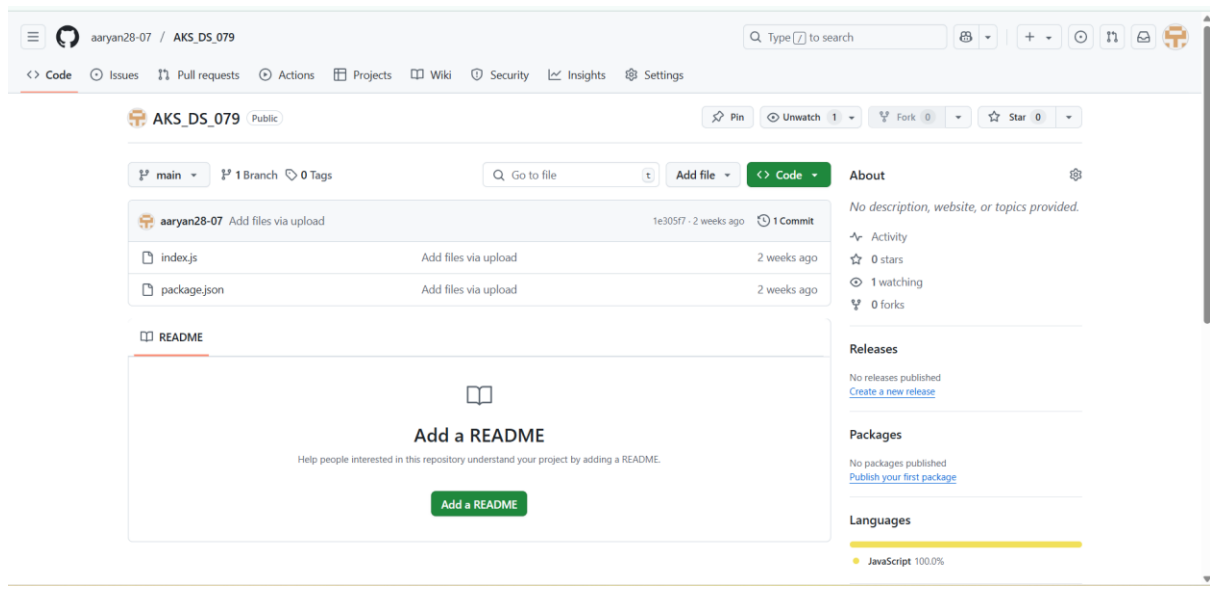


Assignment No:11

Title: Build scaling plans in AWS that balance the load on different EC2 instances.

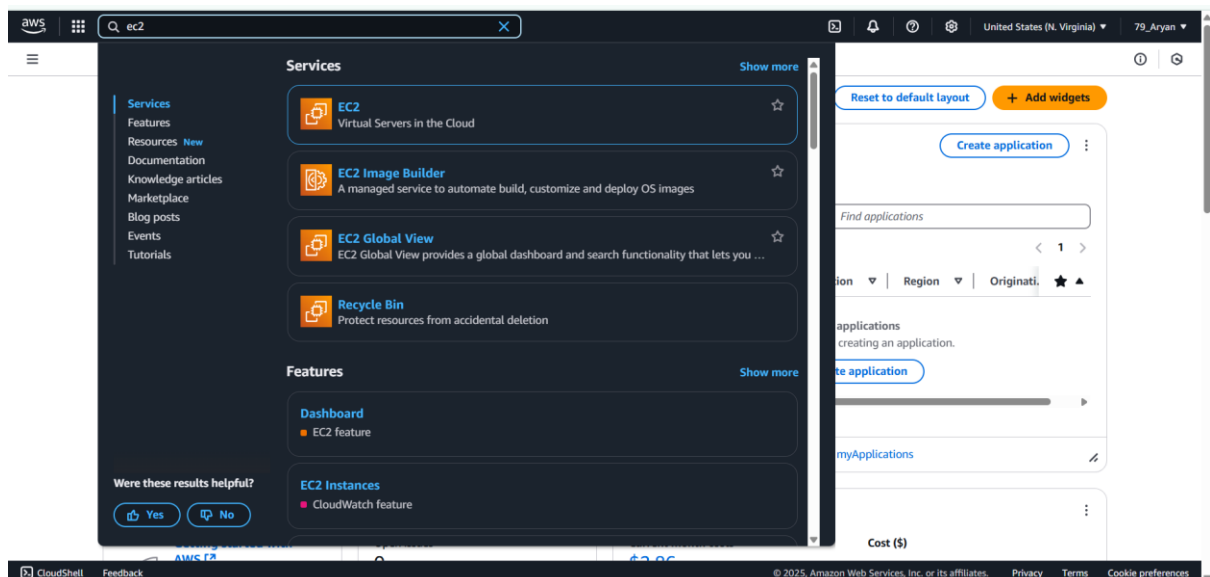
Step-1:

