

Little Bit Advance Labs

Part 1: EC2 with ELB and ASG

Objective:

Learn how to create a scalable and highly available web application environment using Amazon EC2 instances, ELB, and ASG.

Approach:

Launch EC2 Instances: Start by launching two or more EC2 instances. These instances will run a simple web application (e.g., a "Hello World" page or any basic web service).

Configure Load Balancer: Set up an Elastic Load Balancer (ELB) to distribute incoming web traffic across your EC2 instances. This step ensures high availability and fault tolerance.

Set Up Auto Scaling Group (ASG): Create an ASG that uses the launched EC2 instances. Configure ASG policies to automatically scale the number of instances up or down based on criteria like CPU usage or network traffic.

Test Your Setup: Simulate traffic to test the scaling policies and the load balancer. Observe how ASG adds or removes instances and how ELB distributes traffic.

Verify Website Functionality: Ensure that the website hosted on EC2 instances remains accessible and functional during scaling operations.

Goal:

By the end of this lab, students will have a hands-on understanding of setting up a load-balanced and auto-scaled web application using AWS services

1. EC2 Instance

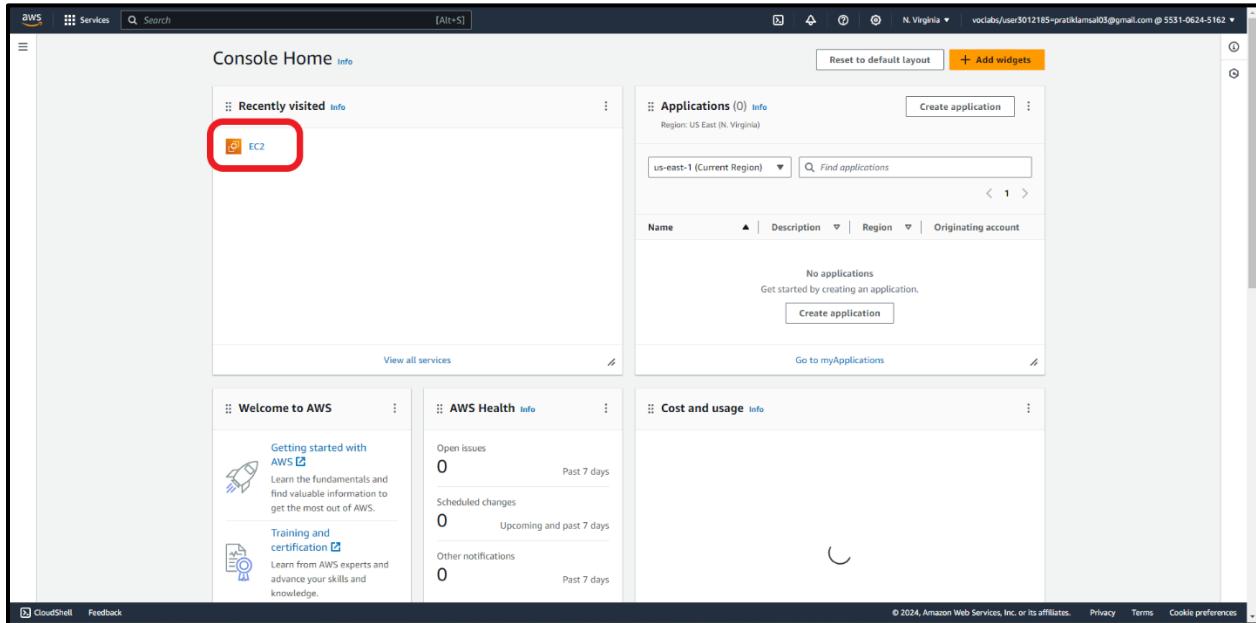


Figure 1 Selecting EC2 Instance

2. Launching an Instance

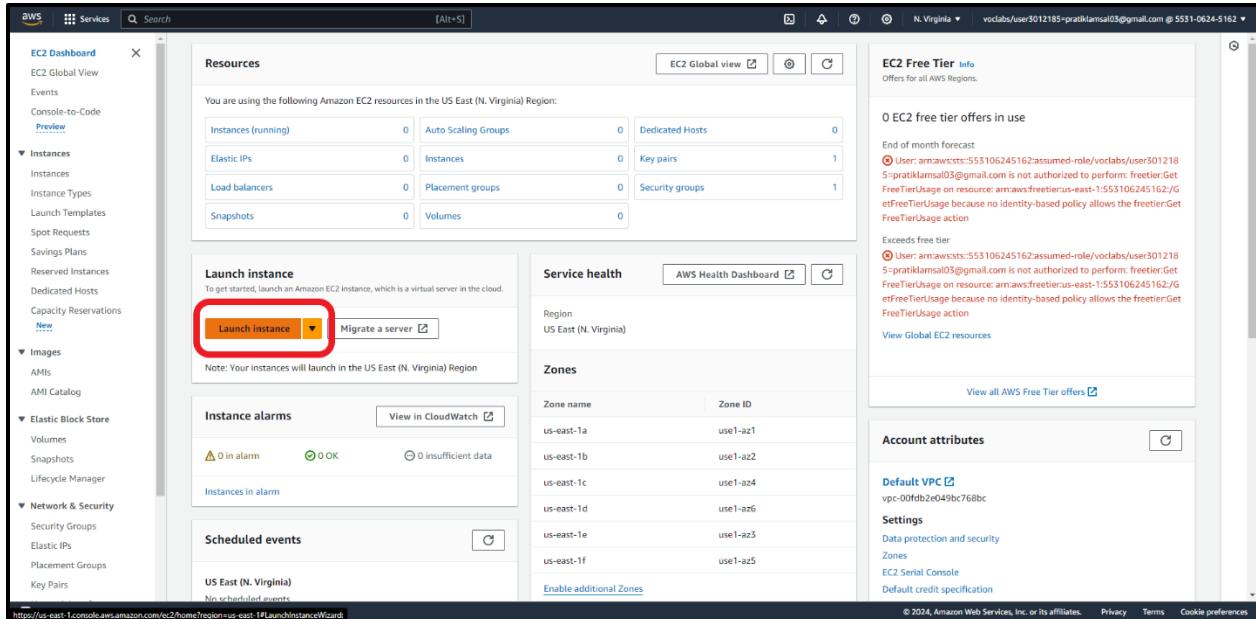


Figure 2 Launching an Instance

3. Instance Name and OS Image

Instance name is given, and an OS image is selected.

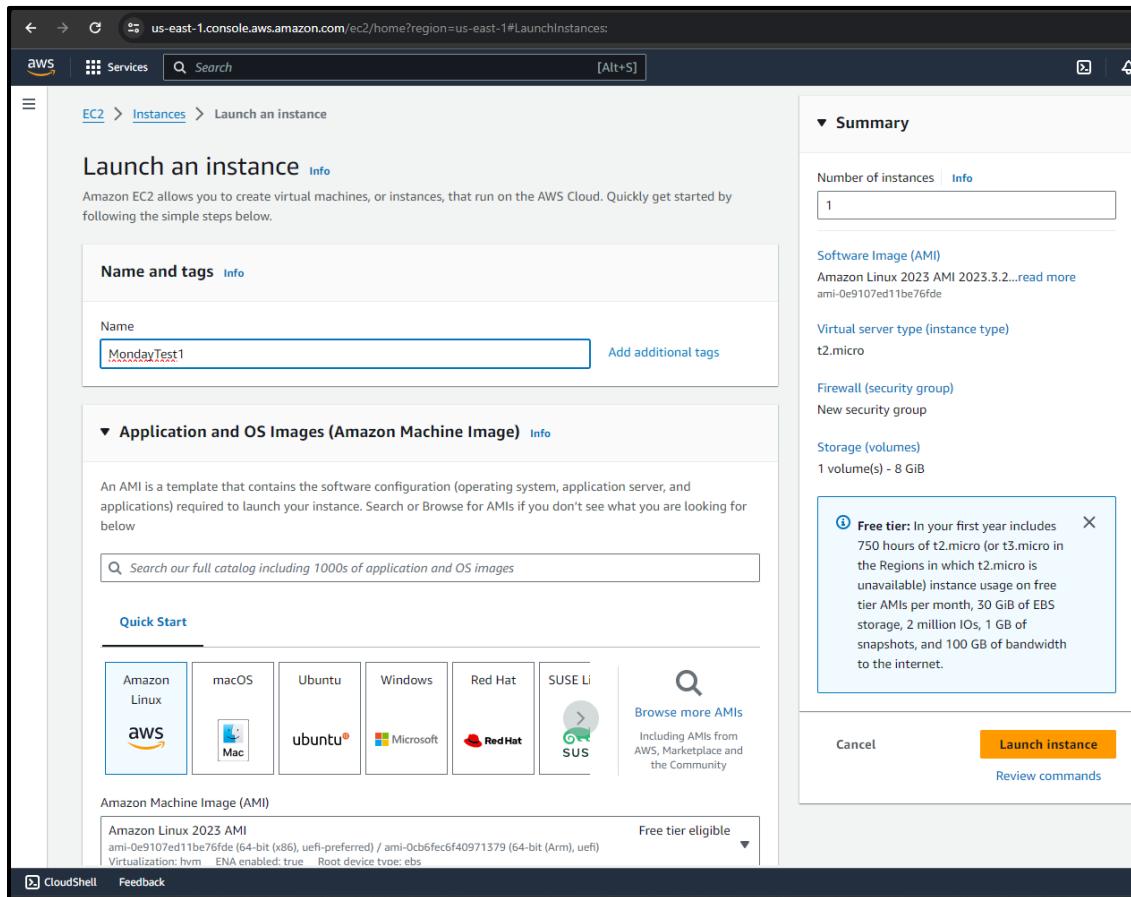


Figure 3 Instance Name and OS Image

4. Creating Key Pair

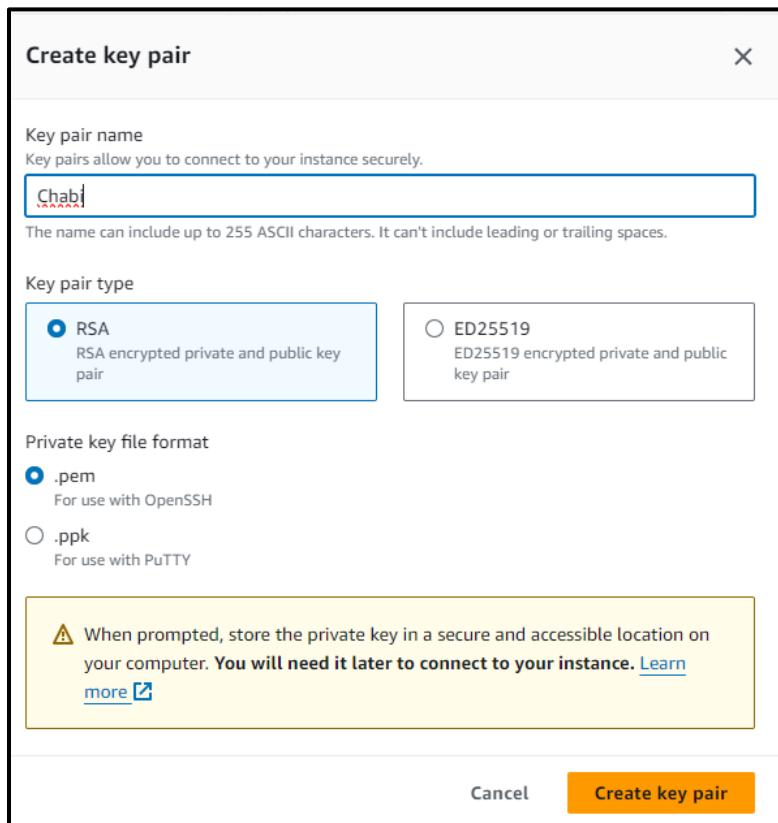


Figure 4 Key Pair

5. Key Pair Downloaded

.pem file is downloaded automatically.

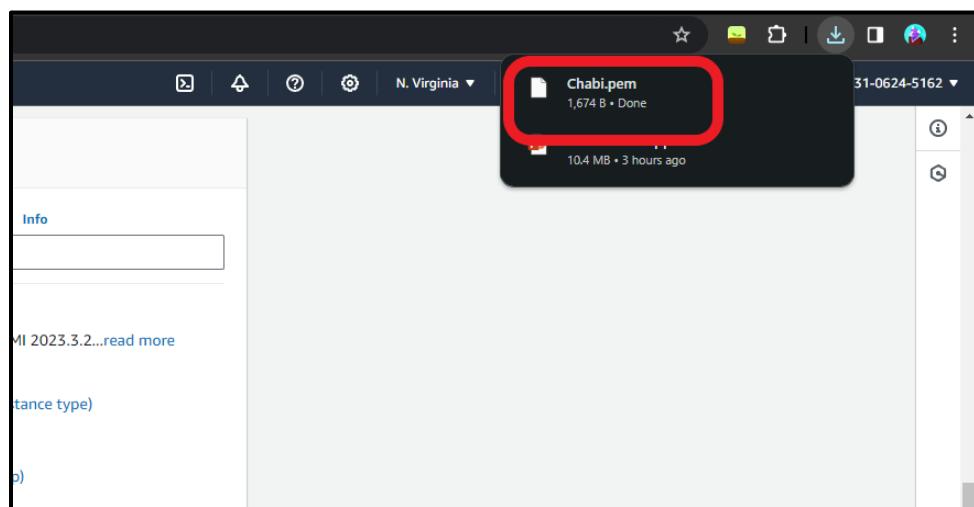


Figure 5 Automatic File Download

6. Network Settings

Security Group launch-wizard-1 is created allowing SSH, HTTPS and HTTP traffic from anywhere.

