

## AWS WAF Project

In this project I will be configuring an AWS WAF with an Application Load Balancer to access an EC2 instance. I will demonstrate how to allow and deny traffic into the EC2 instance. I will start this project first by creating a VPC (Virtual Private Network), an internet gateway, and a subnet for the EC2 instance.

Note: Select the closest region to you for better performance.

### **Create a VPC:**

1. Search for VPC on the AWS console and select it to open its dashboard.
2. Then click “create VPC”.
3. From VPC settings select VPC only, then name the VPC, insert the IPv4 CIDR, then leave all the other settings default and then click create VPC.

The screenshot shows the 'Create VPC' wizard on the 'Info' tab. The 'VPC settings' section is active. Under 'Resources to create', the 'VPC only' option is selected. A 'Name tag - optional' field contains 'dhruval-aws-waf'. Under 'IPv4 CIDR block', 'IPv4 CIDR manual input' is selected, and the CIDR block '12.0.0.0/16' is entered. Under 'IPv6 CIDR block', 'No IPv6 CIDR block' is selected. Under 'Tenancy', 'Default' is chosen. The overall interface is clean with a white background and standard AWS branding.

4. Then double check if the new VPC has been created.

Your VPCs (1/2) <a href="#">Info</a>							Last updated <a href="#">C</a> less than a minute ago	<a href="#">Actions</a>	<a href="#">Create VPC</a>
							<a href="#">C</a>	<a href="#">Actions</a>	<a href="#">Create VPC</a>
	Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR			DHCP option se
<input type="checkbox"/>	-	vpc-0a0f6be450f4e9deb	<span>Available</span>	<input type="radio"/> Off	172.31.0.0/16	-			dopt-0bfca72a2
<input checked="" type="checkbox"/>	dhruval-aws-waf	vpc-04ad72634e16955db	<span>Available</span>	<input type="radio"/> Off	12.0.0.0/16	-			dopt-0bfca72a2

## **Create Internet Gateway:**

Note: Internet Gateway is needed to have internet access to our VPC.

1. On the left side of the navigation bar in VPC, select Internet Gateway.
2. Then select create internet gateway on top right.
3. Name the internet gateway, leave everything else default and then select create internet gateway.

The screenshot shows the 'Create internet gateway' page in the AWS VPC console. In the 'Internet gateway settings' section, a 'Name tag' is specified as 'dhrupal-aws-waf-ig'. Under the 'Tags - optional' section, a single tag 'Name: dhrupal-aws-waf-ig' is listed. At the bottom right, there are 'Cancel' and 'Create internet gateway' buttons.

4. Once created, then attach it to the VPC that was created earlier.

The screenshot shows the details page for the newly created internet gateway 'igw-0fb6f1aed12462b9a'. It displays the gateway ID, state (Detached), VPC ID (empty), and owner (471744311739). The 'Actions' menu includes options like 'Attach to VPC', 'Detach from VPC', 'Manage tags', and 'Delete'. The 'Tags' section shows one tag: 'Name: dhrupal-aws-waf-ig'.

5. Select the VPC and then attach the internet gateway.

The screenshot shows the 'Attach to VPC' dialog for the internet gateway 'igw-0fb6f1aed12462b9a'. It lists available VPCs, with 'vpc-04ad72634e16955db' selected. The 'Attach internet gateway' button is at the bottom right.

## **Create a subnet:**

1. On the left side of the navigation bar in VPC, select Subnets and create subnet.
2. Select the created VPC.

### **Create subnet** Info

**VPC**

**VPC ID**  
Create subnets in this VPC.

**Associated VPC CIDRs**

**IPv4 CIDRs**

3. Create the first subnet with a unique name, availability zone, and IPv4 VPC / subnet CIDRs.

#### **Subnet settings**

Specify the CIDR blocks and Availability Zone for the subnet.

