

💡 AUTOSCALING

Autoscaling is an advanced feature of AWS which will automatically do resource management based on server load.

🔴 Major Components:

- ◆ EC2 instance - Virtual server exists in ec2 , applications are deployed through this.
- ◆ Autoscaling group - collection of EC2 instances and policies , adds/removes instances depend on the load.
- ◆ AMI - Amazon Machine Image - It provides all information required to launch new instances. Multiple instances can be launched from one AMI.
- ◆ Load Balancer - It is used to increase the capacity and reliability of applications. The main function is to divide traffic among instances.

Now let's start the hands-on.....

✅ For creating first instance , setup as following:

Name and tags
Info

Name

Webserver1

Add additional tags

▼ Application and OS Images (Amazon Machine Image)
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

Recents

My AMIs

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE L

SUS

Q

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

Free tier eligible

ami-069d73f3235b535bd (64-bit (x86)) / ami-0e31d4ddf8c30fd2a (64-bit (Arm))

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

Description

Amazon Linux 2023 AMI 2023.1.20230705.0 x86_64 HVM kernel-6.1

Architecture

AMI ID

64-bit (x86)

ami-069d73f3235b535bd

Verified provider

❌ If you do not have any key pair to login , then create it. Here I'm using the existing one.

▼ Instance type [Info](#)

Instance type

t2.micro

Free tier eligible

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

On-Demand Linux pricing: 0.0116 USD per Hour

On-Demand SUSE pricing: 0.0116 USD per Hour

On-Demand Windows pricing: 0.0162 USD per Hour

On-Demand RHEL pricing: 0.0716 USD per Hour

☒ All generations

[Compare instance types](#)

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

myec2instance

[↻ Create new key pair](#)

▼ Network settings [Info](#)

VPC - *required* [Info](#)

vpc-0119ae136d2e37942 (Default VPC) (default) ▼



Subnet [Info](#)

No preference ▼



[Create new subnet](#)

Auto-assign public IP [Info](#)

Enable ▼

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.

☒ Create security group

☐ Select existing security group

Security group name - *required*

example-loadbalancer

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](#)

example-loadbalancer

Inbound Security Group Rules

▶ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Remove

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0)

Remove

Type [Info](#)

HTTP ▼

Protocol [Info](#)

TCP

Port range [Info](#)

80

Source type [Info](#)

Anywhere ▼

Source [Info](#)

Add CIDR, prefix list or security

Description - *optional* [Info](#)

e.g. SSH for admin desktop

▼ **Configure storage** [Info](#)

Advanced

1x GiB ▼

Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

×

Add new volume

0 x File systems

Edit

✖ Under Advanced Details

User data - *optional* [Info](#)

Upload a file with your user data or enter it in the field.

Choose file

```
#!/bin/bash
yum install httpd -y
service httpd start
chkconfig httpd on
hostname > /var/www/html/index.html
```

☐ User data has already been base64 encoded

✖ Launch another instance as same as above , but select the load balancer which we have created above by selecting the existing security group.

▼ Network settings

Info

VPC - required

Info

vpc-0119ae136d2e37942 (Default VPC)

172.31.0.0/16

(default) ▼

↻

Subnet

Info

No preference ▼

↻

Create new subnet

🔗

Auto-assign public IP

Info

Enable ▼

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.

☐ Create security group

☒ Select existing security group

Common security groups

Info

Select security groups ▼

↻

Compare security group rules

example-loadbalancer sg-0f78afe5258262a7b ✕

VPC: vpc-0119ae136d2e37942

Security groups that you add or remove here will be added to or removed from all your network interfaces.

✕ Now we have 2 instances , and we are going to configure Load Balancer

✕ From left side click on load balancer , as we're on learning stage , let us select Classic Load Balancer

Select load balancer type

Elastic Load Balancing supports four types of load balancers: Application Load Balancers, Network Load Balancers, Gateway Load Balancers, and Classic Load Balancers. Choose the load balancer type that meets your needs.
Learn more about which load balancer is right for you

Application Load Balancer	Network Load Balancer	Gateway Load Balancer	Classic Load Balancer
<div>HTTP HTTPS</div> <div>Create</div> <p>Choose an Application Load Balancer when you need a flexible feature set for your web applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.</p> <p>Learn more ></p>	<div>TCP TLS UDP</div> <div>Create</div> <p>Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your application. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.</p> <p>Learn more ></p>	<div>IP</div> <div>Create</div> <p>Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.</p> <p>Learn more ></p>	<div>PREVIOUS GENERATION for HTTP, HTTPS, and TCP</div> <div>Create</div> <p>Choose a Classic Load Balancer when you have an existing application running in the EC2-Classical network.</p> <p>Learn more ></p>

