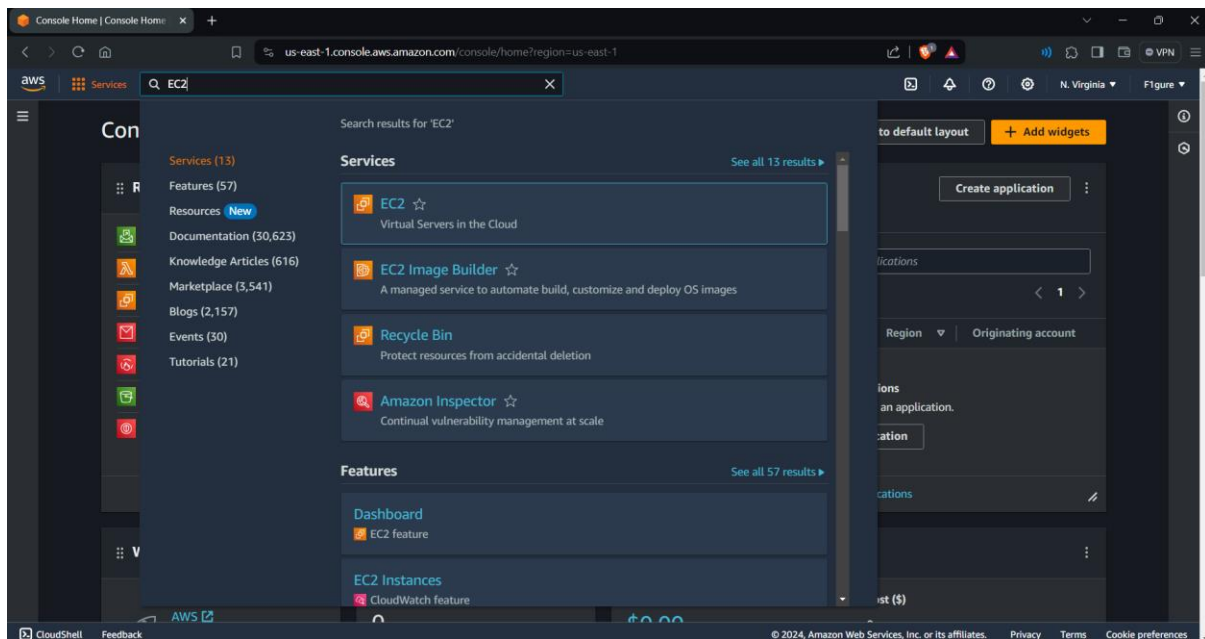


Assignment No: 11

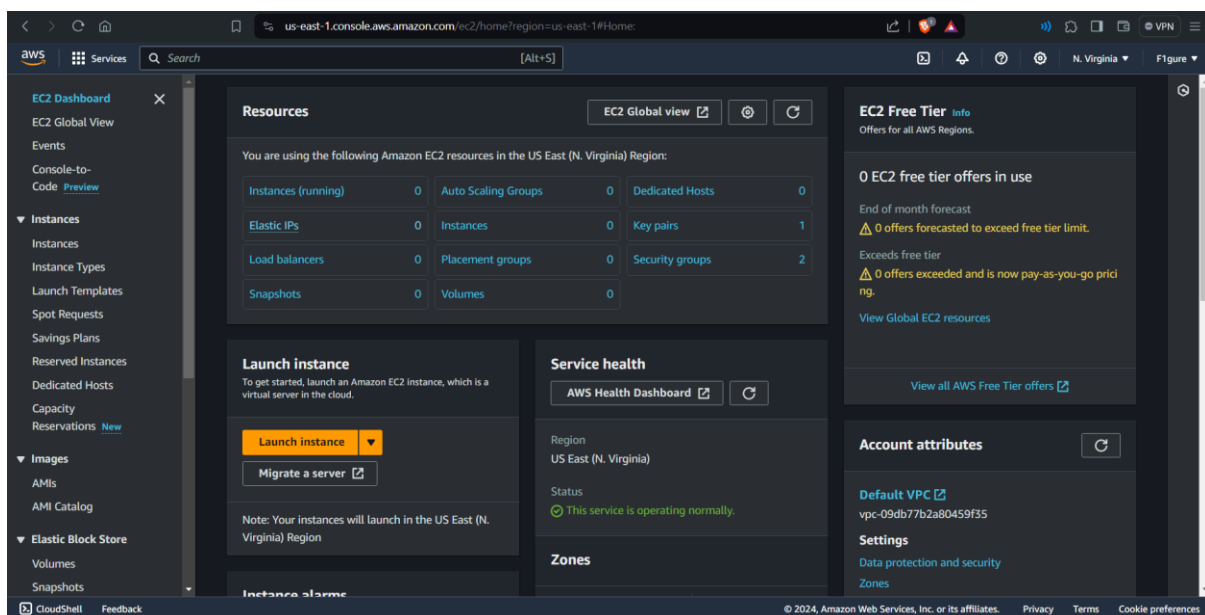
Problem Statement: Build scaling plans in AWS that balance load on different EC2 instances.

The steps are as follows: -

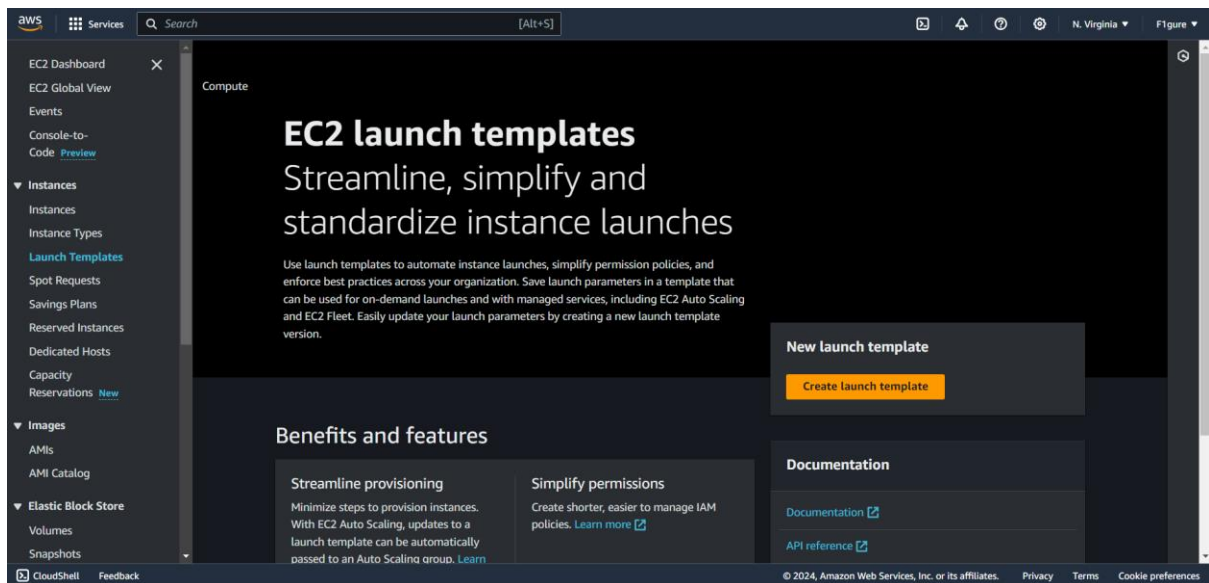
1. Please access AWS and look up EC2, then select the initial option displayed.



