**AWS**

**Elastic Kubernetes Service**

1. **Introduction**

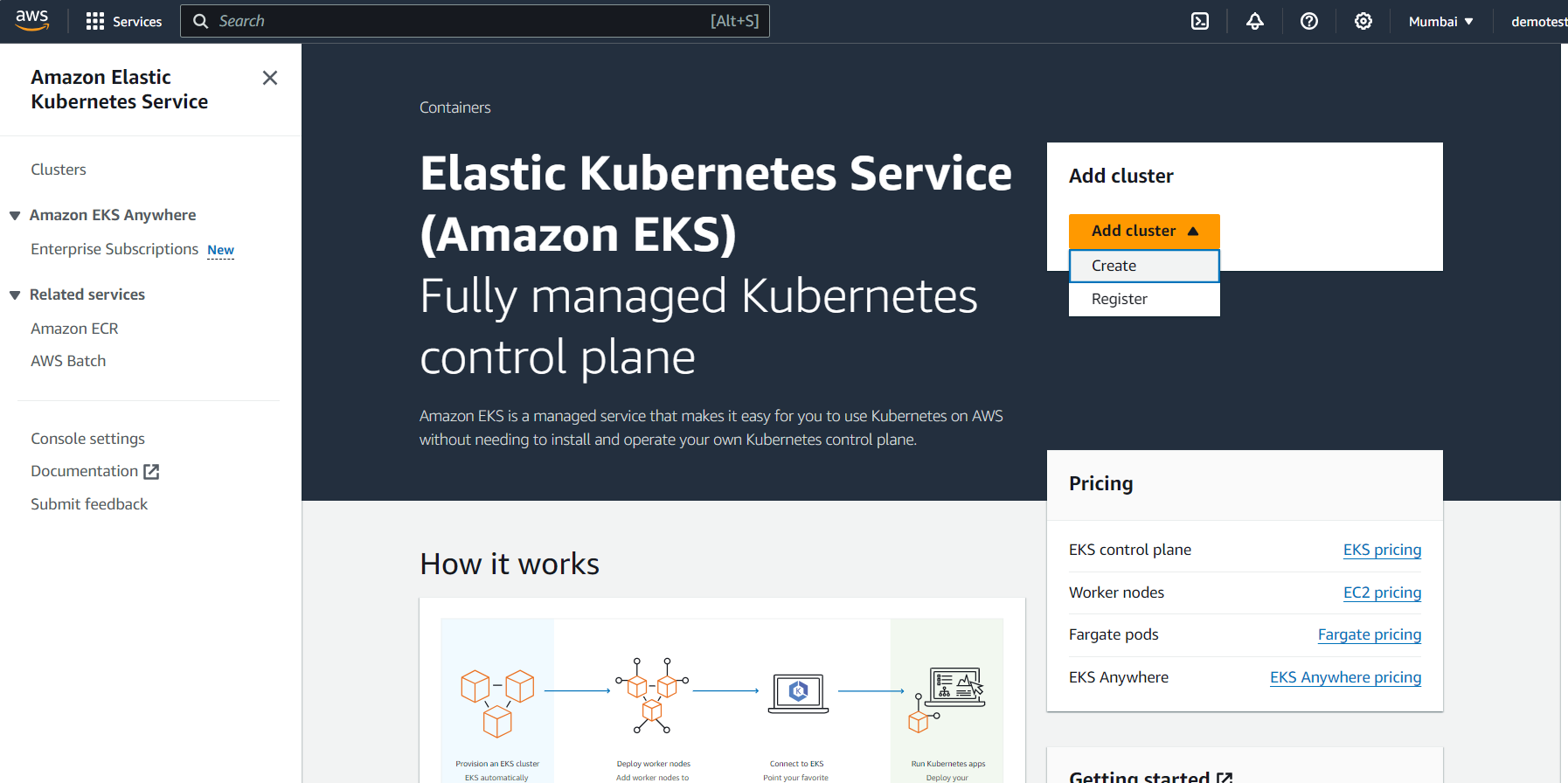
Amazon Elastic Kubernetes Service (EKS) is a managed service that simplifies running Kubernetes on AWS without needing to install and operate your own Kubernetes control plane or nodes. This document provides a step-by-step guide to creating an EKS cluster.