##### **Subnet 1 of 1**

###### **Subnet name**

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

###### **Availability Zone** Info

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

###### **IPv4 VPC CIDR block** Info

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

###### **IPv4 subnet CIDR block**

256 IPs

< > ^ ^

Note: need 2 subnets for different availability zones.

- Create the second subnet with a unique name, different availability zone, and IPv4 VPC / subnet CIDRs. Then click create subnet.

**Subnet 2 of 2**

**Subnet name**  
Create a tag with a key of 'Name' and a value that you specify.  
  
The name can be up to 256 characters long.

**Availability Zone** [Info](#)  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

**IPv4 VPC CIDR block** [Info](#)  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

**IPv4 subnet CIDR block**  
 256 IPs  
< > ^ v

**Tags - optional**

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="dhrupal-aws-waf-subnet1c"/> <a href="#">Remove</a>

[Add new tag](#)  
You can add 49 more tags.

Note: two new subnets have been created.

**Subnets (2)** [Info](#)

Last updated [less than a minute ago](#) [Actions](#) [Create subnet](#)

<input type="checkbox"/> Name	<input type="checkbox"/> Subnet ID	<input type="checkbox"/> State	<input type="checkbox"/> VPC	<input type="checkbox"/> Block Public...	<input type="checkbox"/> IPv4 CIDR	<input type="checkbox"/> IPv6 CIDR
<input type="checkbox"/> dhrupal-aws-waf-subnet1c	<a href="#">subnet-0cf48cad58b9f21c5</a>	<input checked="" type="radio"/> Available	<a href="#">vpc-04ad72634e16955db   dhr...</a>	<input type="radio"/> Off	12.0.2.0/24	-
<input type="checkbox"/> dhrupal-aws-waf-subnet1b	<a href="#">subnet-067a70c26b82ba47c</a>	<input checked="" type="radio"/> Available	<a href="#">vpc-04ad72634e16955db   dhr...</a>	<input type="radio"/> Off	12.0.1.0/24	-

## Create a Route table:

Note: Route table is needed so the internet gateway can be connected to the route table and the subnet can access the internet.

- On the left side of the navigation bar in VPC, select Route table and then create route table.
- Name the route table, select the created VPC, and create route table.

**Create route table** [Info](#)  
A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

**Route table settings**

**Name - optional**  
Create a tag with a key of 'Name' and a value that you specify.

**VPC**  
The VPC to use for this route table.

**Tags**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="dhrupal-public-rt"/> <a href="#">Remove</a>

[Add new tag](#)  
You can add 49 more tags.

[Cancel](#) [Create route table](#)

3. Then associate the subnets to the route table. On the created route table select the “Subnet associations” tab and then select edit subnet associations.

**rtb-0225fb15dfb32ced7 / dhruval-public-rt**

**Actions ▾**

**Details Info**

Route table ID: rtb-0225fb15dfb32ced7

Main: No

VPC: vpc-04ad72634e16955db | dhruval-aws-waf

Owner ID: 471744311739

Explicit subnet associations: –

Edge associations: –

**Routes | Subnet associations | Edge associations | Route propagation | Tags**

**Explicit subnet associations (0)**

**Edit subnet associations**

No subnet associations.  
You do not have any subnet associations.

4. Select the created subnets and then save associations.

**Edit subnet associations**

Change which subnets are associated with this route table.

**Available subnets (2/2)**

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
dhruval-aws-waf-subnet1c	subnet-0cf48cad58b9f21c5	12.0.2.0/24	–	Main (rtb-0599e93a813d943ec)
dhruval-aws-waf-subnet1b	subnet-067a70c26b82ba47c	12.0.1.0/24	–	Main (rtb-0599e93a813d943ec)

**Selected subnets**

subnet-0cf48cad58b9f21c5 / dhruval-aws-waf-subnet1c X    subnet-067a70c26b82ba47c / dhruval-aws-waf-subnet1b X

**Cancel** **Save associations**

Note: the route table still does not have internet access so it's time to create routes.