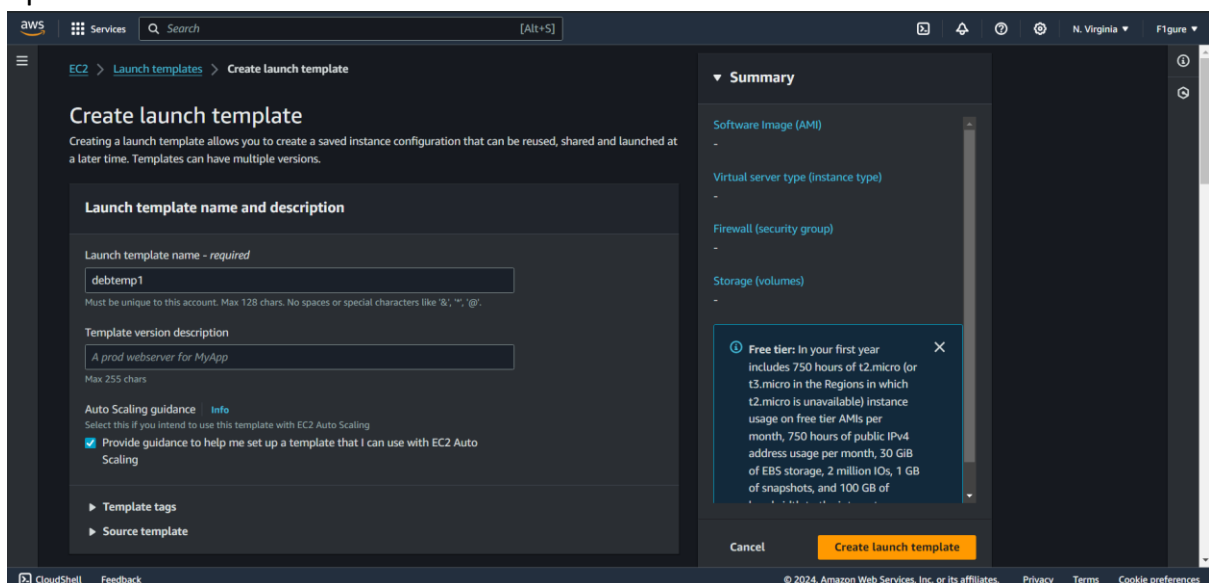
2. Next, select "Launch Templates" from the menu on the left side.



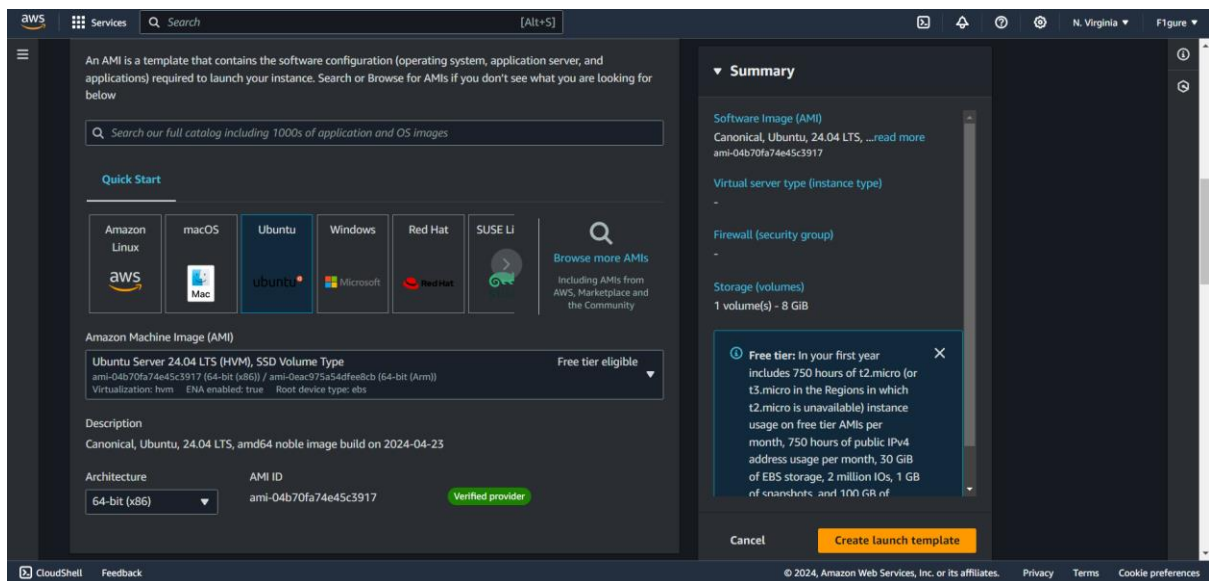
3. Now click on the "Create Template" option.



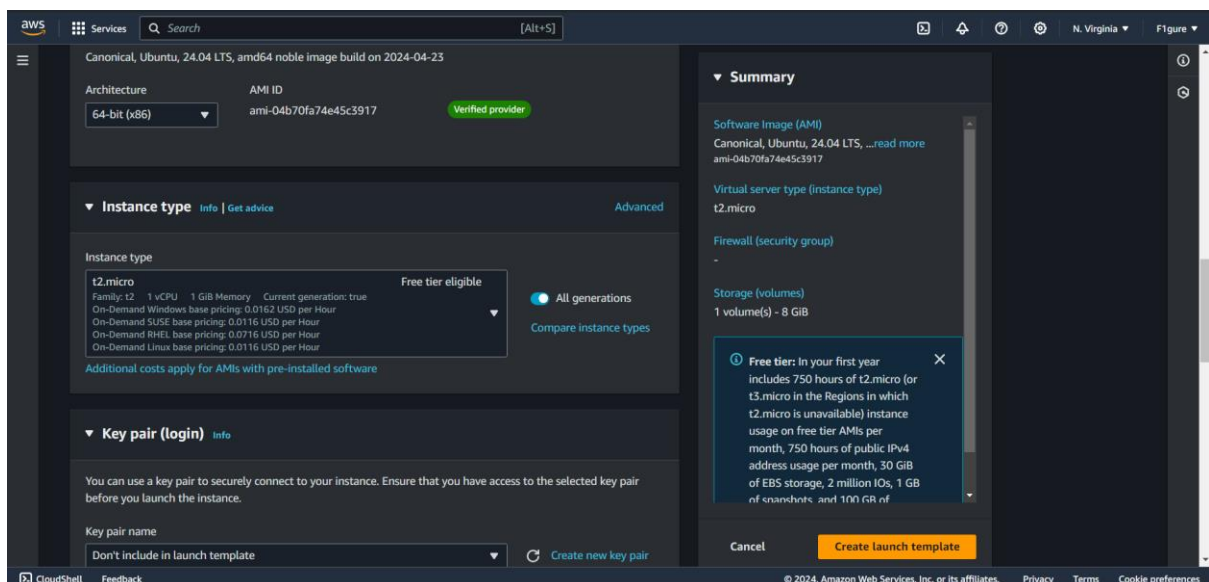
4. Enter a template name, such as "debtemp1," and check the box for autoscaling options.