1. Define Load Balancer 2. Assign Security Groups 3. Configure Security Settings 4. Configure Health Check 5. Add EC2 Instances 6. Add Tags 7. Review

Step 1: Define Load Balancer

Basic Configuration

This wizard will walk you through setting up a new load balancer. Begin by giving your new load balancer a unique name so that you can identify it from other load balancers you might create. You will also need to configure ports and protocols for your load balancer. Traffic from your clients can be routed from any load balancer port to any port on your EC2 instances. By default, we've configured your load balancer with a standard web server on port 80.

Load Balancer name:

Create LB inside: Default VPC

Create an internal load balancer: ☐ (what's this?)

Enable advanced VPC configuration: ☐

Listener Configuration:

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port
<input type="text" value="HTTP"/>	<input type="text" value="80"/>	<input type="text" value="HTTP"/>	<input type="text" value="80"/>

Add

✖ From above the first protocol (load balancer) will be the front-end and Instance protocol will be our server , which we're going to configure (eg: if we're going to configure apache , then port number will be 8080).

Step 2: Assign Security Groups

You have selected the option of having your Elastic Load Balancer inside of a VPC, which allows you to assign security groups to your load balancer. Please select the security groups to assign to this load balancer. This can be changed at any time.

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

Security Group ID	Name	Description	Actions
<input type="checkbox"/> sg-034937d9741e0d5d3d	default	default VPC security group	Copy to new
<input checked="" type="checkbox"/> sg-9f78afe52562a7b7	example-loadbalancer	example-loadbalancer	Copy to new

Load balancer will automatically perform health checks on your EC2 instances and only route traffic to instances that pass the health check. If an instance fails the health check, it is automatically removed from the load balancer.

Step 4: Configure Health Check

Your load balancer will automatically perform health checks on your EC2 instances and only route traffic to instances that pass the health check. If an instance fails the health check, it is automatically removed from the load balancer. Customize the health check to meet your specific needs.

Ping Protocol:

Ping Port:

Ping Path:

Advanced Details

Response Timeout: seconds

Interval: seconds

Unhealthy threshold:

Healthy threshold:

✖ Load Balancer will check the server on port 80 to know the alive status. If more than 2 unsuccessful attempts , then load balancer will not send any further request to the server , but it will keep on checking.

Step 5: Add EC2 Instances

The table below lists all your running EC2 instances. Check the boxes in the Select column to add those instances to this load balancer.

VPC: vpc-6119ae136d2e37942 (172.31.0.0/16) | Default VPC

Instance	Name	State	Security groups	Zone	Subnet ID	Subnet CIDR
<input checked="" type="checkbox"/> i-01eca629162083063	Webserver1	running	example-loadbalancer	us-east-2c	subnet-05cb742...	172.31.32.0/29
<input checked="" type="checkbox"/> i-0d9112c030e8b875	Webserver2	running	example-loadbalancer	us-east-2c	subnet-05cb742...	172.31.32.0/29

Availability Zone Distribution

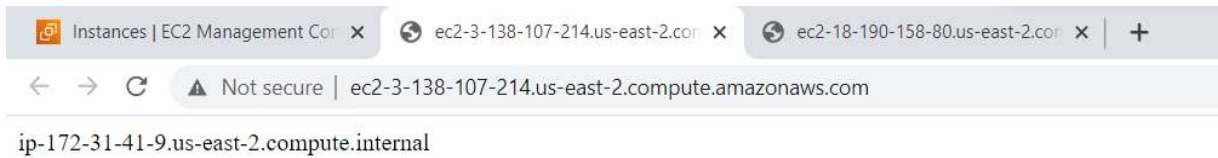
2 instances in us-east-2c

☒ Enable Cross-Zone Load Balancing

☒ Enable Connection Draining seconds

✖ Now goto instances and take the public IP of 2 instances and paste it in the browser on different tabs: We can see the difference in the output.

✖ Webserver2



✖ Webserver1



✖ Now goto the load balancer and open DNS name in a browser and keep reloading it. We can see the IP address is changing.

Name	Webserver1
* DNS name	Webserver1-990436367.us-east-2.elb.amazonaws.com (A Record)

✖ Now we are going to terminate the 2 instances which we have created above. Autoscaling should start with an empty load balancer.

✖ First create a launch template , for that :

✖ From EC2 - click on Launch Templates and then click on create launch template.

