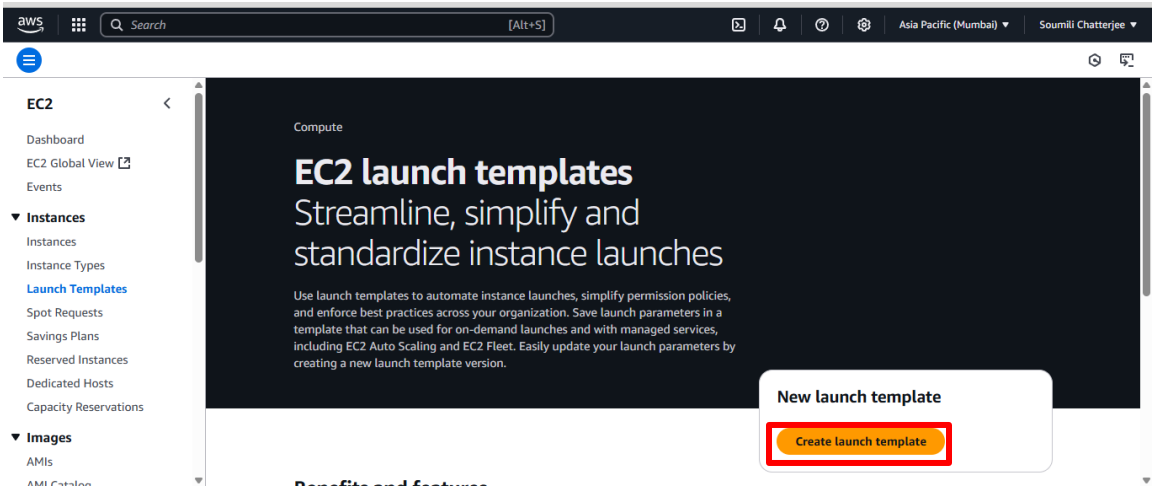
Assignment No. – 11

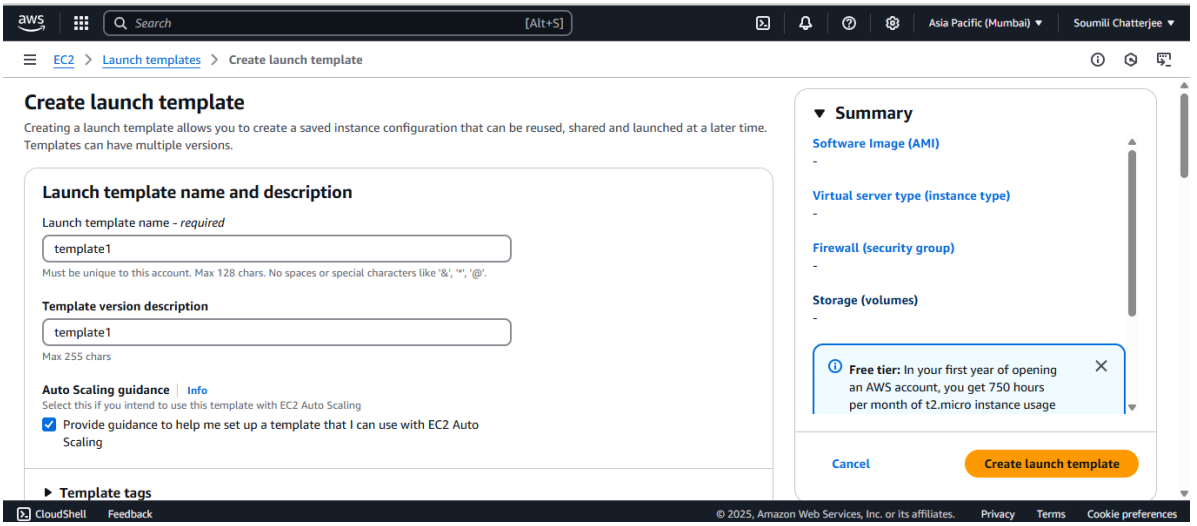
Problem Statement: Build scaling plans in AWS that balance the load on different EC2 instances.

Procedure:

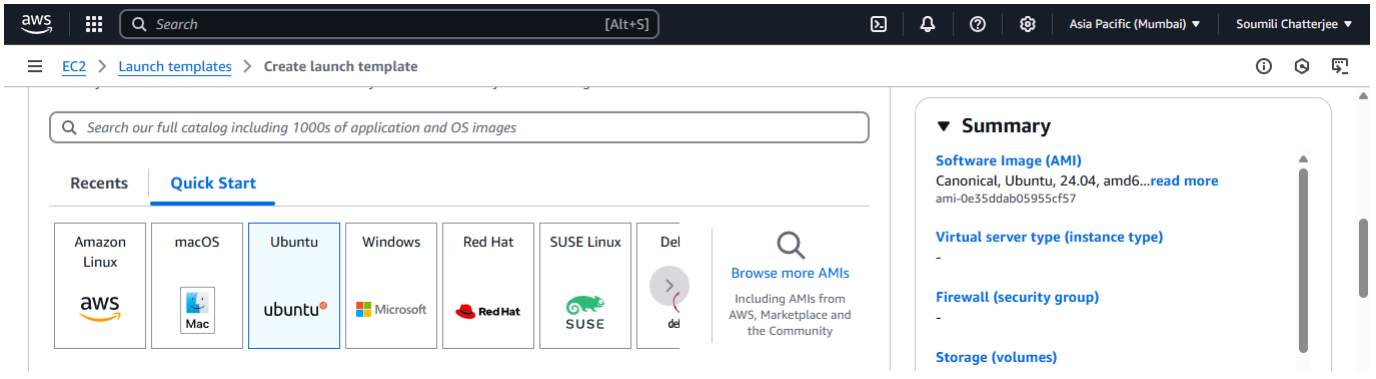
1. **Sign In** to the **AWS** account.
2. Go to **EC2**.
3. Under **Instances** section click on **Launch Templates**. Next, click on **Create launch template**.



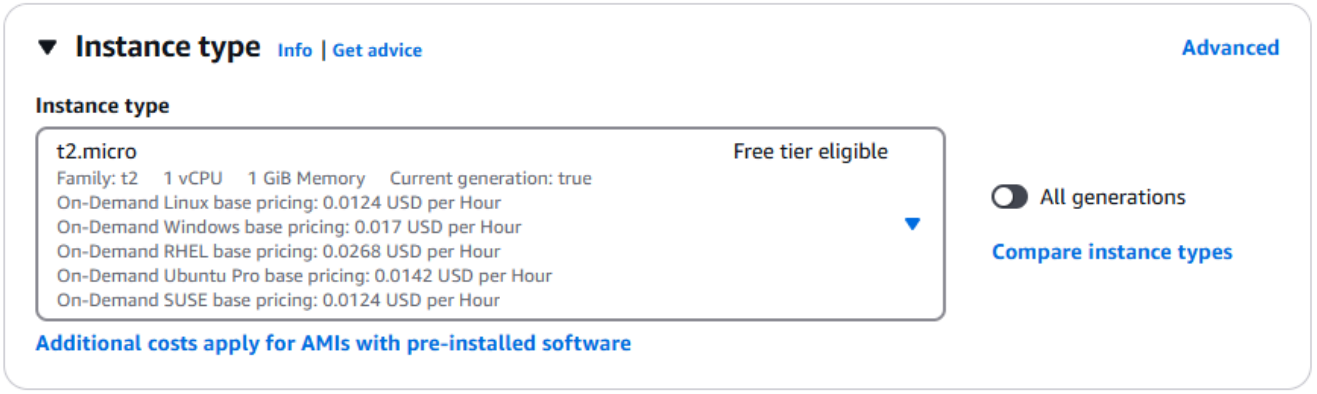
4. Now, give a name to the template and write the same in the description below and check the "**Provide guidance**" box.



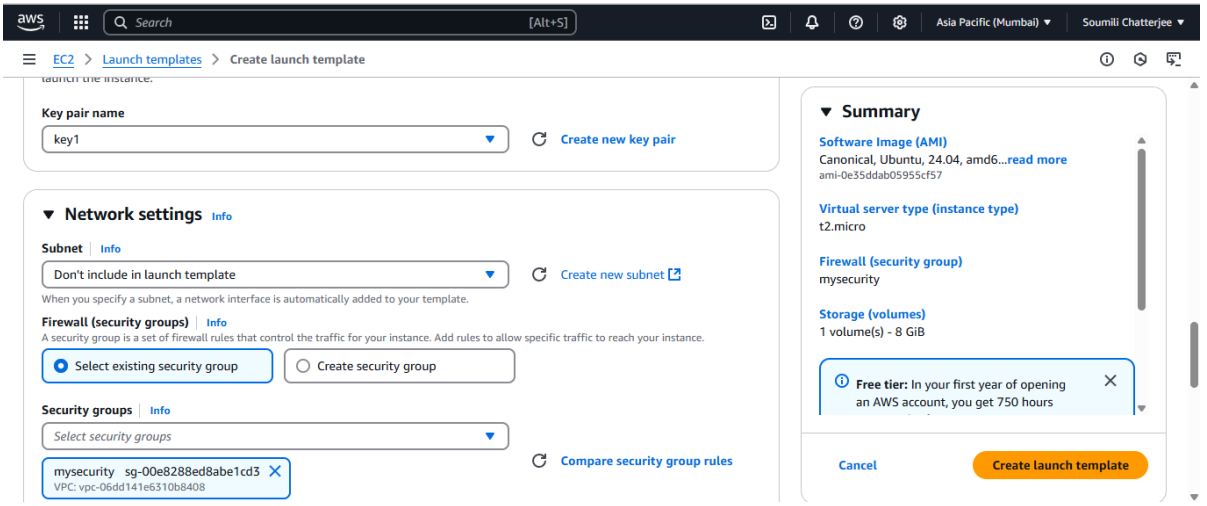
5. Next, under the **Quick Start** menu select **Ubuntu** as the **OS**.



