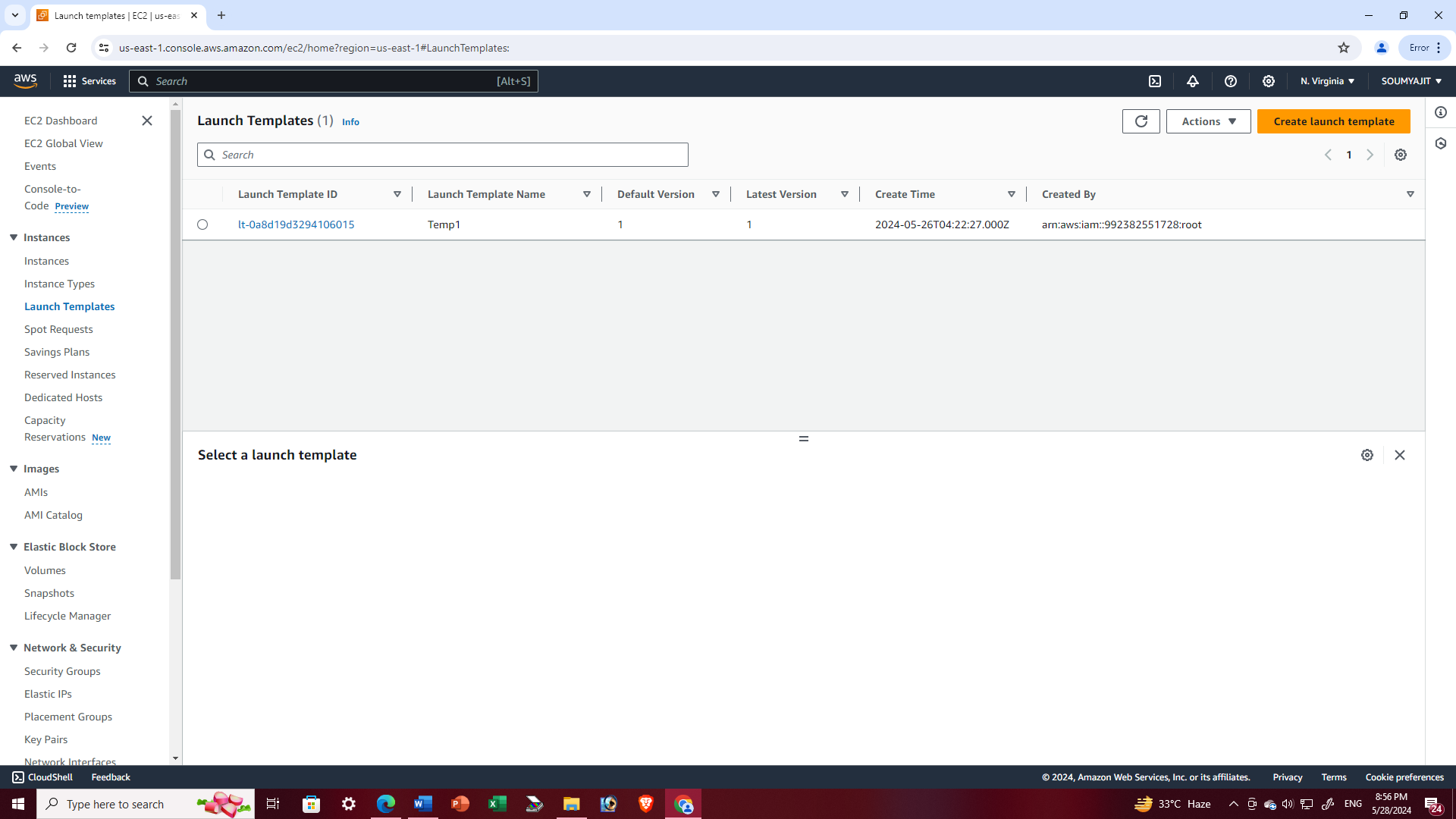
**Assignment 11**

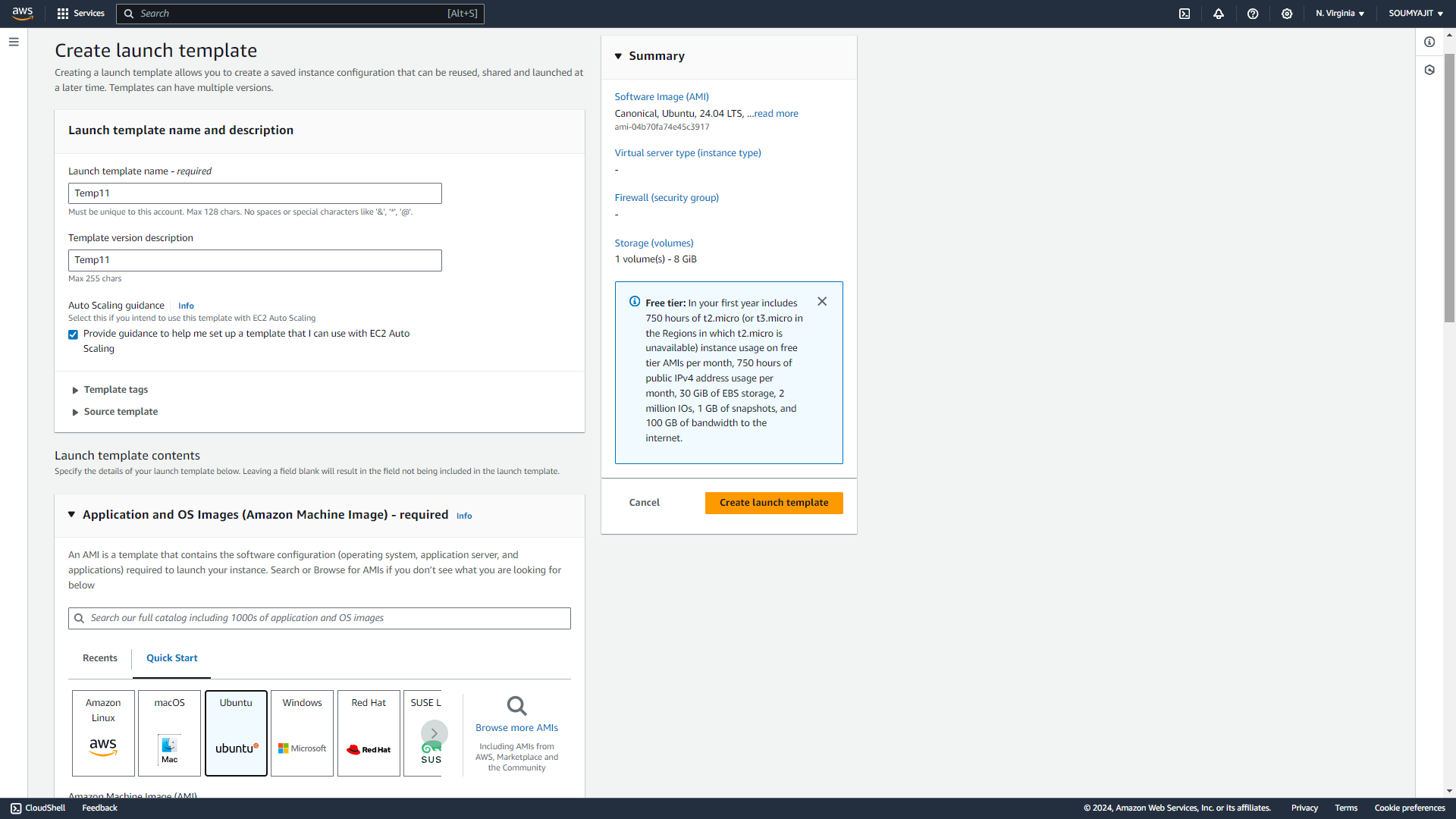
**Problem Statement:** Build scaling plans in AWS that balance load on different EC2 instance

**Steps:**

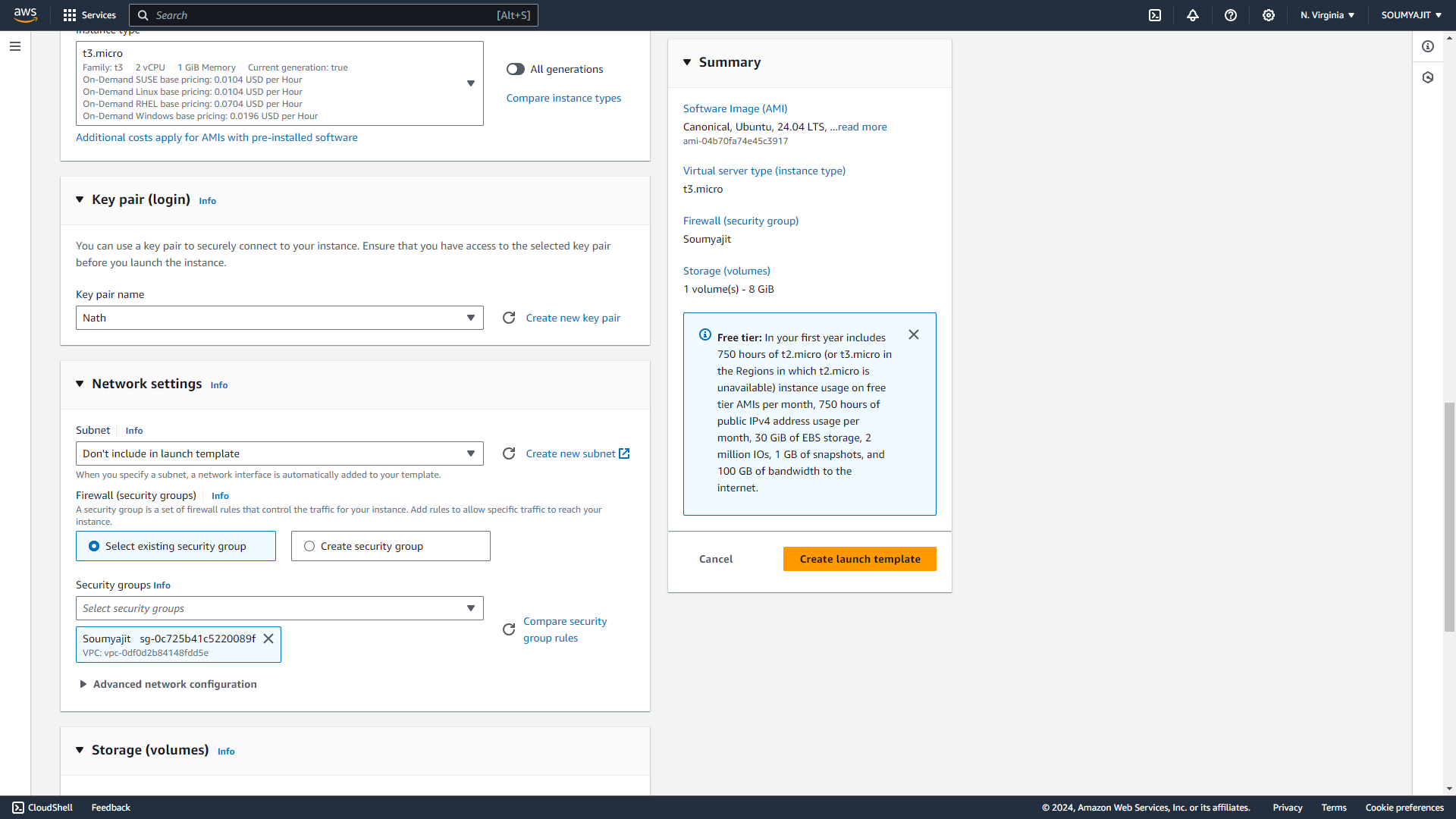
1. Open AWS and click EC2, next, select "Launch Template" from the menu on the left side. Now click on the "Create Launch Template" option.



