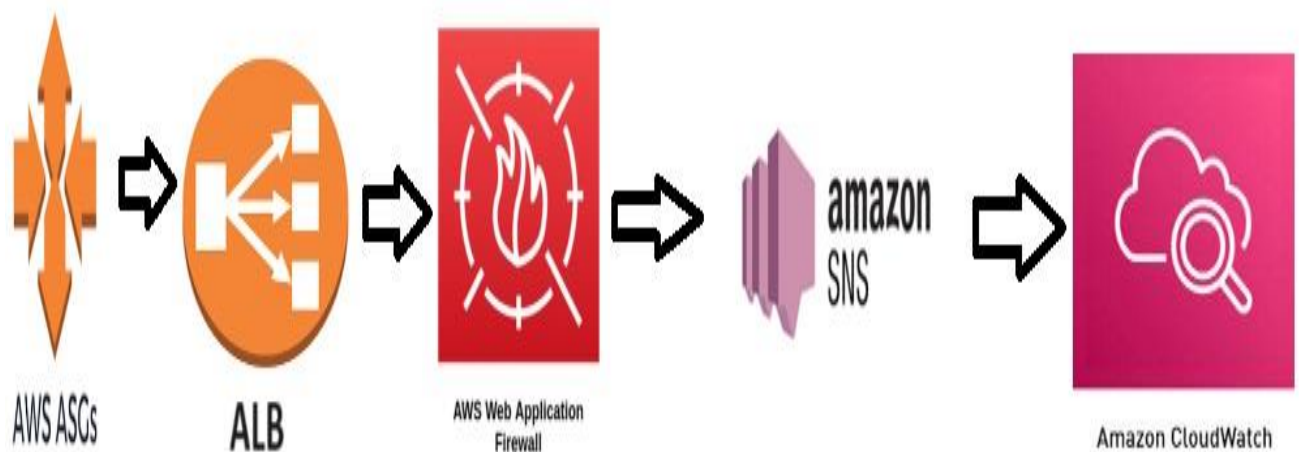




## **Mini project 1:**

**Hosting a website on AWS EC2 Instance with Auto Scaling group, Application Load Balancer and Web Application Firewall (WAF) and Monitor with Cloud Watch and Add Simple Notification Service (SNS) for it.**



# Crate launch template

## Step 1: Crate launch template.

Create launch template | EC2 | x +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateTemplate:

EC2 > Launch templates > Create launch template

### 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*

mytemp-1

Must be unique to this account. Max 128 chars. No spaces or special characters like '\$', '~', '@', ...

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

▶ Template tags

▶ Source template

**Launch template contents**

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

**Summary**

Software Image (AMI)  
Canonical, Ubuntu, 22.04 LTS, ...read more  
ami-080e1f115689ec7408

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
default

Storage (volumes)  
1 volumes - 8 GiB

Cancel Create launch template

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

## Step 2: Add user data in template

- Add web server(apache2)
- Download the website code file in it.
- Unzip the code file.

Create launch template | EC2 | x +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateTemplate:

EC2 > Launch templates > Create launch template

### 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*

mytemp-1

Must be unique to this account. Max 128 chars. No spaces or special characters like '\$', '~', '@', ...

