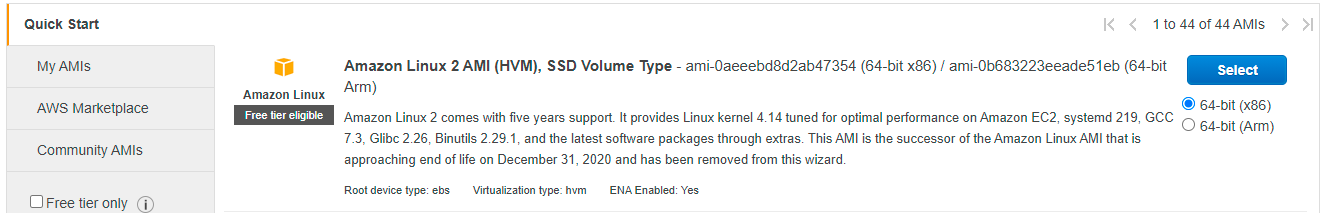
# Kubernetes Cluster Setup

## Step 1: Launching an EC2 Instance

1. First, you need to launch an EC2 instance that will be bastion host for accessing the kubernetes cluster. When asked to select the AMI, select the Amazon Linux 2 AMI.



1. The instance type should be t2.medium. When asked to select VPC for this instance, select the default VPC associated with your account.

Table

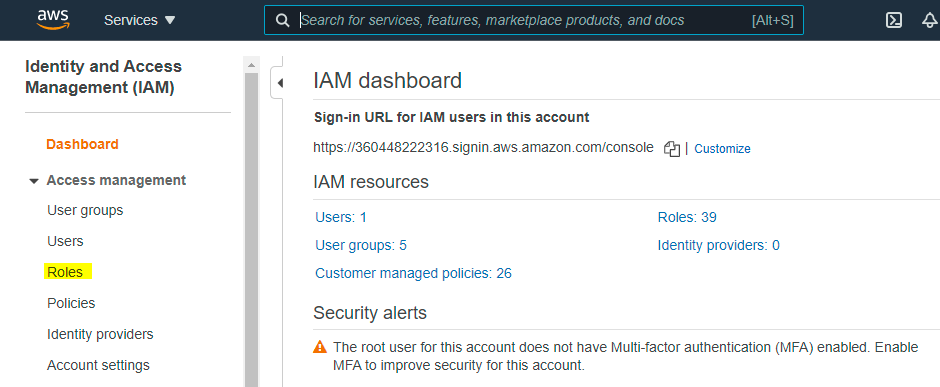
Description automatically generated

## Step 2: Creating and Attaching a Role to the EC2 Instance

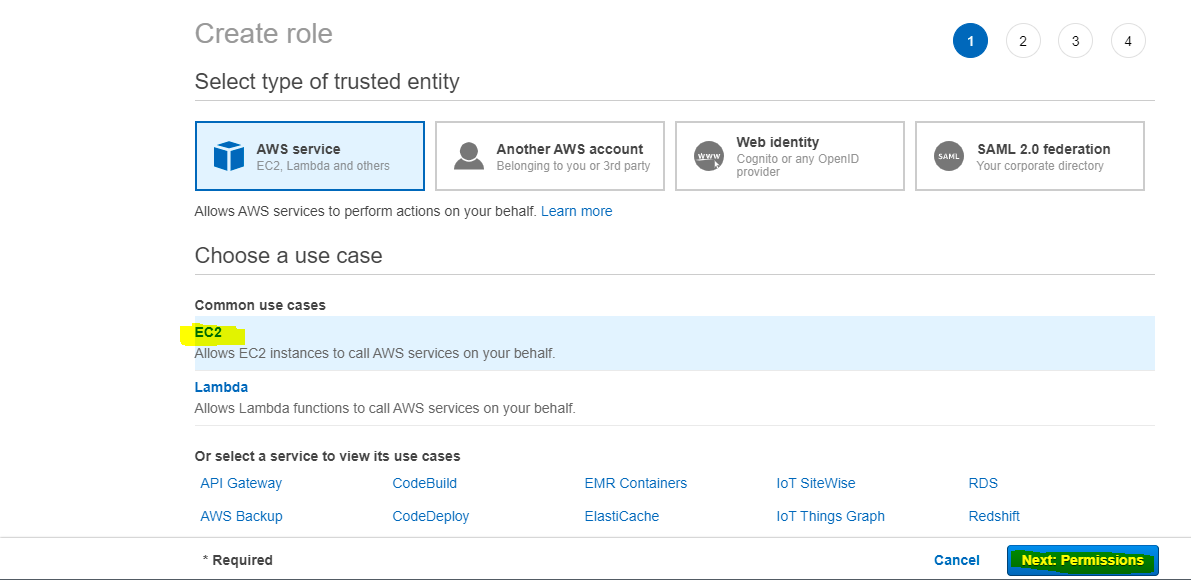
You need to create and attach the relevant policies through the IAM role to EC2 Instance. Kops need permissions to access S3, EC2, VPC, Route53, Autoscaling, etc.

