



Placement Empowerment Program

Cloud Computing and DevOps Centre

Implement Auto-scaling in the Cloud : Set up an auto-scaling group for your cloud VMs to handle variable workloads.

Name: VIJAYA NANDANA M

Department: CSE



Introduction

Auto Scaling in AWS is a powerful feature that automatically adjusts the number of EC2 instances in response to traffic demand. This ensures high availability, cost efficiency, and optimal performance. By defining a Launch Template, creating an Auto Scaling Group (ASG), and setting up scaling policies, we can dynamically scale instances based on CPU utilization or other metrics.

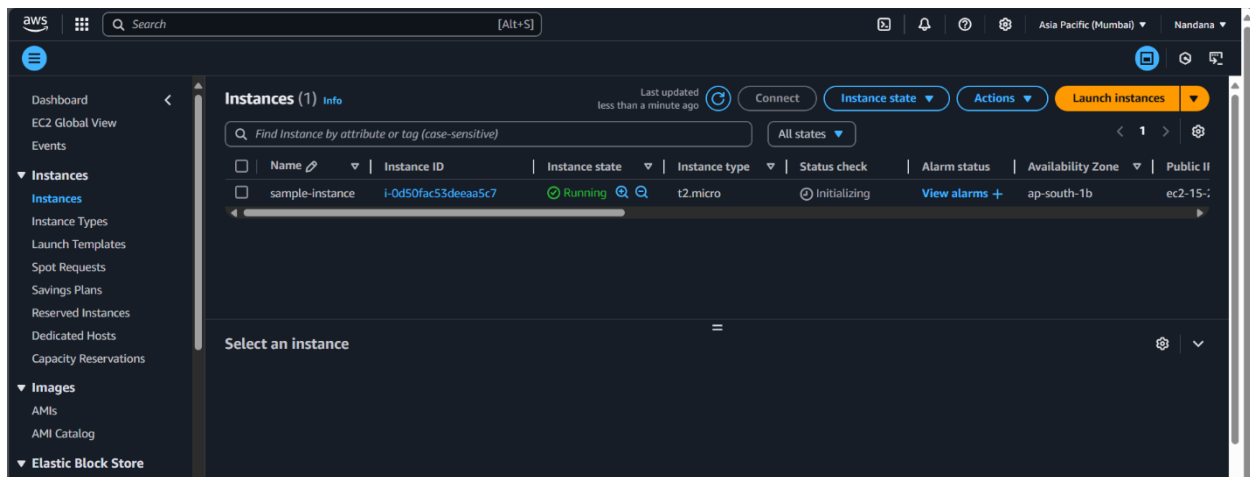
Objectives

- Create a Launch Template to define the configuration for EC2 instances.
- Set up an Auto Scaling Group (ASG) to manage instance scaling.
- Define Scaling Policies to automatically increase or decrease instances based on CPU utilization.
- Test Auto Scaling by simulating high CPU usage and verifying instance scaling.