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

▶ Template tags

▶ Source template

**Launch template contents**

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the 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
sudo apt update -y
sudo apt install apache2 -y
sudo apt install unzip -y
sudo wget https://www.free-css.com/assets/files/free-css-templates/download/page296/oxer.zip
sudo unzip oxer.zip
sudo mv oxer-html/* /var/www/html
sudo systemctl restart apache2
sudo systemctl enable apache2
```

☐ User data has already been base64 encoded

**Summary**

Software Image (AMI)  
Canonical, Ubuntu, 22.04 LTS, ...read more  
ami-080e1f115689ec7408

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
default

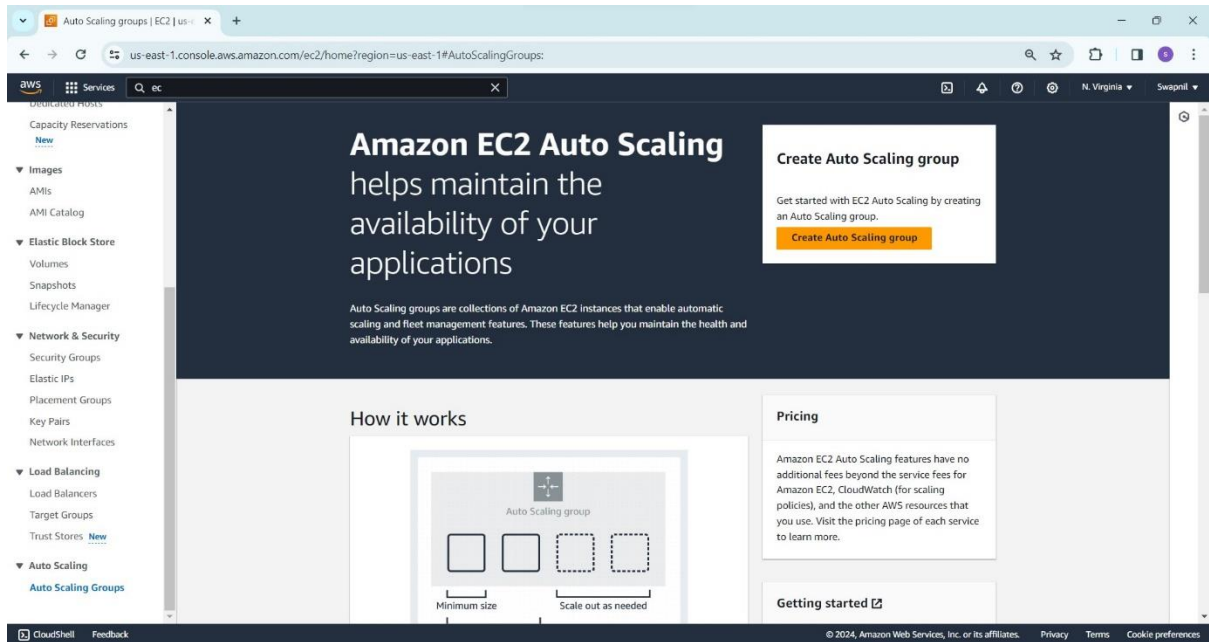
Storage (volumes)  
1 volumes - 8 GiB

Cancel Create launch template

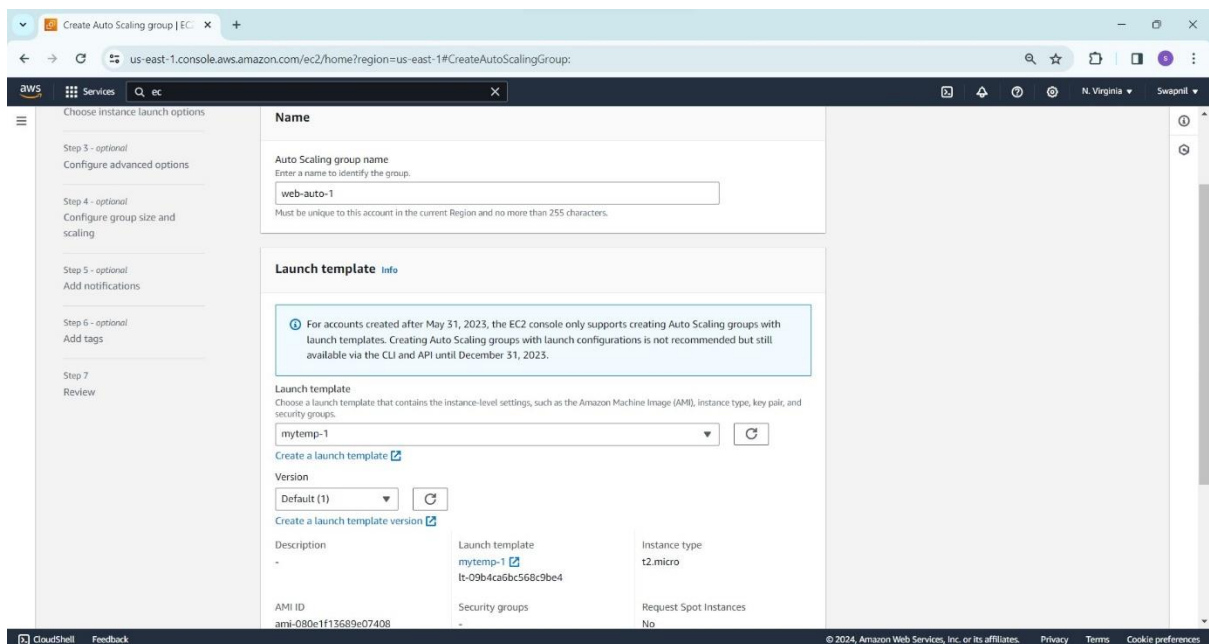
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

# Create auto scaling group from launch template.

## Step 3: Create auto scaling group.



## Step 4: Define name and template.



## Step 5: Choose instance launch options.

➤ Select availability zones and subnets.

The screenshot shows the AWS Management Console interface for creating an Auto Scaling group. The page is titled 'Create Auto Scaling group | EC2' and is in the 'us-east-1' region. The left sidebar shows the navigation menu with steps: Step 5 - optional (Add notifications), Step 6 - optional (Add tags), and Step 7 (Review). The main content area is titled 'Network info' and contains the following sections:

- Instance type:** t2.micro
- VPC:** Choose the VPC that defines the virtual network for your Auto Scaling group. The selected VPC is vpc-01cb5a8ab8705ce3c (172.31.0.0/16).
- Availability Zones and subnets:** Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC. The selected subnets are us-east-1a (subnet-0944f8f8a381654d6) and us-east-1b (subnet-02f3c70a0399e1248).

At the bottom of the page, there are buttons for 'Cancel', 'Skip to review', 'Previous', and 'Next'.

## Step 6: Configure advanced options.

➤ Add application Load balancer option.

The screenshot shows the AWS Management Console interface for creating an Auto Scaling group, specifically the 'Load balancing info' section. The page is titled 'Create Auto Scaling group | EC2' and is in the 'us-east-1' region. The left sidebar shows the navigation menu with steps: Step 3 - optional (Configure advanced options), Step 4 - optional (Configure group size and scaling), Step 5 - optional (Add notifications), Step 6 - optional (Add tags), and Step 7 (Review). The main content area is titled 'Load balancing info' and contains the following sections:

- Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.**
- Load balancing options:** Three radio buttons are present: 'No load balancer' (selected), 'Attach to an existing load balancer', and 'Attach to a new load balancer'.
- 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. The selected type is 'Application Load Balancer' (HTTP, HTTPS).
- Load balancer name:** Name cannot be changed after the load balancer is created. The name is 'web-auto-1-1'.
- Load balancer scheme:** Scheme cannot be changed after the load balancer is created. The selected scheme is 'Internet-facing'.

At the bottom of the page, there are buttons for 'Cancel', 'Skip to review', 'Previous', and 'Next'.

## Step 7: Add load balancer and target group

The screenshot shows the 'Create Auto Scaling group' page in the AWS Management Console, specifically Step 7: Add load balancer and target group. The page is for the 'us-east-1' region. Under 'Listeners and routing', the 'HTTP' protocol is selected with port '80'. The 'Default routing (forward to)' dropdown is set to 'Create a target group'. Below this, the 'New target group name' field is populated with 'web-auto-1-1'. The 'Tags - optional' section shows 'Add tag' and '50 remaining'.

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	Port	Default routing (forward to)
HTTP	80	Create a target group

New target group name

An instance target group with default settings will be created.

web-auto-1-1

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

## Step 8: Configure group size and scaling.

➤ Select the minimum, desired and maximum capacity.

The screenshot shows the 'Create Auto Scaling group' page in the AWS Management Console, specifically Step 8: Configure group size and scaling. The 'Specify your group size' field is set to '1'. Under 'Scaling limits', the 'Min desired capacity' is '1' and the 'Max desired capacity' is '4'. The 'Automatic scaling - optional' section has two radio buttons: 'No scaling policies' (selected) and 'Target tracking scaling policy'. The 'Instance maintenance policy - new' section is also visible.

Specify your group size:

1

Scaling limits

Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity	Max desired capacity
1	4

Equal or less than desired capacity

Equal or greater than desired capacity

Automatic scaling - optional

Choose whether to use a target tracking policy

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.

Instance maintenance policy - new

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.

Control availability and cost during replacement events

