

# Day 40 AWS EC2 Automation

Maninder Singh

*Dear Learners, In this Article we will discuss the AWS Ec2 Automation with the help of Tasks. Now let's start.*

## Automation in EC2:

*Amazon Elastic compute cloud can give secure, reliable, high-performance and cost-effective computing infrastructure to meet demanding business needs.'*

## Launch template in AWS EC2:

*A) You can make a launch template with the configuration information you need to start an instance. you can save launch parameters in launch templates so you don't have to type them in every time you start a new instance.*

*B) You can tell the Amazon Ec2 console to use a Certain launch template when you start an instance.*

## Instance Types:

*In Amazon Ec2 has a large number of instance types that are optimised for different uses. The different combinations of CPU, memory, storage and networking capacity in instance types give you the freedom to choose the right mix of resources for your apps.*

## AMI:

*An Amazon Machine image is an image that AWS supports and keeps up to date. It contains the information needed to start an instance. when you launch an instance you must choose an AMI. when you need multiple instances with the same configuration you can launch them from a single AMI.*

## Task1:

*Create a launch template with Amazon Linux 2 AMI and t2.micro instance type with jenkins and Docker setup.*

1 open the Amazon Ec2 Console.

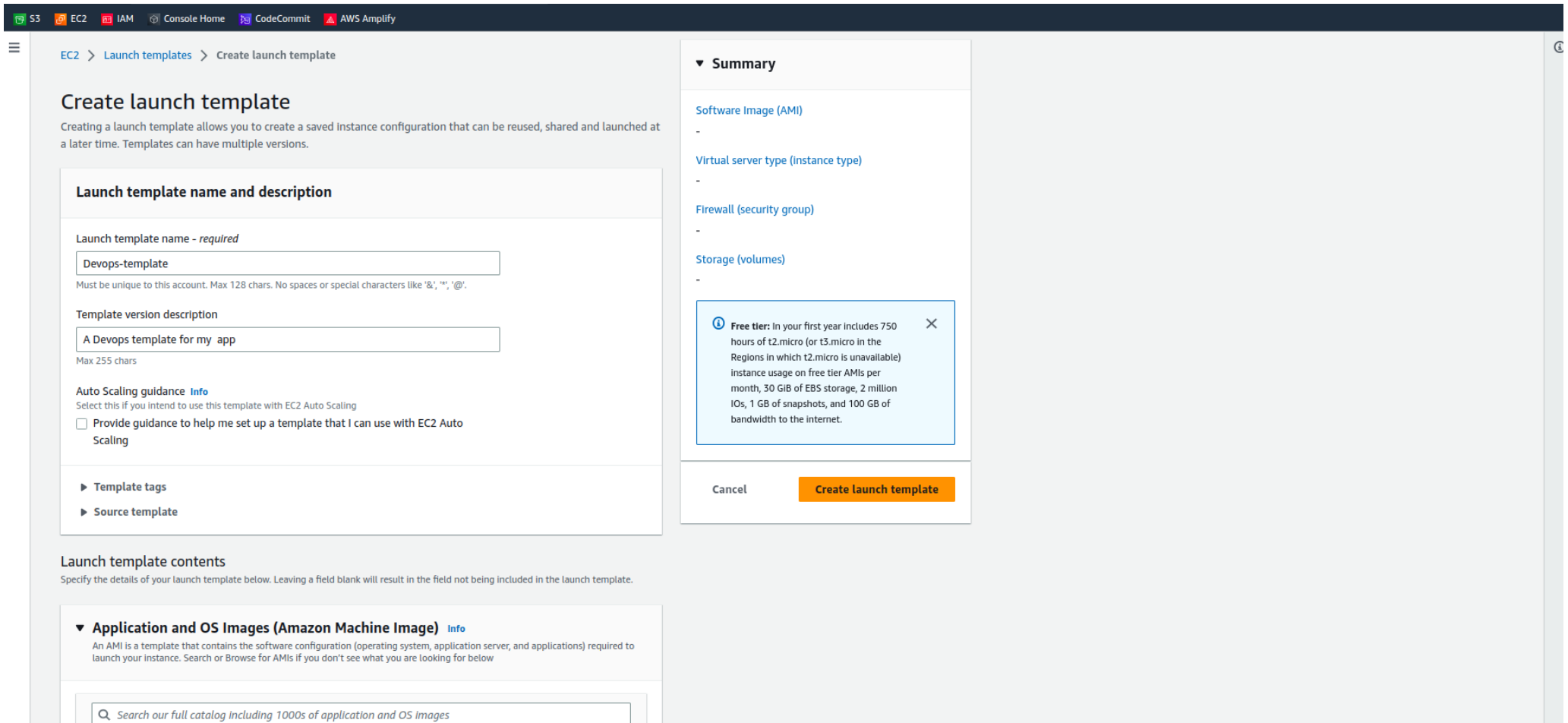
2 In the left navigation pane choose launch templates.

