Problem Statement:

You work for XYZ Corporation that uses on premise solutions and a limited number of systems. With the increase in requests in their application, the load also increases. So, to handle the load the corporation must buy more systems almost on a regular basis. Realizing the need to cut down the expenses on systems, they decided to move their infrastructure to AWS.

Tasks To Be Performed:

1. Manage the scaling requirements of the company by:

a. Deploying multiple compute resources on the cloud as soon as the load increases and the CPU utilization exceeds 80%

b. Removing the resources when the CPU utilization goes under 60%

2. Create an Application Load balancer to distribute the load between compute resources

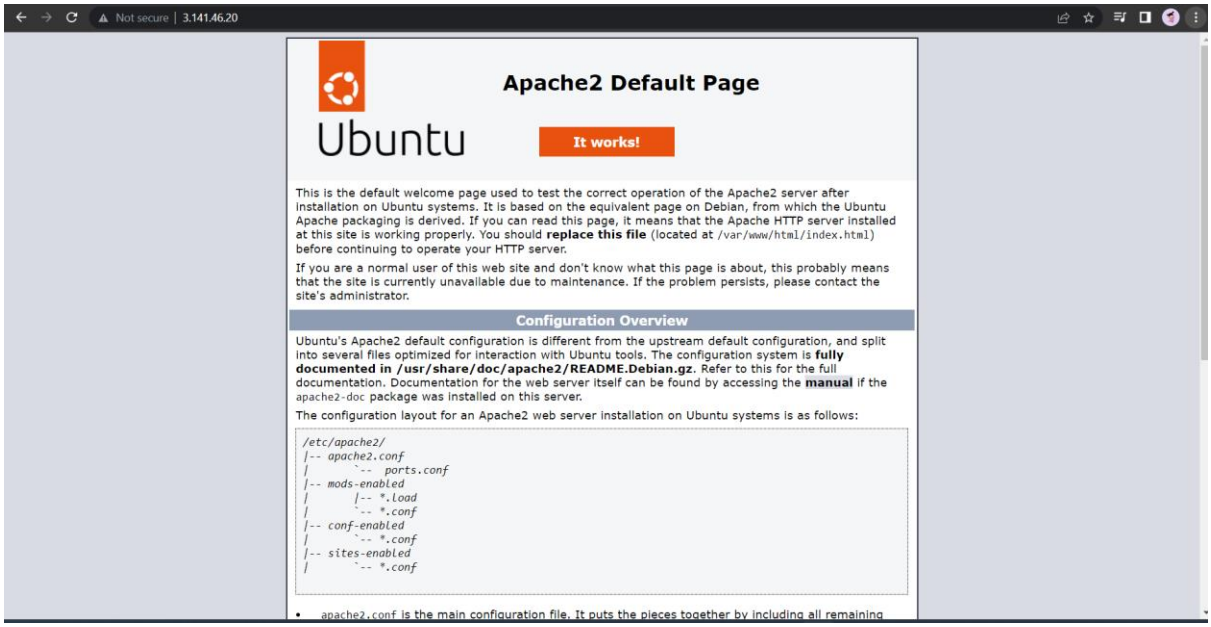
• In your two target groups, make one for Blue deployment and the other for Green

• Use weighted routing to route 70% of the traffic to the Blue target group and 30% of the traffic to the Green target group 3. Route the traffic to the company's domain Note:

You can get a free domain from Freenom//now a days freenom is not providing the free domain names

Procedure: -

- To manage the scaling policy the basic think we need to have EC2 intance to create an AMI.
- Create an instance with the name "ASG_myinstance" with the Ubuntu server and let everything be default.
- Select the instance and click on connect.
- Run the following command.
    - sudo apt-get update
    - sudo apt-get install apache2 -y
- Now if we copy the public IP of the machine to the search engine, we will see the default page of the apache2.