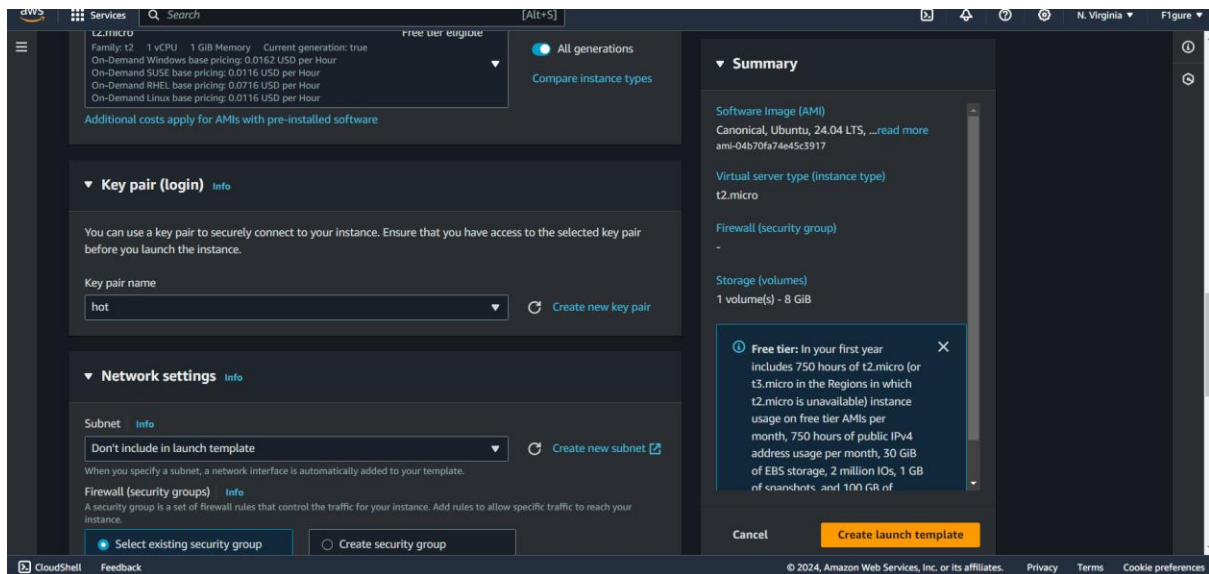
5. Navigate to "Quick Start" and choose "Ubuntu" from the list of available AMIs.



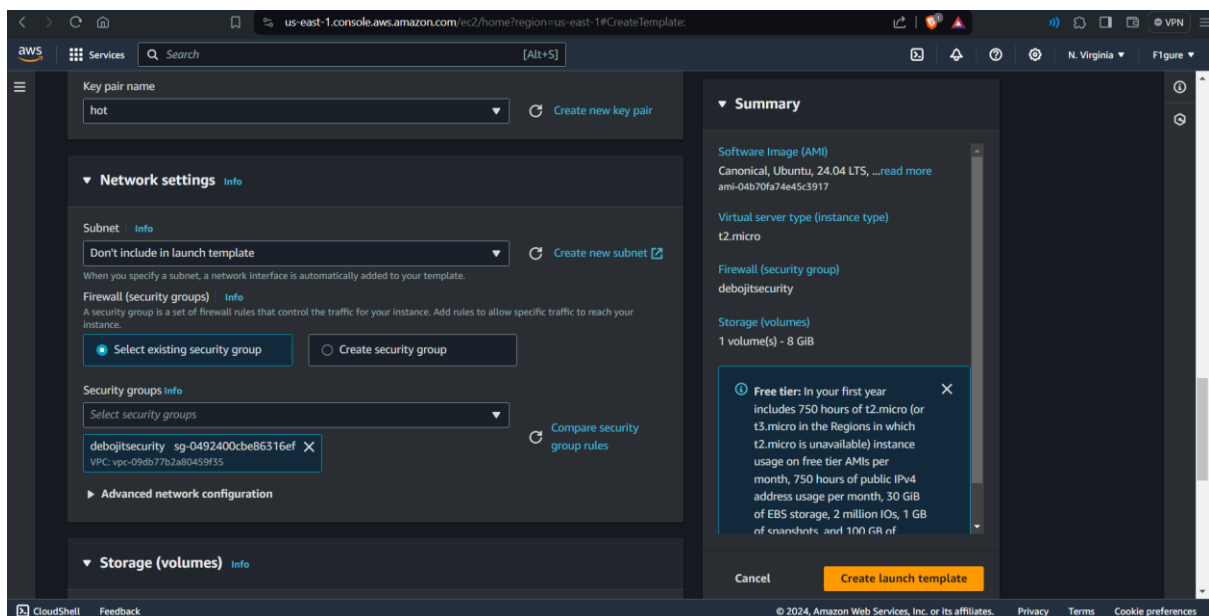
6. Next, select the instance type—either t2.micro or t3.micro, both of which are free tier eligible.



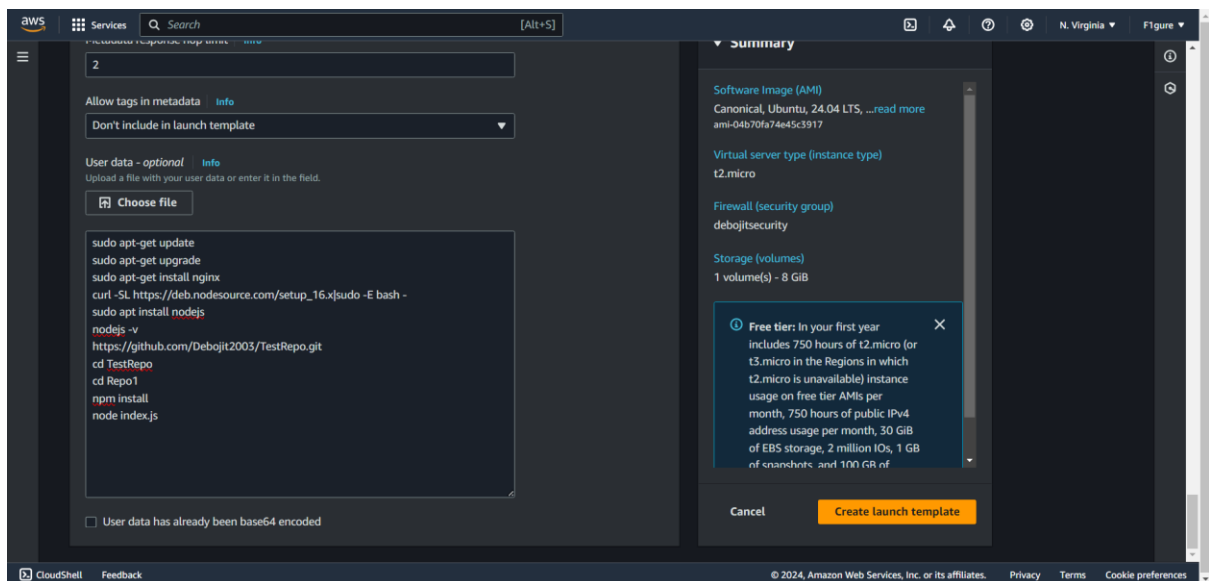
7. Choose either an existing key pair or create a new one if it doesn't exist. For example, use "hot" as the key pair name.



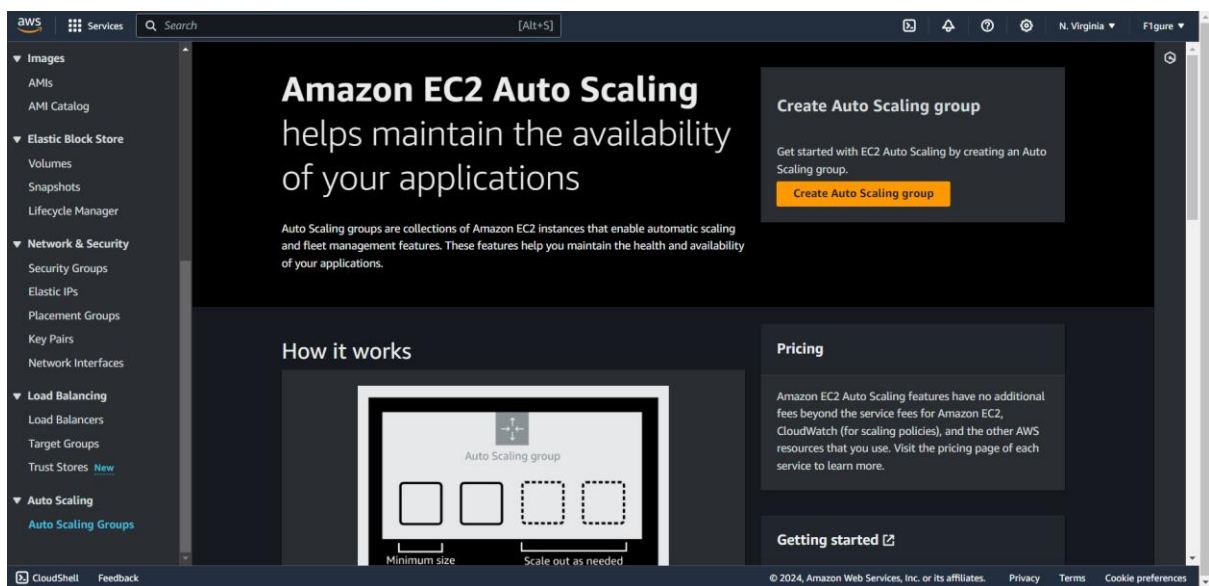
8. Select an existing security group, such as "**debojitsecurity**" which is already in place.



