

A PROJECT REPORT ON
Working with EC2, ELB, ASG, and Cloud Watch

Submitted in partial fulfillment of the requirement for the award of
APSSDC - IDS Cloud Architecture

SUBMITTED BY
CHANDU SIVA SANKAR

SUPERVISED BY
Ms. VORUGANTI ANUPAMA
AWS Cloud Engineer at IDC

Special Thanks To
Mr. Gumpa Jagan Mohan
Associate Manager at APSSDC



AP State Skill Development Corporation (APSSDC)
ANDHRA PRADESH

Amazon Elastic Compute Cloud

Amazon Elastic Compute Cloud (EC2) is a part of Amazon Web Services (AWS). EC2 is Infrastructure as a Service offering from AWS that provides compute resources in AWS cloud.

Working with EC2, ELB, and ASG

Step 1 - Creating an EC2 Instance and Use Public IP address to connect it.

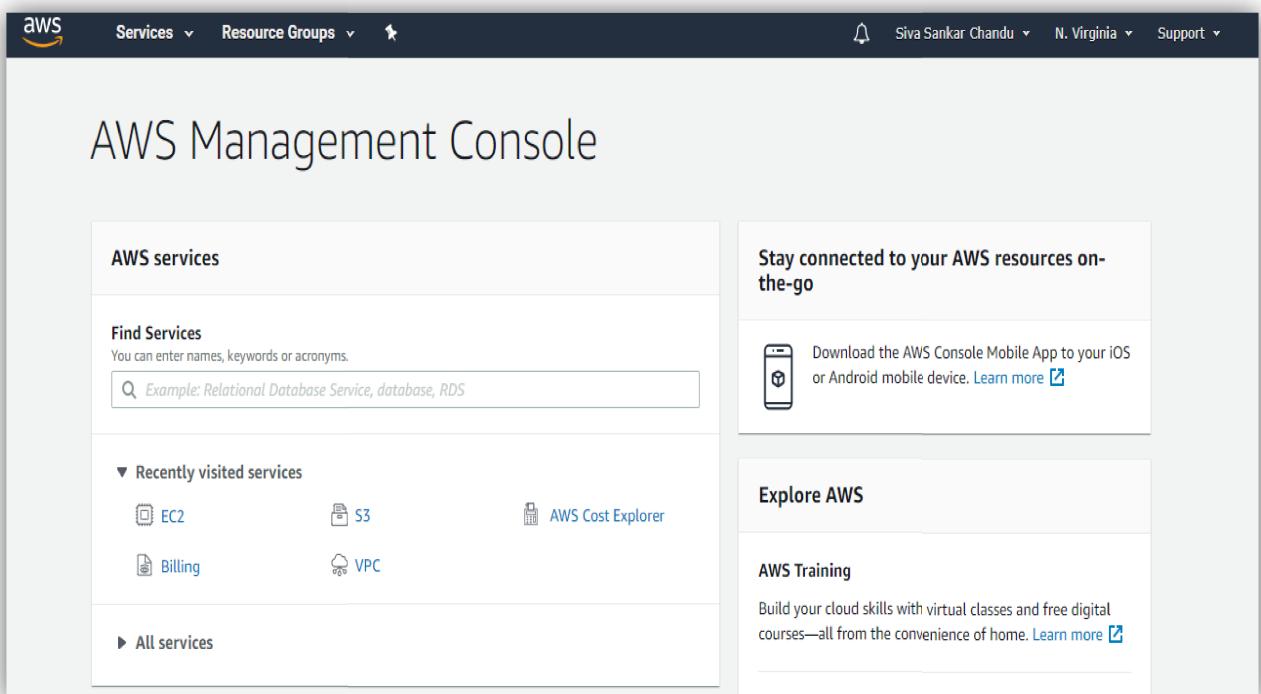
Step 2 - Create an ELB and Target Group, and Through the DNS Name connect it.

Step 3 - Create a Launch configuration then create Auto Scaling Group (ASG).

Step 4 – Cleaning process.

Step 1 - Creating an EC2 Instance and Use Public IP address to connect it.

➤ **First Login into the AWS Console.**



Working with EC2, ELB, ASG and CloudWatch

- Open EC2 Dashboard, Next Click on Running Instances.

New EC2 Experience
Tell us what you think

EC2 Dashboard New

Events New

Tags

Limits

Instances

- Instances
- Instance Types
- Launch Templates
- Spot Requests
- Savings Plans
- Reserved Instances
- Dedicated Hosts New
- Scheduled Instances
- Capacity Reservations

Images

AMIs

Resources

You are using the following Amazon EC2 resources in the US East (N. Virginia) Region:

Running instances	0	Elastic IPs	0
Dedicated Hosts	0	Snapshots	0
Volumes	0	Load balancers	0
Key pairs	1	Security groups	9
Placement groups	0		

Account attributes

Supported platforms [?]
• VPC

Default VPC [?]
vpc-270efd5a

Settings

EBS encryption

Zones

Default credit specification

Console experiments

Explore AWS [?]

- Next, Click on Launch Instance -> Launch Instance.

New EC2 Experience
Tell us what you think

EC2 Dashboard New

Events New

Tags

Limits

Instances

Instances

- Instance Types
- Launch Templates
- Spot Requests
- Savings Plans
- Reserved Instances
- Dedicated Hosts New
- Scheduled Instances
- Capacity Reservations

Images

AMIs

Launch Instance

Launch instance from template

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
Manual Insta...	i-0d82454800ec5f46b	t2.micro	us-east-1c	terminated	None		

Instance: i-0d82454800ec5f46b (Manual Instance) Public DNS: -

Description Status Checks Monitoring Tags

Instance ID	i-0d82454800ec5f46b	Public DNS (IPv4)	-
Instance state	terminated	IPv4 Public IP	-
Instance type	t2.micro	IPv6 IPs	-
Finding	Opt-in to AWS Compute Optimizer for	Elastic IPs	

Feedback English (US) © 2008 – 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Working with EC2, ELB, ASG and CloudWatch

- Choose an AMI (i.e., Amazon Linux 2 AMI 64-bit (x86)).

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace, or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"

Quick Start

- My AMIs
- AWS Marketplace
- Community AMIs
- Free tier only ⓘ

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0c94855ba95c71c99 (64-bit x86) / ami-0d29b48622869df9 (64-bit ARM)

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-00514a528eadbc95b

The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

Feedback English (US) © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

- Choose an Instance Type (t2.micro) and Click Next.

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance types Current generation Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

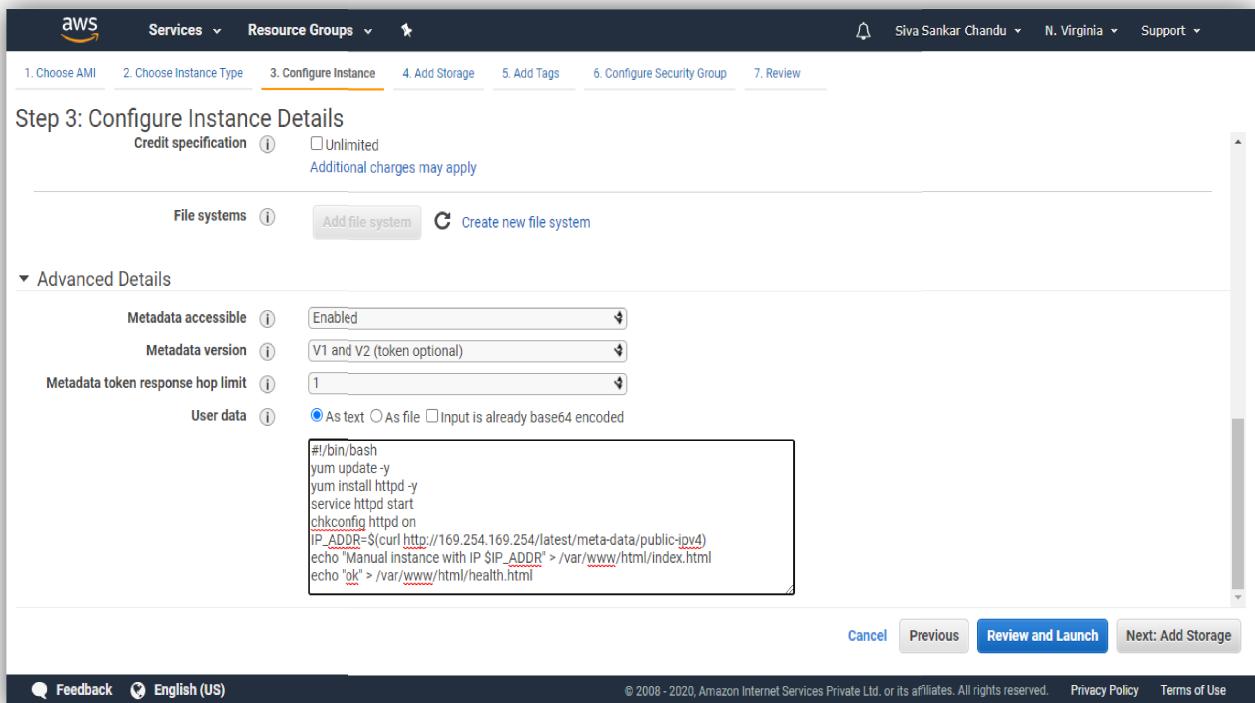
	Family	Type	vCPUs ⓘ	Memory (GiB)	Instance Storage (GB) ⓘ	EBS-Optimized Available ⓘ	Network Performance ⓘ	IPv6 Support ⓘ
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

Feedback English (US) © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Working with EC2, ELB, ASG and CloudWatch

- Next, Configure Instance Details (Insert user data to display information when connecting to the instance through the Public IP address) and Click Next.

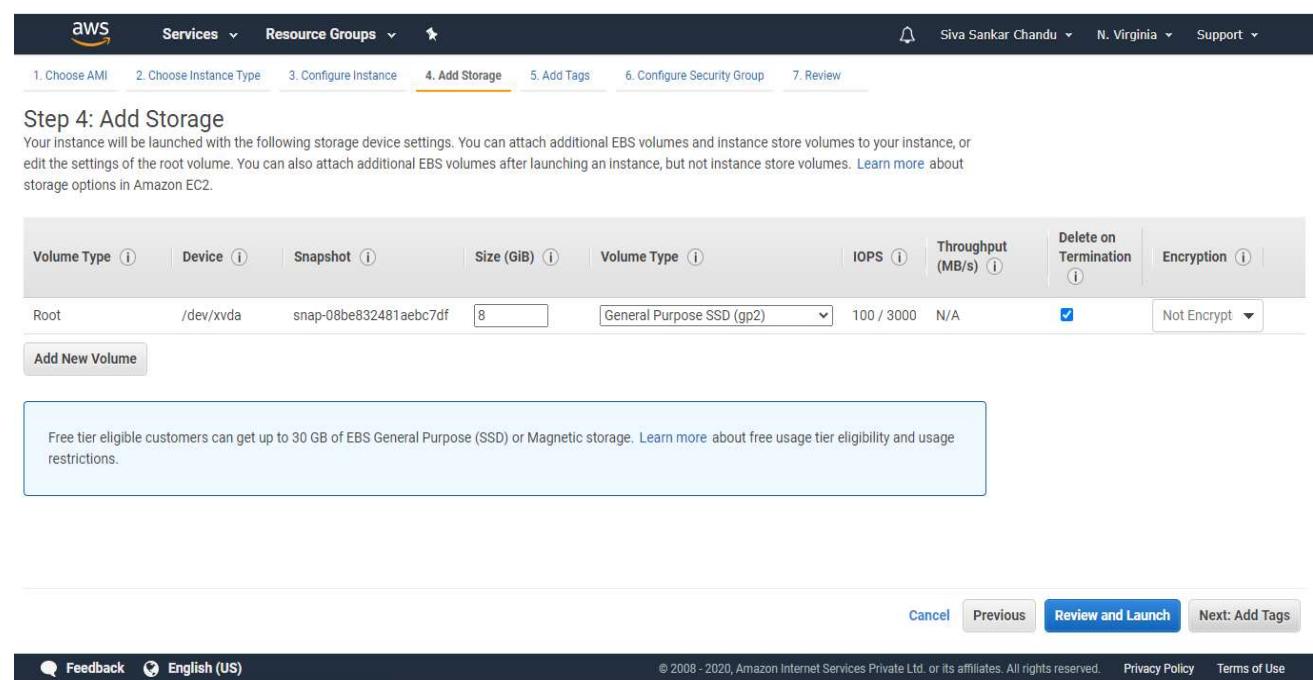


The screenshot shows the 'Configure Instance Details' step of the EC2 wizard. In the 'User data' section, the following script is pasted:

```
#!/bin/bash
yum update -y
yum install httpd -y
service httpd start
chkconfig httpd on
IP_ADDR=$(curl http://169.254.169.254/latest/meta-data/public-ipv4)
echo "Manual instance with IP $IP_ADDR" > /var/www/html/index.html
echo "ok" > /var/www/html/health.html
```

Below the script, there are several configuration options: Metadata accessible (Enabled), Metadata version (V1 and V2 (token optional)), Metadata token response hop limit (1), and User data type (As text). The 'Review and Launch' button is highlighted in blue at the bottom right.

- Add Storage (Keep everything as default) and Click Next.



The screenshot shows the 'Add Storage' step of the EC2 wizard. A new volume is being created with the following settings:

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-08be832481aebc7df	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypt

A note at the bottom left states: "Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Learn more about free usage tier eligibility and usage restrictions."

At the bottom right, the 'Review and Launch' button is highlighted in blue.

Working with EC2, ELB, ASG and CloudWatch

➤ Add Tags (Add tag for Instance i.e., Name)

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	(128 characters maximum)	Value	(256 characters maximum)	Instances	Volumes
Name		Manual Instance		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Add another tag (Up to 50 tags maximum)

Cancel Previous Review and Launch Next: Configure Security Group

Feedback English (US) © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

➤ Configure Security Group (Add Rule i.e., Type is HTTP and Keep source as Anywhere) and Click Next.