- Now our task is to create an AMI of this instance. Therefore, now select the intance and click on action and click “image and templates” and “create image.”
- Give the desired name to the image. Let us say and “ASG_myAMI.” Click on “create image”
- Therefore, an image is created.
- Now we must “Launch configuration”
- **Note**

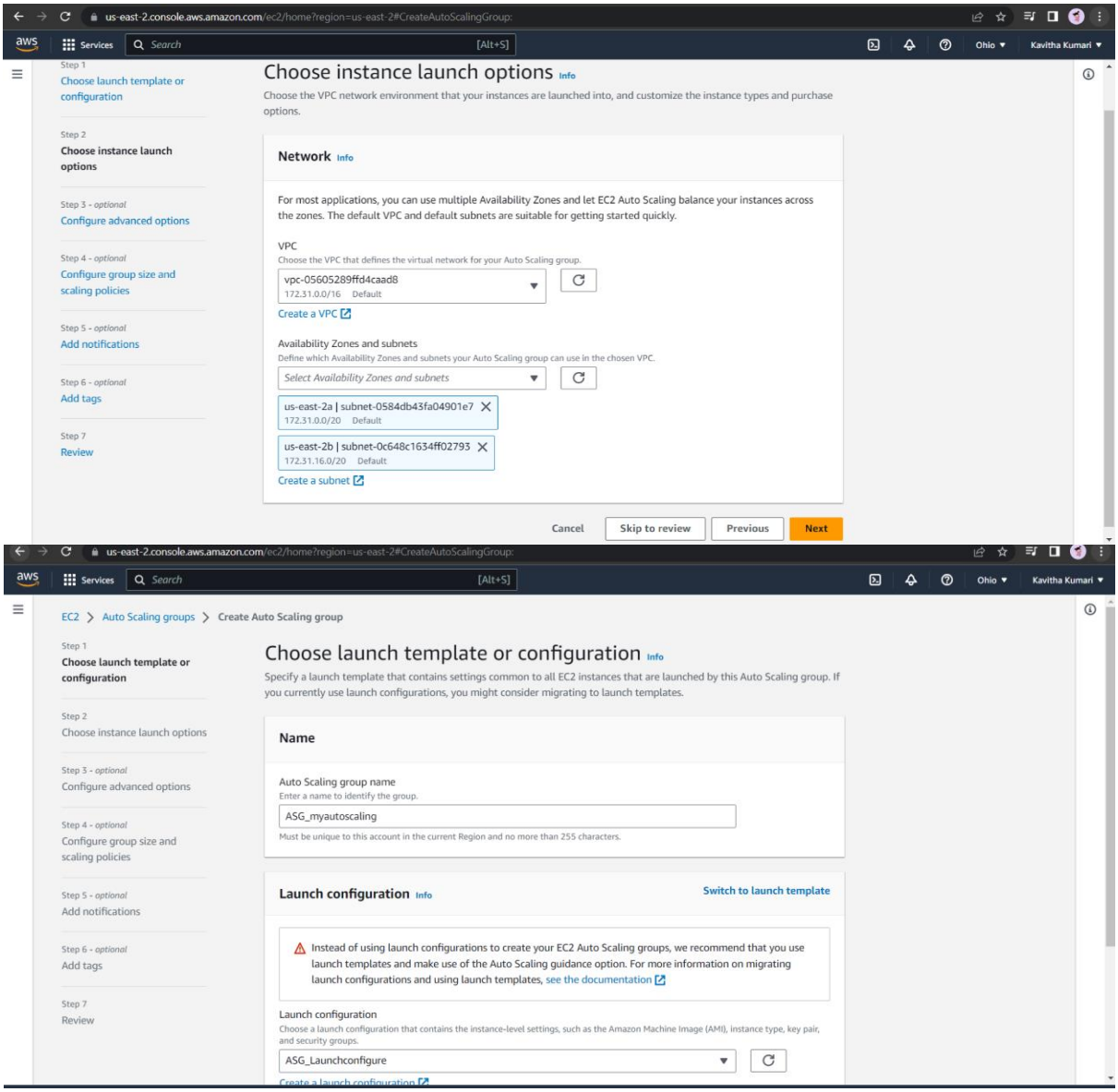
Create a launch configuration

PDF | RSS

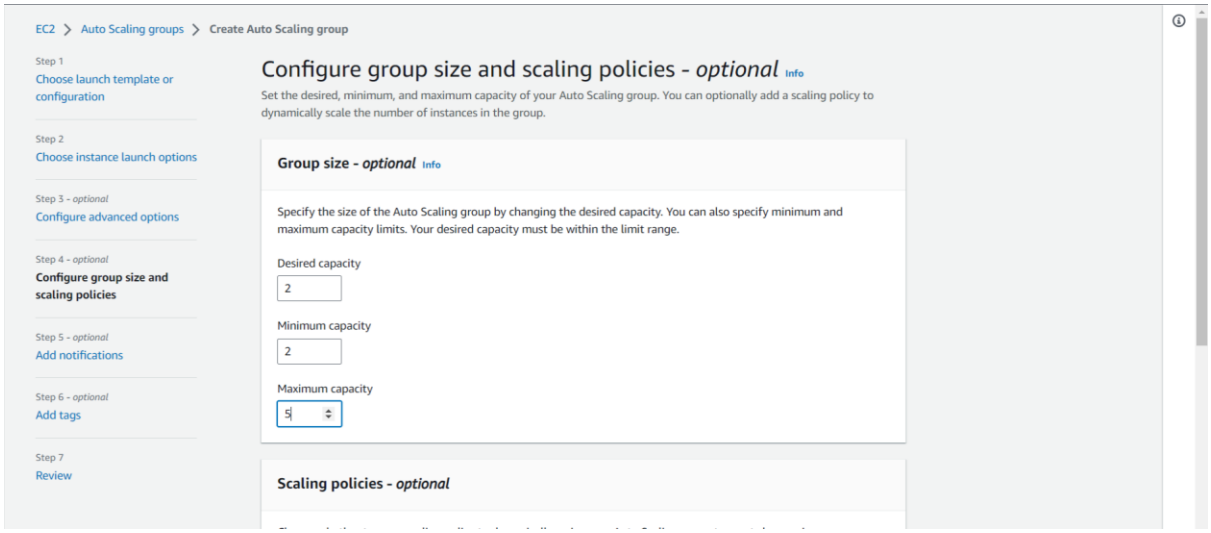
Important

Launch configurations no longer add support for new Amazon EC2 instance types that are released after **December 31, 2022**. In addition, any new accounts created on or after **June 1, 2023** will not have the option to create new launch configurations through the console. However, API, CLI, and CloudFormation access will be available to new accounts created between **June 1, 2023** and **December 31, 2023** to support customers with automation use cases. New accounts created on or after **January 1, 2024** will not be able to create new launch configurations by using the console, API, CLI, and CloudFormation. For information about migrating your Auto Scaling groups to launch templates, see Migrate to launch templates.

