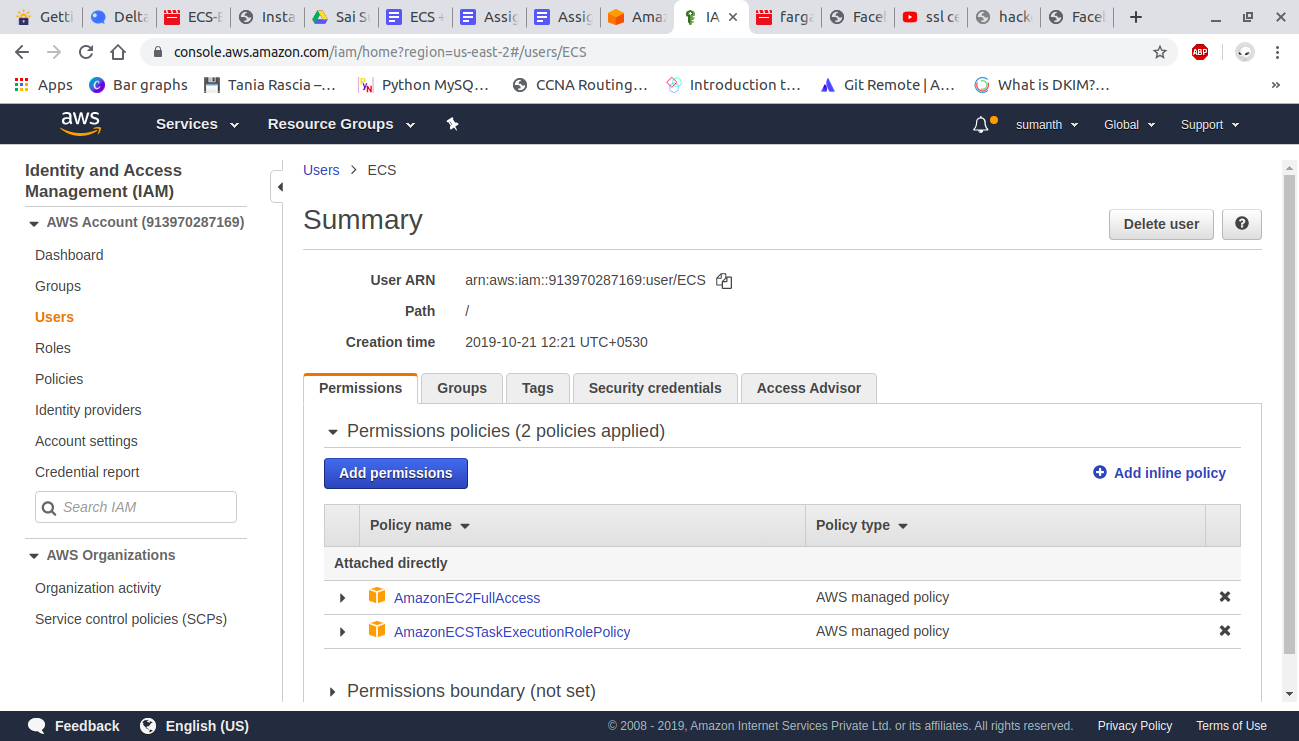
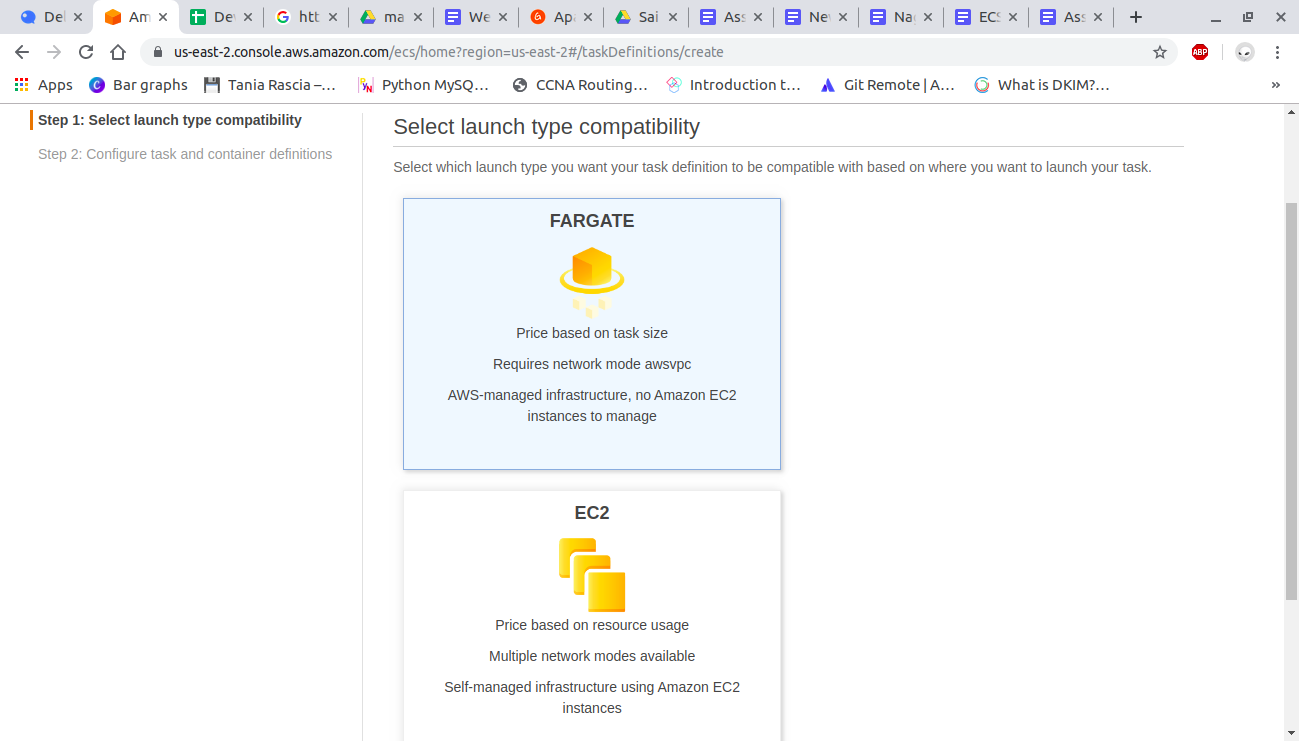
### **ECS:**

