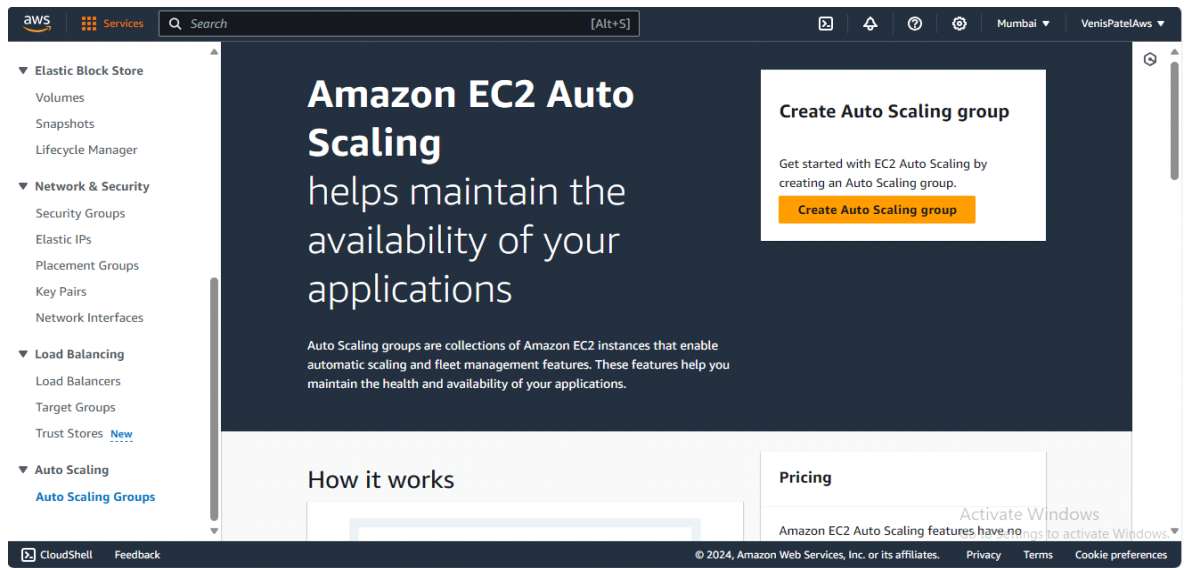


## TASK 5: Create an ASG with minimum 1 and maximum 2 instance requirement.

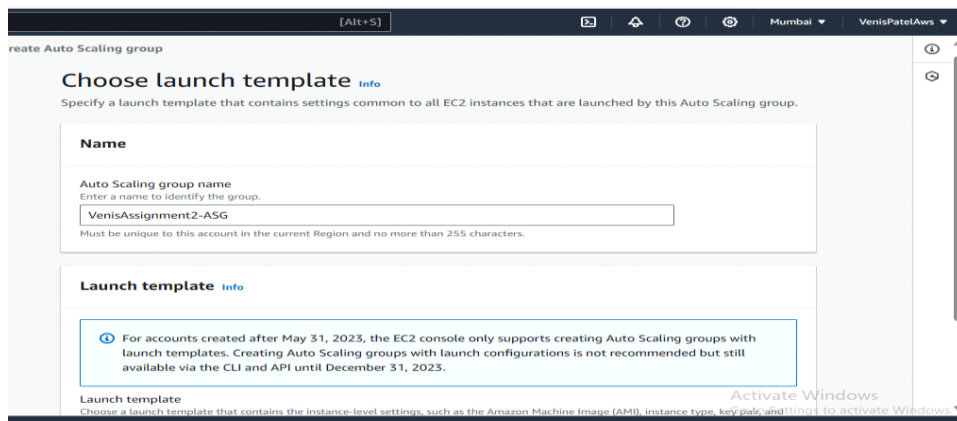
- Use "Stress" command for increasing CPU utilization and it should create 2nd instance automatically

### Steps to create Auto Scaling Group:

1. Click on Auto Scaling Groups from EC2 Dashboard. Then click on “Create Auto Scaling group”.



2. Then we need to enter ASG name and then click on “create a launch template”.



3. Then create launch template by first entering the template name “VenisAssignment2LaunchTemplate”.

**Create launch template**

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

**Launch template name and description**

Launch template name - required

VenisAssignment2LaunchTemplate

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '@', '!', etc.