3 Choose "Create Launch template"

The screenshot displays the AWS Management Console interface. At the top, the navigation bar includes the AWS logo, a 'Services' menu, a search bar, and a '[Alt+S]' shortcut. Below the navigation bar, a secondary bar shows icons for S3, EC2, IAM, and other services, along with links to 'Console Home', 'CodeCommit', and 'AWS Amplify'. On the left side, a 'New EC2 Experience' banner is visible, followed by a navigation pane with categories like 'EC2 Dashboard', 'Events', 'Limits', 'Instances', 'Images', 'Elastic Block Store', and 'Network & Security'. The 'Launch Templates' link under the 'Instances' category is highlighted. The main content area features a dark header with the text 'Compute' and 'EC2 launch templates', followed by a sub-header 'Streamline, simplify and standardize instance launches'. Below this, a paragraph explains the benefits of launch templates. A 'Benefits and features' section contains three cards: 'Streamline provisioning', 'Simplify permissions', and 'Governance'. On the right side, a 'New launch template' box with a 'Create launch template' button is present, along with a 'Documentation' box containing links to 'Documentation' and 'API reference'.

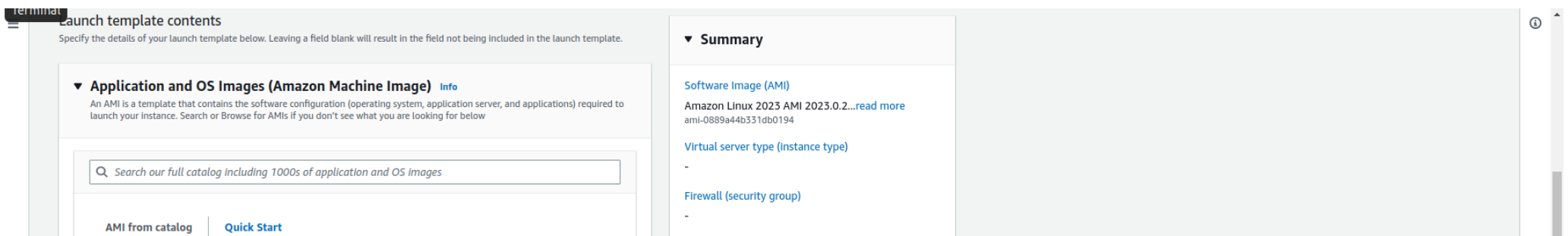
Choose "Create Launch template"