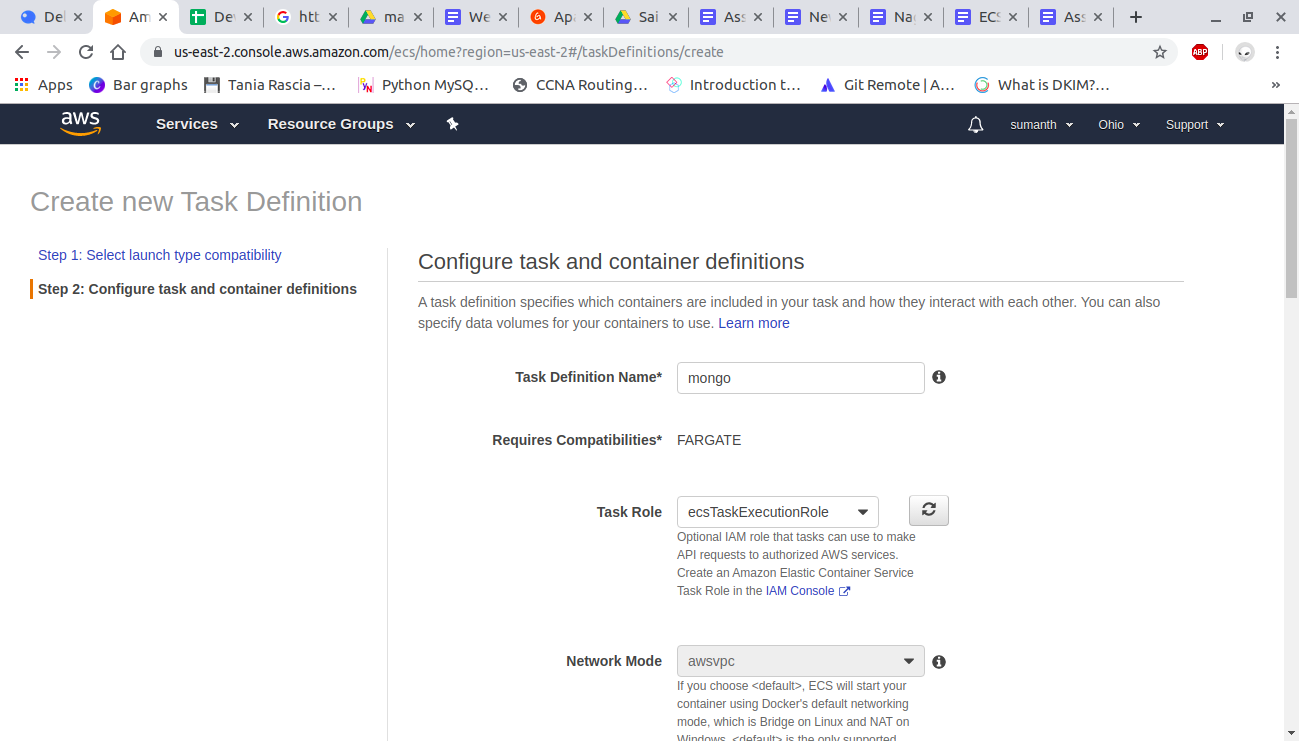
#### **ECS with fargate:**

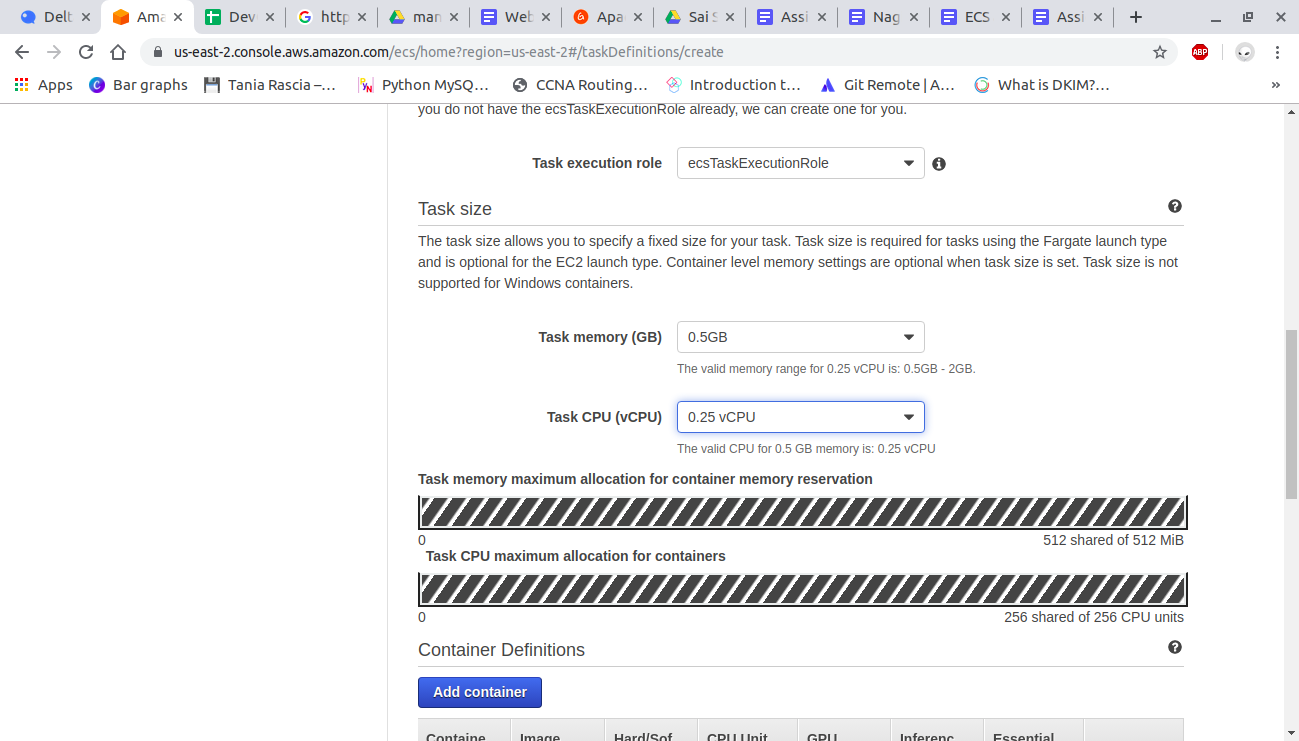
Create IAM user with ECS permissions

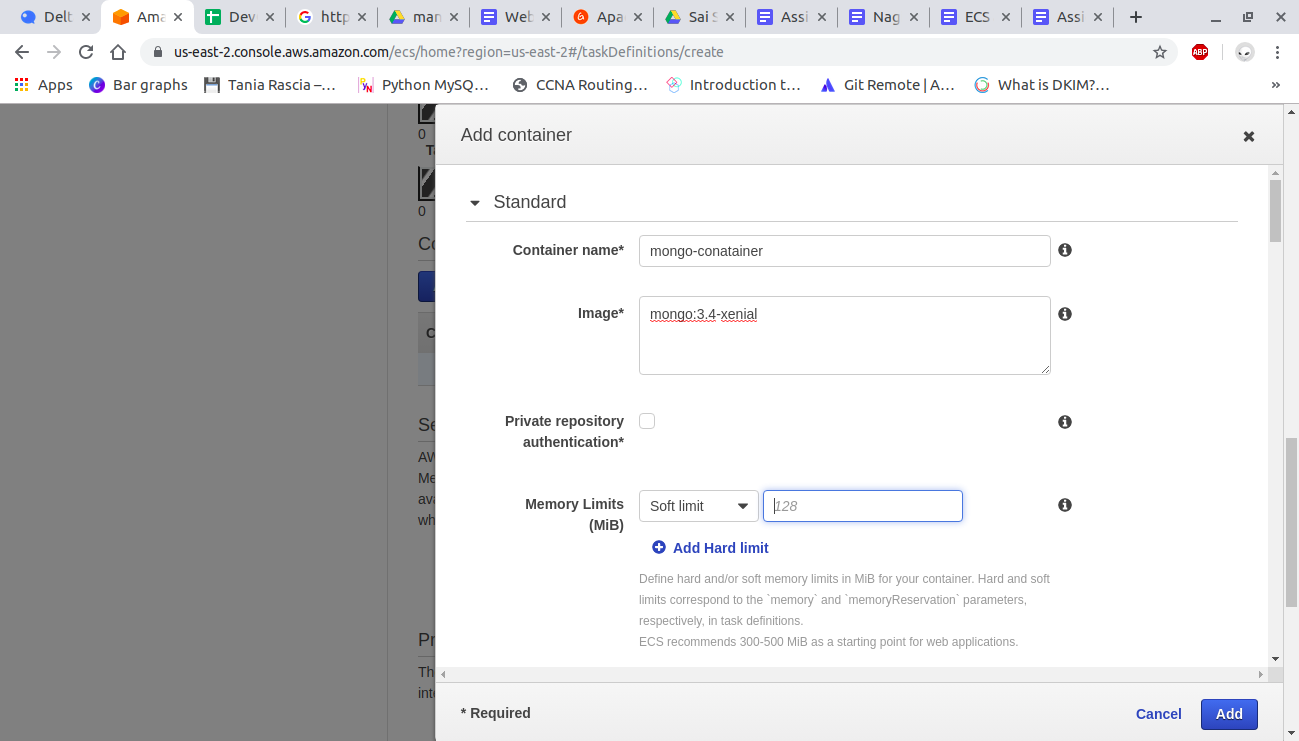


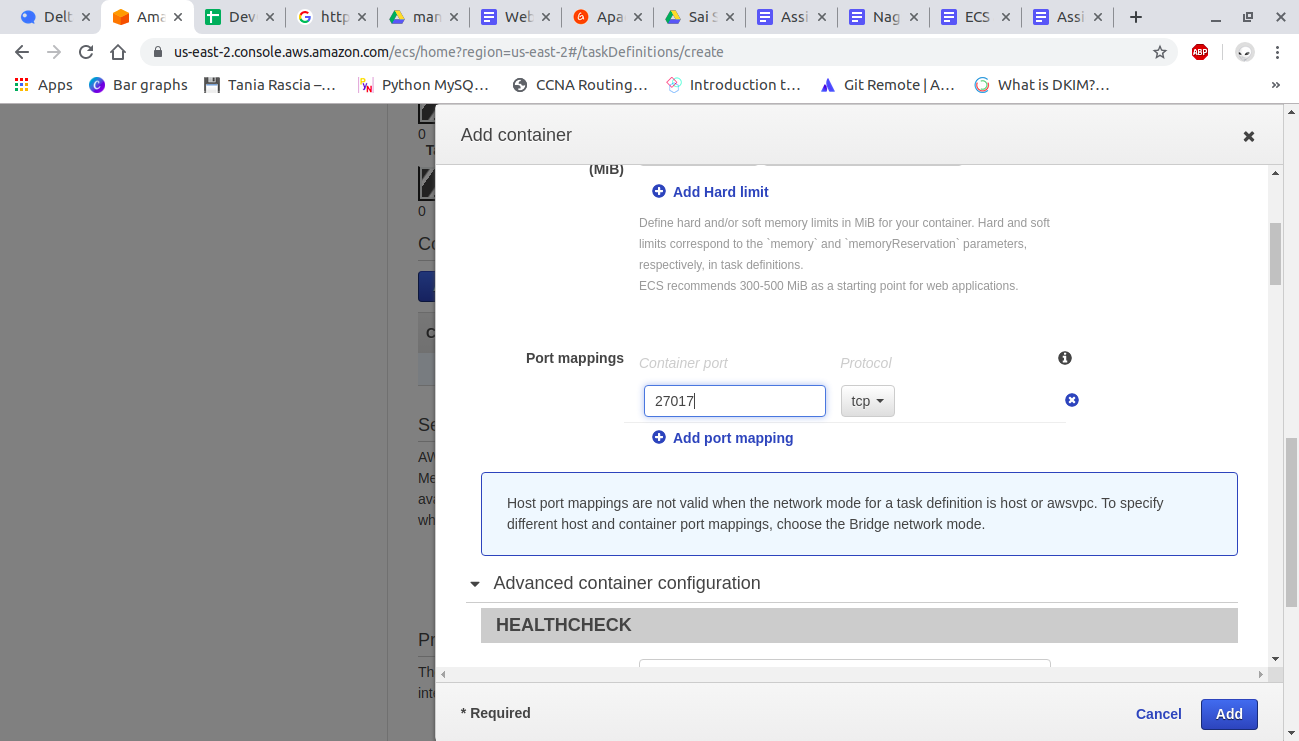
Create a task definition.