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group Select an existing security group

Security group name: launch-wizard-6

Description: launch-wizard-6 created 2020-09-09T15:51:52.340+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere	0.0.0.0/0
HTTP	TCP	80	Custom	0.0.0.0/0

Add Rule

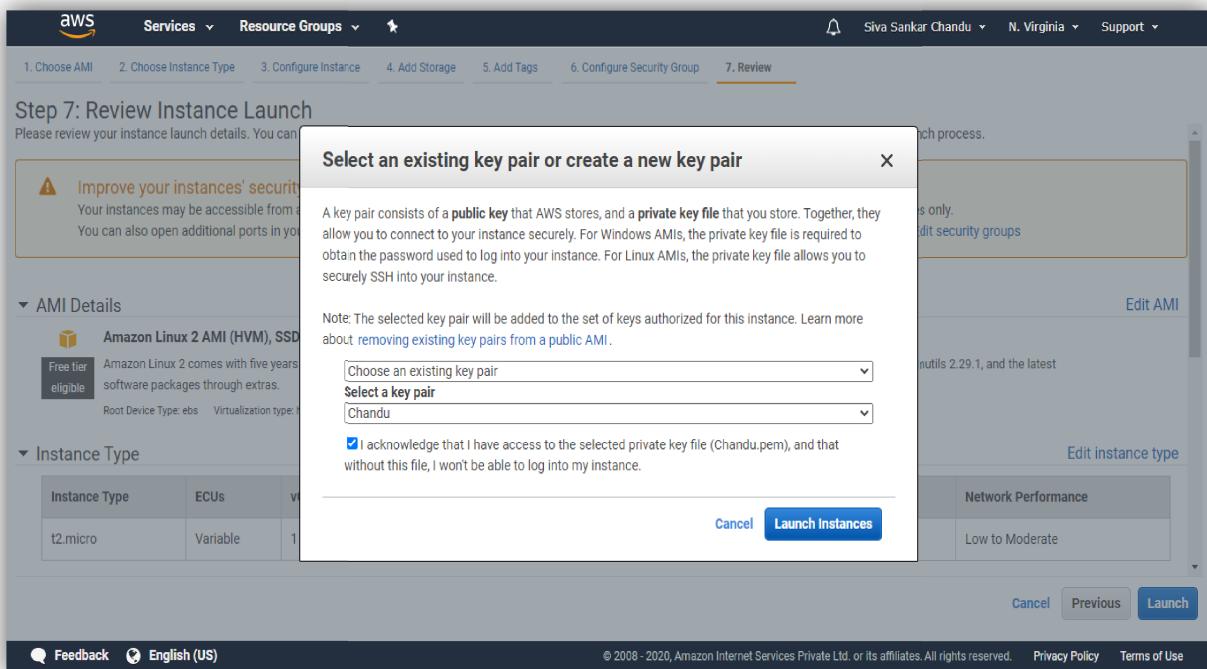
Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Cancel Previous Review and Launch

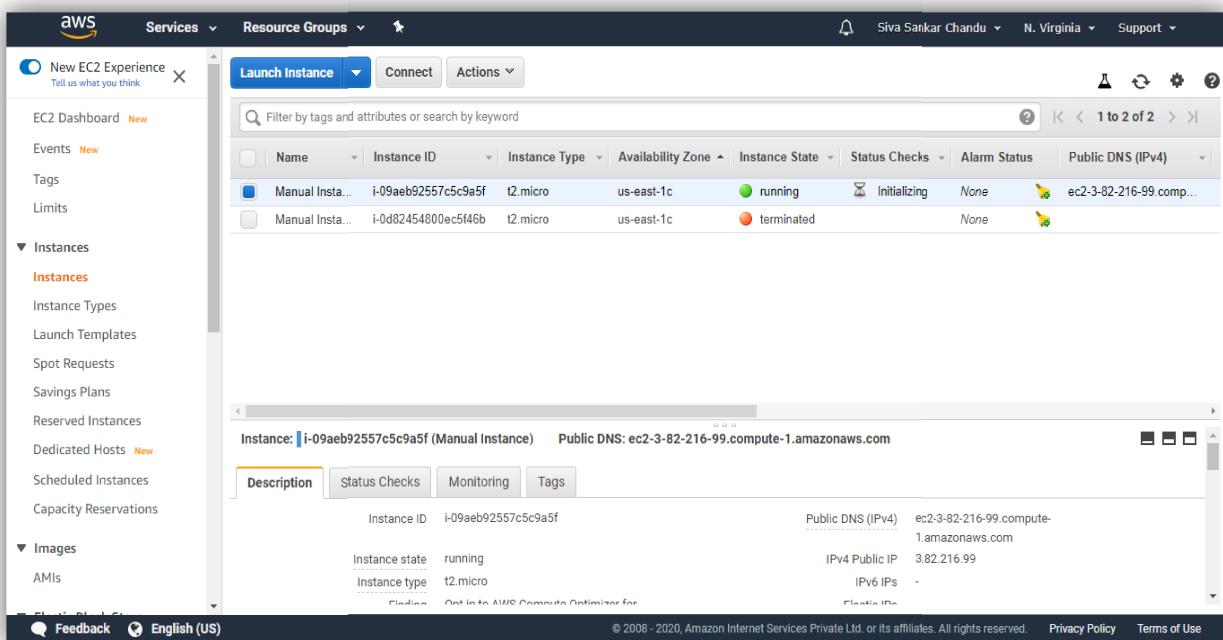
Feedback English (US) © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Working with EC2, ELB, ASG and CloudWatch

- **Review Instance Launch (Choose Key pair (Select New key pair or Existing Key pair. If we choose a new key pair, give name to that key and download the Pem file. After downloading the Pem file, Launch Instance button is enabled Else Choose Existing Key pair)) and Click Launch Instance**



- **EC2 Instance was Launched (Name of the EC2 Instance is Manual Instance) successfully**



Working with EC2, ELB, ASG and CloudWatch

- Copied Public IP Address to host EC2 Instance.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with options like EC2 Dashboard, Events, Tags, Limits, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, and Images. The main area displays two instances:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
Manual Insta...	i-09aeb92557c5c9a5f	t2.micro	us-east-1c	running	Initializing	None	ec2-3-82-216-99.compute-1.amazonaws.com
Manual Insta...	i-0d82454800ec5f46b	t2.micro	us-east-1c	terminated	-	None	-

In the details view for the first instance, the Public DNS (IPv4) field is highlighted with a blue background, and a tooltip 'Copied' is shown above it. The tooltip has a small arrow pointing towards the field.

- After Connecting Instance through the browser with Public IP Address.

The screenshot shows a web browser window with the following details:

- Title bar: WhatsApp, Instances | EC2 Management Con...
- Address bar: 3.82.216.99
- Content area: Manual instance with IP 3.82.216.99
- Toolbar: Back, Forward, Stop, Refresh, Home, New tab, etc.
- Address bar: Not secure | 3.82.216.99
- Taskbar: Apps, HOME | Siva, Capgemini Tech Ch..., (16) Harshit vashist., New Document - Pr..., Hemingway Editor, Amazon - Home ..., AWS Management..., TCS iON Digital Lea...

Step 2 - Create an ELB and Target Group, and Use DNS name to connect it.

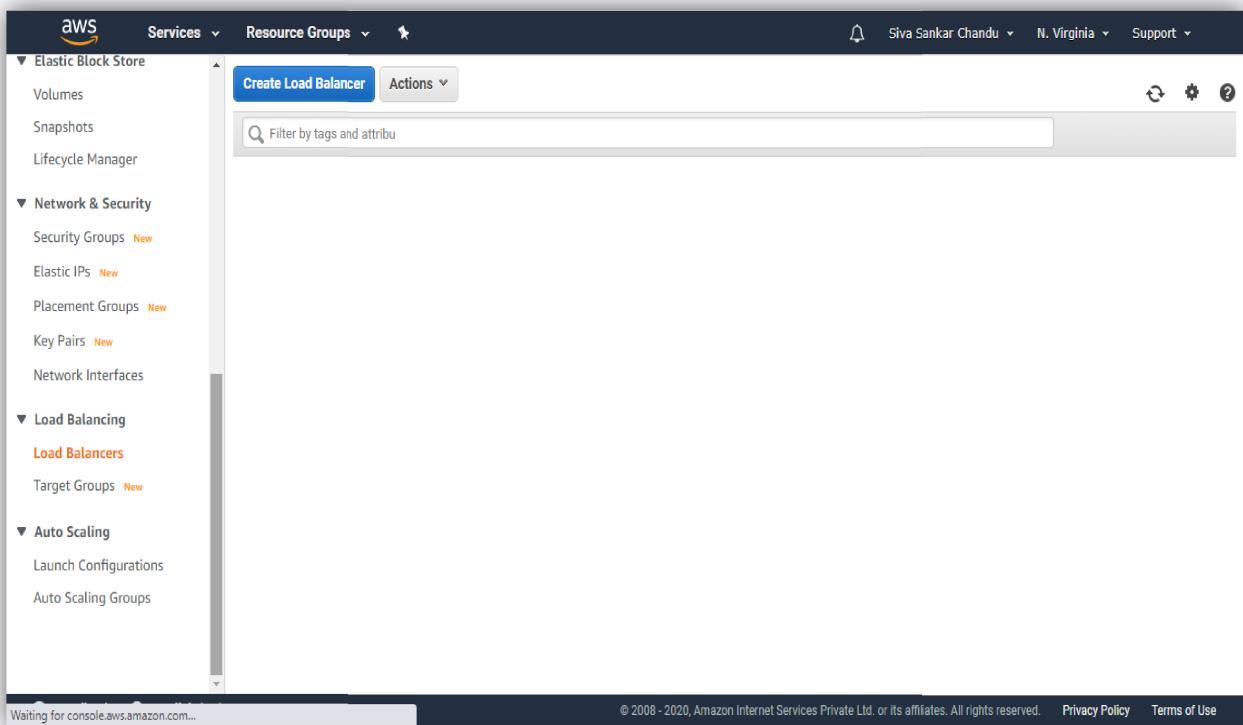
Elastic Load Balancing

Elastic Load Balancing (ELB) automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, and Lambda functions.

Target Group

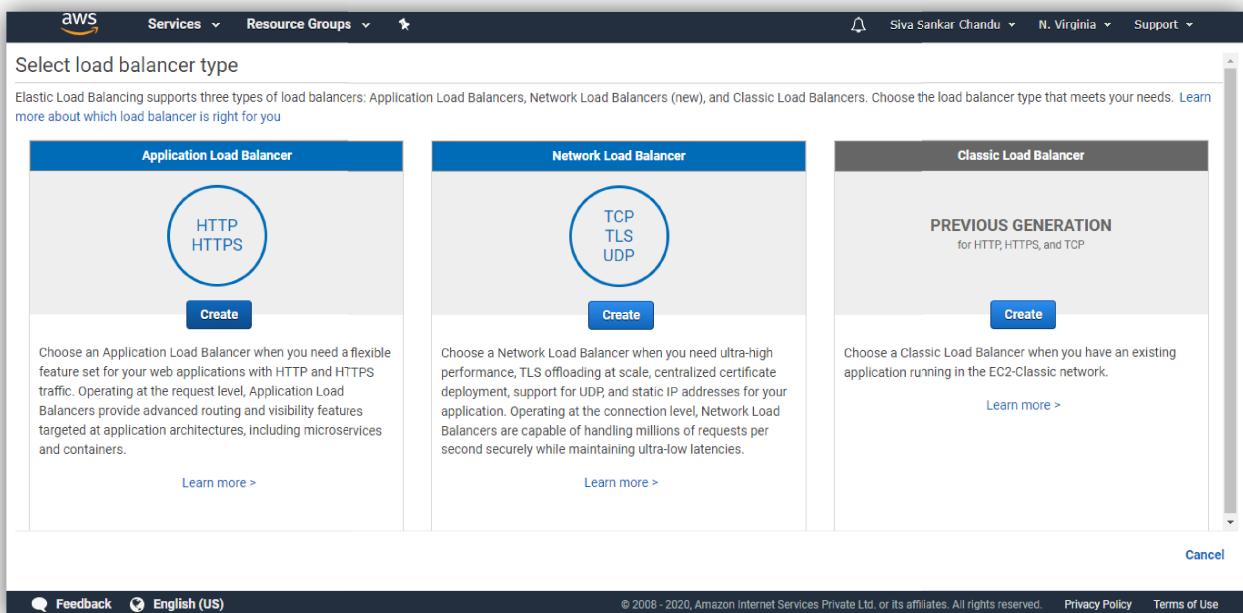
A target group tells a load balancer where to direct traffic to: EC2 instances, fixed IP addresses; or AWS Lambda functions, amongst others. When creating a load balancer, you create one or more listeners and configure listener rules to direct the traffic to one target group

➤ **Open Load Balancer and Click on Create Load Balancer to create Load Balancer.**

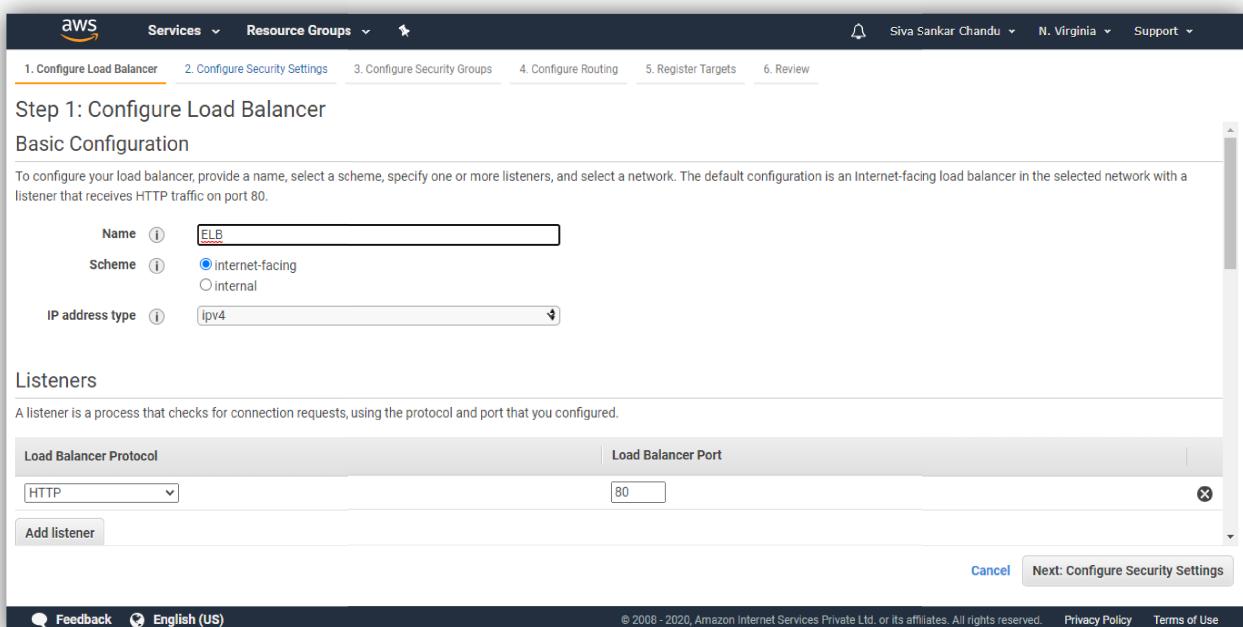


Working with EC2, ELB, ASG and CloudWatch

- Select Load Balancer type (i.e., Application Load Balancer, Click on Create button to create load balancer)



- Give name to Load Balancer i.e., ELB and Scroll down the page.



Working with EC2, ELB, ASG and CloudWatch

- ADD Subnets in Availability Zones section and Click Next.