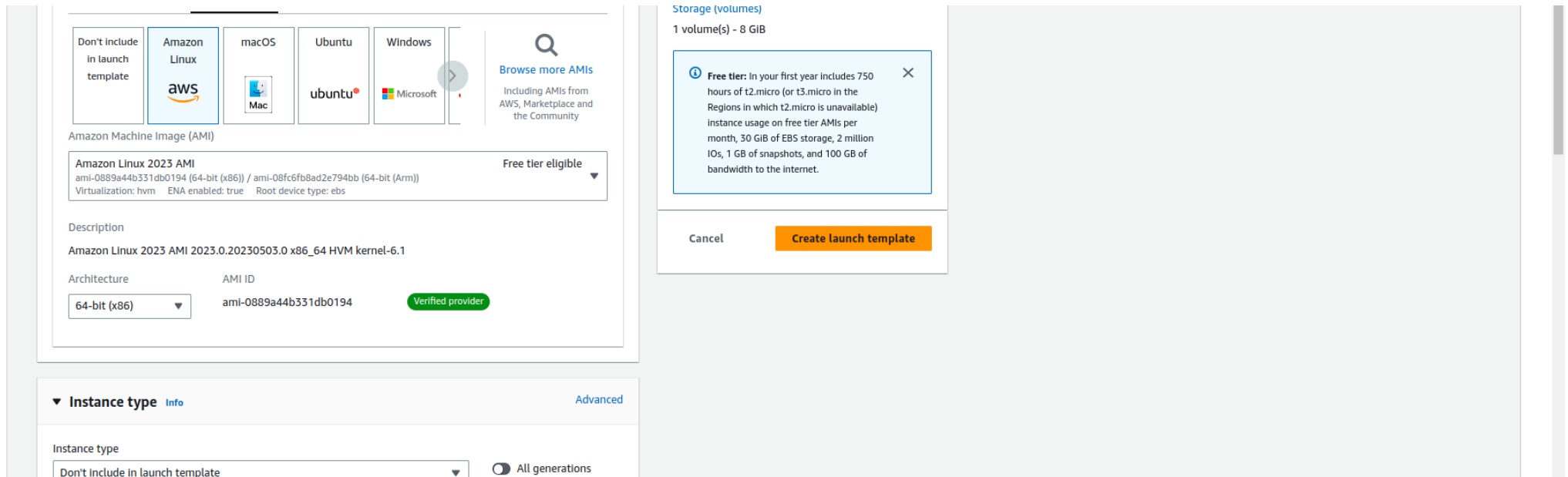
In the "Create launch template page enter a name for the launch template.



Create launch template page Diagram.

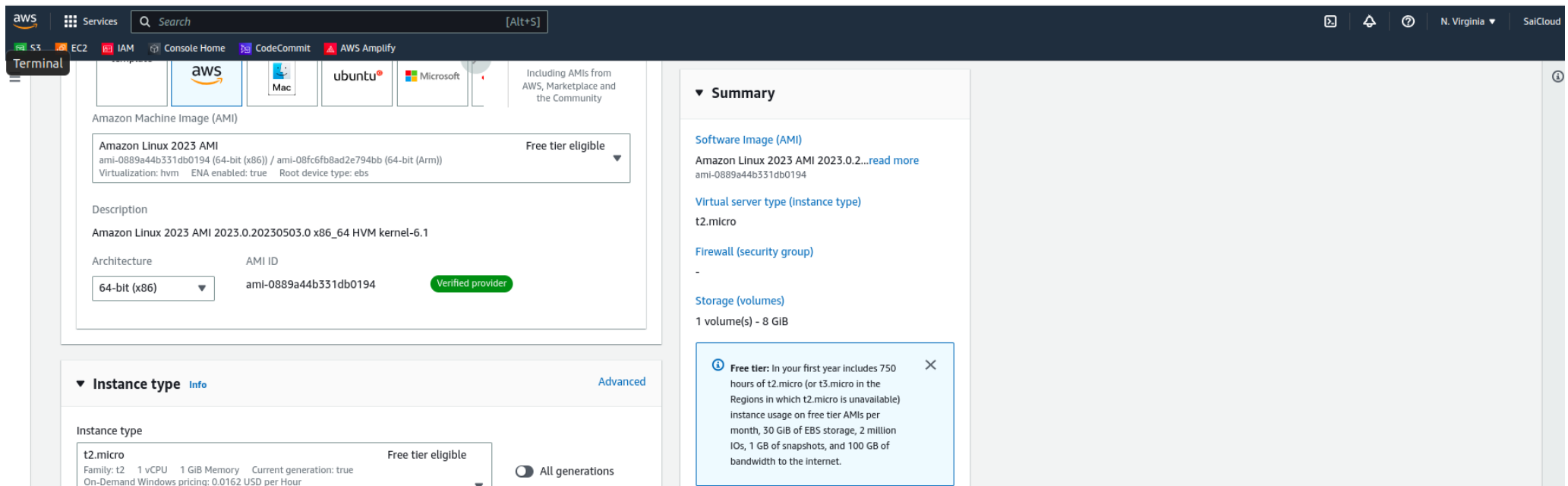
*For "Amazon Machine image" choose Amazon Linux 2 "*