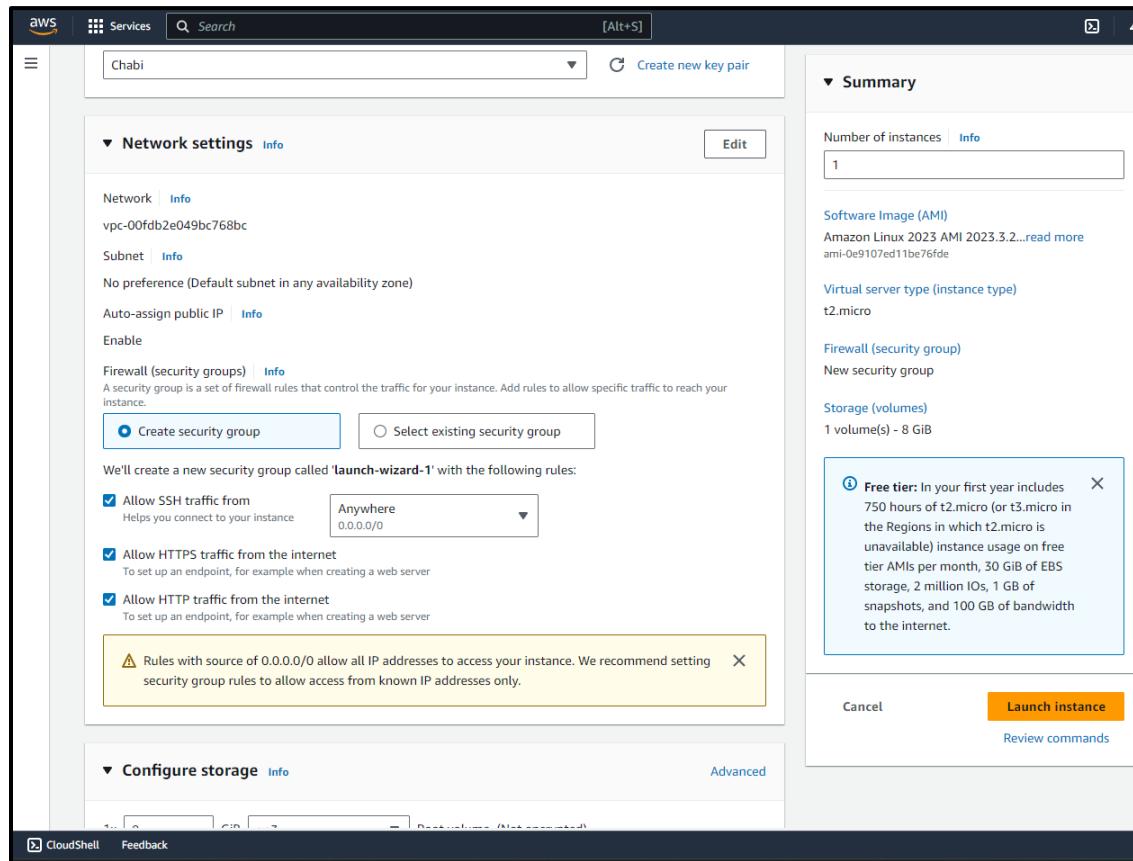


Figure 6 Network Settings

7. Number of Instance

Two instances with same configuration is to be created.

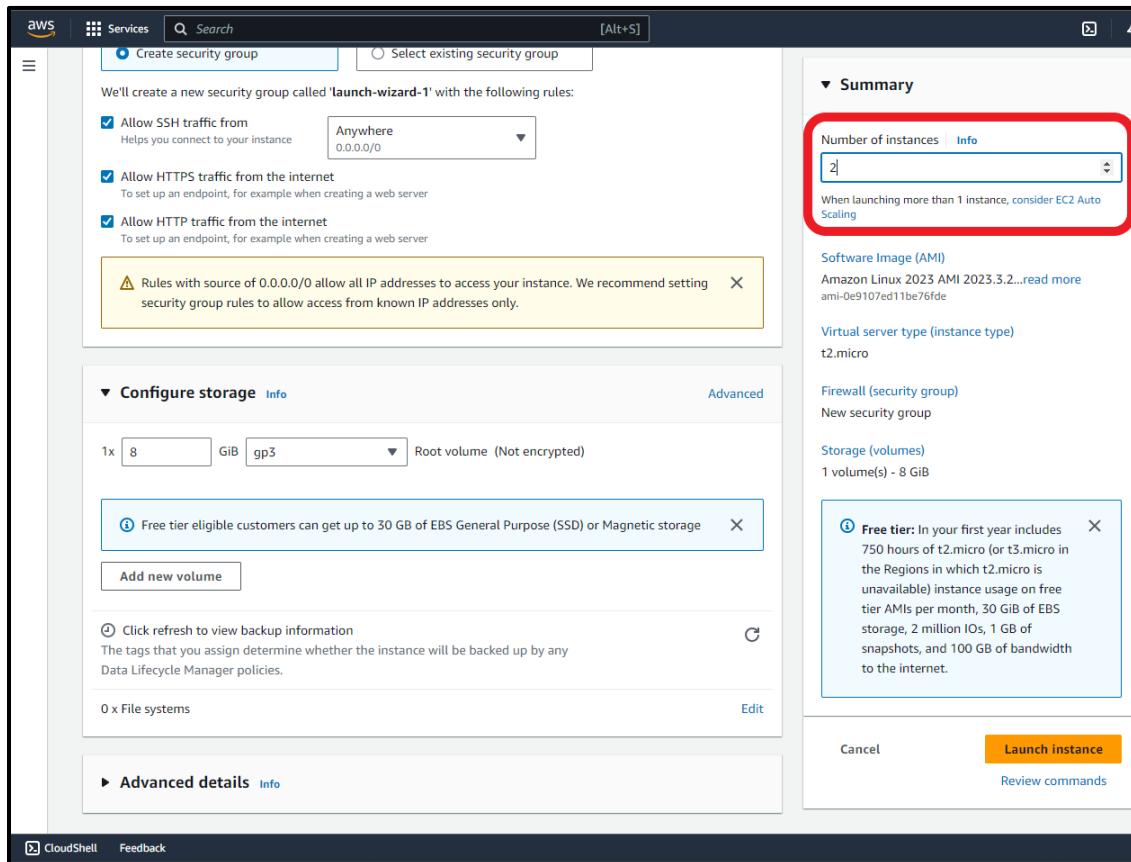


Figure 7 Number of Instances

8. Additional Data

Commands Inserted in User Data field.

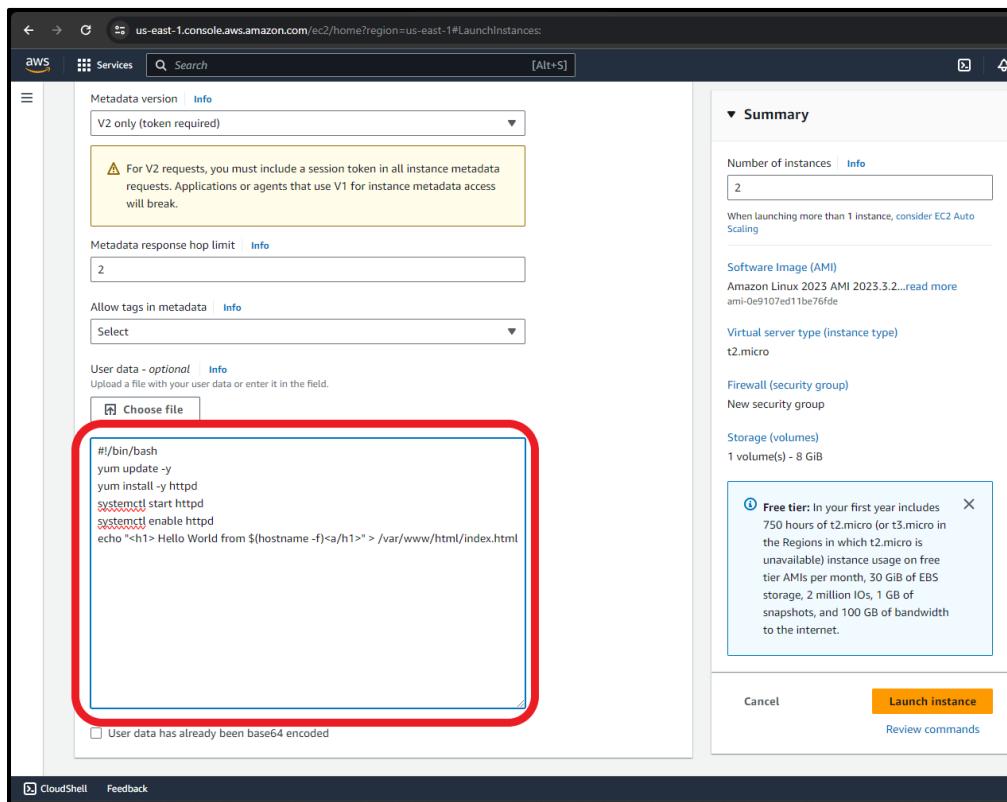
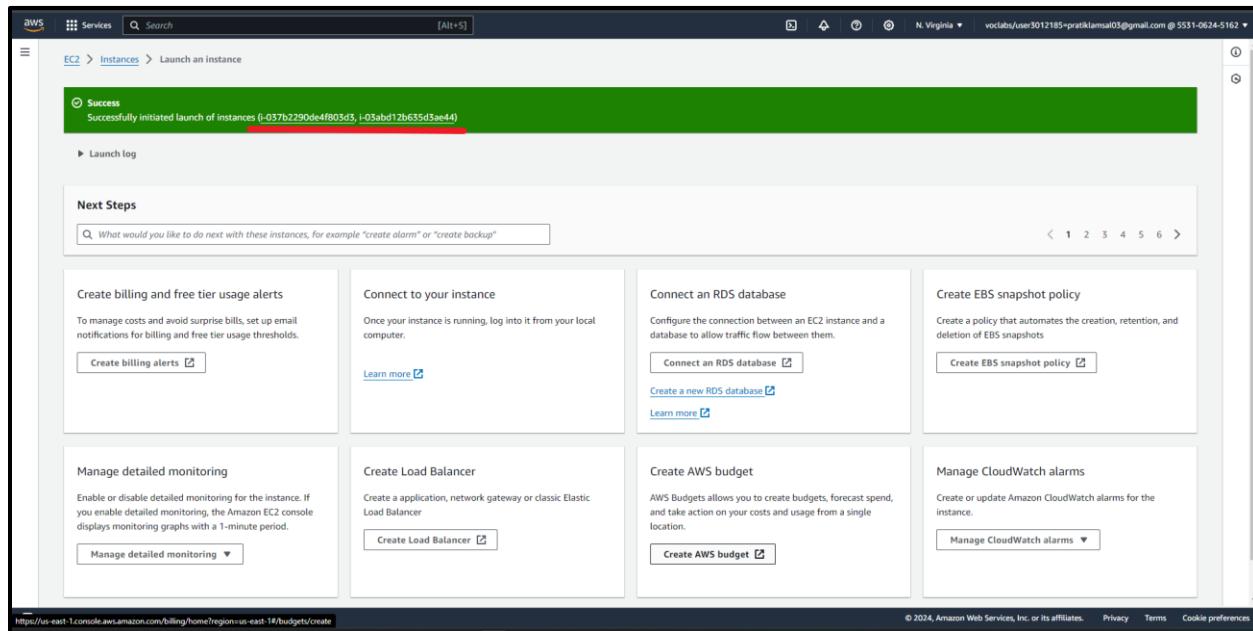


Figure 8 Additional Data

9. Successful Instance Creations

Two instances are successfully created.



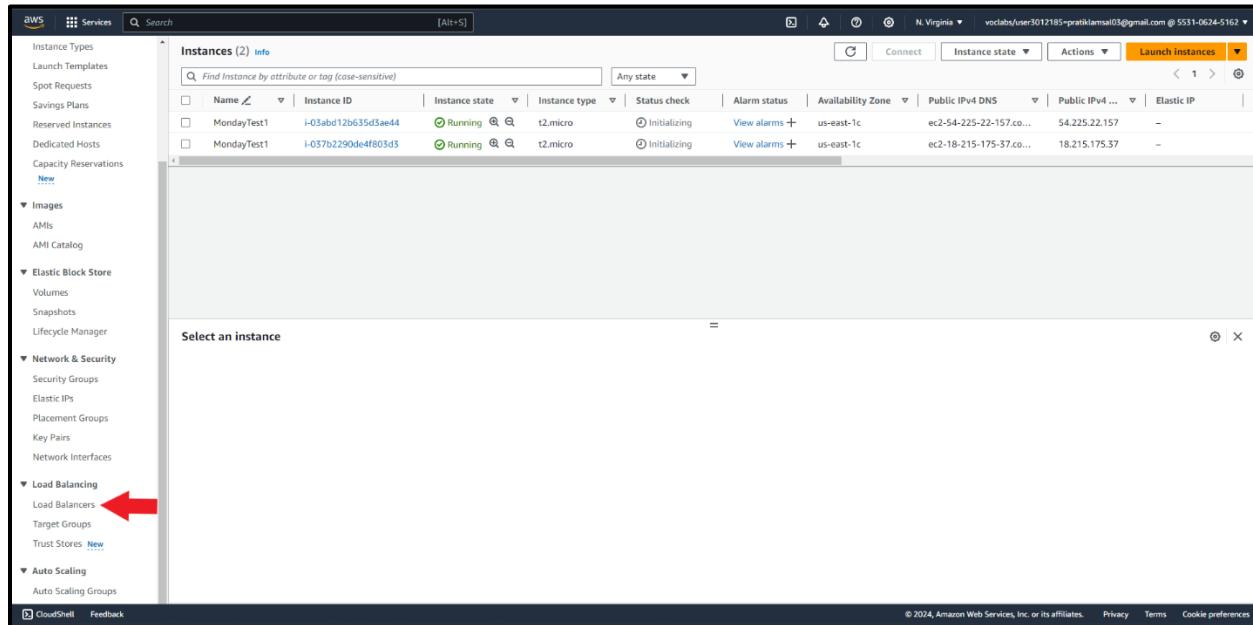
The screenshot shows the AWS EC2 Instances Launch log. At the top, a green success banner states: "Success Successfully initiated launch of instances (i-037b2290de4f803d3, i-03abd12b635d3ae4d)". Below the banner, there is a "Launch log" link. The main area is titled "Next Steps" with a search bar and a navigation menu (1-6). It contains eight cards:

- Create billing and free tier usage alerts**: To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds. Includes a "Create billing alerts" button.
- Connect to your instance**: Once your instance is running, log into it from your local computer. Includes a "Learn more" link.
- Connect an RDS database**: Configure the connection between an EC2 instance and a database to allow traffic flow between them. Includes a "Connect an RDS database" button and a "Learn more" link.
- Create EBS snapshot policy**: Create a policy that automates the creation, retention, and deletion of EBS snapshots. Includes a "Create EBS snapshot policy" button.
- Manage detailed monitoring**: Enable or disable detailed monitoring for the instance. If you enable detailed monitoring, the Amazon EC2 console displays monitoring graphs with a 1-minute period. Includes a "Manage detailed monitoring" dropdown.
- Create Load Balancer**: Create a application, network gateway or classic Elastic Load Balancer. Includes a "Create Load Balancer" button.
- Create AWS budget**: AWS Budgets allows you to create budgets, forecast spend, and take action on your costs and usage from a single location. Includes a "Create AWS budget" button.
- Manage CloudWatch alarms**: Create or update Amazon CloudWatch alarms for the instance. Includes a "Manage CloudWatch alarms" dropdown.