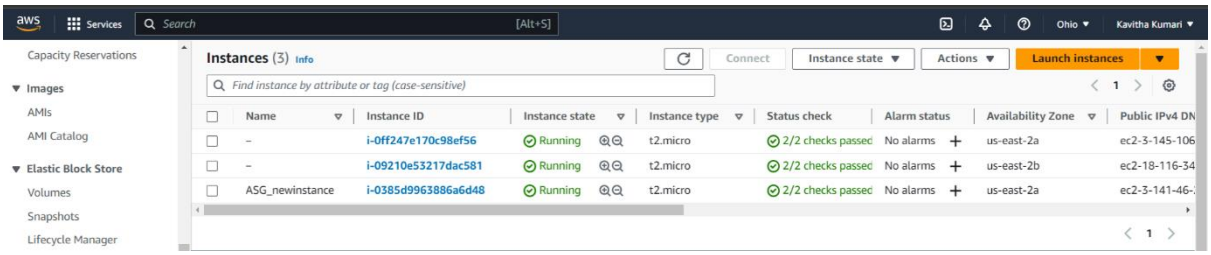
- To launch configuration, navigate to autoscaling and click on “Autoscaling Groups” and from there select “switch to launch template” and then select create a launch configuration” and then we will get the option of create launch configuration.
- Give the name for the configuration example as ASG_Launchconfigure. Select the AMI we created and select the instance type as “t2.micro” and add a new rule in security groups of HTTP and select a keypair or you can also proceed without the keypair.
- Now navigate to autoscaling groups and give the name to the autoscaling group and click on switch to Launch confiration and select the select the launch configuration we created.



- Next, we need to choose the VPC network environment that your instances are launched into, and customize the instance types and purchase options. We can specify the availability zone where our instance can be launched.
- Click on next
- In step 3 let every thing be default and click on next.
- In step 4 we need the choose the desired, minimum and maximum capacity of the instance.