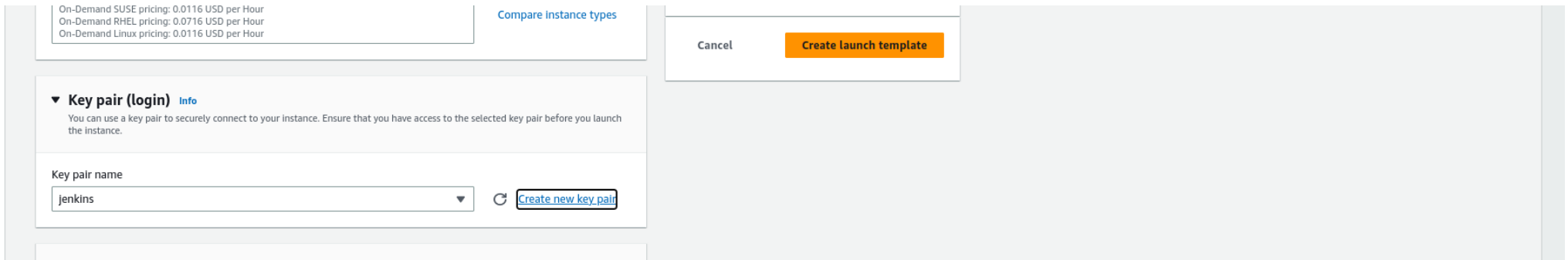




Amazon Machine image Diagram

*For Instance type choose t2.micro.*





Instance type choose t2.micro Diagram

***In the "Advance Details" section paste the user data script for installing Jenkins and Docker in the User data field.***

Advance Details" section paste the user data Diagram.

***Choose "Create launch template" below you can see template is created.***

Create launch template" Diagram.

***Create 3 Instance using Launch template there must be an option that shows number of instances to be launched can you find it.***

***To lauch 3 Instances using the launch template***

***In the Amazon EC2 console, choose "Launch instance form templates" in the left navigation pane.***

Amazon EC2 console, choose "Launch instance form template Diagram.

***Select the Launch template that you just created.***

Launch template that you just created. Diagram.

***Specify the no of instance you want to launch in the number of instances fields on right side. choose the other configuration settings as desired such as VPC subnet, security group.***

desired such as VPC subnet, security group. Diagram

***You can see three instances created from template.***

**Instances (3) Info**

Find Instance by attribute or tag (case-sensitive)

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs
<input type="checkbox"/>	-	<a href="#">i-0c95eff4685f1fa11</a>	<span>⌚</span> Pending	t2.micro	-	No alarms +	us-east-1c	ec2-3-83-201-181.com...	3.83.201.181	-	-
<input type="checkbox"/>	-	<a href="#">i-040b871298faf2adf</a>	<span>⌚</span> Pending	t2.micro	-	No alarms +	us-east-1c	ec2-34-238-52-241.co...	34.238.52.241	-	-
<input type="checkbox"/>	-	<a href="#">i-05b09d1d50205824e</a>	<span>⌚</span> Pending	t2.micro	-	No alarms +	us-east-1c	ec2-3-95-19-225.comp...	3.95.19.225	-	-

Select an instance

You can see three instances created from template. Diagram

*You can go one step ahead and create an auto scaling group.*

*In the Left navigation pane, choose Auto-Scaling groups.*

*Choose "Create Auto Scaling Group"*

**Amazon EC2 Auto Scaling**

Create Auto Scaling group

Reserved Instances

Dedicated Hosts

Scheduled Instances

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

▼ Auto Scaling

Launch Configurations

Auto Scaling Groups

# helps maintain the availability of your applications

Auto Scaling groups are collections of Amazon EC2 instances that enable automatic scaling and fleet management features. These features help you maintain the health and availability of your applications.

