

# CLOUD COMPUTING ARCHITECTURE LAB

NAME- SHUBHI DIXIT

SAP ID- 500094571

BATCH- 05

## Experiment 3: Show how Autoscaling can be performed on EC2 instances

### **1. List the key components of Autoscaling.**

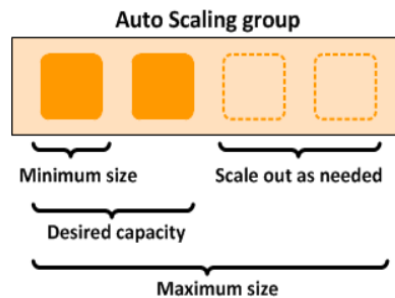
The main components of AWS that are involved in autoscaling include autoscaling groups, Amazon Machine Image (AMI), load balancer, snapshot, and EC2 instance.

The key components of autoscaling groups are :

- I. **Groups** : Your EC2 instances are organized into groups so that they can be treated as a logical unit for the purposes of scaling and management. When you create a group, you can specify its minimum, maximum, and, desired number of EC2 instances.

An autoscaling group performs the following tasks:

- It adds or removes instances depending on the load of the server. When there is a high load, it will add instances. If the load is low, it will remove instances (extra ones).
- It scales EC2 instances up or down, which helps in managing the availability of applications.
- It runs the required number of instances. For example, if the required number of instances is 5, then it will run 5 EC2 instances.



**II. Configuration templates** : Your group uses a launch template, or a launch configuration (not recommended, offers fewer features), as a configuration template for its EC2 instances. You can specify information such as the AMI ID, instance type, key pair, security groups, and block device mapping for your instances

**III. Scaling options**: Amazon EC2 Auto Scaling provides several ways for you to scale your Auto Scaling groups. For example, you can configure a group to scale based on the occurrence of specified conditions (dynamic scaling) or on a schedule or it can be manual, dynamic or predictive scaling

## 2. Differentiate between Launch Template and Launch Configuration. Which is recommended and Why?

<b><u>3. Launch Template</u></b>	<b><u>Launch Configuration</u></b>
More features than autoscaling	Provides autoscaling only
Supports multi-versions	Limited Configuration and immutable
Its recommended	Its not recommended

Launch Templates are preferred over launch configuration as launch template is mutable unlike launch configuration. Launch templates provides provisioning for both spot and on-demand instances and

allows the usage of T2 unlimited burst feature. Launch configurations are recreated every time because modification isn't allowed.

### **3. Explain the different Autoscaling options available in EC2 with the help of suitable options.**

There are majorly four types of Autoscaling options are :-

- ***Manual scaling:***

In this type of scaling, the number of instances is changed manually. It involves a manual execution of scaling actions. The number of instances can be increased or decreased manually using a CLI or console. This type of scaling is ideal when users do not need automatic scaling.

- ***Scheduled scaling:***

This type involves the automatic execution of scaling actions based on certain schedules. They can be executed at a specific time during the day, month, week, or year. This type of scaling is ideal when traffic occurs at a specific time.

For example, it can be used if there is heavy traffic during the weekends and relatively less traffic during the weekdays. In this case, the number of instances can be scheduled to increase when the weekend begins. This number can be reduced when the weekend ends.