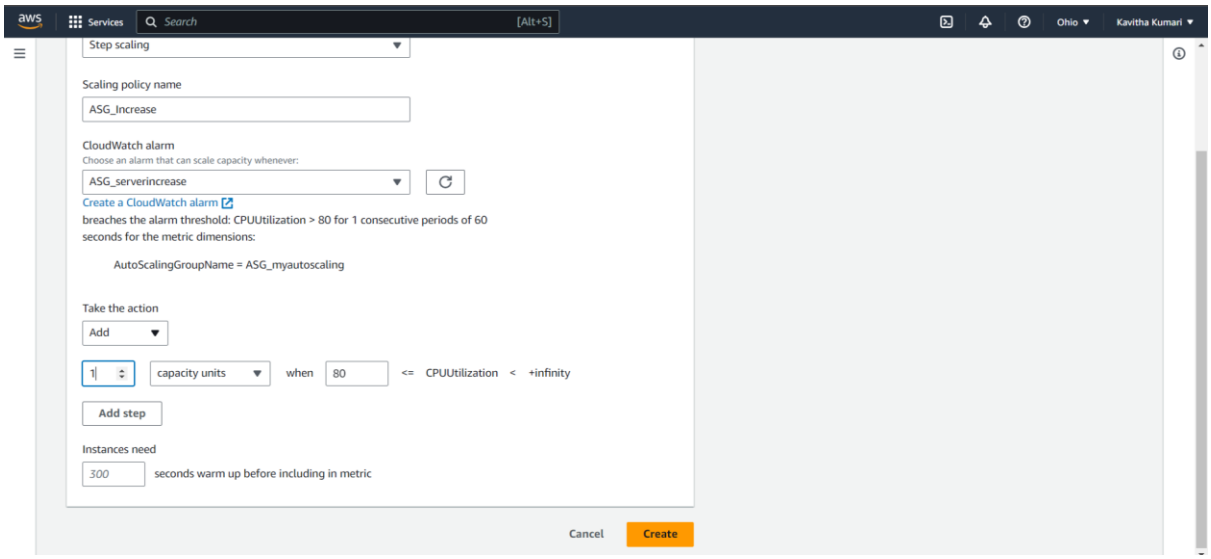
- Rest let everything be default and click on next.
- You can send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group. But right now, I am leaving this option and simply clicking on next.
- We can add tags to help you search, filter, and track your Auto Scaling group across AWS. You can also choose to automatically add these tags to instances when they are launched.
- Step 7 is to review the Autoscaling group and click on create autoscaling group.
- According to the Autoscaling Policy 2 instances will be created initially and the apache2 will