1. **Prerequisites**

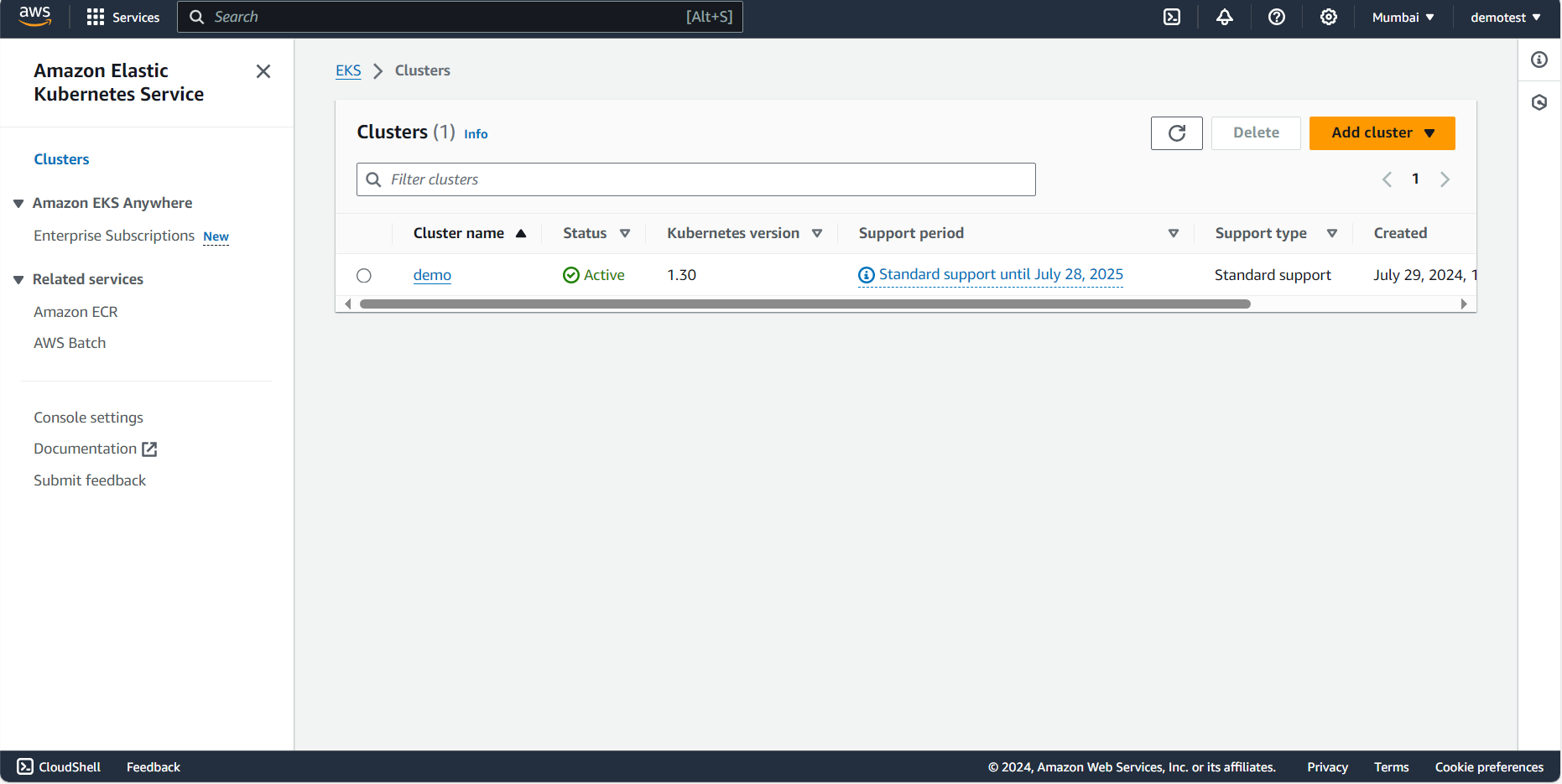
Before you start, ensure you have:

* 1. An AWS account.
  2. AWS CLI installed and configured.
  3. *kubectl* (Kubernetes command-line tool) installed.
  4. *eksctl* (a command-line tool for creating EKS clusters) installed.
  5. IAM permissions to create and manage EKS resources.