- ***Dynamic scaling:***

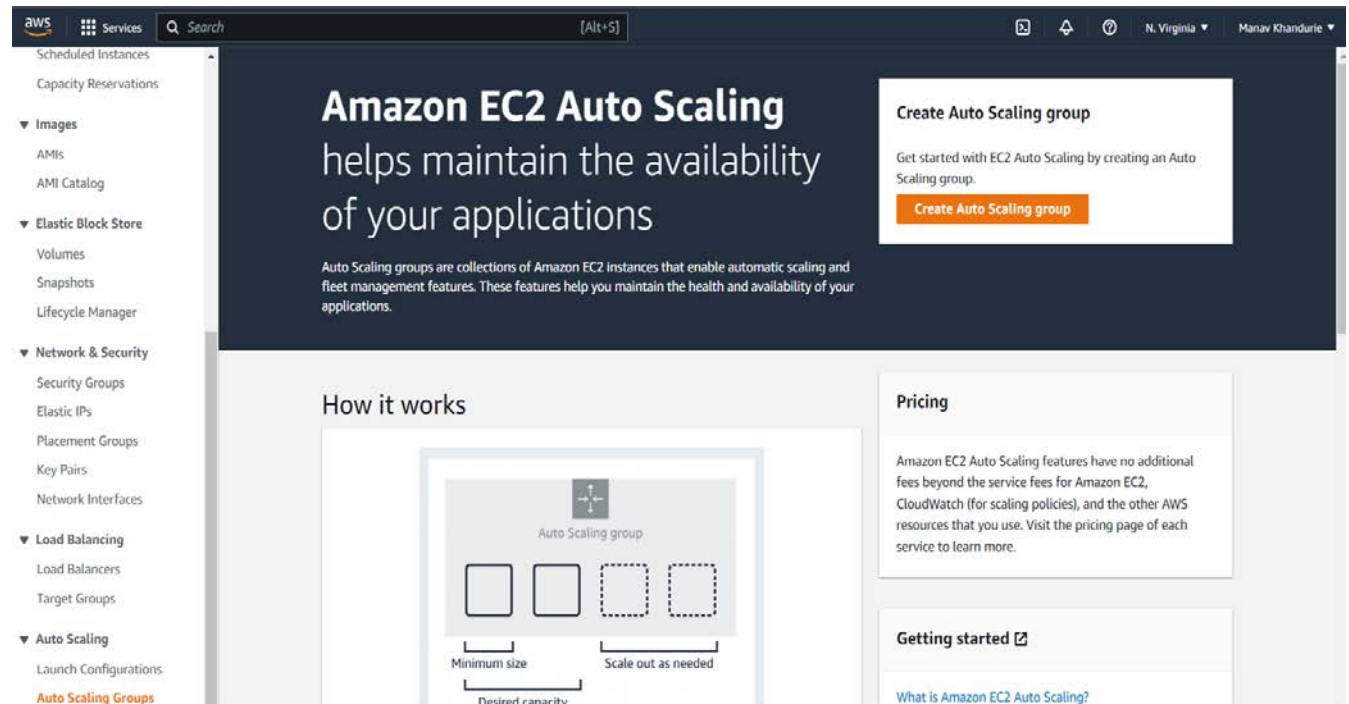
In dynamic scaling, the number of EC2 instances is changed automatically based on signals that are provided by a CloudWatch alarm. Dynamic scaling is mostly employed when there is unpredictable traffic.

- ***Predictive Scaling:***

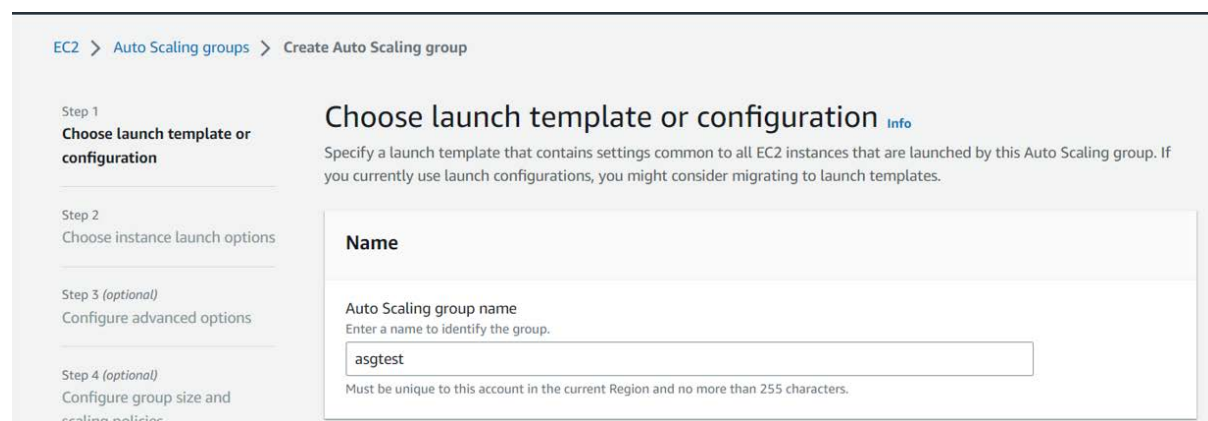
Predictive scaling involves using machine learning algorithms to program the desired number of instances. Future traffic can be predicted to provide the appropriate number of instances. This type of scaling is ideal when the traffic is predictable.