At the bottom of the page, the URL is https://us-east-1.console.aws.amazon.com/billing/home?region=us-east-1#/budgets/create, and the footer includes links for 2024, Privacy, Terms, and Cookie preferences.

Figure 9 Successful Instance Creation

10. Load Balancer



The screenshot shows the AWS Instances page. On the left, a sidebar menu lists various services: Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, Load Balancing (with a red arrow pointing to the "Load Balancers" link), Target Groups, Trust Stores, Auto Scaling, Auto Scaling Groups, CloudShell, and Feedback. The main content area shows a table of instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
MondayTest1	i-03abd12b635d3ae4d	Running	t2.micro	Initializing	View alarms	us-east-1c	ec2-54-225-22-157.co...	54.225.22.157	-
MondayTest1	i-037b2290de4f803d3	Running	t2.micro	Initializing	View alarms	us-east-1c	ec2-18-215-175-37.co...	18.215.175.37	-

Below the table, a "Select an instance" dropdown is open. The footer includes links for 2024, Privacy, Terms, and Cookie preferences.

Figure 10 Load Balancer

11. Creating a Load Balancer

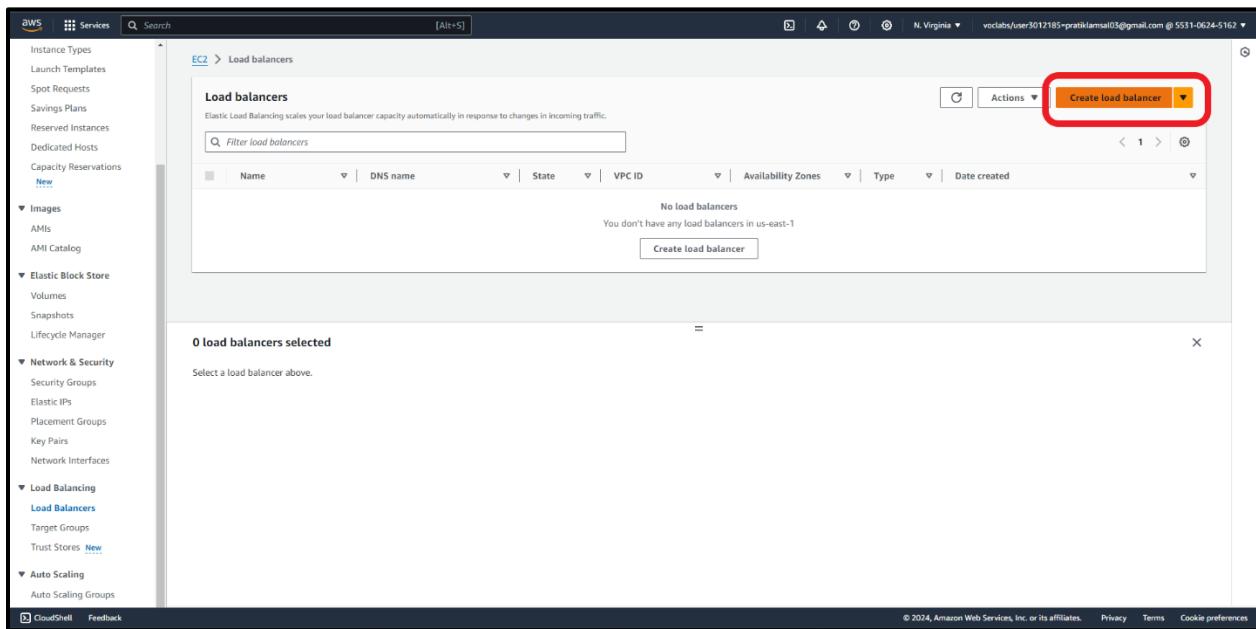


Figure 11 Creating a Load Balancer

12. Application Load Balancer

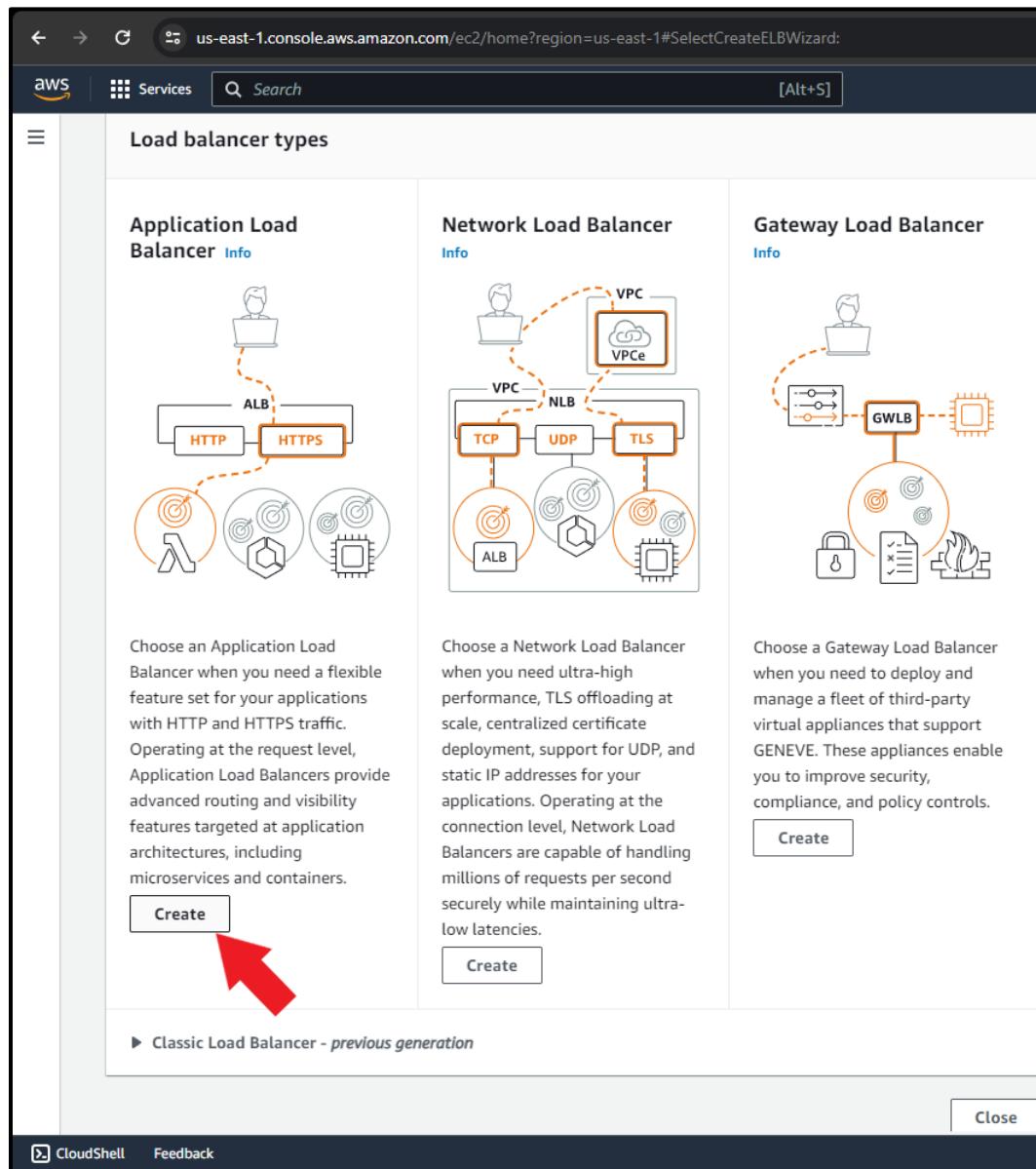


Figure 12 Types of Load Balancer

13. Configuring Application Load Balancer

The screenshot shows the AWS CloudFormation console with the URL us-east-1.console.aws.amazon.com/cloudformation/home?region=us-east-1#CreateStackWizard:. The page is titled "Create Stack" and displays the "Basic Information" step. The stack name is "MyFirstCloudFormationStack". The "Template" section shows a sample CloudFormation template. The "Outputs" section lists the output "MyFirstCloudFormationOutput" which is the ARN of the created CloudWatch Metrics Log Group. The "Next Step" button at the bottom right is labeled "Create Stack".

Figure 13 Application Load Balancer Configuration

14. Application Load Balancer Network Mappings

All options are selected.

The screenshot shows the AWS CloudFormation Network Mappings page. At the top, there is a search bar and a [Alt+S] key combination. Below the search bar, there is a list of network mappings:

- us-east-1b (use1-az2)**: Subnet subnet-0674902b400d47cd9
- us-east-1c (use1-az4)**: Subnet subnet-0c3d46b06e51332b4
- us-east-1d (use1-az6)**: Subnet subnet-0651913218bd408e8
- us-east-1e (use1-az3)**: Subnet subnet-01b090b8561812d30
- us-east-1f (use1-az5)**: Subnet subnet-0cbc74baf909f793a

Each mapping includes an "IPv4 address" and "Assigned by AWS" status indicator. At the bottom of the page, there are "CloudShell" and "Feedback" links.

Figure 14 Network Mappings

15. Application Load Balancer Security Groups

The screenshot shows the AWS Application Load Balancer Security Groups configuration interface. At the top, there's a navigation bar with 'Services' and a search bar. Below it, a sidebar lists 'Subnet' and 'IPv4 address' sections, with 'Assigned by AWS' noted under IPv4 address.

In the main content area, there's a 'Security groups' section with a dropdown menu showing 'Select up to 5 security groups'. Two security groups are selected: 'launch-wizard-1' (sg-090a96cc400638bd8) and 'default' (sg-078699ba5093c8f8f). A note says 'A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can create a new security group [Create security group].'

Below this is a 'Listeners and routing' section. It shows a 'Listener HTTP:80' configuration with 'Protocol' set to 'HTTP', 'Port' set to '80', and 'Default action' set to 'Forward to' (with a dropdown menu showing 'Select a target group'). There's also a link to 'Create target group [Create target group]'. A note says 'A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.'

At the bottom, there are buttons for 'Add listener tag' and links for 'CloudShell' and 'Feedback'.

