

Implementing Setting Up Auto Scaling in AWS

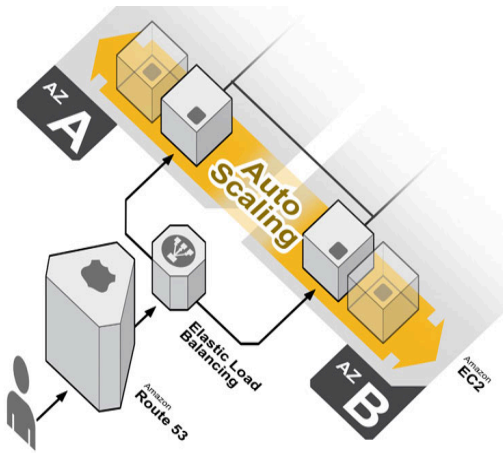
Step 1: Launch an EC2 Instance

1. **Log in to AWS Console:** Navigate to the [AWS Management Console](#).
2. **Go to EC2:** From the services menu, select **EC2**.
3. **Launch an EC2 Instance:**
 - Choose an Amazon Machine Image (AMI) and instance type.
 - Configure network settings and security groups.
 - Install necessary applications or scripts for the instance.
4. **Create a Key Pair:** Download it for SSH access to the instance.
5. Once launched, ensure the instance is accessible and functioning.

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input type="checkbox"/>	instance-3	i-03f98a54a0db89eab	Stopped	t2.micro	-	View alarms +	us-east-1e	-
<input type="checkbox"/>		i-04ccef41ed7c9842	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1e	ec2-34-239-107-155.
<input type="checkbox"/>	instance-2	i-073aba66090d43b44	Stopped	t2.micro	-	View alarms +	us-east-1d	-
<input type="checkbox"/>	instance-1	i-08f62e83e975c6364	Stopped	t2.micro	-	View alarms +	us-east-1c	-
<input type="checkbox"/>		i-0ad832a04c69c0b42	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c	ec2-44-211-220-25.
<input type="checkbox"/>		i-0f2c75b96b13b43b2	Terminated	t2.micro	-	View alarms +	us-east-1b	-
<input checked="" type="checkbox"/>		i-0978e66f1479d1fd7	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1f	ec2-3-239-88-107.

Step 2: Create a Launch Template or Launch Configuration

1. **Go to Launch Templates:**
 - From the EC2 dashboard, select **Launch Templates**.
 - Click **Create Launch Template**.



2. Fill Template Details:

- Provide a name and description.
- Specify the AMI, instance type, and key pair.
- Configure storage, network, and security settings.
- Add any startup scripts in the **User Data** section.

3. Save the Launch Template.

Alternatively, you can create a **Launch Configuration**:

- Navigate to the Auto Scaling section and select **Launch Configurations**.
- Follow similar steps to define instance settings.

The screenshot shows the AWS Management Console interface for a Launch Template. The breadcrumb navigation is **EC2 > Launch templates > MyFirstTemplate**. The left sidebar contains navigation links for Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, and Lifecycle Manager.

The main content area displays the details for **MyFirstTemplate (lt-07efcd2559c623240)**. It includes buttons for **Actions** and **Delete template**. The **Launch template details** section shows:

- Launch template ID:** lt-07efcd2559c623240
- Launch template name:** MyFirstTemplate
- Default version:** 1
- Owner:** arn:aws:sts::525620525649:assumed-role/voclabs/user3312069-Ruturaj_Sonone

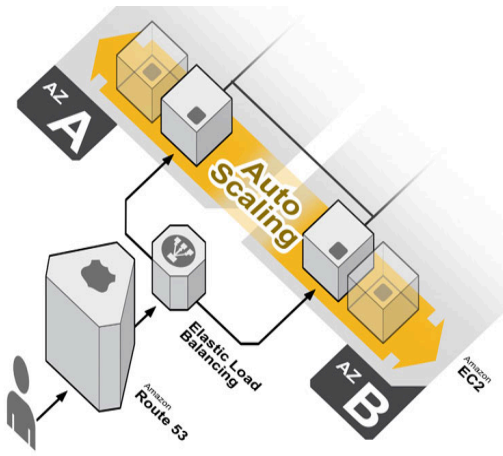
Below this, there are tabs for **Details**, **Versions**, and **Template tags**. The **Launch template version details** section for version 1 (Default) shows:

- Description:** ver1
- Date created:** 2024-12-31T09:42:11.000Z
- Created by:** arn:aws:sts::525620525649:assumed-role/voclabs/user3312069-Ruturaj_Sonone

