

## Kubernetes Task-2

### Task Description:

Create the K8s EKS, further you have to do the deployment of the Nginx application and access the application outside the cluster.

### Techstacks needs to be used :

- AWS EKS
- EKSCtl
- Kubectl

### Step1 : Create Cluster

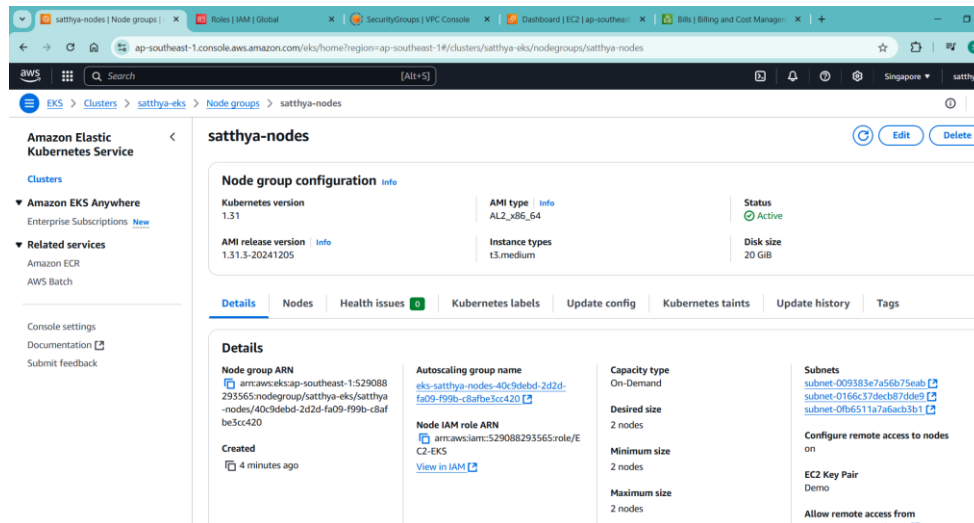
- ❖ Click create cluster.
- ❖ Enter cluster name.
- ❖ Select Kubernetes version.
- ❖ Provide correct Role.
- ❖ Choose VPC.

The screenshot displays the AWS Management Console for the Amazon Elastic Kubernetes Service (EKS). The left sidebar shows the navigation menu with 'Amazon Elastic Kubernetes Service' selected. The main content area shows the details for the cluster 'sathya-eks'. A blue banner at the top indicates the next step: 'Provision compute capacity for your cluster by adding a Managed node group or creating a Fargate profile.' Below this, the cluster status is 'Active'. The 'Compute' tab is selected, showing 'Nodes (0)'. The 'Cluster info' section includes details like 'Kubernetes version 1.31', 'Support period' (Standard support until November 26, 2025), and 'Provider EKS'. The 'Cluster health issues' and 'Upgrade insights' sections both show zero issues or insights.

Node name	Instance type	Compute	Managed by	Created	Status
-----------	---------------	---------	------------	---------	--------