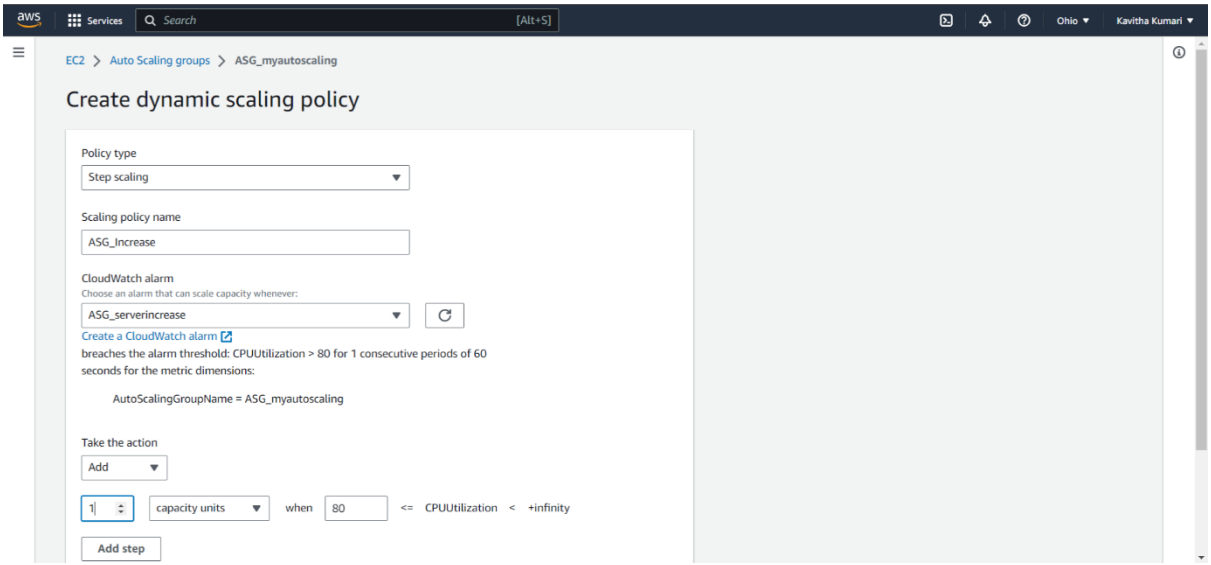


be installed in it. When we select the IP address of that instances and search that IP address, we will see the default page of apache2 server.

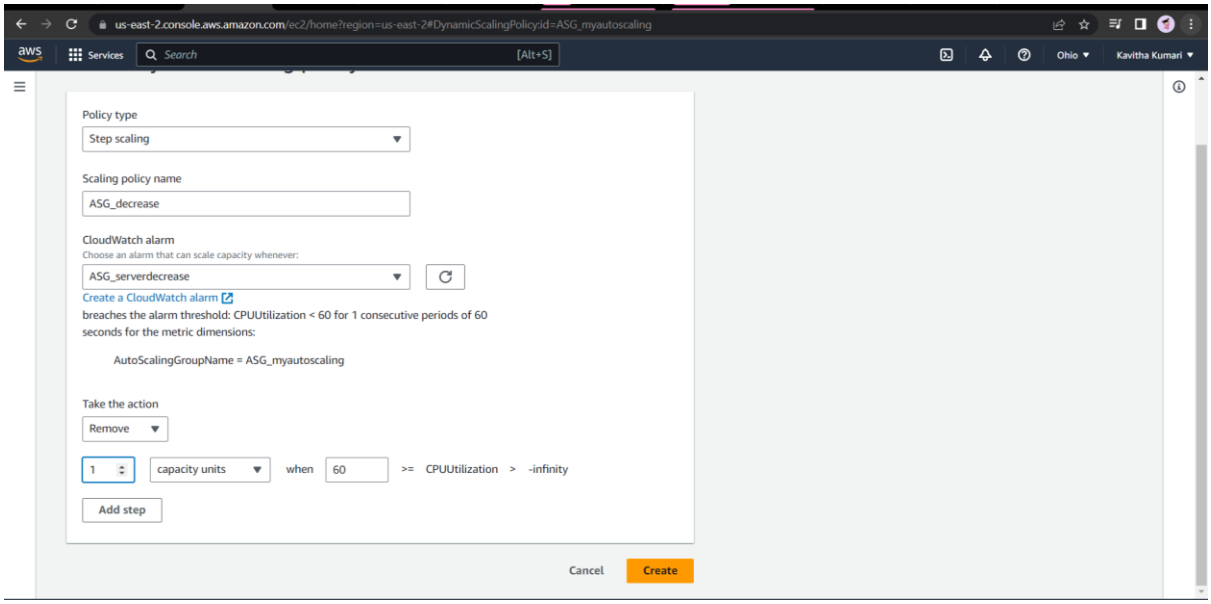
- Now to define the dynamic scaling policies we need to go Automatic scaling and there you will find the dynamic scaling policy. Click on create dynamic scaling policy.
- Choose the policy type as “step scaling” and give the scaling policy name.
- In the next step we need to create an alarm watch for that click on “create an alarm watch.” Click on “select metric” and select EC2 and than By Autoscaling group because the alarm should work upon the instances which are created by the autoscaling groups and next select the “ASG_myautoscaling ---CPU utilisation” because our task is when the cpu utilisation is more than 80% the alarm should be triggered. Let us choose the default period of alarm as “1 minute” let the threshold be static and the threshold value be 80 click on next.
- In the next step remove the notification and click on next.
- In the next step give the name to the alarm as “ASG_serverincrease”
- In the next step review the details of alarm and click on “create alarm.”

- Now when you come to the page of dynamic scaling policy and refresh the page you will be able to select the alarm.