# EXPERIMENT-

Step 1: - Login to the aws account and goto auto-scaling groups



Step 2:- Click on “Create auto scaling group” & choose a name for the group



Step 3:- Select the template or create if not already present by clicking “create a launch template”

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.

Select a launch template

[Create a launch template](#)

Cancel

Next

Step 4:- Provide the template with appropriate configuration

▼ 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 Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

S

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type

ami-01a4f99c4ac11b03c (64-bit (x86)) / ami-0f3443fa43a3a92d2 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Amazon Linux 2 Kernel 5.10 AMI 2.0.20230119.1 x86\_64 HVM gp2

Architecture

64-bit (x86)

AMI ID

ami-01a4f99c4ac11b03c

Verified provider

▼ 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 group name - required

sgc

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and \_-:/()#,@[]+=&;!\$\*

Description - required

Info

allow

VPC - required

Info

vpc-0f2ee231bfdab959c

172.31.0.0/16

(default)

Inbound security groups rules

No security group rules are currently included in this template. Add a new rule to include it in the launch template.

Add security group rule

► Advanced network configuration

### Inbound security groups rules

▼ Security group rule 1 (TCP, 0-65535, 0.0.0.0/0)

Remove

Type [Info](#)

Protocol [Info](#)

Port range [Info](#)

All TCP ▼

TCP

0-65535

Source type [Info](#)

Source [Info](#)

Description - optional [Info](#)

Anywhere ▼

0.0.0.0/0 ✕

▼ Security group rule 2 (UDP, 0-65535, 0.0.0.0/0)

Remove

Type [Info](#)

Protocol [Info](#)

Port range [Info](#)

All UDP ▼

UDP

0-65535

Source type [Info](#)

Source [Info](#)

Description - optional [Info](#)

Anywhere ▼

0.0.0.0/0 ✕

⚠ 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.

✕

Add security group rule

Step 4:- After launching the template select it in step 3 by refreshing the list

EC2 > Launch templates > testemp

testemp (lt-0d03812715fddd53b)

Actions ▼

Delete template

#### Launch template details

Launch template ID	Launch template name	Default version	Owner
lt-0d03812715fddd53b	testemp	1	arn:aws:iam::247477386084:root

Details

Versions

Template tags

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 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.

Name

Auto Scaling group name

Enter a name to identify the group.

asgtestmanav

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.

template2

Create a launch template [↗](#)

Version

Default (1)

Create a launch template version [↗](#)

Description

-

Launch template

template2 [↗](#)

lt-0a53f3204fd4787c3

Instance type

t2.micro

Step 5:- Go to the next step and select all the availability zones and subnets

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-05525b34412abb13b

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-northeast-1c | subnet-09a82cdbb4d6eec1b

172.31.0.0/20 Default

ap-northeast-1a | subnet-09221e2f7f5380109

172.31.32.0/20 Default

ap-northeast-1d | subnet-0acd8030c4124f046

172.31.16.0/20 Default

Create a subnet [↗](#)

Step 6: For now , keep load balancing off



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)

## Configure advanced options [Info](#)

Choose a load balancer to distribute incoming traffic for your application across instances to make it more reliable and easily scalable. You can also set options that give you more control over health check replacements and monitoring.

### Load balancing - optional [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.

## Step 7: In configuration of group size and scaling policies

EC2 > Auto Scaling groups > Create Auto Scaling group

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

## Configure group size and scaling policies [Info](#)

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

2

Minimum capacity

1

Maximum capacity

3

Step 8: - Select the scaling policies where we scale up @ 70% CPU utilization

## 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

Scaling policy name

Target Tracking Policy

Metric type

Average CPU utilization ▼

Target value

70

Instances need

300

seconds warm up before including in metric

☐ Disable scale in to create only a scale-out policy

## Step 9:- Turn off the notifications and tags

### Add notifications [Info](#)

Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.