Template version description

A prod webserver for MyApp

Max 255 chars

Auto Scaling guidance [Info](#)

Select this if you intend to use this template with EC2 Auto Scaling

☒ Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

**Summary**

Software Image (AMI)

-

Virtual server type (instance type)

-

Firewall (security group)

-

Storage (volumes)

-

**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

Cancel **Create launch template**

4. Then select Amazon Linux 2(Free tier eligible) as the AMI.

**Application and OS Images (Amazon Machine Image) - required** [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Amazon Linux 2

[AMI from catalog](#) [Recents](#) [Quick Start](#)

Amazon Machine Image (AMI)

amzn2-ami-kernel-5.10-hvm-2.0.20240124.0-x86\_64-gp2

ami-039e1f129f345d75f

**Verified provider** **Free tier eligible**

[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

ENA Enabled

**Summary**

Software Image (AMI)

Amazon Linux 2 AMI (HVM) - Ker...[read more](#)

ami-039e1f129f345d75f

Virtual server type (instance type)

-

Firewall (security group)

-

Storage (volumes)

1 volume(s) - 8 GiB

**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 I/Os, 1 GiB of snapshots, and 100 GB of bandwidth to the internet

5. Select the t2.micro as the instance type.

**Instance type** [Info](#) [Get advice](#) [Advanced](#)

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Linux base pricing: 0.0124 USD per Hour

On-Demand Windows base pricing: 0.017 USD per Hour

On-Demand RHEL base pricing: 0.0724 USD per Hour

On-Demand SUSE base pricing: 0.0124 USD per Hour

**Free tier eligible**

☐ All generations

[Compare instance types](#)

[Additional costs apply for AMIs with pre-installed software](#)

**Summary**

Software Image (AMI)

Amazon Linux 2 AMI (HVM) - Ker...[read more](#)

ami-039e1f129f345d75f

Virtual server type (instance type)

t2.micro

Firewall (security group)

-

## 6. Select launch-wizard-1 as the security group.

**Network settings** Info

Subnet Info  
Don't include in launch template [Create new subnet](#)

When you specify a subnet, a network interface is automatically added to your template.

Firewall (security groups) Info  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.  
☒ Select existing security group ☐ Create security group

Security groups Info  
Select security groups  
launch-wizard-1 sg-0aabb179b1935f196b X  
VPC: vpc-08e4e2a3c7a763669 [Compare security group rules](#)

► Advanced network configuration

**Summary**

Software Image (AMI)  
Amazon Linux 2 AMI (HVM) - Ker...[read more](#)  
ami-039e1f129f345d75f

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
launch-wizard-1

Storage (volumes)  
1 volume(s) - 8 GiB

**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

## 7. Keep everything as default, then add the following as user data:

```
#!/bin/bash
```

```
yum update -y
```

```
yum install -y httpd
```

```
systemctl start httpd
```

```
systemctl enable httpd
```

```
echo "<h1>IP ADDRESS: $(hostname -f)</h1>" > /var/www/html/index.html
```

Metadata transport  
Don't include in launch template

Metadata version Info  
Don't include in launch template

Metadata response hop limit Info  
2

Allow tags in metadata Info  
Don't include in launch template

User data - optional Info  
Upload a file with your user data or enter it in the field.  
[Choose file](#)

```
#!/bin/bash
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "<h1>IP ADDRESS: $(hostname -f)</h1>" > /var/www/html/Index.html
```

**Summary**

Software Image (AMI)  
Amazon Linux 2 AMI (HVM) - Ker...[read more](#)  
ami-039e1f129f345d75f

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
launch-wizard-1

Storage (volumes)  
1 volume(s) - 8 GiB

**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 I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

8. Then click on create launch template to create this template, Now select VenisAssignment2LaunchTemplate in Launch Template, then click on next.