## How it works

An Auto Scaling group is a collection of Amazon EC2 instances that are treated as a logical unit. You configure settings for a group and its instances as well as define the group's minimum, maximum, and desired capacity. Setting different minimum and maximum capacity values forms the bounds of the group, which allows the group to scale as the load on your application spikes higher or lower, based on demand. To scale the Auto Scaling group, you can either make manual adjustments to the desired capacity or let

Get started with EC2 Auto Scaling by creating an Auto Scaling group.

Create Auto Scaling group

### Pricing

Amazon EC2 Auto Scaling features have no additional fees beyond the service fees for Amazon EC2, CloudWatch (for scaling policies), and the other AWS resources that you use. Visit the pricing page of each service to learn more.

### Getting started

- [What is Amazon EC2 Auto Scaling?](#)
- [Getting started with Amazon EC2 Auto Scaling](#)
- [Set up a scaled and load-balanced application](#)
- [FAQ](#)

CloudShell Feedback Language

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Choose "Create Auto Scaling Group" Diagram

*In the "Create Auto scaling Group: page enter a name for the Auto-scaling group.*

*For "Launch Template" , choose the launch template we created earlier.*

aws

Services

Search [Alt+S]

S3 EC2 IAM Console Home CodeCommit AWS Amplify

N. Virginia SaiCloud

EC2 > Auto Scaling groups > Create Auto Scaling group

Step 1

Choose launch template or configuration [Info](#)

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group. If you currently use launch configurations, you might consider migrating to launch templates.

Step 2

Choose instance launch options

Step 3 - optional

Configure advanced options

Name

Auto Scaling group name

Step 4 - optional

Configure group size and scaling policies

Step 5 - optional

Add notifications

Step 6 - optional

Add tags

Step 7

Review

Enter a name to identify the group.

demo-autoscaling

Must be unique to this account in the current Region and no more than 255 characters.

Launch template

Info

Switch to launch configuration

Launch template

Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

devopstemplate

Create a launch template

Version

Default (1)

Create a launch template version

Description

mydevopstemplate

AMI ID

ami-0889a44b331db0194

Key pair name

jenkins

Launch template

devopstemplate

lt-051d75cc5d66f064d

Security groups

-

Security group IDs

-

Instance type

t2.micro

Request Spot Instances

No

Choose "Create Auto Scaling Group" Diagram

*For "Network: choose the VPC which are default and subnet you want the instance to launch in.*

aws

Services

Search

[Alt+S]

S3

EC2

IAM

Console Home

CodeCommit

AWS Amplify

N. Virginia

SaiCloud

Step 1

Choose launch template or configuration

Step 2

Choose instance launch options

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

Choose instance launch options

Info

Choose the VPC network environment that your instances are launched into, and customize the Instance types and purchase options.

Network

Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC

Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-0e262fdb92274b6b9

172.31.0.0/16

Default

Create a VPC

Availability Zones and subnets

Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets

us-east-1a | subnet-06e4e97168124facc

172.31.32.0/20

Default

us-east-1b | subnet-0c51d370cc561b950

172.31.0.0/20

Default



172.31.0.0/20

us-east-1c | subnet-0f28bfe43227f6d21

172.31.80.0/20

Default

Create a subnet

Instance type requirements

Info

You can keep the same instance attributes or instance type from your launch template, or you can choose to override the launch template by specifying different instance attributes or manually adding instance types.

Override launch template

Launch template	Version	Description
devopstemplate	Default	mydevopstemplate
lt-051d75cc5d66f064d		

"Network: choose the VPC which are default and subnet Diagram.

*For "Load Balancing" choose any option as per your requirement.*

aws

Services

Search

[Alt+S]

S3

EC2

IAM

Console Home

CodeCommit

AWS Amplify

N. Virginia

SaiCloud

Step 2

Choose instance launch options

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

Load balancing

Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

No load balancer

Traffic to your Auto Scaling group will not be fronted by a load balancer.

Attach to an existing load balancer

Choose from your existing load balancers.

Attach to a new load balancer

Quickly create a basic load balancer to attach to your Auto Scaling group.

