

In pervious document, I have made a cluster already.

Now to bring up the already setup cluster:

```
kops update cluster --name=${KOPS_CLUSTER_NAME} --yes
```

```
[ec2-user@ip-172-31-84-37 ~]$ kops update cluster --name=${KOPS_CLUSTER_NAME} --yes

*****

A new kubernetes version is available: 1.10.6
Upgrading is recommended (try kops upgrade cluster)

More information: https://github.com/kubernetes/kops/blob/master/permalinks/upgrade\_k8s.md#1.10.6

*****

I1124 07:31:38.281021 29233 apply_cluster.go:505] Gossip DNS: skipping DNS validation
I1124 07:31:38.546735 29233 executor.go:103] Tasks: 0 done / 77 total; 30 can run
I1124 07:31:38.897018 29233 executor.go:103] Tasks: 30 done / 77 total; 24 can run
I1124 07:31:39.073711 29233 executor.go:103] Tasks: 54 done / 77 total; 19 can run
I1124 07:31:39.692504 29233 executor.go:103] Tasks: 73 done / 77 total; 3 can run
I1124 07:31:40.258397 29233 executor.go:103] Tasks: 76 done / 77 total; 1 can run
I1124 07:31:40.459641 29233 executor.go:103] Tasks: 77 done / 77 total; 0 can run
I1124 07:31:40.935036 29233 update_cluster.go:290] Exporting kubecfg for cluster
kops has set your kubectl context to masharma.k8s.local

Cluster changes have been applied to the cloud.

Changes may require instances to restart: kops rolling-update cluster
```

Here you see, one master and two slave nodes have come up. Reason being, at the time of creation, I kept the configurations like:

```
--node-count=2 \
--node-size=t2.medium \
--zones=us-east-1a \
```

Launch Instance

▼

Connect

Actions

▼

Filter by tags and attributes or search by keyword

<input type="checkbox"/>	Name ▼	Instance ID ▼	Instance Type ▼	Availability Zone ▼	Instance State ▼
<input type="checkbox"/>		i-0e3753e44b49932...	t2.micro	us-east-1a	● stopped
<input type="checkbox"/>	nodes.masharma.k8s.local	i-0c0e45ab54b46943f	t2.medium	us-east-1a	● running
<input checked="" type="checkbox"/>	JenkinsMaster	i-0b2f9919b9a489b21	t2.micro	us-east-1d	● running
<input type="checkbox"/>	nodes.masharma.k8s.local	i-0ac71bb0106b5e39a	t2.medium	us-east-1a	● running
<input type="checkbox"/>	Jenkins Slave	i-05394f5aa9f161763	t2.micro	us-east-1a	● stopped
<input type="checkbox"/>	master-us-east-1a.masters.masharm...	i-00a90a28ae78c93f0	m3.medium	us-east-1a	● running

Then I tried validating the cluster and it failed:

kops validate cluster

```
[ec2-user@ip-172-31-84-37 ~]$ kops validate cluster
Validating cluster masharma.k8s.local

unexpected error during validation: error listing nodes: Get https://api-masharma-k8s-local-n8ppc6-710993594.us-east-1.elb.amazonaws.com/api/v1/nodes: EOF
```

Now let me troubleshoot it:

My thought process:

Find Kops and kubectl versions:

```
[ec2-user@ip-172-31-84-37 ~]$ kops version
Version 1.10.0 (git-8b52ea6d1)
[ec2-user@ip-172-31-84-37 ~]$ kubectl version
Client Version: version.Info{Major:"1", Minor:"5", GitVersion:"v1.5.2", GitCommit:"08e099554f3c31f6e6f07b448ab3ed78d0520507", GitTreeState:"clean", BuildDate:"2017-01-12T04:57:25Z", GoVersion:"go1.7.4", Compiler:"gc", Platform:"linux/amd64"}
Unable to connect to the server: EOF
```

After some debugging I got to know my S3 bucket that saves kops state is not public.

So I made it public:

+ Create bucket

Edit public access settings

Empty

Delete

<input type="checkbox"/>	Bucket name	Access
<input type="checkbox"/>	aws-opsworks-cm-opswork1-7aoa7saocrf	Objects can be public
<input checked="" type="checkbox"/>	manish-kops-state-store	Public

Again kops validation failed with new error this time:

```
[ec2-user@ip-172-31-84-37 ~]$ kops validate cluster
Validating cluster masharma.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    2    2    us-east-1a

NODE STATUS
NAME                                     ROLE    READY
ip-172-20-61-129.ec2.internal          master  True

VALIDATION ERRORS
KIND    NAME                                     MESSAGE
Machine i-0c23a535890da93d8             machine "i-0c23a535890da93d8" has not yet joined cluster
Pod     kube-system/dns-controller-6d6b7f78b-g49br  kube-system pod "dns-controller-6d6b7f78b-g49br" is not healthy

Validation Failed
```

This says one of the node is not healthy.

I know the reason It is because in AWS autoscaling, I have set node value as 1 (to save money). Whereas while defining the cluster, I set node values as 2.

Edit details - nodes.masharma.k8s.local

Launch Instances Using

Launch Template

Launch Configuration

Launch Configuration

nodes.masharma.k8s.local-20180818120547

Desired Capacity

1

Min

1

Max

1

Availability Zone(s)

us-east-1a

Subnet(s)

subnet-03e70c002b40b0e04(172.20.32.0/19) | us-east-1a.masharma.k8s.local | us-east-1a

Classic Load Balancers

Target Groups

Health Check Type

EC2

Health Check Grace Period

0

Instance Protection

Termination Policies

Default

Suspended Processes

Cancel

Save

So, I made the configuration of auto-scale group equal to the original config.

And validation passed

```
[ec2-user@ip-172-31-84-37 ~]$ kops validate cluster
Validating cluster masharma.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	2	2	us-east-1a

NODE STATUS

NAME	ROLE	READY
ip-172-20-40-34.ec2.internal	node	True
ip-172-20-54-5.ec2.internal	master	True

Your cluster masharma.k8s.local is ready

Now my cluster is ready.