[Alt+S]

**Launch template** Info

For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

**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.

VenisAssignment2LaunchTemplate

Create a launch template

**Version**  
Default (1)

Create a launch template version

**Description**  
-

**Launch template**  
VenisAssignment2LaunchTemplate  
lt-07100ba563bd50515

**Instance type**  
t2.micro

**Request Spot Instances**  
No

**AMI ID**  
ami-039e1f129f345d75f

**Security groups**  
-

Activate Windows  
Go to Settings to activate Windows

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

9. Now inside Network select multiple availability zones, then click on next.

[Alt+S]

**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-08e4e2a3c7a763669  
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

ap-south-1a | subnet-099914c00ecd12bd7  
172.31.32.0/20 Default

ap-south-1b | subnet-03f21b974423938ed  
172.31.0.0/20 Default

ap-south-1c | subnet-07fa807bdc68aabd7  
172.31.16.0/20 Default

Create a subnet

Activate Windows  
Go to Settings to activate Windows

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

10. Now inside load balancer choose Attach to an existing load balancer. Select Alp-Load-Balancer-Target-Group from existing load balancer groups. Then click on next.

**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.

**Attach to an existing load balancer**  
Select the load balancers that you want to attach to your Auto Scaling group.

☒ Choose from your load balancer target groups  
This option allows you to attach Application, Network, or Gateway Load Balancers.

☐ Choose from Classic Load Balancers

Existing load balancer target groups  
Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

Select target groups

Alp-LoadBalancer-Target-Group | HTTP  
Application Load Balancer: VenisAssignment3AlpLoadBalancer

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

11. Now desired group size should be 1 and set minimum group size to 1 and maximum group size to 2, then click on skip for review. Then click on create Auto Scaling 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.

1

**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.

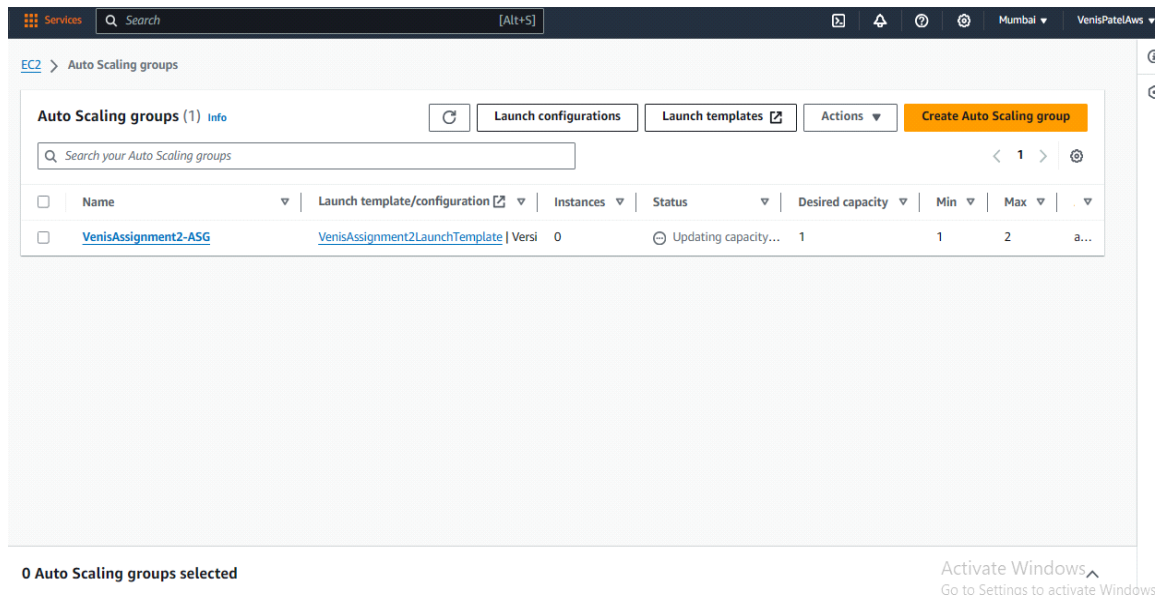
**Min desired capacity** **Max desired capacity**

1 2

Equal or less than desired capacity Equal or greater than desired capacity

**Automatic scaling - optional**  
Choose whether to use a target tracking policy [Info](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)



Using stress command:

In order for stress command to work enable ELB and set health update to 10s.