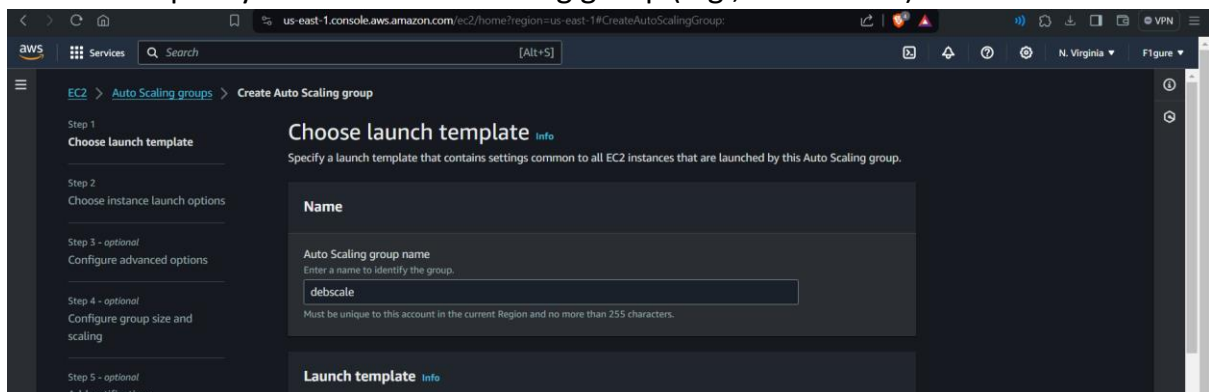
9. Expand the "**Advanced details**" section, navigate to "**User data**", and input the provided code. Then proceed to click on "**Create launch template**".



10. Once the launch template has been successfully created, navigate to the left pane and search for "**Auto Scaling Groups**". Then, select "**Create Auto Scaling Group**".



11. Please specify a name for the scaling group (e.g., "debscale").



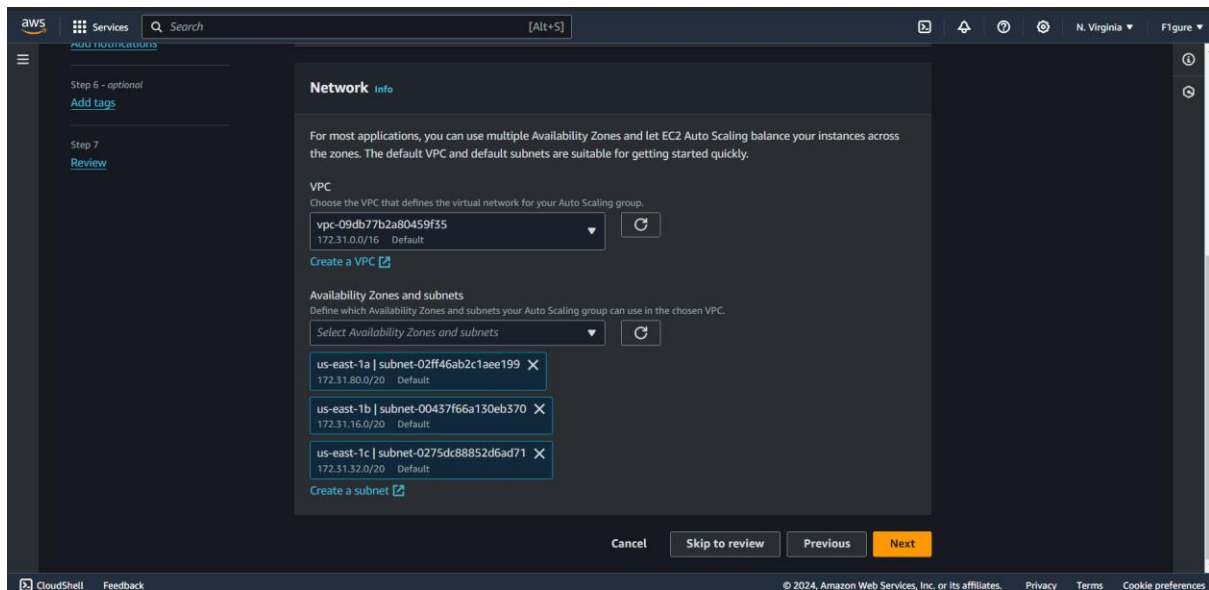
12. Select the template that was created in the preceding steps.

The screenshot shows the AWS Management Console interface for creating an Auto Scaling group. The left sidebar lists the steps: Step 2 (Choose instance launch options), Step 3 (optional, Configure advanced options), Step 4 (optional, Configure group size and scaling), Step 5 (optional, Add notifications), Step 6 (optional, Add tags), and Step 7 (Review). The main content area is titled 'Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.' It includes a 'Name' section with a text input for 'Auto Scaling group name' containing 'debscale'. Below this is the 'Launch template' section, which features a dropdown menu set to 'debtemp1' and a 'Version' dropdown set to 'Default (1)'. A blue informational box states: 'For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.' At the bottom of the console, there are links for 'CloudShell' and 'Feedback', and a footer with copyright information and links for 'Privacy', 'Terms', and 'Cookie preferences'.

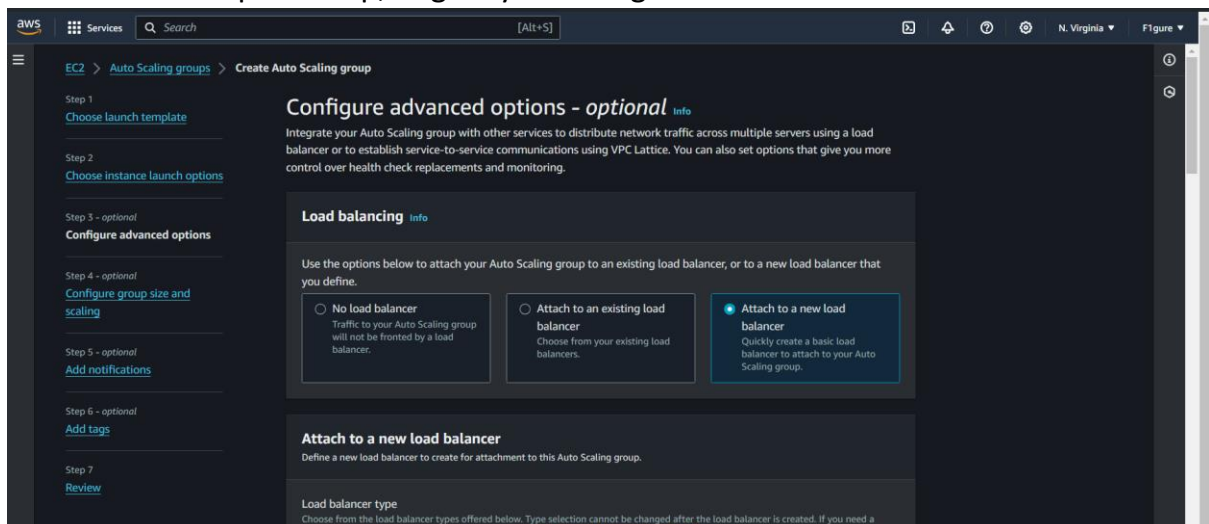
13. Proceed to click on “Next”.