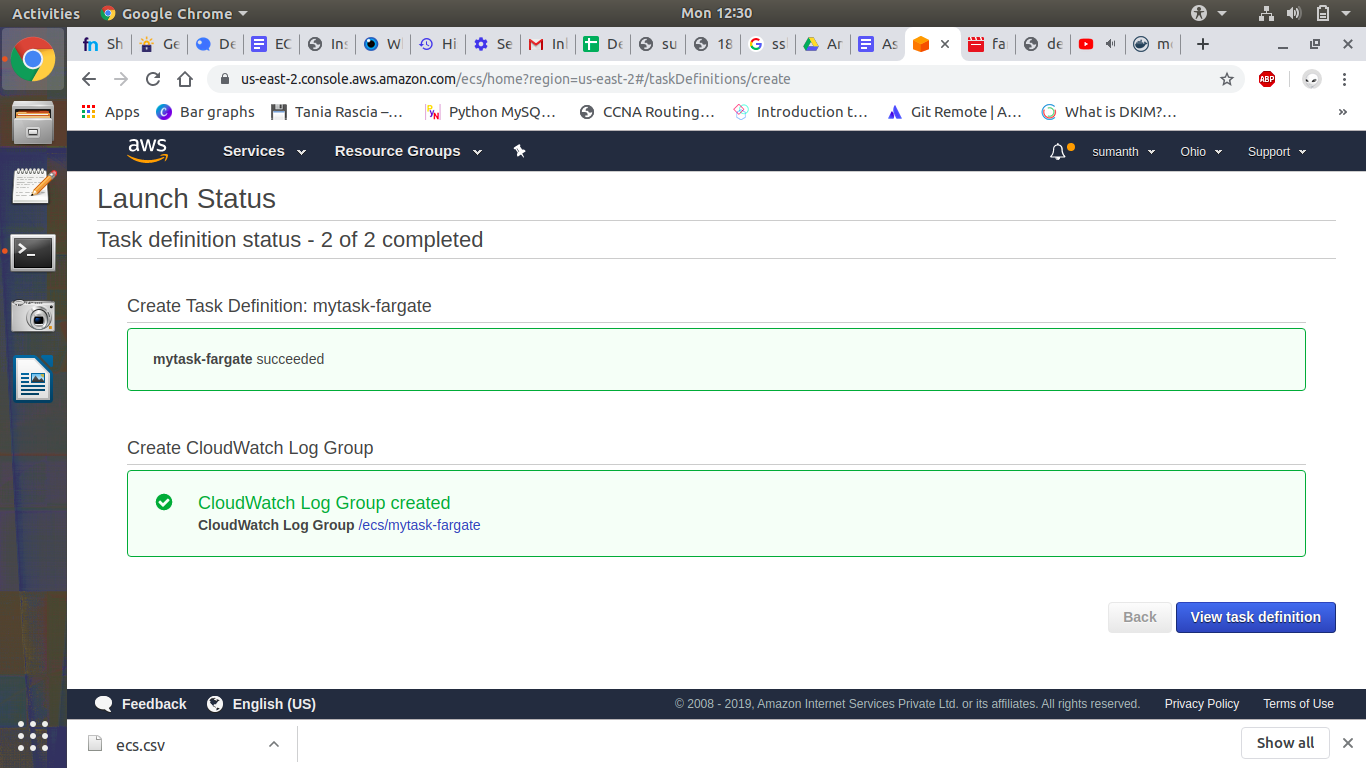




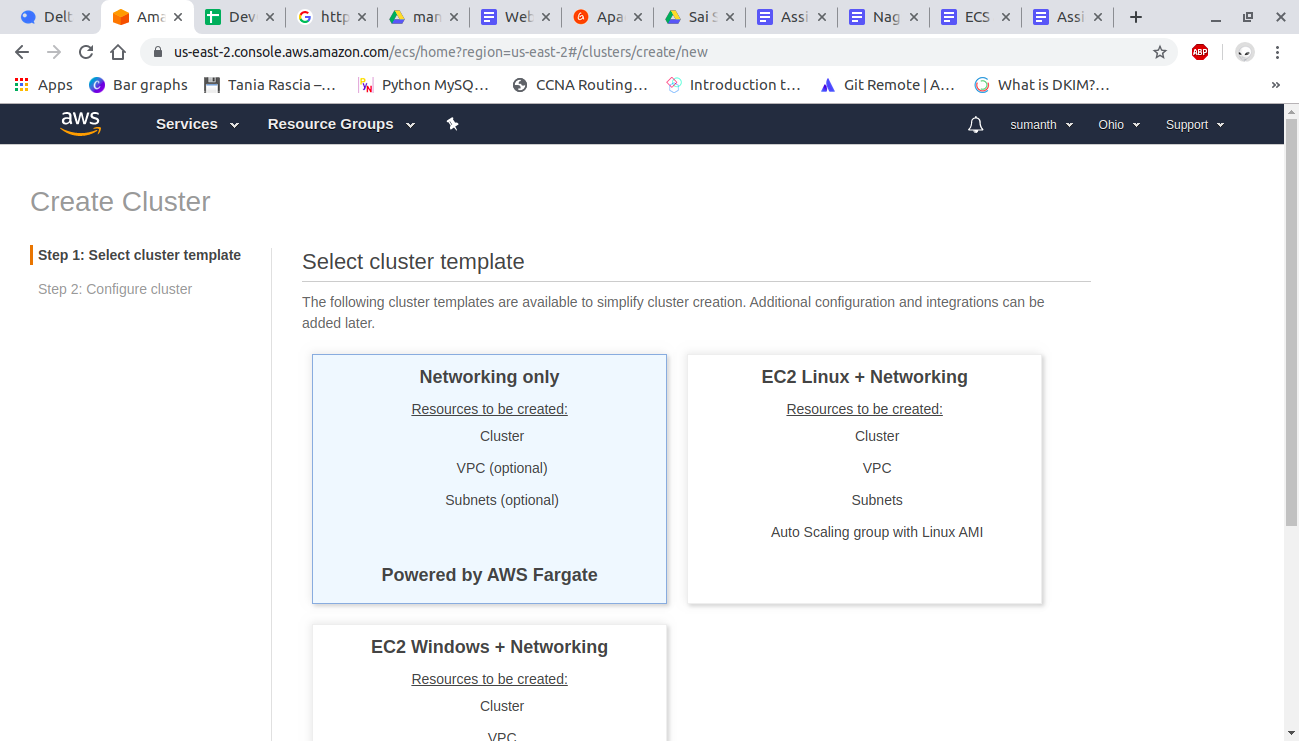


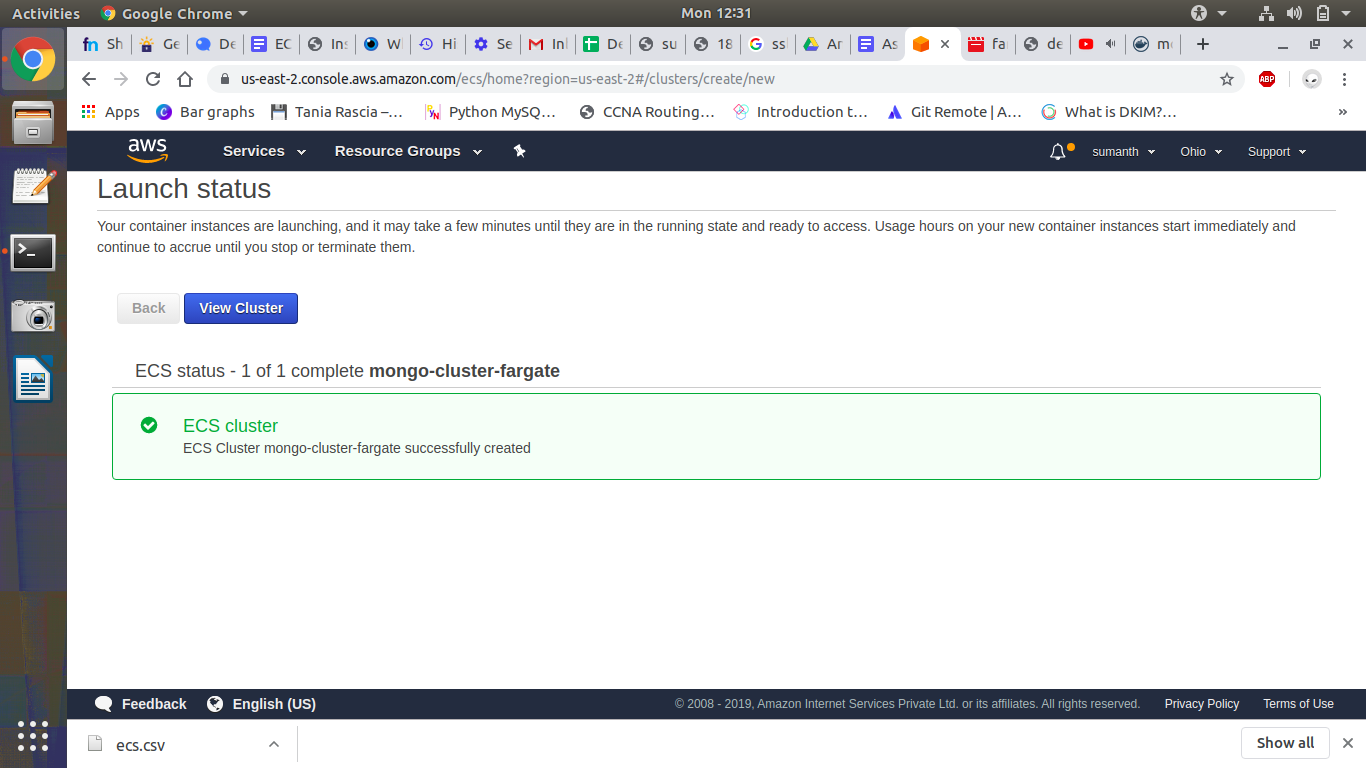




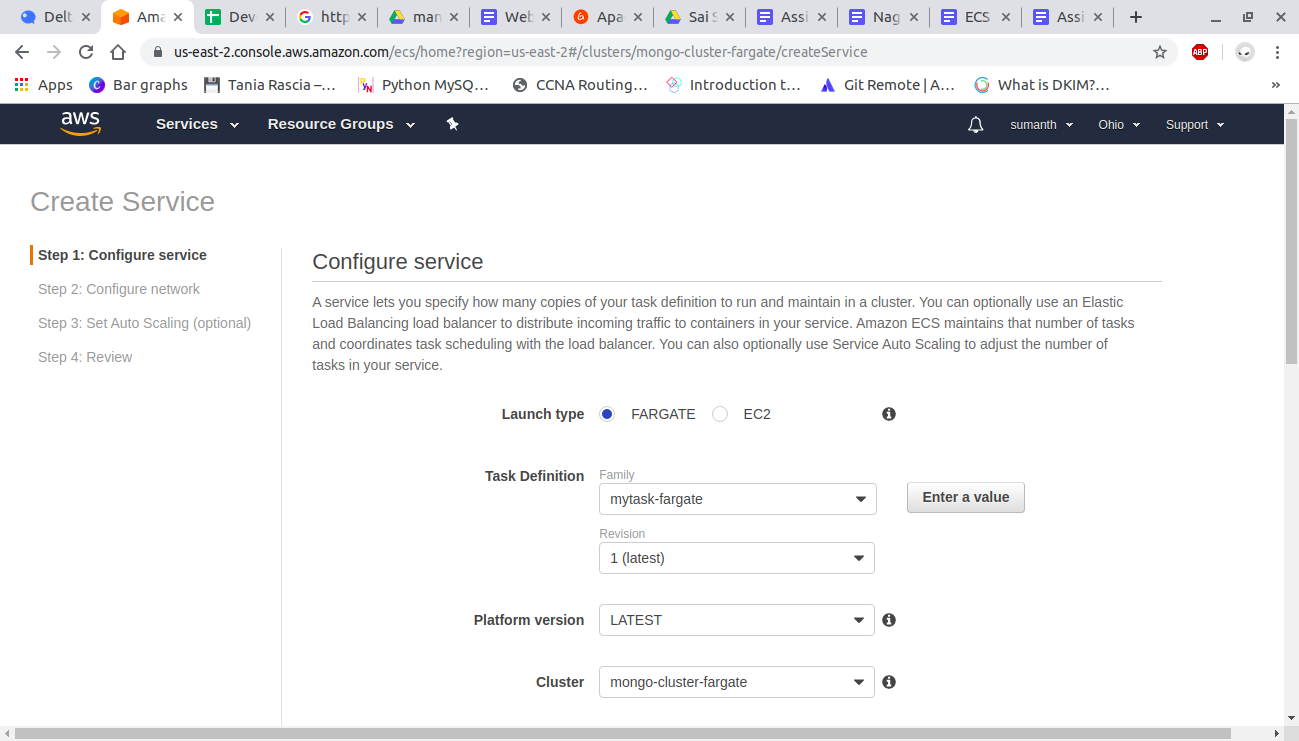


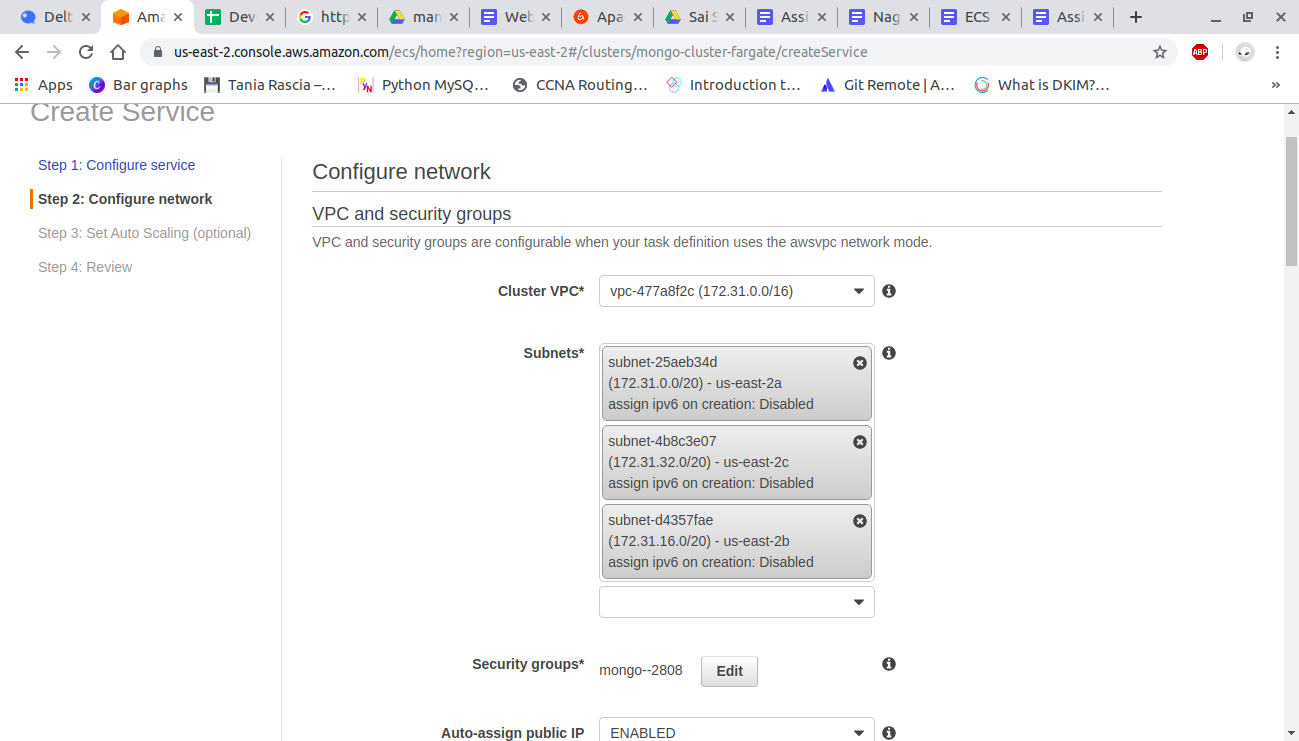
##### **Create a cluster:**





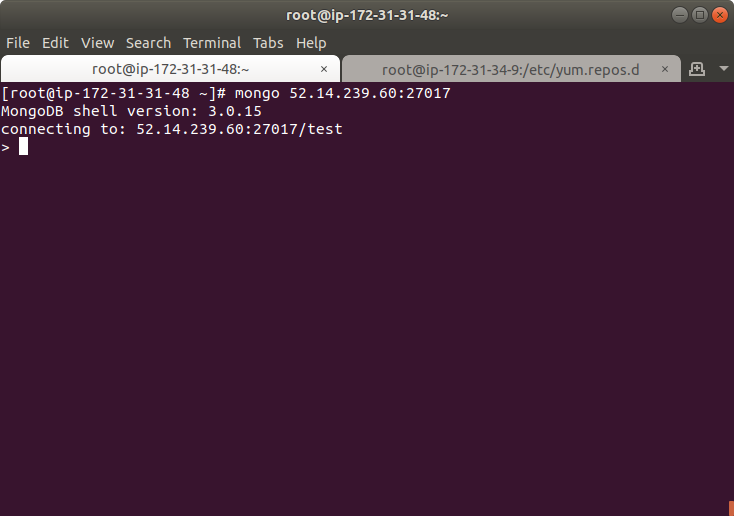
