

Installation of kops:

<https://github.com/kubernetes/kops/blob/master/docs/aws.md>

**Step1:** ssh into the ec2 instance and copy paste this.

curl -Lo kops 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

sudo mv ./kops /usr/local/bin/

**step2:** install kubectl by copy pasting this on ec2 terminal.

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

chmod +x ./kubectl

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

SSH into ec2 instance:

ssh -i video-keypair.pem ec2-user@100.26.0.49

**step3:**

# configure the aws client to use your new IAM user

aws configure # Use your new access and secret key here

aws iam list-users # you should see a list of all your IAM users here

**step4**:

# Because "aws configure" doesn't export these vars for kops to use, we export them now

export AWS\_ACCESS\_KEY\_ID=$(aws configure get aws\_access\_key\_id)

export AWS\_SECRET\_ACCESS\_KEY=$(aws configure get aws\_secret\_access\_key)

**step5**: create bucket

aws s3api create-bucket \

--bucket tushar-state-store \

--region us-east-1

**Step6**:

Creating your first cluster

Prepare local environment

export NAME=myfirstcluster.example.com

export KOPS\_STATE\_STORE=s3://prefix-example-com-state-store

**Step7**: Create cluster configuration

aws ec2 describe-availability-zones --region us-west-2

kops create cluster \

--zones us-west-2a

kops create cluster --zones us-east-1a,us-east-1b,us-east-1c,us-east-1d,us-east-1e,us-east-1f ${NAME}

**step8**: generating ssh public private keypair

ssh-keygen -b 2048 -t rsa -f ~/.ssh/id\_rsa

**step9**: set public key to cluster

kops create secret --name ${NAME} sshpublickey admin -i ~/.ssh/id\_rsa.pub

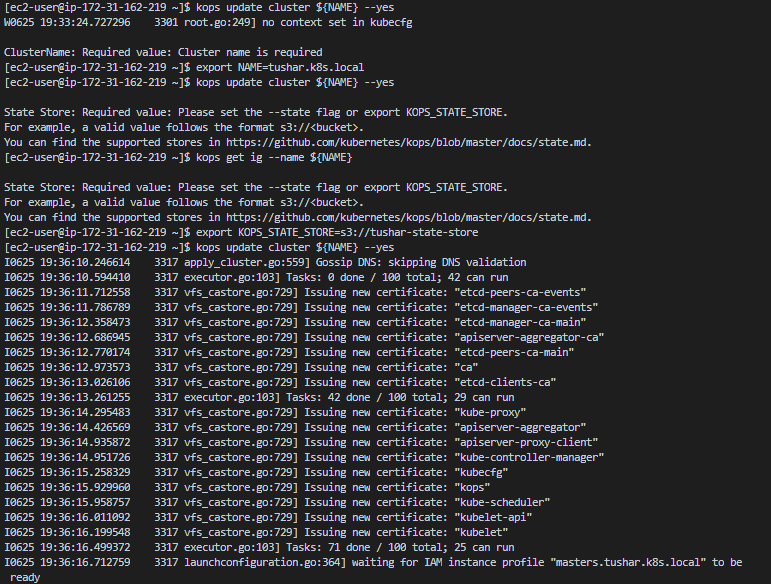
**step10**: kops edit cluster ${NAME}

**step11**: change the cluster size and

kops edit ig nodes --name ${NAME}

**step12**: make resources by kops

kops update cluster ${NAME} –yes

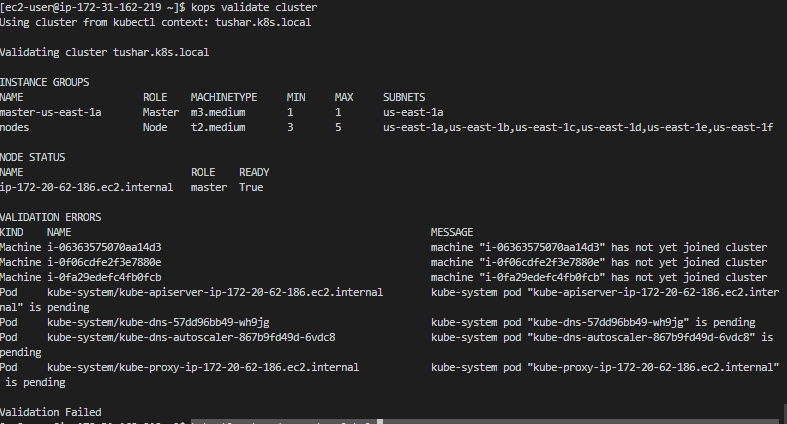


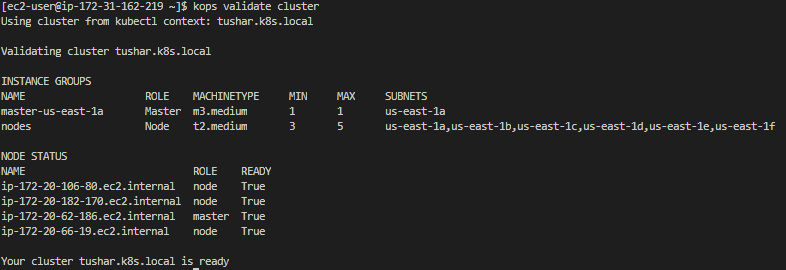
**Step13:** validate the cluster, keep checking with these commands if the cluster is up.

kops validate cluster

kubectl get nodes --show-labels

cluster up commands.





kubectl