✖ Provide the launch template name as per your wish , here I gave Webserver.

✖ Then we need to provide an AMI , for that to take AMI id from the instance or click on the launch instance and copy the AMI ID from there.

✖ Example of AMI ID to be copied - ami-069d73f3235b535bd

▼ Application and OS Images (Amazon Machine Image) [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

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

[AMI from catalog](#)

[Recents](#)

[My AMIs](#)

[Quick Start](#)

Amazon Machine Image (AMI)

al2023-ami-2023.1.20230705.0-kernel-6.1-
x86_64
ami-069d73f3235b535bd

Verified provider

Free tier eligible



[Browse more AMIs](#)

Including AMIs from
AWS, Marketplace and
the Community

Catalog	Published	Architecture	Virtualization	Root device type	ENA Enabled
Quickstart AMIs	2023-07-05T18:00:11.000Z	x86_64	hvm	ebs	Yes

▼ Instance type [Info](#)

[Advanced](#)

Instance type

t2.micro

Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux pricing: 0.0116 USD per Hour
On-Demand SUSE pricing: 0.0116 USD per Hour
On-Demand Windows pricing: 0.0162 USD per Hour
On-Demand RHEL pricing: 0.0716 USD per Hour

☒ All generations

[Compare instance types](#)

✘ Select the existing key pair and security group as load balancer name

▼
Key pair (login)
Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name

myec2instance ▼

[Create new key pair](#)

▼
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 ▼

example-loadbalancer sg-0f78afe5258262a7b ✕

VPC: vpc-0119ae136d2e37942

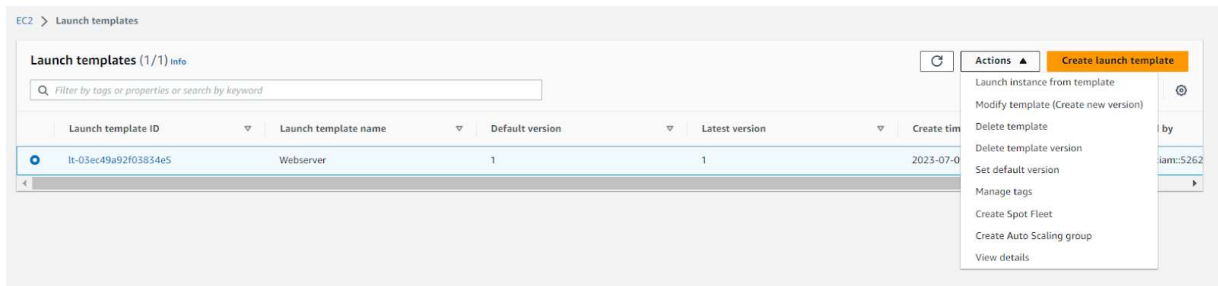
[Compare security group rules](#)

► Advanced network configuration

✕ Now paste the below into User data coming under Advanced details:

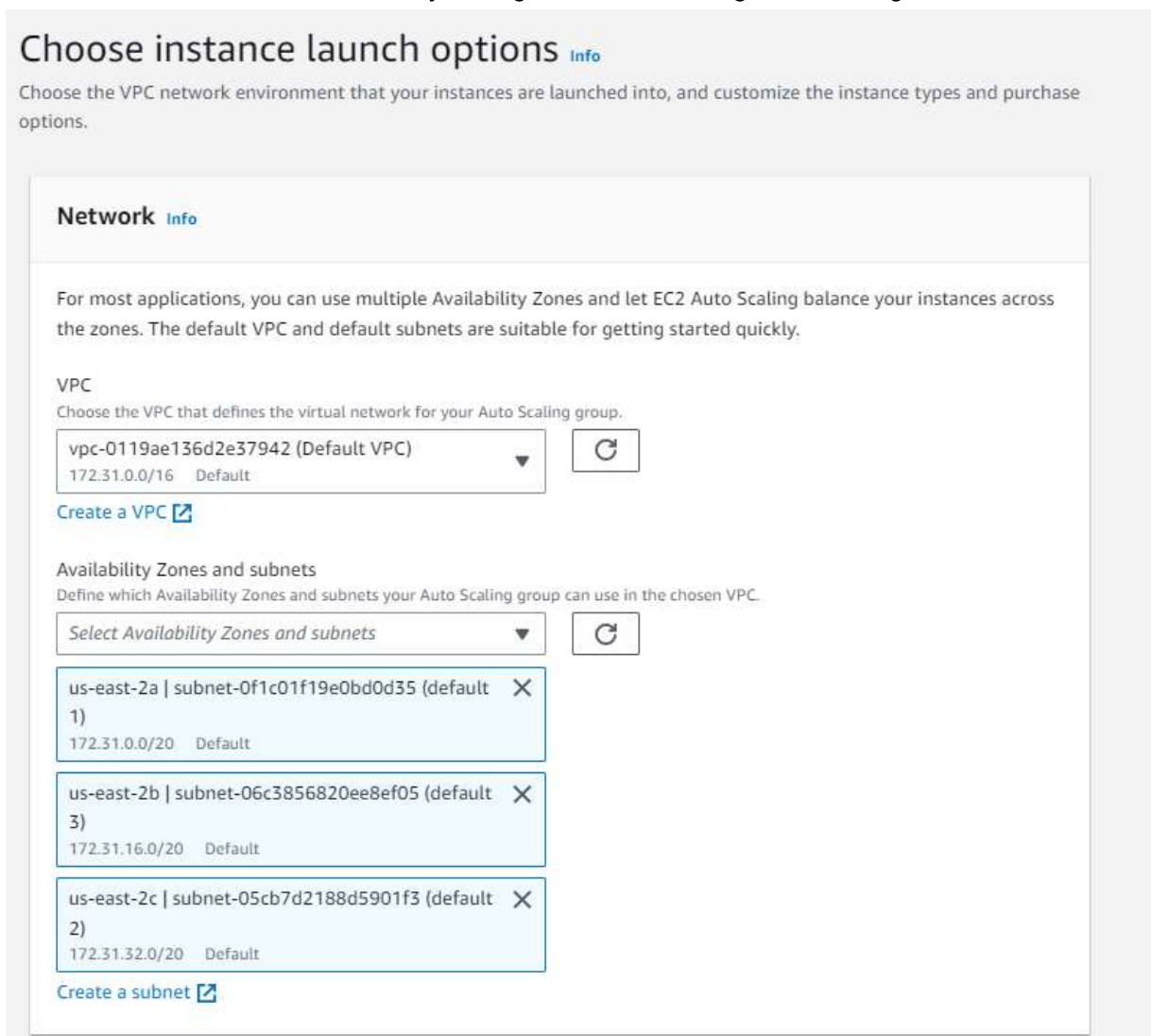
```
#!/bin/bash
yum install httpd -y
service httpd start
chkconfig httpd on
hostname > /var/www/html/index.html
```

✕ Then click on create launch template.



✕ We are going to create auto scaling group , select the template we have created and goto Actions and select create Auto Scaling Group . Please find this from the above image.

✕ Select all AZs and subnets from your region, here I'm using the Ohio region.



✕ Choose existing load balance as following below steps:

Configure advanced options - *optional* [Info](#)

Integrate your Auto Scaling group with other services to distribute network traffic across multiple servers using a load balancer or to establish service-to-service communications using VPC Lattice. You can also set options that give you more control over health check replacements and monitoring.

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

Classic Load Balancers

Select Classic Load Balancers

Webservers ×
Classic Load Balancer

✖ And reduce the health check grace period from 300 to 150.

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 **Recommended**

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

✖ In below , we are giving group size and scaling policies :

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

Scaling policy name

Metric type



Target value

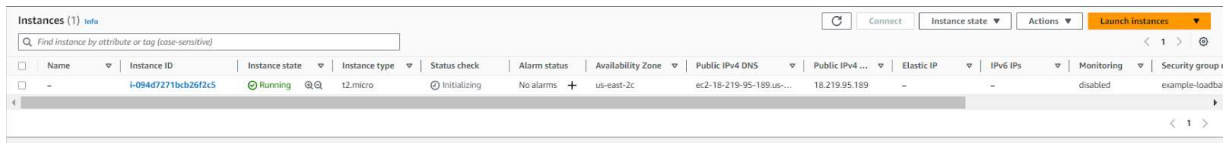
Instances need

seconds warm up before including in metric

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

✖ Now we have created an Auto Scaling Group.