Step 1: Configure Load Balancer

VPC: vpc-270efd5a (172.31.0.0/16) (default)

Availability Zones:

- us-east-1a subnet-f0de7daf IPv4 address Assigned by AWS
- us-east-1b subnet-d03f98b6 IPv4 address Assigned by AWS
- us-east-1c subnet-74852855 IPv4 address Assigned by AWS
- us-east-1d subnet-8a6d1bc7 IPv4 address Assigned by AWS
- us-east-1e subnet-6338d252 IPv4 address Assigned by AWS
- us-east-1f subnet-45ea724b IPv4 address Assigned by AWS

Cancel Next: Configure Security Settings

Feedback English (US) © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

- We are selected http that's why it is showing warning message, ignore it and Click Next.

Step 2: Configure Security Settings

⚠ Improve your load balancer's security. Your load balancer is not using any secure listener.
If your traffic to the load balancer needs to be secure, use the HTTPS protocol for your front-end connection. You can go back to the first step to add/configure secure listeners under Basic Configuration section. You can also continue with current settings.

Cancel Previous Next: Configure Security Groups

Feedback English (US) © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Working with EC2, ELB, ASG and CloudWatch

- Keep Source as Anywhere and Click Next

The screenshot shows the AWS Load Balancer wizard at Step 3: Configure Security Groups. The security group name is 'load-balancer-wizard-3'. A single rule is defined: Type: Custom TCP F, Protocol: TCP, Port Range: 80, Source: Anywhere (0.0.0.0/0, ::/0). The 'Add Rule' button is visible.

- Give name to Target Group (i.e., TG) and give path (i.e., /health.html) and Go to Advance Health Settings to Scroll down and Click Next.

The screenshot shows the AWS Load Balancer wizard at Step 4: Configure Routing. A new target group is being created with the name 'TG'. The target type is set to 'Instance'. The protocol is 'HTTP' and the port is '80'. Under 'Health checks', the protocol is 'HTTP' and the path is '/health.html'. The 'Next: Register Targets' button is visible.

Working with EC2, ELB, ASG and CloudWatch

- **Change Health threshold, Unhealthy threshold, Interval (Make an instance healthy based on this settings; If we reduce time it will make instance healthy in less time) and Click Next.**

The screenshot shows the 'Step 4: Configure Routing' page of the wizard. It includes fields for 'Protocol' (HTTP), 'Port' (80), and 'Health checks' (Protocol: HTTP, Path: /health.html). Below these, under 'Advanced health check settings', there are fields for 'Healthy threshold' (2), 'Unhealthy threshold' (2), 'Timeout' (5 seconds), 'Interval' (6 seconds), and 'Success codes' (200). At the bottom right are 'Cancel', 'Previous', and 'Next: Register Targets' buttons.

- **Click on Add to registered to register instance and Click on Next.**

The screenshot shows the 'Step 5: Register Targets' page of the wizard. It displays a table of 'Registered targets' with one entry: 'i-09aeb92557c5c9a5f' (Manual Instance, port 80, running, security group launch-wizard-6, zone us-east-1c). Below this, an 'Instances' section shows a table with the same data. A 'Remove' button is available for the registered target. At the bottom right are 'Cancel', 'Previous', and 'Next: Review' buttons.

Working with EC2, ELB, ASG and CloudWatch

- Review the Details and Click on Create.

The screenshot shows the AWS Management Console interface for creating a load balancer. The top navigation bar includes 'Services' (dropdown), 'Resource Groups' (dropdown), and account information ('Siva Sankar Chandu', 'N. Virginia', 'Support'). Below the navigation is a breadcrumb trail: '1. Configure Load Balancer', '2. Configure Security Settings', '3. Configure Security Groups', '4. Configure Routing', '5. Register Targets', and '6. Review'. The 'Review' step is highlighted with an orange underline. The main content area is titled 'Step 6: Review' with the sub-instruction 'Please review the load balancer details before continuing'. It displays three sections: 'Load balancer' (Name: ELB, Scheme: internet-facing, Listeners: Port 80 - Protocol: HTTP, IP address type: ipv4, VPC: vpc-270efd5a, Subnets: subnet-f0de7daf, subnet-d03f98b6, subnet-74852855, subnet-8a6d1bc7, subnet-6338d252, subnet-45ea724b, Tags), 'Security groups' (Security group: load-balancer-wizard-3), and 'Routing' (Target group: New target group, Target group name: TG, Port: 80, Target type: instance, Protocol: HTTP). At the bottom right are 'Cancel', 'Previous', and 'Create' buttons, with 'Create' being blue and bold.

- Load Balancer Created successfully and click Close to View the Load Balancer ELB.

The screenshot shows a modal dialog titled 'Load Balancer Creation Status'. It contains a green success message: 'Successfully created load balancer' with a checkmark icon, followed by the text 'Load balancer ELB was successfully created.' and a note: 'Note: It might take a few minutes for your load balancer to be fully set up and ready to route traffic, and for the targets to complete the registration process and pass the initial health checks.' Below this is a 'Suggested next steps' section with two bullet points: 'Discover other services that you can integrate with your load balancer. Visit the [Integrated services](#) tab within [ELB](#)' and 'Consider using AWS Global Accelerator to further improve the availability and performance of your applications. [AWS Global Accelerator console](#)'. At the bottom right of the dialog is a 'Close' button. The footer of the dialog includes 'Feedback', 'English (US)', and standard copyright and policy links.

Working with EC2, ELB, ASG and CloudWatch

- Load Balancer (ELB) created.

The screenshot shows the AWS Elastic Load Balancing (ELB) service page. On the left, there's a navigation sidebar with options like 'Elastic Block Store', 'Network & Security', 'Load Balancing' (which is expanded), and 'Auto Scaling'. The main area displays a table of existing load balancers. One row is selected, showing details for an ELB named 'ELB'. The 'Basic Configuration' section includes fields for Name (ELB), ARN (arn:aws:elasticloadbalancing:us-east-1:854231420140:loadbalancer/app/ELB/6c66d9e4557f3fc2), DNS name (ELB-579581788.us-east-1.elb.amazonaws.com), State (provisioning), Type (application), Scheme (internet-facing), and IP address type (ipv4). Below this, there's a link to 'Edit IP address type'.

- Target Group (TG) also created and Instance status is healthy.

The screenshot shows the AWS Target Groups (TG) service page. The left sidebar has sections for 'Elastic Block Store', 'Network & Security', 'Load Balancing' (expanded), and 'Auto Scaling'. The main area shows a target group named 'TG'. The 'Basic configuration' section includes fields for Target type (instance), Protocol : Port (HTTP : 80), VPC (vpc-270efd5a), and Load balancer (ELB). Below this, the 'Targets' tab is selected, showing a table of registered targets. One target is listed: 'i-09ae92557c5c9a5f' (Manual Instance, Port 80, Zone us-east-1c, Status healthy).

Working with EC2, ELB, ASG and CloudWatch

- Copy the DNS name of Load Balancer ELB.

The screenshot shows the AWS Elastic Load Balancing (ELB) console. On the left, there's a navigation sidebar with options like 'Elastic Block Store', 'Network & Security', 'Load Balancing' (which is expanded to show 'Load Balancers'), and 'Auto Scaling'. The main area displays a table of load balancers. One row is selected, showing details: Name: ELB, DNS name: ELB-579581788.us-east-1.elb.amazonaws.com, State: active, VPC ID: vpc-270efdf5a, Availability Zones: us-east-1f, us-east-1e, Type: application. Below the table, there's a 'Basic Configuration' section with fields for Name, ARN, DNS name, State, Type, Scheme, and IP address type. The 'DNS name' field is currently selected. At the bottom right of this section, there's a green button labeled 'Copied' with a clipboard icon. The status bar at the bottom of the browser window shows the URL: elb-579581788.us-east-1.elb.amazonaws.com.

- Opened new Tab and Paste the DNS name copied from Load Balancer ELB, Click Enter, Then it is connected with EC2 Instance shown in below image

The screenshot shows a web browser window with multiple tabs open. The active tab has the URL 'elb-579581788.us-east-1.elb.amazonaws.com'. The page content is a simple white space with the text 'Manual instance with IP 3.82.216.99' centered. The browser's status bar at the bottom shows the same URL as the active tab.

Step 3 – Create a Launch Configuration then create Auto Scaling Group (ASG).

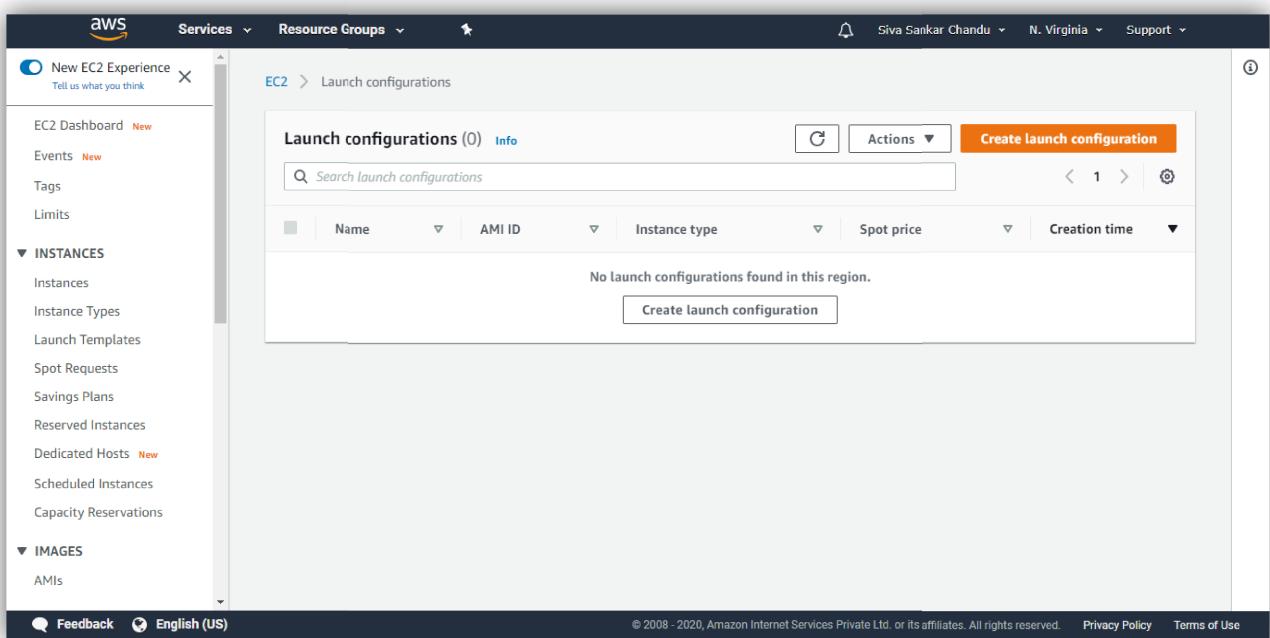
Auto Scaling Group

An Auto Scaling Group (ASG) contains a collection of Amazon EC2 instances that are treated as a logical grouping for the purposes of automatic scaling and management. An Auto Scaling group also enables you to use Amazon EC2 Auto Scaling features such as health check replacements and scaling policies.

Amazon CloudWatch

Amazon CloudWatch is a monitoring and management service that provides data and actionable insights for AWS, hybrid, and on-premises applications and infrastructure resources. With CloudWatch, you can collect and access all your performance and operational data in form of logs and metrics from a single platform.

- Open Launch Configuration and Click on Create Launch Configuration.



Working with EC2, ELB, ASG and CloudWatch

➤ Give Name and Select AMI.

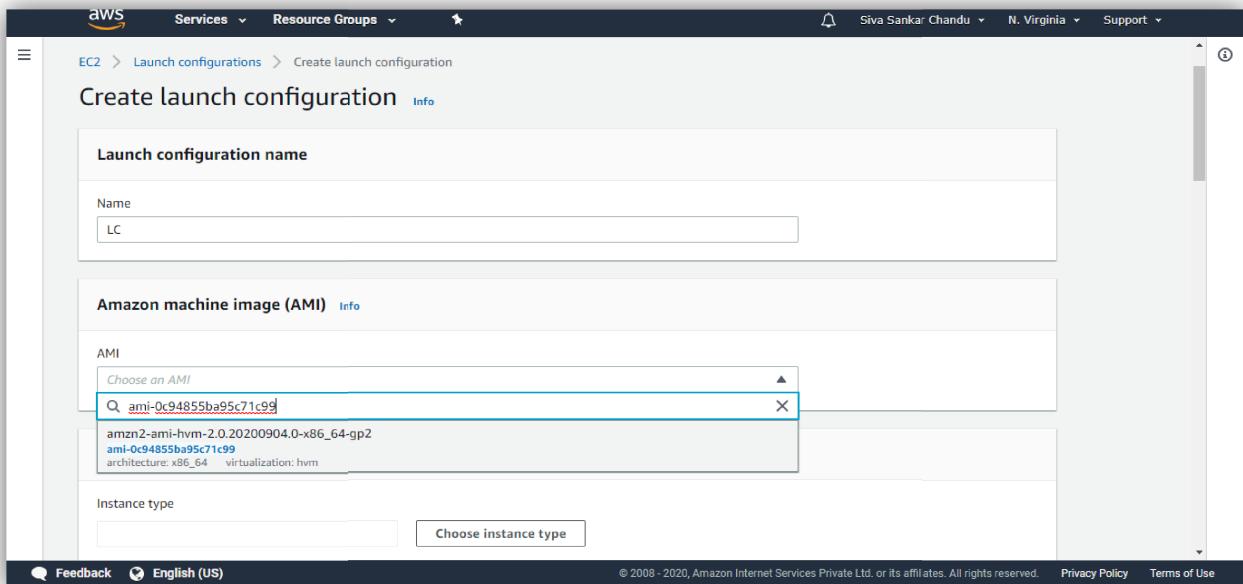
The screenshot shows the 'Create launch configuration' wizard. Step 1: Choose AMI. The 'Launch configuration name' field is empty. The 'Amazon machine image (AMI)' dropdown is set to 'Choose an AMI'. The 'Instance type' dropdown is also empty. At the bottom, there are links for 'Feedback', 'English (US)', and navigation buttons for 'Cancel', 'Next Step', and 'Create'.

➤ Copy the AMI ID (Take Amazon Linux AMI ID in EC2 (New Instance Creation)).