Figure 15 Security Groups

16. Creating a Target Group

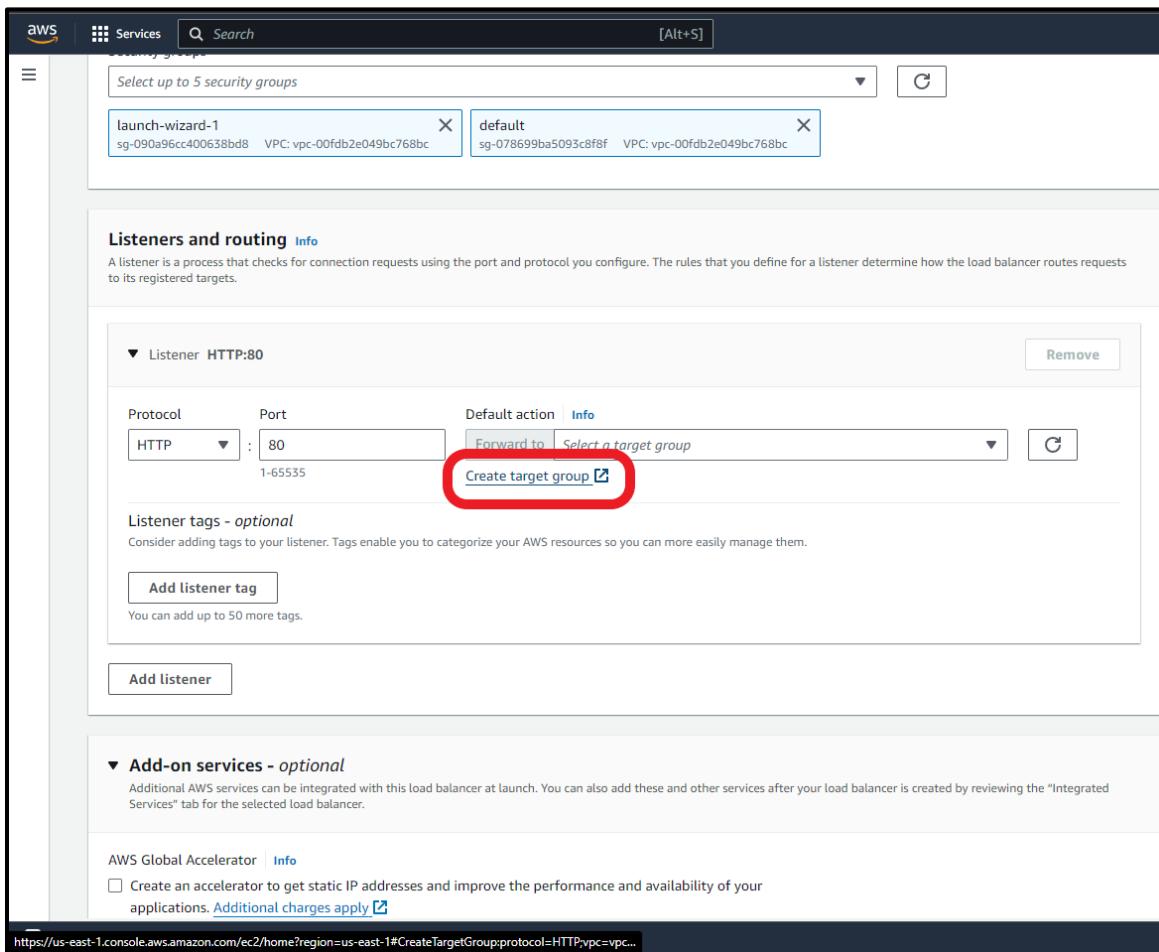


Figure 16 Target Group Creation

17. Group Details

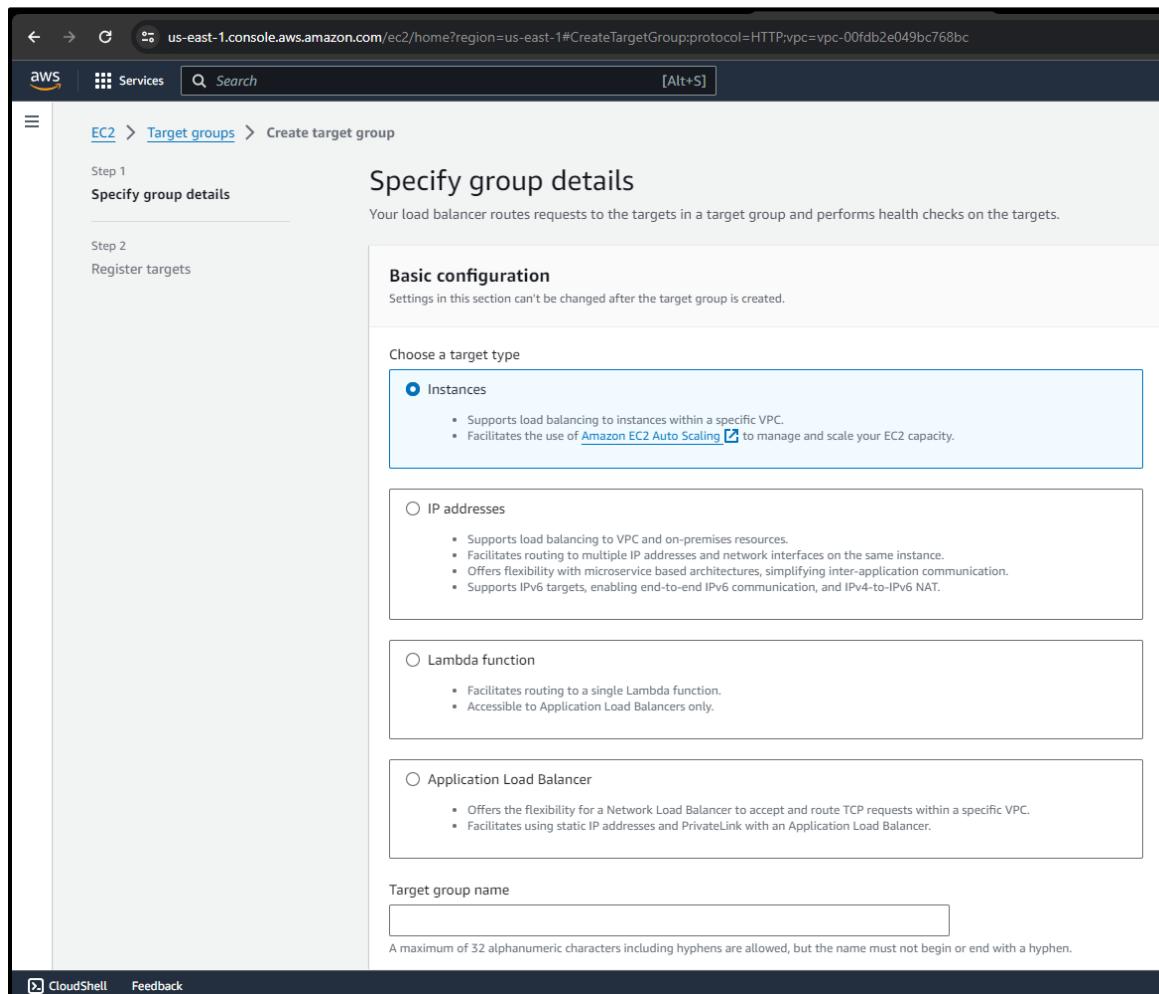


Figure 17 Group Details

18. Group Details 2

The screenshot shows the AWS CloudWatch Metrics Insights console. At the top, there is a search bar with the placeholder "Search" and a "Run" button. Below the search bar, there is a "Metrics" section with a table showing metrics from various services like Lambda, CloudWatch Metrics, and CloudWatch Metrics Insights.

The main area of the screen displays a query results table. The columns are "Time Range", "Metric Name", "Dimensions", "Value", and "Unit". One row is visible, showing a time range of "1 hour ago" to "Now", a metric name of "AWS/Lambda/Function Errors", dimensions of "FunctionName: myfunction", and a value of "1".

At the bottom of the screen, there is a "Run" button and a "Stop" button.

Figure 18 Group Details 2

19. Configuration Review and Target Group Creation

The screenshot shows the AWS EC2 Target Groups configuration screen. At the top, there is a table listing two instances: 'MondayTest1' (Instance ID: i-05abd12b635d3ae44) and another 'MondayTest1' (Instance ID: i-037b2290de4f803d3). Below the table, a section titled 'Ports for the selected instances' allows specifying ports for routing traffic. A dropdown menu shows '80' and 'T-65535 (separate multiple ports with commas)'. A button 'Include as pending below' is present. A note at the bottom says '2 selections are now pending below. Include more or register targets when ready.'

Review targets

Targets (2)

Remove	Health status	Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID
X	Pending	i-05abd12b635d3ae44	MondayTest1	80	Running	launch-wizard-1	us-east-1c	172.31.23.209	subnet-0c3d46b06e51332b4
X	Pending	i-037b2290de4f803d3	MondayTest1	80	Running	launch-wizard-1	us-east-1c	172.31.20.183	subnet-0c3d46b06e51332b4

2 pending

Cancel Previous Create target group

Figure 19 Target Group Creation

20. Successful Target Group Creation

The screenshot shows the AWS EC2 Target Groups page after successful creation. The left sidebar includes 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Console-to-Code', 'Instances' (selected), 'Images', 'Elastic Block Store', and 'Network & Security'. The main area displays 'Details' for a target group named 'MondayLB/38ffdd50da8cb9a1'. It shows 'Target type: Instance', 'Protocol: Port', 'Protocol version: HTTP1', and 'VPC: vpc-0f0db2e049bc768bc'. Below this, a table shows target status: 2 Total targets, 0 Healthy, 0 Unhealthy, 2 Unused, 0 Initial, and 0 Draining. A 'Distribution of targets by Availability Zone (AZ)' table is also present. At the bottom, a 'Registered targets (2)' table lists the two instances from Figure 19 as healthy.

Targets (2)

Instance ID	Name	Port	Zone	Health status	Health status details	Anomaly detection result
i-05abd12b635d3ae44	MondayTest1	80	us-east-1c	Unused	Target group is not co... Normal	Normal
i-037b2290de4f803d3	MondayTest1	80	us-east-1c	Unused	Target group is not co... Normal	Normal