5. Click on the routes tab in the created route table and edit routes.

**rtb-0225fb15dfb32ced7 / dhruval-public-rt**

**Actions ▾**

**Details Info**

Route table ID: rtb-0225fb15dfb32ced7

Main: No

VPC: vpc-04ad72634e16955db | dhruval-aws-waf

Owner ID: 471744311739

Explicit subnet associations: 2 subnets

Edge associations: –

**Routes | Subnet associations | Edge associations | Route propagation | Tags**

**Routes (1)**

Destination	Target	Status	Propagated
12.0.0.0/16	local	Active	No

**Both ▾** **Edit routes**

6. Add route, select 0.0.0.0/0 (anyone within and outside of the subnet can access the internet).
7. For target select Internet Gateway and then select the created internet gateway.

**Edit routes**

Destination	Target	Status	Propagated
12.0.0.0/16	local	Active	No
Q 0.0.0.0/0	Internet Gateway	-	No
	igw-0fb6f1aed12462b9a		

[Add route](#) [Remove](#)

[Cancel](#) [Preview](#) [Save changes](#)

## **Create an EC2 instance:**

Note: this will be the EC2 instance where a user either has access to it or not.

1. Search for EC2 and launch an instance.
2. Name the instance and select appropriate OS.

### **Launch an instance** Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags** Info

Name  [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

[Recent](#) [Quick Start](#)

Amazon Linux	Ubuntu	Windows	Red Hat	SUSE Linux	Debian
	 ubuntu®				

[Browse more AMIs](#)  
Including AMIs from AWS, Marketplace and the Community

**Amazon Machine Image (AMI)**

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type ami-04f7a54071e74f488 (64-bit (x86)) / ami-01fe939b39edeeaa4 (64-bit (Arm)) Virtualization: hvm ENA enabled: true Root device type: ebs	<a href="#">Free tier eligible</a>
--	------------------------------------

### 3. Create a key pair for SSH.

**Create key pair** X

**Key pair name**  
Key pairs allow you to connect to your instance securely.  
 The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

**Key pair type**  
 RSA RSA encrypted private and public key pair  
 ED25519 ED25519 encrypted private and public key pair

**Private key file format**  
 .pem For use with OpenSSH  
 .ppk For use with PuTTY

**⚠️** When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#)

[Cancel](#) [Create key pair](#)

4. Edit the network settings to:
  - a. Select the created VPC
  - b. Select any subnet
  - c. Enable auto-assign public IP

**▼ Network settings** [Info](#)

**VPC - required** [Info](#)

vpc-04ad72634e16955db (dhrupal-aws-waf)  
12.0.0.0/16

**Subnet** [Info](#)

subnet-0cf48cad58b9f21c5 dhruval-aws-waf-subnet1c  
VPC: vpc-04ad72634e16955db Owner: 471744311739 Availability Zone: us-west-1c  
Zone type: Availability Zone IP addresses available: 251 CIDR: 12.0.2.0/24)

**Create new subnet** [Create new subnet](#)

**Auto-assign public IP** [Info](#)

Enable

Additional charges apply when outside of free tier allowance

5. Add a new security group along with SSH.
  - a. For type select HTTP
  - b. Source type must be anywhere so anyone can access it

**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**  
launch-wizard-4

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**  
launch-wizard-4 created 2025-04-23T00:52:02.924Z

**Inbound Security Group Rules**

- ▼ Security group rule 1 (TCP, 22, 0.0.0.0/0) Remove

<b>Type   Info</b> ssh	<b>Protocol   Info</b> TCP	<b>Port range   Info</b> 22
<b>Source type   Info</b> Anywhere	<b>Source   Info</b> <input type="text"/> Add CIDR, prefix list or security group 0.0.0.0/0	<b>Description - optional   Info</b> e.g. SSH for admin desktop
- ▼ Security group rule 2 (TCP, 80, 0.0.0.0/0) Remove

