CREATING CLUSTER USING TERRAFORM

* Terraform everything written in terraform-aws-eks-cluster
* First apply
* 1.vpc
* 2.security group
* 3.bastion
* 4.eks cluster
* Terraform apply -auto-approve
* Here u need to understand one thing from git bash u are doing , so git bash is configure with aws with IAM credentials user , after applying all this when u create new instance use same IAM user with new credentials for that new instance , because that credentials only have AWSEKSClusteradminPolicy
* Below image is there check once use same IAM user and create new credential in terraform user .
* A screenshot of a computer

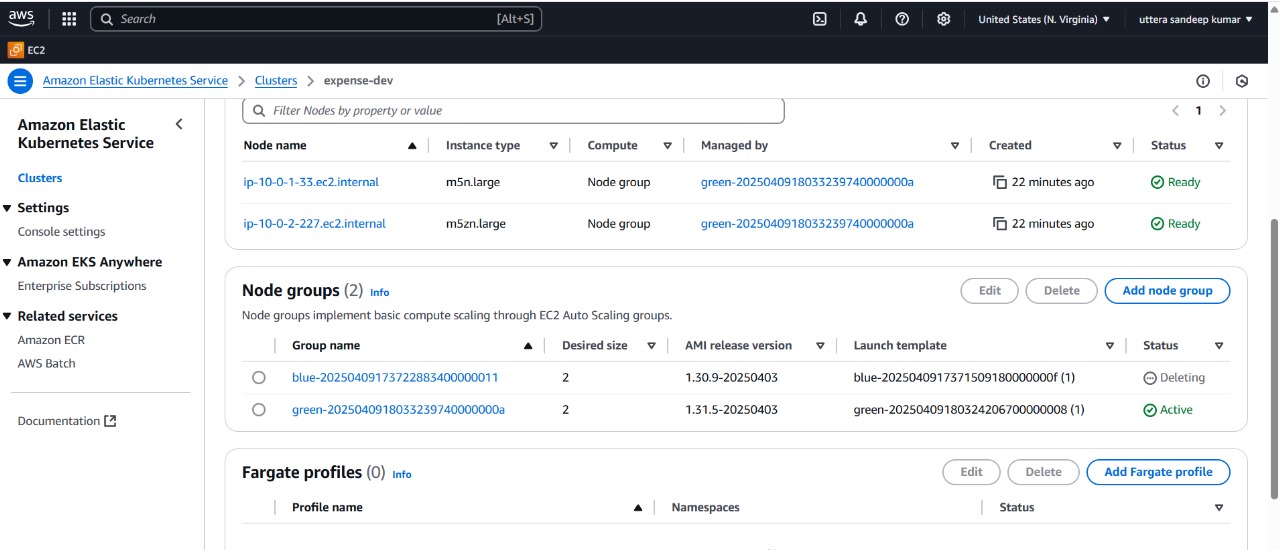
  AI-generated content may be incorrect.
* When u created new instance add user IAM credentials and then install Kubernetes and then login to cluster using below command
* aws eks update-kubeconfig --region region-code --name my-cluster
* check kubectl get pods ,and kubectl get nodes , svc ,pv ,configmaps
* then apply deployment in cluster 08-deployment.yaml
* FROM HERE CLUSTER UPGRADATION START
* No downtime to the existing apps but no new release and no new deployment in the upgrade time
* First, we need to create new nodes group(green)
* Uncommand green node in terraform don’t comment blue now let them run .
* Now run terraform apply -auto-approve

A screenshot of a computer

AI-generated content may be incorrect.

* now taint the green nodes, so that they should not get any pods scheduled

kubectl taint nodes ip-10-0-1-96.ec2.internal project=expense:NoSchedule

* now upgrade your contro; plane , do it from console
* go got eks cluster there is upgration for cluster upgrade it wait for 15 min
* after that go to compute upgrade green node group force update it wait 10 min
* 
* In terraform-aws-eks-cluster also change it to our new upgrade version now
* Nowcordon blue nodes

kubectl cordon nodename

* A screenshot of a computer

  AI-generated content may be incorrect.
* Untaint green nodes, first describe pod check taint is there or not
* A computer screen with white text

  AI-generated content may be incorrect.
* If taint is there then untaint green new nodes

kubectl taint nodes ip-10-0-1-96.ec2.internal project=expense:NoSchedule-

* Now drain bule nodes

Kubectl drain node-name

* Last step be carefull now comment blue node in visual studio and then update version also 1.29 to 1.30
* Now apply terraform blue nodes will be destroyed
* Check pods are running or not and also check in which node pods are running
* Kubectl get pods -o wide