

ASSESSMENT - 18

EKS - 1

**TO
THE
NEW**



1. Create eks cluster using eksctl

During creation, Specify

- Cluster name
- Kubernetes version
- Control plane role
- Subnets for Control Plane
- Control Plane security Group
- Add tag: owner, purpose on Control Plane
- Node Group Name
- Node Instance Role
- Subnets for Node Group
- Node Instance SSH key pair
- Node Instance Security Group
- Node Instance Instance Type
- Node Instance Disk
- Add tag: owner, purpose on Node Group
- Node Group Size: min, max

Create Security Group

✕

Security group name ⓘ

Description ⓘ

VPC ⓘ

security group rules:

Inbound

Outbound

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
SSH ▾	TCP	22	Custom ▾ CIDR, IP or Security Group	e.g. SSH for Admin De
HTTP ▾	TCP	80	Custom ▾ 0.0.0.0/0, ::/0	e.g. SSH for Admin De

Add Rule

```
garima@garima:~$ eksctl version
[i] version.Info{BuiltAt:"", GitCommit:"", GitTag:"0.13.0"}
garima@garima:~$ vim garima.yaml
```

```
apiVersion: eksctl.io/v1alpha5
kind: ClusterConfig
metadata:
  name: my-test
  region: us-east-1
vpc:
  id: "vpc-0af0018a947f6e4b3"
  cidr: "192.168.0.0/16"
  subnets:
    public:
      us-east-1c:
        id: "subnet-055976fde57060ff0"
        cidr: "192.168.192.0/18"
      us-east-1b:
        id: "subnet-0c0e750b3f41ba157"
        cidr: "192.168.128.0/18"
      us-east-1a:
        id: "subnet-0549ccd892830c1ab"
        cidr: "192.168.64.0/18"
iam:
  serviceRoleARN: "arn:aws:iam::044650439222:role/eks_role_garima"
```

```
managedNodeGroups:
- name: managed-ng-1
  instanceType: m5.large
  minSize: 2
  desiredCapacity: 3
  maxSize: 4
  availabilityZones: ["us-east-1a","us-east-1b","us-east-1c"]
  volumeSize: 20
  securityGroups:
    withShared: true
    withLocal: true
    attachIDs: ['sg-07347f027015cae61']
  ssh:
    allow: true
    publicKeyName: 'newawskeypair'
  tags:
    'owner': 'garima'
  iam:
    instanceProfileARN: "arn:aws:iam::044650439222:instance-profile/worker_node_garima"
```

```

garima@garima:~$ eksctl create cluster -f garima.yaml
Error: loading config file "garima.yaml": error unmarshaling JSON: while decoding JSON: json: unknown field "securityGroups"
garima@garima:~$ vim garima.yaml
garima@garima:~$ eksctl create cluster -f garima.yaml
[i] eksctl version 0.13.0
[i] using region us-east-1
=^[===== [!] retryable error (RequestError: send request failed
caused by: Post https://ec2.us-east-1.amazonaws.com/: net/http: TLS handshake timeout) from ec2/DescribeVpcs - will retry aft
078ms
[✓] using existing VPC (vpc-0af0018a947f6e4b3) and subnets (private:[ ] public:[subnet-0549ccd892830c1ab subnet-0c0e750b3f41b
fde57060ff0])
[!] custom VPC/subnets will be used; if resulting cluster doesn't function as expected, make sure to review the configuratio
[i] nodegroup "managed-ng-1" will use "ami-087a82f6b78a07557" [AmazonLinux2/1.14]
[i] using EC2 key pair "newawskeypair"
[i] using Kubernetes version 1.14
[i] creating EKS cluster "my-test" in "us-east-1" region with un-managed nodes
[i] 1 nodegroup (managed-ng-1) was included (based on the include/exclude rules)
[i] will create a CloudFormation stack for cluster itself and 1 nodegroup stack(s)
[i] will create a CloudFormation stack for cluster itself and 0 managed nodegroup stack(s)
[i] if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=us-east-1 --clus
[i] CloudWatch logging will not be enabled for cluster "my-test" in "us-east-1"
[i] you can enable it with 'eksctl utils update-cluster-logging --region=us-east-1 --cluster=my-test'
[i] Kubernetes API endpoint access will use default of {publicAccess=true, privateAccess=false} for cluster "my-test" in "us
[i] 2 sequential tasks: { create cluster control plane "my-test", create nodegroup "managed-ng-1" }
[i] building cluster stack "eksctl-my-test-cluster"
[i] deploying stack "eksctl-my-test-cluster"