<b>Type   Info</b> HTTP	<b>Protocol   Info</b> TCP	<b>Port range   Info</b> 80
<b>Source type   Info</b> Anywhere	<b>Source   Info</b> <input type="text"/> Add CIDR, prefix list or security group	<b>Description - optional   Info</b> e.g. SSH for admin desktop

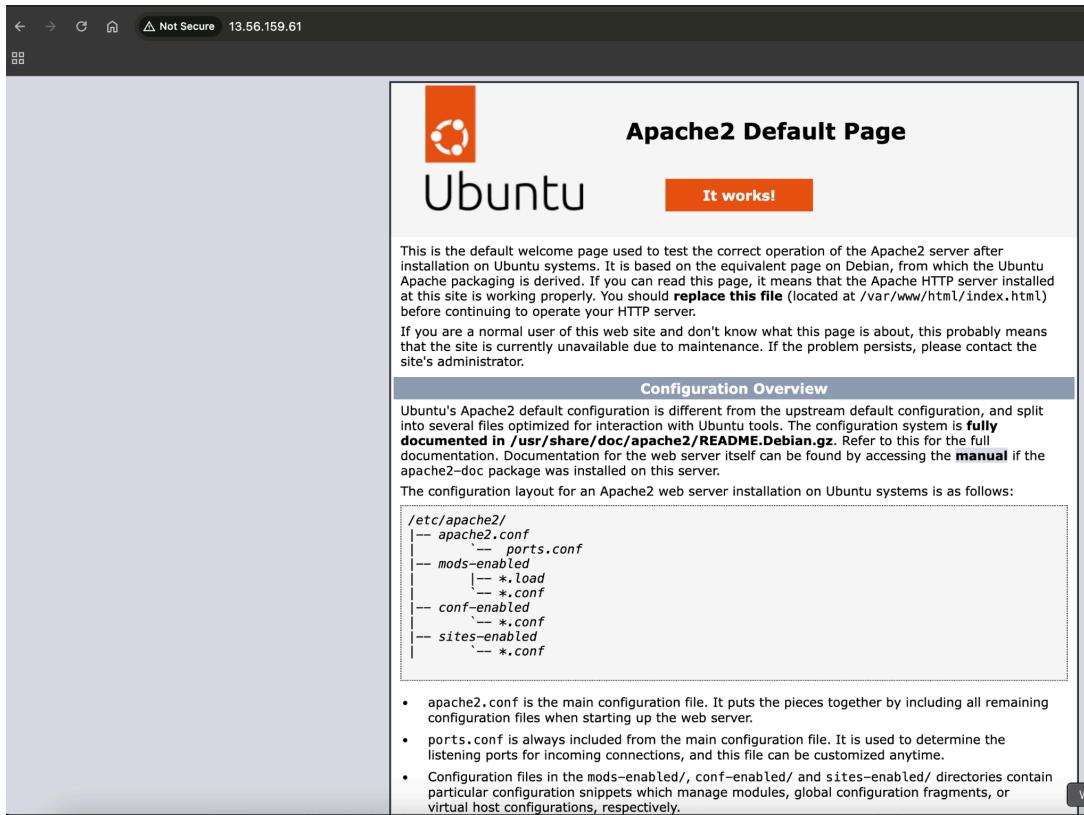
6. Click advanced details, head down to user data and add a script for the apache server for the EC2 instance. Then launch instance.

**User data - optional | Info**  
Upload a file with your user data or enter it in the field.

---

```
#!/bin/bash
yes | sudo apt update
yes | sudo apt install apache2
echo "<h1>Server Details</h1><p><strong>Hostname:</strong> $(hostname)
</p><p><strong>IP Address:</strong> $(hostname -l | cut -d' ' -f1)</p>" >
/var/www/html/index.html
sudo systemctl restart apache2
```

Note: Once the instance is created then copy its public IP, open in a new tab and it should load the apache.



## Create Application Load Balancer:

Note: a target group is needed first in order to create a load balancer.