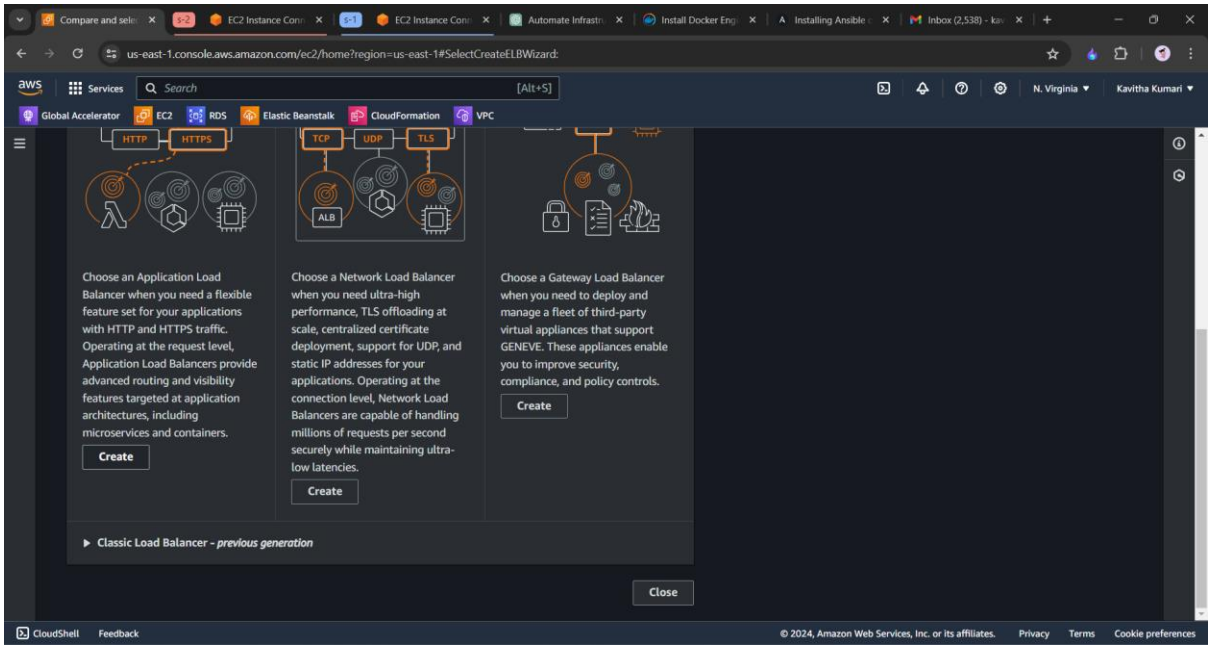
- In the next step we need to select the action for the alarm that the triggered at the point of more than 80% of the cpu utilisation. That is we need to add one instance.
- Now click on create.
- Now we need to also define the policy for the decrease of the cpu utilisation.
- Therefore, we will create another policy of the name ASG_decrease and we will create an alarm named “ASG_serverdecrease. Where the cpu utilisation must be lower than 60.