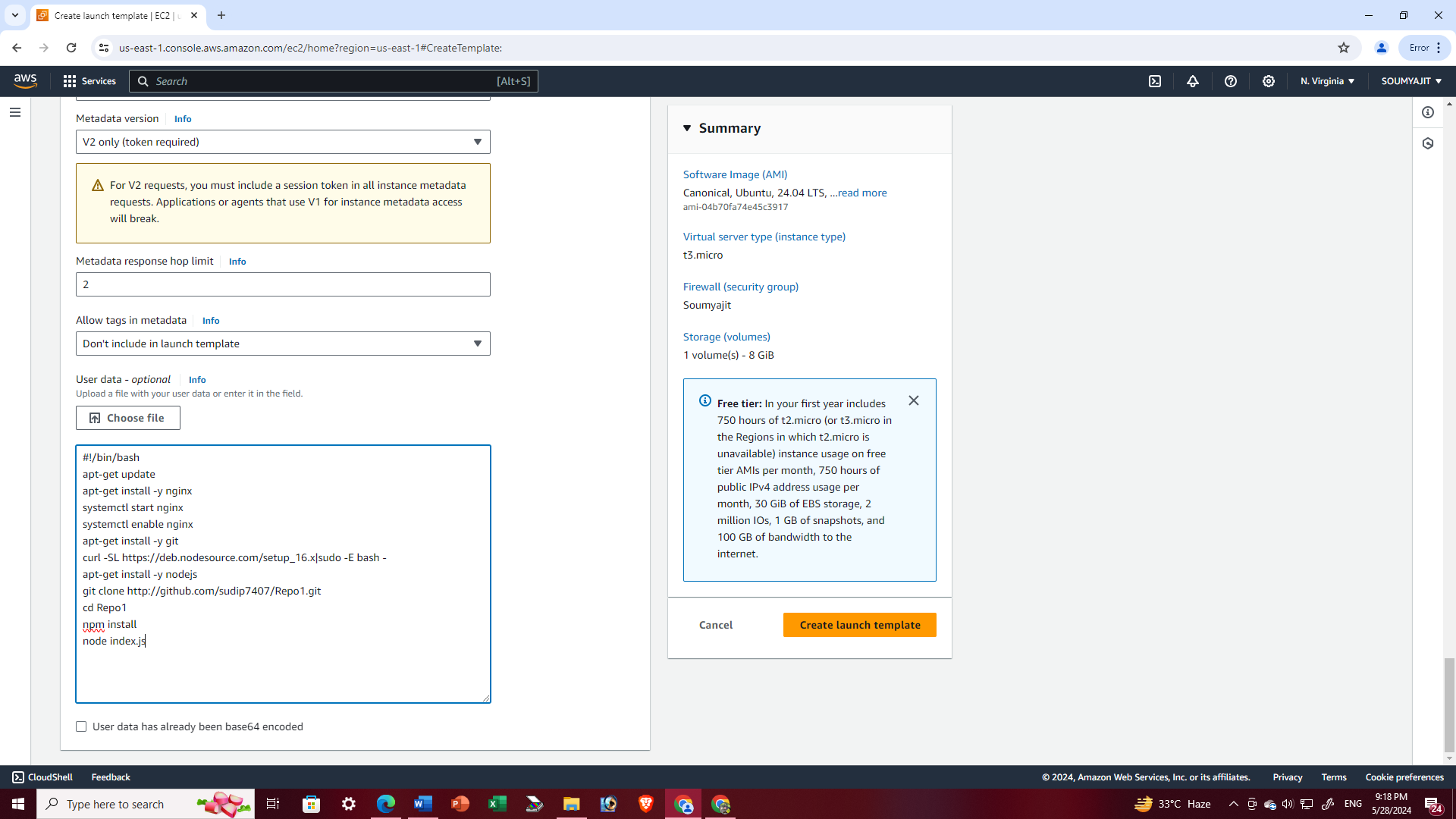
1. Enter a template name, such as "Temp11" and check the box for autoscaling option Navigate to "Quick Start" and choose "Ubuntu" from the list of available AMIs.



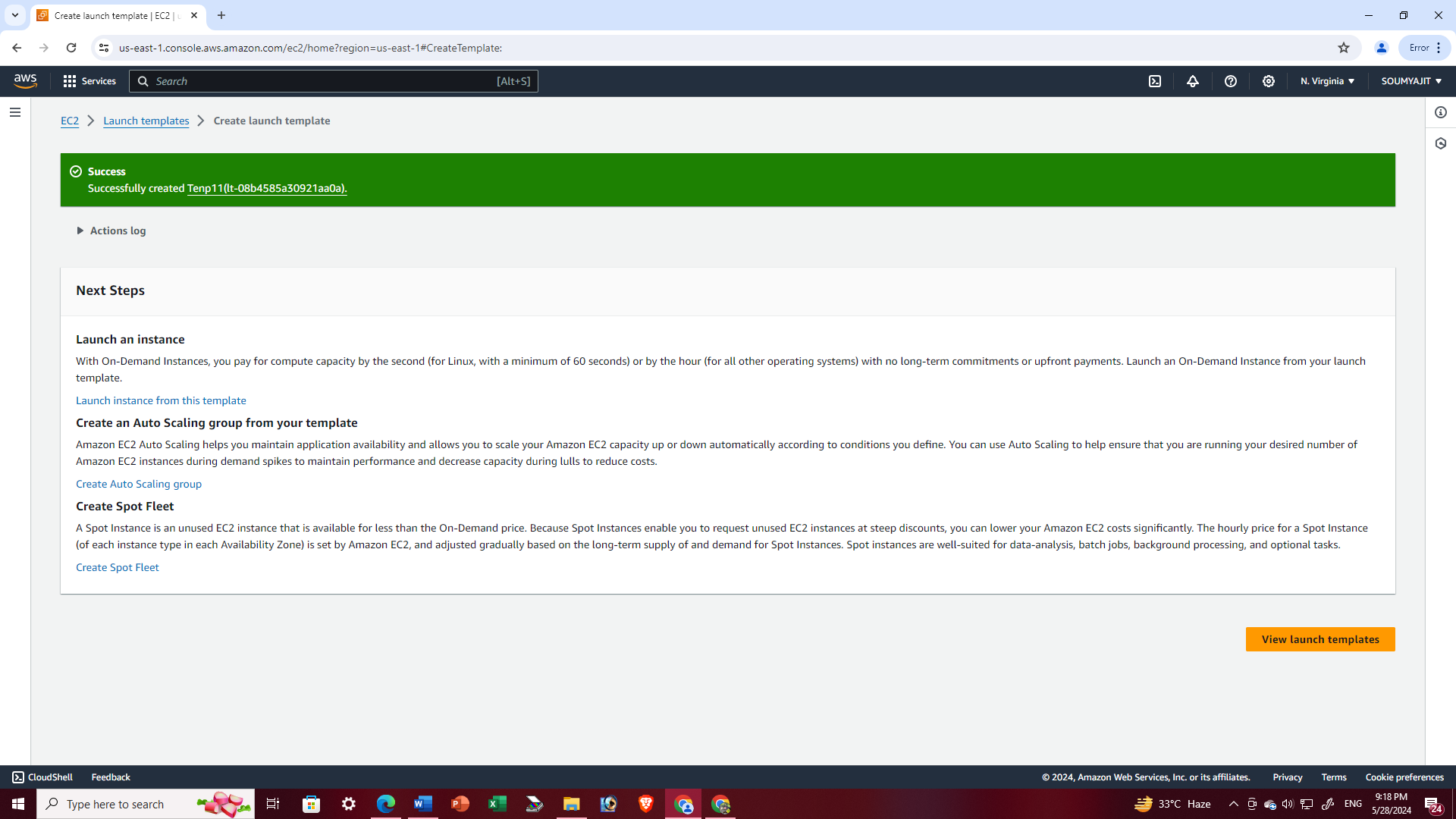
1. Next, select the instance type - either t2.micro or t3.micro, both of which are free tier eligible. Then choose either an existing key pair or create a new one if it doesn't exist. For example, use "nath" as the key pair name. Select an existing security group, such as "Soumyajit" which is already in place.