The screenshot shows the 'Review' step of the AWS Management Console 'Create Auto Scaling group' wizard. The left sidebar now highlights 'Review'. The main content area displays a summary of the configuration. It includes the 'Launch template' dropdown set to 'debtemp1' and the 'Version' dropdown set to 'Default (1)'. Below these are links to 'Create a launch template' and 'Create a launch template version'. A table-like layout shows the following details: 'Description' (empty), 'AMI ID' (ami-04b70fa74e45c3917), 'Key pair name' (hot), 'Launch template' (debtemp1), 'Instance type' (t2.micro), 'Security groups' (empty), 'Request Spot Instances' (No), and 'Security group IDs' (sg-0492400cbe86316ef). An 'Additional details' section shows 'Storage (volumes)' (empty) and 'Date created' (Fri May 03 2024 17:16:29 GMT+0530 (India Standard Time)). At the bottom right, there are 'Cancel' and 'Next' buttons. The footer and sidebar elements are consistent with the previous screenshot.

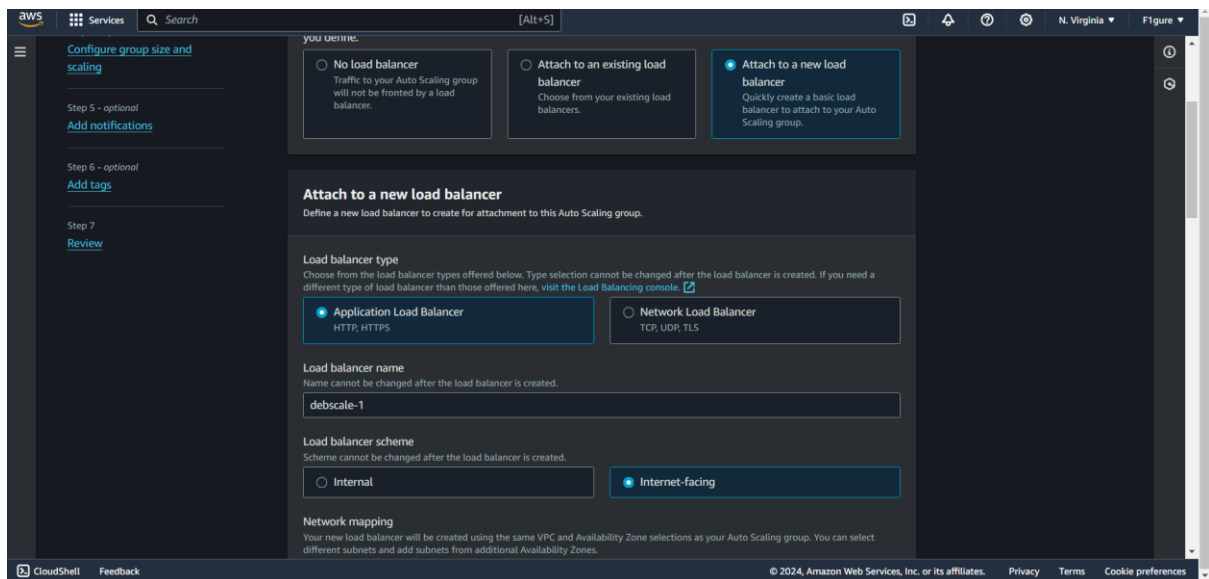
14. In the following step, choose all available availability zones and subnets, then proceed by clicking "Next".



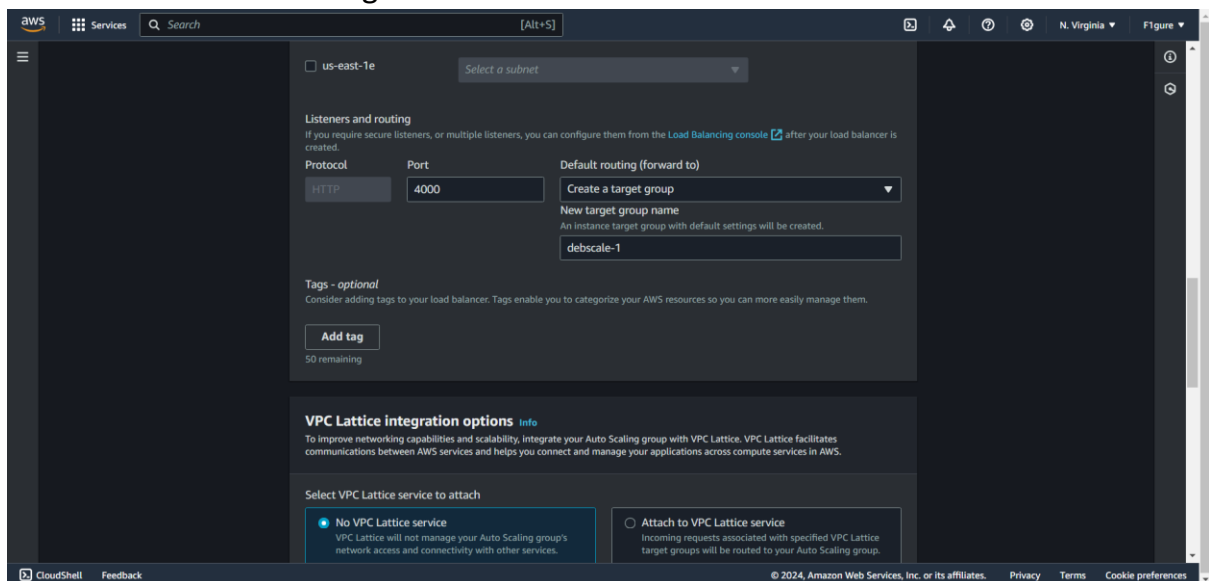
15. In the subsequent step, begin by selecting **"Attach to a new load balancer"**.



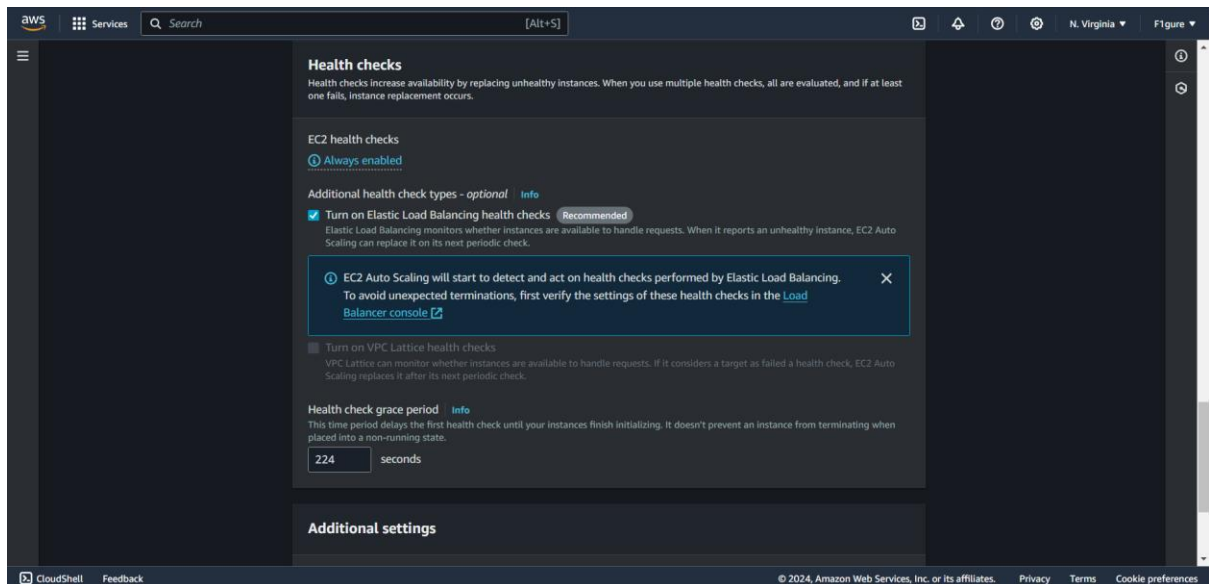
16. Select **"Application Load Balancer"** as the load balancer type and **"Internet-facing"** as the load balancer scheme.