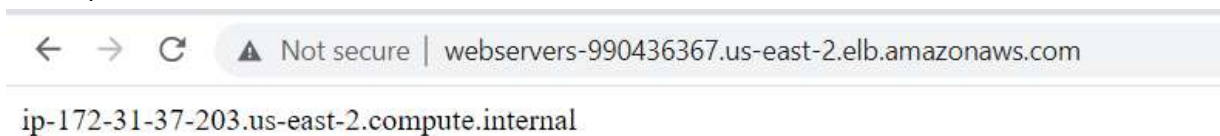
✗ Now go to Instances and refresh the browser. We can see one instance has been created as we have specified the minimum capacity as one



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs	Monitoring	Security group
-	i-094d7271bcb26f2c5	Running	t2.micro	Initializing	No alarms	us-east-2c	ec2-18-219-95-189.us-...	18.219.95.189	-	-	disabled	example-loadba

✗ Please see below to verify the load balancer and newly created instance is syncing:

✗ Output of load balancer DNS name



✗ Output of newly created instance private IP

Instance: i-094d7271bcb26f2c5

Details

Security

Networking

Storage

Status checks

Monitoring

Tags

▼ Instance summary info

Instance ID

i-094d7271bcb26f2c5

Public IPv4 address

18.219.95.189 | [open address](#)

Private IPv4 addresses

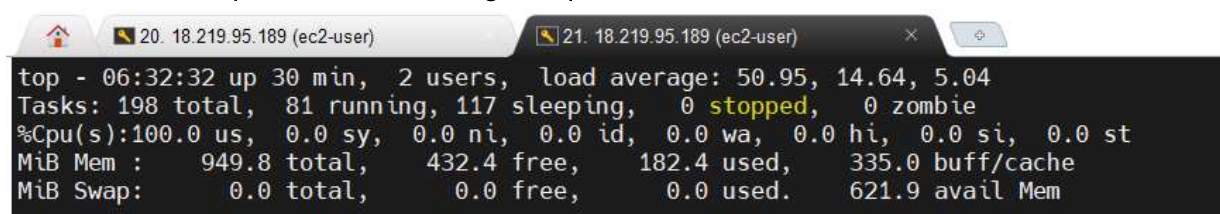
172.31.37.203

✗ Let us give a CPU load to the first instance and see the new instances are created automatically.

✗ Now take the public IP of instance and login to it. I'm using MobaXterm to take the session and run following commands:

- ☐ yum install
https://dl.fedoraproject.org/pub/epel/7Server/x86_64/Packages/e/epel-release-7-14.noarch.rpm -y --skip-broken
- ☐ yum install stress -y
- ☐ stress --cpu 80 -----> we are giving cpu load 80

✗ Then take a duplicate session and give top command



✗ We can see CPU utilization is 100.

✗ Now goto the instances dashboard on AWS console , wait for few seconds and watch new instance is being created.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 address	Elastic IP	IPv6 IPs	Monitoring	Security groups
-	i-0a9d314e16504b5e1	Running	t2.micro	Initializing	No alarms	us-east-2a	ec2-18-119-164-159.us-east-2.amazonaws.com	18.119.164.159	-	-	disabled	example-loadbalancer
autoscaling-1	i-094d7271bcb26f2c5	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	ec2-18-219-95-189.us-east-2.amazonaws.com	18.219.95.189	-	-	disabled	example-loadbalancer

✗ Now goto Load Balancers and see the newly created instance is also attached in load balancer:

Instance ID	Name	Availability Zone	Status	Actions
i-0a9d314e16504b5e1	-	us-east-2a	InService	Remove from Load Balancer
i-094d7271bcb26f2c5	autoscaling-1	us-east-2c	InService	Remove from Load Balancer

✗ Now goto Load balancer DNS which we have copied into browser and keep reload the page. We can see the private IPs of both instances created .

✗ Private IP of second Instance created

← → ↻ ⚠ Not secure | webserver-990436367.us-east-2.elb.amazonaws.com

ip-172-31-1-65.us-east-2.compute.internal

Instance: i-0a9d314e16504b5e1		
Details Security Networking Storage Status checks Monitoring Tags		
▼ Instance summary info		
Instance ID i-0a9d314e16504b5e1	Public IPv4 address 18.119.164.159 open address	Private IPv4 addresses 172.31.1.65

✗ Private IP of first Instance created

Instance: i-094d7271bcb26f2c5 (autoscaling-1)		
Details Security Networking Storage Status checks Monitoring Tags		
▼ Instance summary info		
Instance ID i-094d7271bcb26f2c5 (autoscaling-1)	Public IPv4 address 18.219.95.189 open address	Private IPv4 addresses 172.31.37.203

← → ↻ ⚠ Not secure | webserver-990436367.us-east-2.elb.amazonaws.com

ip-172-31-37-203.us-east-2.compute.internal

● The instances will keep on creating, so once you've done practicing , goto Autoscaling and delete the auto scaling group. It will automatically delete the instances, delete the load balancer as well to avoid the costs.