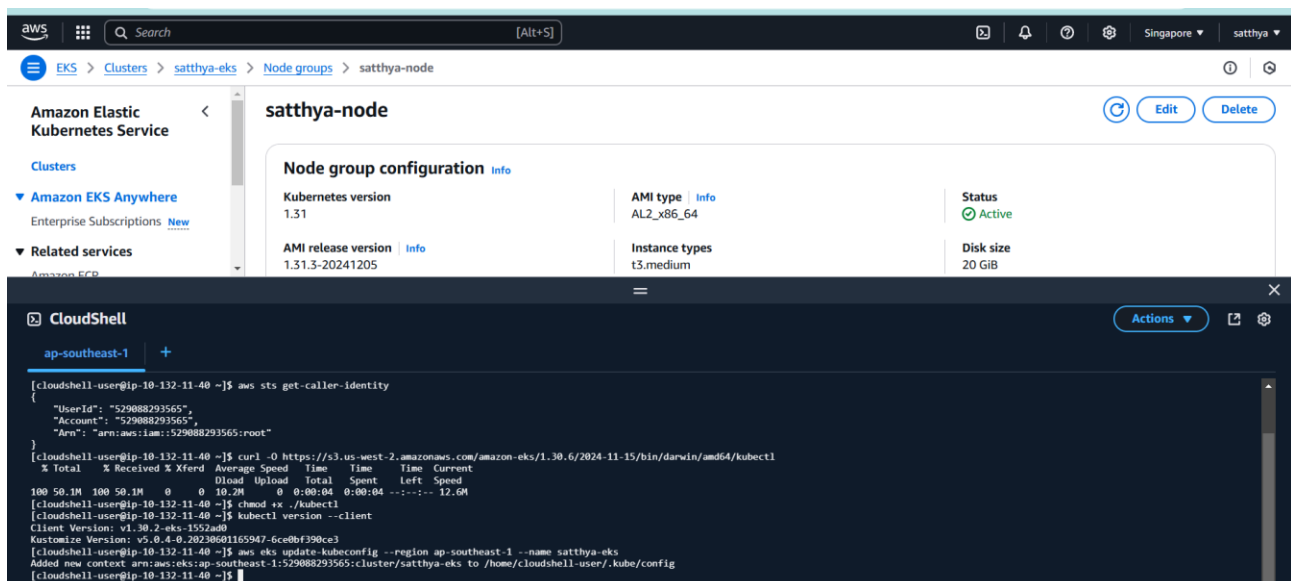
## Step2 : Set Up Worker Nodes

- ❖ Add Node Group
- ❖ Provide name for the node group
- ❖ Select the IAM role
- ❖ Select the instance type for worker nodes
- ❖ Define scaling options



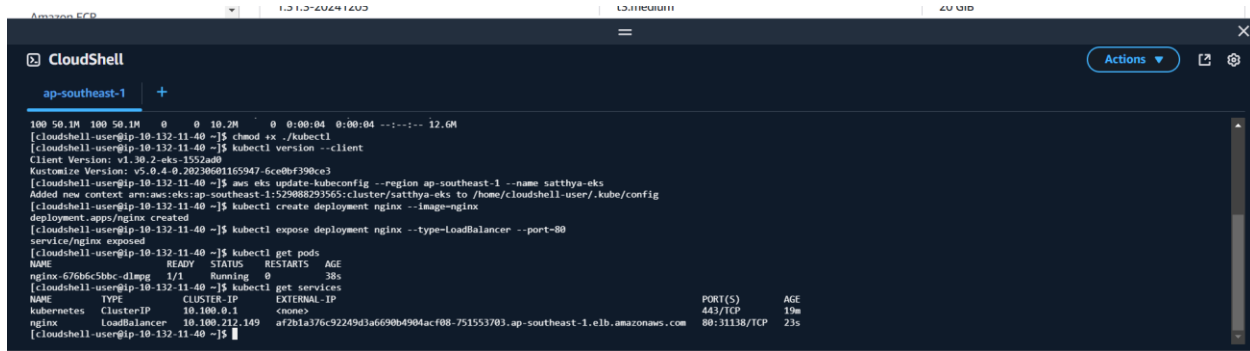
## Step3 : Configure kubectl to connect to the cluster

- ❖ `aws eks update-kubeconfig --region ap-southeast-1 --name satthya-eks`



## Step4 : Deploy the Nginx Application

- ❖ kubectl create deployment nginx --image=nginx
- ❖ kubectl expose deployment nginx --type=LoadBalancer --port=80
- ❖ kubectl get pods
- ❖ kubectl get services



```
100 50.1M 100 50.1M 0 0 10.2M 0 0:00:04 0:00:04 --:--:-- 12.0M
[cloudshell-user@ip-10-132-11-40 ~]$ chmod +x ./kubectl
[cloudshell-user@ip-10-132-11-40 ~]$ kubectl version --client
Client Version: v1.30.2-eks-1552a0
Kubernetes Version: v5.0.4-0.20230601165947-6ce0bf390ce3
[cloudshell-user@ip-10-132-11-40 ~]$ aws eks update-kubeconfig --region ap-southeast-1 --name satthya-eks
Added new context arn:aws:eks:ap-southeast-1:52988293565:cluster/satthya-eks to /home/cloudshell-user/.kube/config
[cloudshell-user@ip-10-132-11-40 ~]$ kubectl create deployment nginx --image=nginx
deployment.apps/nginx created
[cloudshell-user@ip-10-132-11-40 ~]$ kubectl expose deployment nginx --type=LoadBalancer --port=80
service/nginx exposed
[cloudshell-user@ip-10-132-11-40 ~]$ kubectl get pods
NAME READY STATUS RESTARTS AGE
nginx-67b6c5bbc-dlapp 1/1 Running 0 38s
[cloudshell-user@ip-10-132-11-40 ~]$ kubectl get services
NAME TYPE CLUSTER-IP EXTERNAL-IP
kubernetes ClusterIP 10.100.0.1 <none>
nginx LoadBalancer 10.100.212.149 af2b1a376c92249d3a6690b4904acf08-751553703.ap-southeast-1.elb.amazonaws.com 80:31138/TCP 23s
[cloudshell-user@ip-10-132-11-40 ~]$
```

- ❖ af2b1a376c92249d3a6690b4904acf08-751553703.ap-southeast-1.elb.amazonaws.com

