RBAC(role based access control)

first create eks cluster and install kubernetes and docker in instances

aws\_authentication

-------------------

we need to give below command

kubectl get configmap aws-auth -n kube-system -o yaml

u can get like this take it change as per your requirement

--------------------------------------------------------------------

apiVersion: v1

data:

mapRoles: |

- groups:

- system:bootstrappers

- system:nodes

rolearn: arn:aws:iam::650732254329:role/eksctl-abn-nodegroup-abn-NodeInstanceRole-WSY7MjP88PVw

username: system:node:{{EC2PrivateDNSName}}

kind: ConfigMap

metadata:

creationTimestamp: "2025-04-02T15:14:07Z"

name: aws-auth

namespace: kube-system

uid: a37e2af7-989a-4a81-ad8c-38eb83522e0b

-------------------------------------------------------------------------

now need to create a policy in that need to give our clustername:abn and region:us-east-1

\* select service as EKS

A screenshot of a computer

AI-generated content may be incorrect.

\*need to read access

\* select add arn

A screenshot of a computer

AI-generated content may be incorrect.

* Cluster name:abn ,region as :us-east-1 that it now create policy

Now create a user name:Sandeep and attach above policy to Sandeep

A screenshot of a computer

AI-generated content may be incorrect.

Create security key in Security credentials

* U can access aws configure with another linux instance and also u can access cluster also depending up on role and cluster binding
* After that cluster installed instance need to run both rbac and aws-auth
* Kubectl apply -f
* Open another instance install Kubernetes and give aws configure authentication user whatever we created
* And then with name space or default u can view pods ,nodes ,pv’s that it