- To install stress connect to ec2 instance and enter the following command:
  - `sudo amazon-linux-extras install epel`

```
[ec2-user@ip-172-31-1-18 ~]$ sudo amazon-linux-extras install epel
Installing epel-release
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-epel amzn2extra-kernel-5.10
17 metadata files removed
6 sqlite files removed
0 metadata files removed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
amzn2extra-docker
amzn2extra-epel
amzn2extra-kernel-5.10
(1/9): amzn2-core/2/x86_64/group.gz | 3.6 kB 00:00:00
(2/9): amzn2-core/2/x86_64/updateinfo | 2.9 kB 00:00:00
(3/9): amzn2extra-docker/2/x86_64/primary_db | 3.0 kB 00:00:00
(4/9): amzn2extra-epel/2/x86_64/primary_db | 3.0 kB 00:00:00
(5/9): amzn2extra-kernel-5.10/2/x86_64/updateinfo | 2.7 kB 00:00:00
(6/9): amzn2extra-docker/2/x86_64/updateinfo | 1.8 kB 00:00:00
(7/9): amzn2extra-epel/2/x86_64/updateinfo | 44 kB 00:00:00
(8/9): amzn2extra-kernel-5.10/2/x86_64/primary_db | 105 kB 00:00:00
(9/9): amzn2-core/2/x86_64/primary_db | 14 kB 00:00:00
Resolving Dependencies
--> Running transaction check
--> Package epel-release.noarch 0:7-11 will be installed
--> Finished Dependency Resolution
```

- `sudo yum install stress`

```
[ec2-user@ip-172-31-1-187 ~]$ sudo yum install stress
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
226 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
--> Package stress.x86_64 0:1.0.4-16.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====================================================================================================================================
 Package                               Arch                               Version                               Repository                           Size
=====================================================================================================================================
Installing:
 stress                               x86_64                             1.0.4-16.el7                         epel                                  39 k
Transaction Summary
-----
Install 1 Package

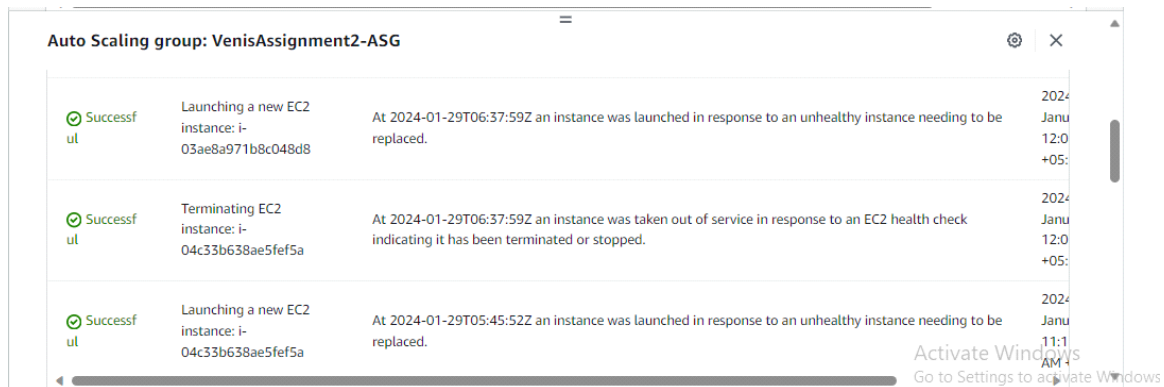
Total download size: 39 k
Installed size: 94 k
Is this ok [y/d/N]: y
Downloading packages:
warning: /var/cache/yum/x86_64/2/epel/packages/stress-1.0.4-16.el7.x86_64.rpm: Header V3 RSA/SHA256 Signature, key ID 352c64e5: NOKEYs | 0 B --:--: ETA
```

- To run stress command in order to max out CPU utilization use the following command:
- Sudo stress --Cpu 1000 --timeout 120

```
[ec2-user@ip-172-31-32-92 ~]$ sudo stress --cpu 1000 --timeout 120
stress: info: [3798] dispatching hogs: 1000 cpu, 0 io, 0 vm, 0 hdd
stress: info: [3798] successful run completed in 120s
[ec2-user@ip-172-31-32-92 ~]$
```

## Results of stress command:

-> First instance became unhealthy then stopped and new instance Launched.



-> we can clearly see increment in Cpu utilization due to stress command we used.

## CPU Utilization (P... ⓘ ⋮

