Create Amazon EKS cluster by eksctl | How to create Amazon EKS cluster using EKSCTL | Create EKS Cluster in command line

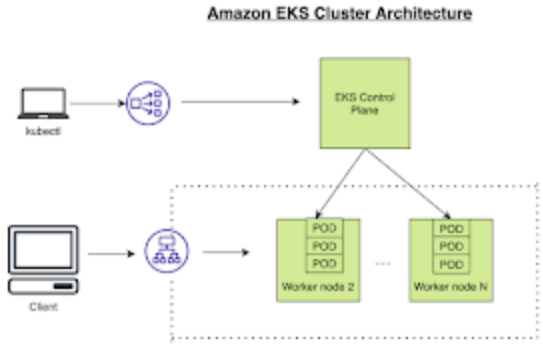
What is Amazon EKS

Amazon EKS is a fully managed container orchestration service. EKS allows you to quickly deploy a production ready Kubernetes cluster in AWS, deploy and manage containerized applications more easily with a fully managed Kubernetes service.

EKS takes care of Master node/control plane. We need to manage worker nodes.

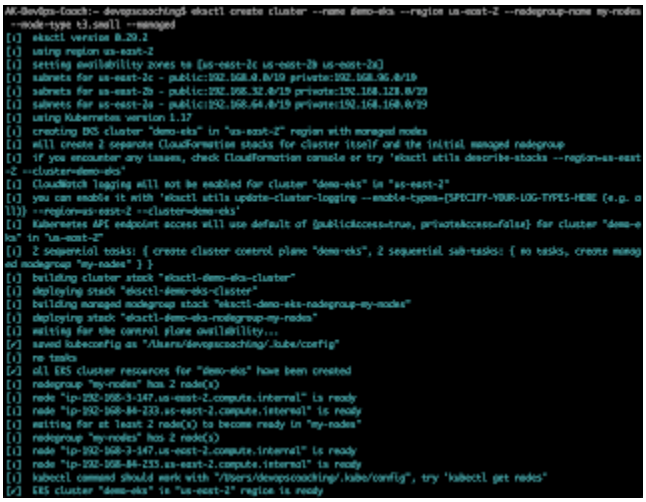
Pre-requistes:

You can use Windows, Macbook or any Ubuntu or Red Hat EC2 instance to setup the below tools.



- **Install AWS CLI** – Command line tools for working with AWS services, including Amazon EKS.
- `Install eksctl` – A command line tool for working with EKS clusters that automates many individual tasks.
- **install kubectl** – A command line tool for working with Kubernetes clusters.
  - **Create AWS Access Keys**
  - Once you install all of the above, you need to have AWS credentials configured in your environment. The `aws configure` command is the fastest way to set up your AWS CLI installation for general use.
  - Make sure you do not have any IAM role attached to the EC2 instance. You can remove IAM role by going into AWS console and Detach the IAM role.
  - aws configure
  - the AWS CLI prompts you for four pieces of information: `access key`, `secret access key`, `AWS Region`, and `output format`.
  - **Create EKS Cluster using eksctl**
  - eksctl create cluster --name demo-eks --region us-east-2 --nodegroup-name my-nodes --node-type t3.small --managed --nodes 2

- the above command should create a EKS cluster in AWS, it might take 15 to 20 mins. The ***eksctl*** tool uses CloudFormation under the hood, creating one stack for the EKS master control plane and another stack for the worker nodes.



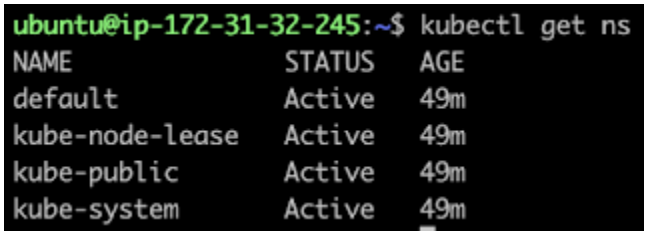
eksctl get cluster --name demo-eks --region us-east-2

This should confirm that EKS cluster is up and running.

Connect to EKS cluster

kubectl get nodes

kubectl get ns



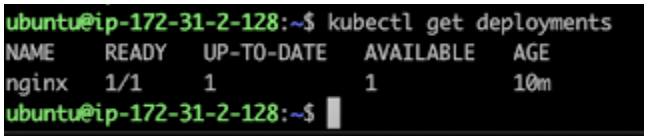
Deploy Nginx on a Kubernetes Cluster

Let us run some apps to make sure they are deployed to Kuberneter cluster. The below command will create deployment:

kubectl create deployment nginx --image=nginx

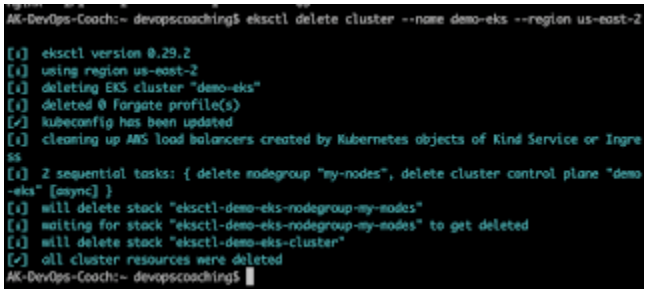
View Deployments

kubectl get deployments



Delete EKS Cluster using eksctl

eksctl delete cluster --name demo-eks --region us-east-2



the above command should delete the EKS cluster in AWS, it might take a few mins to clean up the cluster.

Errors during Cluster creation

If you are having issues when creating a cluster, try to delete the cluster by executing the below command and re-create it.

eksctl delete cluster --name demo-eks --region us-east-2

or login to AWS console --> AWS Cloudformation --> delete the stack manually.

you can also delete the cluster under AWS console --> Elastic Kubernetes Service --> Clusters Click on Delete cluster