Upload required files to github



Step-2:

Log into AWS and open EC2.



Step-3:

Open the launch templates from left side.

aws

Search

[Alt+S]

United States (N. Virginia)

79_Aryan

EC2 > Launch templates > Create launch template

▼ Instance type Info | Get advice

Advanced

Instance type

t2.micro

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

On-Demand Windows base pricing: 0.0162 USD per Hour

On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour

On-Demand SUSE base pricing: 0.0116 USD per Hour On-Demand RHEL base pricing: 0.026 USD per Hour

On-Demand Linux base pricing: 0.0116 USD per Hour

Free tier eligible

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

Ak079

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

▼ Summary

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd64...read more

ami-084568db438326404

Virtual server type (instance type)

t2.micro

Firewall (security group)

-

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Create launch template

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Search

[Alt+S]

United States (N. Virginia)

79_Aryan

EC2 > Launch templates > Create launch template

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
apt-get update
apt-get upgrade
apt-get install -y nginx
systemctl start nginx
systemctl enable nginx
apt-get install -y git
curl -sL https://deb.nodesource.com/setup_18.x | sudo -E bash -
apt-get install -y nodejs
git clone https://github.com/aaryan28-07/AKS_DS_079.git
cd AKS_DS_079
npm install
node index.js

User data has already been base64 encoded

▼ Summary

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd64...read more

ami-084568db438326404

Virtual server type (instance type)

t2.micro

Firewall (security group)

-

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Create launch template

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Search

[Alt+S]

United States (N. Virginia)

79_Aryan

EC2

Dashboard

EC2 Global View

Events

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

▼ Images

AMIs

AMI Catalog

▼ Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

Launch Templates (1) Info

Actions

Create launch template

Search

Launch Template ID	Launch Template Name	Default Version	Latest Version	Create Time	Created By
lt-05eee6fbb27d8cd81	AryanTemp	1	1	2025-04-08T18:28:29.000Z	arn:aws:iam::8673

Select a launch template

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step-5:

Next open the auto scaling groups and click on create auto scaling group. Then name the group then select the launch template that we just created then select availability zones then click next.

The screenshot displays the AWS Management Console interface for creating an Amazon EC2 Auto Scaling group. The top navigation bar shows the AWS logo, a search bar, and the user's profile (79_Aryan). The breadcrumb trail indicates the path: EC2 > Auto Scaling groups > Create Auto Scaling group.

The main content area features a large hero section with the text "Amazon EC2 Auto Scaling helps maintain the availability of your applications". Below this, a "Create Auto Scaling group" button is prominently displayed. To the right, a "Pricing" section explains that there are no additional fees beyond service fees for Amazon EC2, CloudWatch, and other AWS resources. A "Getting started" link is also present.

The "How it works" section includes a diagram illustrating the Auto Scaling group architecture, showing a central "Auto Scaling group" box connected to four EC2 instances (represented by squares). Below the diagram, a "Pricing" section states that Amazon EC2 Auto Scaling features have no additional fees beyond the service fees for Amazon EC2, CloudWatch (for scaling policies), and other AWS resources that you use. A "Getting started" link is also provided.