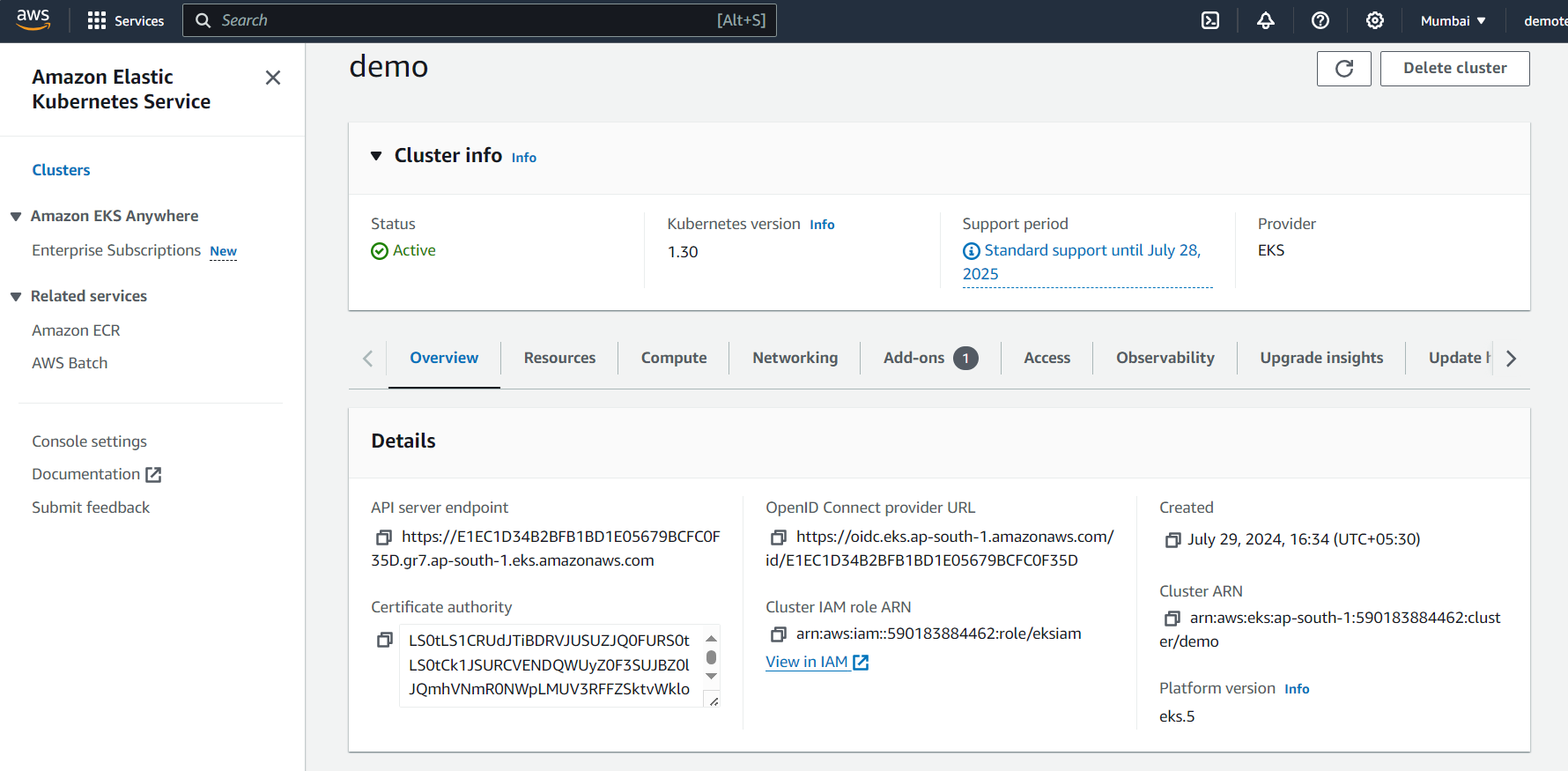
1. **Creating an EKS Cluster**
   1. Through AWS EKS console:
      * Open Amazon Elastic Kubernetes Service
      * Select add Cluster and Create