##### **Create service:**





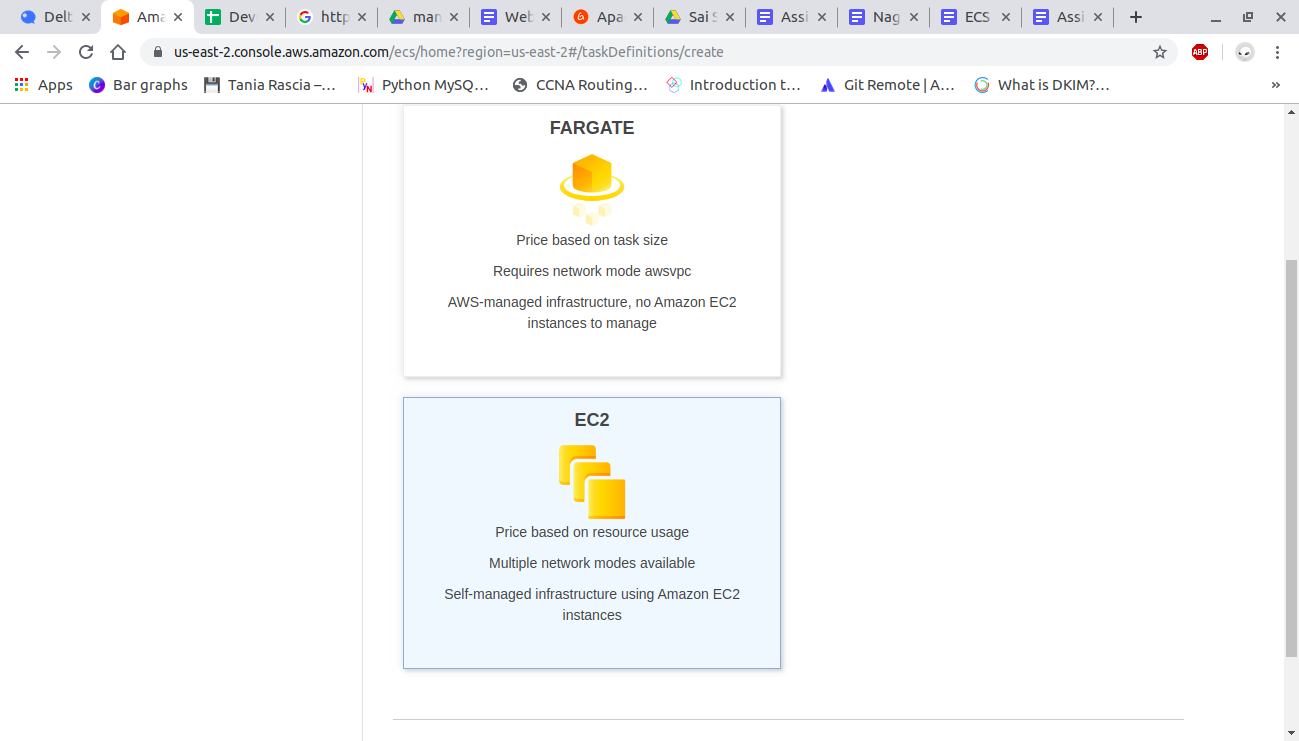
##### **Create tasks :**

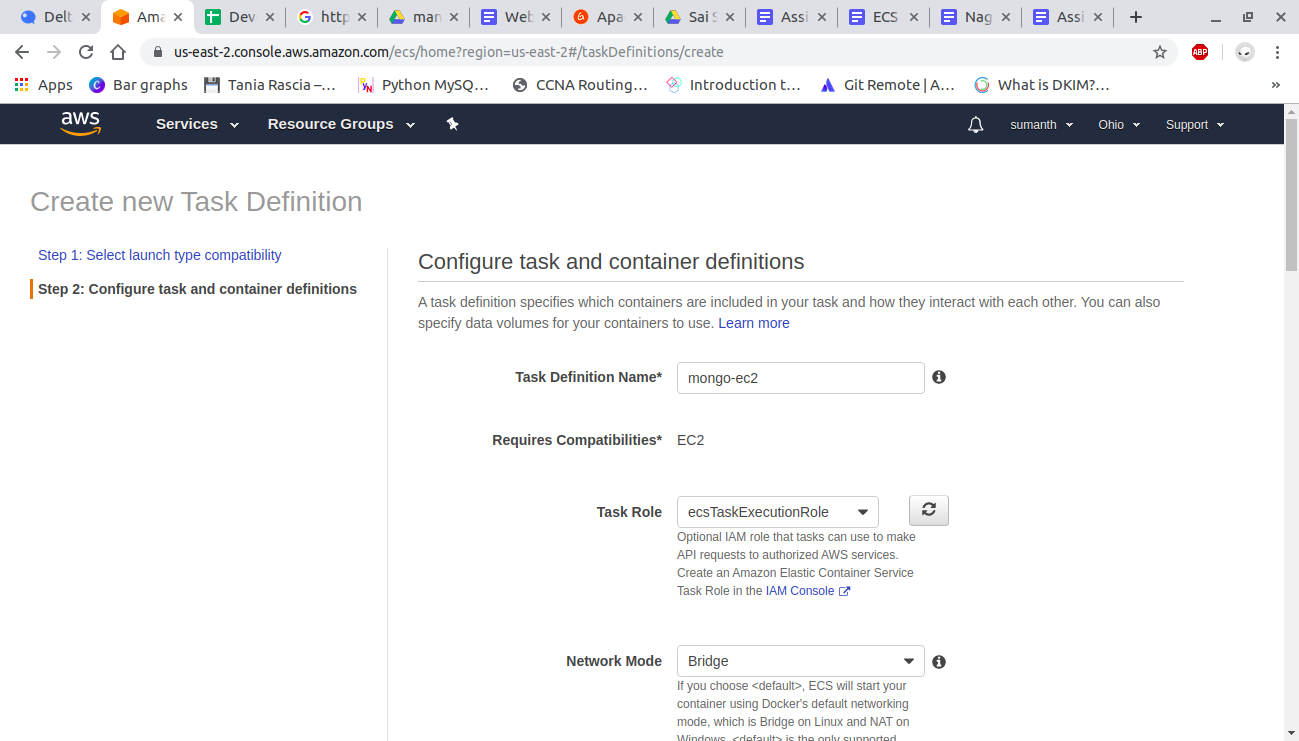
And in tasks open running tasks and copy the public ip address and open an ec2 instance and open mongo using -h and ip address.