The "Create Auto Scaling group" wizard is shown in the lower half of the screenshot. The first step, "Choose launch template", is selected. The wizard includes a sidebar with steps: Step 1: Choose launch template (selected), Step 2: Choose instance launch options, Step 3 - optional: Integrate with other services, Step 4 - optional: Configure group size and scaling, Step 5 - optional: Add notifications, Step 6 - optional: Add tags, and Step 7: Review.

The "Choose launch template" step includes a "Name" section where the "Auto Scaling group name" is entered as "Aryan_Scale". Below this, a "Launch template" section shows the selected template "AryanTemp" and its version "Default (1)". A note indicates that for accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. A "Create a launch template" link is provided.

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-021f527247516bf8a
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-04f5a5f0f21bc97b7
172.31.32.0/20 Default

ap-south-1b | subnet-0645a9154debd190c
172.31.0.0/20 Default

ap-south-1c | subnet-06aa0259d60b115ab
172.31.16.0/20 Default

[Create a subnet](#)

Availability Zone distribution - new
Auto Scaling automatically balances instances across Availability Zones. If launch failures occur in a zone, select a strategy.

☒ **Balanced best effort**
If launches fail in one Availability Zone, Auto Scaling will attempt to launch in another healthy Availability Zone.

☐ **Balanced only**
If launches fail in one Availability Zone, Auto Scaling will continue to attempt to launch in the unhealthy Availability Zone to preserve balanced distribution.

Step-6:

Here attach a load balancer then network layer balancer after that name the load balancer and create a target group then attach a vpc lattice service and 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 a new load balancer
Define a new load balancer to create for attachment to this Auto Scaling group.

Load balancer type
Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the [Load Balancing console](#).

☐ Application Load Balancer
HTTP, HTTPS

☒ **Network Load Balancer**
TCP, UDP, TLS

Load balancer name
Name cannot be changed after the load balancer is created.

AryanScale-1

Load balancer scheme
Scheme cannot be changed after the load balancer is created.

☒ **Internal**

☐ Internet-facing

Network mapping

Listeners and routing

If you require secure listeners, or multiple listeners, you can configure them from the [Load Balancing console](#) after your load balancer is created.

Protocol TCP **Port** 80 **Default routing (forward to)** Create a target group

New target group name
An instance target group with default settings will be created.
AryanScale-1

Tags - optional
Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them.
[Add tag](#)
50 remaining

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](#)

Choose VPC Lattice target group
Only VPC Lattice target groups with a target type of instance that are in the same VPC as your Auto Scaling group are available for selection (vpc-021f327247516bf8a).

Step-7:

After that specify the desired capacity (here 2) then give min and max desired capacity and then select target tracking scaling policy and give the policy a name then select the policy for creating new server minimum load. Next and then review it and create the group.

Group size
Specify your group size.
2

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 2 **Max desired capacity** 3
Equal or less than desired capacity Equal or greater than desired capacity

Automatic scaling - optional
Choose whether to use a target tracking policy [info](#)
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

☐ No scaling policies
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

☒ Target tracking scaling policy
Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Scaling policy name
Target Tracking Policy

Metric type [info](#)
Monitored metric that determines if resource utilization is too low or high. If using EC2 metrics, consider enabling detailed monitoring for better scaling performance.

Average CPU utilization

aws

Search

[Alt+S]

Asia Pacific (Mumbai)

79_Aryan

EC2

Auto Scaling groups

Create Auto Scaling group

No scaling policies

Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

Target tracking scaling policy

Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Instance maintenance policy

Info

Control your Auto Scaling group's availability during instance replacement events. This includes health checks, instance refreshes, maximum instance lifetime features and events that happen automatically to keep your group balanced, called rebalancing events.

Choose a replacement behavior depending on your availability requirements

Mixed behavior

No policy

For rebalancing events, new instances will launch before terminating others. For all other events, instances terminate and launch at the same time.

Prioritize availability

Launch before terminating

Launch new instances and wait for them to be ready before terminating others. This allows you to go above your desired capacity by a given percentage and may temporarily increase costs.