VPC Lattice integration options

Info

To improve networking capabilities and scalability, integrate your Auto Scaling group with VPC Lattice. VPC Lattice facilitates communications between AWS services and helps you connect and manage your applications across compute services in AWS.

Select VPC Lattice service to attach

No VPC Lattice service

VPC Lattice will not manage your Auto Scaling group's network access and connectivity with other services.

Attach to VPC Lattice service

Incoming requests associated with specified VPC Lattice target groups will be routed to your Auto Scaling group.

Create new VPC Lattice service

Health checks

Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

EC2 health checks

Always enabled

Additional health check types - optional

Info

Turn on Elastic Load Balancing health checks

Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.

Turn on VPC Lattice health checks

VPC Lattice can monitor whether instances are available to handle requests. If it considers a target as failed a health check, EC2 Auto Scaling replaces it after its next periodic check.

health check grace period [Info](#)  
This time period delays the first health check until your instances finish initializing. It doesn't prevent an instance from terminating when placed into a non-running state.

seconds

Load Balancing" choose any option as per your requirement. Diagram.

*In the "Group Size" page enter the desired capacity for the auto-scaling group such as or like 2 .*

The screenshot shows the AWS Management Console interface for creating an Auto Scaling group. The top navigation bar includes the AWS logo, 'Services' menu, a search bar, and regional/service dropdowns (N. Virginia, SaaSCloud). The left sidebar shows the navigation menu with steps 1 through 7. The main content area is titled 'Configure group size and scaling policies - optional' and includes an 'Info' link. Below the title is a description: '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.'

**Group size - optional** [Info](#)

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.

Desired capacity

Minimum capacity

Maximum capacity

**Scaling policies - optional**

Choose whether to use a scaling policy to dynamically resize your Auto Scaling group to meet changes in demand. [Info](#)

☐ Target tracking scaling policy  
Choose a desired outcome and leave it to the scaling policy to add and remove capacity as needed to achieve that outcome.

☒ None

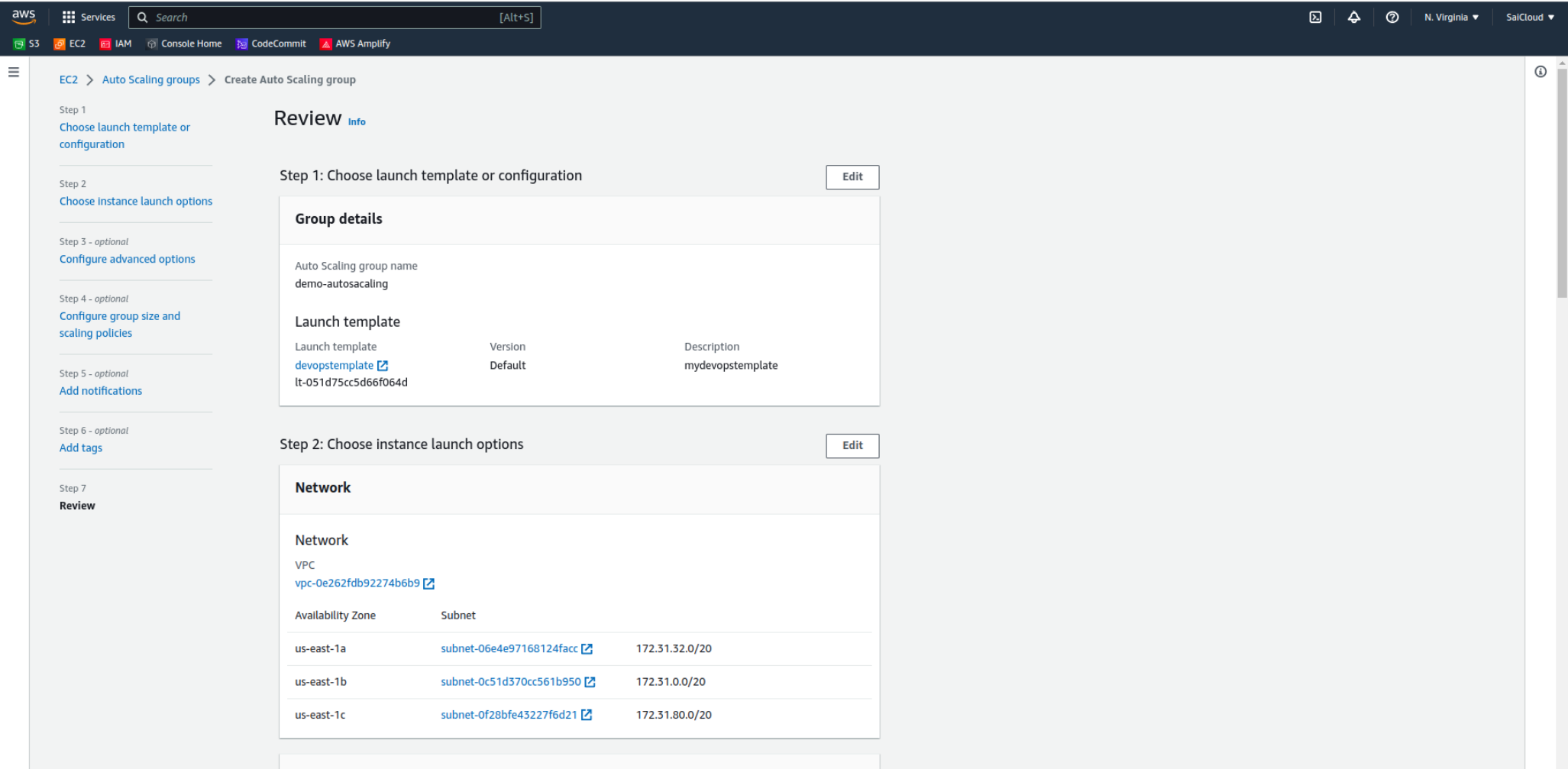
**Instance scale-in protection - optional**