An instance maintenance policy determines how much availability your application has when EC2 Auto Scaling replaces instances. It also establishes guardrails that limit the amount of capacity that

## Step 9: Add notifications.

- Create a new topic.
- Enter name and endpoint user email.

The screenshot shows the 'Add notifications - optional' step in the AWS Management Console. The left sidebar lists steps 1 through 7, with 'Add notifications' selected. The main content area has a title 'Add notifications - optional' and a description: 'Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.' Below this is a 'Notification 1' section with a 'Remove' button. It contains a 'Send a notification to' field with the value 'my\_topic', a 'With these recipients' field with the value 'swap12321232@gmail.com', and a 'Use existing topic' button. There is also an 'Event types' section with checkboxes for 'Launch', 'Terminate', 'Fail to launch', and 'Fail to terminate', all of which are checked. At the bottom are buttons for 'Add notification', 'Cancel', 'Skip to review', 'Previous', and 'Next'.

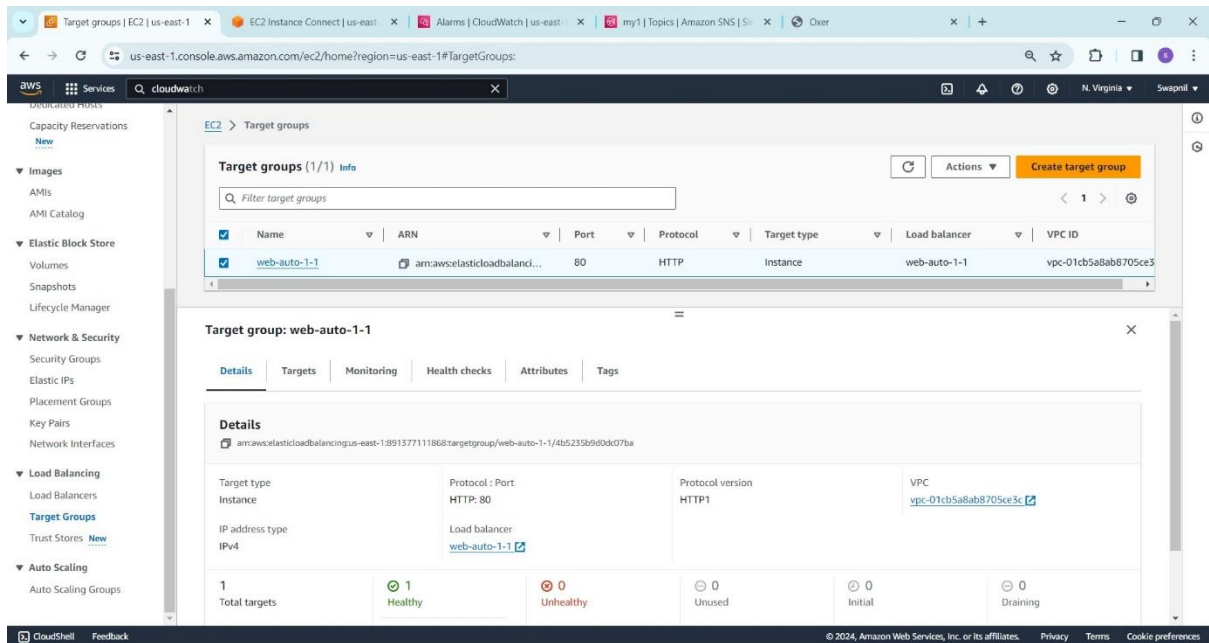
## Step 10: Add tags and review.

- Click on create Auto Scaling groups.

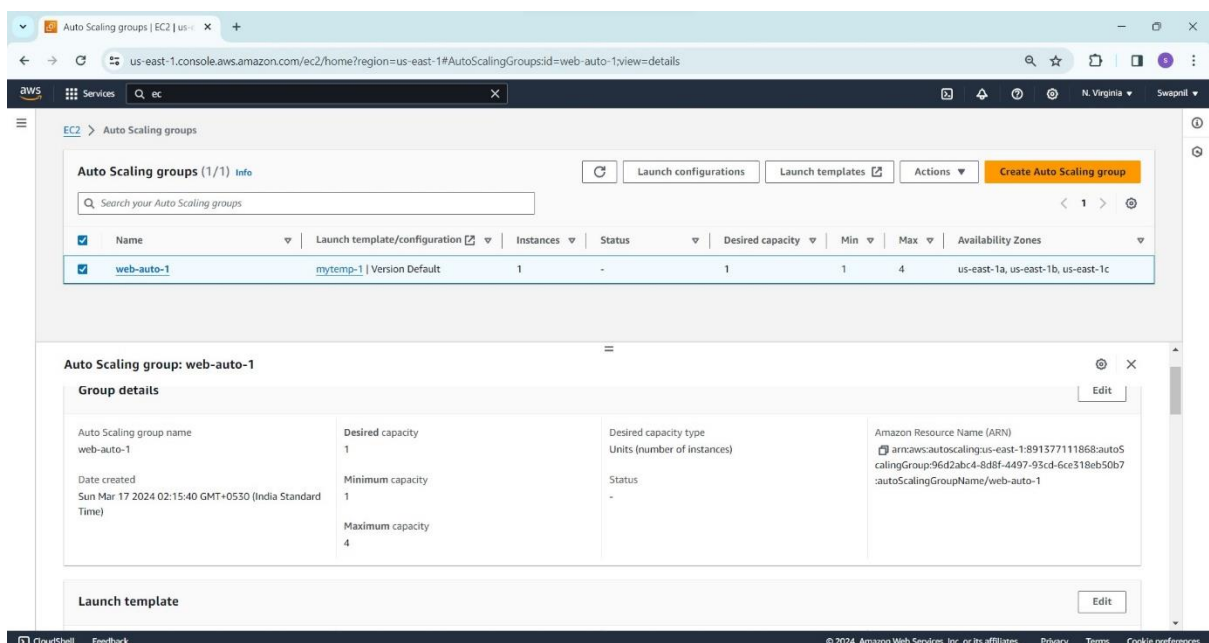
The screenshot shows the 'Add tags' step in the AWS Management Console. The left sidebar lists steps 1 through 7, with 'Add tags' selected. The main content area has a title 'Add tags' and a description: 'Add tags to your Auto Scaling group. Tags are labels that you can use to categorize and organize your resources. Each tag consists of a key and a value.' Below this is a 'Tags (0)' section with a table for adding tags. The table has columns for 'Key', 'Value', and 'Tag new instances'. There is a 'No tags' message below the table. At the bottom are buttons for 'Cancel', 'Previous', and 'Create Auto Scaling group'.

# Application load balancer is automatically created

## Step 11: New target group is automatically created.



## Step 12: New Application Load Balancer is created automatically.

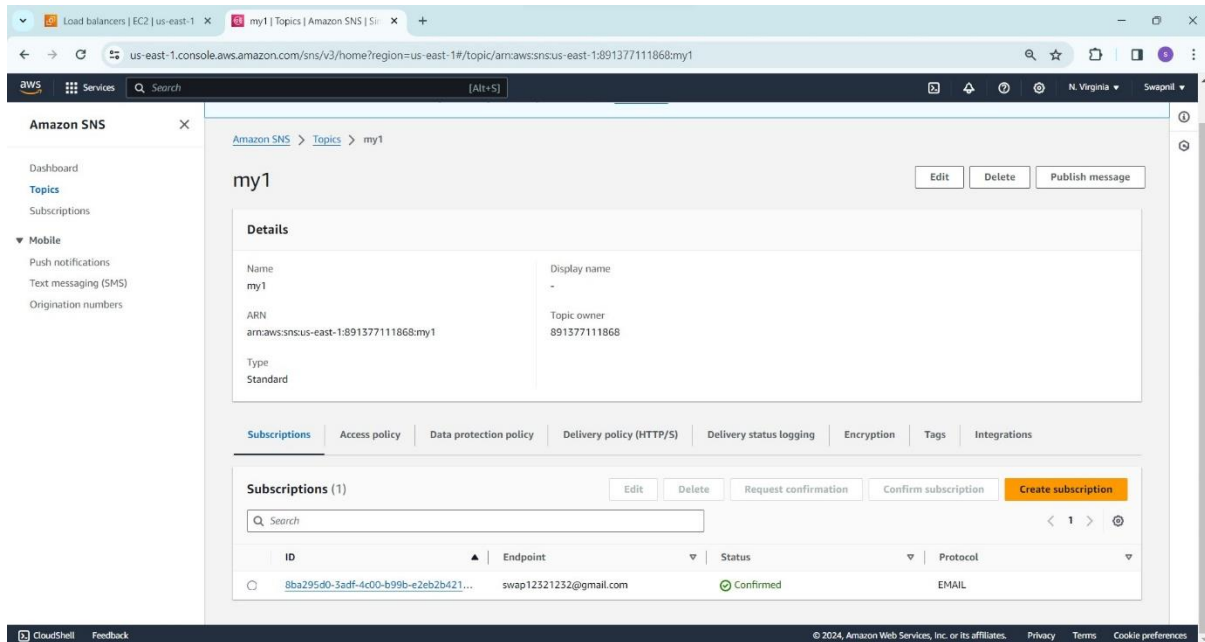




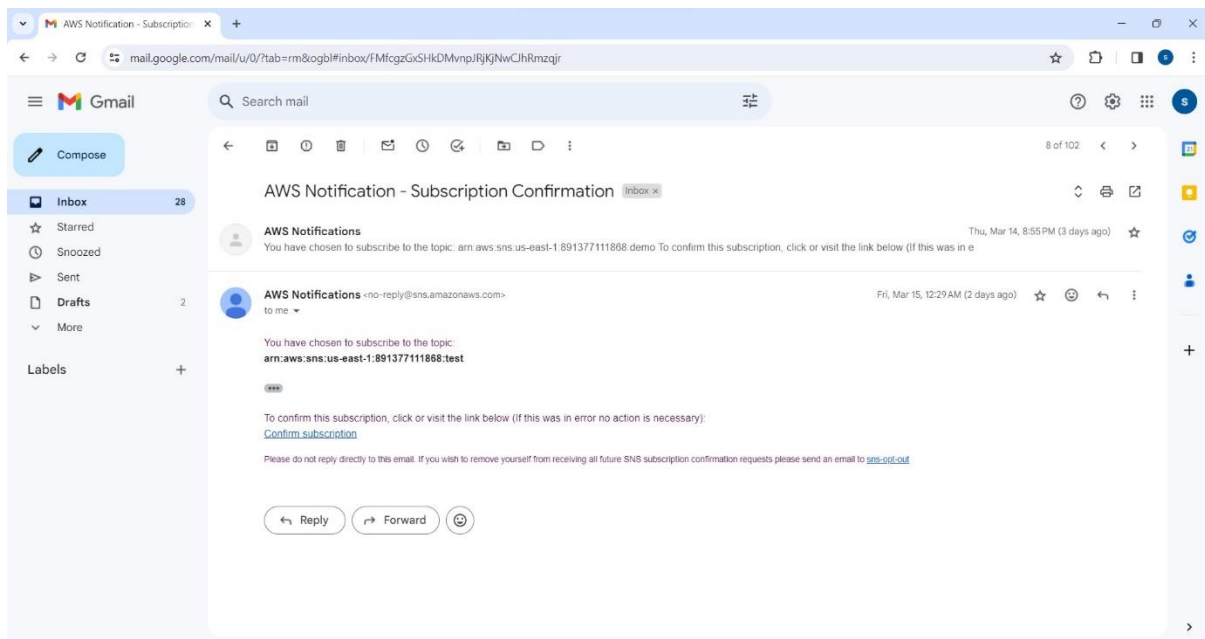
# Check Simple Notification Service (SNS)

## Step 13: Check SNS service.

➤ New topic and subscription is created.

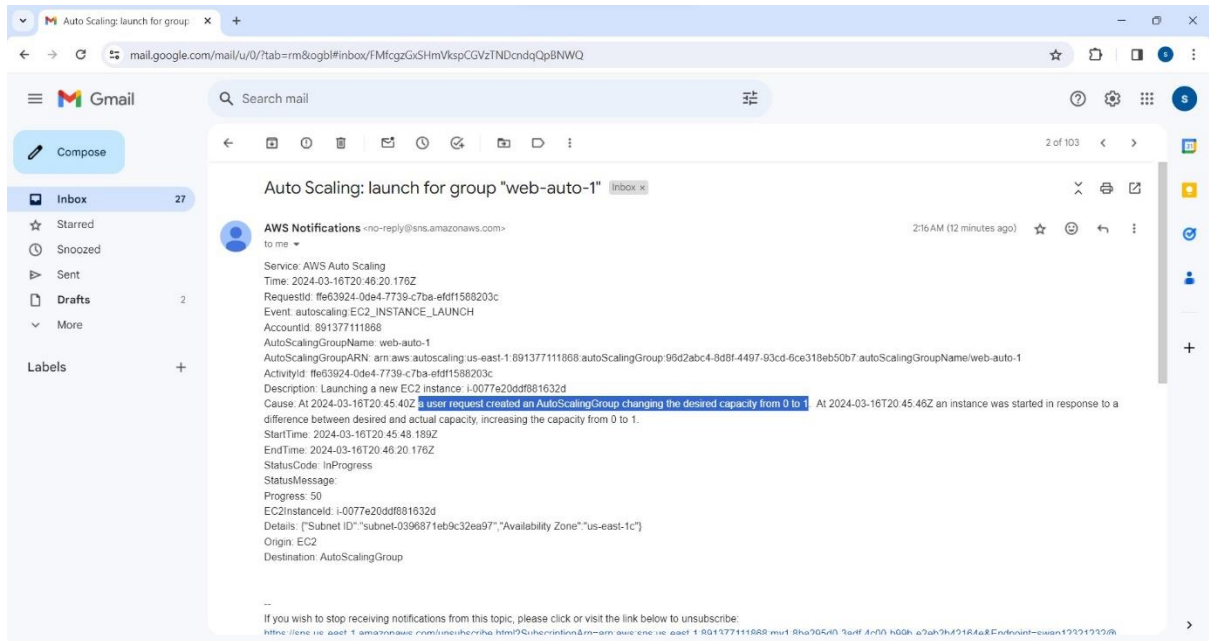


## Step 14: Confirm subscription email

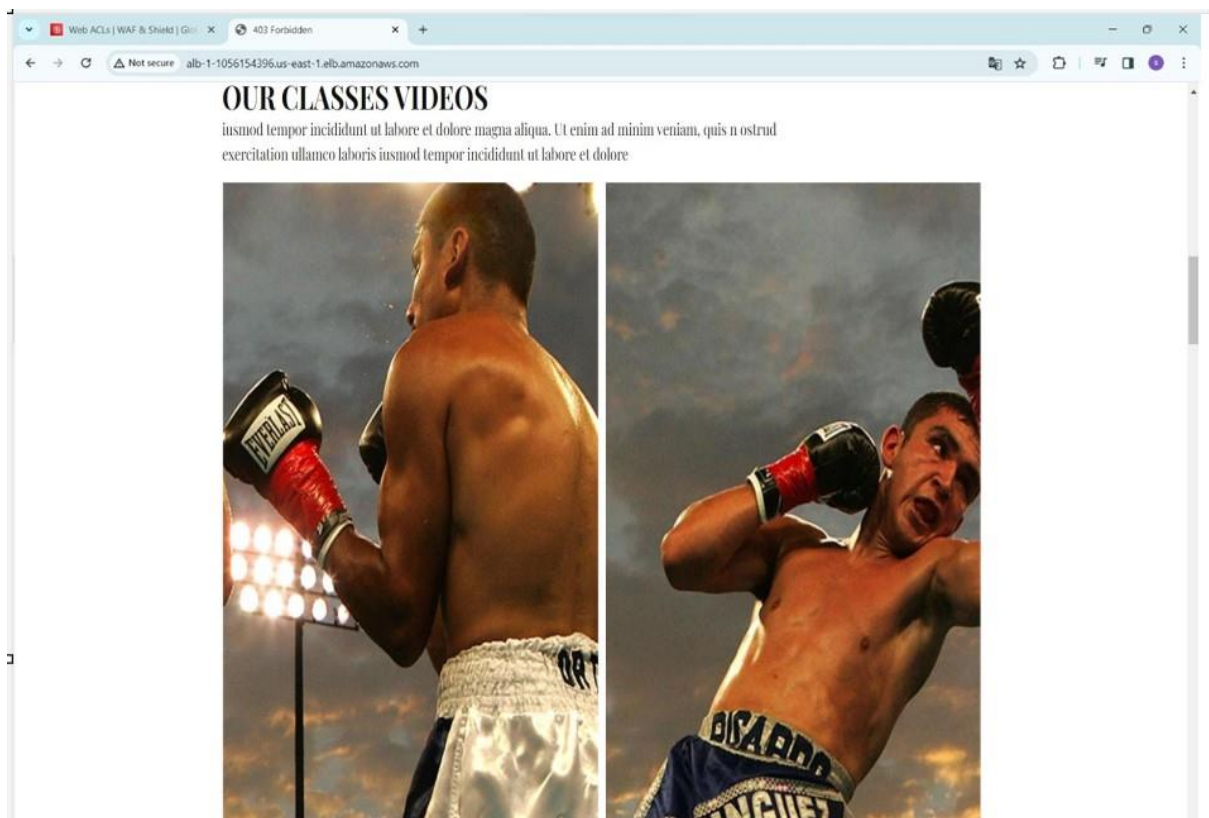




**Step 15:** We get the notification mail of launch an instance.



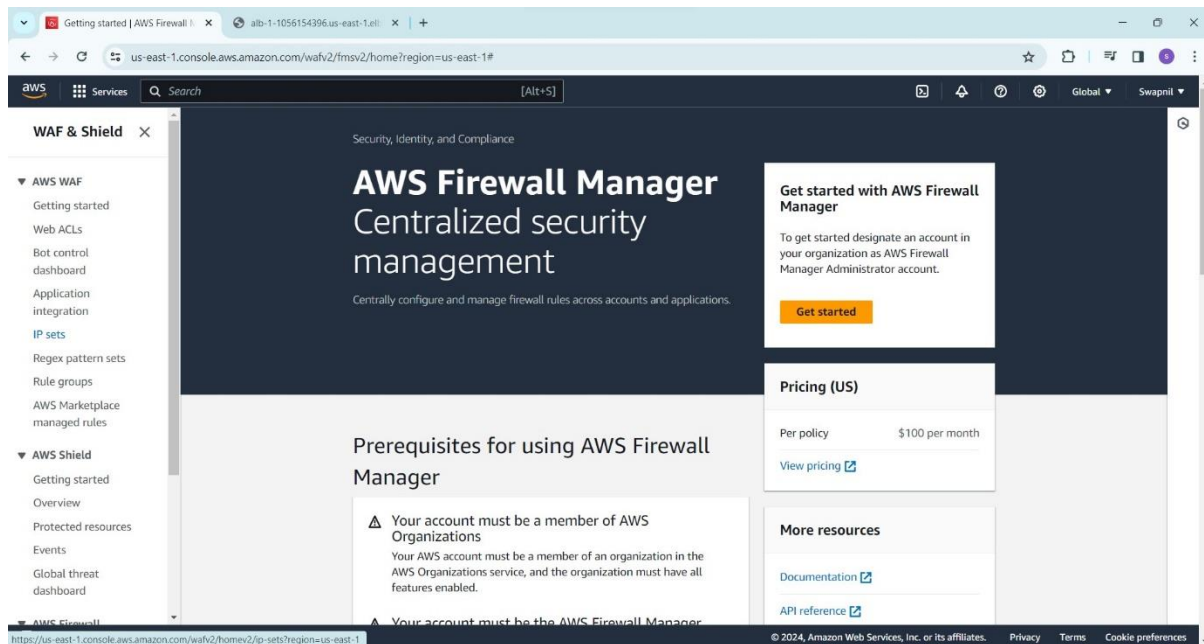
**Step 16:** Access the webpage though the load balancer DNS.



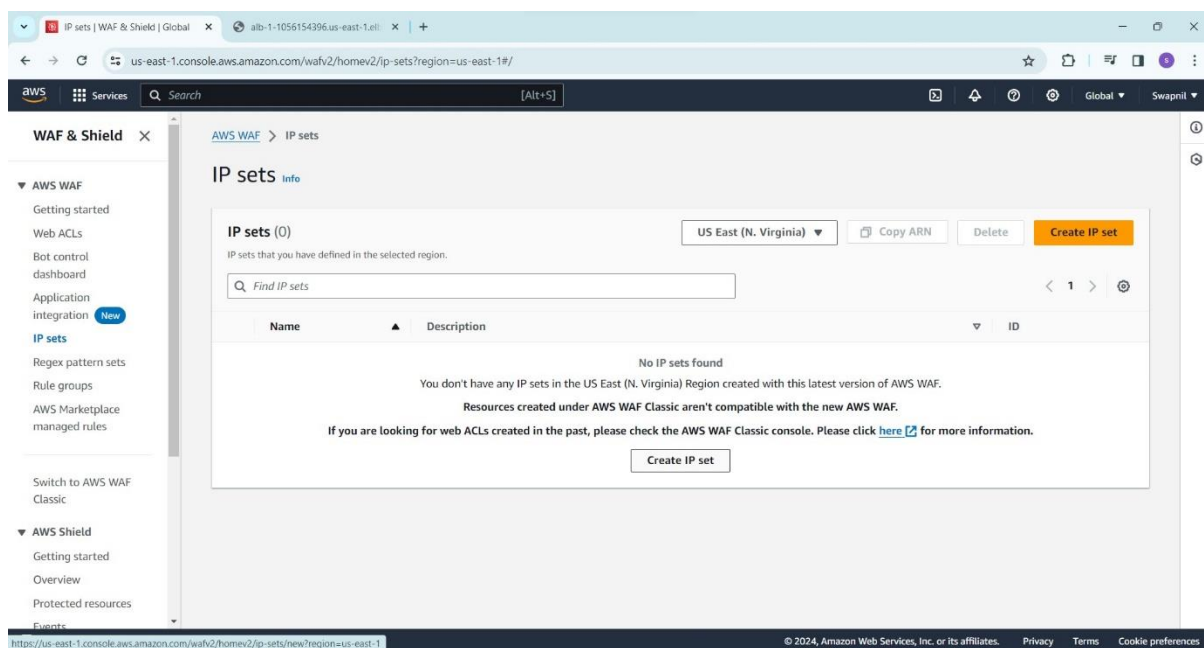