Figure 20 Successful Target Group Creation

21. Target Group Selected

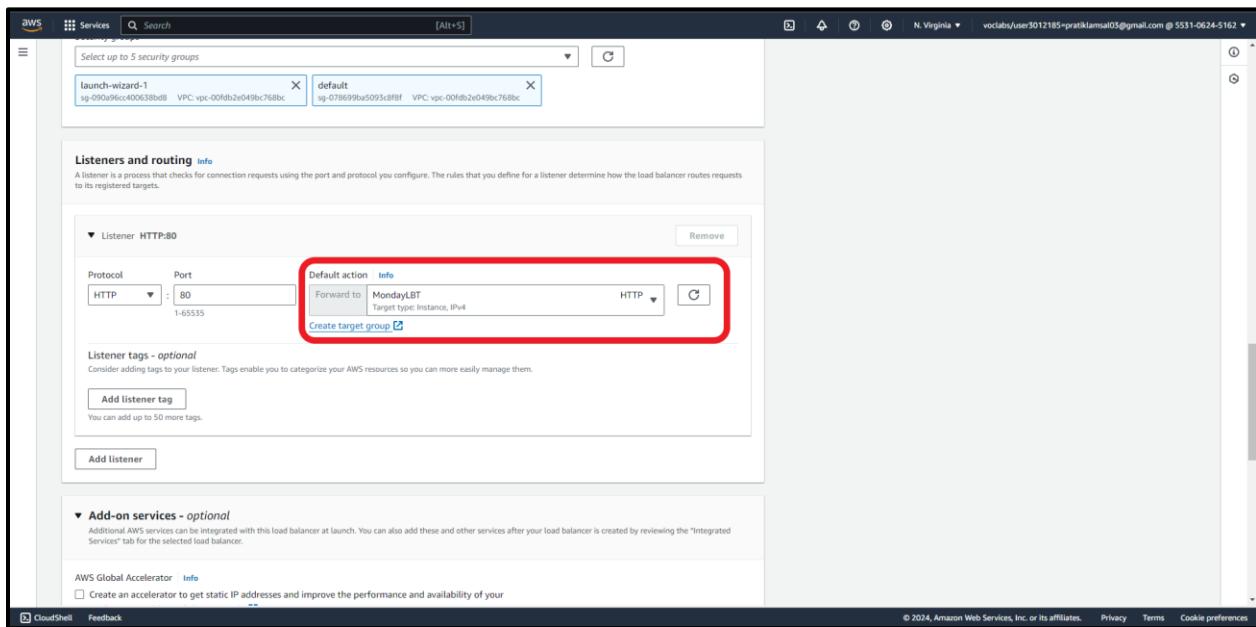


Figure 21 Target Group Selected

22. Summary and Load Balancer Creation

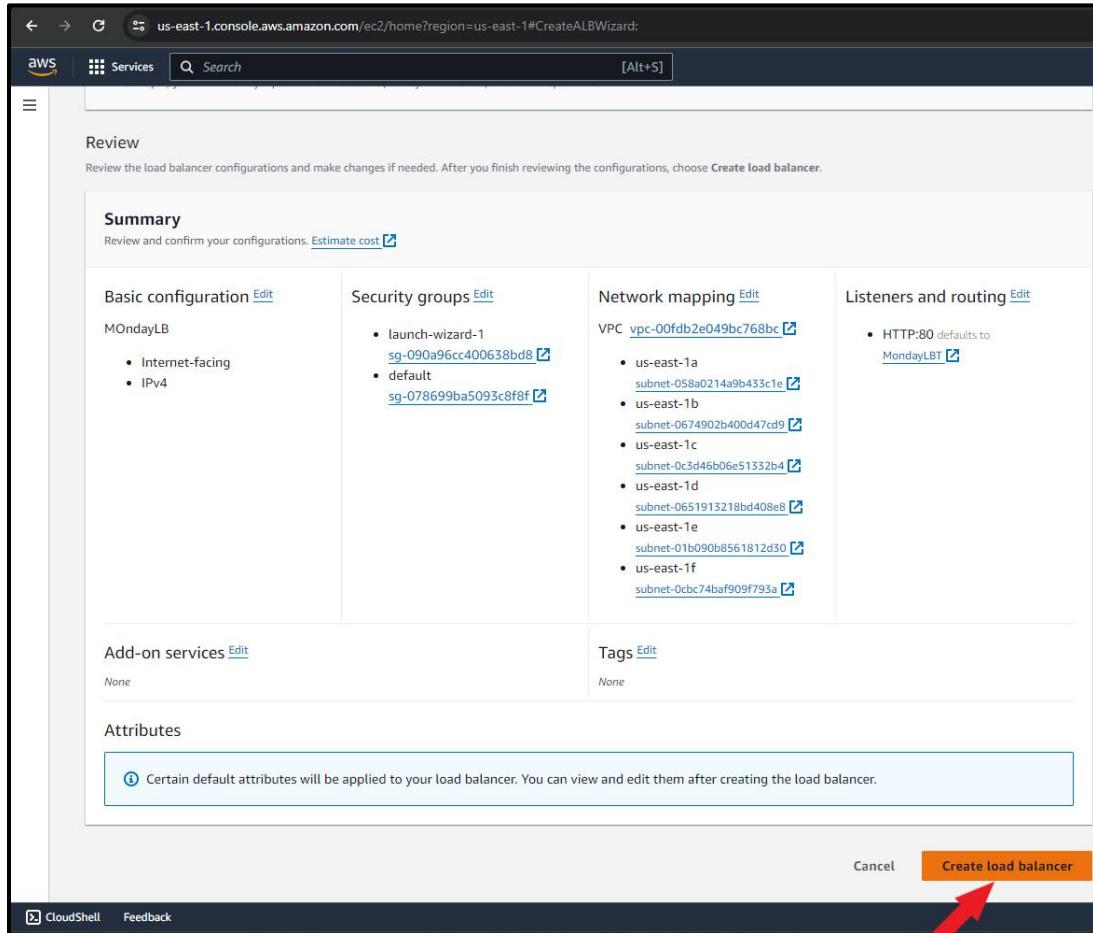


Figure 22 Summary and Load Balancer Creation

23. Successful Application Load Balancer Creation

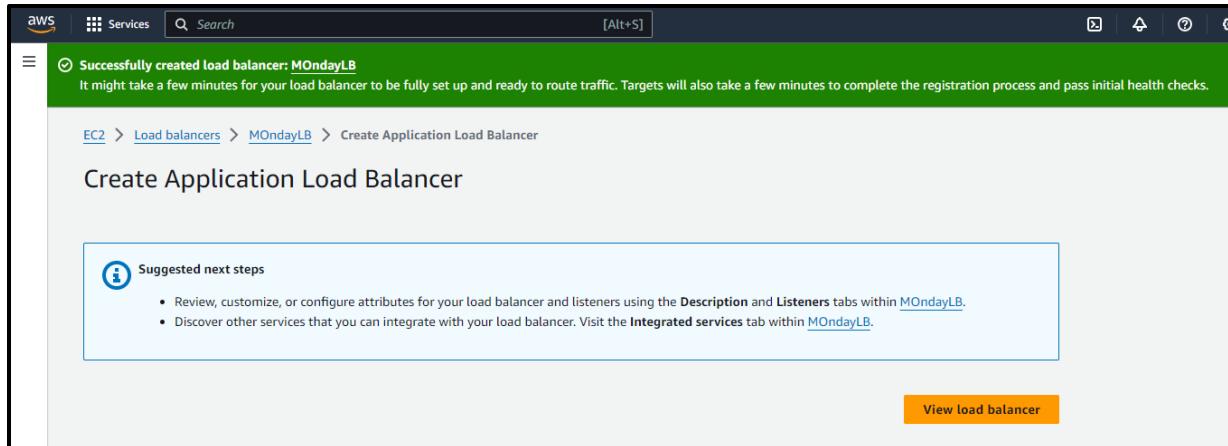
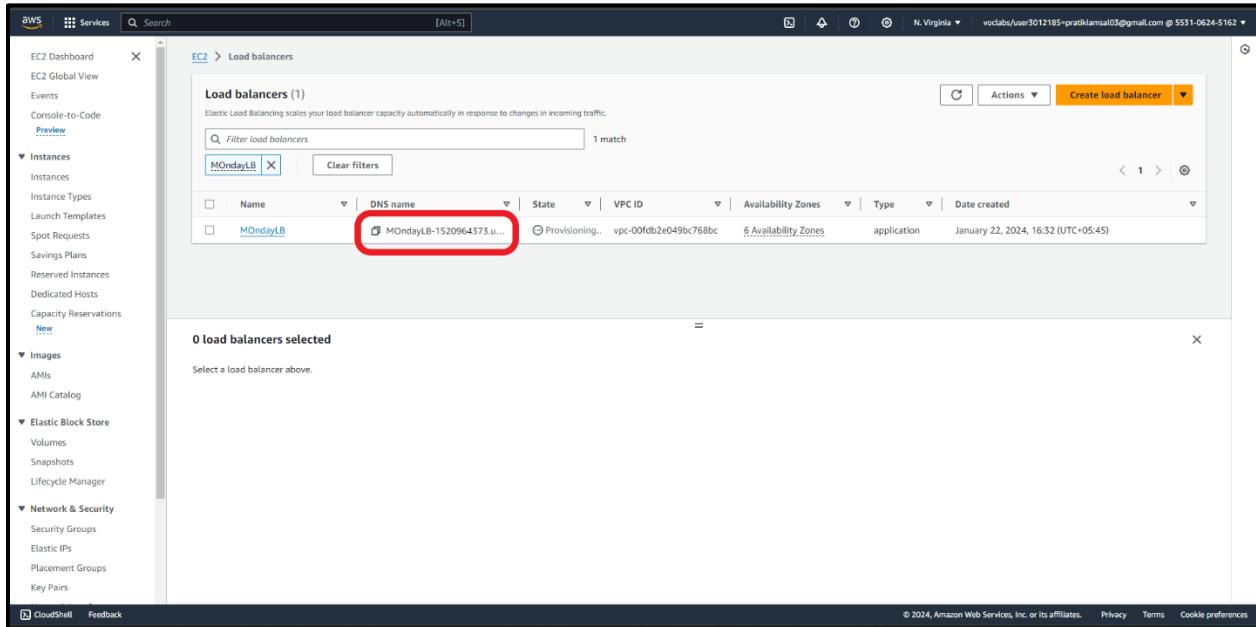


Figure 23 Successful Application Load Balancer Creation

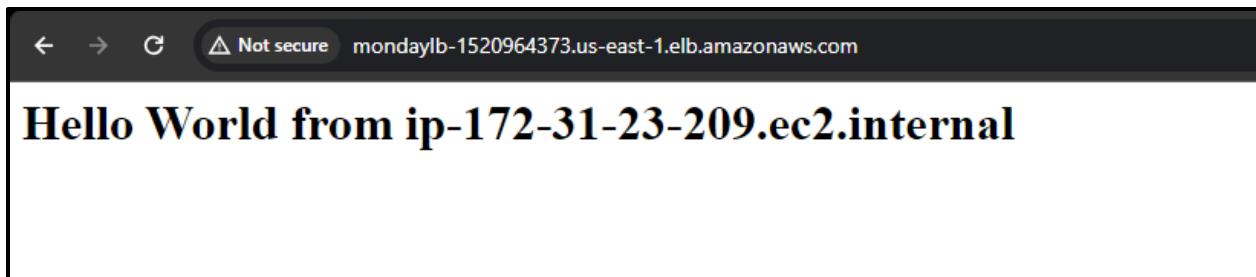
24. Copying the DNS name



The screenshot shows the AWS EC2 Load Balancers console. On the left, there's a sidebar with various EC2-related options like Instances, Images, and Network & Security. The main area is titled 'Load balancers (1)' and shows a table with one row. The row for 'MOndayLB' has its 'DNS name' field selected, which is highlighted with a red box. The table includes columns for Name, DNS name, State, VPC ID, Availability Zones, Type, and Date created.

Figure 24 Copying the DNS Name

25. IP from one of the instances



The screenshot shows a web browser window with the URL 'mondaylb-1520964373.us-east-1.elb.amazonaws.com'. The page content displays the text 'Hello World from ip-172-31-23-209.ec2.internal'.

Figure 25 One Instance IP

26. IP from the other instance



The screenshot shows a web browser window with the URL 'mondaylb-1520964373.us-east-1.elb.amazonaws.com'. The page content displays the text 'Hello World from ip-172-31-20-183.ec2.internal'.

Figure 26 Other Instance IP

27. Creating an Auto Scaling Group

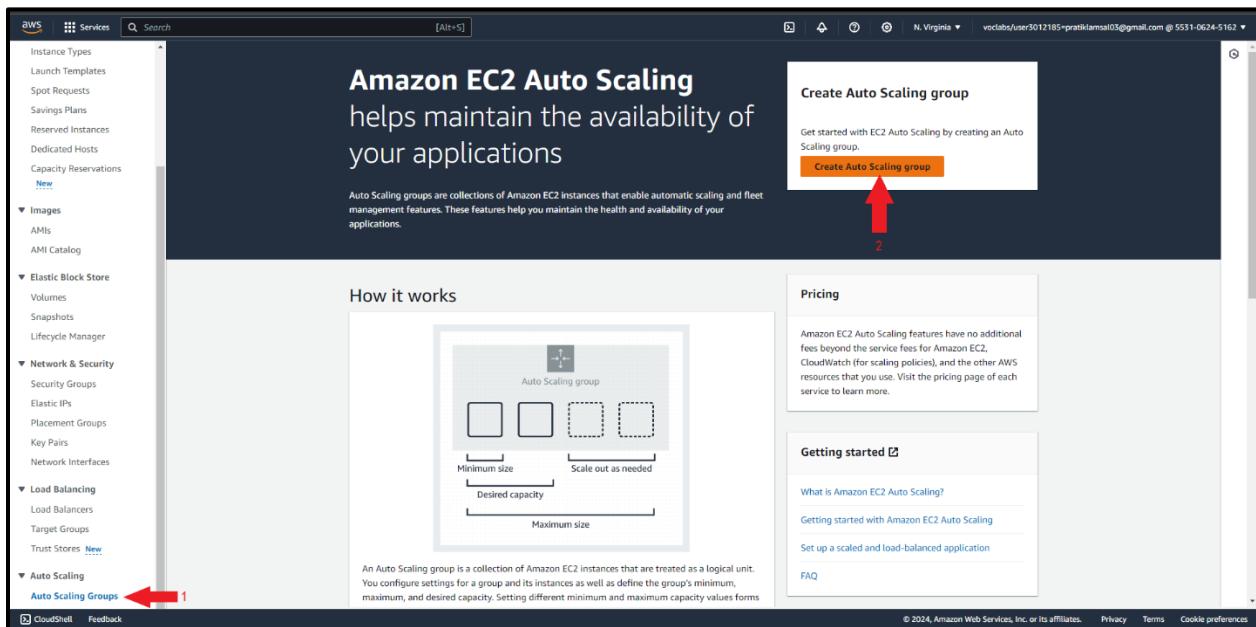


Figure 27 Auto Scaling Group

28. Creating a Launch Template

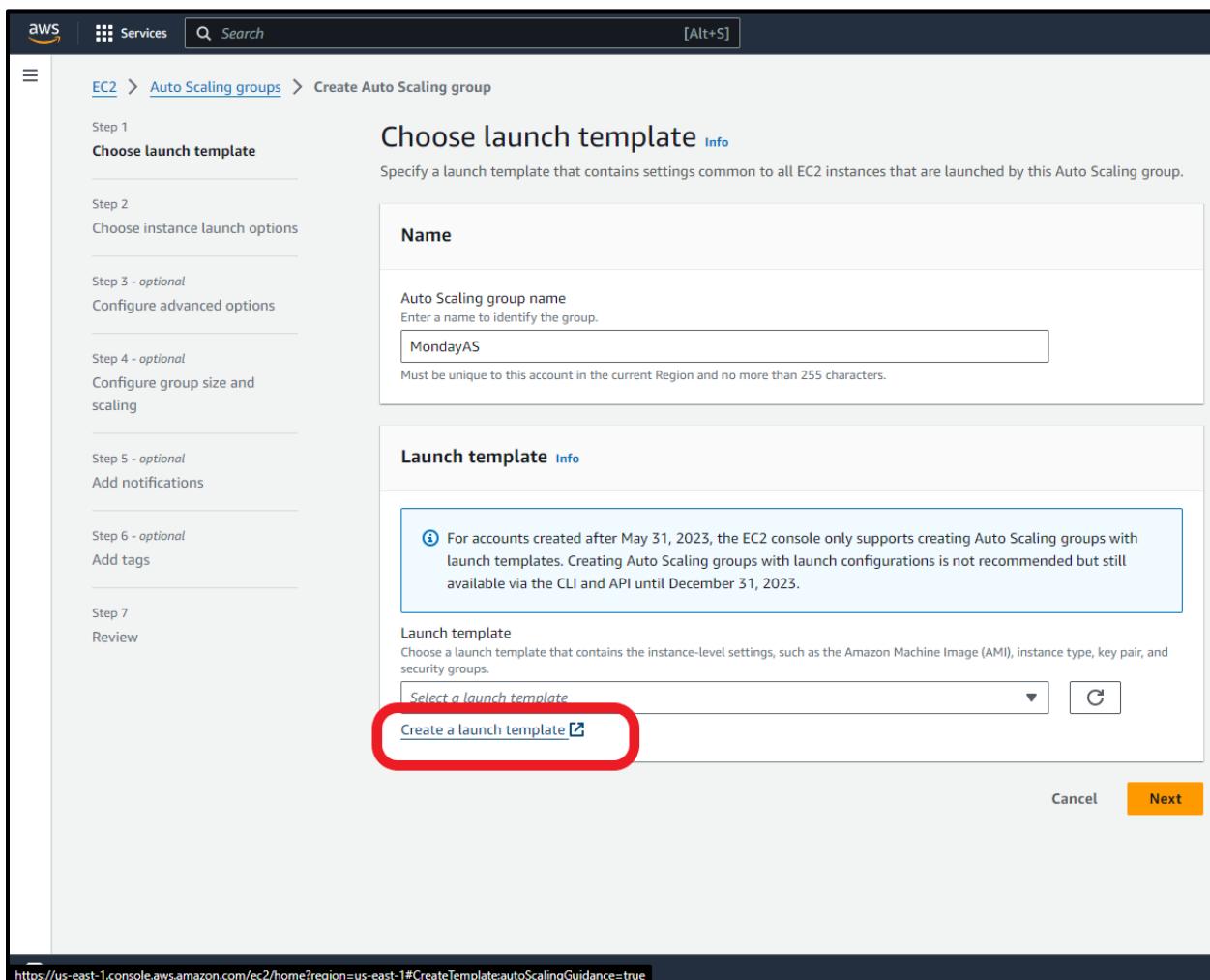


Figure 28 Creating a Launch Template

29. Application and OS Image

AMI in current use is selected.

The screenshot shows the AWS Launch Template configuration interface. The top navigation bar includes the AWS logo, Services, a search bar, and a keyboard shortcut [Alt+S]. The main content area is titled "Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template." A sub-section titled "Application and OS Images (Amazon Machine Image) - required" is expanded, showing a description of what an AMI is and a search bar for finding more AMIs. Below this, a "Recently used" section highlights the "Currently in use" AMI: "al2023-ami-2023.3.20240117.0-kernel-6.1-x86_64". This AMI is described as "Amazon Linux 2023 AMI 2023.3.20240117.0 x86_64 HVM kernel-6.1". It has the following details: Architecture: x86_64, AMI ID: ami-0e9107ed11be76fde, Creation Date: 2024-01-17T21:43:12.000Z, Virtualization: hvm, ENA enabled: true, Root device type: ebs, and Boot mode: uefi-preferred. A "Verified provider" badge is present. To the right of the AMI details is a summary panel with sections for Software Image (AMI), Virtual server type (instance type), Firewall (security group), Storage (volumes), and a Free tier information box. The Free tier box states: "Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet." At the bottom right of the summary panel is a "Create launch template" button. The bottom of the screen shows standard AWS navigation links: CloudShell, Feedback, and a dark footer bar.