Control costs

Terminate and launch

Terminate and launch instances at the same time. This allows you to go below your desired capacity by a given percentage and may temporarily reduce availability.

Flexible

Custom behavior

Set custom values for the minimum and maximum amount of available capacity. This gives you greater flexibility in setting how far below and over your desired capacity EC2 Auto Scaling goes when replacing instances.

Set healthy percentage

Set the minimum percentage of the desired capacity that must be healthy and ready for use for EC2 Auto Scaling to proceed with replacing instances.

Min

32

% to

100

% of 2 instances

ⓘ Your group's scaling limits will be temporarily exceeded based on current calculations.

► View capacity during replacements based on your desired capacity

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Search

[Alt+S]

Asia Pacific (Mumbai)

79_Aryan

EC2

Auto Scaling groups

Create Auto Scaling group

Capacity Reservation preference

Preference

Default

Capacity Reservation IDs

-

Resource Groups

-

Step 5: Add notifications

Edit

Notifications

Notification 1

SNS Topic

Event types

☒ Launch

☒ Terminate

☒ Fail to launch

☒ Fail to terminate

Step 6: Add tags

Edit

Tags (0)

Key	Value	Tag new instances
No tags		

Preview code

Cancel

Previous

Create Auto Scaling group

aws

Search

[Alt+S]

Asia Pacific (Mumbai)

79_Aryan

EC2

Auto Scaling groups

Auto Scaling groups (1/1)

Info

Launch configurations

Launch templates

Actions

Create Auto Scaling group

Search your Auto Scaling groups

<input checked="" type="checkbox"/>	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
<input checked="" type="checkbox"/>	AryanScale	AryanTemp Version Default	0	Updating capacity...	2	2	3	ap-south-1c, ap-south-1...

Auto Scaling group: AryanScale

Details

Integrations - new

Automatic scaling

Instance management

Instance refresh

Activity

Monitoring

AryanScale Capacity overview

Edit

arn:aws:autoscaling:ap-south-1:867344440469:autoScalingGroup:97d140a2-7f50-433f-a8b2-4ed3ee38bf83:autoScalingGroupName/AryanScale

Desired capacity

2

Scaling limits (Min - Max)

2 - 3

Desired capacity type

Units (number of instances)

Status

Updating capacity

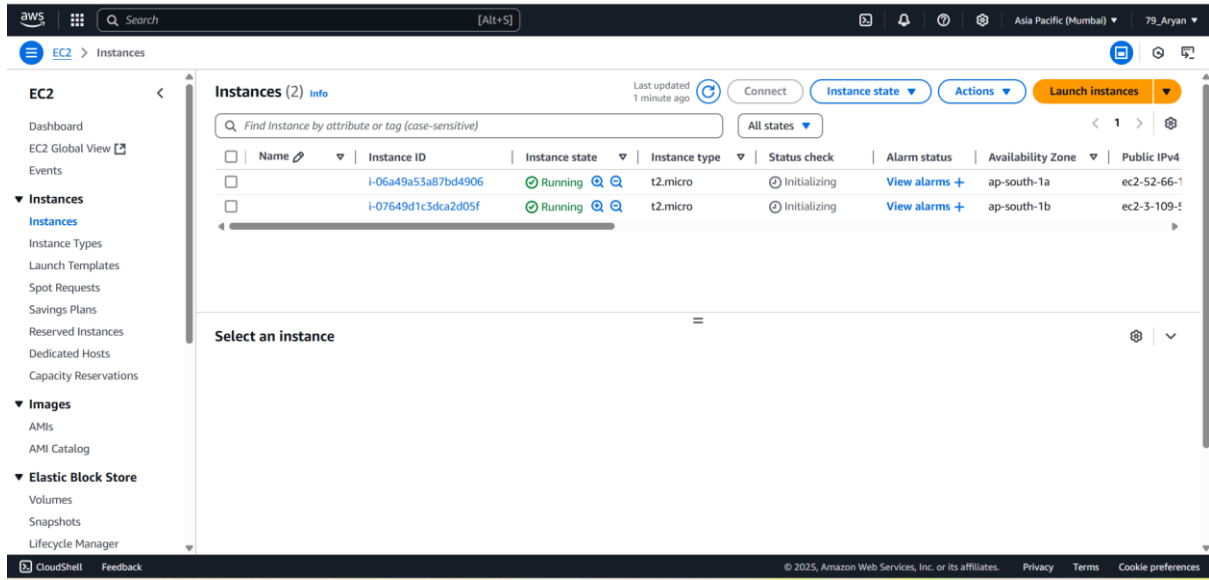
CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