(**Note:** You can directly attach the AdministratorAccess policy to the role)

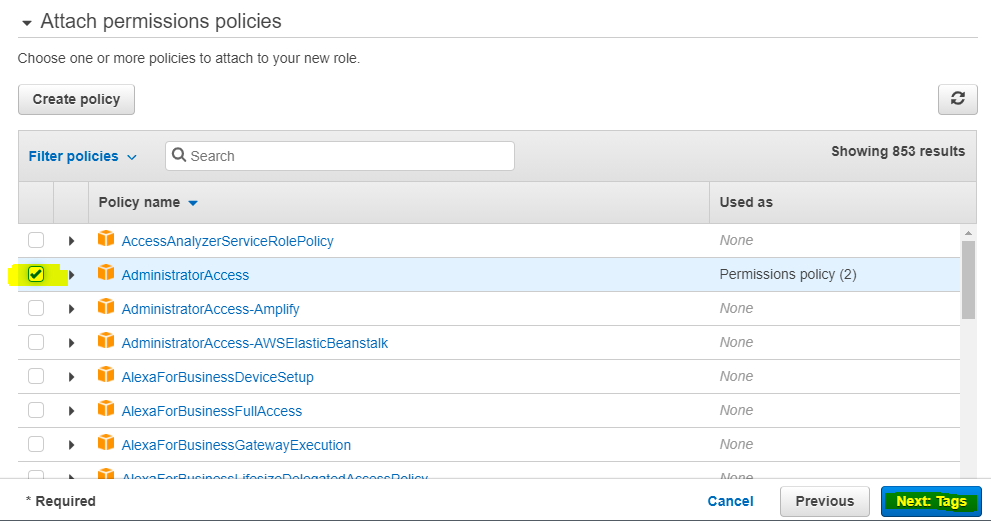
1. To create an IAM role search for IAM in the services and go to the IAM policy page and click on the roles.



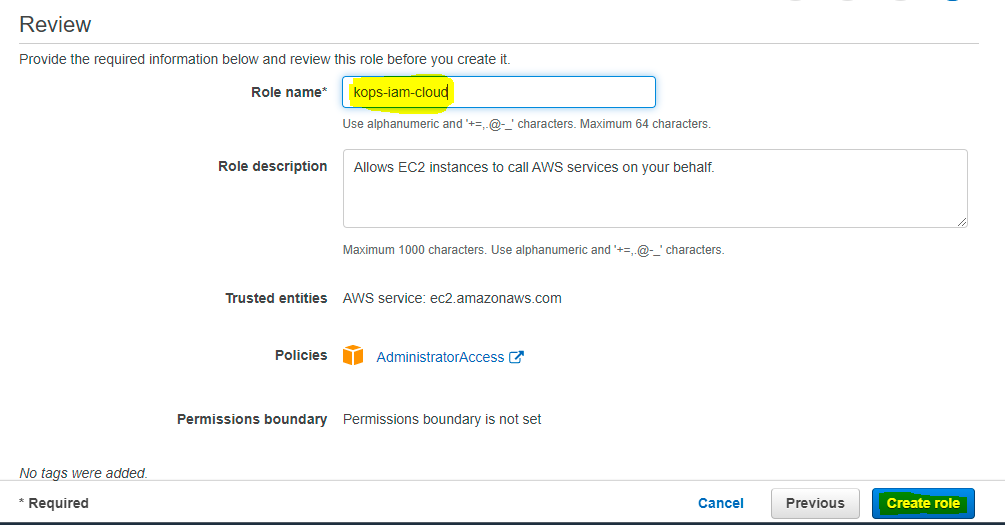
1. Click on **Create Role**. When prompted to select the use case, click on **EC2** and then click on **Next: Permissions**.



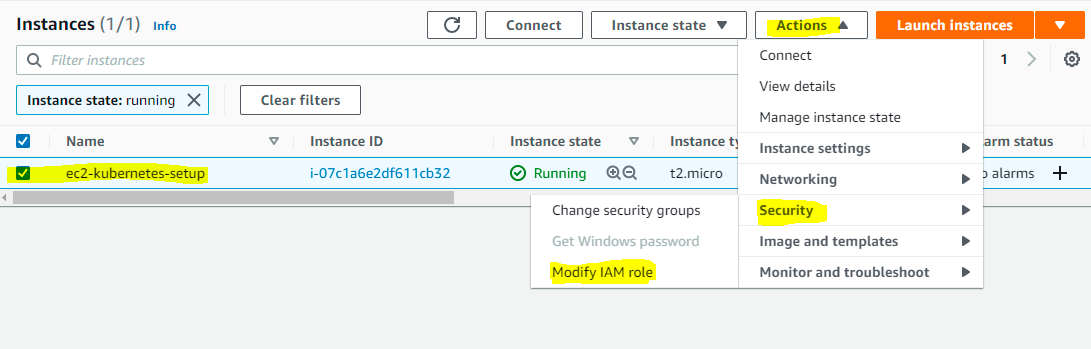
1. Here, select the **AdministratorAccess** policy. Then click on **Next:Tags**. If you want to add a tag, you can add it. Then click on **Next: Review**.