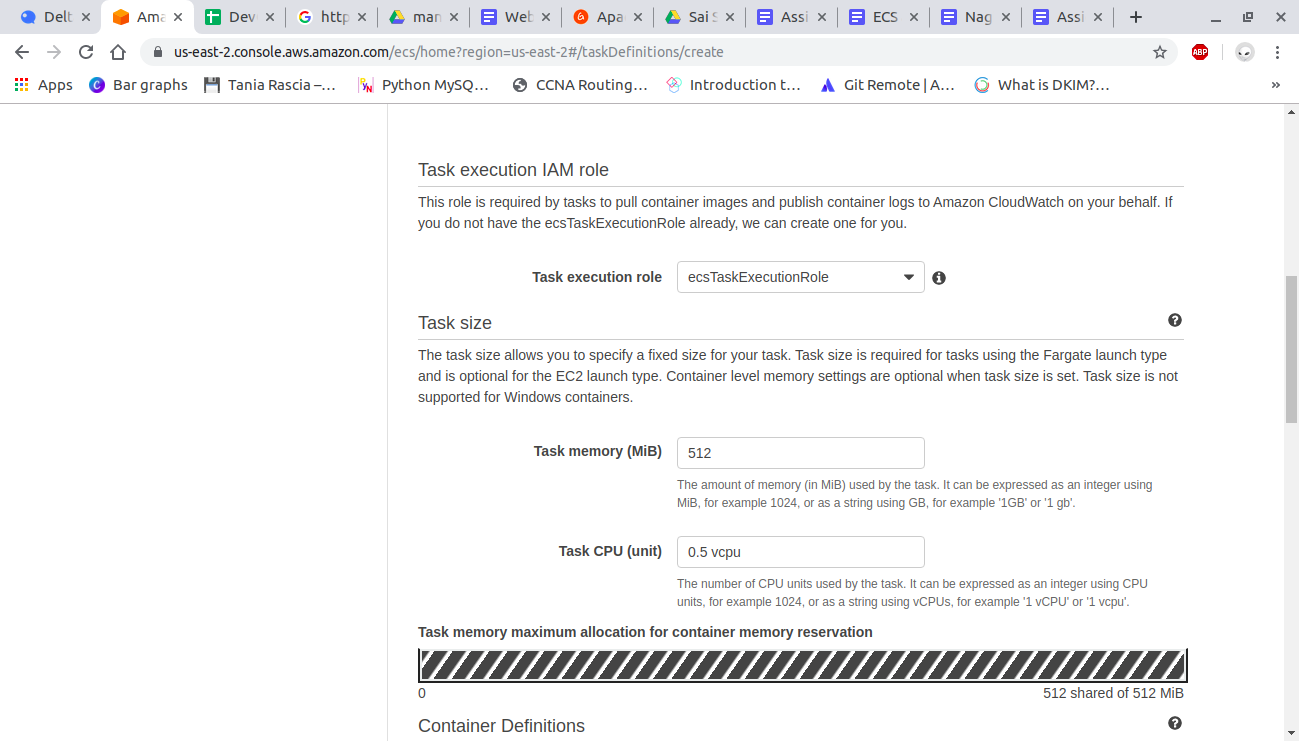


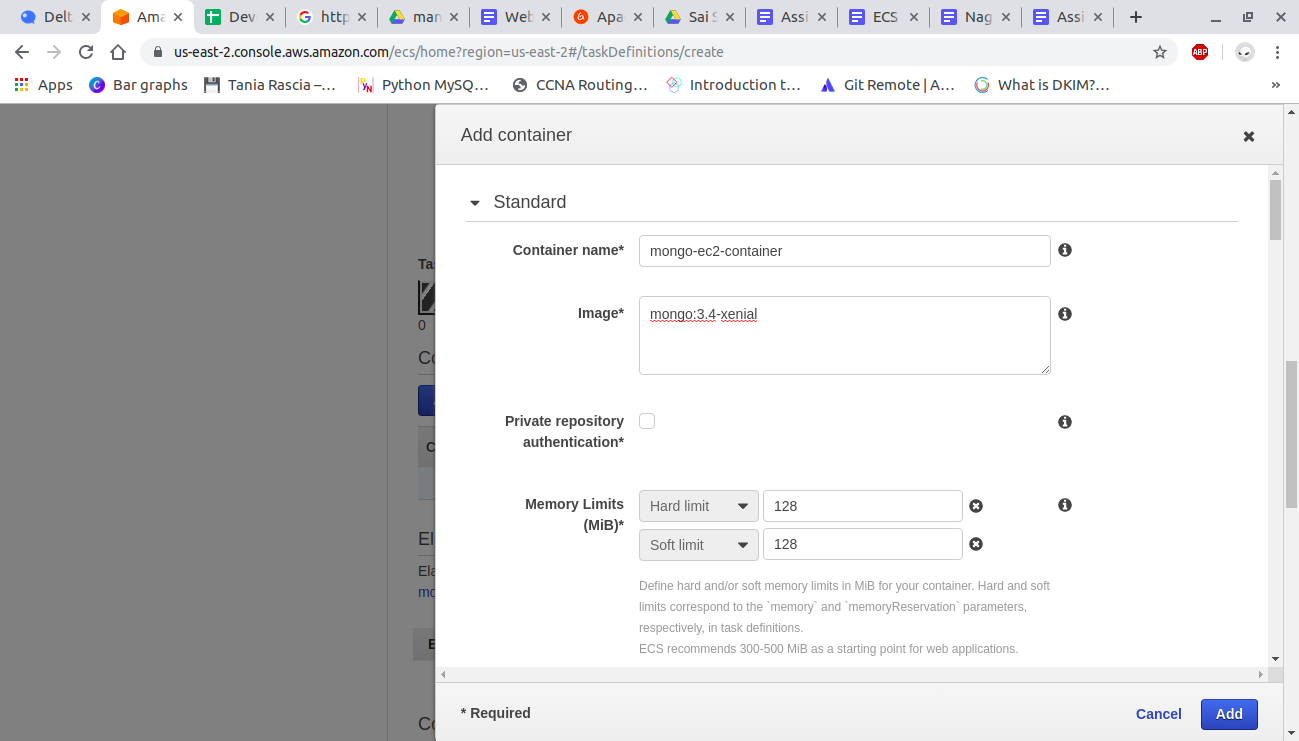
#### **ECS with EC2:**

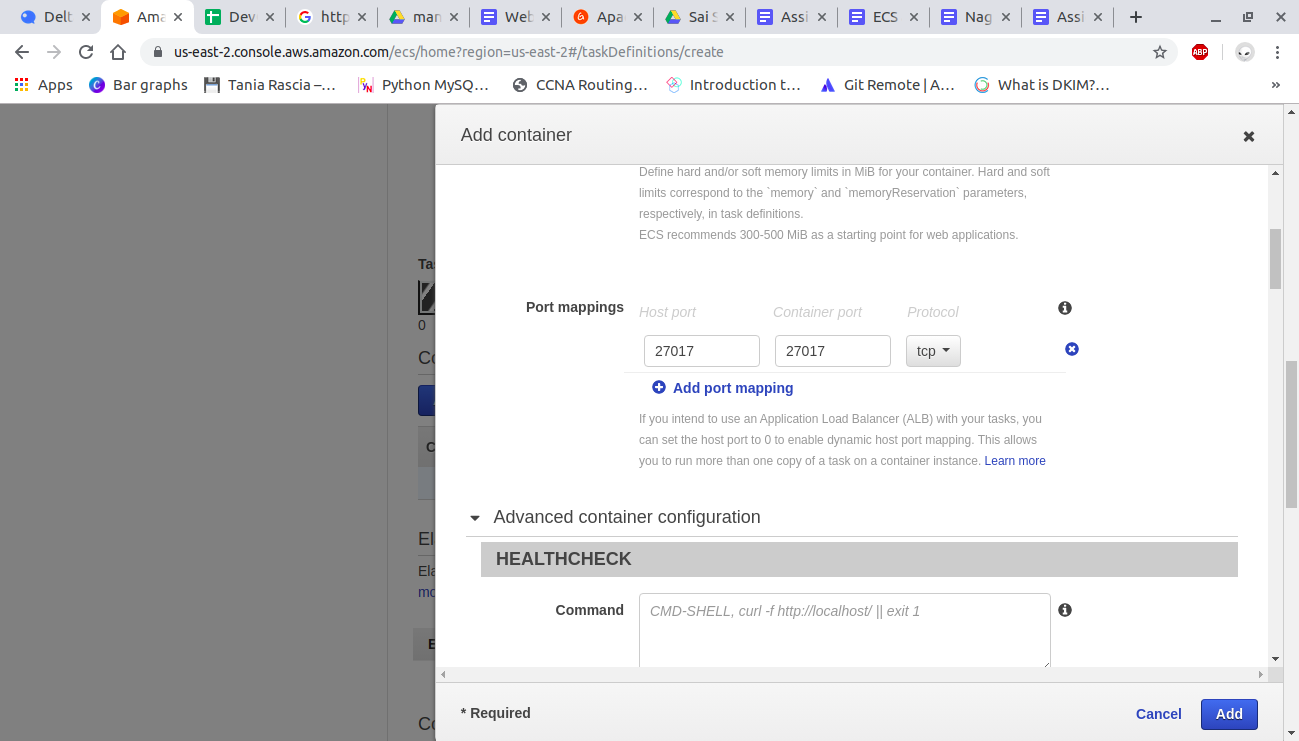
##### **Create task definition:**

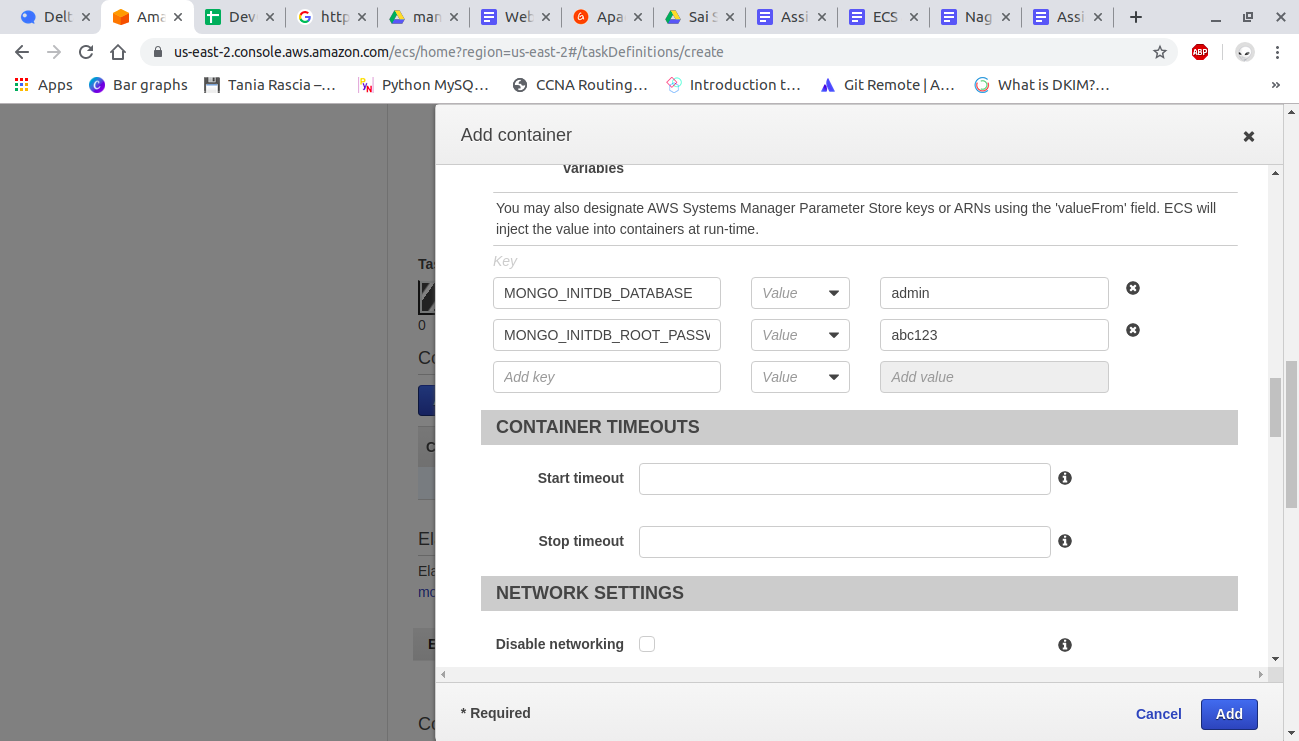


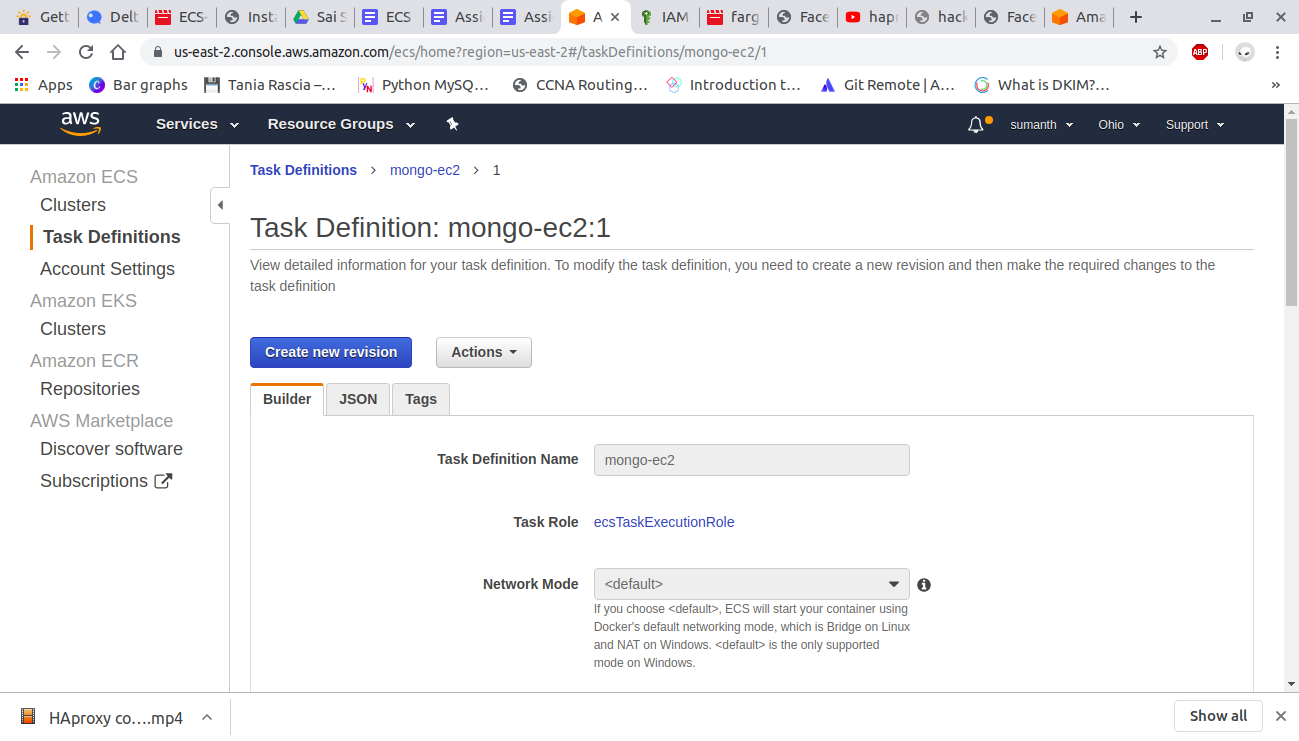




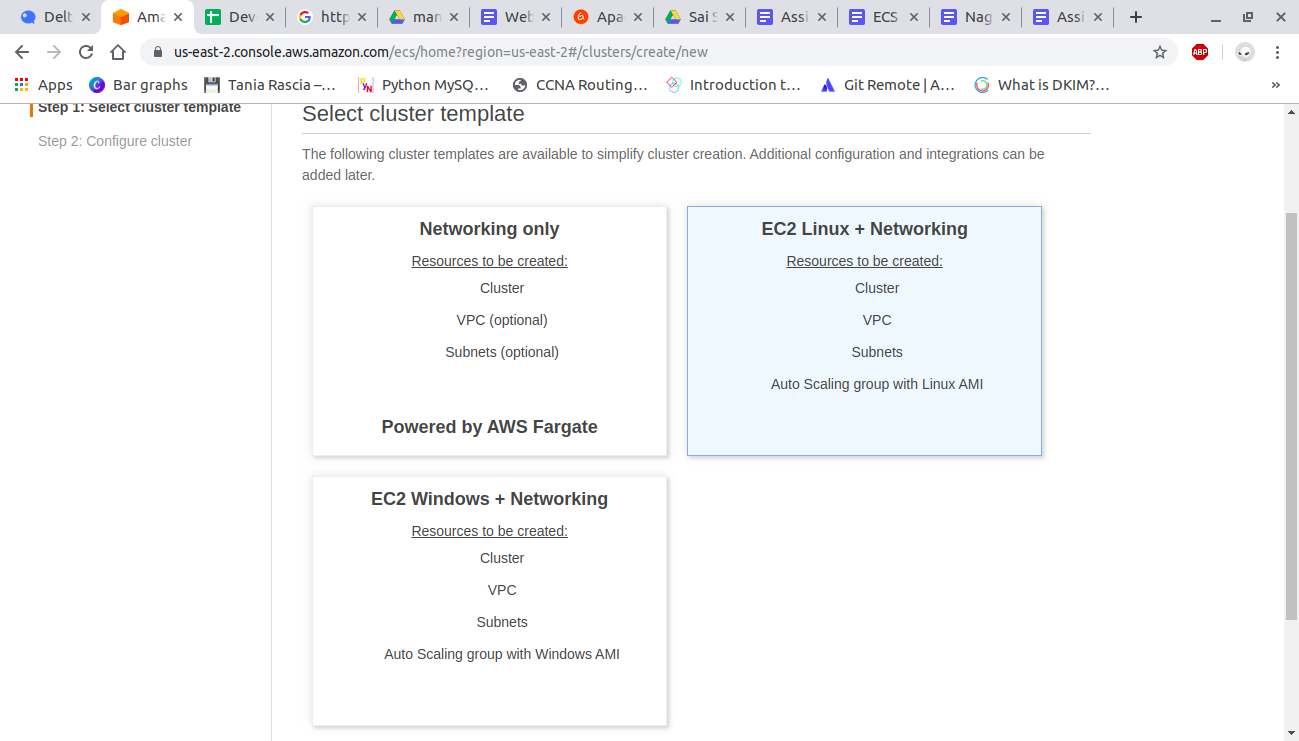


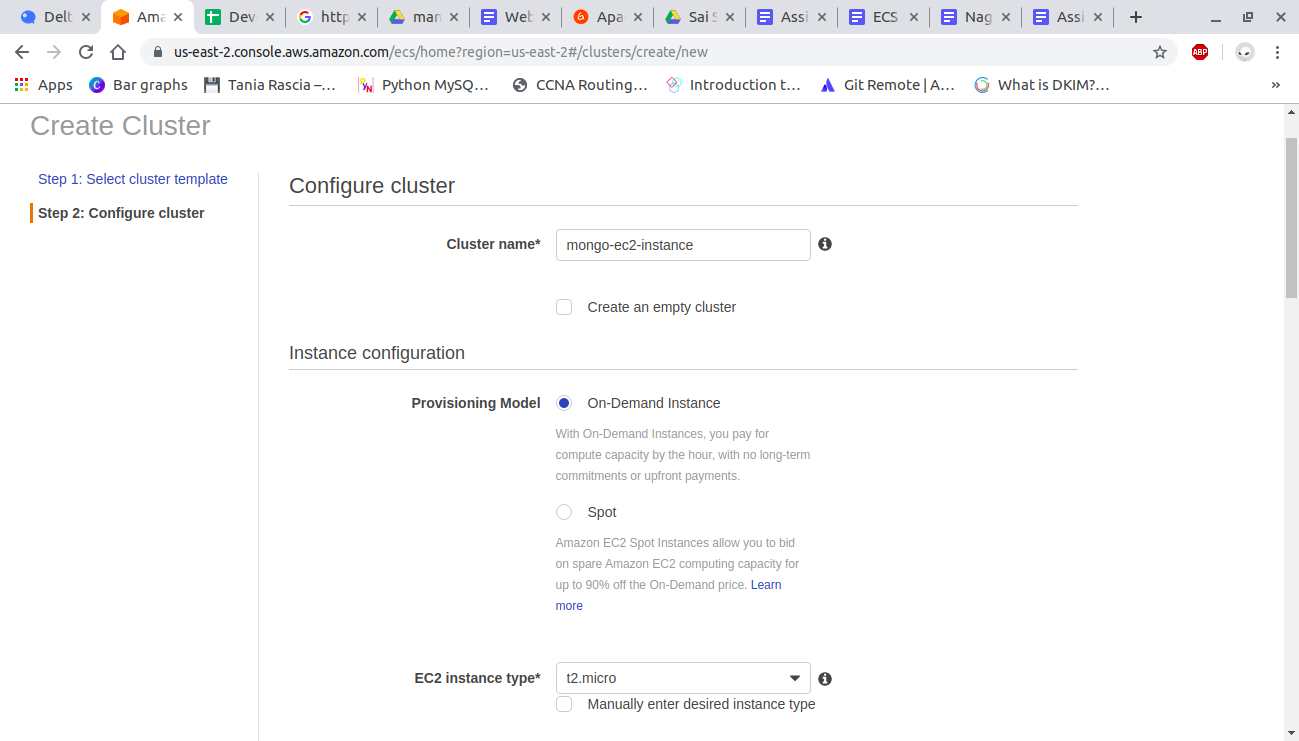


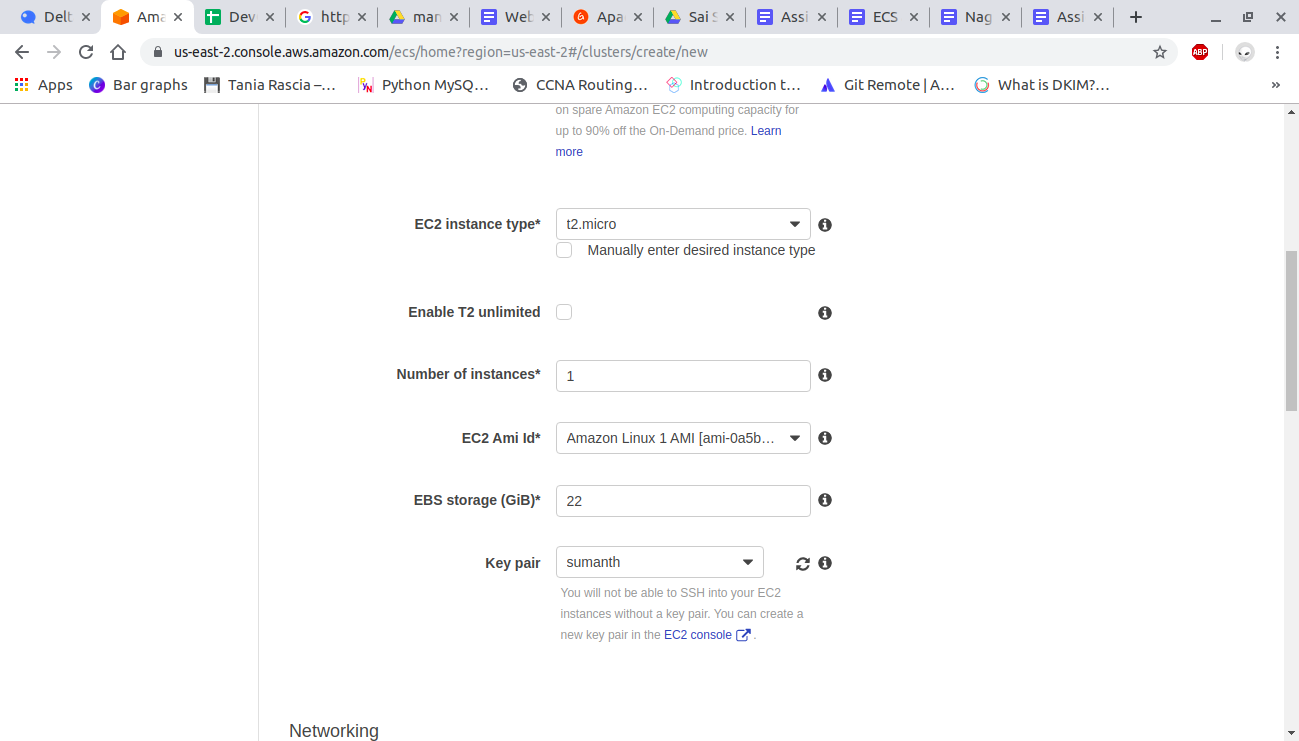


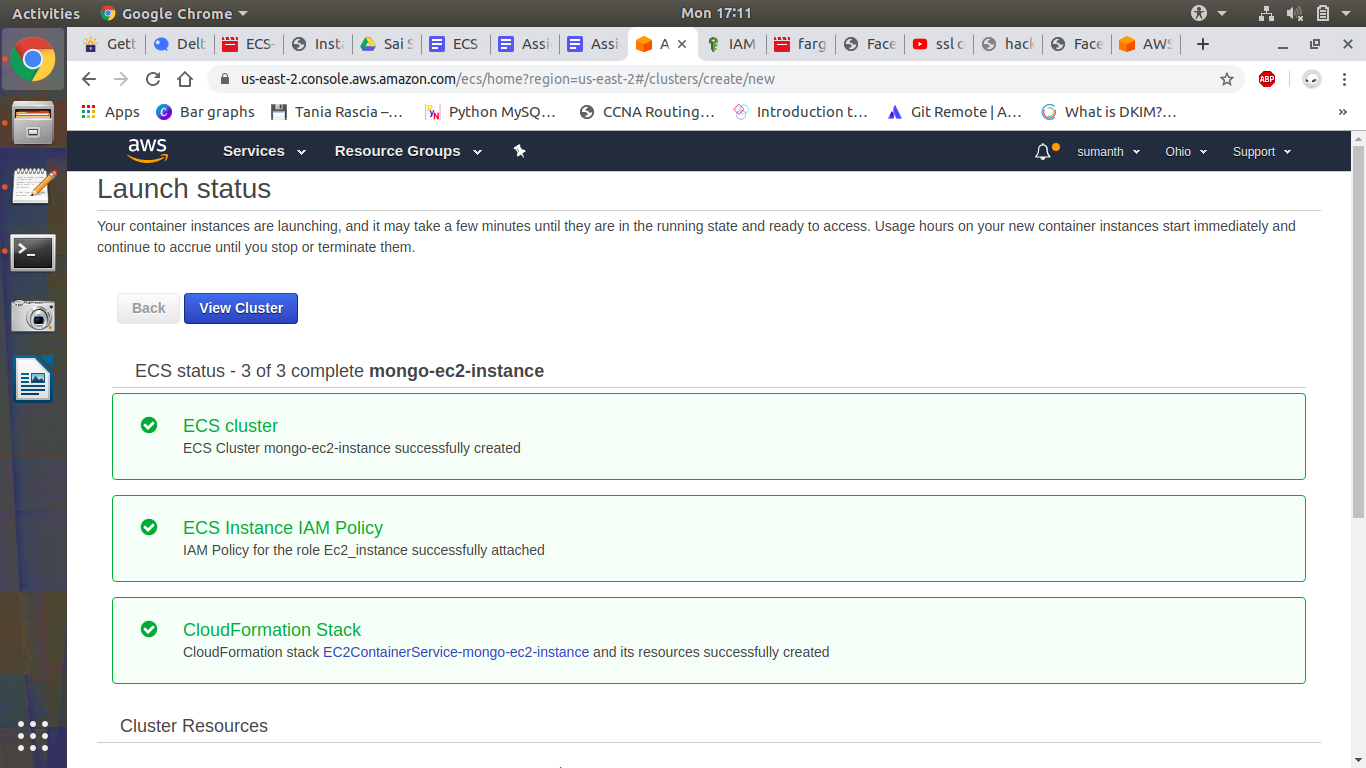