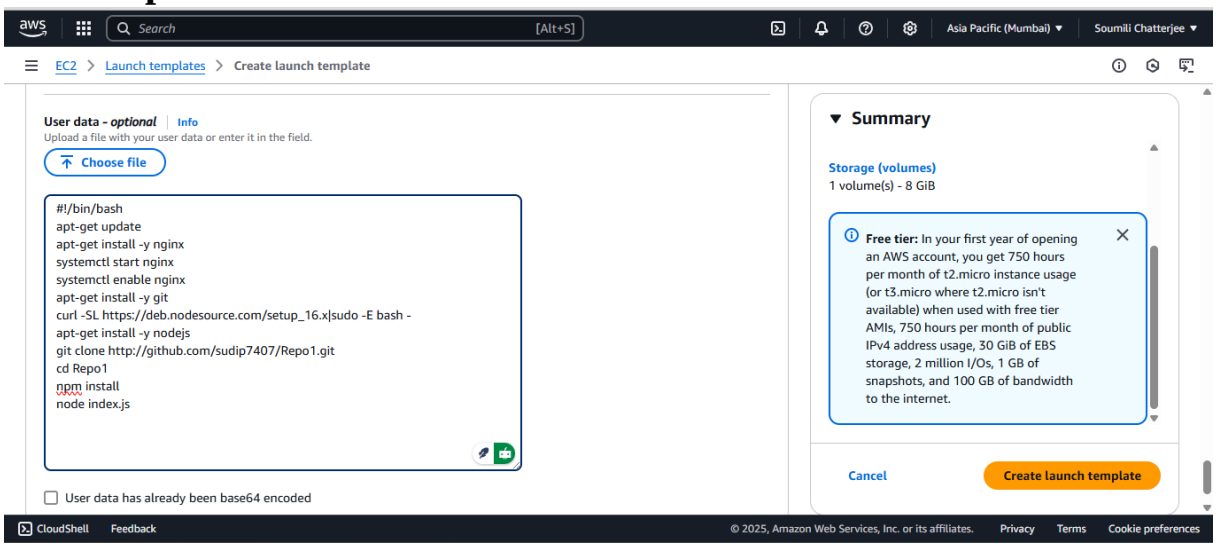
6. Next, under **Instance type** select **t2.micro** type of configuration.



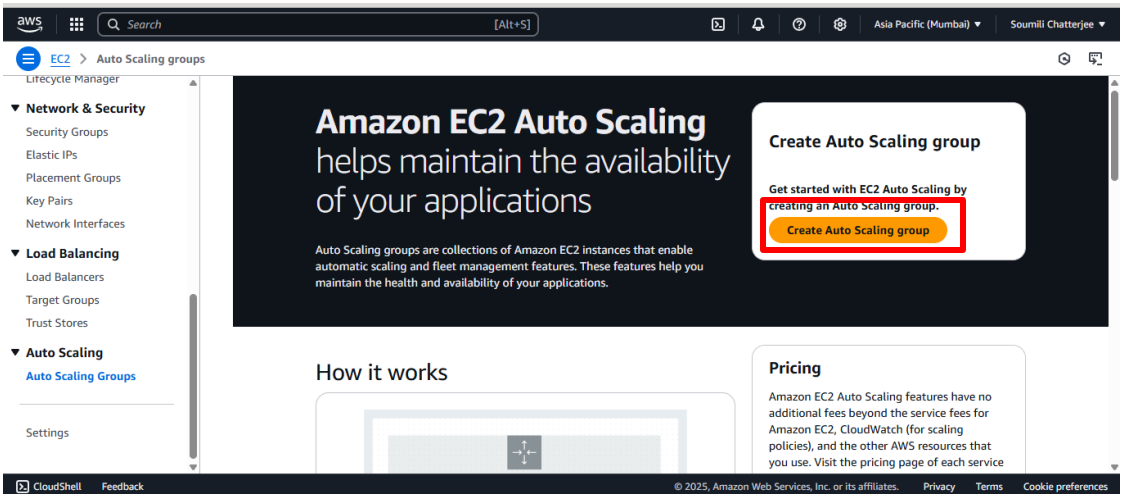
7. Select **Existing Key-Pair** and **Security Group** and if not applicable then Generate or Create a Key-Pair or Security Group wherever required.