# Add web application firewall

## Step 17: Add Web Application Firewall (WAF) to load balancer.

➤ Go to the navigation panel and select WAF.



## Step 18: Create IP sets.



## Step 19: IP set details.

- Enter IP set rule name
- Choose region and IP version
- Add the IP address list

The screenshot shows the 'IP set details' page in the AWS console. The page has a left sidebar with a menu and a main content area. The main content area contains the following fields:

- IP set name:** A text input field with the value 'ip\_rule'. Below it, a note states: 'The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, -, (hyphen), and \_ (underscore).' There is also a 'Description - optional' text input field with a note: 'The description can have 1-256 characters.'
- Region:** A dropdown menu showing 'US East (N. Virginia)'.
- IP version:** Two radio buttons: 'IPv4' (selected) and 'IPv6'.
- IP addresses:** A text area containing the IP address '103.162.158.172/31'. Below the text area, a note states: 'Enter one IP address per line in CIDR format.'

The bottom of the page shows the AWS footer with 'CloudShell', 'Feedback', and copyright information.

## Step 20: Create web ACL (access control list)

The screenshot shows the 'Web ACLs' page in the AWS console. The page has a left sidebar with a menu and a main content area. The main content area contains the following elements:

- Web ACLs (0):** A heading indicating that no web ACLs are currently defined in the selected region.
- Region:** A dropdown menu showing 'US East (N. Virginia)'.
- Buttons:** 'Copy ARN', 'Delete', and 'Create web ACL' (highlighted in orange).
- Search:** A search bar with the placeholder text 'Find web ACLs'.
- Table:** A table with columns 'Name', 'Description', and 'ID'. The table is currently empty.
- Message:** A message stating: 'No web ACLs found. You don't have any web ACLs in the US East (N. Virginia) Region created with this latest version of AWS WAF. Resources created under AWS WAF Classic aren't compatible with the new AWS WAF. If you are looking for web ACLs created in the past, please check the AWS WAF Classic console. Please click [here](#) for more information.' Below the message is a 'Create web ACL' button.

The bottom of the page shows the AWS footer with 'CloudShell', 'Feedback', and copyright information.

# Step 21: Describe web ACL and associate it to AWS resources

**Web ACL details**

**Resource type**  
Choose the type of resource to associate with this web ACL. Changing this setting will reset the page.

☐ Amazon CloudFront distributions

☒ Regional resources (Application Load Balancers, Amazon API Gateway REST APIs, Amazon App Runner services, AWS AppSync GraphQL APIs, Amazon Cognito user pools and AWS Verified Access Instances)

