-94-144:~# kops get secrets kube --type secret -oplaintext Or grep password: ~/.kube/config

XcxhdTyPAIK98ixB7jJnoMTqMgkhA49c -lab1

Launch kubernetes dashboard url:

https://<master-ip>:<apiserver-port>/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/

admin  
 password

-- Select the token option and paste the below one.

Token generation for admin:

root@ip-172-31-94-144:~# kops get secrets admin --type secret -oplaintext  
m9OJwmL4SbomYjnWklEBtwv4GBxYIZS1 -lab1

root@ip-172-31-94-144:~# kubectl cluster-info

Kubernetes master is running at <https://api-lab-k8s-local-df1a7n-1016419148.us-east-1.elb.amazonaws.com>  
 KubeDNS is running at <https://api-lab-k8s-local-df1a7n-1016419148.us-east-1.elb.amazonaws.com/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy>

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.

root@ip-172-31-94-144:~# kubectl get nodes -- To get the nodes status   
NAME STATUS ROLES AGE VERSION  
ip-172-20-59-100.ec2.internal Ready node 8m v1.9.8  
ip-172-20-63-182.ec2.internal Ready master 9m v1.9.8