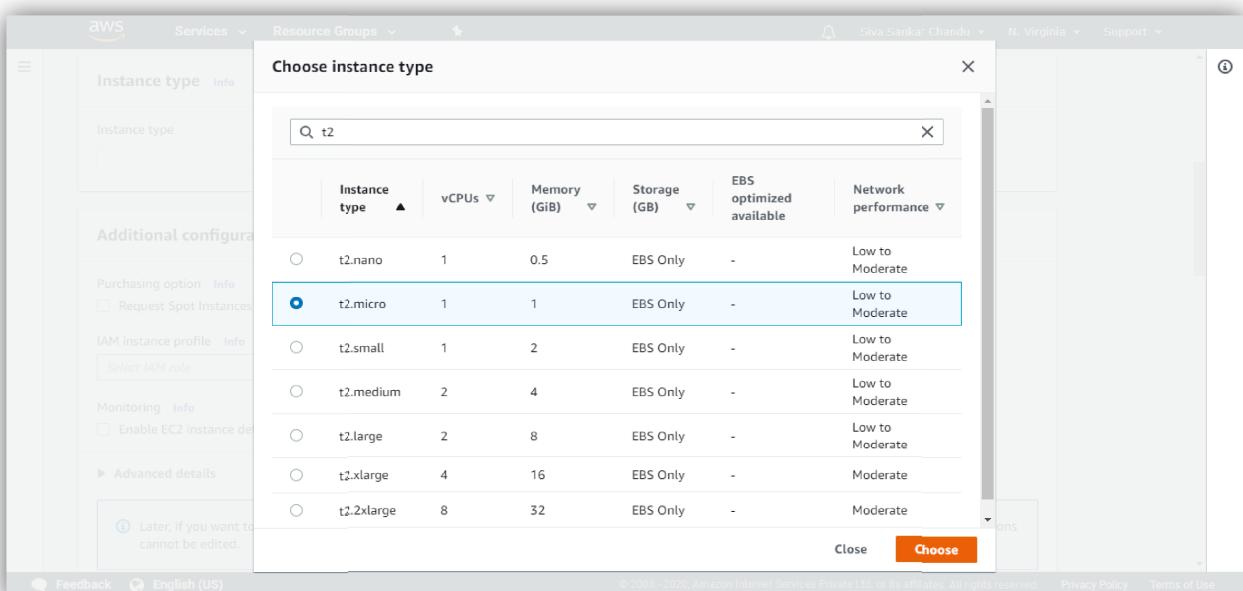
The screenshot shows the 'Step 1: Choose an Amazon Machine Image (AMI)' step. The 'Search for an AMI by entering a search term e.g. "Windows"' field is empty. The results list shows two items: 'Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0c94855ba95c71c99' (selected) and 'Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-00514a528eadbc95b'. Both items have a 'Select' button. The sidebar on the left includes sections for 'My AMIs', 'AWS Marketplace', 'Community AMIs', and a 'Free tier only' checkbox. At the bottom, there are links for 'Feedback', 'English (US)', and navigation buttons for 'Cancel and Exit', '1 to 40 of 40 AMIs', and 'Next Step'.

Working with EC2, ELB, ASG and CloudWatch

- Paste the copied AMI ID in AMI Field and Select the AMI, and Scroll down.

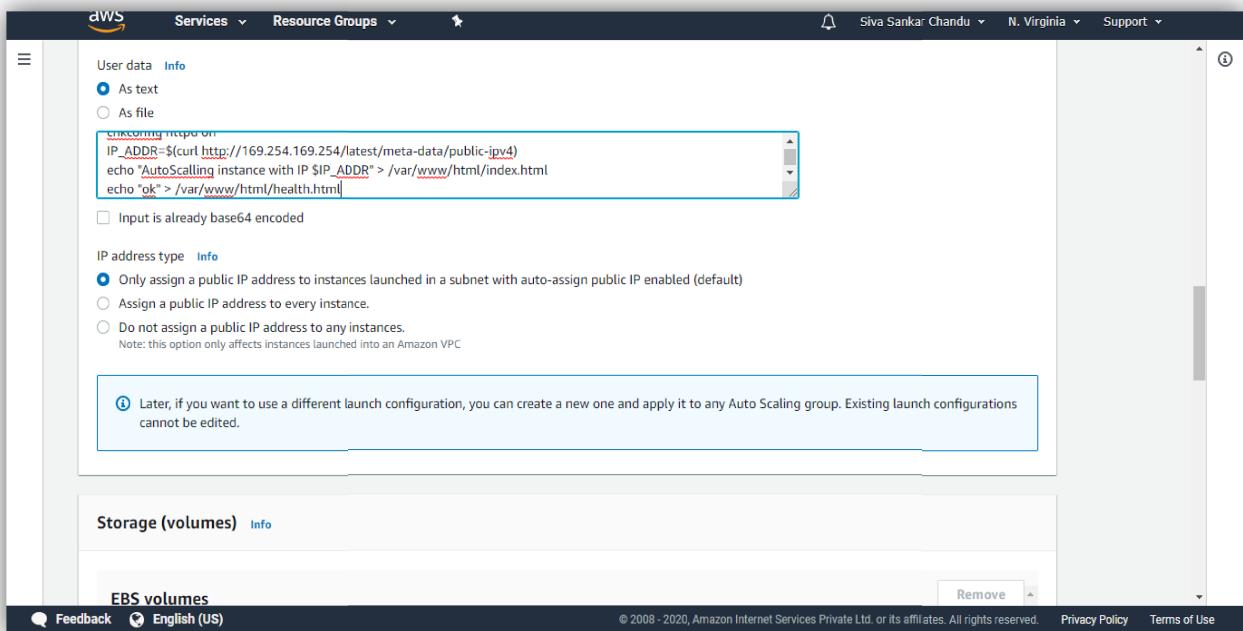


- Choose Instance Type (t2.micro) and Scroll down.

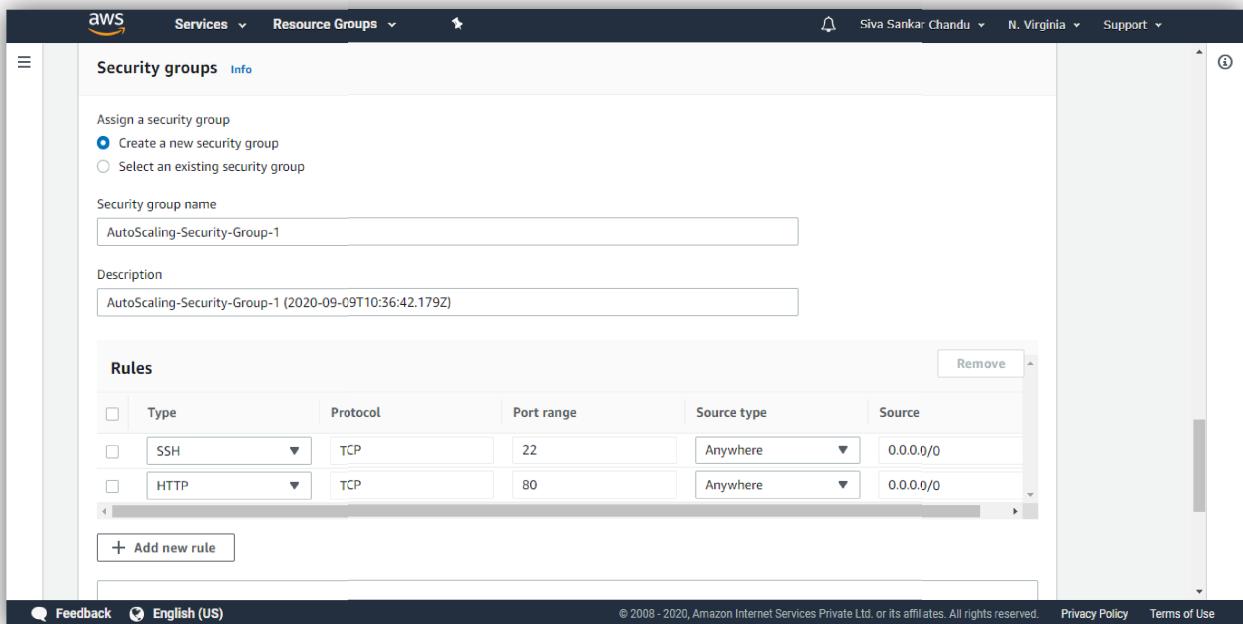


Working with EC2, ELB, ASG and CloudWatch

- Give code to user data through the text and Scroll down.

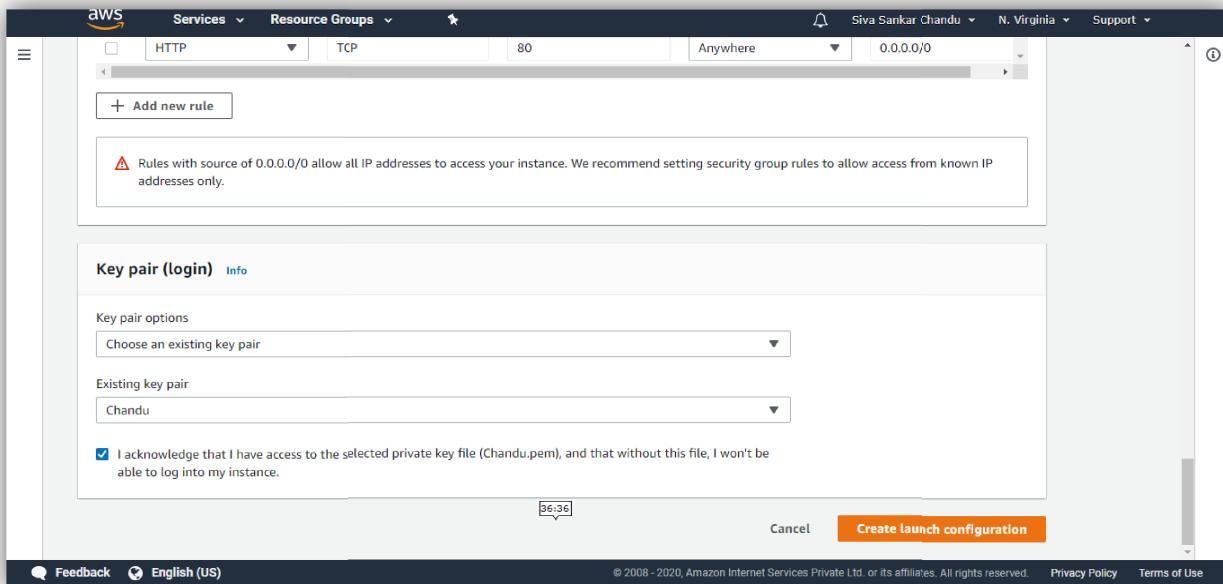


- Add HTTP Rule and Keep source as Anywhere and Scroll down.