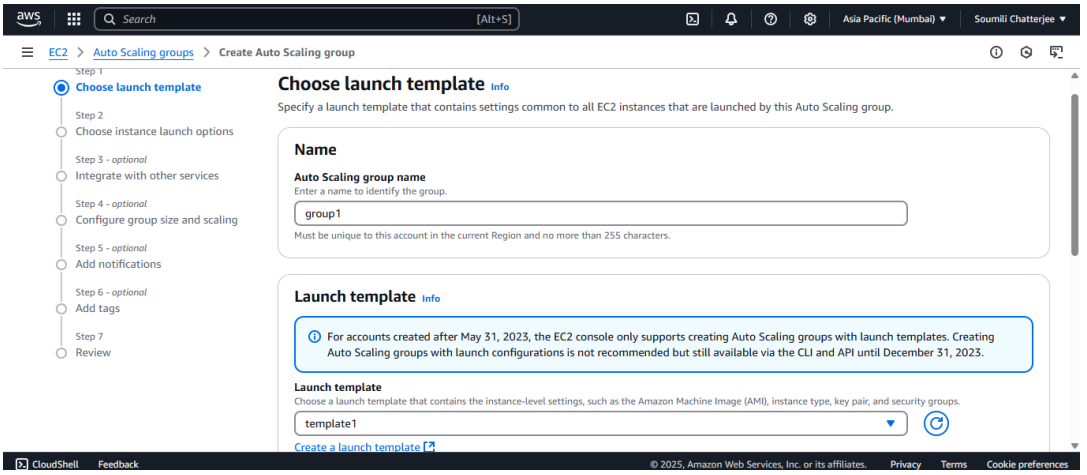
8. Now, click on the **Advanced Details** at the bottom. Scroll down to **User Data** Section and paste the commands in the box provided and click on the **Create Launch Template** button.



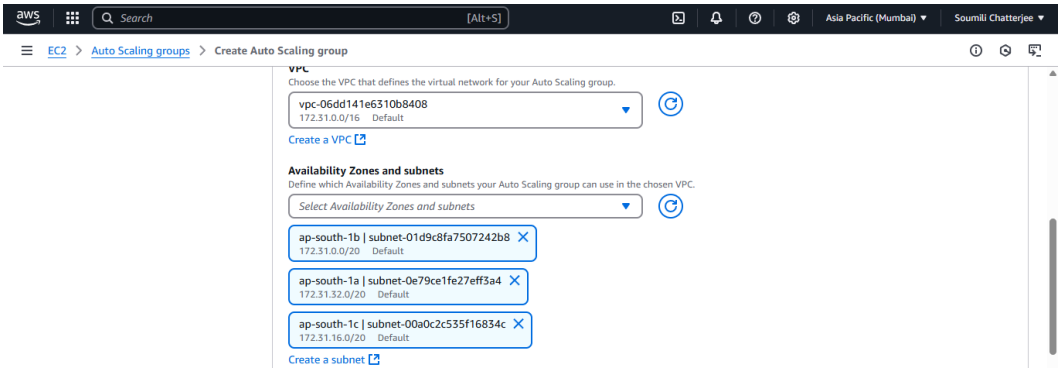
9. Go back to EC2 and from the Left side navigation bar go to **Auto Scaling Groups** under **Auto-Scaling section** and click on **Create Auto Scaling Group**.



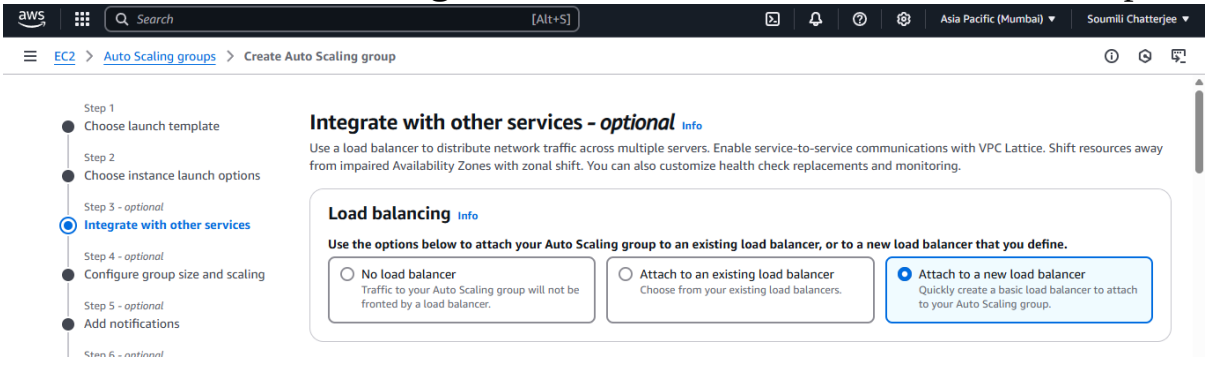
10. Give a **name** to your new Auto Scaling Group. Also, select the Launch Template that we recently created by using the drop-down menu under **Launch Templates section** and then click on **next**.



