1. **What is mean by kubeconfig in k8s cluster?**

The kubeconfig file is a YAML file that defines how kubectl connects to and interacts with one or more Kubernetes clusters.

Kubeconfig file is located at ~/.kube/config.

**It contains the information about**

1. **Clusters:**

Cluster name, API server endpoint and certification authority information.

1. **Users:**

Authentication credentials like tokens, client certificates, access keys, username, password.

1. **Context:**

A link between a cluster and a user with an optional namespace.

**Why Is It Important?**

* Authentication: It holds your credentials, ensuring that only authorized users can interact with the cluster.
* Connectivity: It tells kubectl exactly where to find the cluster's API server.
* Multi-Cluster Management: It allows you to define multiple clusters and contexts in one place, making it easy to switch between them without having to re-enter details every time.

1. **If I want to connect to private end point EKS cluster, what do i need to do?**

When EKS cluster’s endpoint is set to private, it can be accessed only within the vpc or through some secure networking setup.

Step 1. We need to check whether the cluster has a private endpoint or not.

Check private end point is enabled or not by AWS console->EKS->Cluster->networking

section.

Step 2. Launch an EC2 instance inside the cluster’s VPC.

Add an IAM role with permissions:

* Attach AmazonEKSClusterPolicy.
* Optionally AmazonEKSWorkerNodePolicy if needed.

Step 3. Connect to the EC2 instance using private key.

Step 4. Install AWS CLI and kubectl.

* Install aws cli

sudo apt update

sudo apt install -y awscli

* Install kubectl

curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"

chmod +x kubectl

sudo mv kubectl /usr/local/bin/

* verify installation

aws --vesion

kubectl version --client

Step 5. Configure aws using access keys.

aws configure

Step 6. Update kubeconfig to use the private endpoint.

aws eks update-kubeconfig --region <your-region> --name <your-cluster-name>

check whether kubeconfig file is created or not using command cat ~/.kube/config.

Step 7. Test the connection

Kubectl get nodes