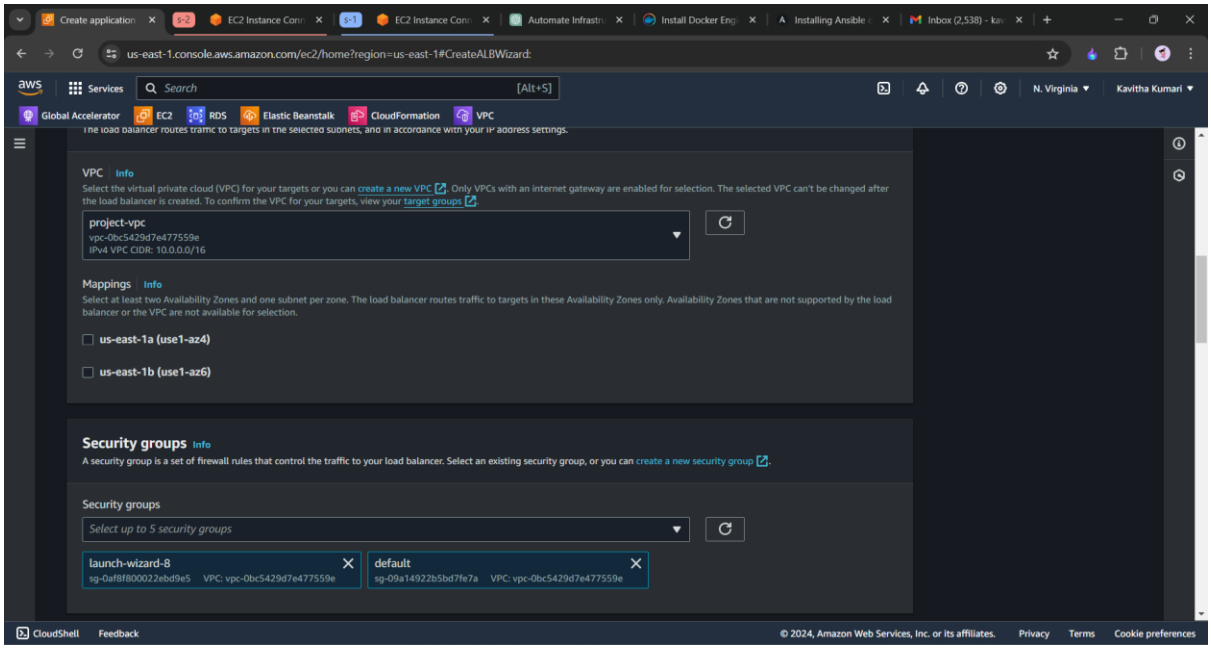


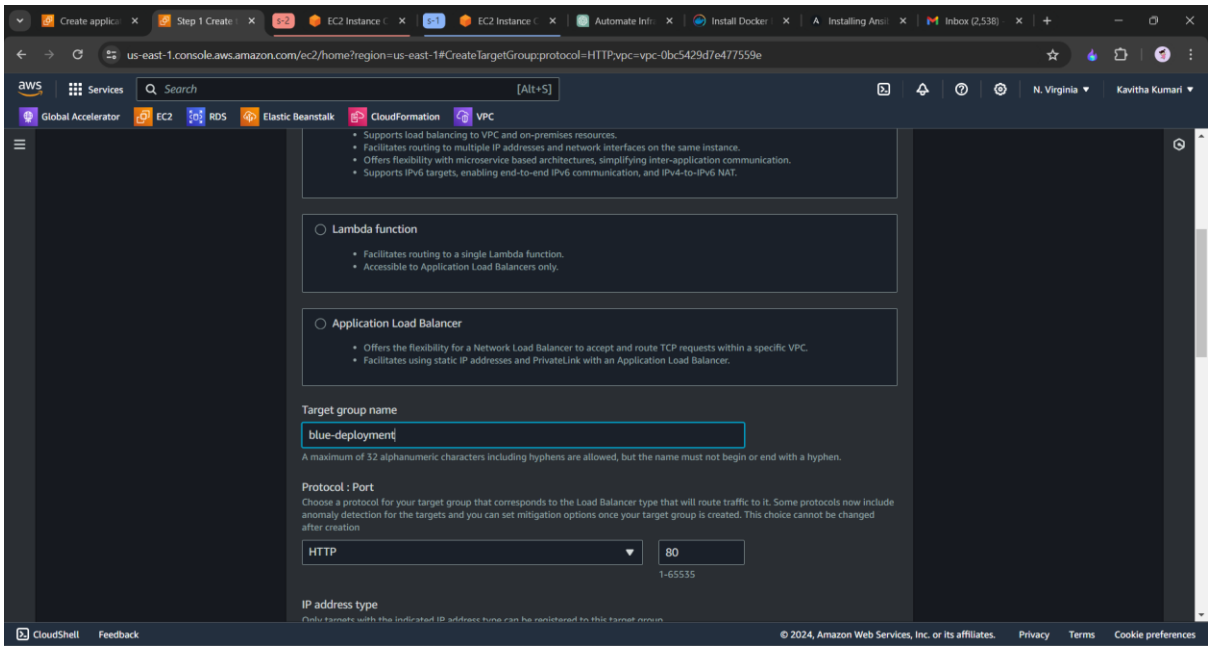
- Therefore, both the policies have been created.

Now when the cpu utilisation is more than 80 the instance will be added and when it is lower than 60 the instance will be removed until the minimum number of instances is reached.