##### **Create a cluster:**

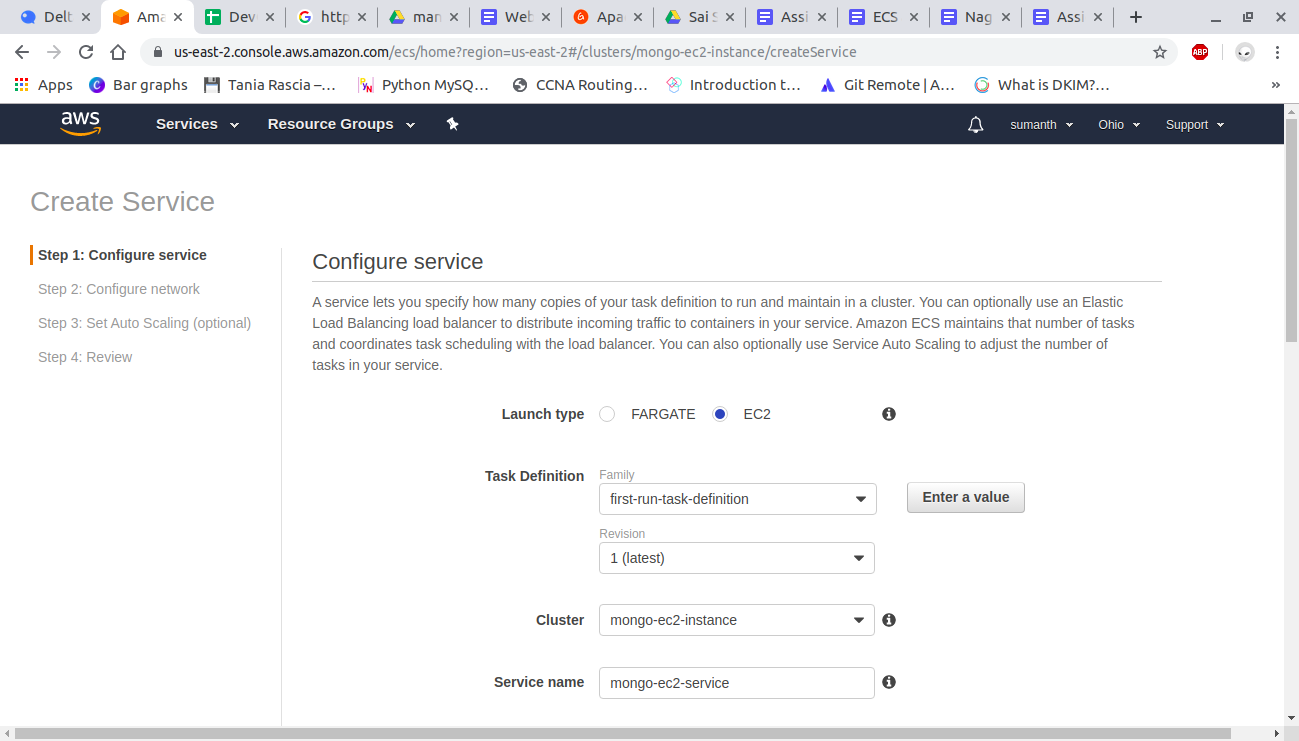




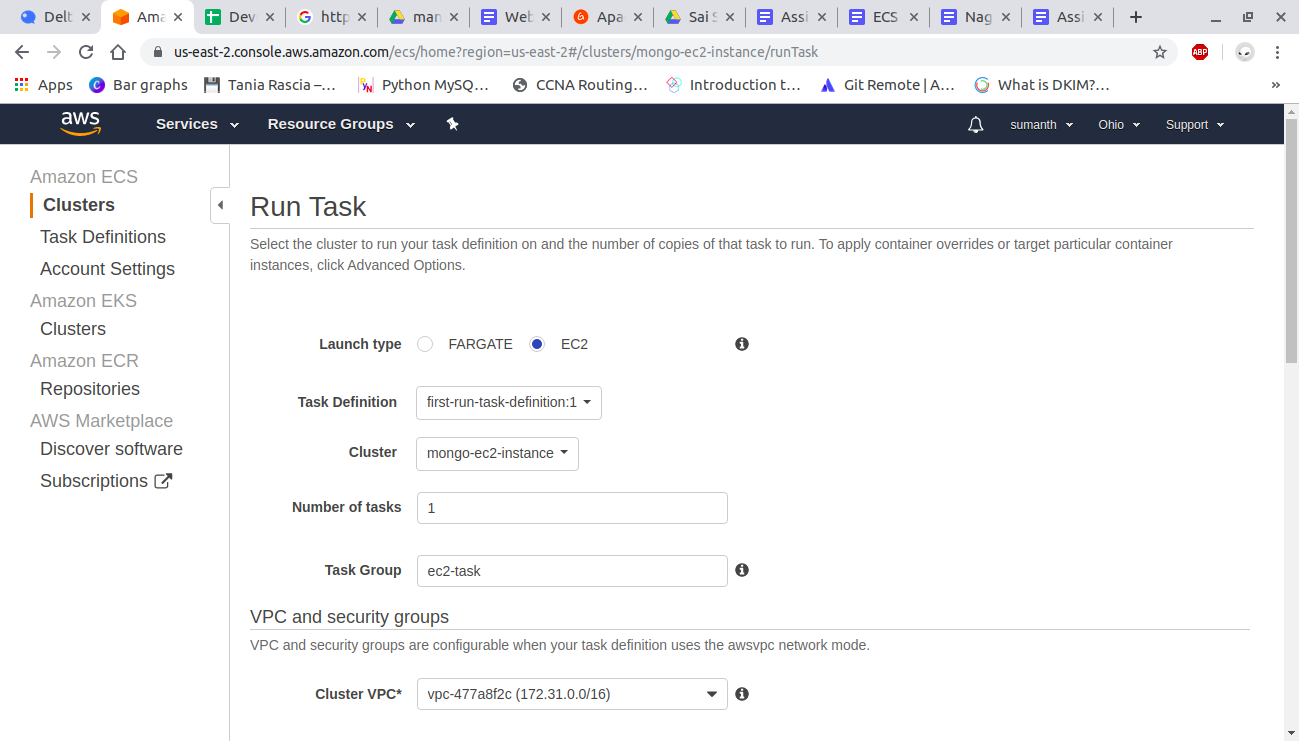




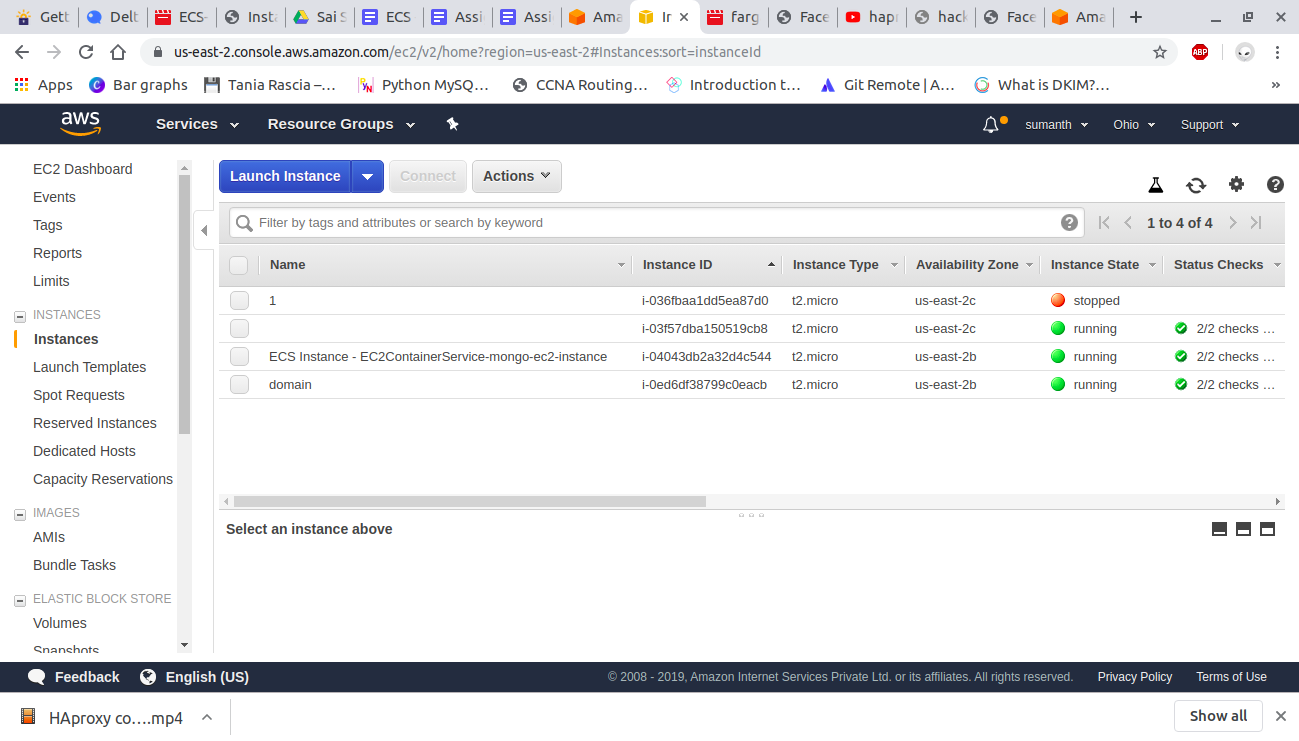
##### **Create a service:**



Create task:



Ec2 instance is created.



In that instance verify whether mongodb is running or not.

### 

### 

### 

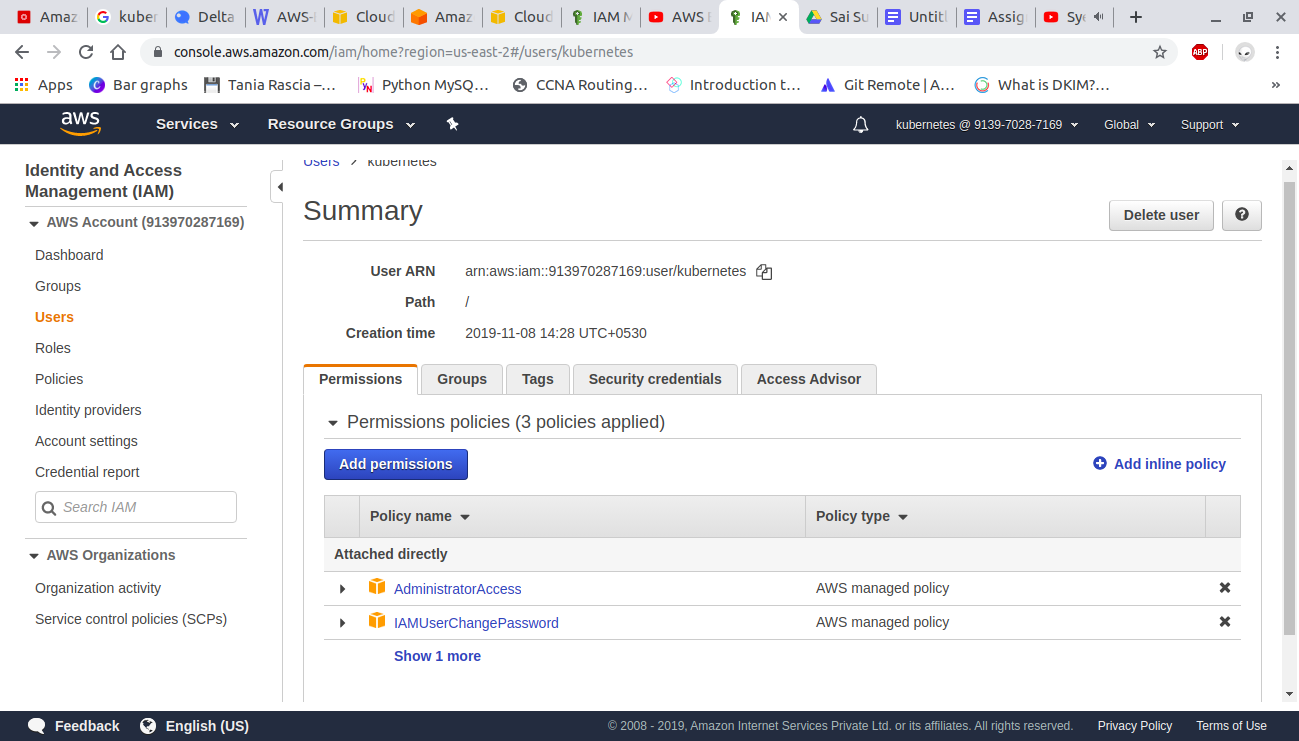
### 

### 

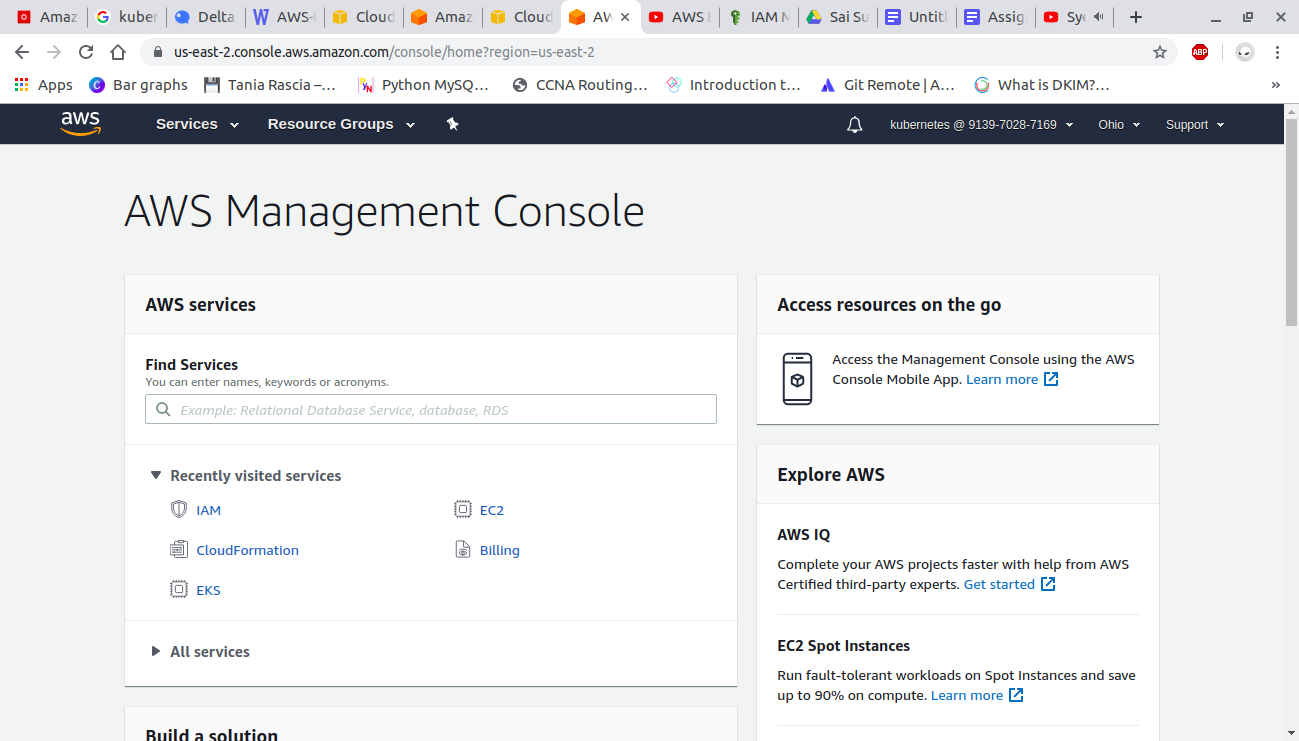
### 

### **EKS:**

Create an IAM user