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



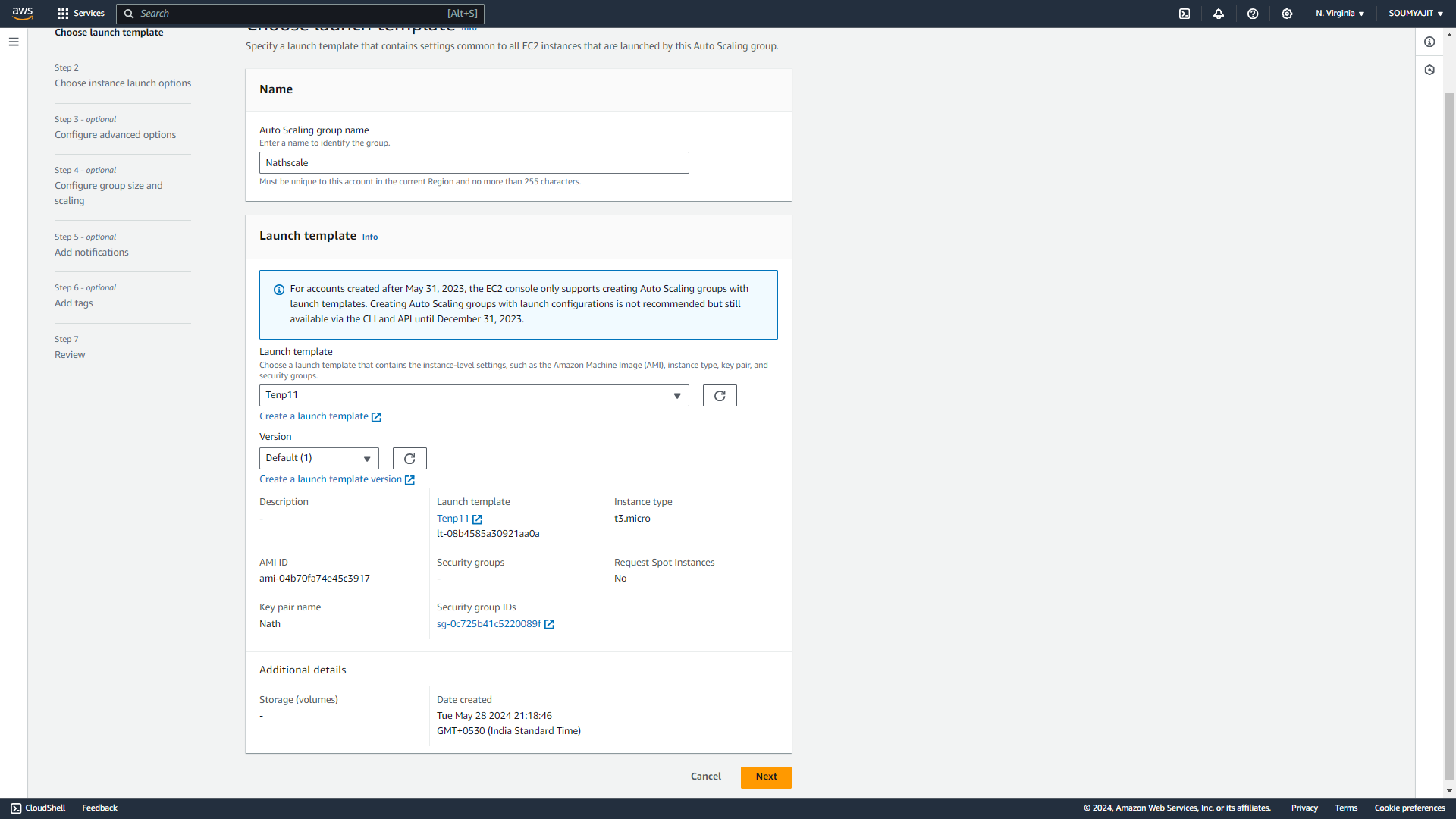
1. Template is successfully created



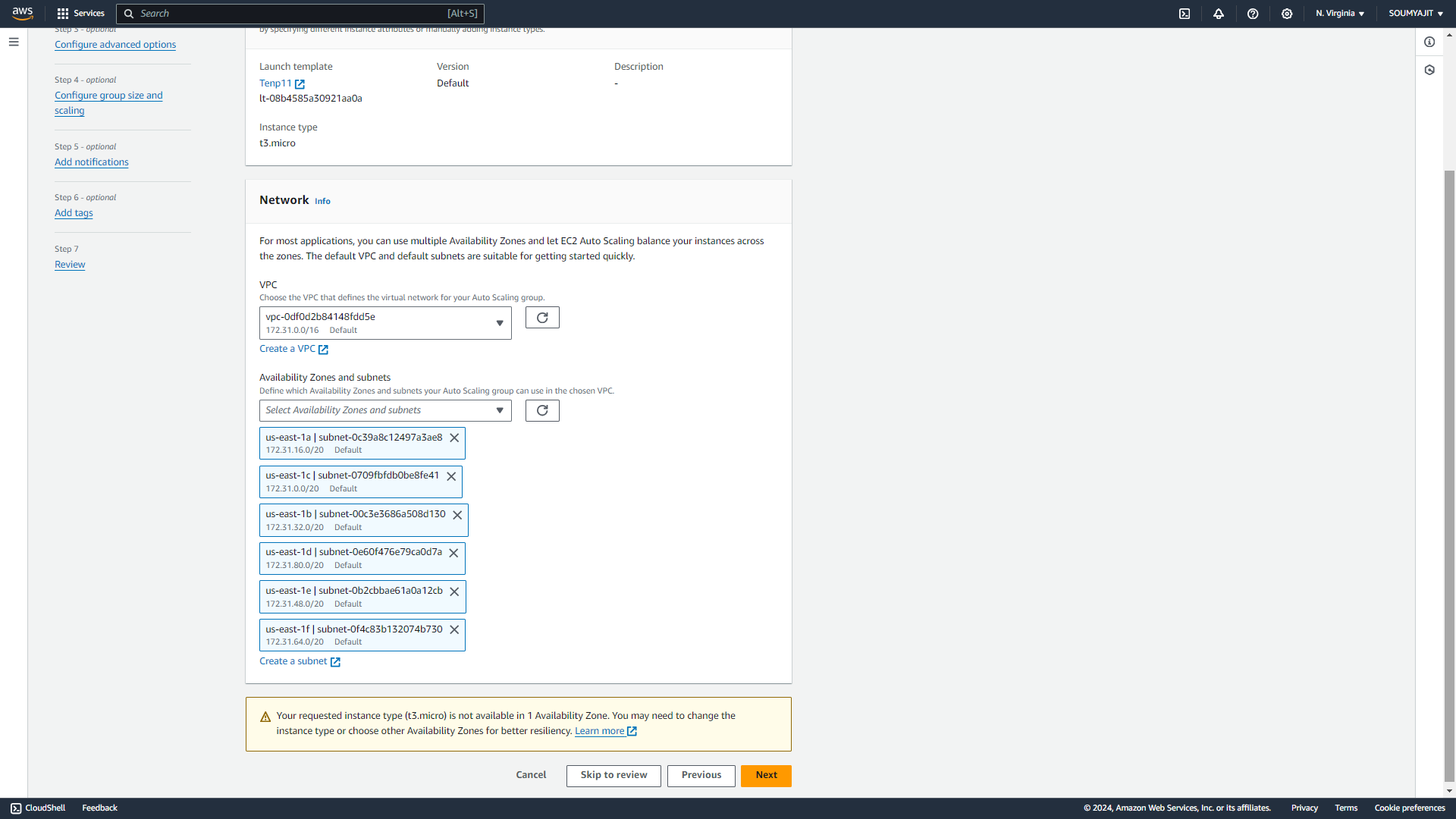
1. Once the launch template has been successfully created, navigate to the left pane and search for "Auto Scaling Groups". Then, select "Create Auto Scaling Group"



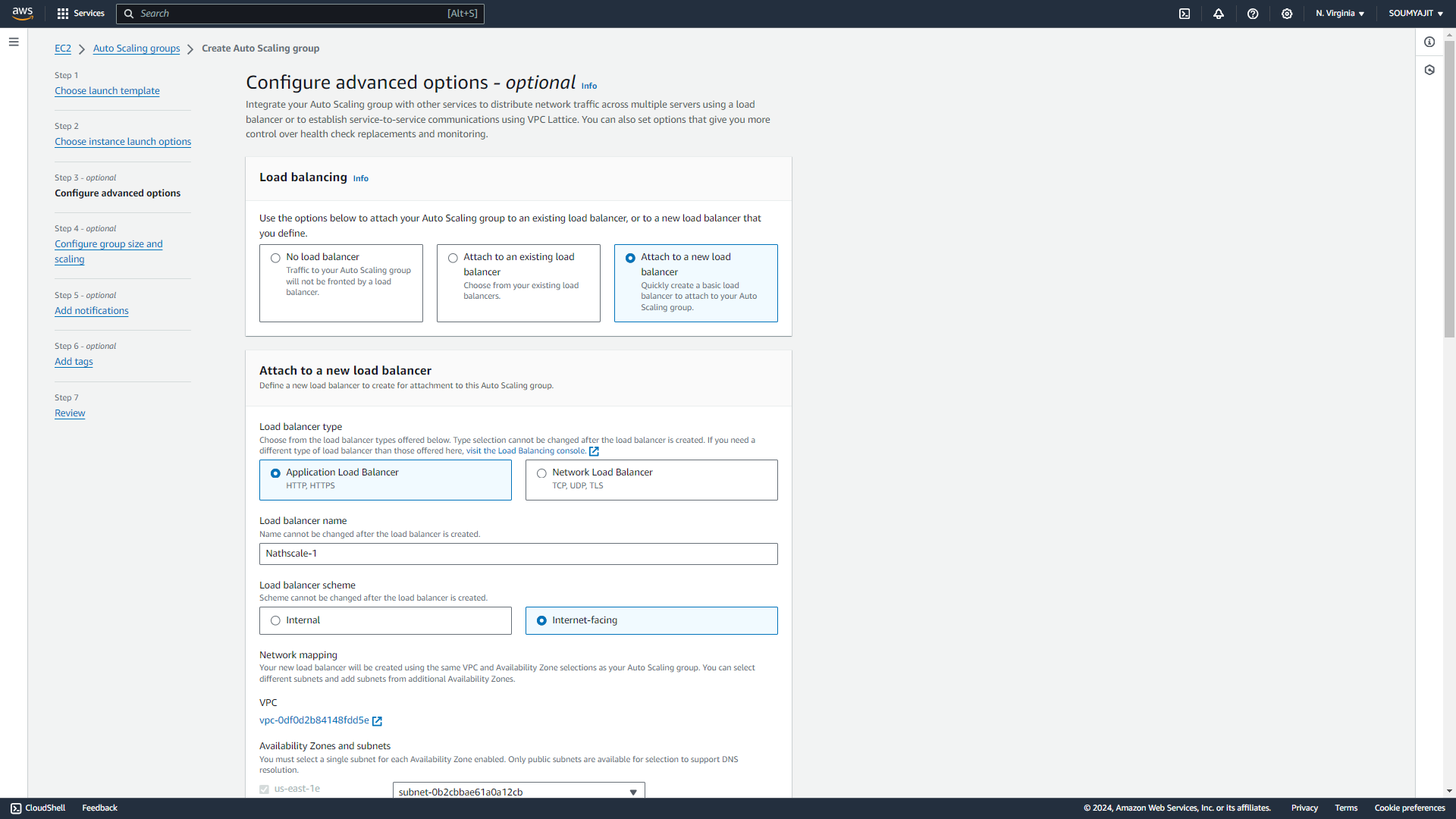
1. Please specify a name for the scaling group (e.g., "nathscale". Select the template that was created in the preceding steps (e.g. Temp1). Proceed to click on “Next”



