|  |
| --- |
| CREDABLE |
| [ Creating EKS Cluster in AWS] |
|  |

|  |
| --- |
| **Haji Mohammed**  **Devops Engineer**  **3-7-2023** |

**What is Amazon EKS?**

* Amazon EKS is a managed service that used to run Kubernetes on AWS.
* Using EKS users don’t have to maintain a Kubernetes control plan on their own.
* It is used to automate the deployment, scaling and maintenance of the containerized application.
* It works with most operating systems.

Application, Word

Description automatically generated

**EKS with various AWS Services**

* ECR (Elastic Container Registry) for container images
* Elastic Load Balancer for distributing traffic.
* IAM for providing authentication and authorization.
* VPC (Virtual Private Cloud) for isolating resources.

**Amazon EKS Components**

**Application, logo

Description automatically generated**

**What are a Nodes?**

A node is a physical or virtual machine. In EKS both Master Node and Worker Node are managed by EKS. There are two types of nodes.

Master Nodes:

A master node is a collection of components like storage,Controller,Scheduler,API server that makes up the control plan of Kubernetes.

The EKS itself creates the master node and manages it.

API Servers:

It controls the API servers whether it is kubctl (Kubernetes CLI) or rest API.

Etcd:

It is a highly available key-value store that is distributed among the Kubernetes cluster to store configuration data.

Controller Manager:

Cloud controller manager is used to manage the VMs,storage,databases and other resources associated with the Kubernetes cluster. It makes sure that you are using as much as the container needed at a point in time. It keeps a count of containers used and also records the state.

Scheduler:

It validates what and when the work needs to be done. It integrates with the controller manager and API servers.

Worker Nodes:

The worker nodes in a cluster are the machines or physical servers that run your application.

The user is responsible for creating and managing worker nodes.

Kublet:

It controls the flow to and from the API.

It makes sure containers are running in the pod.

Kubproxy:

It includes networking rules and access control.

It is like a Firewall

Diagram

Description automatically generated

Pods:

A group of containers is called pods.

They share networking, storage, IP address and port spaces.

Daemon Set:

It makes sure that all nodes runs a copy of a certain pod.

It is like a monitoring tool.

**Amazon EKS Workflow**

Diagram

Description automatically generated

Provision EKS cluster using AWS Console, AWS CLI, or one of the AWS SDKs.

Deploy worker nodes to the EKS cluster. There is already a predefined template that will automatically configure nodes.

Now we configure Kubernetes tools such as kubctl to communicate with the Kubernetes cluster.

We are now all set to deploy an application on the Kubernetes cluster.

**Creating EKS Kubernetes Cluster Using GUI**

Important steps

1. Creating a Master Node
2. Installing and configuring AWS CLI & kubectl
3. Creating a worker Node.

Step 1:

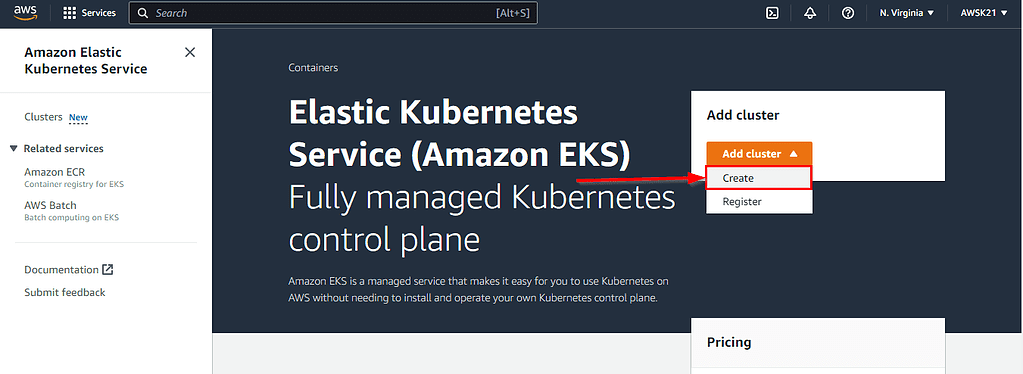
To create an AWS account

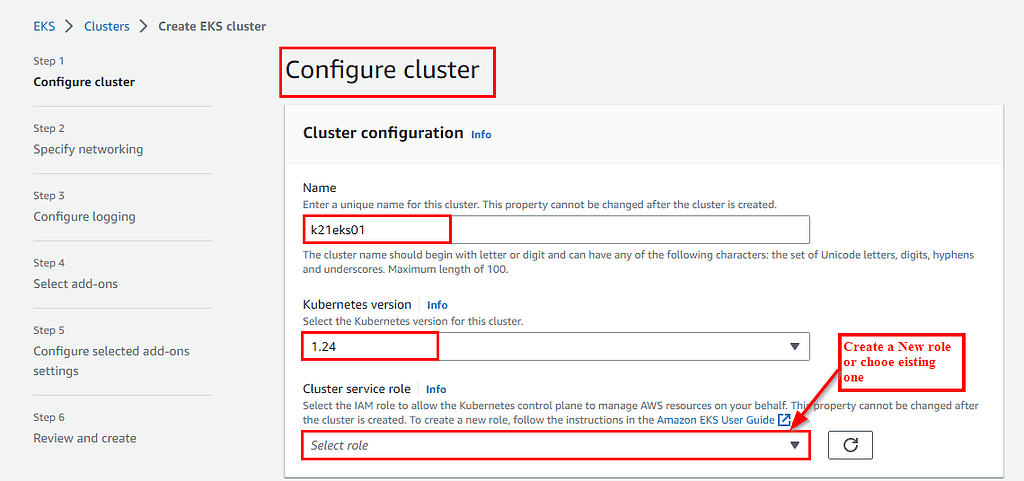
Step 2:

Creating Master Node

**Steps to create Master Node**

1. Log in to the AWS portal find the Kubernetes service by searching for EKS.
2. Create Kubernetes cluster and specify the name for the Cluster.





1. Next is to create the role click on create role -> AWS Service -> EKS (from AWS Services) -> Select Cluster -> Next Permissions

Text

Description automatically generated with medium confidence

Graphical user interface, text, application, email

Description automatically generated

1. Leave the selected policies as it is and click on Review Page.
2. Enter a name for the role and hit the create role button at the bottom of the page to create the IAM role is created.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

1. Now Master node will be created (Takes time to create)

Graphical user interface, text, application, website

Description automatically generated

Step 3:

Next step is to install & configure AWS CLI on the EC2 instance.

Follow the as steps below.

Create a Amazon Linux Instance <https://k21academy.com/amazon-web-services/linux-ec2-instance/>

To get latest EKS user guide <https://docs.aws.amazon.com/eks/latest/userguide/getting-started-console.html>

Configure CLI on EC2 <https://k21academy.com/docker-kubernetes/amazon-eks-kubernetes-on-aws/?utm_source=youtube&utm_medium=referral&utm_campaign=kubernetes17_july21_k21#a>

1. Click the account name and select Security Credentials. Scroll down and click on Create Access Key

Graphical user interface, text, application, email

Description automatically generated

A screenshot of a computer

Description automatically generated

Step 4 :

To install & configure the kubectl by checking your cluster name & region Name where the EKS master node is running from the console.

Check the status of cluster and configure kubectl with EKS API Server and validate Kubectl configuration to the master node.

Step 5:

**To create worker node**

1. On the cluster page select the compute tab and then choose add node group.
2. On the configure node group page fill out the parameters accordingly and then choose next.

Name -Entre a unique name for your managed node group.

Node IAM role name- Choose the node instance role to use with your node group.

Graphical user interface, text, application, email

Description automatically generated

To Create an IAM role follow the below-given steps

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Leave the other settings to default and proceed further.

Step 6:

Next is to configure the compute & Scaling of worker Nodes.

1. In this step we are providing the compute configurations.

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

1. Keep rest as default. on the review and create the page
2. Review your managed node group configuration and choose to create
3. Worker node group is under creation ( takes few minutes to create)

Graphical user interface, application

Description automatically generated

Step 7:

Verify the worker node status from kubectl

Perform the given command.

Text

Description automatically generated with medium confidence

**Conclusion:**

We have created EKS cluster from AWS GUI