Step by Step Overview

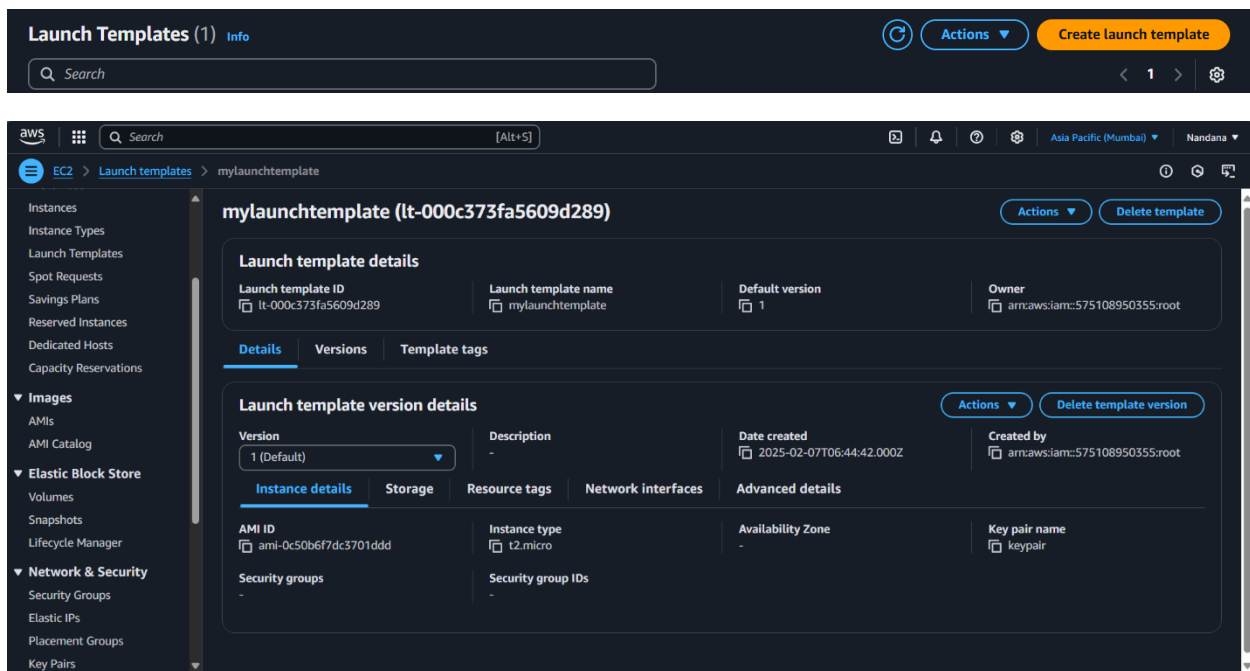
1. Create an EC2 instance

- log into your aws account.
- create an EC2 instance.



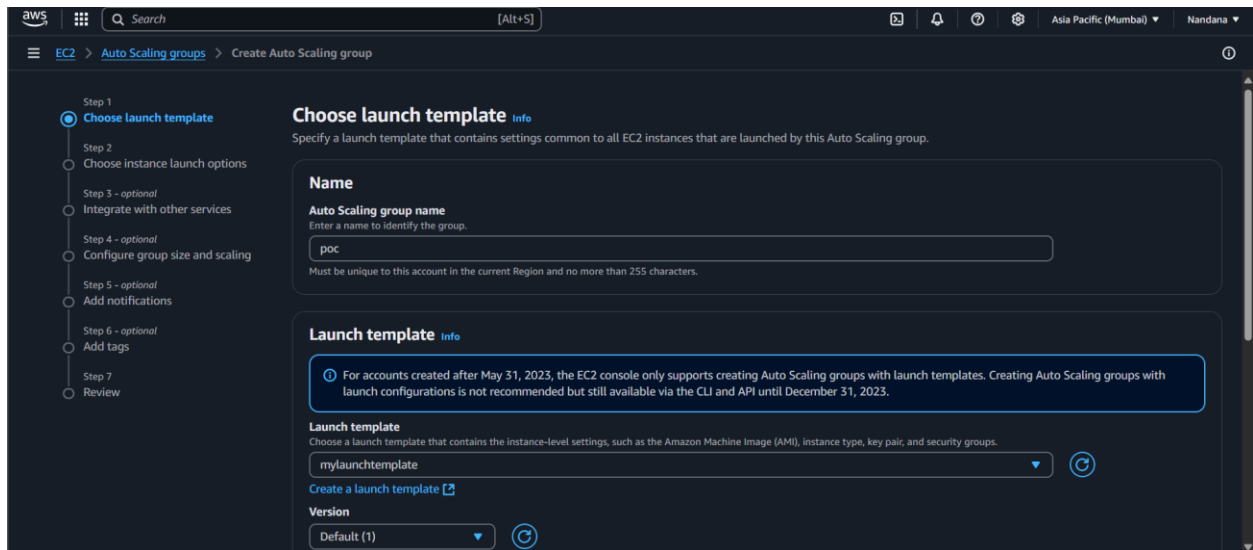
2. Create Launch Template

- In the left panel, click on Launch Templates
- Click Create Launch Template
- Enter a name (e.g., MyLaunchTemplate)
- Select an AMI (Amazon Machine Image)
 - Choose a relevant Linux or Windows AMI
- Choose an Instance Type (e.g., t2.micro)
- Choose an IAM Role (if required)
- Add Key Pair for SSH access
- Add Security Groups (allow SSH & required ports)