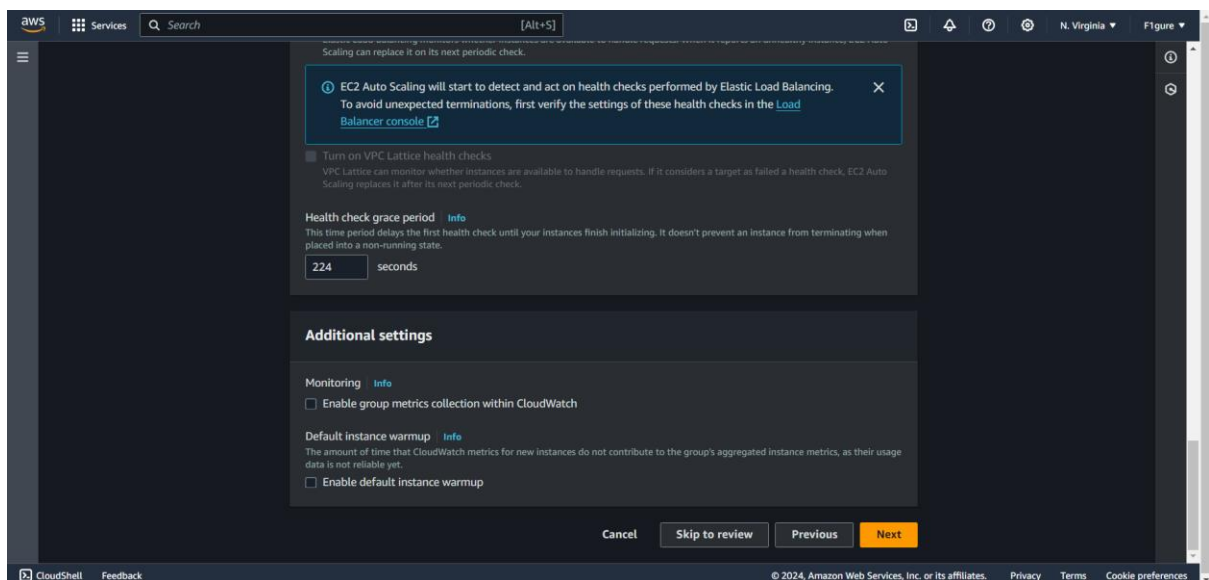
17. Modify the HTTP port number from 80 to 4000 and designate the scaling group created for default routing.



18. Enable the checkbox to activate health checks and specify a “health check grace period”, set here to 224 seconds.



19. Without any further modifications, proceed to click on **“Next”**.



20. In this step, specify the desired, minimum, and maximum capacities.

Define your group's desired capacity and scaling limits. You can optionally add automatic scaling to adjust the size of your group.

Group size Info
Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

Desired capacity type
Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.
Units (number of instances) ▼

Desired capacity
Specify your group size.
2

Scaling Info
You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity
1
Equal or less than desired capacity

Max desired capacity
3
Equal or greater than desired capacity

21. Next, opt for the **"Target Tracking Scaling Policy"** and configure the **CPU utilization target value** to 50. Additionally, set the **instance warm-up time** to 240 seconds.

Automatic scaling - optional Info
Choose whether to use a target tracking policy. You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

☐ No scaling policies
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

☒ **Target tracking scaling policy**
Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Scaling policy name
Target Tracking Policy

Metric type Info
Monitored metric that determines if resource utilization is too low or high. If using EC2 metrics, consider enabling detailed monitoring for better scaling performance.
Average CPU utilization ▼

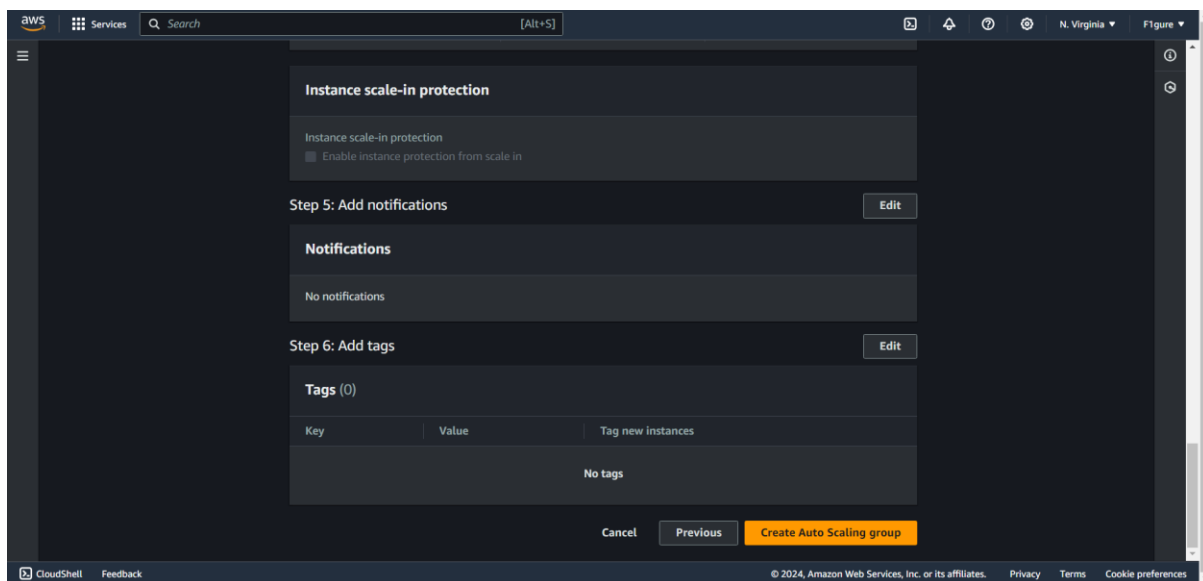
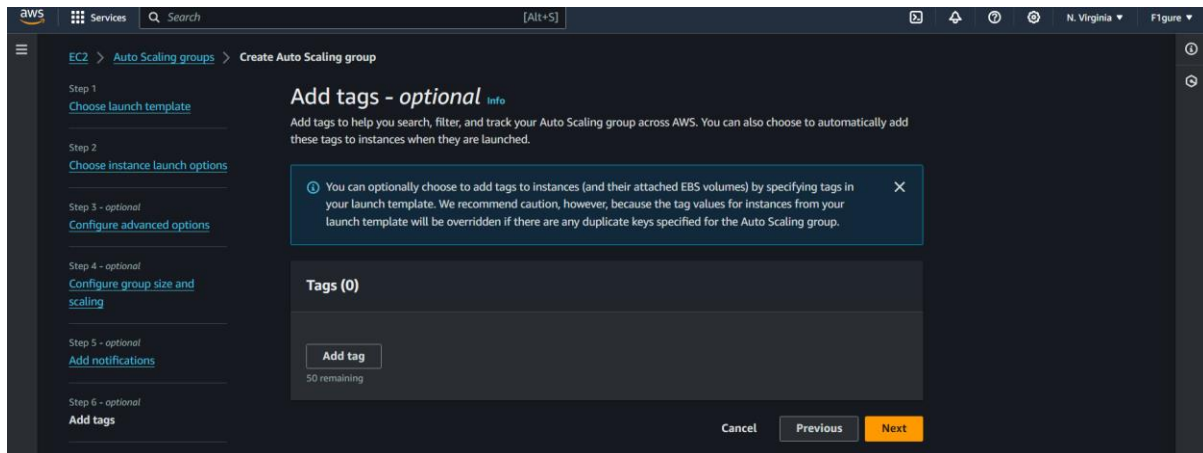
Target value
50