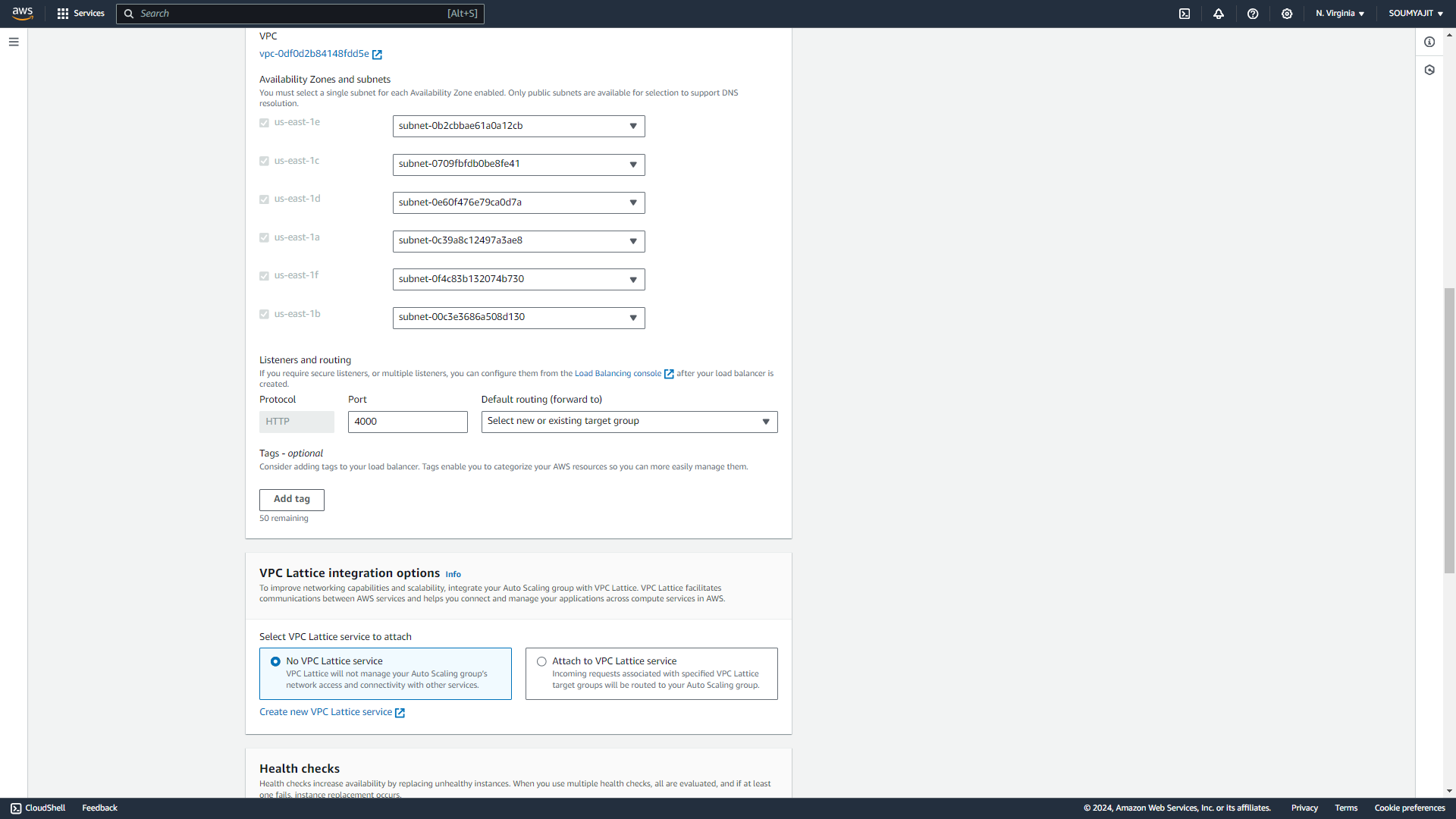
1. In the following step, choose all available availability zones and subnets, then proceed by clicking "Next"



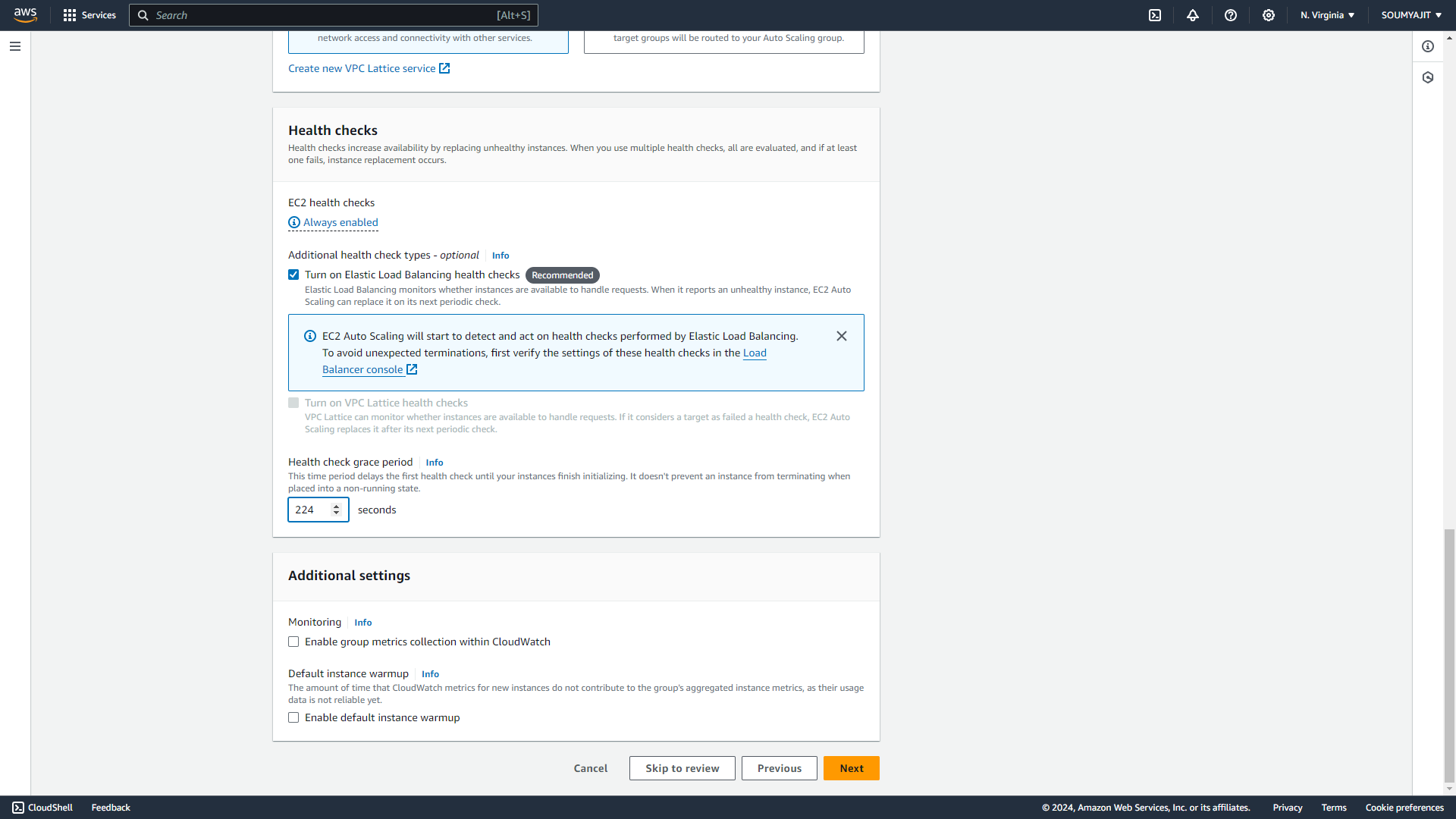
1. In the subsequent step, begin by selecting "Attach to a new load balancer. Select "Application Load Balancer" as the load balancer type and "Internet-facing" as the load balancer scheme



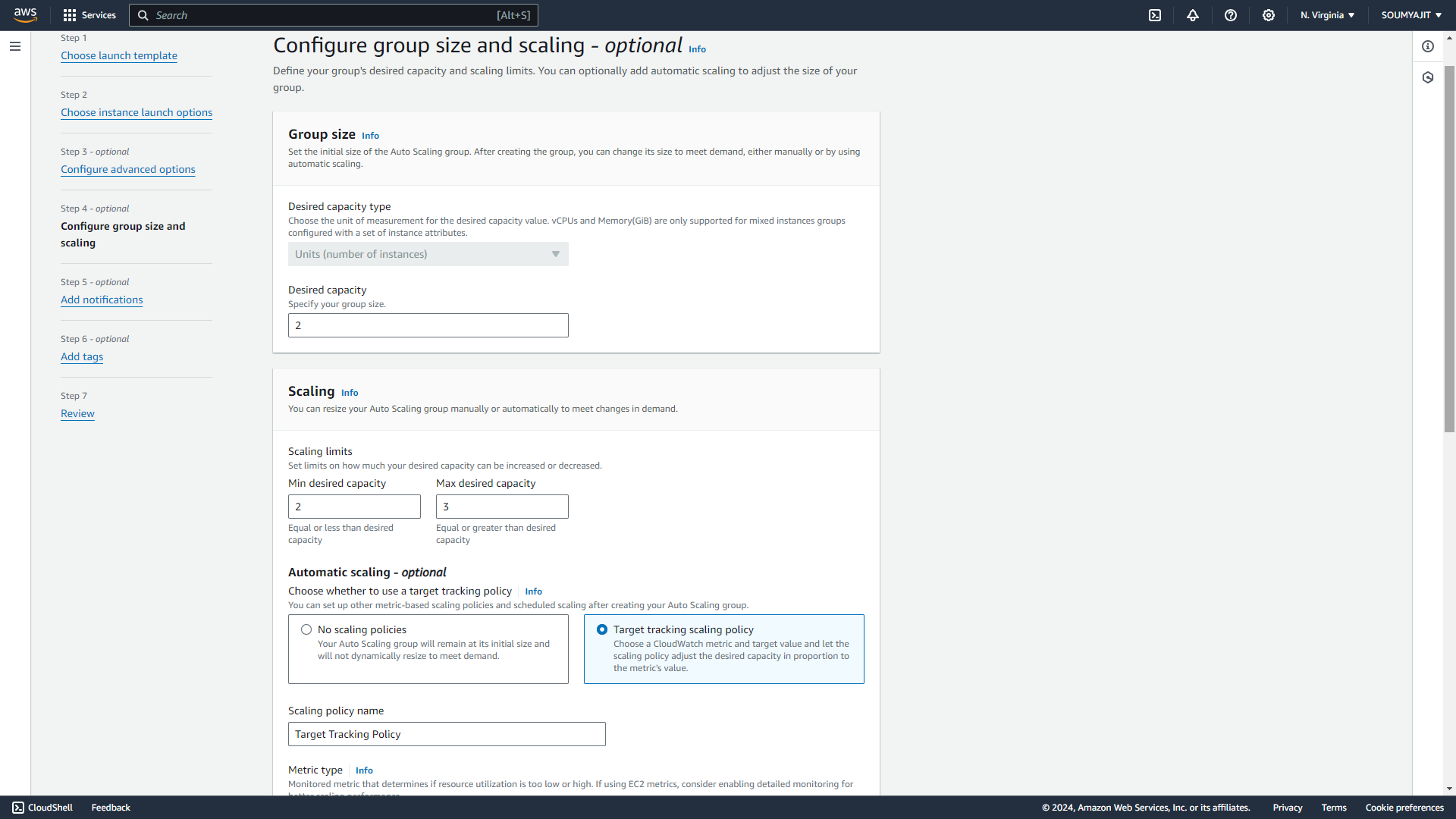
1. Modify the HTTP port number from 80 to 4000 and designate the scaling group created for default routing.