Further down, there are tabs for **Instance details**, **Storage**, **Resource tags**, **Network interfaces**, and **Advanced details**. The **Instance details** tab shows:

- AMI ID:** ami-0e2c8caa4b6378d8c
- Instance type:** t2.micro
- Availability Zone:** -
- Key pair name:** sample-key
- Security groups:** -
- Security group IDs:** sg-01aae6d9fcddee60

Step 3: Create an Auto Scaling Group



1. **Navigate to Auto Scaling Groups:**
 - From the EC2 dashboard, select **Auto Scaling Groups**.
2. **Create a New Auto Scaling Group:**
 - Choose the **Launch Template** or **Launch Configuration** created earlier.
3. **Configure Group Details:**
 - Specify the VPC and subnets where instances will be launched.
 - Set the minimum, maximum, and desired number of instances.
4. **Attach Load Balancer (Optional):**
 - Attach an existing Application Load Balancer or Classic Load Balancer to distribute traffic.
5. **Set Scaling Policies:**
 - Enable scaling based on metrics like **CPU Utilization** or custom CloudWatch alarms.
 - Example: Add instances if CPU > 70% and remove instances if CPU < 20%.
6. **Review and Create:**
 - Confirm the settings and create the Auto Scaling Group.

firstautoscalinggroup

firstautoscalinggroup Capacity overview
Edit

am:aws:autoScaling:us-east-1:525620525649:autoScalingGroup:19901848-04a5-4abb-a26f-76e6cbddae23:autoScalingGroupName/firstautoscalinggroup

Desired capacity	Scaling limits (Min - Max)	Desired capacity type	Status
1	1 - 10	Units (number of instances)	-

Date created
Tue Dec 31 2024 15:18:32 GMT+0530 (India Standard Time)

Details
Integrations - new
Automatic scaling
Instance management
Instance refresh
Activity
Monitoring

Launch template
Edit

Launch template
lt-07efcd2559c623240
MyFirstTemplate

Version
Default

Description
ver1

AMI ID
ami-0e2c8caa4b6378d8c

Security groups
-

Storage (volumes)
/dev/sda1

Instance type
t2.micro

Security group IDs
sg-01aaea6d9fcfdee60

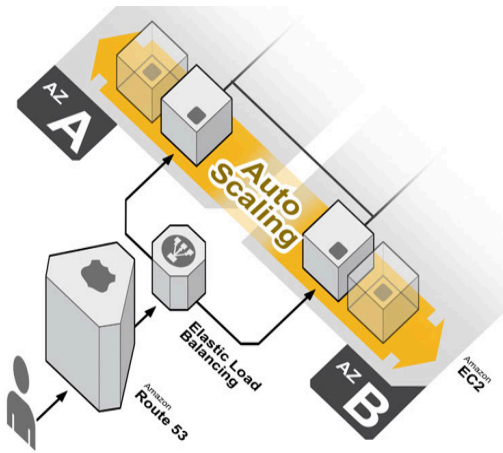
Key pair name
sample-key

Owner
arn:aws:sts:525620525649:assumed-role/voclabs/user3312069:Ruturaj_Sonone

Create time
Tue Dec 31 2024 15:12:11 GMT+0530 (India Standard Time)

Request Spot Instances
No

View details in the launch template console



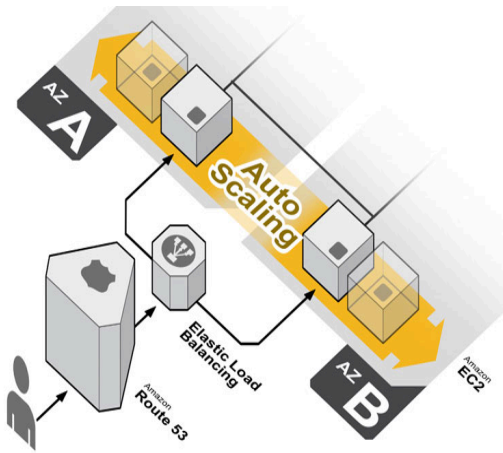
Network Edit		
Availability Zones us-east-1a, us-east-1b, us-east-1c, us-east-1d, us-east-1e, us-east-1f	Subnet ID subnet-0bb5d397b3bf8cb51, subnet-0bd5e57209d2e5943, subnet-0a2fd496038b5cddb, subnet-0c3d973192139fa5e, subnet-0225eb098617c4d8b, subnet-0c3d545871064c720	Availability Zone distribution Balanced best effort
Instance type requirements Edit Your Auto Scaling group adheres to the launch template for purchase option and instance type.		
<i>Load balancing and VPC lattice options have moved to the new integrations tab.</i> View integrations tab		
Health checks Edit		
Health check type EC2	Health check grace period 300	
Instance maintenance policy Edit		
Replacement behavior Terminate and launch	Min healthy percentage 40	Max healthy percentage 100