**Region**  
Choose the AWS Region to create this web ACL in. Changing this setting will reset the page.

US East (N. Virginia)

**Name**  
ACL-1  
The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, -, (hyphen), and \_ (underscore).

**Description - optional**  
  
The description can have 1-256 characters.

**CloudWatch metric name**  
ACL-1  
The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, -, (hyphen), and \_ (underscore).

**Associated AWS resources - optional (1)** Remove Add AWS resources

## ➤ Add AWS resources

**Add AWS resources**

**Resource type**  
Select the resource type and then select the resource you want to associate with this web ACL.

☒ Application Load Balancer ☐ Amazon API Gateway REST API ☐ Amazon App Runner service

☐ AWS AppSync GraphQL API ☐ Amazon Cognito user pool ☐ AWS Verified Access

Select the resources you want to associate with the web ACL.

Find AWS resources to associate

<input checked="" type="checkbox"/>	Name
<input checked="" type="checkbox"/>	ALB-1

Cancel Add

## Step 22: Add rule and rule groups

The screenshot shows the AWS WAF console interface for creating a new web ACL. The page title is 'Add my own rules and rule groups'. On the left, a sidebar lists the steps: Step 1 (Describe web ACL), Step 2 (Add my own rules and rule groups), Step 3 (Set rule priority), Step 4 (Configure metrics), and Step 5 (Review and create web ACL). The main content area is divided into three sections: 'Rule type', 'Rule', and 'IP set'. In the 'Rule type' section, 'IP set' is selected. In the 'Rule' section, the name 'rule-1' is entered. In the 'IP set' section, 'ip\_rule' is selected from a dropdown menu. A footer bar contains 'CloudShell', 'Feedback', and copyright information for Amazon Web Services, Inc. or its affiliates.

**Rule type**

Rule type

- ☒ IP set  
Use IP sets to identify a specific list of IP addresses.
- ☐ Rule builder  
Use a custom rule to inspect for patterns including query strings, headers, countries, and rate limit violations.
- ☐ Rule group  
Use a rule group to combine rules into a single logical set.

**Rule**

Name

rule-1

The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and \_ (underscore).

**IP set**

IP set

ip\_rule

IP address to use as the originating address

When a request comes through a CDN or other proxy network, the source IP address identifies the proxy and the original IP address is sent.