Figure 29 Application and OS Image in Use

30. Selecting Key Pair

Key Pair created previously is selected.

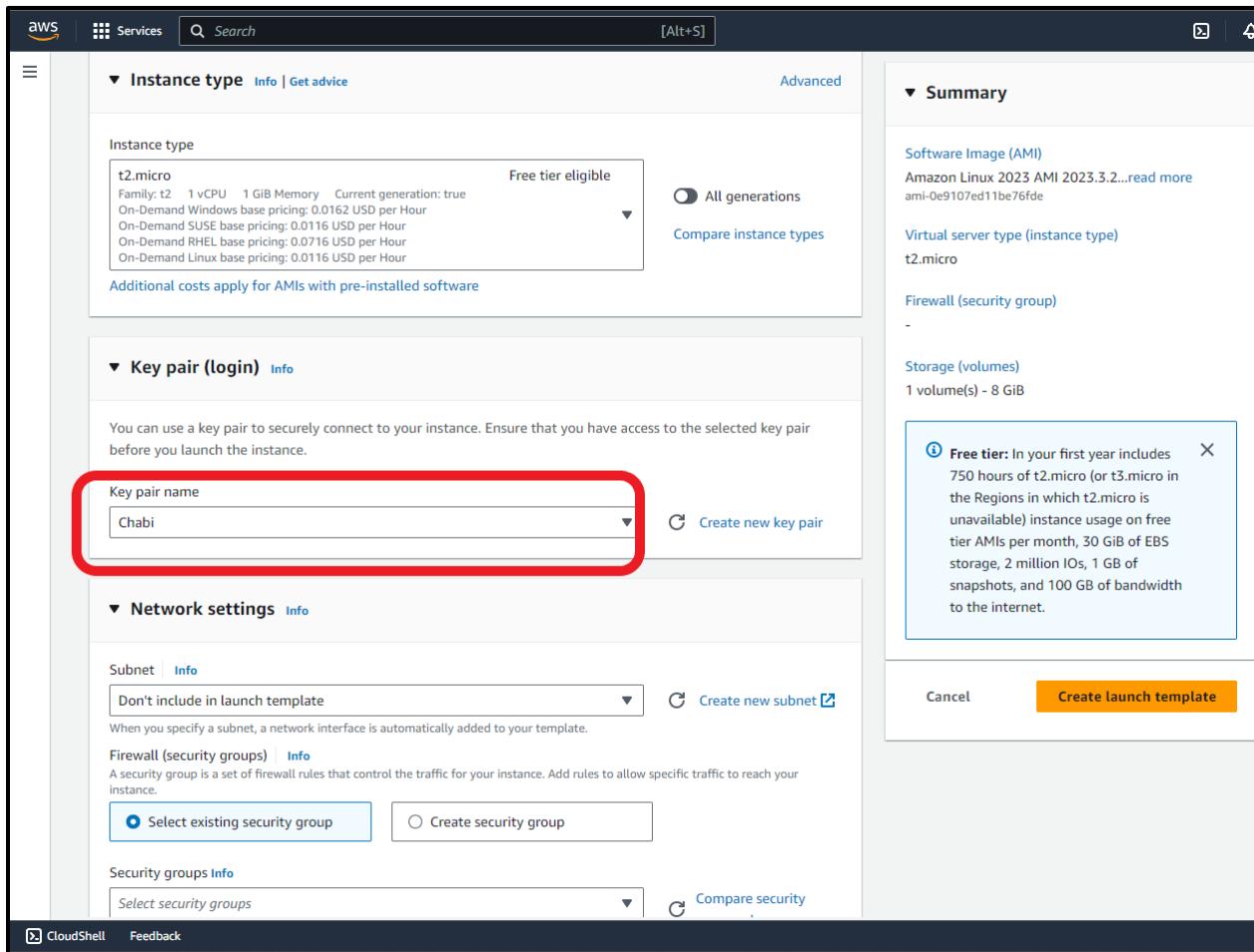


Figure 30 Key Pair Selection

31. Network Settings

Existing Security Groups are selected.

The screenshot shows the AWS Network Settings configuration page. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, a search bar, and keyboard shortcut '[Alt+S]'. On the left, a sidebar has a 'Network settings' section expanded. Under 'Subnet', it says 'Don't include in launch template' and 'Create new subnet'. Below that, under 'Firewall (security groups)', there are two options: 'Select existing security group' (which is selected) and 'Create security group'. A 'Security groups info' section shows two selected security groups: 'launch-wizard-1' and 'default'. There's also a 'Compare security group rules' button. Under 'Storage (volumes)', it shows 'EBS Volumes' with 'Volume 1 (AMI Root)' (8 GiB, EBS, General purpose SSD (gp3)). A note states that AMI Volumes are not included in the template unless modified. A callout box for the 'Free tier' explains that it includes 750 hours of t2.micro usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth. At the bottom right, there are 'Cancel' and 'Create launch template' buttons.

Figure 31 Network Settings

32. Configured a Launch Template

Commands are entered and Launch Template is Created as per the previous instance configurations.

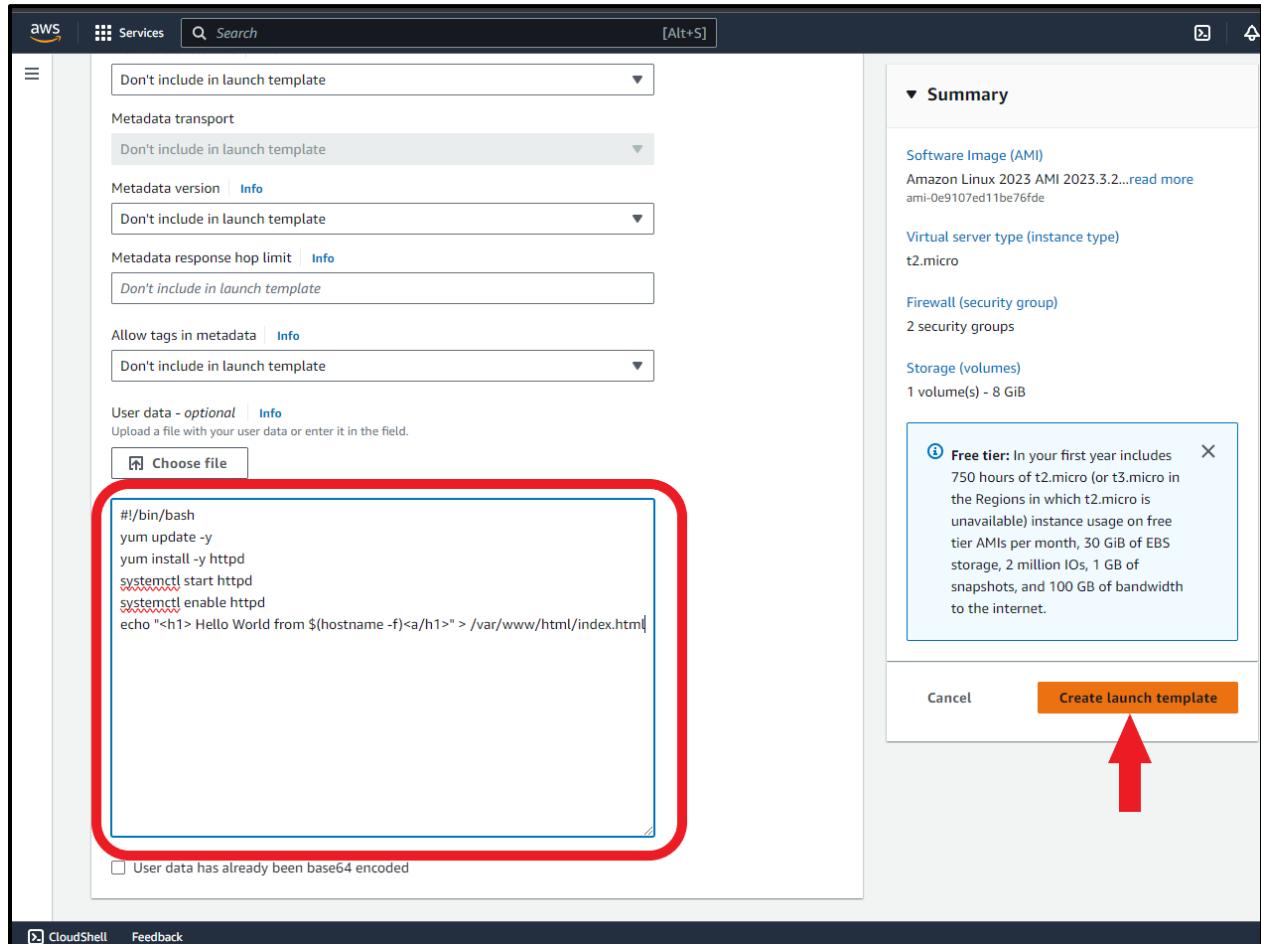


Figure 32 Launch Template Configured

33. Launch Template Created

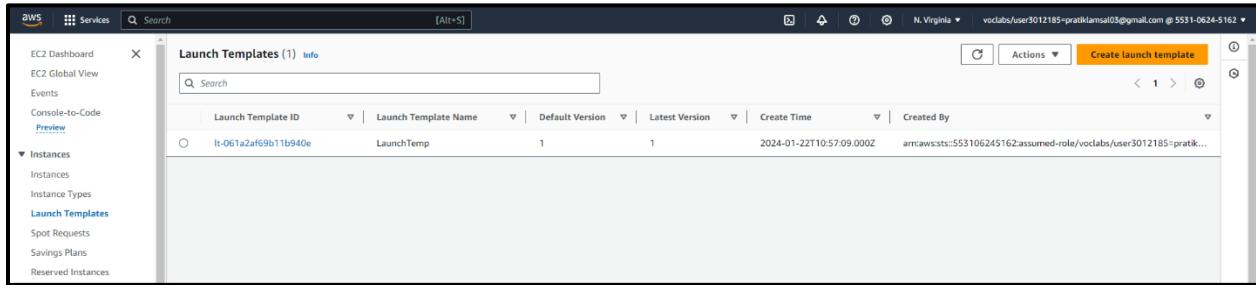


Figure 33 Launch Template Created

34. Launch Template Selected

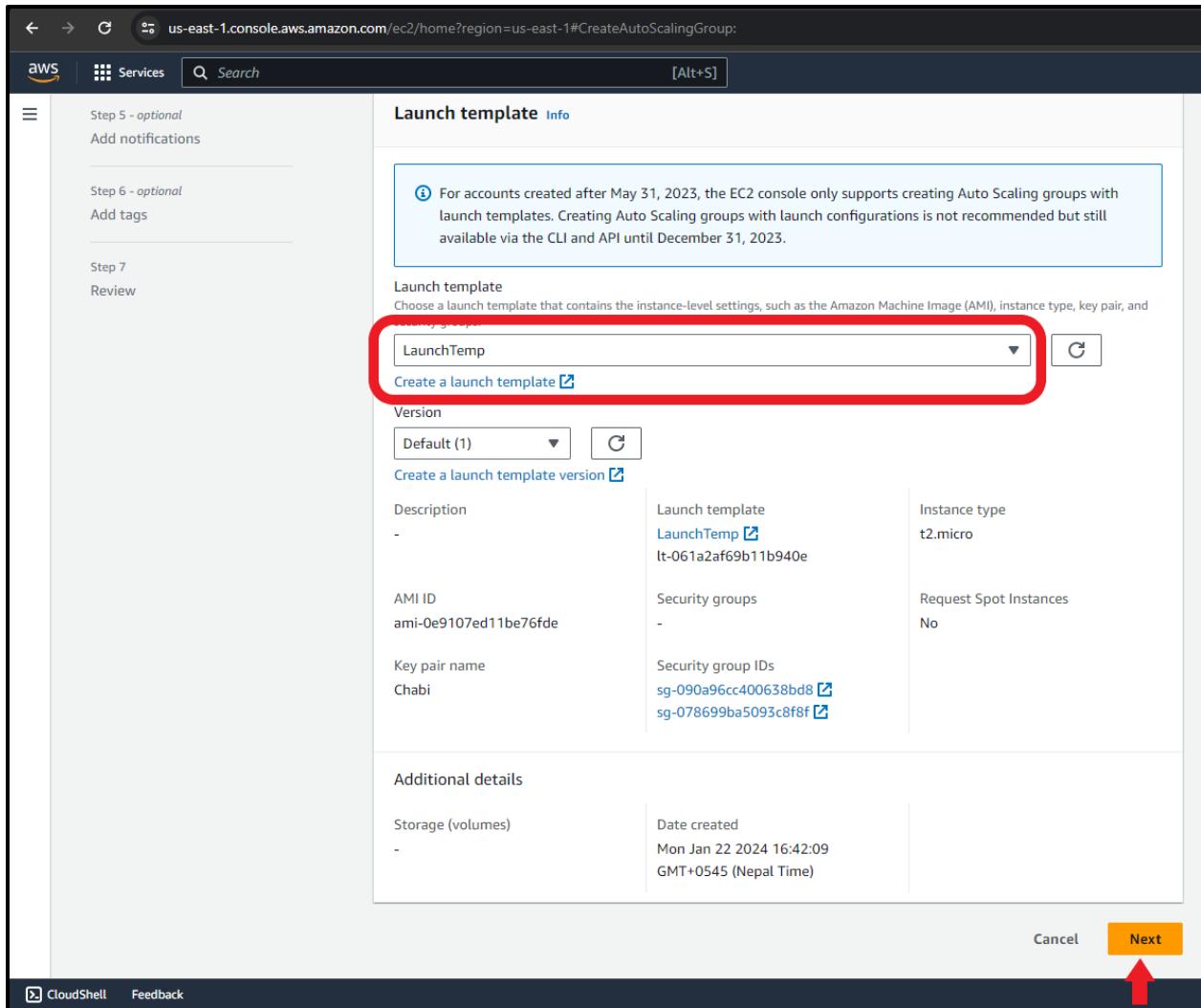


Figure 34 Selecting Launch Template

35. Network Settings

VPC selected along with all Availability Zones and subnets

The screenshot shows the AWS EC2 Auto Scaling Group creation wizard at Step 7: Review. The instance type is set to t2.micro. The VPC dropdown is set to 'vpc-00fdb2e049bc768bc' (172.31.0.0/16, Default). Under 'Availability Zones and subnets', several subnets from the 'us-east-1' region are listed: us-east-1a, us-east-1b, us-east-1c, us-east-1d, us-east-1e, and us-east-1f. Each entry shows the subnet ID, name, and CIDR range (e.g., 172.31.0.0/20, Default). Buttons for 'Create a VPC' and 'Create a subnet' are visible. At the bottom, there are buttons for 'Cancel', 'Skip to review', 'Previous', and 'Next'.