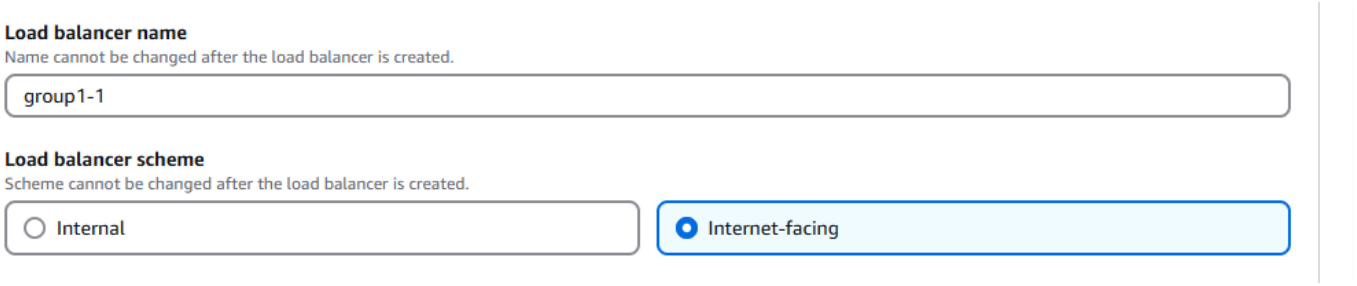
11. After that, under **Availability Zones and Subnets** select all the zones that appear and then click on **next**.



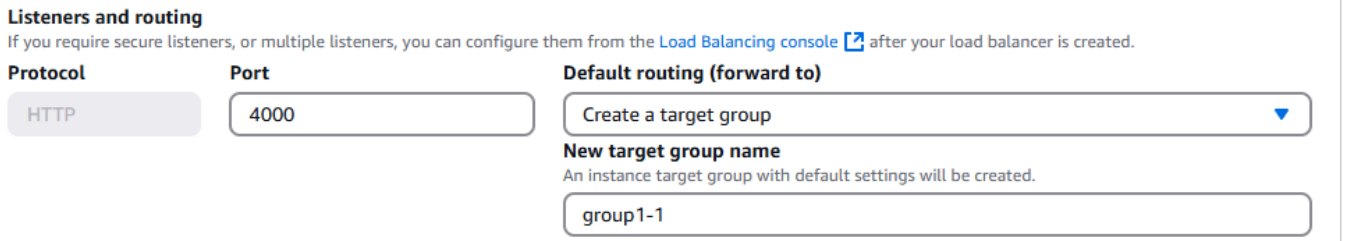
12. Now, under **Load Balancing** select **Attach to a New Load balancer** option.



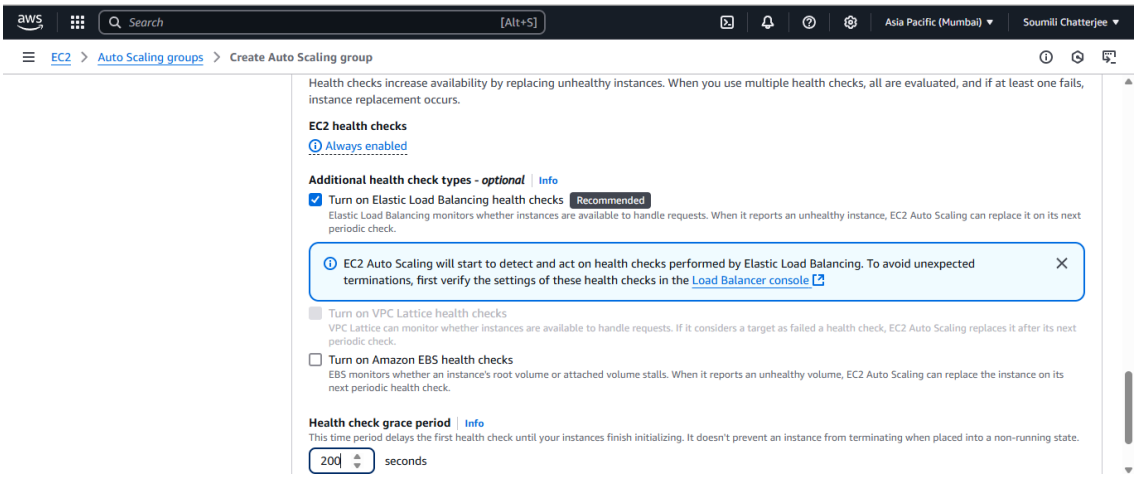
13. Now, select **Internet-Facing** under Load balancer scheme.