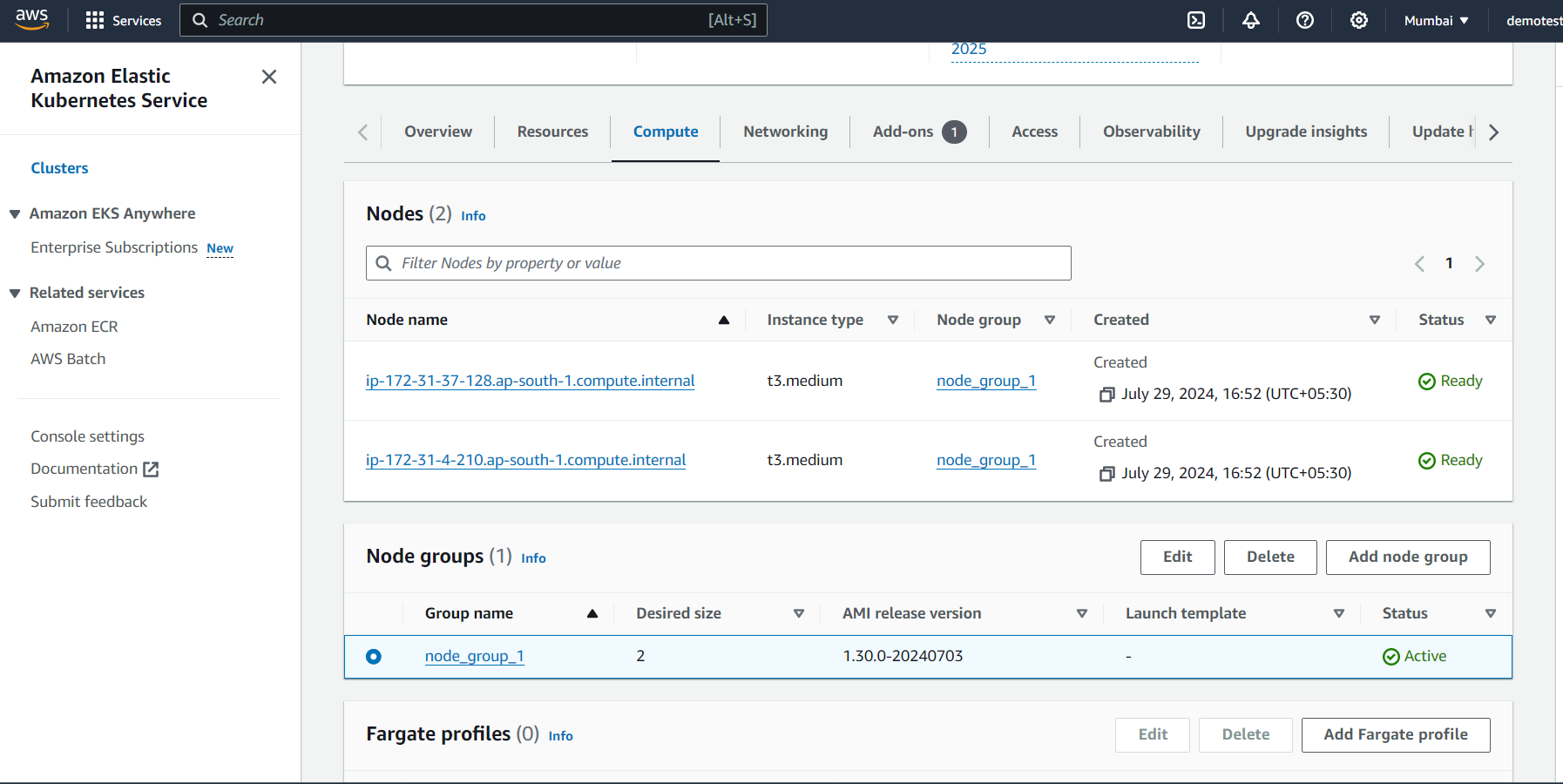
* + - Configure the cluster settings with the default values unless you need to make changes.
    - Then review and create the cluster.
    - After the cluster creation, need to create worker nodes.
      * Select the created cluster from the clusters tab.