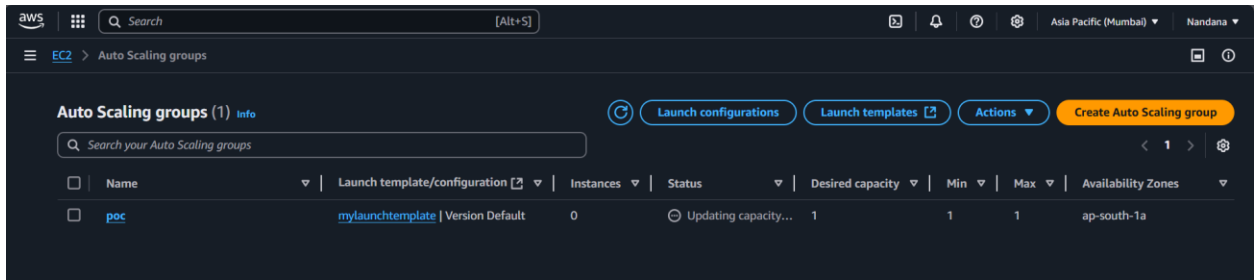


3. Create Auto Scaling Group

- In the EC2 Dashboard, click Auto Scaling Groups
- Click Create Auto Scaling Group
- **Select the Launch Template**
- Choose the Launch Template created earlier

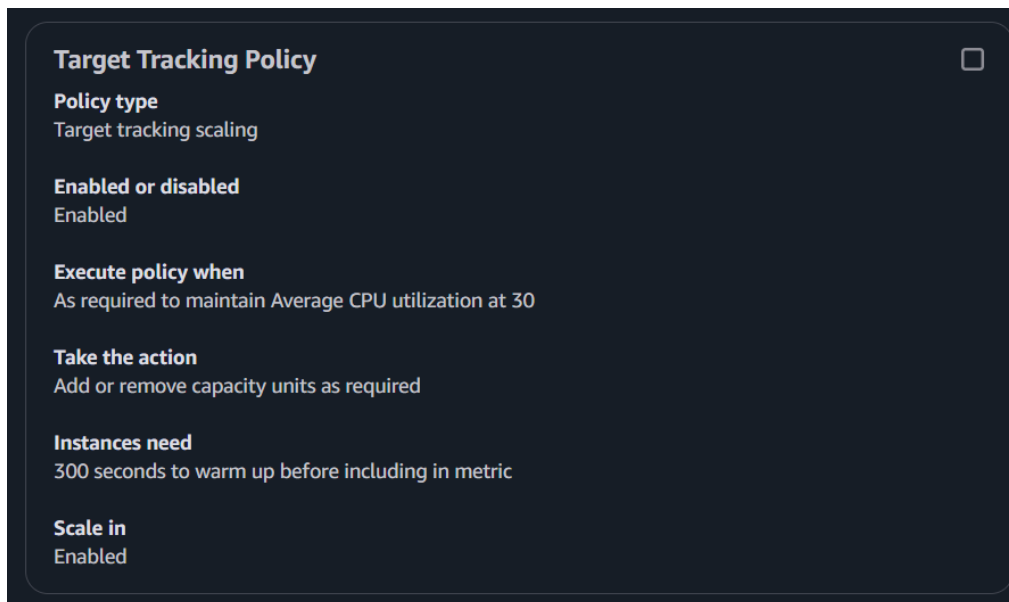
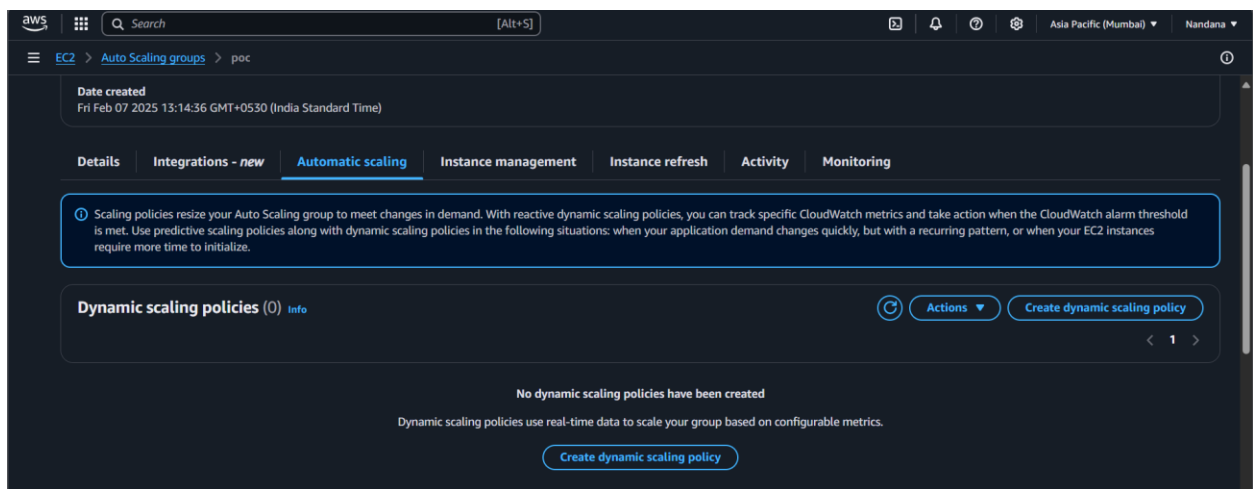