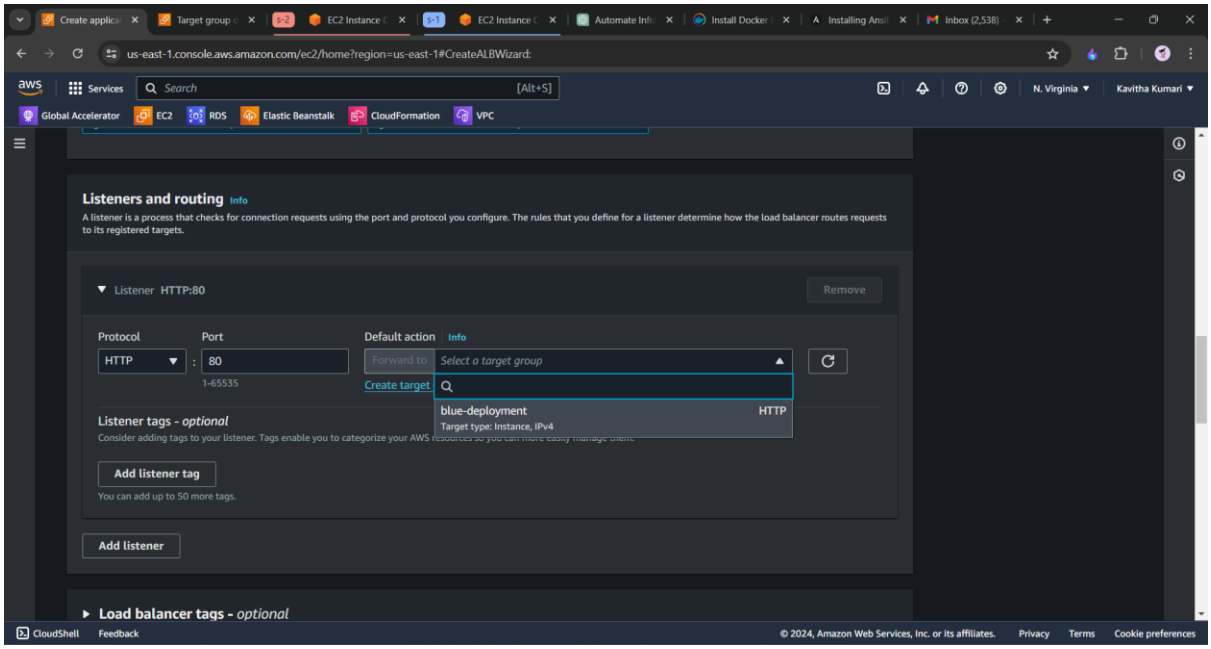
2. Create an Application Load balancer to distribute the load between compute resources







Similarly we will also create the green-deployment target groupt



- To create two target groups for blue and green deployments and configure weighted routing, you can follow these steps:

1. Create Target Groups:

- - Navigate to the AWS Management Console and open the Amazon EC2 service.
- - In the navigation pane, under "Load Balancing", choose "Target Groups".
- - Click on "Create target group".
- - Configure the target group for the blue deployment, specifying a name, protocol, port, and other settings as needed. Repeat this step to create another target group for the green deployment.

2. Create Load Balancer (if not already created):

- - If you do not have a load balancer already, you will need to create one. This can be an Application Load Balancer (ALB) or Network Load Balancer (NLB) depending on your requirements.
- - Configure the load balancer to listen on the desired ports and protocols.

3. Configure Listener Rules:

- Once the load balancer is created, navigate to its configuration and add listener rules.
- For each listener rule, specify conditions for routing traffic to the blue and green target groups. You'll likely want to use HTTP or HTTPS protocols and specify a path pattern or host header if needed.
- For weighted routing, configure the weights for each target group. For example, set 70% weight for the blue target group and 30% weight for the green target group.

4. Test and Verify:

- Once the listener rules are configured, test the setup by sending traffic to the load balancer and verify that it's distributed according to the weighted routing configuration.
- Monitor the target groups and load balancer to ensure that traffic is being routed correctly.

- By following these steps, you will be able to create two target groups for blue and green deployments and configure weighted routing to route traffic between them according to the specified percentages. Click on Route 53 and we must create the hosted zone and and we must give our DNS name and keep it as public hosted zone and click on next.
- After the hosted zone is created, we will now create the record. A record contains the public IP. When we click on the DNS name it should take us to the public IP which we have selected.

----------End----------