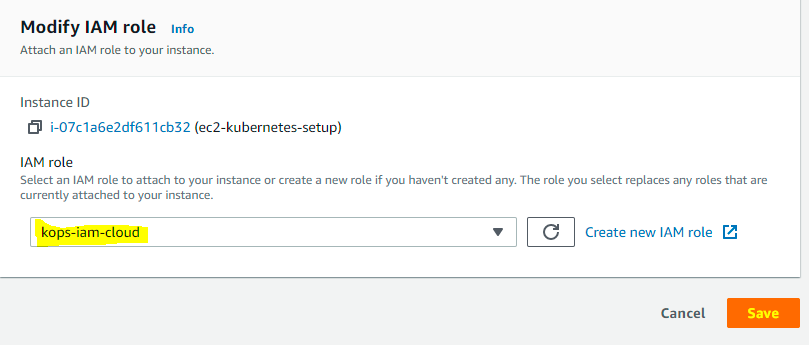
1. Here, add the **Role name** as kops-iam-cloud and then click on **Create Role**.



1. Now, you need to attach this role to the EC2 instance that you launched in the previous step. For this, select the EC2 instance and click on **Actions** and then click on **Security** and then finally click on **Modify IAM role**.



1. Here, you will need to search for the IAM role that you have created. Select that role and then click on **Save**.



## Step 3: Install Kops on EC2

In this step, you will install Kops on the EC2 instance that you launched in the previous step. First you need to login to the EC2 instance. Once done, follow the steps given below to install Kops.

1. To download the Kops binaries enter the following command:

|  |
| --- |
| 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 |

1. Execute the permission for the binaries:

|  |
| --- |
| chmod +x kops |

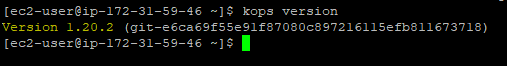
1. Move the binary to usr/local/bin. The command for the same is:

|  |
| --- |
| sudo mv kops /usr/local/bin/kops |

1. You can check if the installation was successful or not by using the following command:

|  |
| --- |
| kops version |

You will get something as shown below.



## Step 4: Install kubectl on EC2

1. To download the kubectl binaries enter the following command:

|  |
| --- |
| 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 |

1. Execute permissions for binary

|  |
| --- |
| chmod +x ./kubectl |

1. Move the binary to usr/local/bin. The command for the same is:

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

## Step 5: Create S3 Bucket In AWS

The S3 bucket is used by kubernetes to persist cluster state.

**Note: You need to make sure that the name that you choose for the bucket is unique across all AWS accounts otherwise you will not be able to create the bucket using the command given below.**

The command to create the S3 bucket is as follows:

|  |
| --- |
| aws s3 mb s3://<nameOfYourBucket> --region us-east-1 |

So, if you wish to create a bucket named kops-setup-s3-bucket, the command for the same will be as follows:

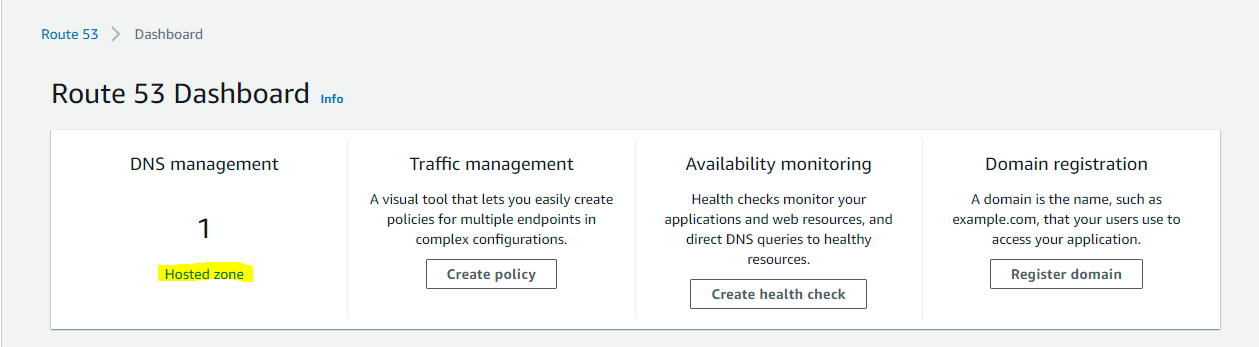
|  |
| --- |
| aws s3 mb s3://kops-setup-s3-bucket --region us-east-1 |

All the letters in the name of the bucket need to be in small letters.