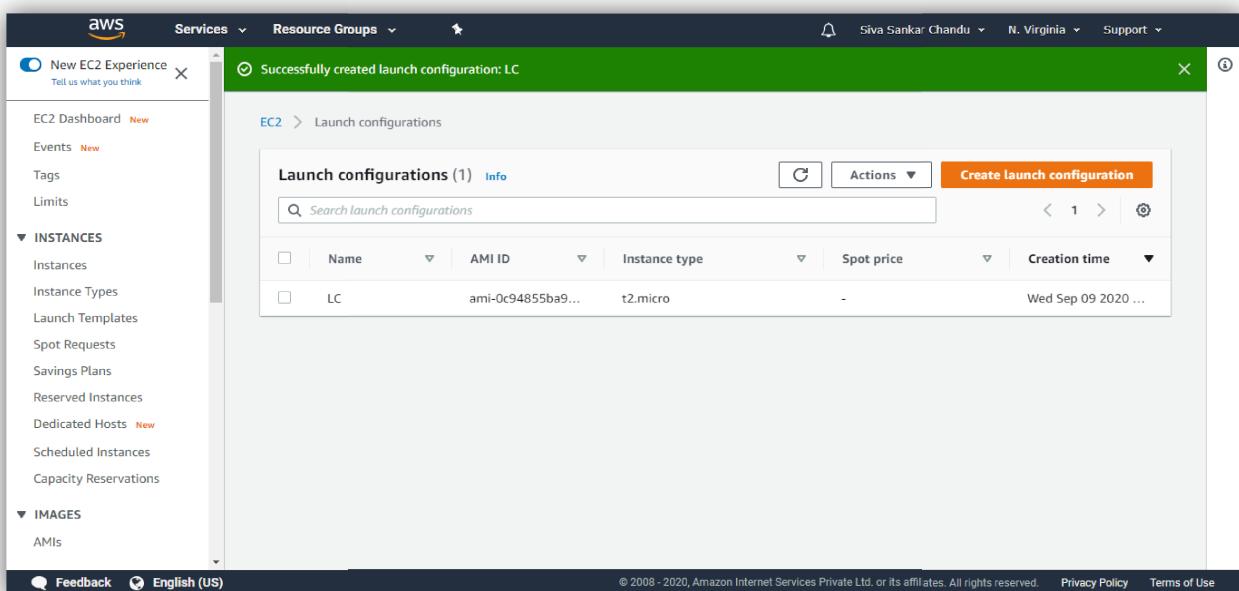


Working with EC2, ELB, ASG and CloudWatch

➤ Select Key Pair and Click Launch Configuration

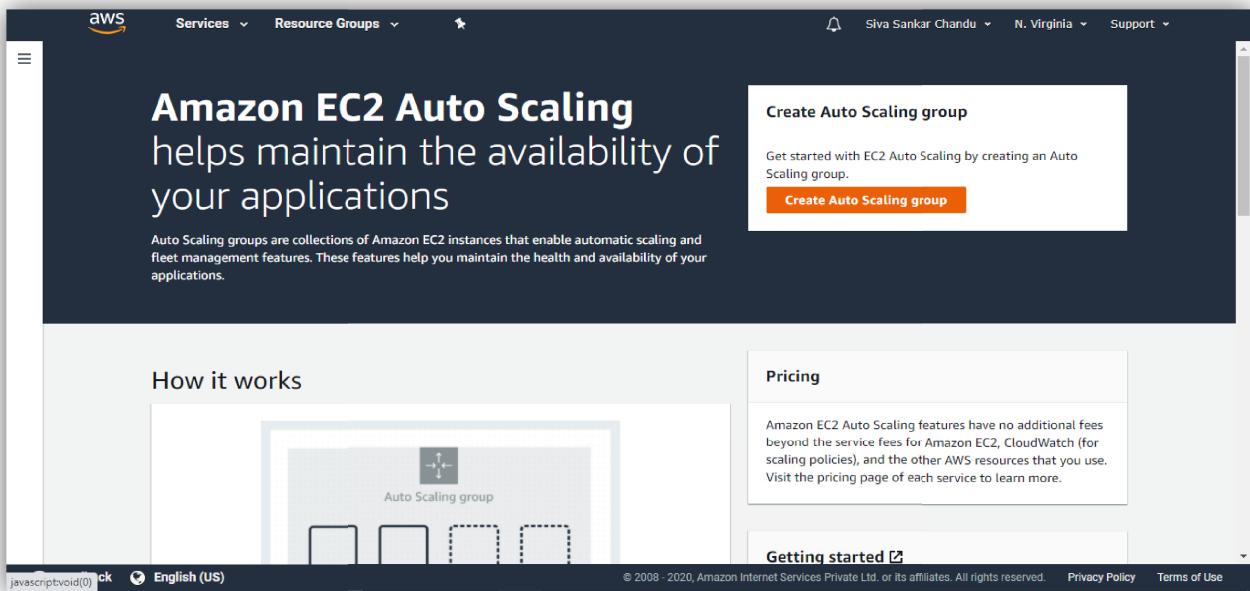


➤ Successfully created Launch Configuration: LC



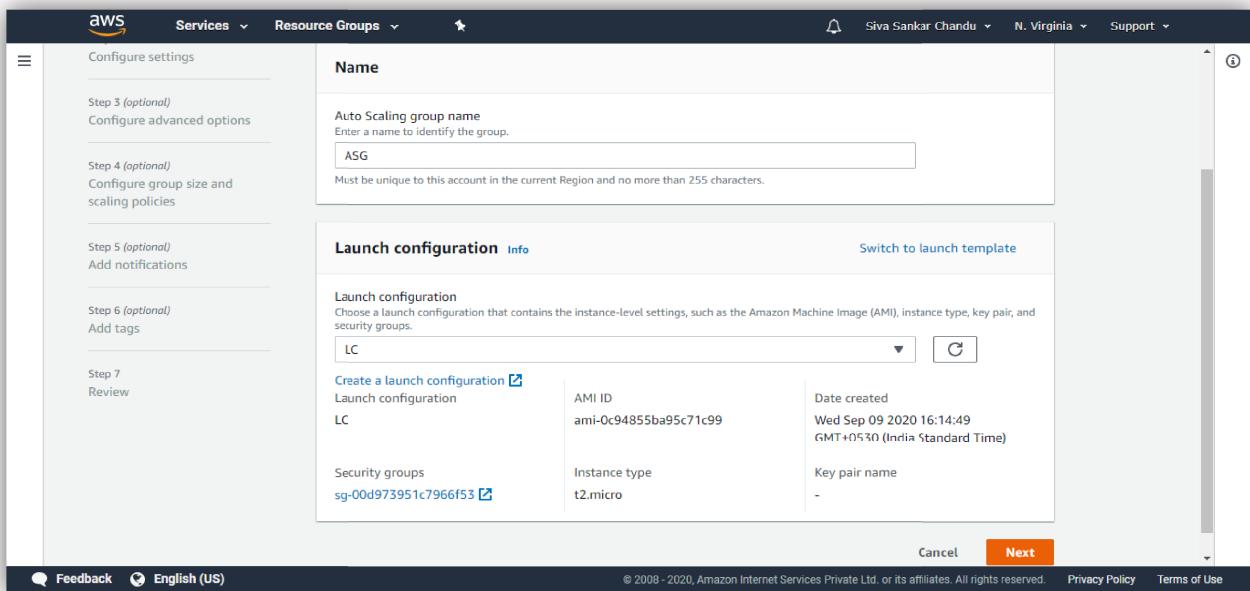
Working with EC2, ELB, ASG and CloudWatch

- Click on Creating Auto Scaling group (ASG) to create Auto Scaling group.



The screenshot shows the Amazon EC2 Auto Scaling homepage. The main heading is "Amazon EC2 Auto Scaling helps maintain the availability of your applications". Below it, a sub-section titled "How it works" features a diagram of an "Auto Scaling group" containing several EC2 instances. To the right, there are sections for "Pricing" and "Getting started". A prominent orange button labeled "Create Auto Scaling group" is located in the top right corner of the main content area.

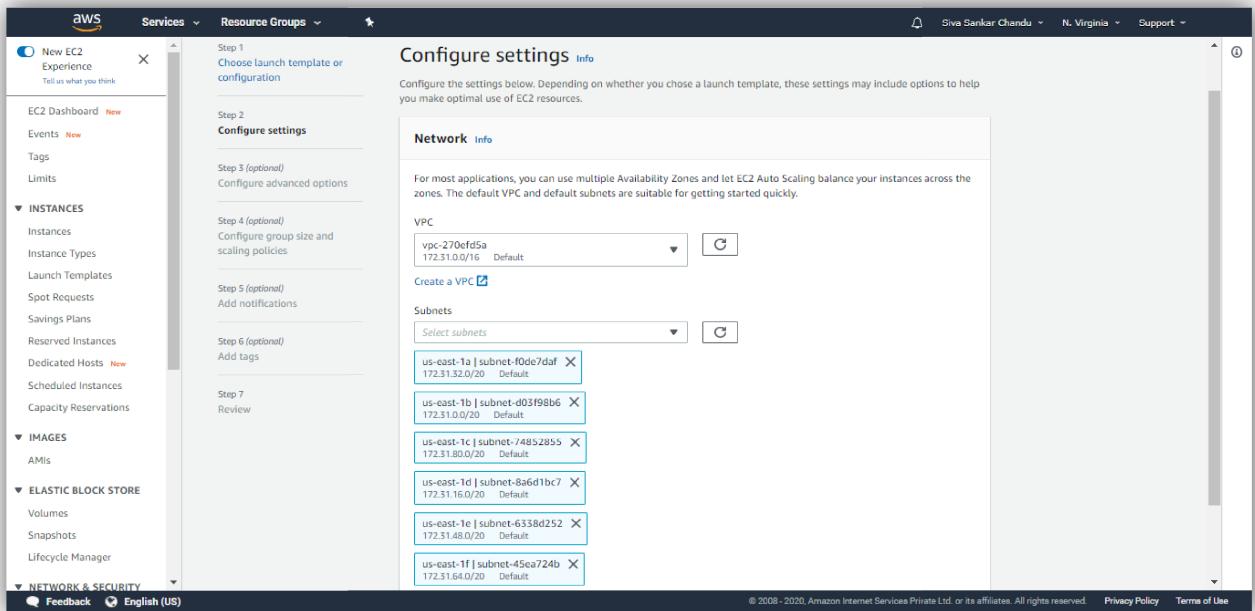
- Give name to the Auto Scaling Group and Switch to Launch Configuration then select Launch Configuration (LC) i.e., created in previous step. Click Next.



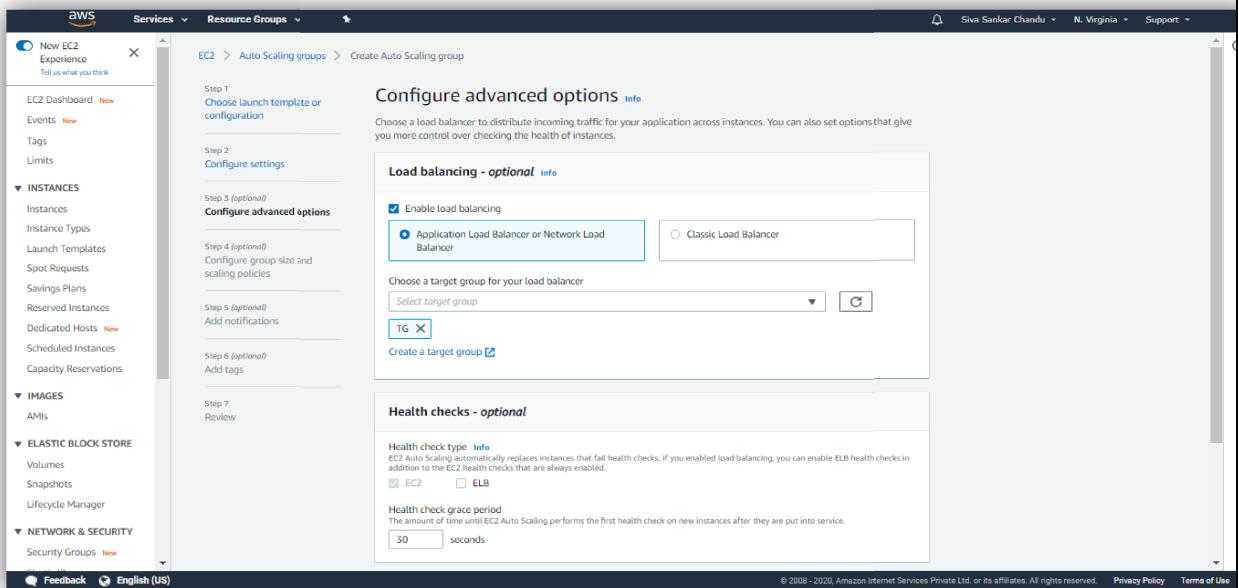
The screenshot shows the "Create Auto Scaling group" wizard at Step 3. On the left sidebar, options like "Configure settings", "Step 3 (optional)", "Step 4 (optional)", "Step 5 (optional)", "Step 6 (optional)", and "Step 7 Review" are listed. The main panel shows the "Launch configuration" section. It includes a dropdown menu set to "LC", a "Create a launch configuration" link, and a table with columns for "Launch configuration" (set to "LC"), "AMI ID" (set to "ami-0c94855ba95c71c99"), "Date created" (set to "Wed Sep 09 2020 16:14:49 GMT+0530 (India Standard Time)"), "Security groups" (set to "sg-00d973951c7966f53"), "Instance type" (set to "t2.micro"), and "Key pair name" (set to "-"). At the bottom right, there are "Cancel" and "Next" buttons.

Working with EC2, ELB, ASG and CloudWatch

- Add Subnets to Auto Scaling Group (Select all subnets) and Click Next.



- Select Application Load Balancer and Choose TG Target Group i.e., created by Previous Step. Click Next.



Working with EC2, ELB, ASG and CloudWatch

- Select Desired Capacity, Minimum Capacity, Maximum Capacity based on requirement and Scroll down for more options

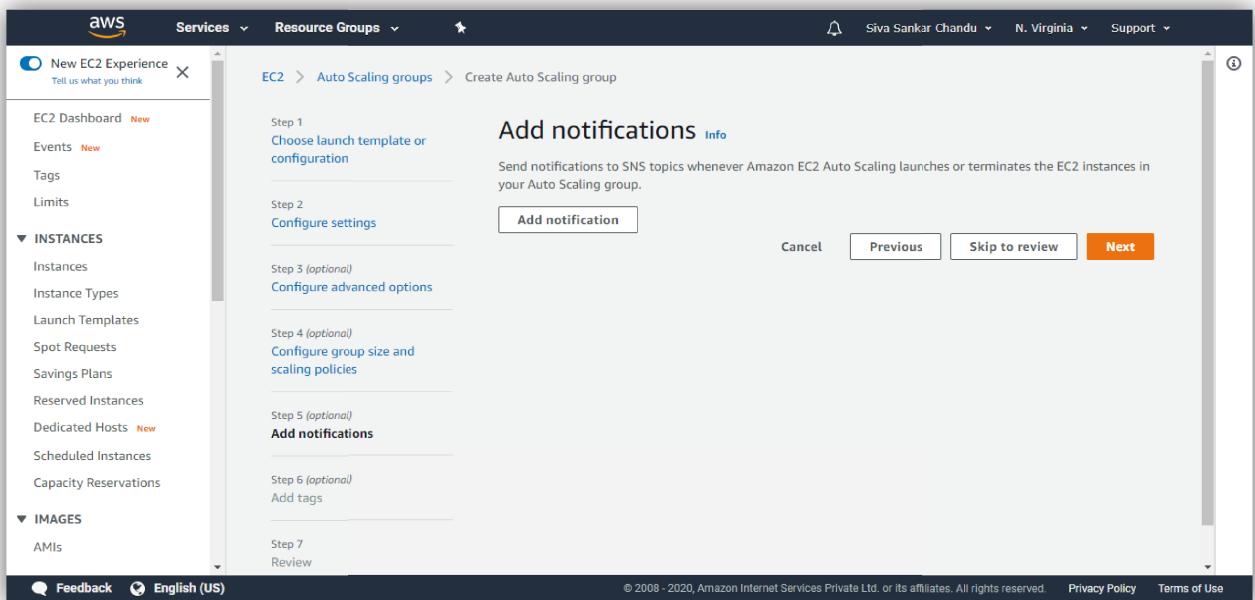
The screenshot shows the AWS Auto Scaling Groups 'Create Auto Scaling group' wizard. The current step is Step 4 (optional) titled 'Configure group size and scaling policies'. The page header says 'Configure group size and scaling policies' with an 'Info' link. Below it, a note states: 'Set the desired, minimum, and maximum capacity of your Auto Scaling group. You can optionally add a scaling policy to dynamically scale the number of instances in the group.' The 'Group size - optional' section contains fields for 'Desired capacity' (set to 2), 'Minimum capacity' (set to 2), and 'Maximum capacity' (set to 5). A note below says: 'Specify the size of the Auto Scaling group by changing the desired capacity. You can also specify minimum and maximum capacity limits. Your desired capacity must be within the limit range.' The sidebar on the left lists various EC2 services like Instances, Images, and Events.

- Edit Scaling Policy Name and Click Next.

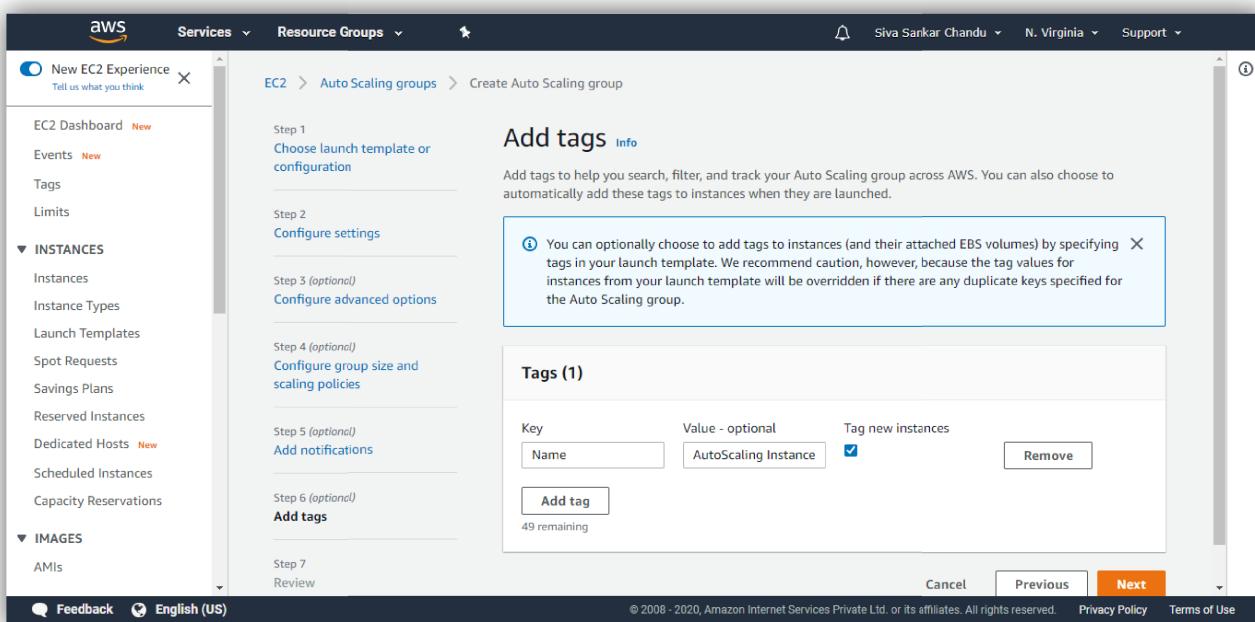
The screenshot shows the AWS Auto Scaling Groups 'Create Auto Scaling group' wizard. The current step is Step 7: Review. The page header says 'Scaling policies - optional'. The note below says: 'Choose whether to use a scaling policy to dynamically resize your Auto Scaling group to meet changes in demand.' The 'Target tracking scaling policy' option is selected, with a note: 'Choose a desired outcome and leave it to the scaling policy to add and remove capacity as needed to achieve that outcome.' The 'Scaling policy name' field is set to 'Target Tracking Policy1'. The 'Metric type' dropdown is set to 'Average CPU utilization'. The 'Target value' input is set to 80. The 'Instances need' input is set to 30, with a note: 'seconds warm up before including in metric'. A checkbox 'Disable scale in to create only a scale-out policy' is unchecked. The sidebar on the left lists various EC2 services like Instances, Images, and Events.