## Step 23: Set rule priority.

The screenshot shows the AWS WAF console interface for setting the priority of rules. The page title is 'Set rule priority'. On the left, the same sidebar as in Step 22 is visible, with 'Set rule priority' highlighted. The main content area shows a table with one rule, 'rule-1', with a capacity of 1 and an action of 'Block'. Above the table, there are 'Move up' and 'Move down' buttons. Below the table, there are 'Cancel', 'Previous', and 'Next' buttons. A footer bar contains 'CloudShell', 'Feedback', and copyright information for Amazon Web Services, Inc. or its affiliates.

**Set rule priority**

Rules (1/1)

If a request matches a rule, take the corresponding action. The rules are prioritized in order they appear.

Name	Capacity	Action
rule-1	1	Block

Cancel Previous Next

## Step 24: Configure metrics

The screenshot shows the 'Configure metrics' step in the AWS WAF console. The left sidebar lists five steps: Step 1 (Describe web ACL and associate it to AWS resources), Step 2 (Add rules and rule groups), Step 3 (Set rule priority), Step 4 (Configure metrics), and Step 5 (Review and create web ACL). The main content area is titled 'Configure metrics' and includes an 'Info' icon. It contains two sections: 'Amazon CloudWatch metrics' and 'Request sampling options'. In the 'Amazon CloudWatch metrics' section, there is a table with two columns: 'Rules' and 'CloudWatch metric name'. The 'Rules' column has a checkbox next to 'rule-1'. The 'CloudWatch metric name' column has a text input field containing 'rule-1'. The 'Request sampling options' section has a heading and a sub-heading, followed by three radio button options: 'Enable sampled requests' (selected), 'Disable sampled requests', and 'Enable sampled requests with exclusions'. At the bottom right of the main content area are three buttons: 'Cancel', 'Previous', and 'Next'.

Step 1  
[Describe web ACL and associate it to AWS resources](#)

Step 2  
[Add rules and rule groups](#)

Step 3  
[Set rule priority](#)

Step 4  
**Configure metrics**

Step 5  
[Review and create web ACL](#)

### Configure metrics Info

**Amazon CloudWatch metrics**  
CloudWatch metrics allow you to monitor web requests, web ACLs, and rules.

Rules	CloudWatch metric name
<input checked="" type="checkbox"/> rule-1	<input type="text" value="rule-1"/>

**Request sampling options**  
If you disable request sampling, you can't view requests that match your web ACL rules.

Options

- ☒ Enable sampled requests
- ☐ Disable sampled requests
- ☐ Enable sampled requests with exclusions

Cancel Previous Next