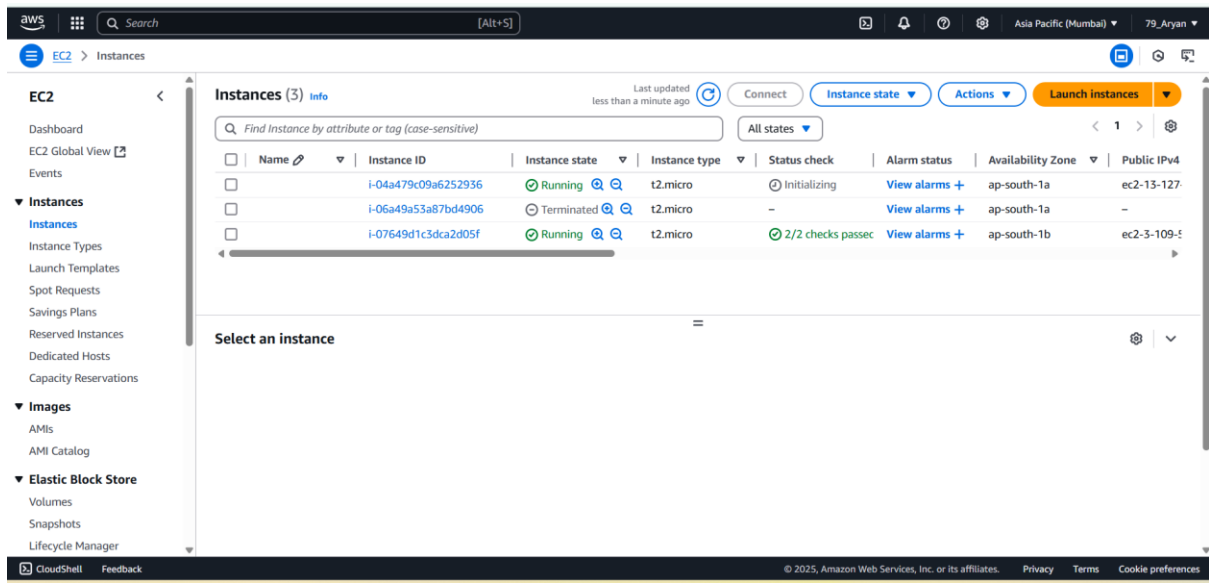
Step-8:

Now goto the instances and notice there are 2 instances automatically created. If we try to delete them new instances created automatically within few seconds.



The screenshot shows the AWS Management Console 'Instances' page. The left sidebar shows the navigation menu with 'Instances' selected. The main content area shows 'Instances (2)' with a table of two running instances. The table columns are: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4. The instances are both t2.micro type and are in the 'Running' state. The status check for both is 'Initializing'. The alarm status for both is 'View alarms +'. The availability zones are 'ap-south-1a' and 'ap-south-1b'. The public IPv4 addresses are 'ec2-52-66-1' and 'ec2-3-109-5'.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
	i-06a49a53a87bd4906	Running	t2.micro	Initializing	View alarms +	ap-south-1a	ec2-52-66-1
	i-07649d1c3dca2d05f	Running	t2.micro	Initializing	View alarms +	ap-south-1b	ec2-3-109-5