Working with EC2, ELB, ASG and CloudWatch

- Keep everything default and Click Next.



- Add Name Tag and Click next.



Working with EC2, ELB, ASG and CloudWatch

➤ Review the details

The screenshot shows the AWS Auto Scaling Groups creation wizard. The left sidebar shows navigation links for EC2 Dashboard, Events, Tags, Limits, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, and Images. The main area is titled 'Review' and shows the following steps:

- Step 1: Choose launch template or configuration (selected).
 - Group details:
 - Auto Scaling group name: ASG
 - Launch configuration: LC
- Step 2: Configure settings.
 - Network:
 - Network
 - VPC
- Step 3 (optional): Configure advanced options.
- Step 4 (optional): Configure group size and scaling policies.
- Step 5 (optional): Add notifications.
- Step 6 (optional): Add tags.
- Step 7: Review.

➤ Click on Create Auto Scaling Group to Create ASG.

The screenshot shows the AWS Auto Scaling Groups creation wizard. The left sidebar shows navigation links for EC2 Dashboard, Events, Tags, Limits, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, and Images. The main area is titled 'Step 5: Add notifications' and shows the following sections:

- Instance scale-in protection:
 - Enable instance protection from scale in
- Step 5: Add notifications (selected).
 - Notifications:
 - No notifications
- Step 6: Add tags.
 - Tags (1):

Key	Value	Tag new instances
Name	AutoScaling Instance	Yes

A 'Create Auto Scaling group' button is located at the bottom right of the wizard.

Working with EC2, ELB, ASG and CloudWatch

- Created Auto Scaling Group ASG successfully.

The screenshot shows the AWS Auto Scaling Groups page. At the top, there is a blue banner with the text "Lower the Possibility of Spot Interruptions" and "Capacity Optimized allocation identifies the deepest Spot Instance pools, possibly lowering the chance of Spot interruptions." Below the banner, a green notification bar says "ASG, 1 Scaling policy created successfully". The main table has columns for Name, Launch template/configuration, Instances, Status, Desired capacity, and Min. The ASG row shows 0 instances, status as "Updating capacity", desired capacity of 2, and min of 2. The left sidebar includes sections for Instances (Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations) and Images (AMIs).

- All instances shown in below image.

The screenshot shows the AWS Instances page. The main table lists three instances: "AutoScaling..." (running, t2.micro, us-east-1c), "Manual Insta..." (running, t2.micro, us-east-1c), and "AutoScaling..." (running, t2.micro, us-east-1e). The left sidebar includes sections for Instances (Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations) and Images (AMIs). A detailed view of the "Manual Insta..." instance is shown at the bottom, displaying its ID (i-09aeb92557c5c9a5f), Public DNS (ec2-3-82-216-99.compute-1.amazonaws.com), and various status metrics like Instance ID, Instance state, and Instance type.

Working with EC2, ELB, ASG and CloudWatch

- Every Instance is in healthy state, then now check ELB through the DNS Name

The screenshot shows the AWS EC2 Target Groups interface. On the left, there's a sidebar with options like EC2 Dashboard, Events, Tags, Limits, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, and Images. The main area is titled 'Basic configuration' for target group 'arn:aws:elasticloadbalancing:us-east-1:854231420140:targetgroup/TG/e36d739e3e4b526e'. It shows 'Target type: instance', 'Protocol: Port HTTP : 80', 'VPC: vpc-270efd5a', and 'Load balancer: ELB'. Below this, tabs for Group details, Targets (selected), Monitoring, and Tags are visible. Under 'Registered targets (3)', three instances are listed:

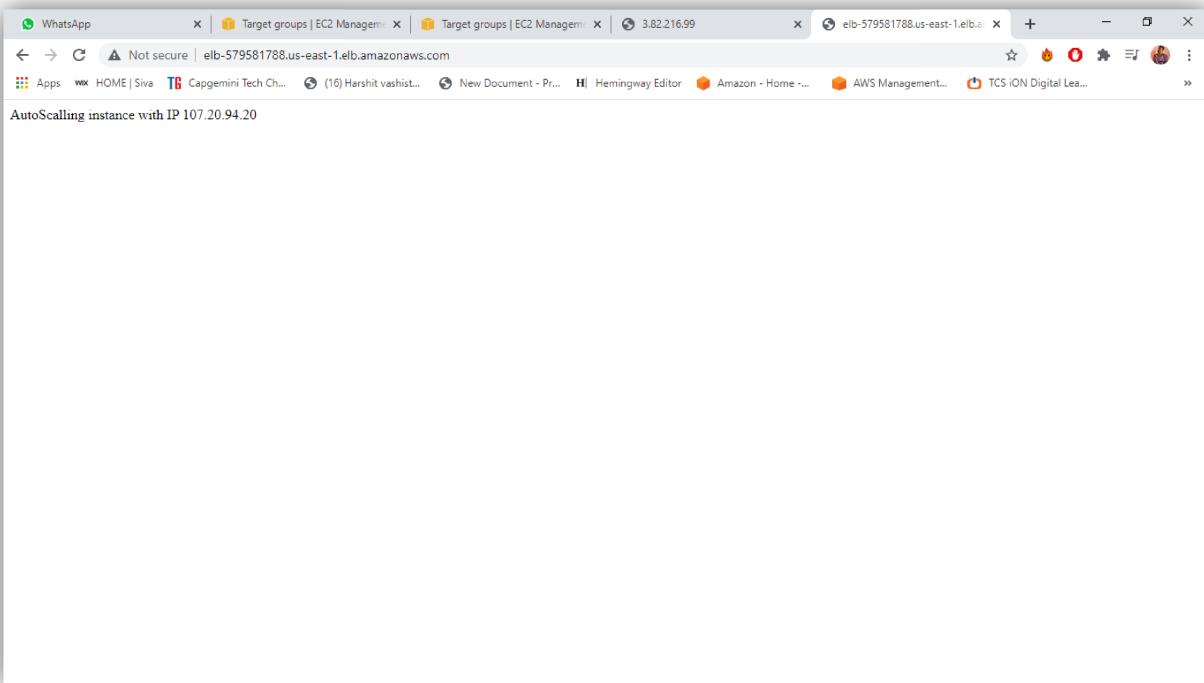
Instance ID	Name	Port	Zone	Status
i-0145eb65442afc01	AutoScaling Instance	80	us-east-1e	healthy
i-04c079726db4d09d	AutoScaling Instance	80	us-east-1c	healthy
i-09aeb92557c5c9a5f	Manual Instance	80	us-east-1c	healthy

- First time Manual Instance is Loaded.

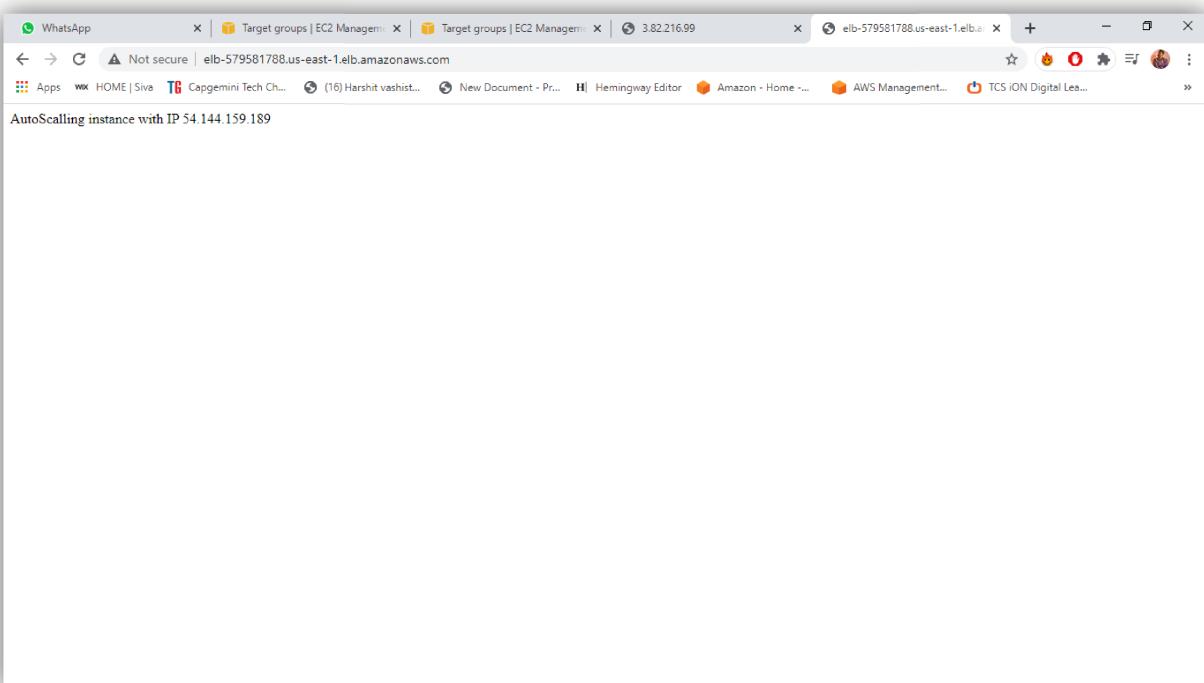
The screenshot shows a web browser window with the URL 'elb-579581788.us-east-1.elb.amazonaws.com'. The page content is 'Manual instance with IP 3.82.216.99'. The browser toolbar includes WhatsApp, Target groups | EC2 Manager, Target groups | EC2 Manager, 3.82.216.99, elb-579581788.us-east-1.elb.amazonaws.com, Apps, HOME | Siva, Capgemini Tech Ch..., New Document - Pr..., Hemingway Editor, Amazon - Home ..., AWS Management..., TCS iON Digital Le..., and TCS iON Digital Le...'. The address bar shows 'Not secure | elb-579581788.us-east-1.elb.amazonaws.com'.

Working with EC2, ELB, ASG and CloudWatch

- After refreshing the page, Auto Scaling Instance Loaded.



- Again refreshing the page, Another Auto Scaling Instance Loaded, Successfully.



Working with EC2, ELB, ASG and CloudWatch

- All Instances are running state.

This screenshot shows the AWS EC2 Instances page. The left sidebar includes options like EC2 Dashboard, Events, Tags, Limits, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, and Images. The main content area displays a table of instances. One instance is selected, showing its details in a modal window. The instance is an AutoScaling instance with the ID i-0145e8b65442afc01, running in us-east-1e, with a Public DNS of ec2-54-144-159-189.compute-1.amazonaws.com. The modal also shows the instance state as running, type t2.micro, and public IP 54.144.159.189.

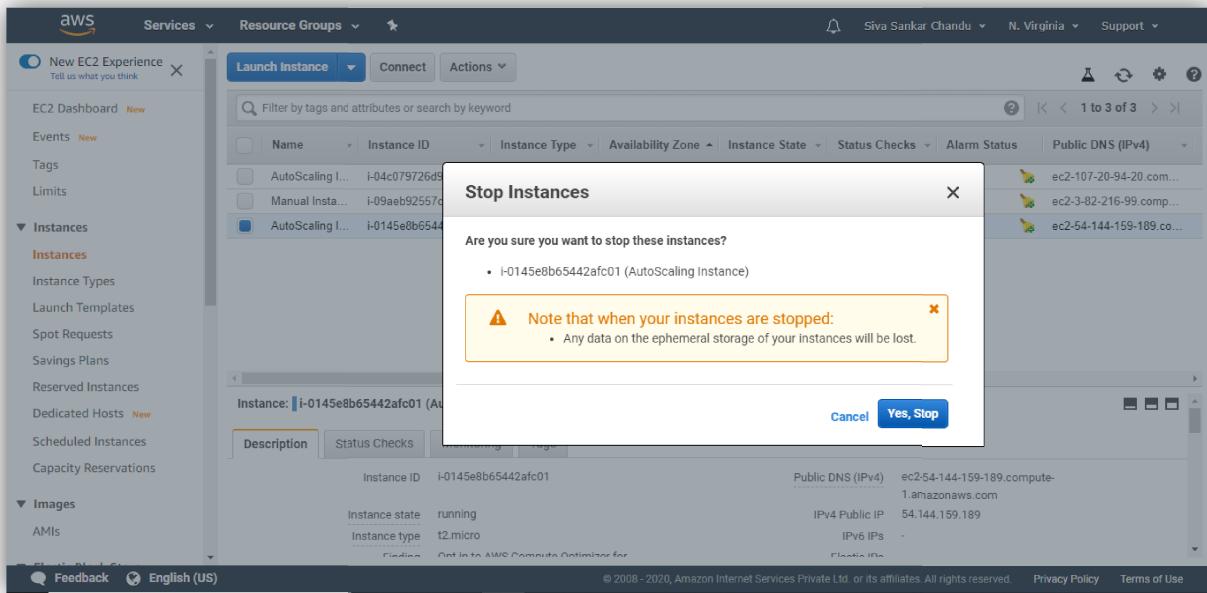
- Now, if we stop one Auto Scaling Instance then automatically created a new Instance, created instance replaces the stopped instance state and it works like same as previous instance. Whenever new Instance is created, stopped Instance is terminated automatically. Shown in below Screenshots i.e., Screenshot 1, Screenshot 2, Screenshot 3, Screenshot 4, and Screenshot 5.