Add notification

Cancel

Previous

Skip to review

Next

## Step 10:- Review all the configuration

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**

### Review [Info](#)

Step 1: Choose launch template or configuration

Edit

#### Group details

Auto Scaling group name  
asgtestmanav

#### Launch template

Launch template  
template2 [🔗](#)  
lt-0a53f3204fd4787c3

Version  
Default

Description

Step 2: Choose instance launch options

Edit

#### Network

#### Network

VPC  
vpc-05525b34412abb13b [🔗](#)

Step 6 (optional)  
Add tags

Step 7  
Review

Step 2: Choose instance launch options

Edit

Network

Network

VPC

vpc-05525b34412abb13b

Availability Zone	Subnet	
ap-northeast-1c	subnet-09a82cdbb4d6eec1b	172.31.0.0/20
ap-northeast-1a	subnet-09221e2f7f5380109	172.31.32.0/20
ap-northeast-1d	subnet-0acd8030c4124f046	172.31.16.0/20

Step 4: Configure group size and scaling policies

Edit

Group size

Desired capacity	Minimum capacity	Maximum capacity
2	1	3

Scaling policy

<div>Target tracking scaling</div> <div>Policy type</div> <div>Target tracking scaling</div> <div>Take the action</div> <div>Add or remove capacity units as required</div>	<div>Scaling policy name</div> <div>Target Tracking Policy</div> <div>Instances need</div> <div>300 seconds to warm up before including in metric</div>	<div>Execute policy when</div> <div>As required to maintain Average CPU utilization at 70</div> <div>Scale in</div> <div>Enabled</div>
---	---	--

Instance scale-in protection

Instance scale-in protection

☐ Enable instance protection from scale in

Step 11: Launch the group and view instances in instance manager

aws

Services

Search

[Alt+S]

Tokyo

Manav Khandurie

asgtestmanav, 1 Scaling policy created successfully

EC2 > Auto Scaling groups

Auto Scaling groups (1) Info

Refresh

Edit

Delete

Create an Auto Scaling group

Search your Auto Scaling groups

< 1 >

ⓘ

<input type="checkbox"/>	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availabil...
<input type="checkbox"/>	asgtestmanav	template2   Version Default	0	Updating capacity...	2	1	3	ap-northea...

aws Services Search [Alt+S] Tokyo Manav Khandurie

New EC2 Experience Tell us what you think

EC2 Dashboard  
EC2 Global View  
Events  
Tags  
Limits

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
-	i-0ea09fb08b4f19200	Running	t2.micro	Initializing	No alarms	ap-northeast-1a	ec2-3-113-3-91.ap-...
-	i-0192c3472ddbef19d	Running	t2.micro	Initializing	No alarms	ap-northeast-1c	ec2-18-183-62-57.z...
-	i-0147c3e8b320a85f0	Terminated	t2.micro	-	No alarms	ap-northeast-1c	-

Step 12: Now as the instances are part of a scaling group , they will get reinitialized every time an instance is destroyed/terminated

Instances (1/3) 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
-	i-0ea09fb08b4f19200	Running	t2.micro	Initializing	No alarms	ap-northeast-1a	ec2-3-113-3-91.ap-...
-	i-0192c3472ddbef19d	Running	t2.micro	Initializing	No alarms	ap-northeast-1c	ec2-18-183-62-57.z...
-	i-0147c3e8b320a85f0	Terminated	t2.micro	-	No alarms	ap-northeast-1c	-

Instance: i-0ea09fb08b4f19200

Details Security Networking Storage Status checks Monitoring

CPU utilization (%) 41

Status check failed (any) (co... 1

Terminate instance

Instance settings Networking Security Image and templates Monitor and troubleshoot

Manage detailed monitoring Add to dashboard

Status check failed (system) (... 1

Successfully terminated i-0ea09fb08b4f19200

Instances (1/3) 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
-	i-0ea09fb08b4f19200	Shutting-down	t2.micro	Initializing	No alarms	ap-northeast-1a	ec2-3-113-3-91.ap-...
-	i-0192c3472ddbef19d	Running	t2.micro	Initializing	No alarms	ap-northeast-1c	ec2-18-183-62-57.z...
-	i-0147c3e8b320a85f0	Terminated	t2.micro	-	No alarms	ap-northeast-1c	-

Also you can check autoscaling by increasing the stress.