Figure 35 Network Settings

36. Configuring Advanced Options

It is attached to an existing Load Balancer created above.

The screenshot shows the AWS EC2 console interface for creating an Auto Scaling group. The left sidebar lists steps: Step 1 (Choose launch template), Step 2 (Choose instance launch options), Step 3 (Configure advanced options - currently selected), Step 4 (Configure group size and scaling), Step 5 (Add notifications), Step 6 (Add tags), and Step 7 (Review). The main content area is titled "Configure advanced options - optional". It includes a section for "Load balancing" with three options: "No load balancer" (selected), "Attach to an existing load balancer" (selected), and "Attach to a new load balancer". Below this is a "Attach to an existing load balancer" section where "Choose from your load balancer target groups" is selected. A dropdown menu shows "MondayLBT | HTTP" as a selected target group. At the bottom is a "VPC Lattice integration options" section.

Figure 36 Advanced Options Configuration

37. Group Size and Scaling Configuration

The screenshot shows the AWS Auto Scaling Group Size and Scaling Configuration wizard, Step 2: Configure group size and scaling. The left sidebar lists steps: Choose launch template, 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, Step 7: Review.

Configure group size and scaling - optional

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
2

Max desired capacity
4

Equal or less than desired capacity

Automatic scaling - optional

Choose whether to use a target tracking policy Info
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.

CloudShell Feedback

Figure 37 Group Size and Scaling Configuration

38. No Instance Maintenance Policy Selected

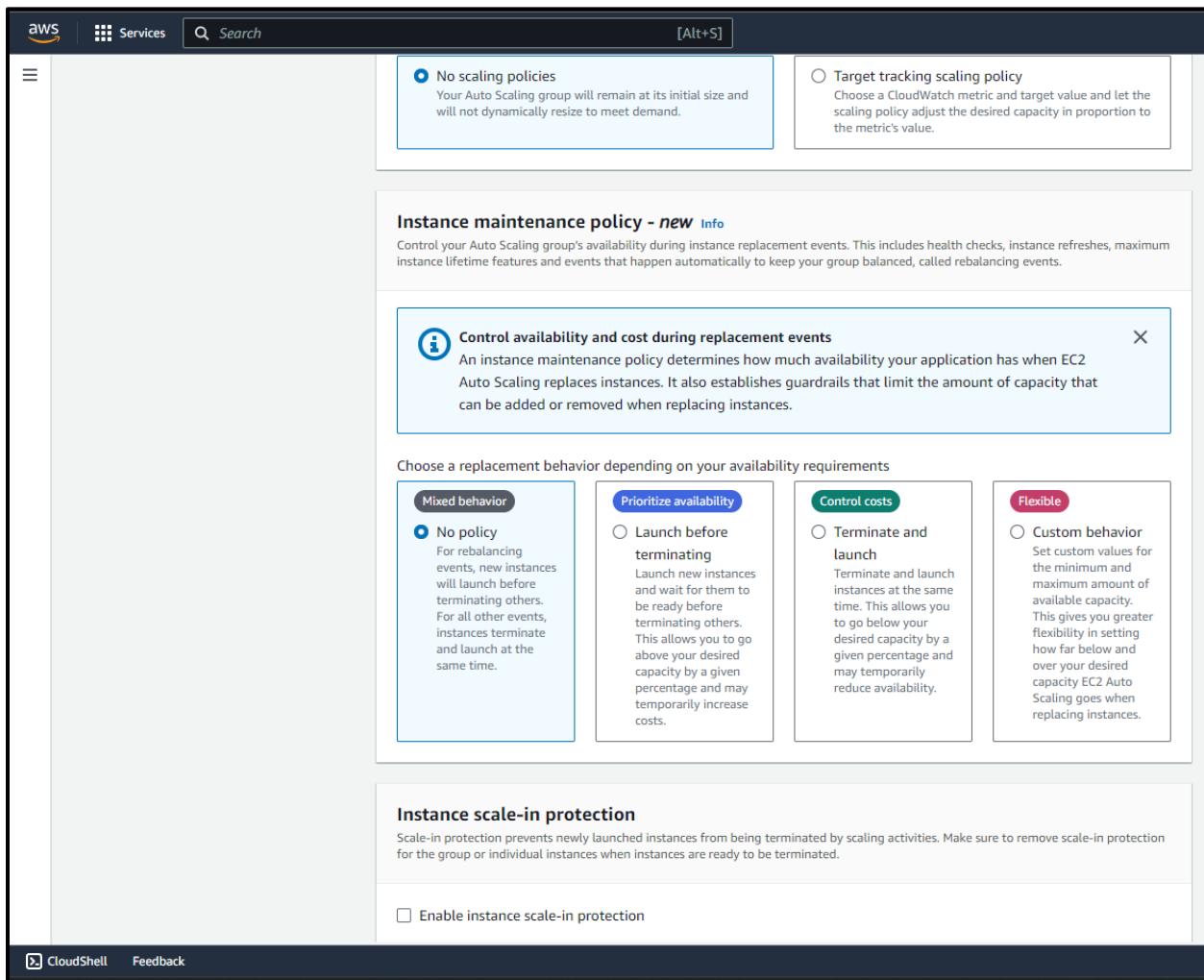


Figure 38 Instance Maintenance Policy

39. No notifications added.

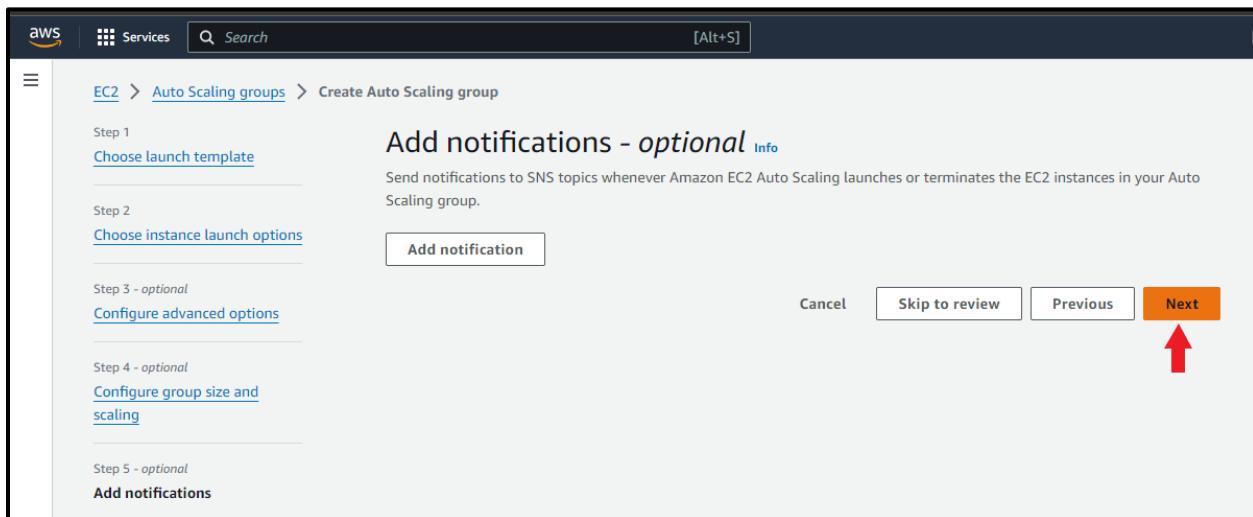


Figure 39 Adding Notifications

40. No tags added.

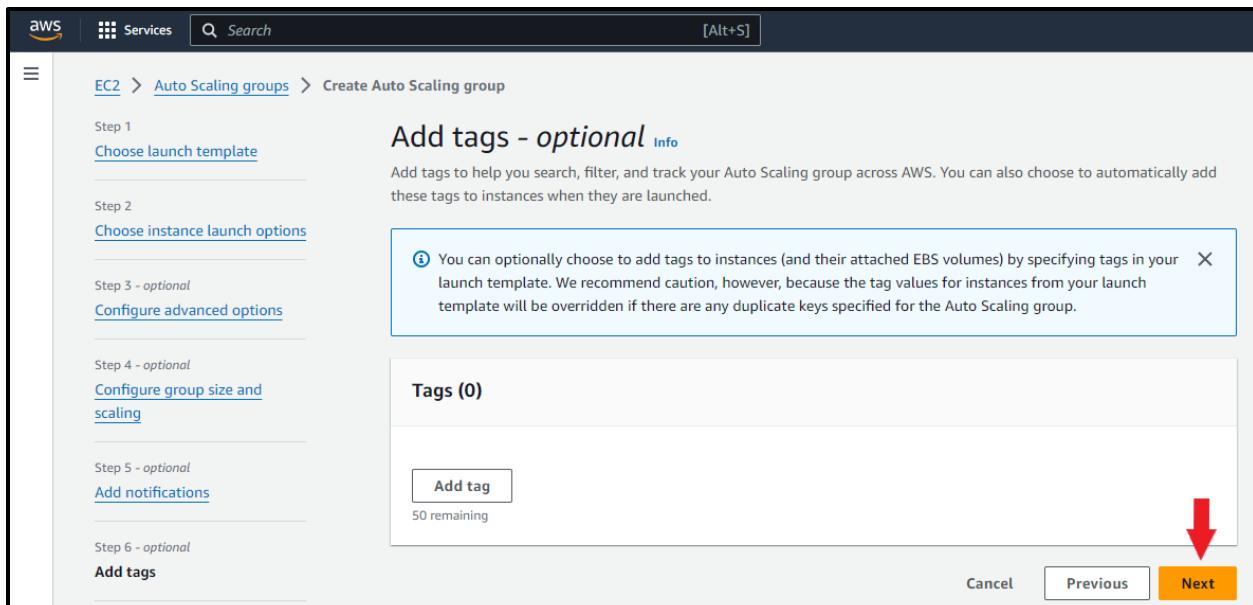


Figure 40 Adding Tags

41. Auto-Scaling Group Configured

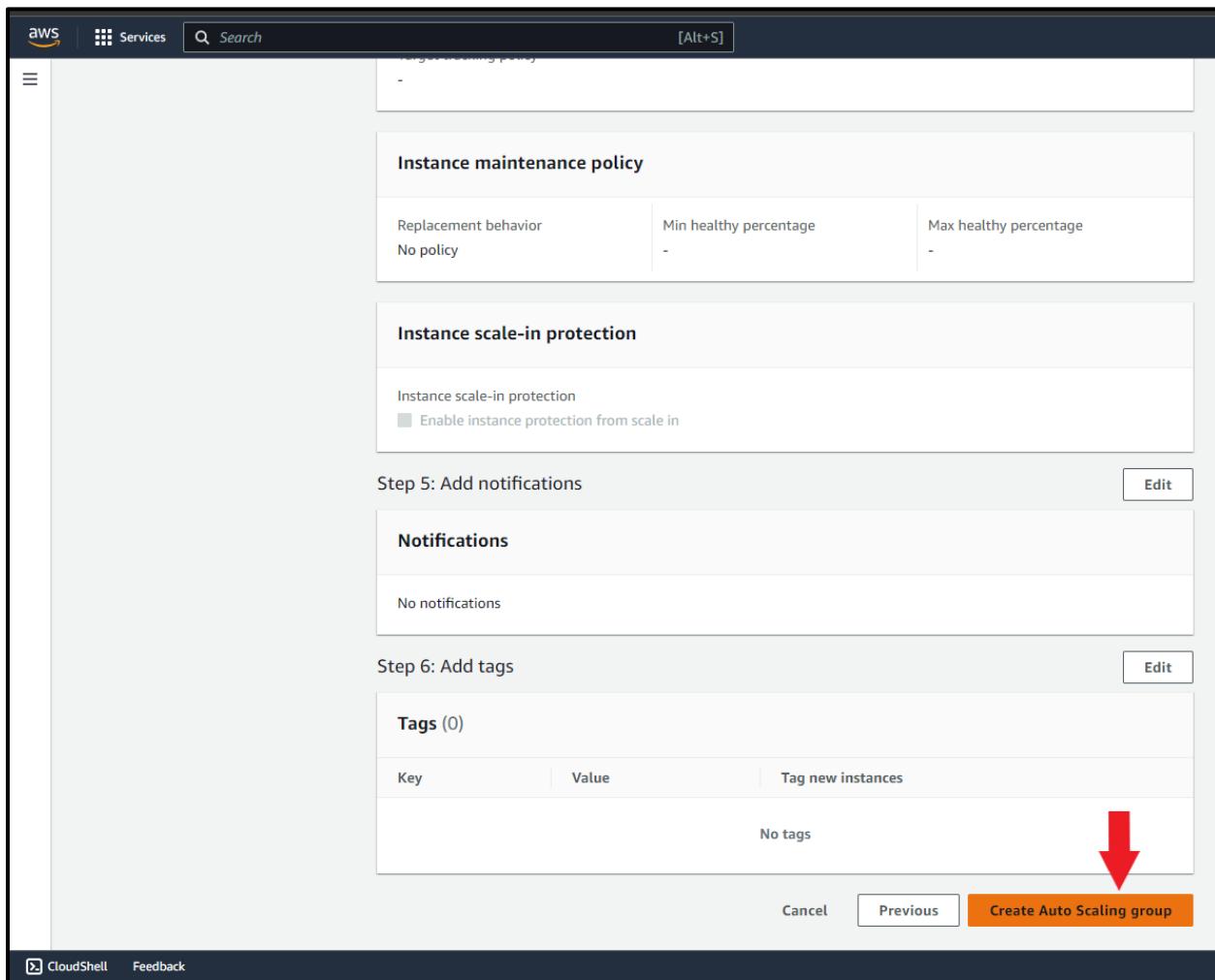


Figure 41 Auto-Scaling Group Configured

42. Two Instances in Pending State Automatically

The screenshot shows the AWS EC2 Instances page. The left sidebar includes options like EC2 Dashboard, EC2 Global View, Events, Console-to-Code, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, and Key Pairs. The main content area displays a table of instances with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, and Public IPv4. There are four instances listed: 'MondayTest1' (Running, t2.micro, 2/2 checks passed, us-east-1c, ec2-54-225-22-157.co..., 54.225.22.157), 'MondayTest1' (Running, t2.micro, 2/2 checks passed, us-east-1c, ec2-18-215-175-37.co..., 18.215.175.37), 'i-0a149f1a3ebcf067' (Pending, t2.micro, -), and 'i-02a6f2ed32c61c672' (Pending, t2.micro, -). A red box highlights the last two pending instances.