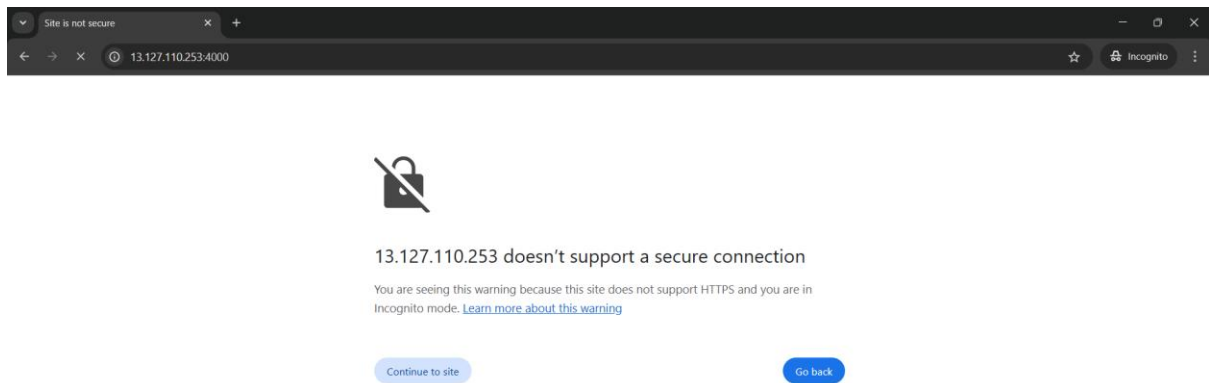


The screenshot shows the AWS Management Console 'Instances' page after a few seconds. The table now shows three instances. The first instance is still 'Running'. The second instance is now 'Terminated'. The third instance is still 'Running'. The status check for the terminated instance is '-'. The alarm status for the terminated instance is 'View alarms +'. The availability zone for the terminated instance is 'ap-south-1a'. The public IPv4 address for the terminated instance is '-'. The status check for the running instances is '2/2 checks passed'.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
	i-04a479c09a6252936	Running	t2.micro	Initializing	View alarms +	ap-south-1a	ec2-13-127-
	i-06a49a53a87bd4906	Terminated	t2.micro	-	View alarms +	ap-south-1a	-
	i-07649d1c3dca2d05f	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1b	ec2-3-109-5

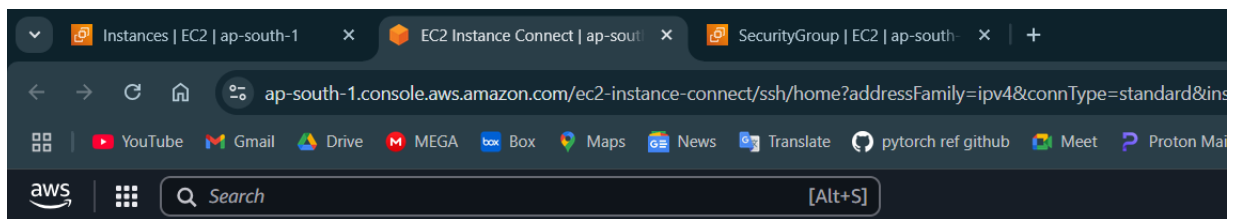
Step-9:

Copy public IPv4 of any of the running instances and open it to see the website opened.



Step-10:

Now open both instances and write the below bash or shell code and run it in them. It will increase the load in the servers.



```
GNU nano 7.2                                                                    infil.sh
while true
do
    echo "Looping"
done
```

```
Usage of /: 32.7% of 6.71GB  Users logged in: 0
Memory usage: 24%          IPv4 address for enx0: 172.31.43.171
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

49 updates can be applied immediately.
23 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-43-171:~$ nano infil.sh
ubuntu@ip-172-31-43-171:~$ chmod 777 infil.sh
ubuntu@ip-172-31-43-171:~$ nano infil.sh
ubuntu@ip-172-31-43-171:~$ ./infil.sh
```

Step-11:

After a certain threshold the servers will over load and the auto scaling group will create a new server to handle the load. Open CPU utilization graph of the