```

```

[!] retryable error (RequestError: send request failed
caused by: Post https://cloudformation.us-east-1.amazonaws.com/: dial tcp:
er misbehaving) from cloudformation/DescribeStacks - will retry after delay
[!] retryable error (RequestError: send request failed
caused by: Post https://cloudformation.us-east-1.amazonaws.com/: dial tcp:
er misbehaving) from cloudformation/DescribeStacks - will retry after delay
[!] retryable error (RequestError: send request failed
caused by: Post https://cloudformation.us-east-1.amazonaws.com/: dial tcp:
er misbehaving) from cloudformation/DescribeStacks - will retry after delay
[!] retryable error (RequestError: send request failed
caused by: Post https://cloudformation.us-east-1.amazonaws.com/: dial tcp:
er misbehaving) from cloudformation/DescribeStacks - will retry after delay
[!] retryable error (RequestError: send request failed
caused by: Post https://cloudformation.us-east-1.amazonaws.com/: dial tcp:
er misbehaving) from cloudformation/DescribeStacks - will retry after delay
[!] retryable error (RequestError: send request failed
caused by: Post https://cloudformation.us-east-1.amazonaws.com/: dial tcp:
er misbehaving) from cloudformation/DescribeStacks - will retry after delay
[!] retryable error (RequestError: send request failed
caused by: Post https://cloudformation.us-east-1.amazonaws.com/: dial tcp:
er misbehaving) from cloudformation/DescribeStacks - will retry after delay
[!] retryable error (RequestError: send request failed
caused by: Post https://ec2.us-east-1.amazonaws.com/: net/http: TLS handsh
549248ms
[i] building nodegroup stack "eksctl-my-test-nodegroup-managed-ng-1"
[i] deploying stack "eksctl-my-test-nodegroup-managed-ng-1"
[!] retryable error (RequestError: send request failed
caused by: Post https://iam.amazonaws.com/: net/http: TLS handshake timeout
s

```



```

[i] building nodegroup stack "eksctl-my-test-nodegroup-managed-ng-1"
[i] deploying stack "eksctl-my-test-nodegroup-managed-ng-1"
[!] retryable error (RequestError: send request failed
caused by: Post https://iam.amazonaws.com/: net/http: TLS handshake timeout) from iam/GetIns
s
[✓] all EKS cluster resources for "my-test" have been created
[✓] saved kubeconfig as "/home/garima/.kube/config"
[i] adding identity "arn:aws:iam::044650439222:role/worker_node_garima" to auth ConfigMap
[i] nodegroup "managed-ng-1" has 1 node(s)
[i] node "ip-192-168-123-23.ec2.internal" is not ready
[i] waiting for at least 2 node(s) to become ready in "managed-ng-1"
[i] nodegroup "managed-ng-1" has 3 node(s)
[i] node "ip-192-168-123-23.ec2.internal" is ready
[i] node "ip-192-168-143-62.ec2.internal" is not ready
[i] node "ip-192-168-240-135.ec2.internal" is ready
[i] kubectl command should work with "/home/garima/.kube/config", try 'kubectl get nodes'
[✓] EKS cluster "my-test" in "us-east-1" region is ready