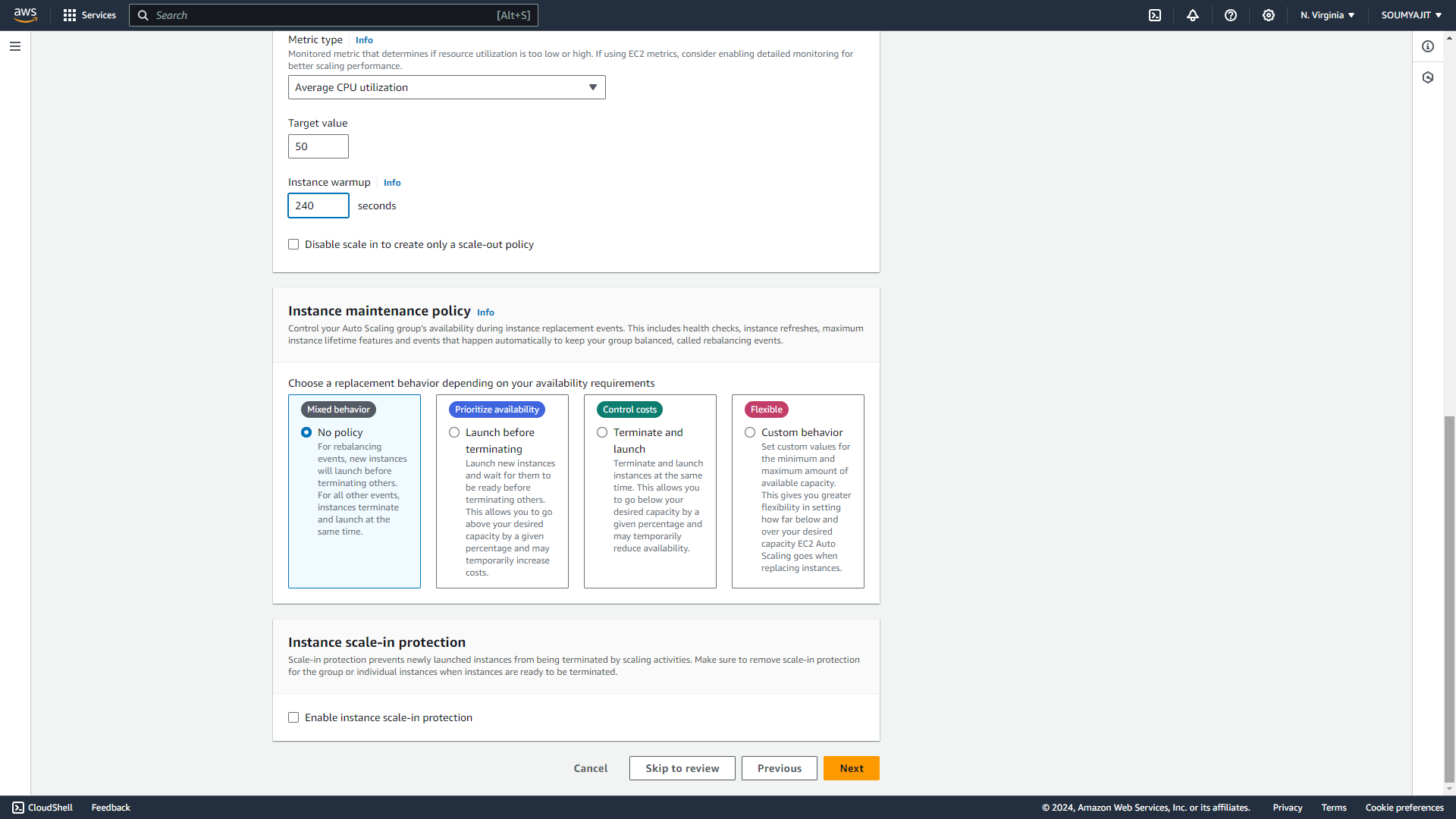
1. Enable the checkbox to activate health checks and specify a “health check grace period”, set here to 224 seconds. Without any further modifications, proceed to click on “Next”.



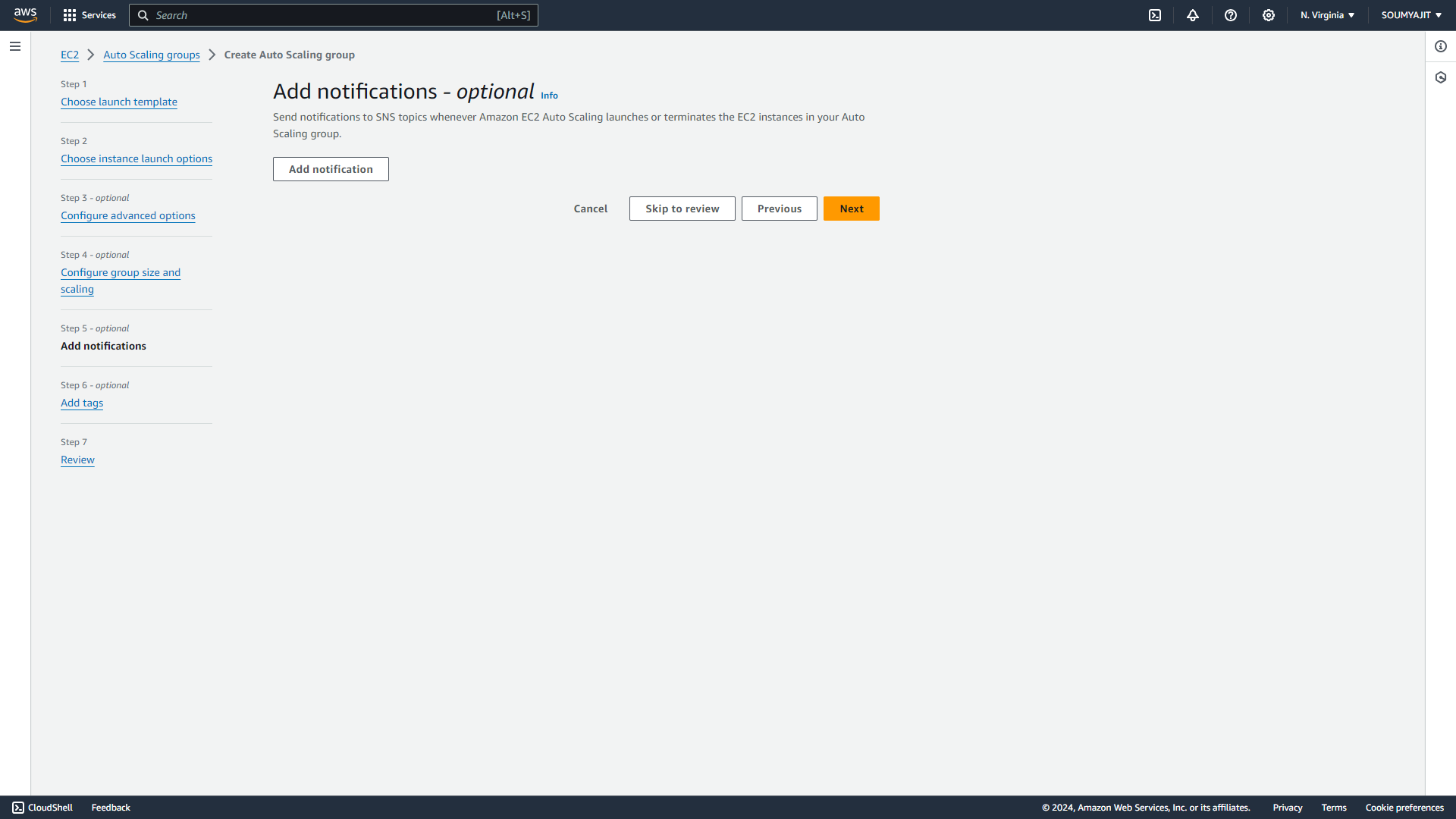
1. In this step, specify the desired, minimum, and maximum capacities. Next, opt for the "Target Tracking Scaling Policy"