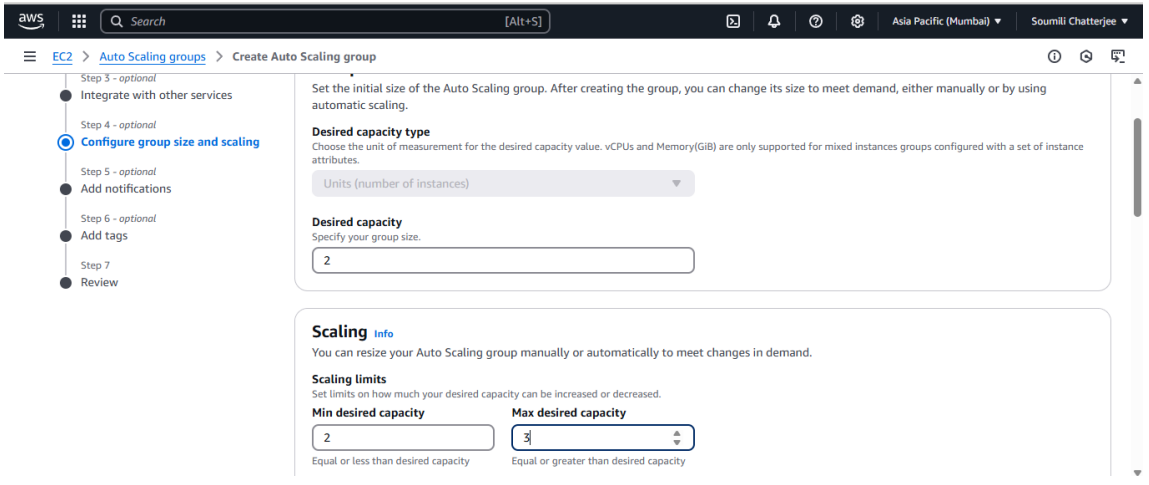
14. Under **Listeners and Routing** enter the port number (4000) and select **Create target group**.



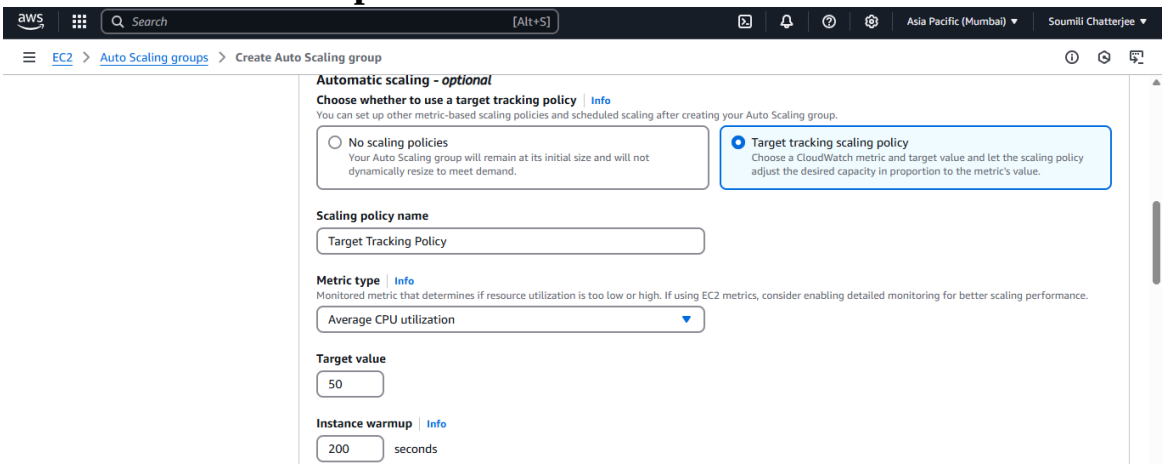
15. Next, under **Health checks** section, check the **turn of elastic load balancing health checks** box and change the health check grace period to **200** seconds and then click on **next**.



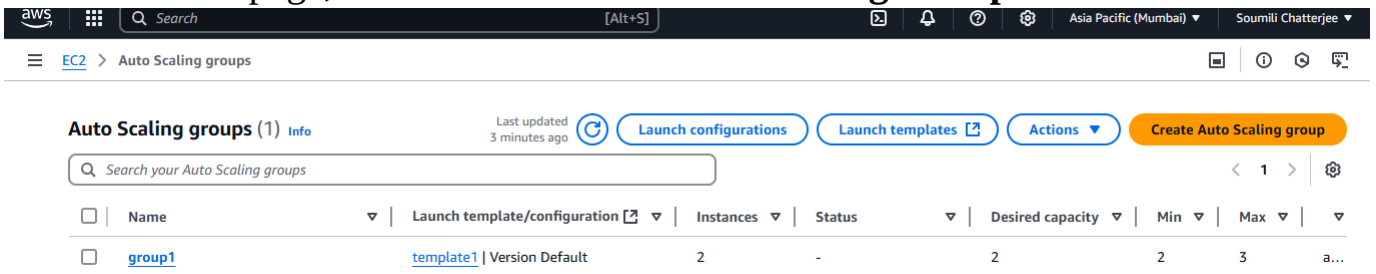
16. Under Group Size mention:
Desired Capacity = 2
Minimum Capacity = 2
Maximum Capacity = 3