Step 4: Configure Scaling Policies

1. Dynamic Scaling:

- Navigate to the Auto Scaling Group settings.
- Add policies to scale in or out based on CloudWatch alarms.
- Example:
 - **Scale Out:** Add instances when CPU exceeds 70%.
 - **Scale In:** Remove instances when CPU drops below 20%.

2. I have chosen **dynamic scaling** , but you can choose **Predictive scaling policies** or **Scheduled actions** as per your choice.



Dynamic scaling policies (1) [Info](#)

Target Tracking Policy ☐

Policy type

Target tracking scaling

Enabled or disabled

Enabled

Execute policy when

As required to maintain Average CPU utilization at 40

Take the action

Add or remove capacity units as required

Instances need

300 seconds to warm up before including in metric

Scale in

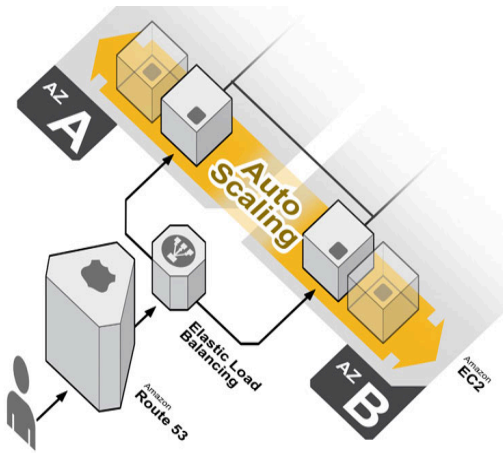
Enabled

Step 5: Test the Auto Scaling

1. Simulate Load:

- Use tools like **stress** to increase CPU usage or generate traffic.
 - To increase CPU stress for testing purposes on an AWS EC2 instance via SSH, you can use the **stress** tool. Here's how you can do it:

[Alt+S]									
N. Virginia voclabs/user3312069-Ruturaj_Sonone @ 5256-2052-5649									
Instances (1/10) Info Last updated 1 minute ago Connect Instance state Actions Launch instances									
Find Instance by attribute or tag (case-sensitive) All states									
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	
<input type="checkbox"/>	instance-3	i-03f98a54a0db89eab	Stopped	t2.micro	-	View alarms	us-east-1e	-	
<input checked="" type="checkbox"/>		i-06866f702eb96fa95	Running	t2.micro	2/2 checks passed	View alarms	us-east-1e	ec2-18-208-119-93.0	
<input type="checkbox"/>		i-04ccef41ed7c9842	Terminated	t2.micro	-	View alarms	us-east-1e	-	



Connect to instance [Info](#)

Connect to your instance i-06866f702eb96fa95 using any of these options

[EC2 Instance Connect](#)
[Session Manager](#)
[SSH client](#)
[EC2 serial console](#)

Instance ID

[i-06866f702eb96fa95](#)

Connection Type

☒ Connect using EC2 Instance Connect
 Connect using the EC2 Instance Connect browser-based client, with a public IPv4 or IPv6 address.

☐ Connect using EC2 Instance Connect Endpoint
 Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

☒ Public IPv4 address

[18.208.119.93](#)

☐ IPv6 address

-

Username

Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ubuntu.

Note: In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

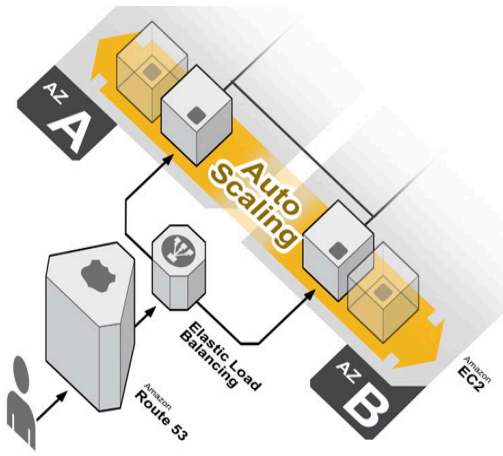
Connect

- Steps to Stress Test the CPU:
- Install Stress Tool (if not already installed):

```
ubuntu@ip-172-31-60-180:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [7196 B]
Get:6 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [52.0 kB]
Get:7 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [212 B]
Get:8 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [212 B]
Fetched 186 kB in 1s (285 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-60-180:~$
```