- **Configure the Auto Scaling Group**
 - Set Auto Scaling Group Name (e.g., MyAutoScalingGroup)
 - Select VPC & Subnets
 - Click Next
- **Set Desired Capacity & Scaling**
 - Desired Capacity: 1
 - Minimum Instances: 1
 - Maximum Instances: 5
 - Click Next
- **Configure Health Checks & Load Balancing (optional)**
 - Enable ELB (optional)
 - Enable Health Checks
 - Click Create Auto Scaling Group



4. Create Scaling Policy

- Go to Automatic scaling and create scaling policy.



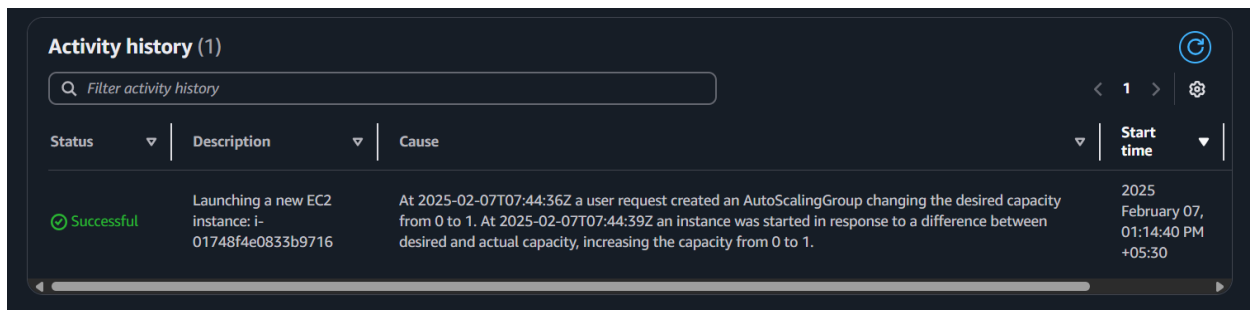
5. Simulate High CPU Usage

SSH into your system through command prompt. And then simulate stress.

```
sudo yum install -y stress
stress --cpu 4 --timeout 300
```

6. Monitor Scaling Events

- Go to Auto Scaling Groups
- Click on Activity to check scaling actions

A screenshot of the AWS Auto Scaling console showing the 'Activity history' for a specific Auto Scaling Group. The interface includes a search bar, a table with columns for Status, Description, Cause, and Start time, and a single activity entry. The status is 'Successful', the description mentions launching a new EC2 instance, and the start time is 2025 February 07, 01:14:40 PM +05:30.

Activity history (1)			
Filter activity history			
Status	Description	Cause	Start time
Successful	Launching a new EC2 instance: i-01748f4e0833b9716	At 2025-02-07T07:44:36Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 1. At 2025-02-07T07:44:39Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 1.	2025 February 07, 01:14:40 PM +05:30

Outcome:

- A Launch Template configured with an EC2 instance setup.
- An Auto Scaling Group (ASG) that ensures automatic instance scaling.
- Scaling policies that trigger new instance launches or terminations based on CPU usage.
- Successfully tested Auto Scaling by generating high CPU load and observing instance scaling in real time.