Step1: Select any instance and connect it.

Successfully terminated i-0ea09fb08b4f19200

Instances (4) Info

Find instance by attribute or tag (case-sensitive)

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input type="checkbox"/>	-	i-0d43d1365a3dacbad	Running	t2.micro	Initializing	No alarms	ap-northeast-1d	ec2-3-112-102-7
<input type="checkbox"/>	-	i-0ea09fb08b4f19200	Terminated	t2.micro	-	No alarms	ap-northeast-1a	-
<input type="checkbox"/>	-	i-0192c3472ddbef19d	Running	t2.micro	2/2 checks passed	No alarms	ap-northeast-1c	ec2-18-183-62-5
<input type="checkbox"/>	-	i-0147c3e8b320a85f0	Terminated	t2.micro	-	No alarms	ap-northeast-1c	-

**Connect to instance** Info

Connect to your instance i-087d7b71ef48b17ed using any of these options

EC2 Instance Connect Session Manager SSH client EC2 serial console

Instance ID  
i-087d7b71ef48b17ed

Public IP address  
65.0.133.3

User name  
Enter the user name defined in the AMI used to launch this instance. If you didn't define a custom user name, use the default user name, ec2-user.  
ec2-user

Note: In most cases, the default user name, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

Cancel Connect

Step 2: Write the following commands on the editor to check for stress.

**sudo amazon-linux-extras install epel -y**

```
[ec2-user@ip-172-31-43-79 ~]$ sudo amazon-linux-extras install epel -y
Installing epel-release
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amazon-core amazonxtra-docker amazonxtra-epel amazonxtra-kernel-5.10
0 obsolete files removed
0 obsolete files removed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amazon-core
amazonxtra-docker
amazonxtra-epel
amazonxtra-kernel-5.10
(1/9): amazon-core/2/x86_64/group.gz | 3.7 kB | 00:00:00
(2/9): amazon-core/2/x86_64/updatesinfo | 3.0 kB | 00:00:00
(3/9): amazonxtra-epel/2/x86_64/primary.db | 3.0 kB | 00:00:00
(4/9): amazonxtra-kernel-5.10/2/x86_64/updatesinfo | 3.0 kB | 00:00:00
(5/9): amazonxtra-docker/2/x86_64/updatesinfo | 2.5 kB | 00:00:00
(6/9): amazonxtra-epel/2/x86_64/updatesinfo | 584 kB | 00:00:00
(7/9): amazonxtra-docker/2/x86_64/primary.db | 1.8 kB | 00:00:00
(8/9): amazonxtra-kernel-5.10/2/x86_64/primary.db | 24 kB | 00:00:00
(9/9): amazon-core/2/x86_64/primary.db | 8.6 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 -y**

```
[ec2-user@ip-172-31-43-79 ~]$ sudo yum install stress -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
114 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: 84 K
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 352c69d1: NOKEY
Public key for stress-1.0.4-16.el7.x86_64.rpm is not installed
```

**stress --cpu 8 --io 4 --vm 2 --vm-bytes 128M --timeout 10s**

```
[ec2-user@ip-172-31-43-79 ~]$ stress --cpu 8 --io 4 --vm 2 --vm-bytes 128M --timeout 10s
stress: info: [3861] dispatching hogs: 8 cpu, 4 io, 2 vm, 0 hdd
stress: info: [3861] successful run completed in 10s
[ec2-user@ip-172-31-43-79 ~]$
```

Step 3: Check your instances, as soon as stress is generated, new instances will launch.

Instances (2) info

Refresh

Connect

Instance state

Actions

Launch instances

Find instance by attribute or tag (case-sensitive)

Instance state: running

Clear filters

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input type="checkbox"/>	-	j-0426b557bd4546c77	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1a	ec2-15-232-124-
<input type="checkbox"/>	-	j-0c9f0990315ef7cc5	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1b	ec2-3-310-174-



Step 13 : Terminate the auto scaling group by terminating it from EC2->Auto scaling groups