Screenshot 1

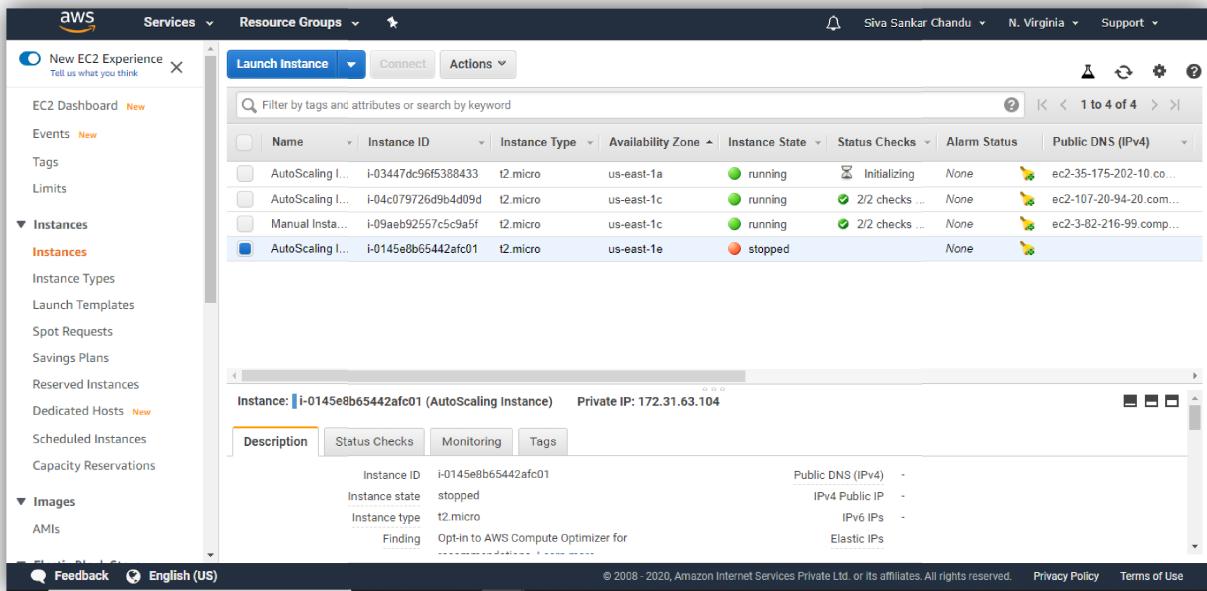
This screenshot shows the same AWS EC2 Instances page as before, but with a context menu open over the selected instance. The menu is titled 'Actions' and includes options: Connect, Get Windows Password, Create Template From Instance, Launch More Like This, Instance State (with sub-options Start, Stop, Stop - Hibernate, Reboot, Terminate), Instance Settings, Image, Networking, and CloudWatch Monitoring. The 'Stop' option is highlighted.

Working with EC2, ELB, ASG and CloudWatch

Screenshot 2



Screenshot 3



Working with EC2, ELB, ASG and CloudWatch

Screenshot 4

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with links like EC2 Dashboard, Events, Tags, Limits, Instances (selected), Images, and AMIs. The main area displays a table of instances with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS (IPv4). There are four instances listed:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
AutoScaling I...	i-03447dc96f5388433	t2.micro	us-east-1a	running	2/2 checks ...	None	ec2-35-175-202-10.co...
AutoScaling I...	i-04c079726d9b4d09d	t2.micro	us-east-1c	running	2/2 checks ...	None	ec2-107-20-94-20.com...
Manual Insta...	i-09aeb92557c5c9a5f	t2.micro	us-east-1c	running	2/2 checks ...	None	ec2-3-82-216-99.comp...
AutoScaling I...	i-0145e0b65442afc01	t2.micro	us-east-1e	terminated		None	

A message at the bottom says "Select an instance above".

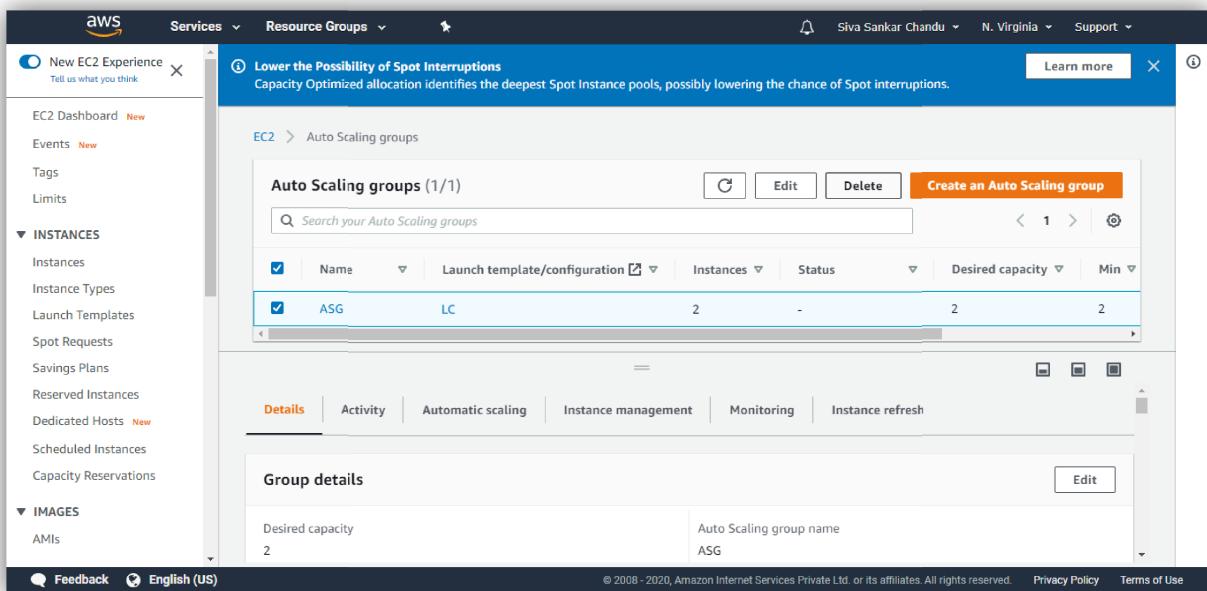
Screenshot 5

The screenshot shows a web browser window with the address bar displaying "elb-579581788.us-east-1.elb.amazonaws.com". The page content is a plain white space with the text "AutoScaling instance with IP 35.175.202.10".

Working with EC2, ELB, ASG and CloudWatch

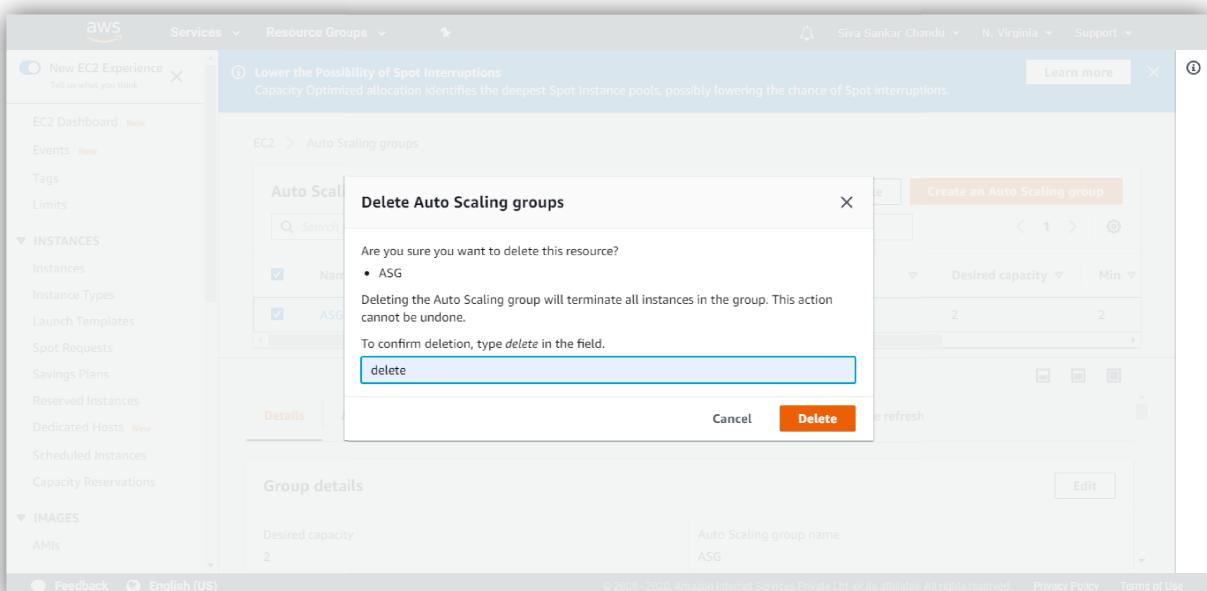
Step 4 – Cleaning process.

- First select the Auto Scaling Group ASG then click on Delete.



The screenshot shows the AWS EC2 Auto Scaling Groups page. On the left, there's a sidebar with options like Instances, Launch Templates, and Auto Scaling groups. The main area shows a table titled "Auto Scaling groups (1/1)". A single row is listed: "ASG" with a "LC" launch configuration, 2 instances, and a status of "-". Below the table, there are tabs for Details, Activity, Automatic scaling, Instance management, Monitoring, and Instance refresh. Under the Details tab, there's a "Group details" section with fields for Desired capacity (set to 2) and Auto Scaling group name (set to ASG). At the top right of the main area, there are buttons for Edit, Delete, and Create an Auto Scaling group. The "Delete" button is highlighted.

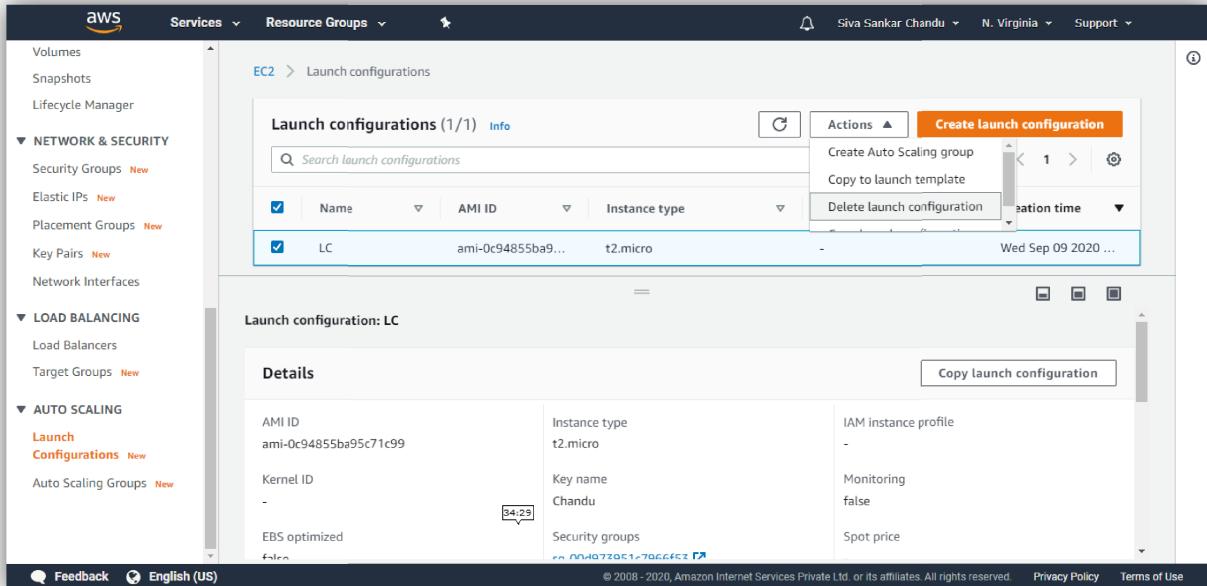
- Type the delete word on text box then click on delete button. After clicking Delete button Auto Scaling Group was deleted.



The screenshot shows the same AWS EC2 Auto Scaling Groups page as before, but with a modal dialog box in the foreground. The dialog is titled "Delete Auto Scaling groups" and contains the message: "Are you sure you want to delete this resource? • ASG". It explains that deleting the Auto Scaling group will terminate all instances in the group and that the action cannot be undone. Below this, it says "To confirm deletion, type delete in the field." and has a text input field containing "delete". At the bottom of the dialog are "Cancel" and "Delete" buttons. The background of the page is dimmed, and the "ASG" row in the table is still visible.

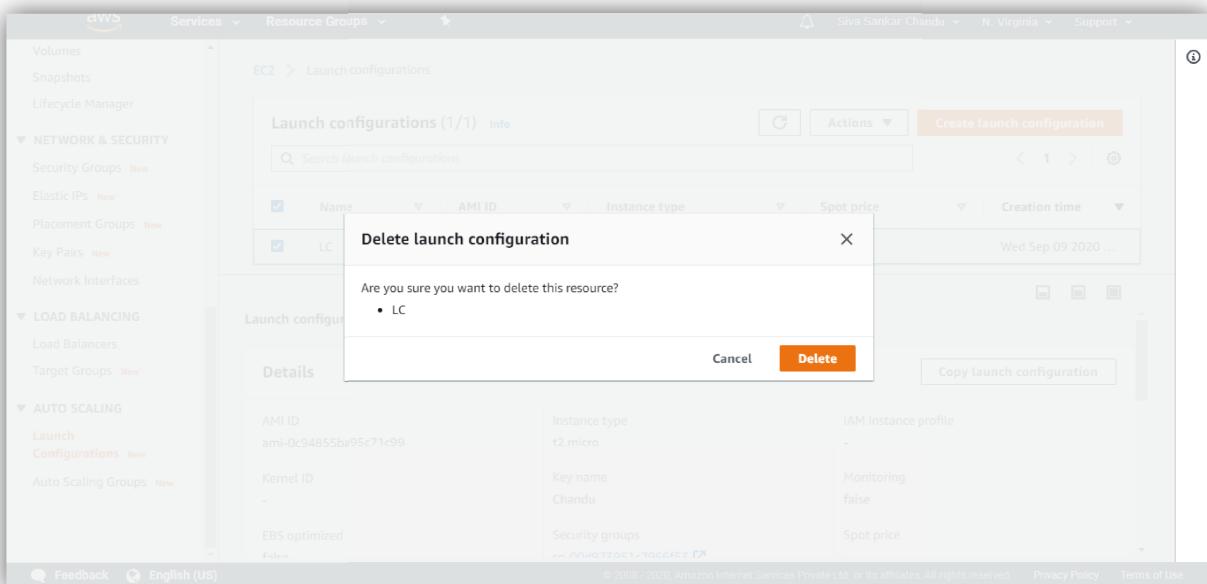
Working with EC2, ELB, ASG and CloudWatch

- After deleting Auto Scaling Group Delete the Launch Configuration. Select Launch Configuration LC, and Click on Actions -> Delete Launch Configuration.



The screenshot shows the AWS EC2 console with the 'Launch Configurations' page open. A single launch configuration named 'LC' is listed. The 'Actions' menu is open, and the 'Delete launch configuration' option is highlighted. The 'Details' section below shows the configuration settings: AMI ID (ami-0c94855ba95c71c99), Instance type (t2.micro), Key name (Chandu), Security groups (auto-04072021-706ccfc2), and IAM instance profile (not specified). A timestamp indicates the configuration was created on Wednesday, September 9, 2020.