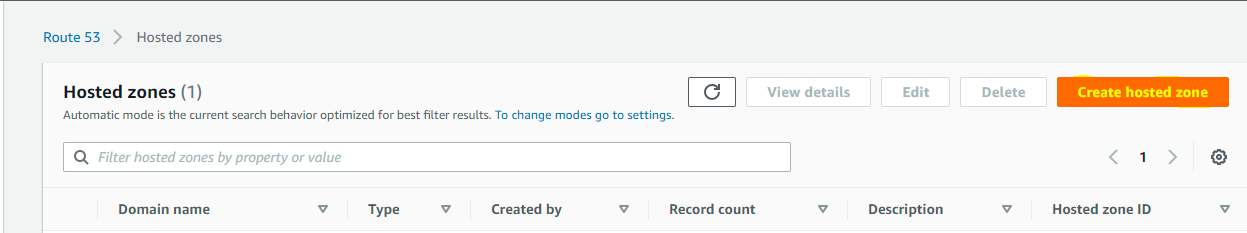
## Step 6: Create a Private Hosted Zone

To create a private hosted zone, you will need to search for the service **Route53**. Follow the steps below to create a hosted zone.

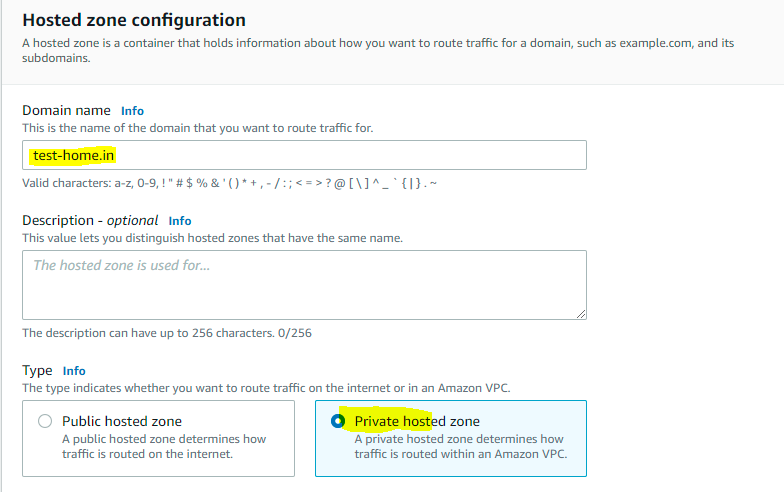
1. Search for the **Route53** service and head over to it. Once you are on the dashboard for this service, click on **Hosted Zone** to create a hosted zone.



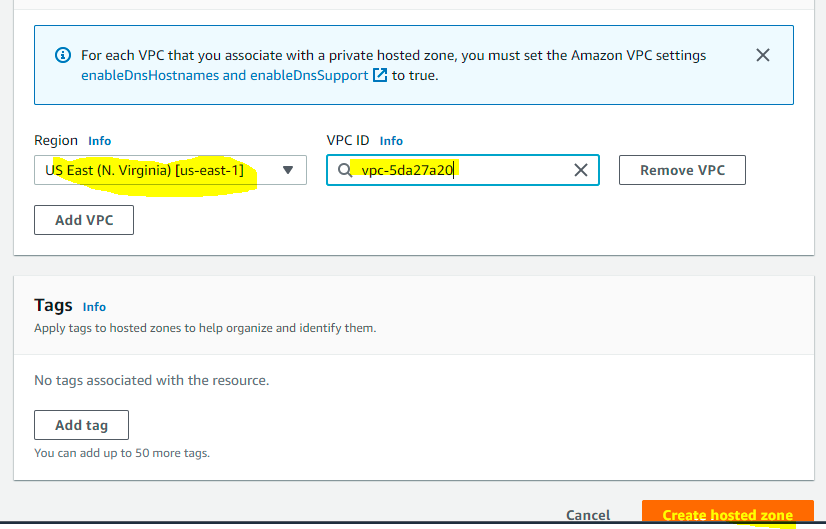
1. On the next page, click on **Create hosted zone** to create a hosted zone.



1. You can choose a name for the hosted zone. For this document, the name selected is **test-home.in**. Select the type as **Private hosted zone**.



1. Next, you need to select the VPC. Select the region as us-east-1 and VPC as the default VPC in that region and then click on **Create hosted zone**.



Now, you have successfully created the Hosted Zone.

## Step 7: Configure Environment Variables

Now, you need to configure the environment variables. Follow these steps to configure them.

1. Open **.bashrc** file. The command for the same is given below:

|  |
| --- |
| vi ~/.bashrc |

Add the following content into .bashrc.

|  |
| --- |
| export KOPS\_CLUSTER\_NAME=test-home.in export KOPS\_STATE\_STORE=s3://kops-setup-s3-bucket |

**Note:**  Make sure that the bucket name is the same as the one that you created in **Step 5**. The cluster name should be the same as the name of the Hosted zone that you created in **Step 6**.

Next, you need to run the following command to reflect the variables added to .bashrc.