2. Run Stress Command:

Example to stress the CPU with 5 workers for 60 seconds:



```
ubuntu@ip-172-31-60-180:~$ sudo apt-get install stress -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  stress
0 upgraded, 1 newly installed, 0 to remove and 58 not upgraded.
Need to get 18.1 kB of archives.
After this operation, 52.2 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 stress amd64 1.0.7-1 [18.1 kB]
Fetched 18.1 kB in 0s (1195 kB/s)
Selecting previously unselected package stress.
(Reading database ... 70649 files and directories currently installed.)
Preparing to unpack .../stress_1.0.7-1_amd64.deb ...
Unpacking stress (1.0.7-1) ...
Setting up stress (1.0.7-1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...
```

```
Running kernel seems to be up-to-date.
```

```
No services need to be restarted.
```

```
No containers need to be restarted.
```

```
No user sessions are running outdated binaries.
```

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
```

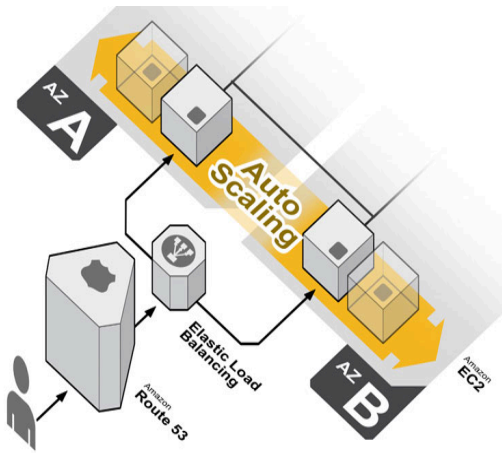
```
ubuntu@ip-172-31-60-180:~$
```

```
ubuntu@ip-172-31-60-180:~$ stress --cpu 5 --timeout 50
stress: info: [2583] dispatching hogs: 5 cpu, 0 io, 0 vm, 0 hdd
```

3. Check CPU Usage (Optional):

Use the top or htop command in another terminal session to monitor the CPU usage:

```
top
```

4. Stop Stress Test:

If you want to stop the stress test before it completes, press **Ctrl+C** in the terminal running the stress command.

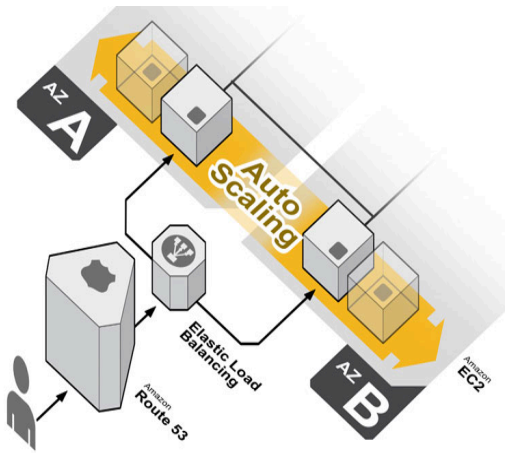
5. Monitor Scaling Activity:

- Go to the **Activity** tab in the Auto Scaling Group to track instance launches and terminations.

Welcome RUTURAJ this is ip-172-31-52-214

Auto Scaling group: firstautoscalinggroup

✓ Successful	Launching a new EC2 instance: i-087c1c92db238bdee	At 2024-12-31T12:02:55Z an instance was launched in response to an unhealthy instance needing to be replaced.	2024 December 31, 05:32:57 PM +05:30	2024 December 31, 05:33:59 PM +05:30
✓ Successful	Terminating EC2 instance: i-06866f702eb96fa95	At 2024-12-31T12:02:55Z an instance was taken out of service in response to an EC2 health check indicating it has been terminated or stopped.	2024 December 31, 05:32:55 PM +05:30	2024 December 31, 05:33:17 PM +05:30
✓ Successful	Terminating EC2 instance: i-0568a654eb39a12a0	At 2024-12-31T10:44:38Z a user request update of AutoScalingGroup constraints to min: 1, max: 10, desired: 1 changing the desired capacity from 4 to 1. At 2024-12-31T10:44:49Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 4 to 1. At 2024-12-31T10:44:49Z instance i-00afb7f1b42409be2 was selected for termination. At 2024-12-31T10:44:49Z instance i-0121302bf0e1964db was selected for termination. At 2024-12-31T10:44:49Z instance i-0568a654eb39a12a0 was selected for termination.	2024 December 31, 04:14:49 PM +05:30	2024 December 31, 04:15:51 PM +05:30



✓ Successful	Terminating EC2 instance: i-0121302bf0e1964db	At 2024-12-31T10:44:38Z a user request update of AutoScalingGroup constraints to min: 1, max: 10, desired: 1 changing the desired capacity from 4 to 1. At 2024-12-31T10:44:49Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 4 to 1. At 2024-12-31T10:44:49Z instance i-00afb7f1b42409be2 was selected for termination. At 2024-12-31T10:44:49Z instance i-0121302bf0e1964db was selected for termination. At 2024-12-31T10:44:49Z instance i-0568a654eb39a12a0 was selected for termination.	2024 December 31, 04:14:49 PM +05:30	2024 December 31, 04:16:12 PM +05:30
✓ Successful	Terminating EC2 instance: i-00afb7f1b42409be2	At 2024-12-31T10:44:38Z a user request update of AutoScalingGroup constraints to min: 1, max: 10, desired: 1 changing the desired capacity from 4 to 1. At 2024-12-31T10:44:49Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 4 to 1. At 2024-12-31T10:44:49Z instance i-00afb7f1b42409be2 was selected for termination. At 2024-12-31T10:44:49Z instance i-0121302bf0e1964db was selected for termination. At 2024-12-31T10:44:49Z instance i-0568a654eb39a12a0 was selected for termination.	2024 December 31, 04:14:49 PM +05:30	2024 December 31, 04:16:11 PM +05:30

Auto Scaling group: firstautoscalinggroup



✓ Successful	Terminating EC2 instance: i-04ccefe41ed7c9842	At 2024-12-31T10:42:18Z a monitor alarm TargetTracking-firstautoscalinggroup-AlarmLow-beb7f08f-409a-435c-9246-d9cd7213edc4 in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 7 to 6. At 2024-12-31T10:42:29Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 7 to 6. At 2024-12-31T10:42:29Z instance i-04ccefe41ed7c9842 was selected for termination.	2024 December 31, 04:12:29 PM +05:30	2024 December 31, 04:15:11 PM +05:30
✓ Successful	Launching a new EC2 instance: i-06866f702eb96fa95	At 2024-12-31T10:25:45Z a monitor alarm TargetTracking-firstautoscalinggroup-AlarmHigh-a04c62fb-1111-4026-8468-9439308b2e3f in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 3 to 7. At 2024-12-31T10:25:53Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 3 to 7.	2024 December 31, 03:55:55 PM +05:30	2024 December 31, 04:01:26 PM +05:30
✓ Successful	Launching a new EC2 instance: i-0568a654eb39a12a0	At 2024-12-31T10:25:45Z a monitor alarm TargetTracking-firstautoscalinggroup-AlarmHigh-a04c62fb-1111-4026-8468-9439308b2e3f in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 3 to 7. At 2024-12-31T10:25:53Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 3 to 7.	2024 December 31, 03:55:55 PM +05:30	2024 December 31, 04:01:11 PM +05:30
✓ Successful	Terminating EC2 instance: i-0978e66f1479d1fd7	At 2024-12-31T10:44:18Z a monitor alarm TargetTracking-firstautoscalinggroup-AlarmLow-beb7f08f-409a-435c-9246-d9cd7213edc4 in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 5 to 4. At 2024-12-31T10:44:28Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 5 to 4. At 2024-12-31T10:44:28Z instance i-0978e66f1479d1fd7 was selected for termination.	2024 December 31, 04:14:28 PM +05:30	2024 December 31, 04:15:30 PM +05:30
✓ Successful	Terminating EC2 instance: i-0ad832a04c69c0b42	At 2024-12-31T10:43:18Z a monitor alarm TargetTracking-firstautoscalinggroup-AlarmLow-beb7f08f-409a-435c-9246-d9cd7213edc4 in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 6 to 5. At 2024-12-31T10:43:28Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 6 to 5. At 2024-12-31T10:43:28Z instance i-0ad832a04c69c0b42 was selected for termination.	2024 December 31, 04:13:28 PM +05:30	2024 December 31, 04:14:51 PM +05:30