1. Create auto scaling group with launch template,we do configure in template for instances.and we run some bash script in user data to install nginx.
2. We can select load balancer if we had,if not we can go with new load balancer option.
3. The launched instances by asg will be in the target group of load balancer.
4. After asg created,we do check how many instances we got by the information of

minimum,desire and Maximum values.

1. For automatic scaling we do create policy like target tracking,dynamic,step.

For that we do edit the asg where we create policy that is for now target tracking.

1. In this policy we set average cpu utilization some number and whenever cpu utilization may cross this limit We get one more instance to balance this load.like this we created

One target tracking policy here.

1. To check we do connect our instance,there we install stress utility by apt and

We do manually increase cpu utilization by stress command like

$ stress -c 5 #we can check cpu utilization by top command without interrupting to this.

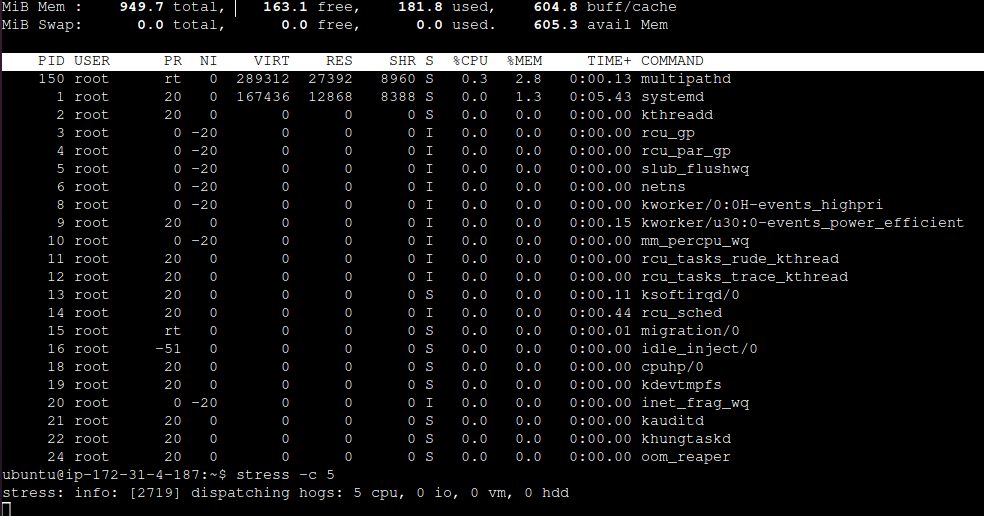
1. Few minutes later we can get one more instance by asg we can see in ec2-dashboard.

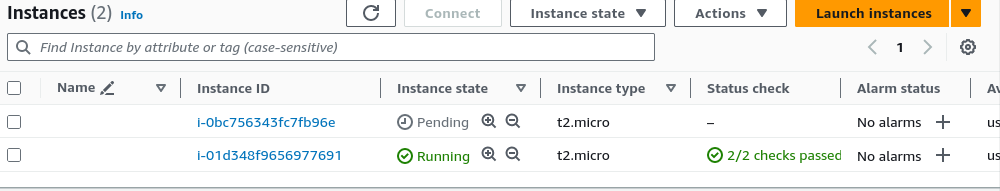
We can check server ip we get nginx web page.

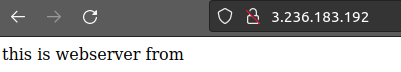
1. After we decrease manually by ctrl+c,a few sec we can see instance get terminating

automatically.

1. So this is the lab to check scaling by manually increasing cpu utilization load.









**Note :**

1. **Cooldown Time:Definition: Cooldown time is a setting in Auto Scaling Groups that represents the amount of time the Auto Scaling group should wait before allowing another scaling activity to take place.**
2. **Grace Period:Definition: The grace period is a related concept but refers to the time that Auto Scaling should wait after a new instance is launched before it starts to contribute to the metrics used for scaling decisions.**
3. **Some times we get too much time for 2 by 2 status check because of user data scripts.instead it we should take a golden image by this we can achieve better experience.**