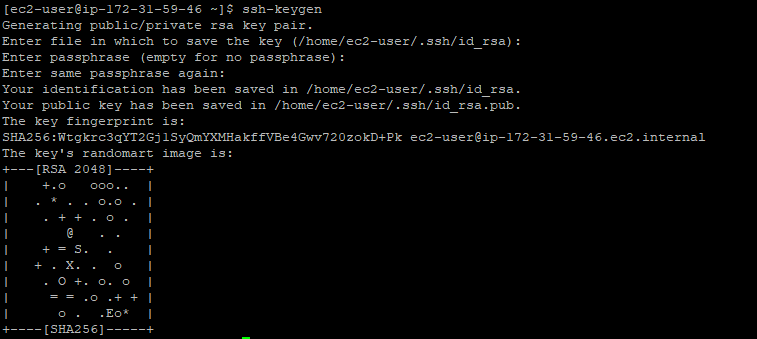
|  |
| --- |
| source ~/.bashrc |

## Step 8: Create SSH Key Pair

The key pair that you will be creating will be used to ssh into the Kubernetes cluster. Enter the following command to create the keypair:

|  |
| --- |
| ssh-keygen |

When you run this it will ask you to enter a few details such as the file where you want to save and the passphrase. Press enter each time it asks you to enter something. Once done it will return something as shown below:



## Step 9: Create a Kubernetes Cluster Definition

Now, you need to create a Kubernetes cluster definition using the following command:

|  |
| --- |
| kops create cluster --state=${KOPS\_STATE\_STORE} --node-count=3 --master-size=t2.medium --node-size=t2.medium --zones=us-east-1b --name=${KOPS\_CLUSTER\_NAME} --dns private --master-count 1 |

This will create the configuration for the cluster.

## Step 10: Create the Cluster

Now to create the cluster use the following command:

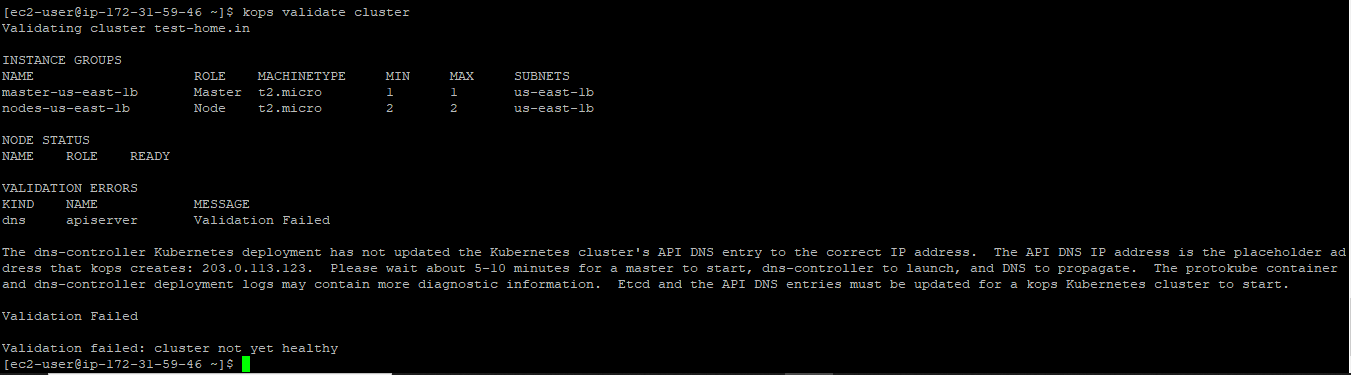
|  |
| --- |
| kops update cluster --name test-home.in --yes --admin |

Above command may take some time to create the required infrastructure resources on AWS. Execute the validate command to check its status and wait until the cluster becomes ready.

|  |
| --- |
| kops validate cluster |

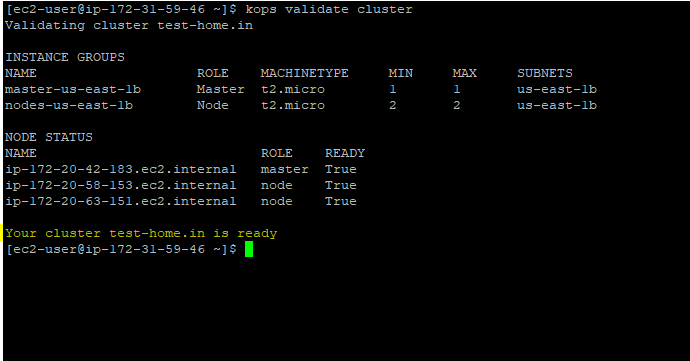
For the above command, you might see validation failed errors initially when you create a

cluster and it is an expected behaviour.



This means you will have to wait for some time as the cluster is still in its formation stage and you can try and run this command again after sometime to test if the cluster is ready or not.

you can check the health of the cluster by using the command **kops validate cluster**. If you get the message: **Your cluster test-home.in is ready** that means the cluster is successfully setup.



If you do not get the above message, you might need to wait for 5-10 minutes so that the cluster is in ready state.

## Step 11: Stop/Delete Cluster

After using cluster for deployment you would either want to terminate the cluster instances i.e. master along with nodes without deleting the cluster or would want to delete the cluster entirely. below are the steps for the same.