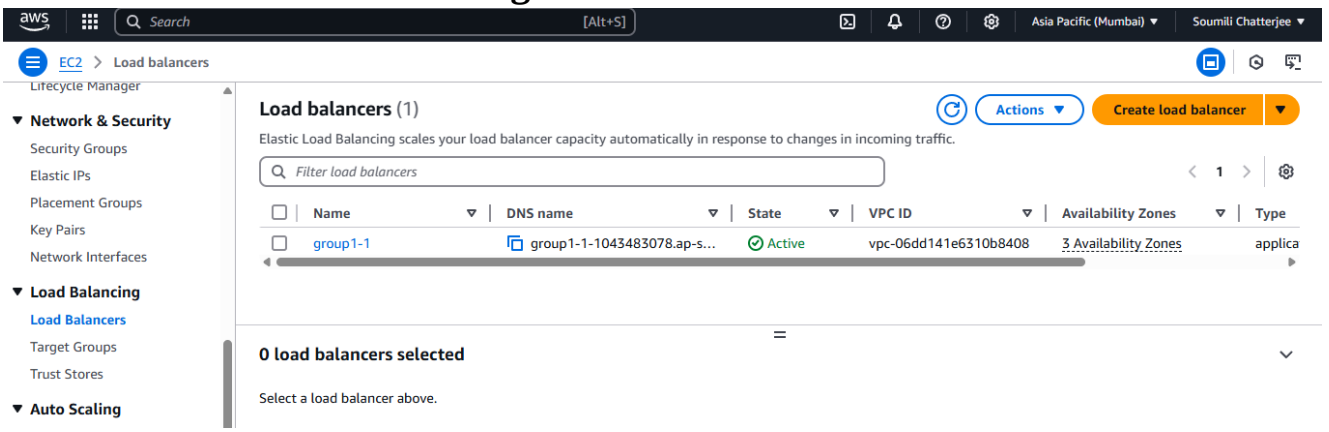
17. Now, under **Scaling policies** choose the **Target Tracking Scaling policy** option and set the **instance warmup** to **200** seconds and then click on **next**.



18. Nothing to change in **Notifications and Tags** page. So, click on the **Next** button. Under **Review** page, click on the **Create Auto-Scaling Group.**



19. Again go back to EC2 and from the Left side navigation bar go to **Load Balancers** under **Load Balancing section**.



20. Copy the **DNS name** and paste it in a new window followed by a colon and then the port number (4000).

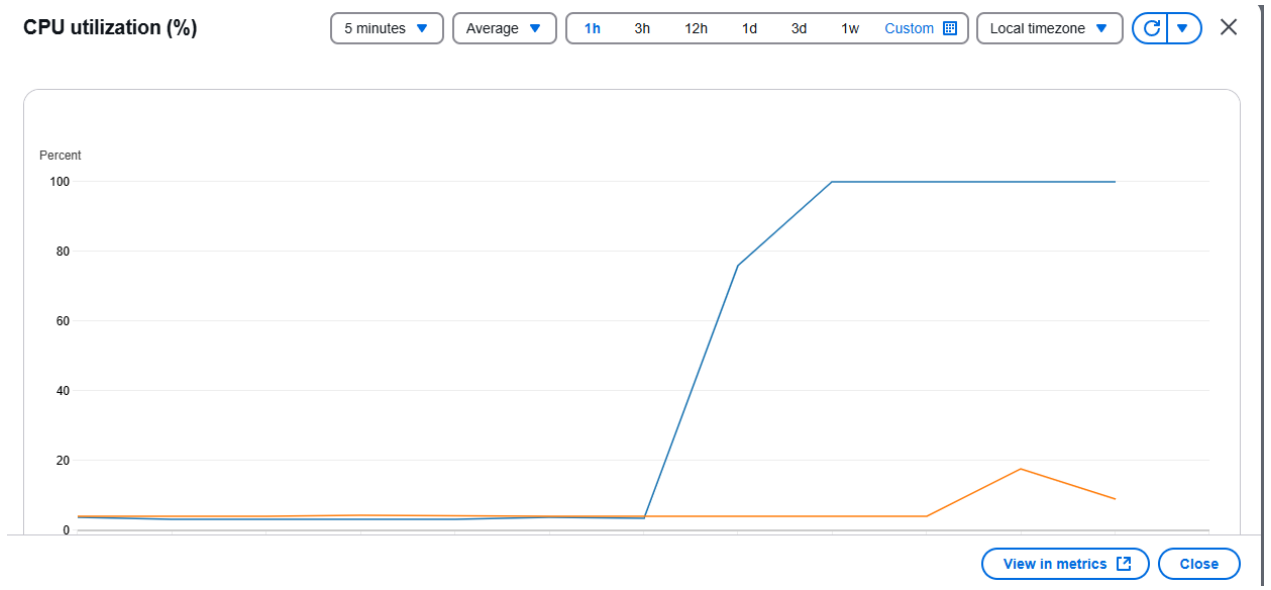
21. We will now **overload the server instances** and for that we need **Bitvise SSH client** for **instance 1**.