Figure 42 Automatic Instance Creation

43. Terminating an Instance

Terminating an Instance to check for proper functioning of Auto-Scaling

The screenshot shows the 'Terminate instance?' dialog box. At the top, it displays the instance state (Pending), instance type (t2.micro), availability zone (us-east-1e), and public IP (ec2-54-173-163-163). The dialog title is 'Terminate instance?'. A warning message in a yellow box states: '⚠️ On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.' Below the warning, a question asks: 'Are you sure you want to terminate these instances?'. A table shows the selected instance (i-03abd12b635d3ae44, MondayTest1) with 'Termination protection' set to 'Disabled'. At the bottom, a message says: 'To confirm that you want to terminate the instances, choose the terminate button below. Instances with termination protection enabled will not be terminated. Terminating the instance cannot be undone.' There are 'Cancel' and 'Terminate' buttons at the bottom right, with 'Terminate' being orange.

Figure 43 Terminating an Instance

44. Other Instance Running Automatically

The screenshot shows the AWS EC2 Instances page. The left sidebar includes options like EC2 Dashboard, EC2 Global View, Events, Console-to-Code Preview, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, CloudShell, and Feedback. The main content area displays a table titled "Instances (4) Info" with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, Public IPv4, and Elastic IP. The table lists four instances: MondayTest1 (terminated), MondayTest1 (running), i-0a149f1a1ebcf067 (running), and i-02a6f2ed32c61c672 (running). The last instance is highlighted with a red box.

Figure 44 Instance Running Automatically

45. Dynamic Scaling Policy

The screenshot shows the AWS Auto Scaling Groups page. The left sidebar includes options like Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, Load Balancing, Load Balancers, Target Groups, Trust Stores, Auto Scaling, and Auto Scaling Groups. The main content area shows a message "Dynamic scaling policy created or edited successfully." under the heading "MondayAS". It features tabs for Details, Activity, Automatic scaling (which is selected), Instance management, Monitoring, and Instance refresh. A callout box explains that scaling policies resize your Auto Scaling group to meet changes in demand. Below this, the "Automatic scaling" section shows "Dynamic scaling policies (1) Info". It lists a "Target Tracking Policy" with settings: Target tracking scaling, Enabled, As required to maintain Average CPU utilization at 50, Add or remove capacity units as required, 30 seconds to warm up before including in metric, and Enabled. Another section for "Predictive scaling policies (0) Info" is also present. The bottom of the page includes CloudShell, Feedback, and standard footer links.

Figure 45 Dynamic Scaling Policy

46. Connecting to the Instance via SSH

```
ec2-user@ip-172-31-20-183:~$ >icacls chabi.pem /inheritance:r /grant:r "%USERNAME%:R"
processed file: chabi.pem
Successfully processed 1 files; Failed processing 0 files

ec2-user@ip-172-31-20-183:~$ >ssh -i "chabi.pem" ec2-user@18.215.175.37
The authenticity of host '18.215.175.37 (18.215.175.37)' can't be established.
ED25519 key fingerprint is SHA256:J+H0/Scdy0S+Dk1uD4u1amx0m9w+up5BhJ7duKDuzNE.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '18.215.175.37' (ED25519) to the list of known hosts.

#_
~\_ #####_      Amazon Linux 2023
~~ \_\#\#\#\`\
~~ \#\#\#
~~ \#/| ___ https://aws.amazon.com/linux/amazon-linux-2023
~~ V~,`>
~~ /
~~ .`_/
~~ /-/
~/`'

[ec2-user@ip-172-31-20-183 ~]$ |
```

Figure 46 Connecting to Instance via SSH

47. Stress Test Installation

```
ec2-user@ip-172-31-20-183:~$ sudo yum install stress -y
Last metadata expiration check: 1:24:12 ago on Mon Jan 22 10:01:37 2024.
Dependencies resolved.
=====
Package           Architecture      Version       Repository      Size
=====
Installing:
stress            x86_64          1.0.4-28.amzn2023.0.2   amazonlinux    37 k
Transaction Summary
=====
Install 1 Package

Total download size: 37 k
Installed size: 78 k
Downloading Packages:
stress-1.0.4-28.amzn2023.0.2.x86_64.rpm          477 kB/s | 37 kB   00:00
Total                                         264 kB/s | 37 kB   00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing           : 1/1
Installing         : stress-1.0.4-28.amzn2023.0.2.x86_64 1/1
Running scriptlet: stress-1.0.4-28.amzn2023.0.2.x86_64 1/1
Verifying          : stress-1.0.4-28.amzn2023.0.2.x86_64 1/1
```

Figure 47 Stress Test Installation

48. Performing Stress Test

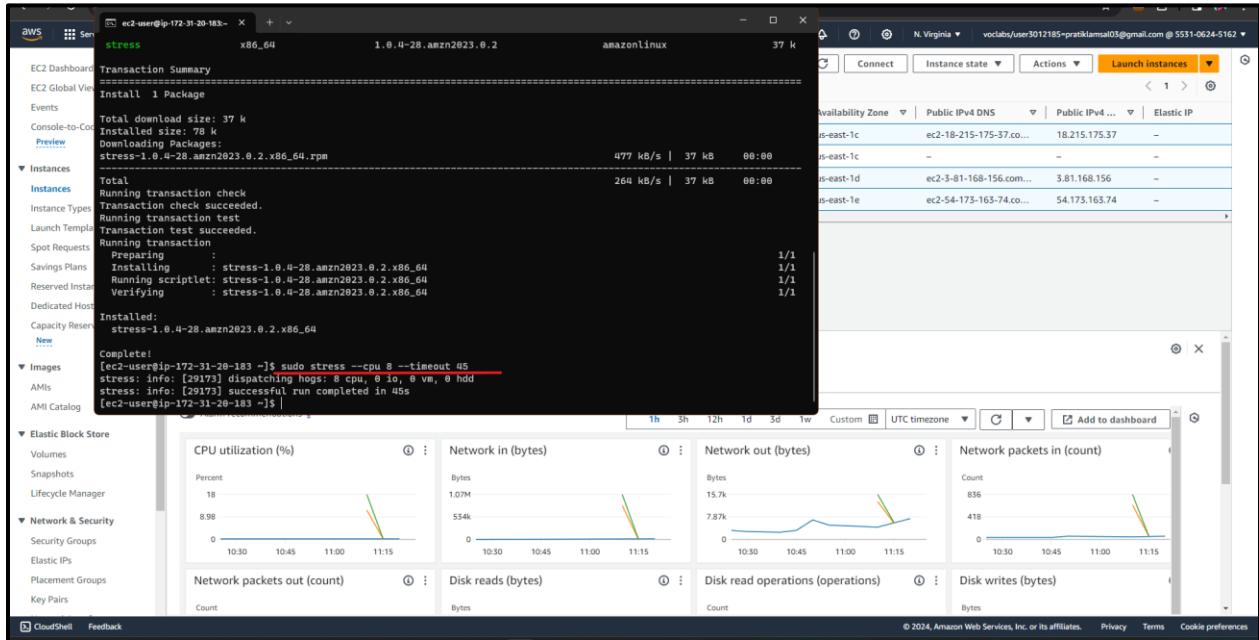


Figure 48 Stress Test

49. Stress Test Complete

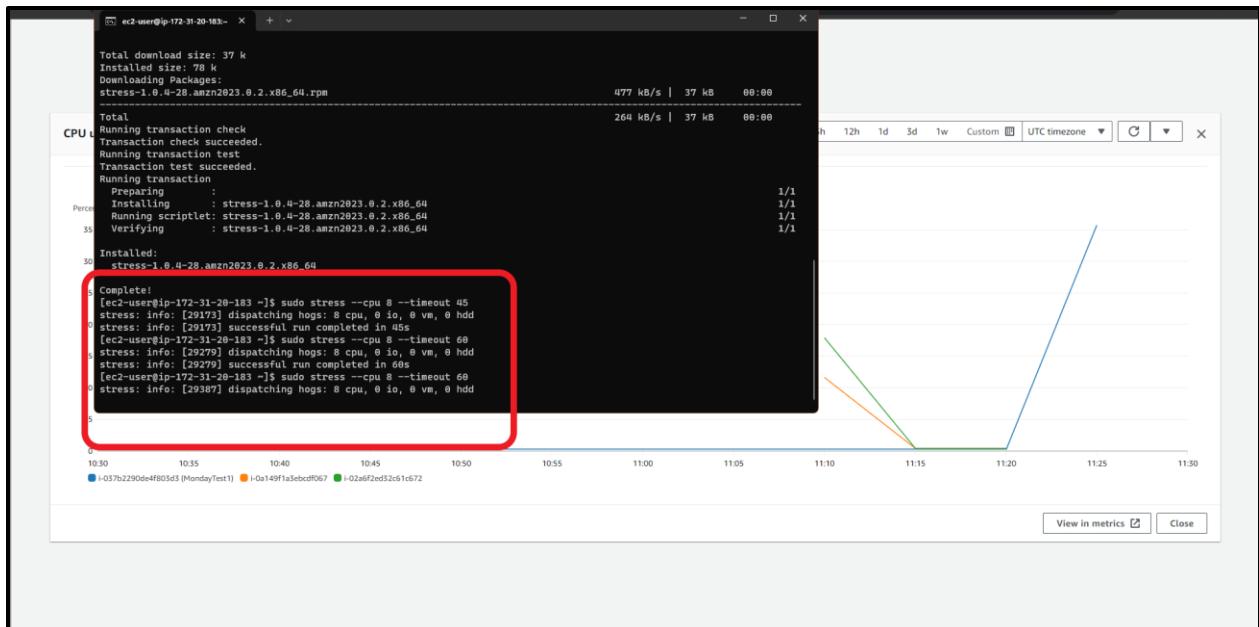


Figure 49 Stress-Test Complete

50. Monitoring Instances

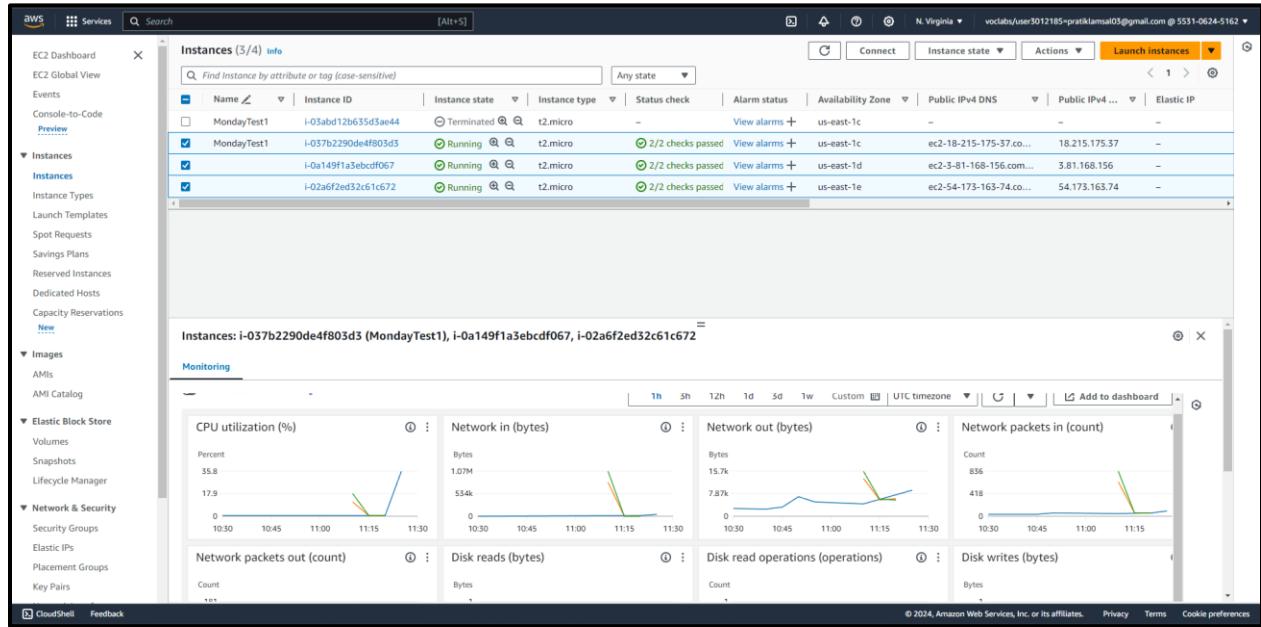


Figure 50 Monitoring Instances

51. TASK COMPLETE

Task Complete