Stop the cluster

Steps to shutdown the cluster nodes, wherein “**kops get ig”** command needs to be used to find out the instance groups and then “**kops edit ig <igname>”** command needs to be used to set both minSize and maxSize as zero (Press :wq to save the instance group after that).update the cluster thereafter and perform rolling update on cluster.

|  |
| --- |
| [ec2-user@ip-172-31-17-90 helm]$ **kops get ig**  NAME ROLE MACHINETYPE MIN MAX ZONES  master-us-east-1b Master t2.medium 1 1 us-east-1b  nodes-us-east-1b Node t2.medium 3 3 us-east-1b  [ec2-user@ip-172-31-17-90 helm]$  [ec2-user@ip-172-31-17-90 helm]$ **kops edit ig master-us-east-1b**  apiVersion: kops.k8s.io/v1alpha2  kind: InstanceGroup  metadata:  creationTimestamp: "2021-06-23T20:56:55Z"  generation: 8  labels:  kops.k8s.io/cluster: test-cluster.in  name: master-us-east-1b  spec:  image: 099720109477/ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-20210415  machineType: t2.medium  **maxSize: 0**  **minSize: 0**  nodeLabels:  kops.k8s.io/instancegroup: master-us-east-1b  role: Master  subnets:  - us-east-1b  [ec2-user@ip-172-31-17-90 helm]$  [ec2-user@ip-172-31-17-90 helm]$  [ec2-user@ip-172-31-17-90 helm]$ **kops edit ig nodes-us-east-1b**  apiVersion: kops.k8s.io/v1alpha2  kind: InstanceGroup  metadata:  creationTimestamp: "2021-06-23T20:56:55Z"  generation: 10  labels:  kops.k8s.io/cluster: test-cluster.in  name: nodes-us-east-1b  spec:  image: 099720109477/ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-20210415  machineType: t2.medium  **maxSize: 0**  **minSize: 0**  nodeLabels:  kops.k8s.io/instancegroup: nodes-us-east-1b  role: Node  subnets:  - us-east-1b  [ec2-user@ip-172-31-17-90 helm]$ **kops update cluster --yes**  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  A new kubernetes version is available: 1.20.8  Upgrading is recommended (try kops upgrade cluster)  More information: https://github.com/kubernetes/kops/blob/master/permalinks/upgrade\_k8s.md#1.20.8  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  I0626 09:39:40.461543 24911 dns.go:97] Private DNS: skipping DNS validation  I0626 09:39:40.740762 24911 executor.go:111] Tasks: 0 done / 77 total; 42 can run  I0626 09:39:41.130299 24911 executor.go:111] Tasks: 42 done / 77 total; 15 can run  I0626 09:39:41.398949 24911 executor.go:111] Tasks: 57 done / 77 total; 18 can run  I0626 09:39:43.761151 24911 executor.go:111] Tasks: 75 done / 77 total; 2 can run  I0626 09:39:44.054990 24911 executor.go:111] Tasks: 77 done / 77 total; 0 can run  I0626 09:39:44.055053 24911 dns.go:157] Pre-creating DNS records  I0626 09:39:44.334379 24911 update\_cluster.go:313] Exporting kubecfg for cluster  W0626 09:39:44.370125 24911 create\_kubecfg.go:91] Did not find API endpoint for gossip hostname; may not be able to reach cluster  kOps has set your kubectl context to test-cluster.in  W0626 09:39:44.405046 24911 update\_cluster.go:337] Exported kubecfg with no user authentication; use --admin, --user or --auth-plugin flags with `kops export kubecfg`  Cluster changes have been applied to the cloud.  [ec2-user@ip-172-31-17-90 helm]$ **kops rolling-update cluster --yes**  [ec2-user@ip-172-31-17-90 helm]$ **kops rolling-update cluster --cloudonly --force --yes** |

Start the stopped cluster

Change minSize and maxSize to 1 in master instance group and In node instance group change

minSize and maxSize to 3 as shown below. update the cluster thereafter and perform rolling update on cluster.use “kops validate cluster” command to verify that cluster is ready to be used for deployment. Initally for 5-10 minutes “kops validate cluster” command will show validation errors so wait for 5-10 minutes and then again use command “kops validate cluster” command to verify that cluster is in ready state.