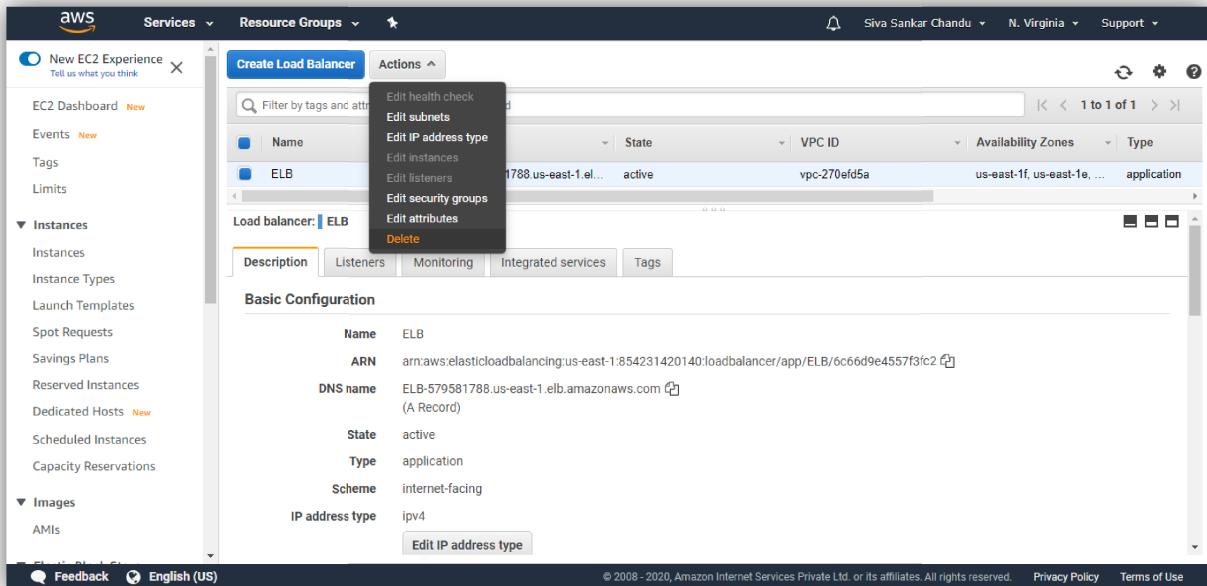
- Click on Delete button to delete the Launch Configuration.



The screenshot shows the same 'Launch Configurations' page after the 'Delete launch configuration' action was initiated. A confirmation dialog box is overlaid on the page, asking 'Are you sure you want to delete this resource?'. The radio button next to 'LC' is selected. At the bottom right of the dialog are 'Cancel' and 'Delete' buttons. The background details of the launch configuration remain visible.

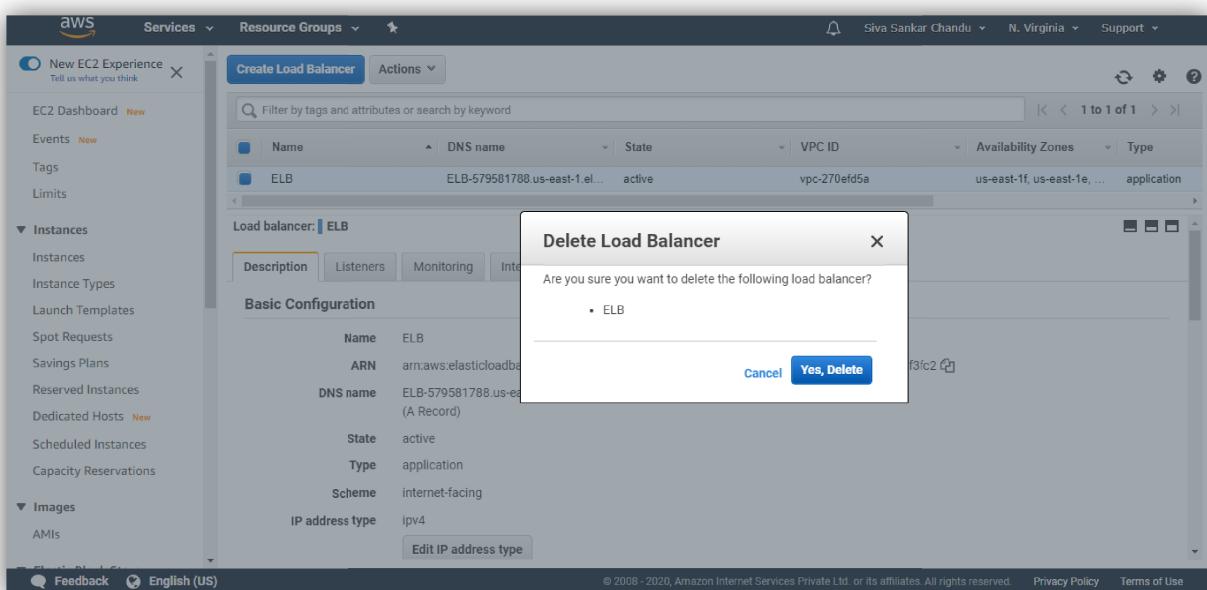
Working with EC2, ELB, ASG and CloudWatch

- After deleting Launch Configuration, Now delete the Load Balancer. Select the Load Balancer ELB, and Click on Action -> Delete to delete Load Balancer.



The screenshot shows the AWS Elastic Load Balancing (ELB) console. On the left, there's a navigation sidebar with options like EC2 Dashboard, Events, Tags, Limits, Instances, and Images. The main area displays a table of load balancers. One row is selected, showing details for an ELB named 'ELB'. A context menu is open over this row, with 'Delete' being the last item in the list. Below the table, there are tabs for Description, Listeners, Monitoring, Integrated services, and Tags. Under 'Basic Configuration', it shows the ELB's Name, ARN, DNS name (ELB-579581788.us-east-1.elb.amazonaws.com), State (active), Type (application), Scheme (internet-facing), and IP address type (ipv4). At the bottom right of the main area, there are links for Feedback, English (US), Privacy Policy, and Terms of Use.

- Click on Yes, Delete button to confirm the deletion.



This screenshot shows the same AWS ELB console as the previous one, but with a modal dialog box in the foreground. The dialog is titled 'Delete Load Balancer' and contains the message 'Are you sure you want to delete the following load balancer?'. Below this message is a list with a single item: 'ELB'. At the bottom right of the dialog are two buttons: 'Cancel' and 'Yes, Delete'. The 'Yes, Delete' button is highlighted with a blue border. The background of the main interface is dimmed to indicate it's not active while the dialog is open.

Working with EC2, ELB, ASG and CloudWatch

- After deleting the Load Balancer, delete Target Groups. To delete Target Groups, Select the Target Group TG, and Click on Action -> Delete.

The screenshot shows the AWS EC2 Target groups page. On the left, there's a navigation menu with options like Elastic Block Store, Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs), Load Balancing (Load Balancers, Target Groups), and Auto Scaling (Launch Configurations, Auto Scaling Groups). The 'Target Groups' section is selected. The main area displays a table titled 'Target groups (1/1)'. It has columns for Name, ARN, Port, Protocol, Target type, Load balancer, and VPC ID. A single row is shown for 'TG' with the ARN 'arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/TG'. There are 'Actions' and 'Delete' buttons at the top right of the table. At the bottom, there are links for Feedback, English (US), Privacy Policy, and Terms of Use.

- Confirm the deletion through the click Yes, delete button.

The screenshot shows the same EC2 Target groups page as before, but with a modal dialog box in the foreground. The dialog is titled 'Delete target group?' and contains the message 'You cannot undo this action.' Below that, it says 'Deleting a target group deletes the group; the individual resources registered to the target group do not get deleted as a result of this action.' There is a question 'Are you sure you want to delete this target group?' followed by a radio button selected for 'TG'. At the bottom of the dialog are 'Cancel' and 'Yes, delete' buttons. The background of the page is dimmed.

Working with EC2, ELB, ASG and CloudWatch

- Successfully deleted Target Group: TG.

The screenshot shows the AWS CloudWatch Metrics interface. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, a notification bell, user name 'Siva Sankar Chandu', region 'N. Virginia', and support links. On the left, a sidebar lists various AWS services: Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs), Load Balancing (Load Balancers, Target Groups), Auto Scaling (Launch Configurations, Auto Scaling Groups). The 'Target Groups' section is currently selected. The main content area displays a green banner with the message 'Successfully deleted target group: TG'. Below the banner, the heading 'Target groups (0)' is followed by a table with columns: Name, ARN, Port, Protocol, Target type, Load balancer, and VPC ID. A note below the table states 'No Target groups' and 'No resources to display'. At the bottom, there are links for Feedback, English (US) language, and copyright information: '© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.' and 'Privacy Policy Terms of Use'.

- After deleting Auto Scaling Group, Launch Configuration, Load Balancer and Target Group, automatically Auto Scaling Instances are terminated. Now delete the EC2 Instance.

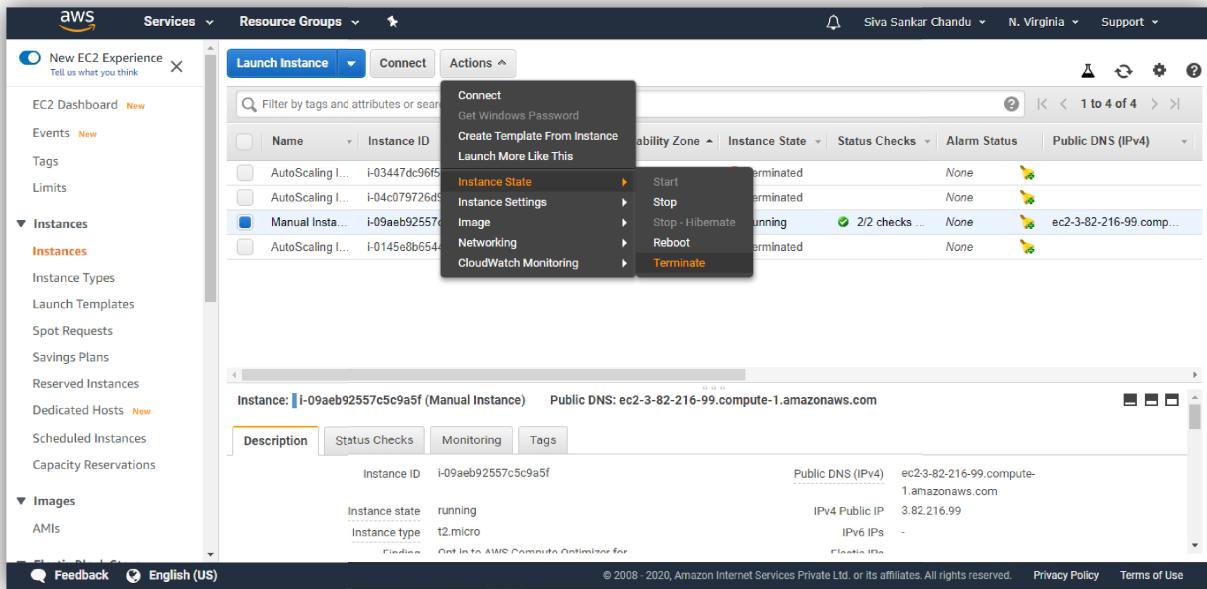
The screenshot shows the AWS EC2 Instances page. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, a notification bell, user name 'Siva Sankar Chandu', region 'N. Virginia', and support links. On the left, a sidebar lists EC2 Dashboard, Events, Tags, Limits, Instances (Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations), and Images (AMIs). The 'Instances' section is currently selected. The main content area shows a table of terminated instances. The table has columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS (IPv4). The instances listed are:

- AutoScaling I... i-03447dc96f5388433 t2.micro us-east-1a terminated None ec2-3-82-216-99.comp...
- AutoScaling I... i-04c079726d9b4d09d t2.micro us-east-1c terminated None ec2-3-82-216-99.comp...
- Manual Insta... i-09aeb92557c5c9a5f t2.micro us-east-1c running 2/2 checks ... None ec2-3-82-216-99.comp...
- AutoScaling I... i-0145e8b65442afc01 t2.micro us-east-1e terminated None ec2-3-82-216-99.comp...

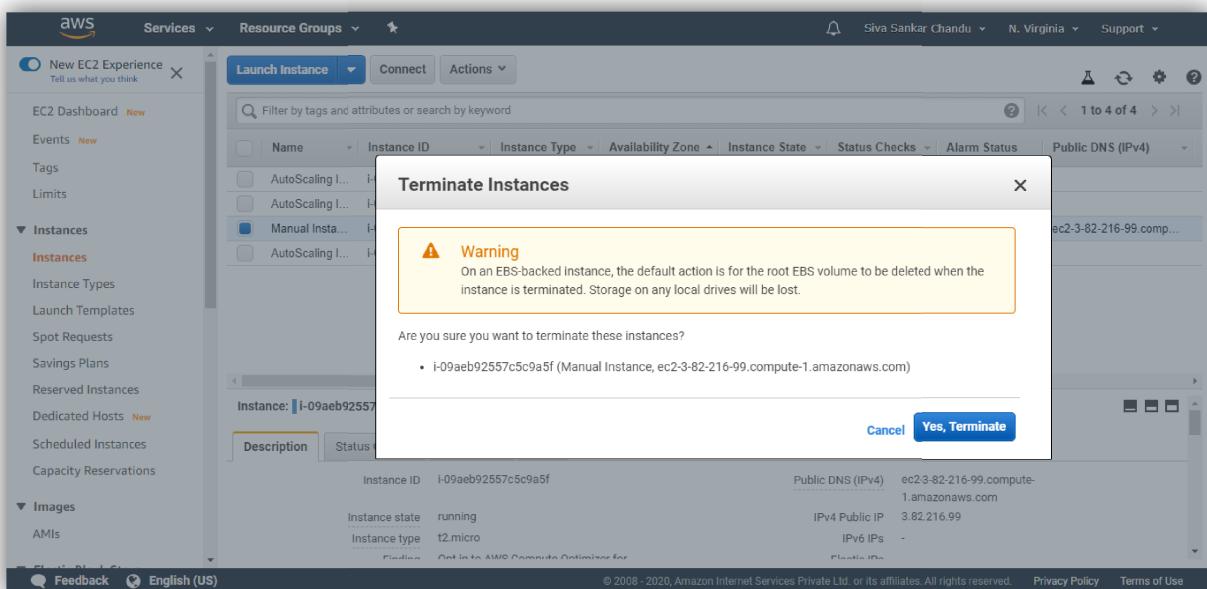
A note at the bottom of the table says 'Select an instance above'.

Working with EC2, ELB, ASG and CloudWatch

- To delete EC2 Instance, Select Instance then Click on Instance State -> Terminate it.



- Confirm the termination by clicking Yes, Terminate button.



Working with EC2, ELB, ASG and CloudWatch

- All the Instances are terminated successfully. Cleaning process is completed.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with options like EC2 Dashboard, Events, Tags, Limits, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, and Images (AMIs). The main content area displays a table of instances. The table has columns for Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS (IPv4). There are four rows in the table, all of which are marked as 'terminated'. The first three rows are from AutoScaling groups, and the fourth row is a 'Manual Instance'. The instance details for the manual instance are shown in a modal window at the bottom. The modal shows the instance ID (i-09aeb92557c5c9a5f), state (terminated), type (t2.micro), and a note about opting-in to the AWS Compute Optimizer. It also lists Public DNS (IPv4) and Elastic IPs.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
AutoScaling I...	i-03447dc96f5388433	t2.micro	us-east-1a	terminated	None	None	-
AutoScaling I...	i-04c079726d94d09d	t2.micro	us-east-1c	terminated	None	None	-
Manual Insta...	i-09aeb92557c5c9a5f	t2.micro	us-east-1c	terminated	None	None	-
AutoScaling I...	i-0145e0b65442afc01	t2.micro	us-east-1e	terminated	None	None	-