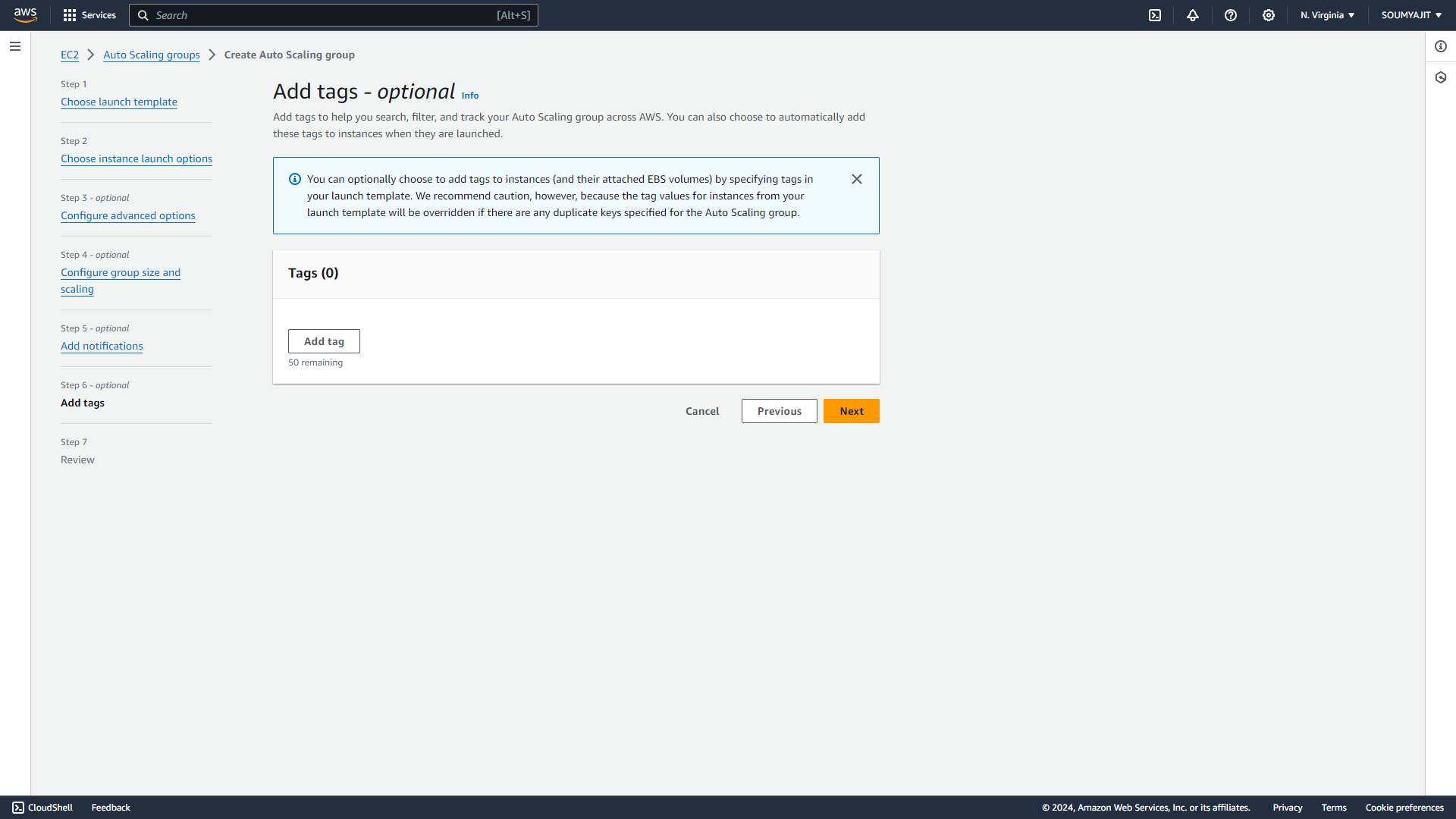


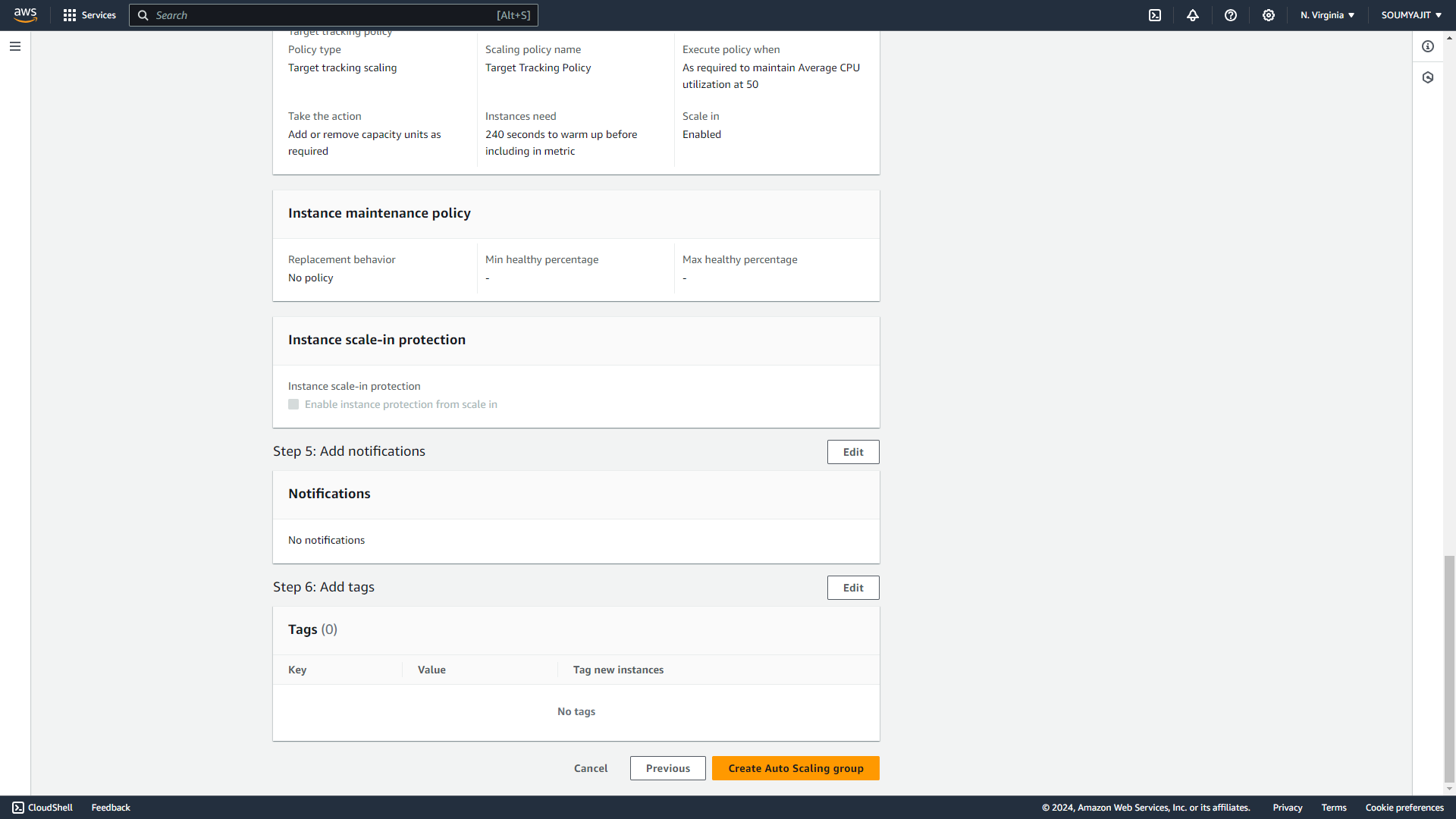
1. Configure the CPU utilization target value to 50. Additionally, set the instance warm-up time to 240 seconds. Proceed by clicking "Next" without making any changes