For instance 1:

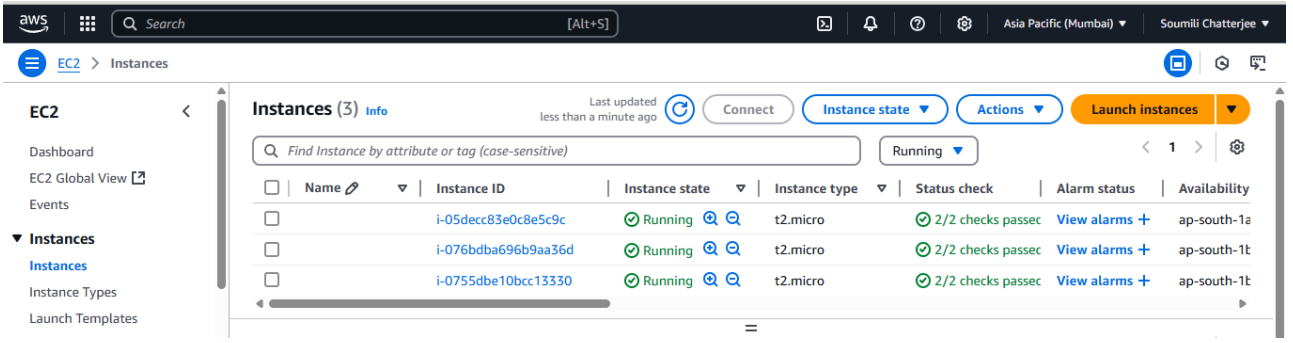
- Copy the public IPv4 address.
- Open Bitvise SSH client.
- Paste the IP and specify the necessary options.
- Now Log-In to the server.
- Open the new Terminal.
- Now enter the command:
  sudo nano infy.sh
- After the command a new nano Editor window will open. Type the following:
  #!/bin/bash
  while(true)
  do
  echo "Inside loop"
  done
  Now, to save and close the shell script we need to press the following shortcuts and keys sequentially:
  Ctrl+X
  Y
  Enter
- Now in the terminal type the following commands:
  sudo chmod +x infy.sh
  sh infy.sh

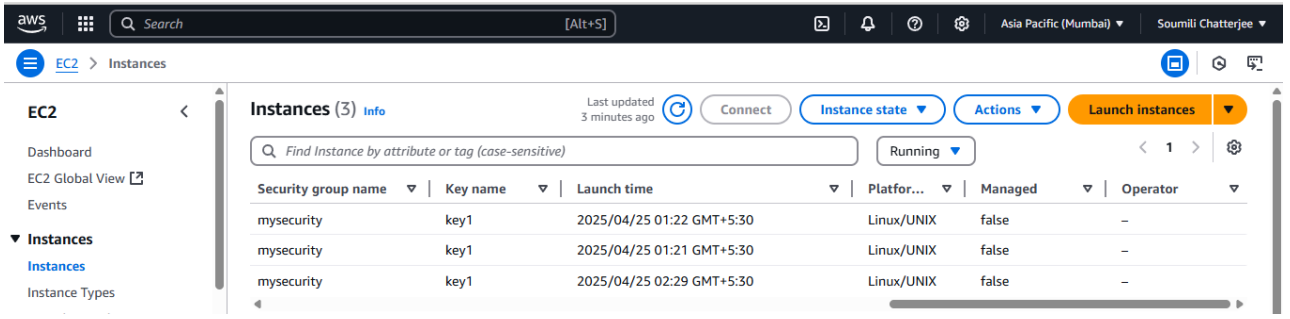
Now the script will start running infinitely. Do not close the terminal. Keep it Minimized and go to Ec2.

22. Now, select both the running instances and under **monitoring** go to **CPU Utilisation** and **enlarge** the window by clicking on the three dots. Choose the timezone as **Local Timezone.**



23. After some time, we can see there are 3 servers running. There can only be 3 servers running at a time for us as specified in our Auto-Scaling group when we were creating it. Hence, we have reached our maximum limit of instances running concurrently.





We have successfully created, configured and tested our Auto-Scaling Group.