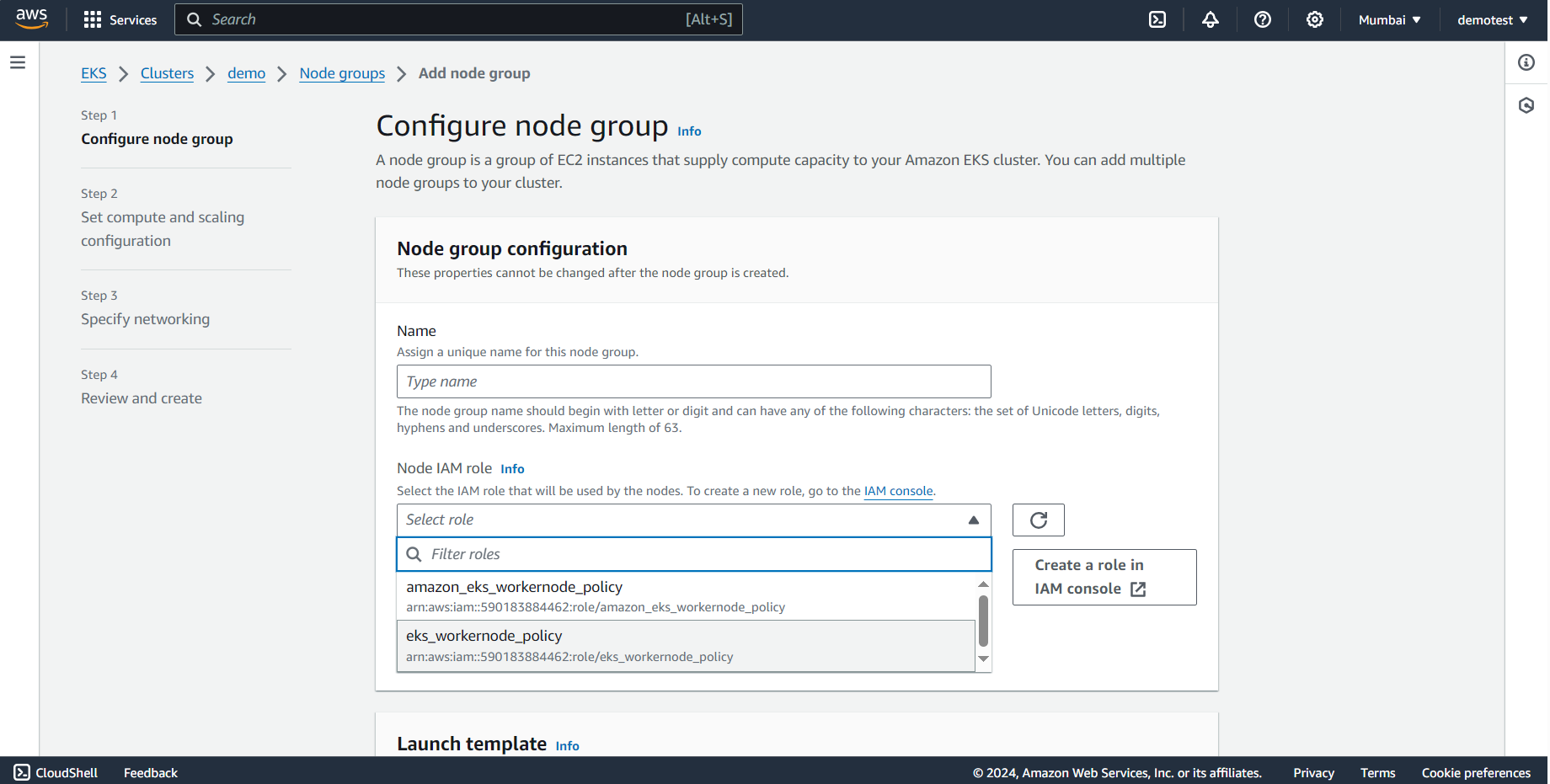
* + - * Select compute tab:

