

## THIS DOCUMENT STATES HOW TO SET UP KUBERNETES CLUSTER IN AWS:-

You must have aws cli installed on your server and an access key and secret key with admin access permission in it and it should be configured on your ec2 instance, for doing this execute the below command:-

```
sudo apt-get install awscli  
aws configure
```

Now enter your access key and your secret key along with the region, after that you are all set to create cluster in kubernetes on AWS

You also need to install docker on the server and login to your docker registry, for installing docker follow the link below

<https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-16-04>

To login to your docker registry

```
sudo docker login docker-registry.enveu.com:5443
```

Login with username and password: docker-admin/3Zj7btENx9hb

## FOR CREATING CLUSTER FOLLOW THE BELOW STEPS:-

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  
pip install awscli
```

To check your awscli version execute the below command:-

```
aws --version
```

4. Create IAM user & make a note of access key & security key and then Create S3 bucket and enable versioning.

```
aws configure

##Give access & security access key details here along with
#####the region
```

5. kubectl installation (K8s cli)

```
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
```

Now create an ssh key for creating cluster it will act as a pem file for accessing your master and nodes server and it is also essential for creating your kubernetes cluster, for creating it execute the below command:-

```
ssh-keygen -f .ssh/id_rsa
```

6. Environment variables setup -- Remember cluster name should ends with k8s.local and updated these two vars in .bashrc & .profile in ~ dir.

Cluster name should end with **.k8s.local**

```
export KOPS_CLUSTER_NAME=<cluster name>
export KOPS_STATE_STORE=s3://s3-bucket-name
```

Restart .bashrc using comand:- `source ~/.bashrc`

7. Create cluster:: -- This will actually prepare the configuration files.

```
kops create cluster \  
--node-count=1 \  
--node-size=t2.micro \  
--master-size=t2.micro \  
--zones=us-east-1a \  
--name=${KOPS_CLUSTER_NAME}
```

For multiaz you can pass multiple zone seperated by comma

```
kops create cluster \  
--node-count=3 \  
--node-size=t2.micro \  
--master-size=t2.micro \  
--zones=ap-south-1a,ap-south-1b,ap-south-1c \  
--vpc=<vpc-id> \  
--name=${KOPS_CLUSTER_NAME}
```

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

```
kops edit cluster --name ${KOPS_CLUSTER_NAME}
```

For using your existing vpc to launch cluster follow link mentioned blow:-

<https://medium.com/@while1eq1/launching-a-kubernetes-cluster-into-an-existing-vpc-in-aws-using-kops-terraform-81ac856dfee7>

If you get any errors regarding subnets while using existing vpc for launching cluster, follow the link below:-

<https://github.com/kubernetes/kops/issues/2294>

RUN if you're okay with 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.advith.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 advith.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 advith.k8s.local is ready

#### 8. deploying dashboard feature::

Follow the link below to apply kubernetes dashboard

<https://kubernetes.io/docs/tasks/access-application-cluster/web-ui-dashboard/>

Edit master's security group:

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

To get admin user's password::

```
kops get secrets kube --type secret -o plaintext
```

You will get a password on your screen like this

```
sr1myMCrxeIWfV6fhdlz1alo7lKWteg
```

Launch kubernetes url:

<http://master.dns/ui>

admin

password

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

Token generation for admin:

```
kops get secrets admin --type secret -oplaintext  
8XmR3sAZCsV38gGCa50hTYXtOPpBztTR
```

For getting your cluster info

```
kubectl cluster-info  
  
Kubernetes master is running at  
https://api-advith-k8s-local-df1a7n-1016419148.us-east-1.elb.amazonaws.com  
  
KubeDNS is running at  
https://api-advith-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'.
```

```
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

### TO DELETE YOUR CLUSTER:-

```
kops delete cluster --name=<cluster-name> --yes
```

### BASIC KOPS COMMANDS THAT WILL HELP YOU IN MODIFYING YOUR CLUSTER AND NODES:-

#### To edit cluster

```
Kops edit cluster --name=(cluster-name) --yes
```

#### To edit nodes

```
Kops edit ig --name=(cluster-name) nodes
```

#### To edit master

```
Kops edit ig --name=(cluster-name) master
```



For allowing kubernetes to pull image from your docker private registry you need to provide the credentials path to kubernetes so that it should read the docker config.json other it will show error failed to pull images from your registry, to overcome this issue execute the below command

```
kubectl create secret generic regcred \  
  --from-file=.dockerconfigjson=<path/to/.docker/config.json> \  
  --type=kubernetes.io/dockerconfigjson
```