Instance scale-in protection  
If selected, instances that are in the process of being terminated will be protected from scale-in by default.

"Group Size" page enter the desired capacity Diagram.

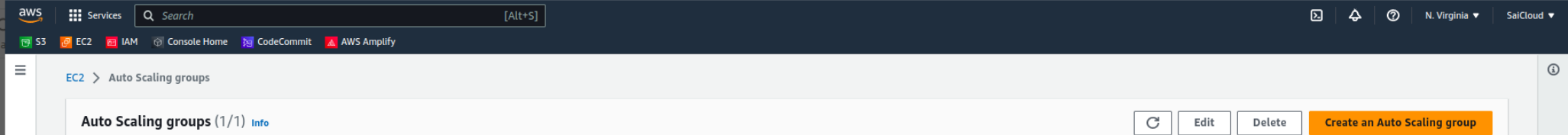
*in the next options you can choose set up an scaling policies based on various metrics such as CPU utilization, network in/out and others.choose the policy*

target tracking policy.



such as CPU utilization, Diagram

Choose "Create Auto Scaling Group" to create the auto-scaling group.



Search your Auto Scaling groups

< 1 > ⚙

<input checked="" type="checkbox"/>	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
<input checked="" type="checkbox"/>	demo-autoscaling	devopstemplate   Version Default	2	-	2	1	5	us-east-1a, us-east-1b, us-east-1c

Auto Scaling group: demo-autoscaling

⚙ ×

Details

Activity

Automatic scaling

Instance management

Monitoring

Instance refresh

Group details

Edit

Auto Scaling group name	demo-autoscaling	Desired capacity	2	Status	-	Amazon Resource Name (ARN)	arn:aws:autoscaling:us-east-1:414310061589:autoScalingGroup:5
-------------------------	------------------	------------------	---	--------	---	----------------------------	---

Create Auto Scaling Group Diagram

*After a few moments, the auto scaling group will launch the desired number of instances based on the launch template and the configuration you specified and the count of instances are 2 new instances. launched by auto scaling group.*

aws Services Search [Alt+S]