|  |
| --- |
| [ec2-user@ip-172-31-17-90 helm]$ **kops edit ig nodes-us-east-1b**  apiVersion: kops.k8s.io/v1alpha2  kind: InstanceGroup  metadata:  creationTimestamp: "2021-06-23T20:56:55Z"  generation: 16  labels:  kops.k8s.io/cluster: test-cluster.in  name: nodes-us-east-1b  spec:  image: 099720109477/ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-20210415  machineType: t2.medium  **maxSize: 3**  **minSize: 3**  nodeLabels:  kops.k8s.io/instancegroup: nodes-us-east-1b  role: Node  subnets:  - us-east-1b  [ec2-user@ip-172-31-17-90 helm]$ **kops edit ig master-us-east-1b**  apiVersion: kops.k8s.io/v1alpha2  kind: InstanceGroup  metadata:  creationTimestamp: "2021-06-23T20:56:55Z"  generation: 16  labels:  kops.k8s.io/cluster: test-cluster.in  name: master-us-east-1b  spec:  image: 099720109477/ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-20210415  machineType: t2.medium  **maxSize: 1**  **minSize: 1**  nodeLabels:  kops.k8s.io/instancegroup: master-us-east-1b  role: Master  subnets:  - us-east-1b  [ec2-user@ip-172-31-17-90 helm]$ **kops update cluster --yes**  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  A new kubernetes version is available: 1.20.8  Upgrading is recommended (try kops upgrade cluster)  More information: https://github.com/kubernetes/kops/blob/master/permalinks/upgrade\_k8s.md#1.20.8  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  I0627 16:38:40.025588 11013 dns.go:97] Private DNS: skipping DNS validation  I0627 16:38:40.329701 11013 executor.go:111] Tasks: 0 done / 77 total; 42 can run  I0627 16:38:40.678854 11013 executor.go:111] Tasks: 42 done / 77 total; 15 can run  I0627 16:38:40.863879 11013 executor.go:111] Tasks: 57 done / 77 total; 18 can run  I0627 16:38:41.070402 11013 executor.go:111] Tasks: 75 done / 77 total; 2 can run  I0627 16:38:41.368939 11013 executor.go:111] Tasks: 77 done / 77 total; 0 can run  I0627 16:38:41.368995 11013 dns.go:157] Pre-creating DNS records  I0627 16:38:41.784902 11013 update\_cluster.go:313] Exporting kubecfg for cluster  W0627 16:38:41.824405 11013 create\_kubecfg.go:91] Did not find API endpoint for gossip hostname; may not be able to reach cluster  kOps has set your kubectl context to test-cluster.in  W0627 16:38:41.877696 11013 update\_cluster.go:337] Exported kubecfg with no user authentication; use --admin, --user or --auth-plugin flags with `kops export kubecfg`  Cluster changes have been applied to the cloud.  Changes may require instances to restart: kops rolling-update cluster  [ec2-user@ip-172-31-17-90 helm]$ kops update cluster --yes  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  A new kubernetes version is available: 1.20.8  Upgrading is recommended (try kops upgrade cluster)  More information: https://github.com/kubernetes/kops/blob/master/permalinks/upgrade\_k8s.md#1.20.8  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  I0627 16:38:40.025588 11013 dns.go:97] Private DNS: skipping DNS validation  I0627 16:38:40.329701 11013 executor.go:111] Tasks: 0 done / 77 total; 42 can run  I0627 16:38:40.678854 11013 executor.go:111] Tasks: 42 done / 77 total; 15 can run  I0627 16:38:40.863879 11013 executor.go:111] Tasks: 57 done / 77 total; 18 can run  I0627 16:38:41.070402 11013 executor.go:111] Tasks: 75 done / 77 total; 2 can run  I0627 16:38:41.368939 11013 executor.go:111] Tasks: 77 done / 77 total; 0 can run  I0627 16:38:41.368995 11013 dns.go:157] Pre-creating DNS records  I0627 16:38:41.784902 11013 update\_cluster.go:313] Exporting kubecfg for cluster  W0627 16:38:41.824405 11013 create\_kubecfg.go:91] Did not find API endpoint for gossip hostname; may not be able to reach cluster  kOps has set your kubectl context to test-cluster.in  W0627 16:38:41.877696 11013 update\_cluster.go:337] Exported kubecfg with no user authentication; use --admin, --user or --auth-plugin flags with `kops export kubecfg`  Cluster changes have been applied to the cloud.  Changes may require instances to restart: kops rolling-update cluster  [ec2-user@ip-172-31-17-90 helm]$ **kops rolling-update cluster --yes**  Unable to reach the kubernetes API.  Use --cloudonly to do a rolling-update without confirming progress with the k8s API  error listing nodes in cluster: Get "https://api.test-cluster.in/api/v1/nodes": dial tcp 54.210.201.245:443: i/o timeout  [ec2-user@ip-172-31-17-90 helm]$  [ec2-user@ip-172-31-17-90 helm]$  [ec2-user@ip-172-31-17-90 helm]$ **kops rolling-update cluster --cloudonly --force --yes**  NAME STATUS NEEDUPDATE READY MIN TARGET MAX  master-us-east-1b Ready 0 1 1 1 1  nodes-us-east-1b Ready 0 3 3 3 3  W0627 16:39:47.379961 11097 instancegroups.go:449] Not validating cluster as cloudonly flag is set.  