And login to aws console using that user



##### **Create a VPC :**

We can directly create it using vpc service or we can use cloudformation to create a vpc

To create vpc using cloudformation you have to write a yml file

Open AWS cloudformation console

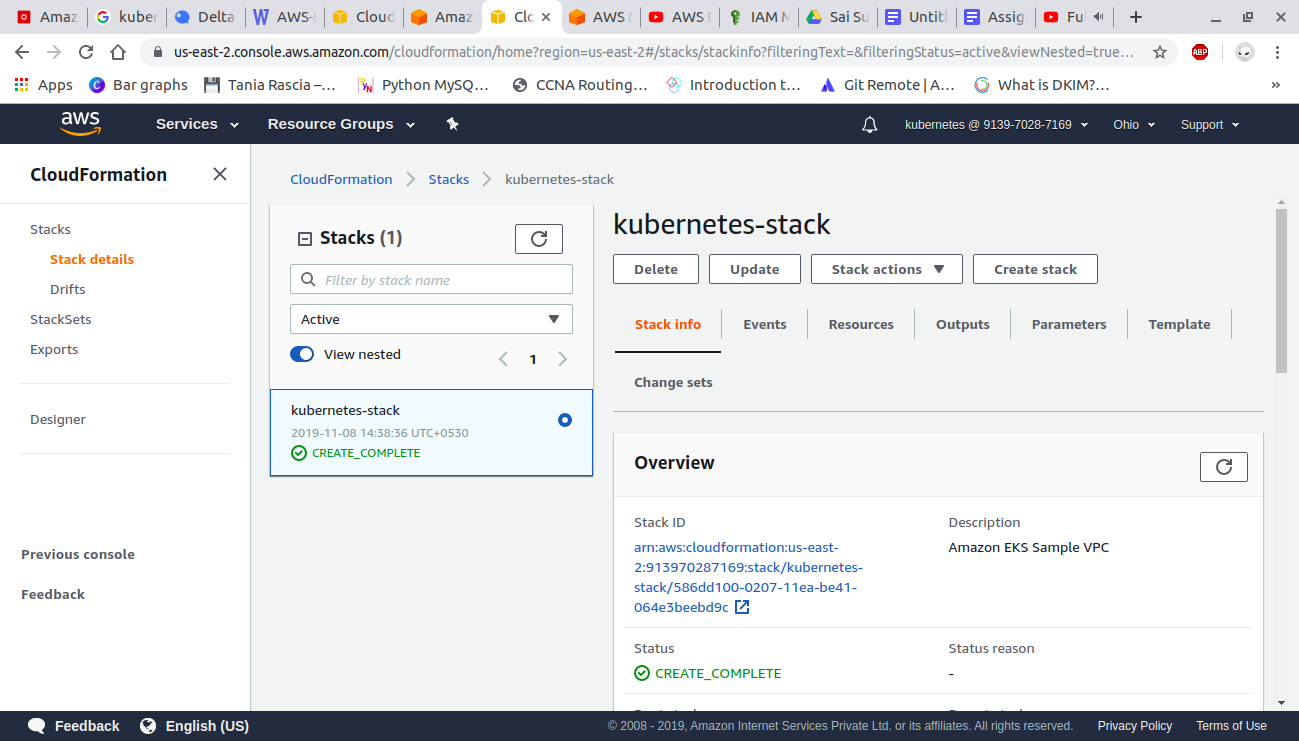
Choose create a stack

And in specify template paste the following url in amazon s3 url

*https://amazon-eks.s3-us-west-2.amazonaws.com/1.10.3/2018-06-05/amazon-eks-vpc-*