1. Click “Create Auto Scaling group”

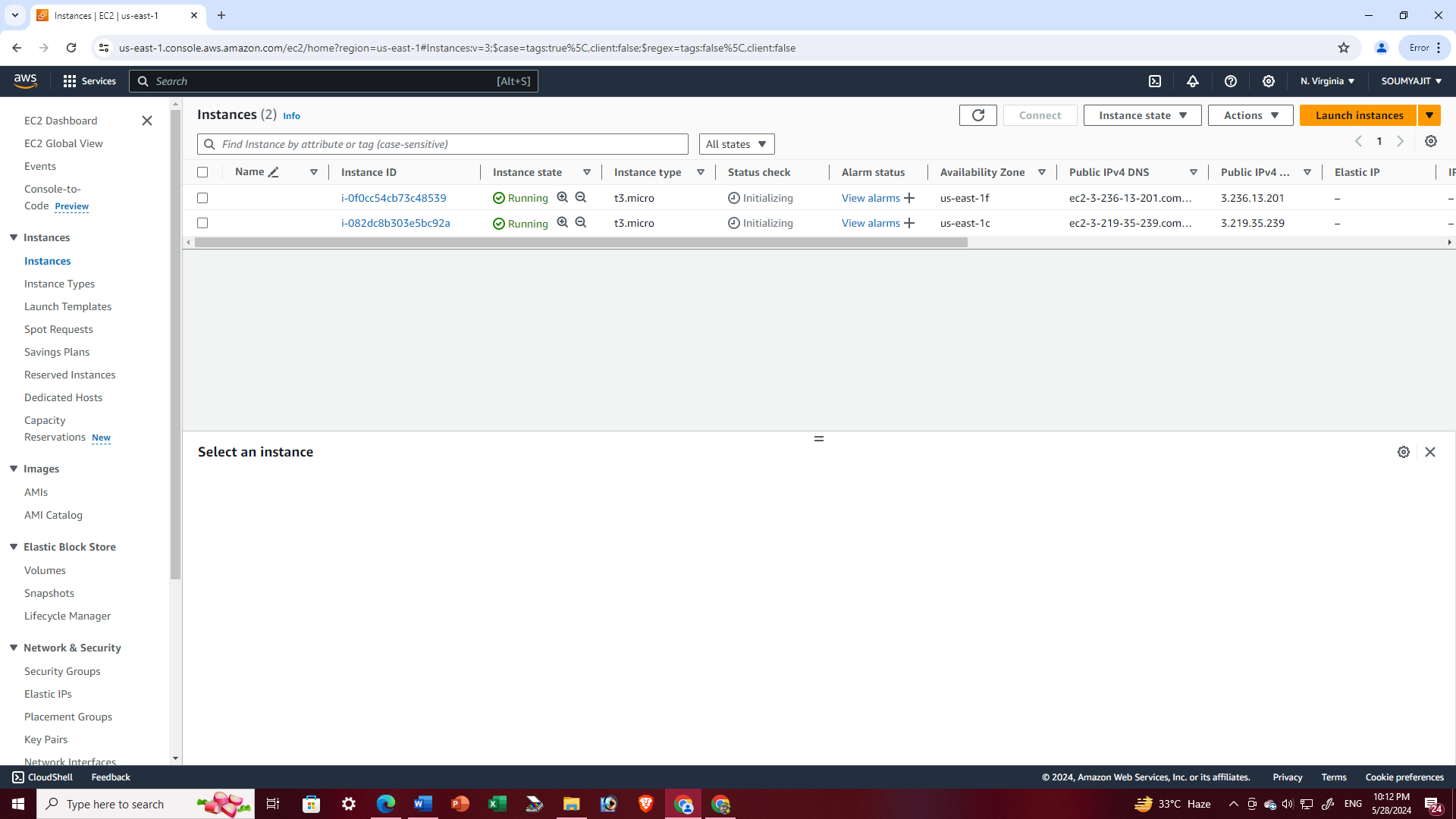


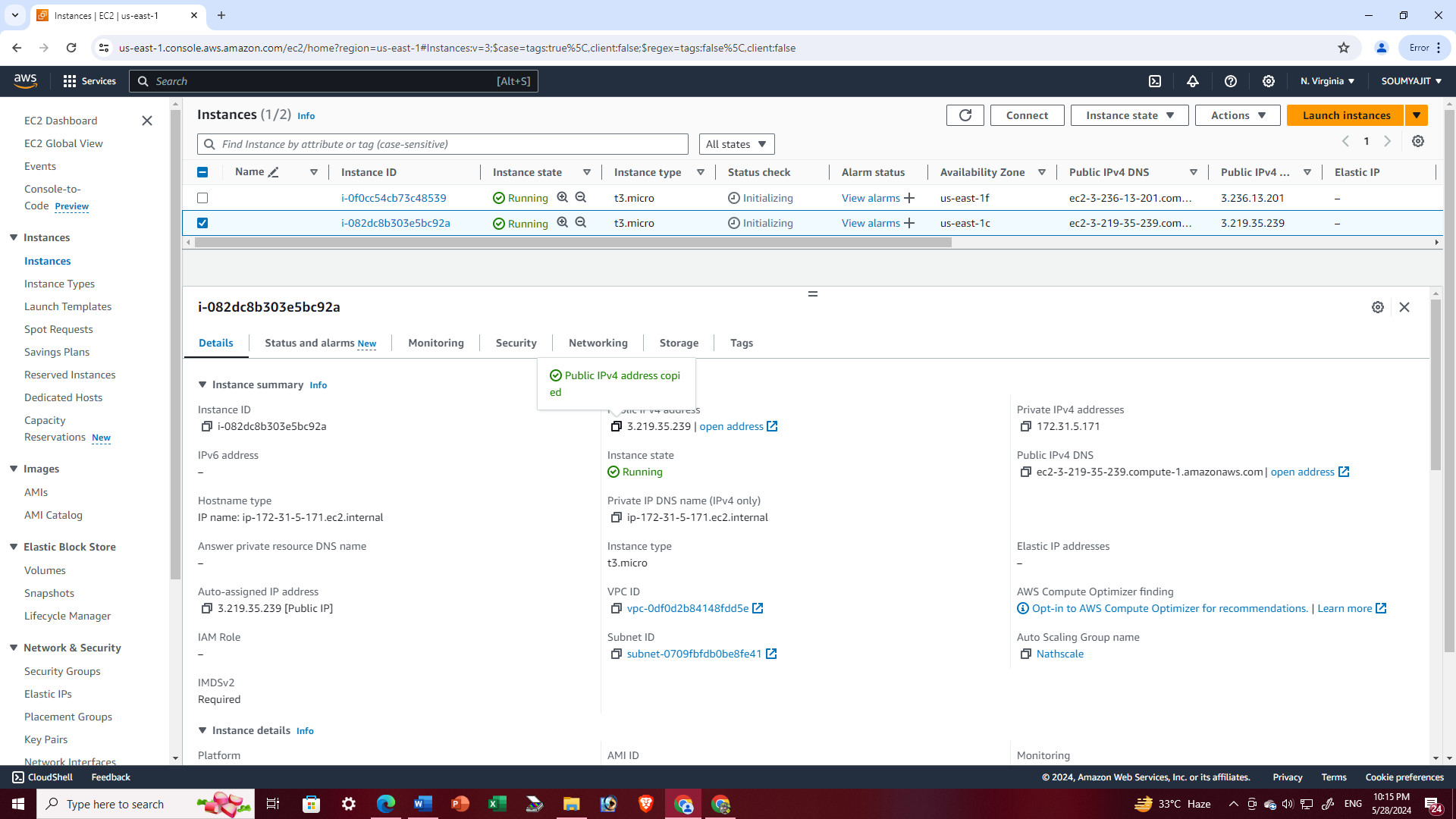




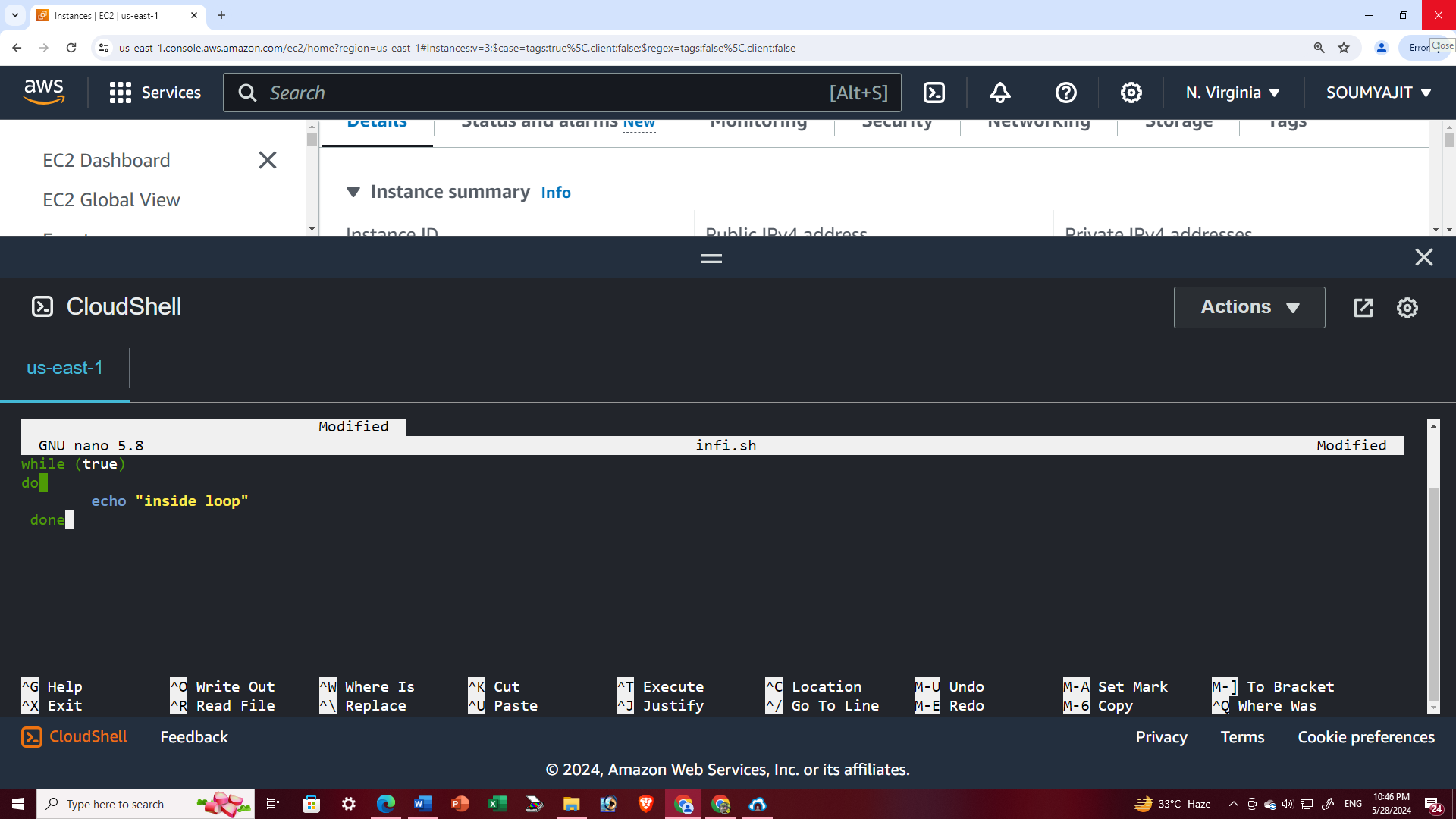
1. Auto scaling group created successfully