```

my-test
Refresh
Delete

General configuration

Kubernetes version 1.14	Platform version eks.9	Status Active
API server endpoint Copy https://C8E12415423905061E2316B042113CA1.gr7.us-east-1.eks.amazonaws.com	Certificate authority Copy LS0tLS1CRUdJTiBDRVJUSUZJQ0FUR50tLS0tCk1JSUN5RENDQWJDZ0F3SUJBZ0lCQURBTklna3Foa2lHOXcwQkFRc0ZBREFTVTJNd0VRWURWUVFERXdwcmlRXS5mWkY201bGRHVnpNQjRFRFRjd01ETXdPVVE0TlRjME5sb	
OpenID Connect provider URL Copy https://oidc.eks.us-east-1.amazonaws.com/oidc/C8E12415423905061E2316B042113CA1		
Cluster ARN Copy arn:aws:eks:us-east-1:044650439222:cluster/my-test	Cluster IAM Role ARN Copy arn:aws:iam::044650439222:role/eks_role_garima	

```

garima@garima:~$ eksctl get cluster
NAME          REGION
garima_cluster us-east-1
my-test       us-east-1
garima@garima:~$ kubectl get nodes
NAME                                STATUS    ROLES    AGE    VERSION
ip-192-168-123-23.ec2.internal      Ready    <none>    28m    v1.14.8-eks-b8860f
ip-192-168-143-62.ec2.internal      Ready    <none>    28m    v1.14.8-eks-b8860f
ip-192-168-240-135.ec2.internal     Ready    <none>    28m    v1.14.8-eks-b8860f
garima@garima:~$

```

2. Authentication Management

a. Add new 2 IAM user into the cluster

```
garima@garima:~$  
garima@garima:~$ kubectl edit -n kube-system configmap/aws-auth
```

```
apiVersion: v1  
data:  
  mapRoles: |  
    - groups:  
      - system:bootstrappers  
      - system:nodes  
      rolearn: arn:aws:iam::044650439222:role/worker_node_garima  
      username: system:node:{{EC2PrivateDNSName}}  
  mapUsers: |  
    groups:  
      - system:bootstrappers  
      - system:nodes  
      rolearn: arn:aws:iam::187632318301:user/diksha.tomar@tothenew.com  
      username: diksha  
kind: ConfigMap  
metadata:  
  creationTimestamp: "2020-03-09T09:43:13Z"  
  name: aws-auth  
  namespace: kube-system  
  resourceVersion: "4107"  
  selfLink: /api/v1/namespaces/kube-system/configmaps/aws-auth  
  uid: 64dfac93-61ea-11ea-92d7-0e893341cf29
```

b. Enable a EC2 server to access Cluster master API without using access/secret key

Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a po

Visual editor

JSON

[Expand all](#) | [Collapse all](#)

▼ Select a service

► Service [Choose a service](#)

Add ARN(s)



Amazon Resource Names (ARNs) uniquely identify AWS resources. Resources are unique to each service. [Learn more](#)

Specify ARN for cluster

[List ARNs manually](#)

arn:aws:eks:us-east-1:044650439222:cluster/my-test

Region *

us-east-1

☐ Any

Account *

044650439222

☐ Any

Cluster name *

my-test

☐ Any

Cancel

Add

Visual editor

JSON

[Expand all](#)
[Collapse all](#)

▼ EKS (9 actions)

▶ Service

EKS

▶ Actions

List

ListClusters

ListFargateProfiles

ListNodegroups

ListTagsForResource

ListUpdates

Read

DescribeCluster

DescribeFargateProfile

DescribeNodegroup

DescribeUpdate

▶ Resources

arn:aws:eks:us-east-1:044650439222:cluster/my-test

arn:aws:eks:*:*:fargateprofile/*/*/*

arn:aws:eks:*:*:nodegroup/*/*/*

Review policy

Name*

new_eksctl_policy

Use alphanumeric and '+=, @-_' characters. Maximum 128 characters.

Description

Maximum 1000 characters. Use alphanumeric and '+=, @-_' characters.