*sample.yaml*

Fill out the parameters accordingly and create a stack



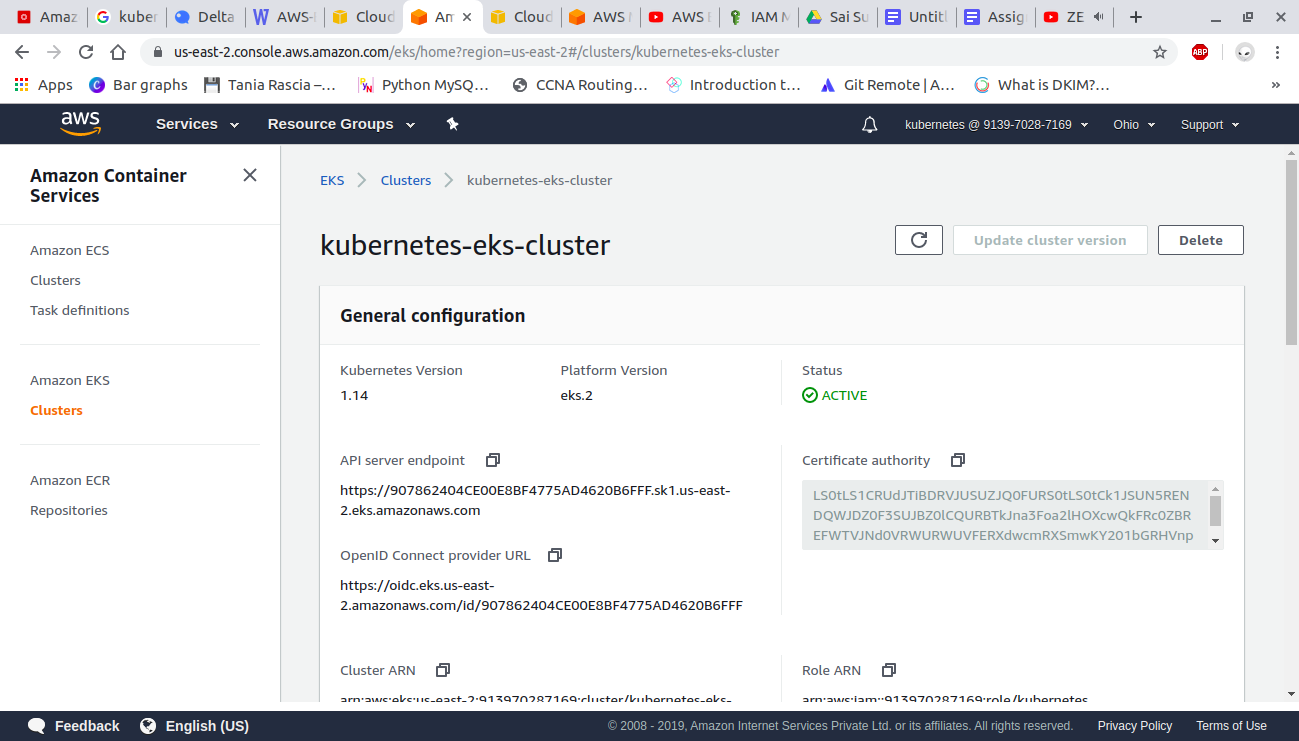
##### **Create EKS cluster:**

Open the amazon EKS console

Choose create cluster

On the Create cluster page, fill in the following fields and then choose Create:

* Cluster Name: A unique name for cluster
* Role name: Select the IAM role that you created
* VPC: Select VPC that you created
* Subnets: Select subnetId values which are recorded from vpc you created
* Security Groups: Select SG value recored from vpc

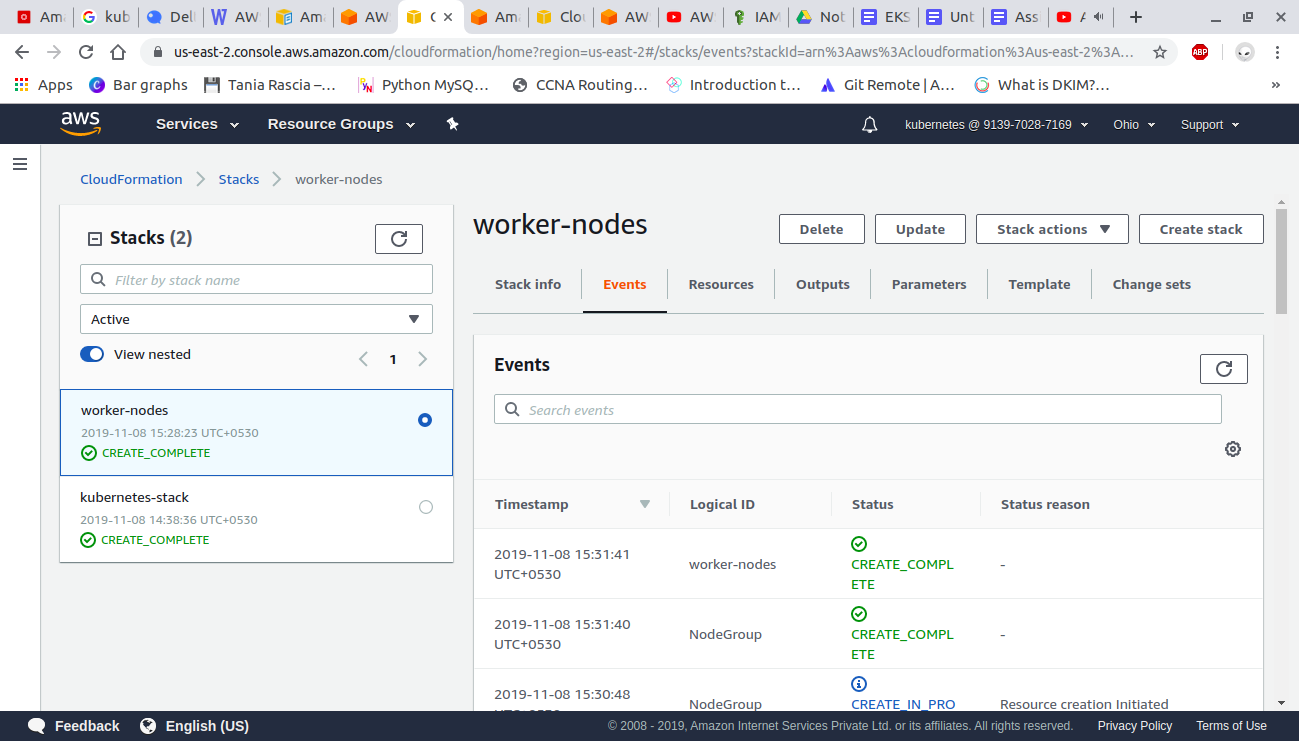


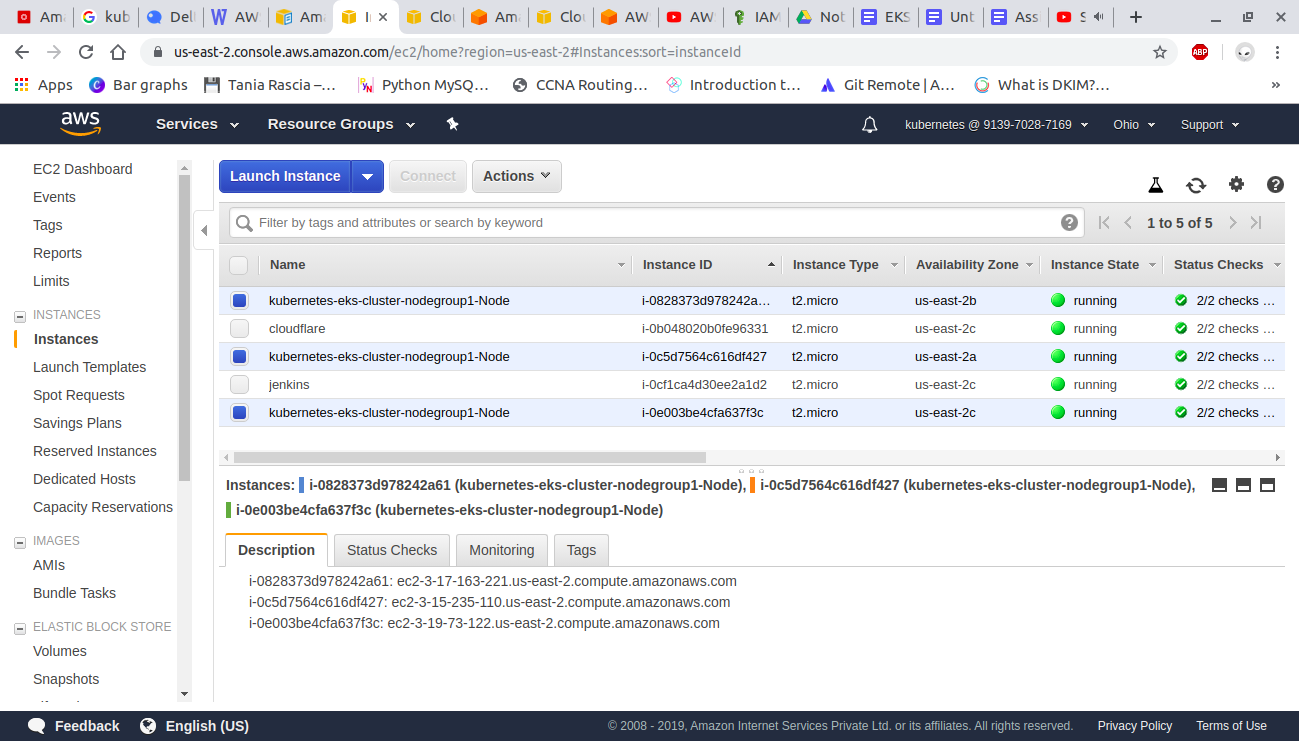
##### **Launch and configure EKS worker nodes:**

Open AWS cloudformation console and create a stack

yml file for creating worker nodes

Link - <https://docs.google.com/document/d/1KmQwMzfgyhJLzFLVesc8oxWPsw4-sMsBXBvZafKRX_0/edit?usp=sharing>





To install aws-iam-authenticator

Download the Amazon EKS-vended aws-iam-authenticator binary from Amazon S3:

*curl -o aws-iam-authenticator* [*https://amazon-eks.s3-us-west-2.amazonaws.com/1.14.6/2019-08-22/bin/linux/amd64/aws-iam-authenticator*](https://amazon-eks.s3-us-west-2.amazonaws.com/1.14.6/2019-08-22/bin/linux/amd64/aws-iam-authenticator)

Apply execute permissions to the binary.

*chmod +x ./aws-iam-authenticator*

Copy the binary to a folder in your $PATH. We recommend creating a $HOME/bin/aws-iam-authenticator and ensuring that $HOME/bin comes first in your $PATH.

*mkdir -p $HOME/bin && cp ./aws-iam-authenticator $HOME/bin/aws-iam-authenticator && export PATH=$HOME/bin:$PATH*

Add $HOME/bin to your PATH environment variable.

*echo 'export PATH=$HOME/bin:$PATH' >> ~/.bashrc*

Test that the aws-iam-authenticator binary works.

*aws-iam-authenticator help*

##### **Create, Launch and configure EKS VPC, cluster and worker nodes VIA EKSCTL utility:**

Refer documentation also : eksctl.io

Download and extract the latest release of eksctl with the following command.

*curl --silent --location "https://github.com/weaveworks/eksctl/releases/download/latest\_release/eksctl\_$(uname -s)\_amd64.tar.gz" | tar xz -C /tmp*

Move the extracted binary to /usr/local/bin.

*sudo mv /tmp/eksctl /usr/local/bin*

*eksctl version*

Step 2: Create Cluster

In your terminal,

*eksctl create cluster --name=demo-eks-test --nodes=2 --region=us-west-2*

It will create the cluster on aws within 10-15 minutes, please give the name as per your choice and

choose location accordingly.

Step 3: Delete Cluster

*eksctl delete cluster --region=us-west-2 --name=demo-eks-test*

Kubetcl Common Commands for Connectivity via Terminal:

*kubectl get nodes* #get-nodes

*kubectl create -f nginx.yaml*. #configuration-apply

*kubectl get services -o wide* #getall-services

*kubectl desribe svc nginx-deployment* #describe-service

*kubectl get deployment* #get-deployment-details

*kubectl get rs* #get-replica-details

kubectl describe deployment | grep Image #get-deployment-image

kubectl scale deployment.v1.apps/nginx-deployment --replicas=9 #scale-pods