S3 EC2 IAM Console Home CodeCommit AWS Amplify

New EC2 Experience Tell us what you think

EC2 Dashboard EC2 Global View Events Limits

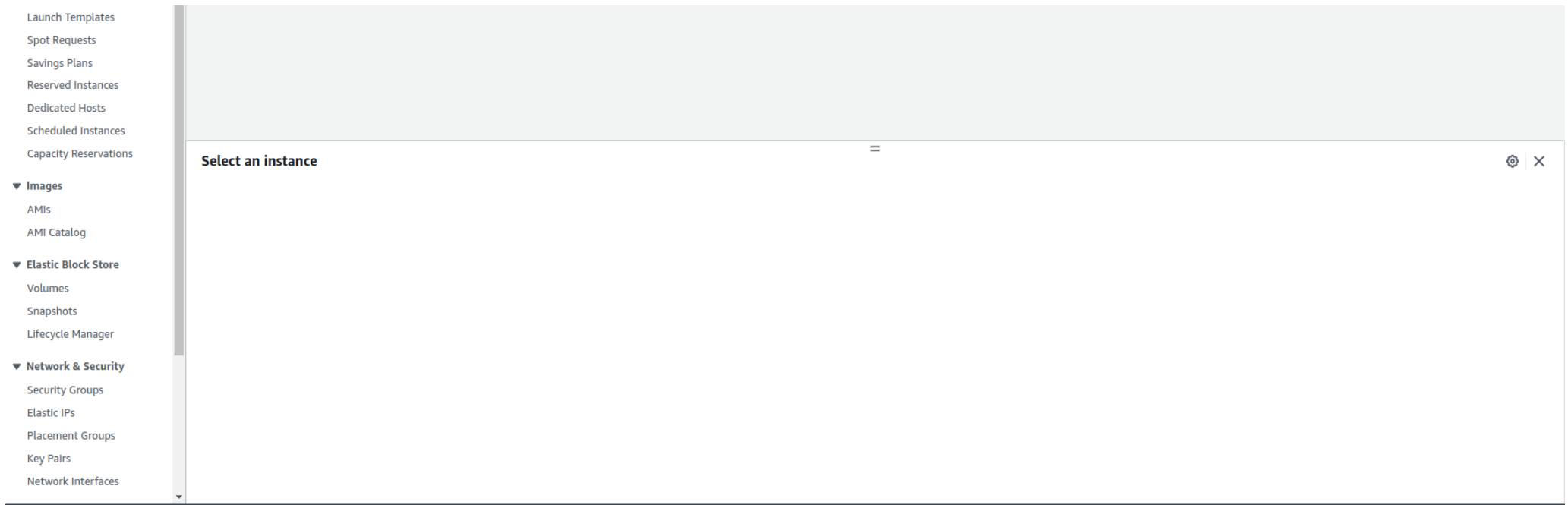
Instances Instances Instance Types

Instances (5) Info

Find Instance by attribute or tag (case-sensitive)

< 1 > ⚙

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs
<input type="checkbox"/>	-	i-02117b81238304f03	Running	t2.micro	Initializing	No alarms +	us-east-1c	ec2-44-211-163-211.co...	44.211.163.211	-	-
<input type="checkbox"/>	-	i-009ff53a59261ea3b	Running	t2.micro	Initializing	No alarms +	us-east-1b	ec2-35-170-52-180.co...	35.170.52.180	-	-
<input type="checkbox"/>	-	i-0c95eff4685f1fa11	Running	t2.micro	2/2 checks passed	No alarms +	us-east-1c	ec2-3-83-201-181.com...	3.83.201.181	-	-
<input type="checkbox"/>	-	i-040b871298faf2adf	Running	t2.micro	2/2 checks passed	No alarms +	us-east-1c	ec2-34-238-52-241.co...	34.238.52.241	-	-
<input type="checkbox"/>	-	i-05b09d1d50205824e	Running	t2.micro	2/2 checks passed	No alarms +	us-east-1c	ec2-3-95-19-225.comp...	3.95.19.225	-	-



2 Instances are added with the help of Autoscaling option Diagram.

*In this blog, I have discussed AWS Ec2 Automation with the help of task how we can launch a template. If you have any questions or would like to share your experiences, feel free to contact me or leave a comment.*

*Happy Learning!!*

*Maninder Singh*

*Next Topic:*

*Day 41: We will learn the Load Balancing with AWS EC2.*