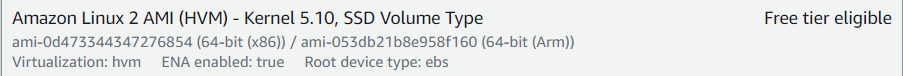
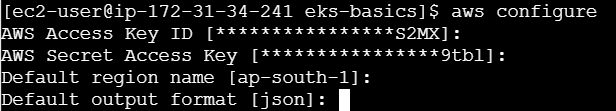
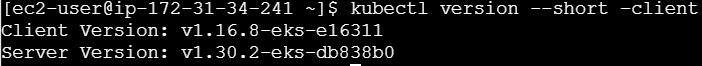
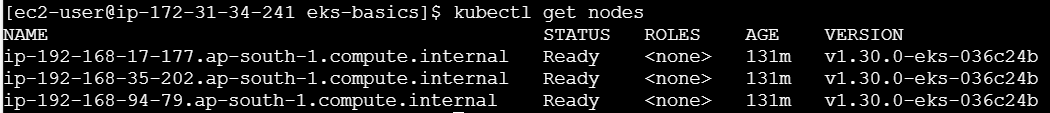
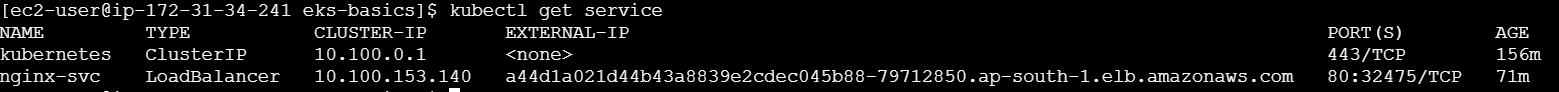
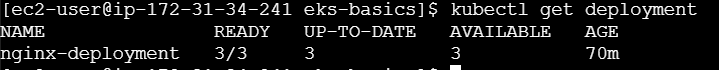


* + - * Select add node group:



* + - * Create a new node group with the default configurations (\*\*create a new IAM role):



* 1. **Instance deployment and Connecting clusters.**
     + **Create an AWS machine virtual machine instance.**
       - 
       - Use the default configurations.
     + **AWS CLI installation:**
       - Download latest aws cli: *curl "https://awscli.amazonaws.com/awscli-exe-linux-x86\_64.zip" -o "awscliv2.zip"*
       - Unzip the installation file: *unzip awscliv2.zip*
       - Changing to bin directory*: sudo ./aws/install --bin-dir /usr/bin --install-dir /user/bin/aws-cli --update*
       - Checking the aws version: *aws –version*
       - 
     + **AWS CLI configuration:**
       - **(\*\*create the access and secret key for the desired IAM user)**
       - **Run: -** *aws configure*
       - Provide the access key and secret key obtained in IAM page:
       - 
     + Kubectl installation:
       - Download the kubectl: *curl -o kubectl https://amazon-eks.s3.us-west-2.amazonaws.com/1.16.8/2020-04-16/bin/linux/amd64/kubectl*
       - Provide executable permission: *chmod +x ./kubectl*
       - *Change the file path: mkdir -p $HOME/bin && cp ./kubectl $HOME/bin/kubectl && export PATH=$PATH:$HOME/bin*
       - Check the version: *kubectl version --short –client.*
       - 
     + AWS EKS CLI installation:
       - Download the eksctl: *curl --silent --location* *"https://github.com/weaveworks/eksctl/releases/latest/download/eksctl\_$(uname -s)\_amd64.tar.gz" | tar xz -C /tmp*
       - Move the file to bin folder: *sudo mv /tmp/eksctl /usr/local/bin*
       - Get cluster details: *eksctl get cluster*
     + Updating the cluster details:
       - Run:- *aws eks update-kubeconfig --name <cluster\_name> --region <region>*
       - Get cluster nodes*:* run:- *kubectl get nodes*
       - 
       - Get nodes pod: Run: - *kubectl get pods*
  2. Deploying a sample nginx welcome page:
     + Install git command:
       - Run: - *sudo yum install -y git*
     + Cloning the (demo project) git repository:
       - *git clone* [*https://github.com/nspacer/eks-basics.git*](https://github.com/nspacer/eks-basics.git)
     + deploying service and deployment as pods:
       - changing to git file directory: *cd eks-basics/*
       - list the files: *ls*
       - Applying service pod: *kubectl apply -f ./nginx-svc.yaml*
       - Applying deployment pod: *kubectl apply -f ./nginx-deployment.yaml*
       - View deployed service : *kubectl get service*
       - 
       - View deployed deployment: *kubectl get deployment*
       - 
     + View deployed pods:
       - Run: - *kubectl get pods*
       - 