## Step 25: Review and create web ACL

The screenshot shows the 'Review and create web ACL' step in the AWS WAF console. The left sidebar lists five steps: Step 1 (Describe web ACL and associate it to AWS resources), Step 2 (Add rules and rule groups), Step 3 (Set rule priority), Step 4 (Configure metrics), and Step 5 (Review and create web ACL). The main content area is titled 'Token domain list (0)' and shows a table with one column: 'Name'. Below the table is a message: 'No items. No items to display'. Below this is a section titled 'Step 4: Configure metrics' with an 'Edit step 4' button. This section contains two tables: 'Amazon CloudWatch metrics (1)' and 'Sampled requests'. The 'Amazon CloudWatch metrics (1)' table has two columns: 'Rules' and 'CloudWatch metric name'. The 'Rules' column has a checkbox next to 'rule-1'. The 'CloudWatch metric name' column has a text input field containing 'rule-1'. The 'Sampled requests' table has two columns: 'Sampled requests' and 'Sampled requests for web ACL default actions'. The 'Sampled requests' column has a radio button next to 'Enabled'. The 'Sampled requests for web ACL default actions' column has a radio button next to 'Enabled'. At the bottom right of the main content area are three buttons: 'Cancel', 'Previous', and 'Create web ACL'.

Token domain list (0)

Name
No items No items to display

**Step 4: Configure metrics** Edit step 4

**Amazon CloudWatch metrics (1)**

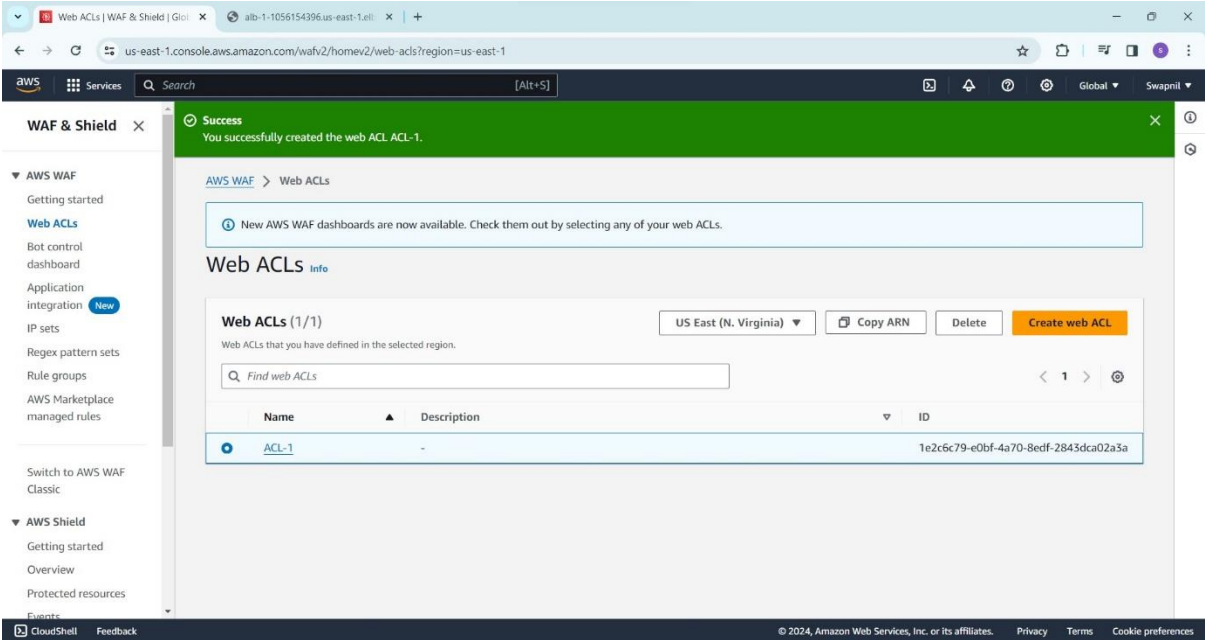
Rules	CloudWatch metric name
<input checked="" type="checkbox"/> rule-1	<input type="text" value="rule-1"/>

**Sampled requests**

Sampled requests	Sampled requests for web ACL default actions
<input checked="" type="radio"/> Enabled	<input checked="" type="radio"/> Enabled

Cancel Previous Create web ACL

# Step 26: Web ACL is created



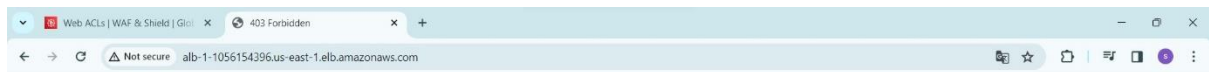


## **Step 27: Check the result**

**Try to access a load balancer from the IP which is define in the IP sets rules group**

**We get 403 forbidden message because WAF block that IP.**

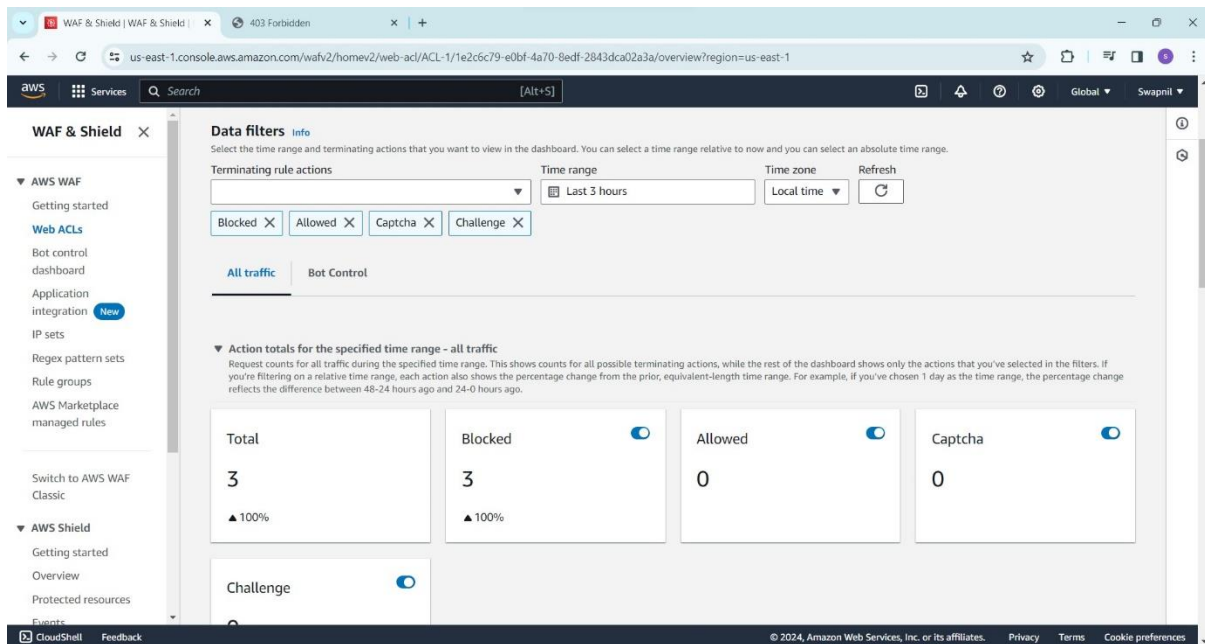
**403 Forbidden error, it means that you do not have permission to view the requested file or resource.**



## Step 28:

From WEB ACL we filter the traffic and check all details

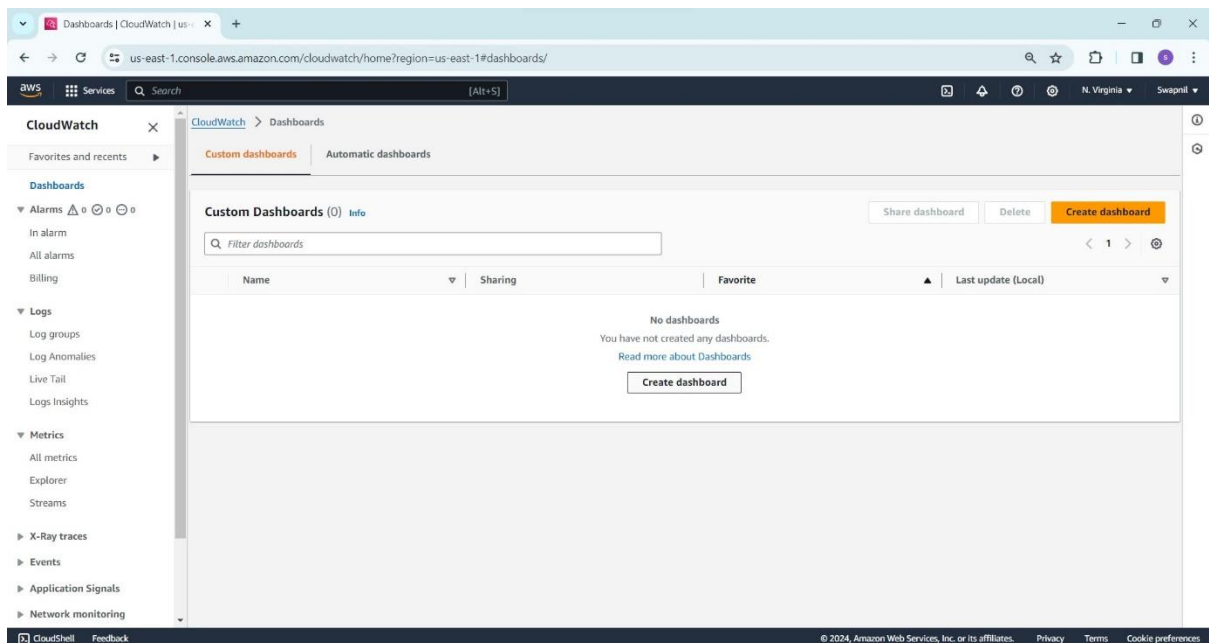
Like blocked, allowed IP, Sample of bot detection, client device types, attack type, top 10 countries, etc.



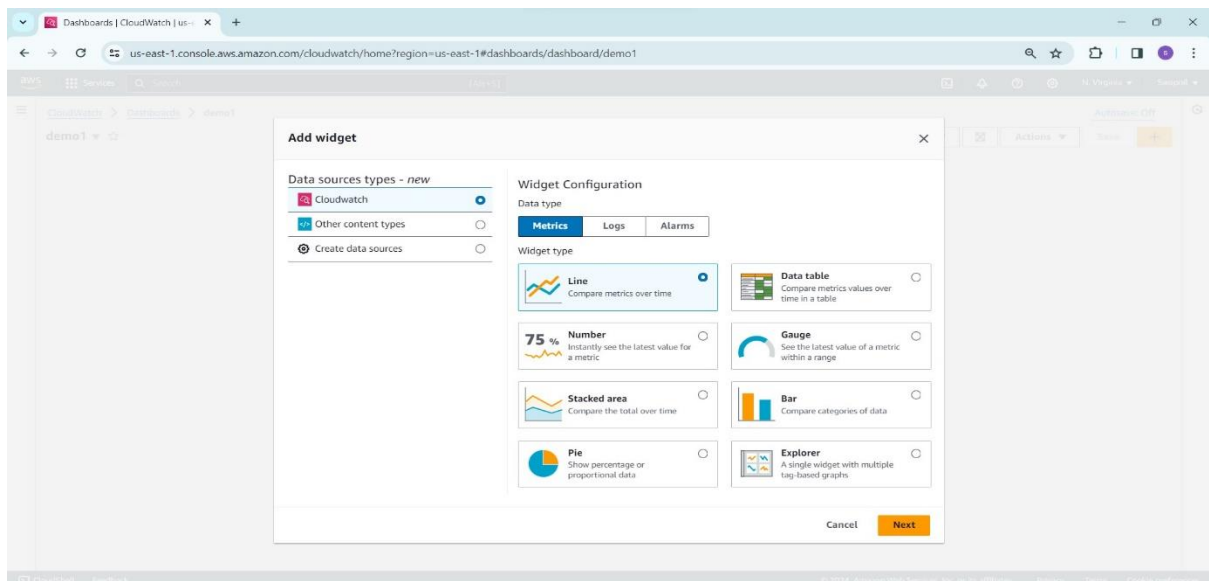
# Add cloud watch: monitor the system services from the cloud watch.

## Step 29: Create a dashboard.

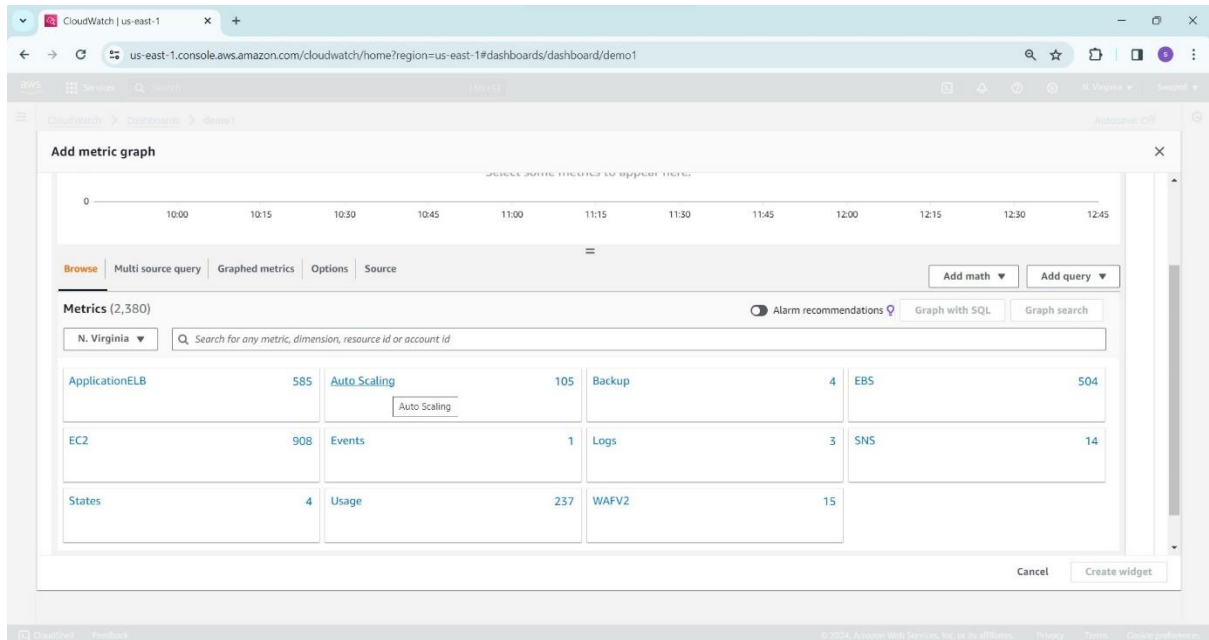
➤ Define the name to dashboard.



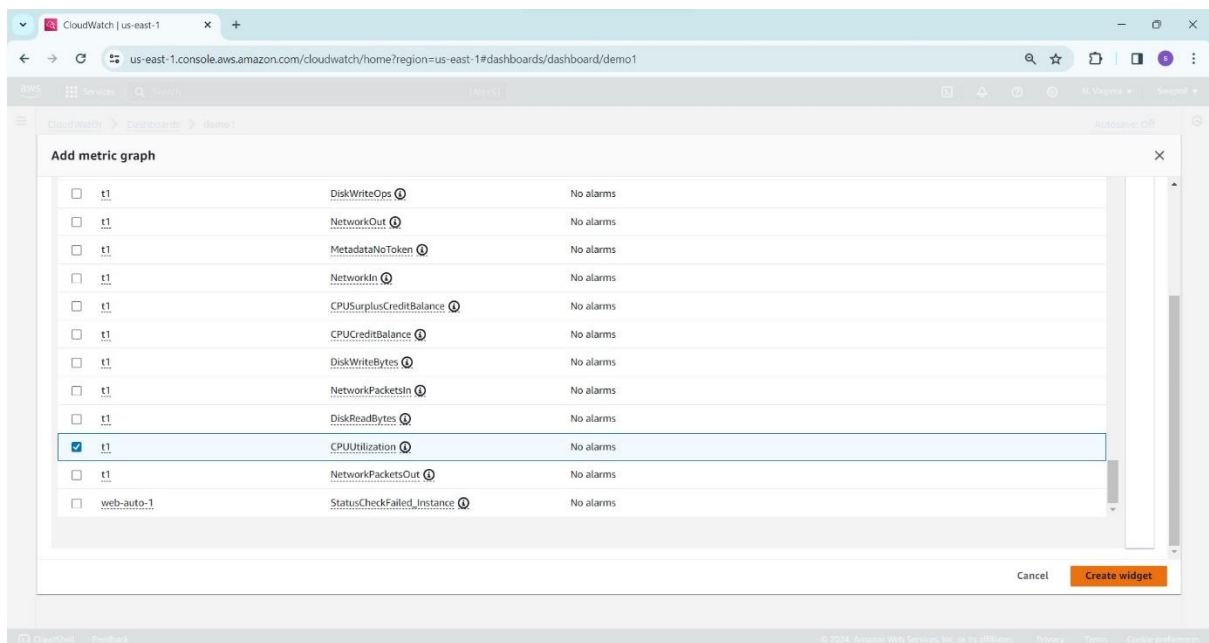
## Step 30: Add the widget.



## Step 31: Select metrics graph.



## Step.32: Add the selected metrics graph to widget.



## Step 33: Monitor the system performance.