Instance warmup Info
240 seconds

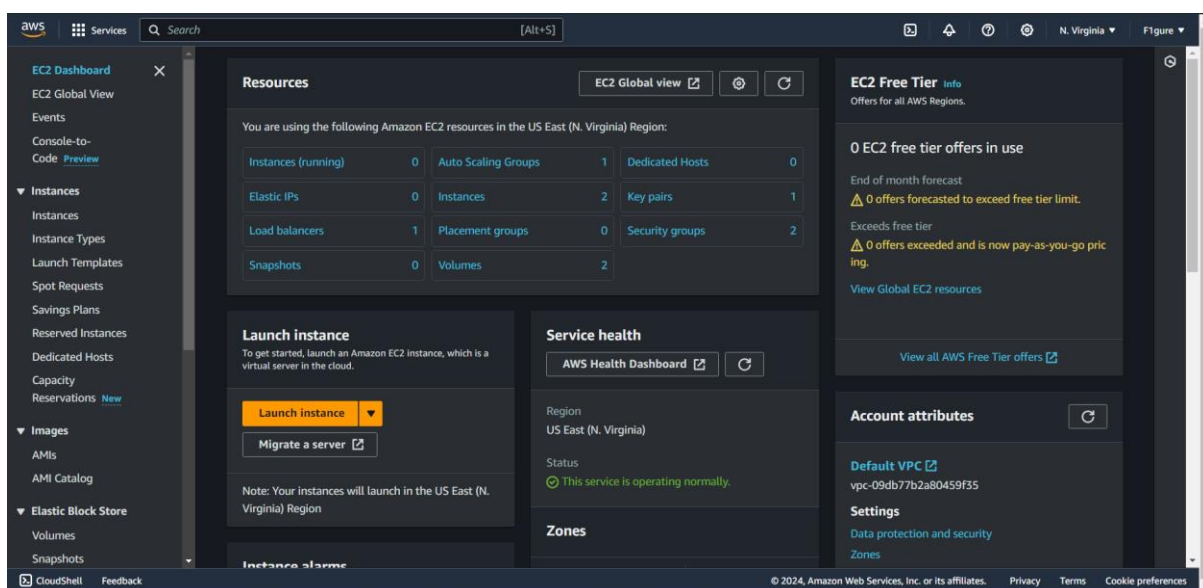
☐ Disable scale in to create only a scale-out policy

22. Proceed by clicking **"Next"** without making any changes, and finally, select **"Create auto Scaling Group"**.

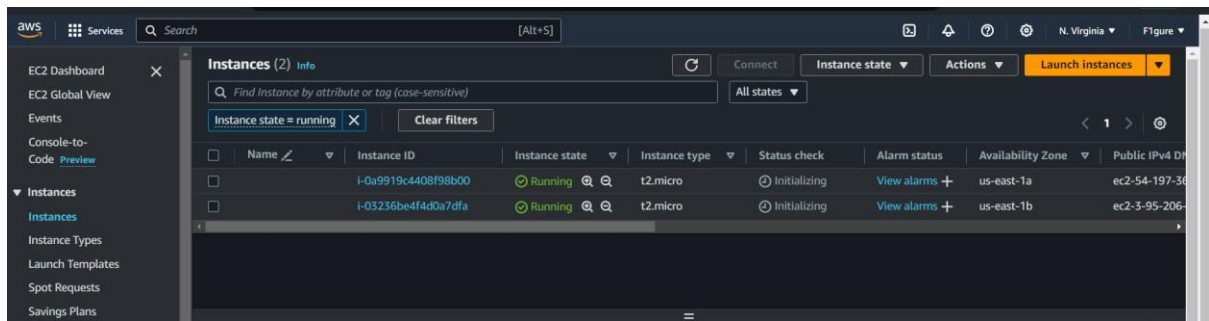
Add notifications - optional Info
Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.



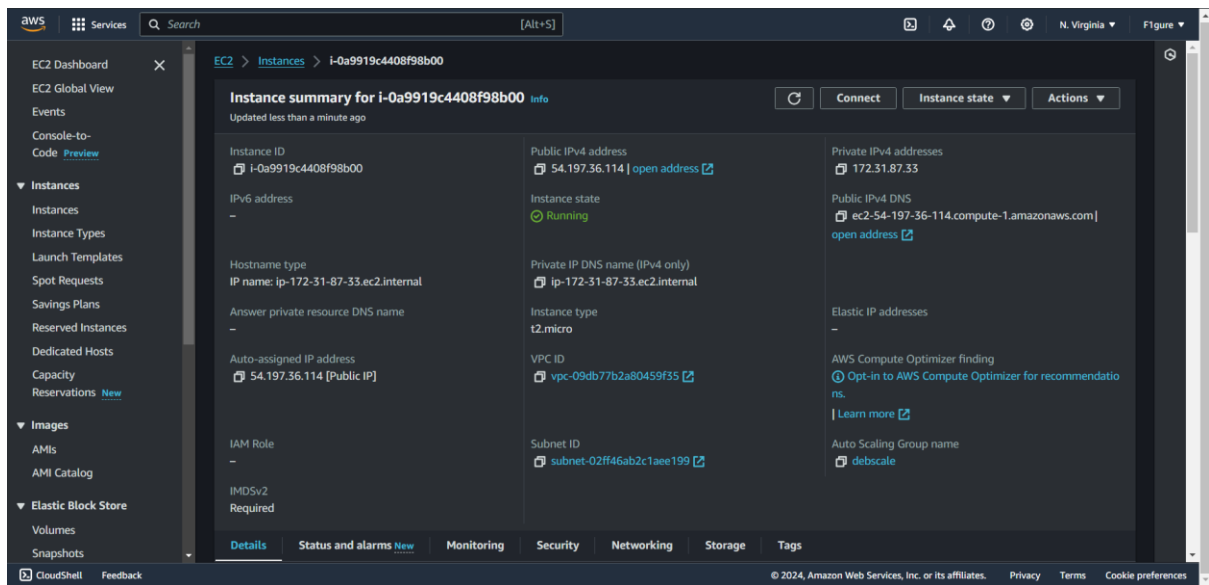
23. After the auto-scaling group is created, return to the EC2 dashboard and navigate to the "Instances" section for running instances.



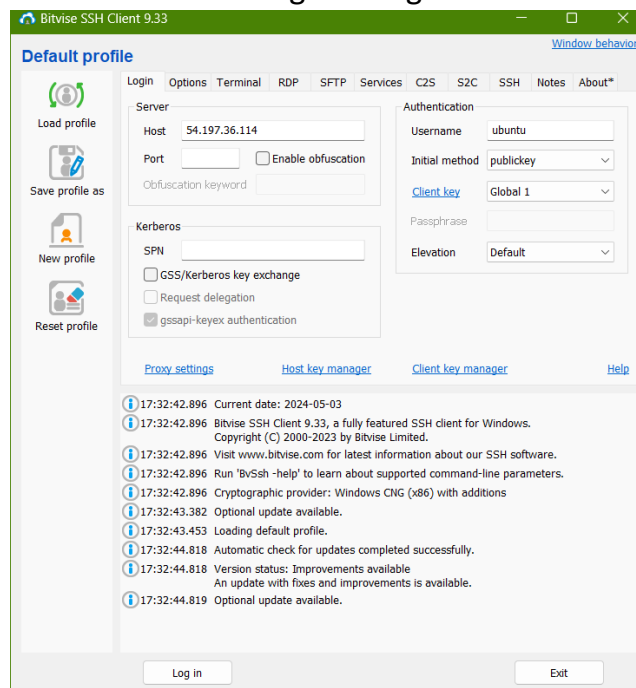
24. Since the minimum capacity chosen was 2, two instances have been created.