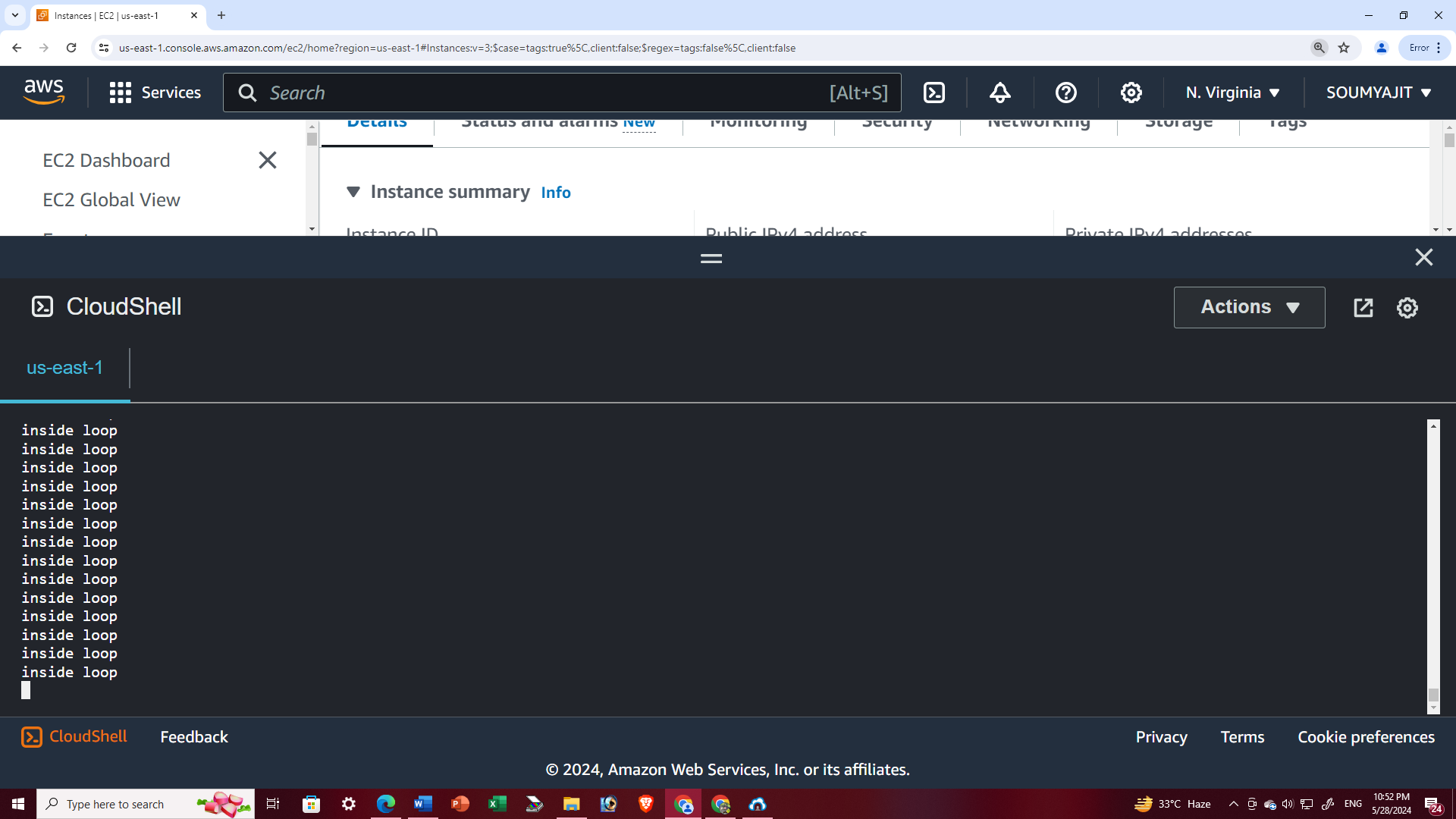


1. After the auto-scaling group is created, return to the EC2 dashboard and navigate to the "Instances" Since the capacity was given as 2, two instances are created. Now open any one of the instance by clicking on its id.  
   
2. Choose any one of the instance IDs and copy the public IPv4 address.



1. Open Bitwise SSH Client and log in using the IPv4 address we copied earlier. If bitvise not open then open CloudShell terminal and write “**sudo nano infi.sh**” command so creates a .sh file and then write this following code in the file “infi.sh” to run an infinite loop. Press CTRL+X ,Y then enter for save the file.  
   write “**sudo chmod 777 infi.sh**” (to provide all permission to the file)  
   **sh infi.sh** run the .sh file infinite time

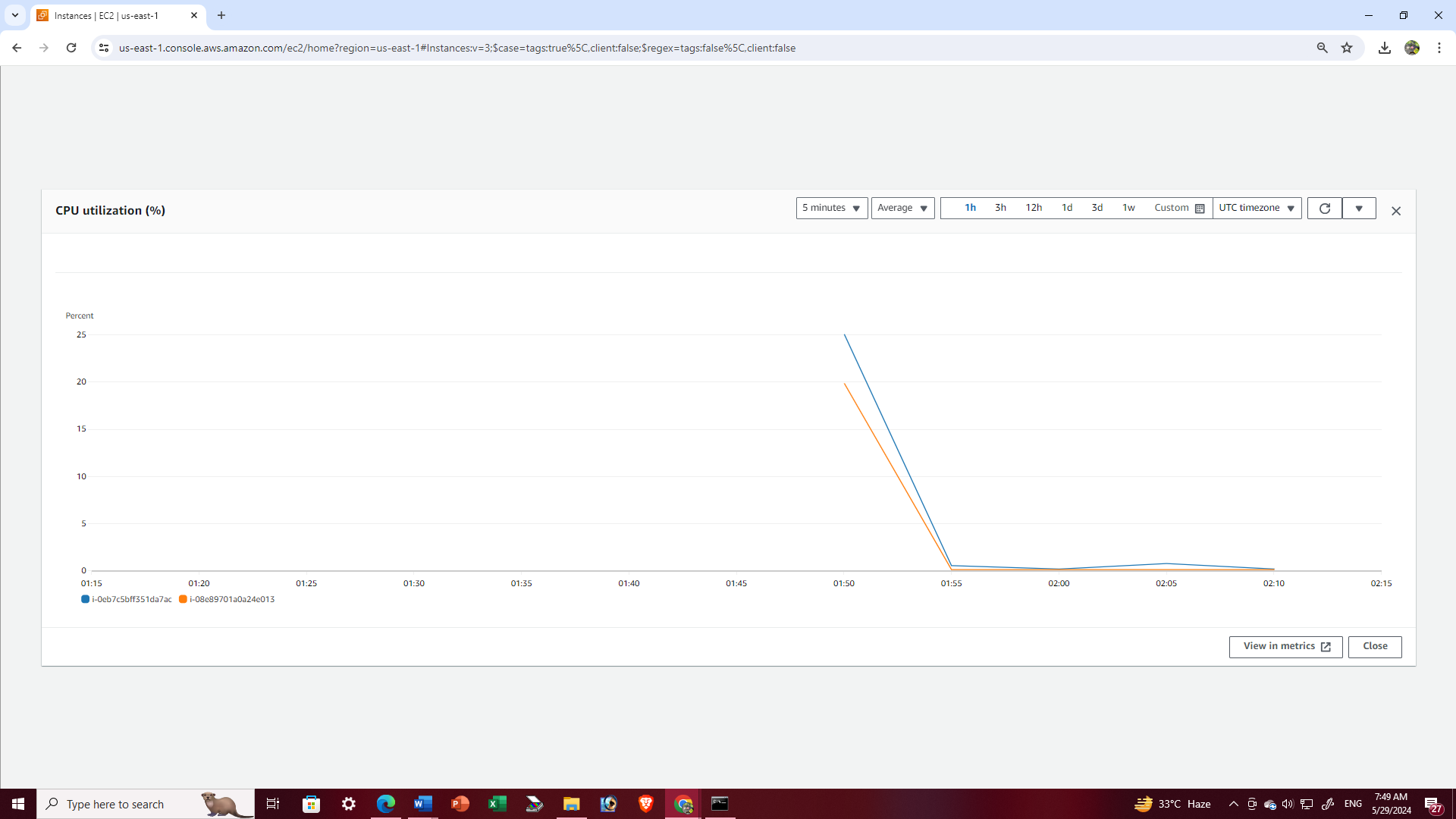


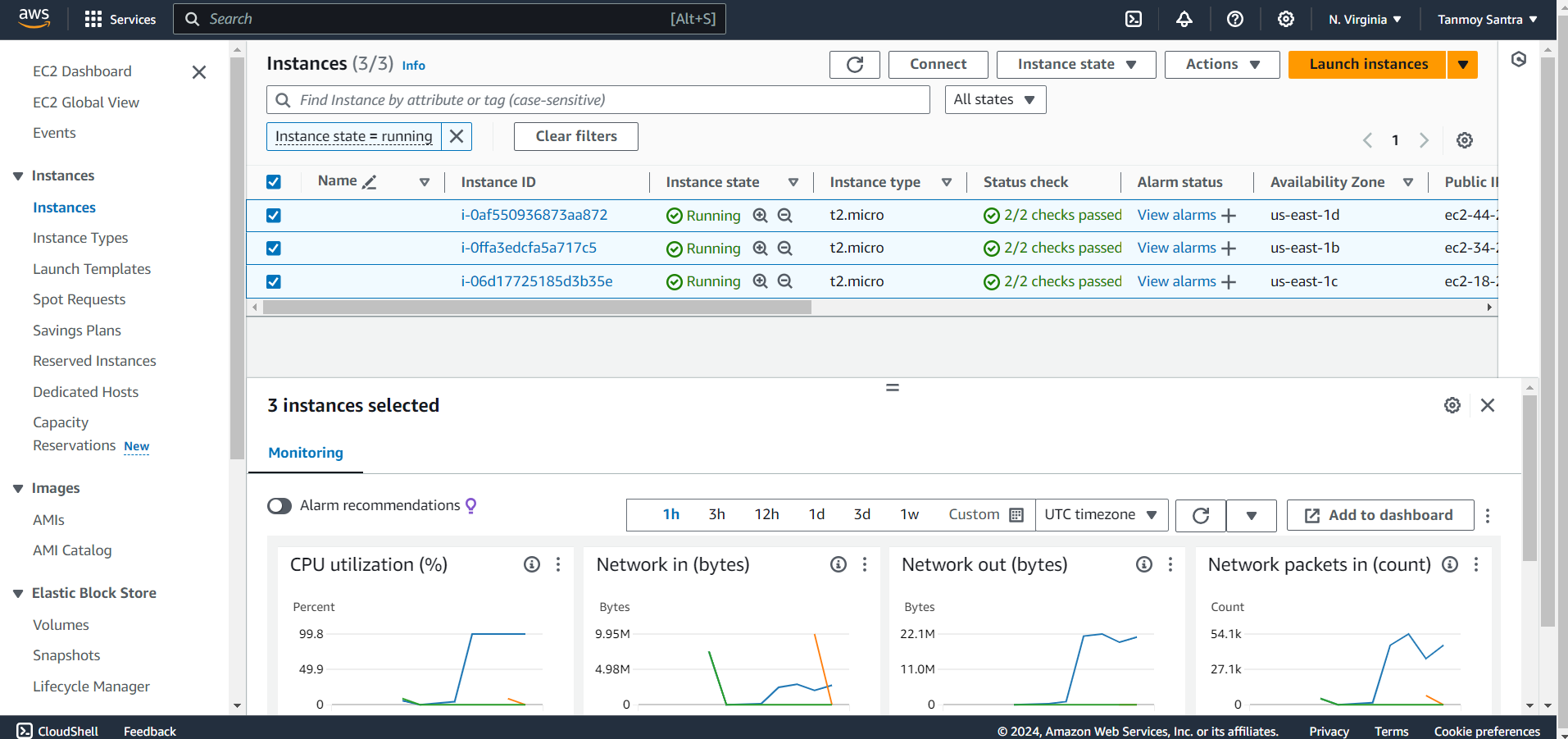


1. Return to AWS and select both running instances. Below, locate the monitoring options, and choose "CPU utilization". Then, enlarge the view.
2. From the panel above, select "Local timezone." The graph displays CPU utilization for both instances.

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

CPU utilization of two instance





After exceed CPU utilization limit the three instances are created