Summary

Q Filter

Service ▼	Access level	Resource	Request condition
Allow (1 of 224 services) Show remaining 223			
EKS	Full: List, Read	Multiple	None

Create role

▼ Attach permissions policies

Choose one or more policies to attach to your new role.

Create policy

Filter policies ▼

	Policy name ▼
<input checked="" type="checkbox"/>	new_eksctl_policy

Review

Provide the required information below and review this role before you create it.

Role name*

new_eksctl_role

Use alphanumeric and '+=, @-_' characters. Maximum 64 characters.

Role description

Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use alphanumeric and '+=, @-_' characters.

Trusted entities AWS service: ec2.amazonaws.com

Policies [new_eksctl_policy](#) 

Now launch an instance and attach this role to that instance.

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of lower prices, or launch on-demand instances. For more information, see [Amazon EC2 Instance Types and Pricing](#).

Number of instances ⓘ [Launch into Auto Scaling Group ⓘ](#)

Purchasing option ⓘ ☐ Request Spot instances

Network ⓘ ⓘ [Create new VPC](#)

Subnet ⓘ ⓘ [Create new subnet](#)
16369 IP Addresses available

Auto-assign Public IP ⓘ ⓘ

Placement group ⓘ ☐ Add instance to placement group

[Launch Instance](#) [Connect](#) [Actions](#)

search : i-04c33364937374f14 ⓘ Add filter

<input type="checkbox"/>	Name	Instance ID	Instance Type	Availability Zone	Instance State
<input type="checkbox"/>	new_instance	i-04c33364937374f14	t2.micro	us-east-1b	running

[Launch Instance](#) [Connect](#) [Actions](#)

search : i-04c33364937374f14 ⓘ

<input type="checkbox"/>	Name	Instance Type	Availability Zone	Instance State
<input type="checkbox"/>	new_instance		us-east-1b	running

Connect

Get Windows Password

Create Template From Instance

Launch More Like This

Instance State

Instance Settings

Image

Networking

Add/Edit Tags

Attach to Auto Scaling Group

Attach/Replace IAM Role

Attach/Replace IAM Role

Select an IAM role to attach to your instance. If you don't have any IAM roles, choose Create new IAM role to create a role in the IAM console. If an IAM role is already attached to your instance, the IAM role you choose will replace the existing role.

Instance ID - ⓘ

IAM role*

new_eksctl_role



Create new IAM role ⓘ

```
garima@garima:~/Downloads$ ssh -i newawskeypair.pem ec2-user@3.233.222.221
Last login: Tue Mar 10 08:33:31 2020 from 122.162.179.136

  _ _ | _ _ | _ _ )
 _ | ( _ _ | _ _ /   Amazon Linux 2 AMI
 _ _ | \ _ _ | _ _ |

https://aws.amazon.com/amazon-linux-2/
1 package(s) needed for security, out of 26 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-192-168-134-172 ~]$ aws eks describe-cluster --name my-test --region us-east-1
{
  "cluster": {
    "status": "ACTIVE",
    "endpoint": "https://C8E12415423905061E2316B042113CA1.gr7.us-east-1.eks.amazonaws.com",
    "logging": {
      "clusterLogging": [
        {
          "enabled": false,
          "types": [
            "api",
            "audit",
            "authenticator",
            "controllerManager",
            "scheduler"
          ]
        }
      ]
    },
    "name": "my-test",
    "tags": {},
    "certificateAuthority": {
```

3. Eksctl command to terminate the stack

```
garima@garima:~$ eksctl delete cluster -f garima.yaml
[i] eksctl version 0.13.0
[i] using region us-east-1
[i] deleting EKS cluster "my-test"
[i] deleted 0 Fargate profile(s)
[✓] kubeconfig has been updated
[i] cleaning up LoadBalancer services
[i] 2 sequential tasks: { delete nodegroup "managed-ng-1", delete cluster control plane "my-test" [async] }
[i] will delete stack "eksctl-my-test-nodegroup-managed-ng-1"
[i] waiting for stack "eksctl-my-test-nodegroup-managed-ng-1" to get deleted
```