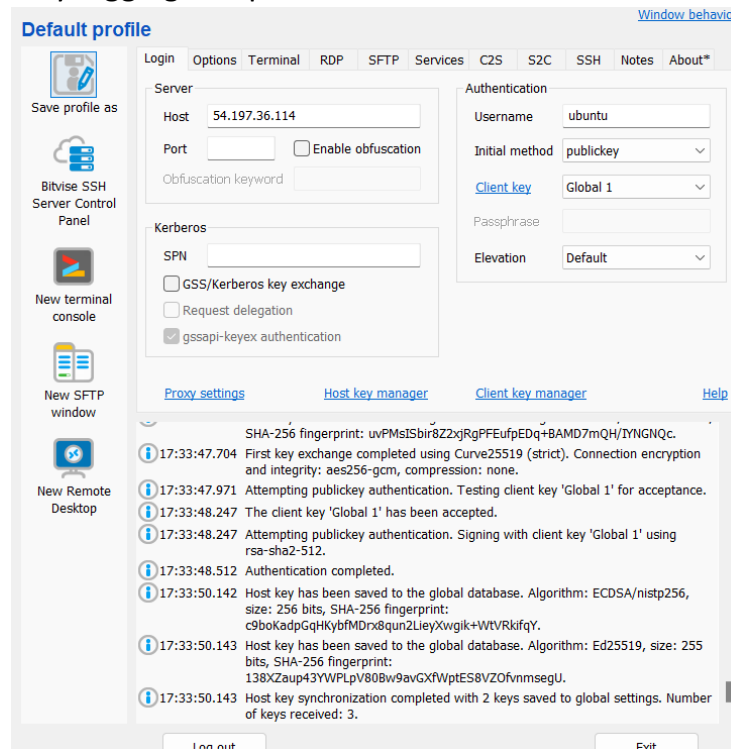
25. Choose any one of the instance IDs and copy the public IPv4 address.



26. Launch the Bitwise SSH Client and log in using the IPv4 address you copied earlier.



27. After successfully logging in, open a new terminal console from the left pane.



28. Now write the commands in the terminal as follows: -

→ ***sudo nano infi.sh*** (creates a .sh file)

```
ubuntu@ip-172-31-87-33:~$ sudo nano infi.sh
```

→ Write this code in the file "infi.sh" to run an infinite loop.

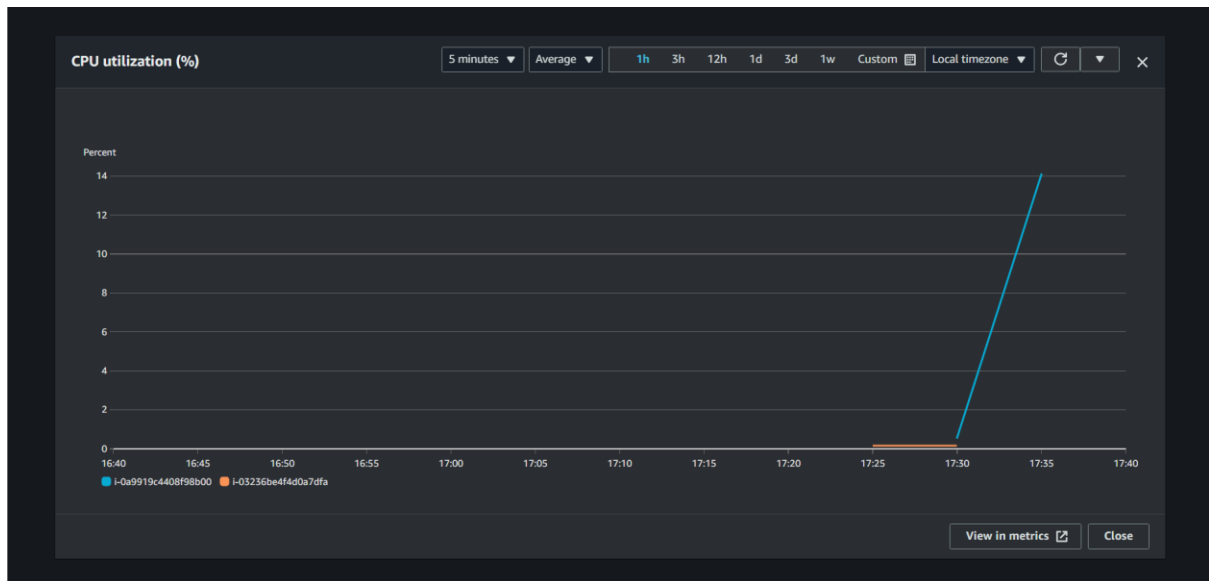
```
GNU nano 7.2                               infi.sh *
#!/bin/bash
while(true)
do
    echo "Inside loop"
done
```

→ Press Ctrl+X and write 'y' to save the file.

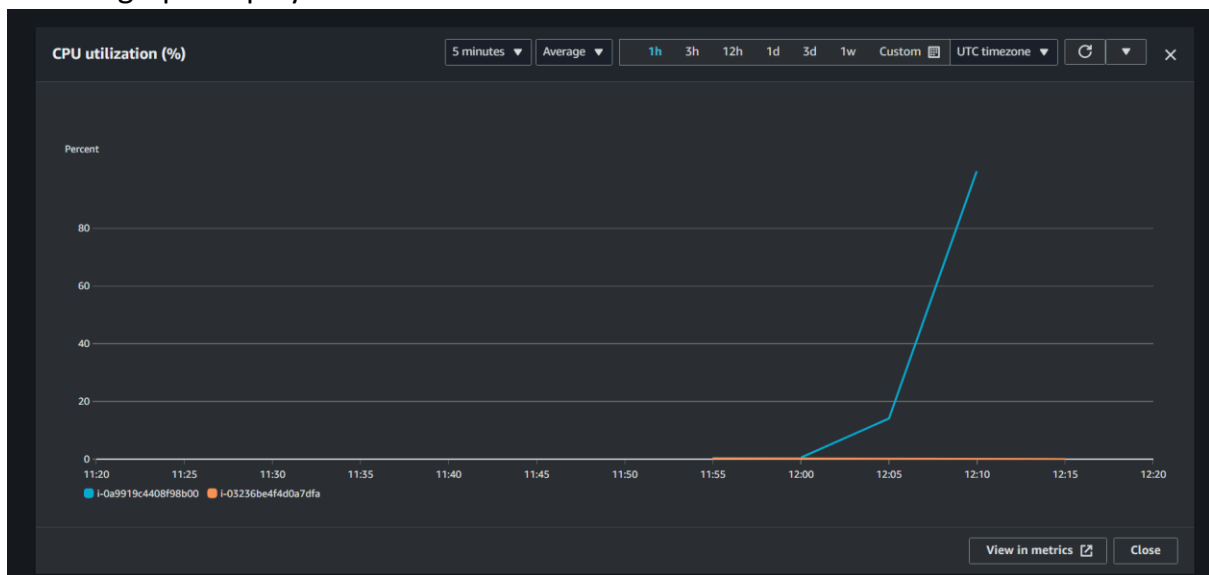
→ ***sudo chmod 777 infi.sh*** (to provide all permission to the file)

```
ubuntu@ip-172-31-87-33:~$ sudo chmod 777 infi.sh
```

→ ***sh infi.sh*** (Run the .sh file)



31. The graph displays CPU utilization for both instances.



When the CPU utilization exceeds the limit for both instances, another instance is created, as we have set the maximum capacity to 3.