W0627 16:39:47.380033 11097 instancegroups.go:375] Not draining cluster nodes as 'cloudonly' flag is set.  I0627 16:39:47.380044 11097 instancegroups.go:575] Stopping instance "i-082d102a99001000b", in group "master-us-east-1b.masters.test-cluster.in" (this may take a while).  I0627 16:39:47.569481 11097 instancegroups.go:417] waiting for 15s after terminating instance  W0627 16:40:02.569731 11097 instancegroups.go:449] Not validating cluster as cloudonly flag is set.  W0627 16:40:02.569770 11097 instancegroups.go:449] Not validating cluster as cloudonly flag is set.  W0627 16:40:02.569804 11097 instancegroups.go:375] Not draining cluster nodes as 'cloudonly' flag is set.  I0627 16:40:02.569815 11097 instancegroups.go:575] Stopping instance "i-07cbc46f22f903fdc", in group "nodes-us-east-1b.test-cluster.in" (this may take a while).  I0627 16:40:02.791395 11097 instancegroups.go:417] waiting for 15s after terminating instance  W0627 16:40:17.791637 11097 instancegroups.go:449] Not validating cluster as cloudonly flag is set.  W0627 16:40:17.791692 11097 instancegroups.go:375] Not draining cluster nodes as 'cloudonly' flag is set.  I0627 16:40:17.791705 11097 instancegroups.go:575] Stopping instance "i-081eff87d0bfbdff6", in group "nodes-us-east-1b.test-cluster.in" (this may take a while).  I0627 16:40:17.978346 11097 instancegroups.go:417] waiting for 15s after terminating instance  W0627 16:40:32.978547 11097 instancegroups.go:449] Not validating cluster as cloudonly flag is set.  W0627 16:40:32.978605 11097 instancegroups.go:375] Not draining cluster nodes as 'cloudonly' flag is set.  I0627 16:40:32.978632 11097 instancegroups.go:575] Stopping instance "i-0ce2b578ab566d6d1", in group "nodes-us-east-1b.test-cluster.in" (this may take a while).  I0627 16:40:33.156361 11097 instancegroups.go:417] waiting for 15s after terminating instance  W0627 16:40:48.156605 11097 instancegroups.go:449] Not validating cluster as cloudonly flag is set.  I0627 16:40:48.156645 11097 rollingupdate.go:190] Rolling update completed for cluster "test-cluster.in"!  [ec2-user@ip-172-31-17-90 helm]$ **kops export kubecfg --admin**  W0627 16:42:18.290011 11266 create\_kubecfg.go:91] Did not find API endpoint for gossip hostname; may not be able to reach cluster  kOps has set your kubectl context to test-cluster.in  [ec2-user@ip-172-31-17-90 helm]$ **kops validate cluster**  Validating cluster test-cluster.in  Validation failed: unexpected error during validation: error listing nodes: Get "https://api.test-cluster.in/api/v1/nodes": dial tcp 54.210.201.245:443: i/o timeout  [ec2-user@ip-172-31-17-90 helm]$ **kops validate cluster**  Validating cluster test-cluster.in  Validation failed: unexpected error during validation: error listing nodes: Get "https://api.test-cluster.in/api/v1/nodes": dial tcp 54.210.201.245:443: i/o timeout  [ec2-user@ip-172-31-17-90 helm]$ kops validate cluster  Validating cluster test-cluster.in  INSTANCE GROUPS  NAME ROLE MACHINETYPE MIN MAX SUBNETS  master-us-east-1b Master t2.medium 1 1 us-east-1b  nodes-us-east-1b Node t2.medium 3 3 us-east-1b  NODE STATUS  NAME ROLE READY  ip-172-20-52-58.ec2.internal master True  ip-172-20-61-151.ec2.internal node True  VALIDATION ERRORS  KIND NAME MESSAGE  Machine i-03b0fe517670aaa76 machine "i-03b0fe517670aaa76" has not yet joined cluster  Machine i-098ec033f9328ac9b machine "i-098ec033f9328ac9b" has not yet joined cluster  Pod kube-system/coredns-5489b75945-2dw9m system-cluster-critical pod "coredns-5489b75945-2dw9m" is not ready (coredns)  Pod kube-system/coredns-5489b75945-wz8nl system-cluster-critical pod "coredns-5489b75945-wz8nl" is pending  Pod kube-system/coredns-autoscaler-6f594f4c58-lfwp8 system-cluster-critical pod "coredns-autoscaler-6f594f4c58-lfwp8" is pending  Pod kube-system/metrics-server-766c9b8df-2zncg system-cluster-critical pod "metrics-server-766c9b8df-2zncg" is pending  Validation Failed  Validation failed: cluster not yet healthy  [ec2-user@ip-172-31-17-90 helm]$ **kops validate cluster**  Validating cluster test-cluster.in  INSTANCE GROUPS  NAME ROLE MACHINETYPE MIN MAX SUBNETS  master-us-east-1b Master t2.medium 1 1 us-east-1b  nodes-us-east-1b Node t2.medium 3 3 us-east-1b  NODE STATUS  NAME ROLE READY  ip-172-20-52-58.ec2.internal master True  ip-172-20-54-234.ec2.internal node True  ip-172-20-59-166.ec2.internal node True  ip-172-20-61-151.ec2.internal node True  Your cluster test-cluster.in is ready |

Delete the cluster

Alternatively, you can use following command to delete the cluster.

|  |
| --- |
| kops delete cluster --yes |