1. On the left side of the navigation bar in EC2, select Target Groups and then create target group.
2. Choose instances as target type, name the target group, leave everything else default and then click next.

### Specify group details

Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

#### Basic configuration

Settings in this section can't be changed after the target group is created.

#### Choose a target type

Instances

- Supports load balancing to instances within a specific VPC.
- Facilitates the use of [Amazon EC2 Auto Scaling](#) to manage and scale your EC2 capacity.

**Target group name**

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

**Protocol : Port**

Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and created. This choice cannot be changed after creation

HTTP	▼
80	1-65535

**IP address type**

Only targets with the indicated IP address type can be registered to this target group.

- IPv4**  
Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.
- IPv6**  
Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). [Learn more](#) 

**VPC**

Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

dhrupal-aws-waf	▼
vpn-04ad72634e16955db	
IPv4 VPC CIDR: 12.0.0.0/16	

6. Select the EC2 instance, include as pending below, it will be part of the target group and then create target group.

Instance ID	Name	State	Security groups	Zone	Private IPv4 address
i-0709bbe4e82ba176f	dhrupal-aws-waf	Running	launch-wizard-4	us-west-1c	12.0.2.94

**0 selected**

**Ports for the selected instances**  
Ports for routing traffic to the selected instances.

80
1-65535 (separate multiple ports with commas)

[Include as pending below](#)

1 selection is now pending below. Include more or register targets when ready.

**Review targets**

**Targets (1)**

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID	Launch time
i-0709bbe4e82ba176f	dhrupal-aws-waf	80	Running	launch-wizard-4	us-west-1c	12.0.2.94	subnet-0cf48cad58b9f21c5	April 22, 2025, 18:00

1 pending

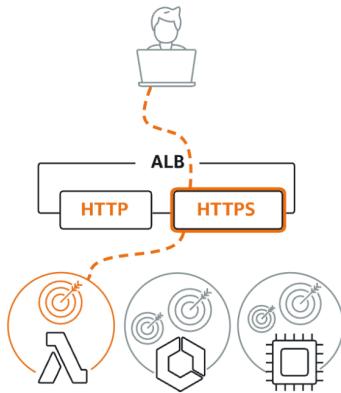
[Cancel](#) [Previous](#) [Create target group](#)

7. Now to create a load balancer again from the left side of the navigation bar in EC2, select Load balancers and create load balancer.

8. Choose the Application load balancer type out of the three and create.

### Load balancer types

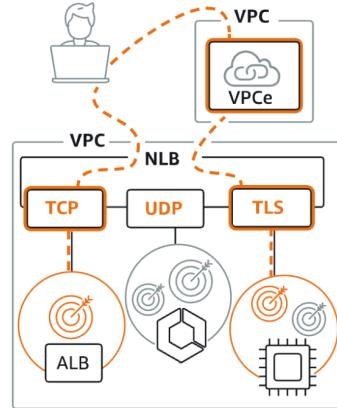
#### Application Load Balancer [Info](#)



Choose an Application Load Balancer when you need a flexible feature set for your 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.

[Create](#)

#### Network Load Balancer [Info](#)



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 applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.

[Create](#)

#### Gateway Load Balancer [Info](#)



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.

[Create](#)

## 9. Name the load balancer.

### Basic configuration

#### Load balancer name

Name must be unique within your AWS account and can't be changed after the load balancer is created.

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

## 10. Select the created VPC, the two availability zones, and subnets.

### Network mapping Info

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

#### VPC Info

The load balancer will exist and scale within the selected VPC. The selected VPC is also where the load balancer targets must be hosted unless routing to Lambda or on-premises targets, or if using VPC peering. To confirm the VPC for your targets, view [target groups](#). For a new VPC, create a VPC.

dhrupal-aws-waf  
vpc-04ad72634e16955db  
IPv4 VPC CIDR: 12.0.0.0/16



#### IP pools - new Info

You can optionally choose to configure an IPAM pool as the preferred source for your load balancers IP addresses. Create or view Pools in [Amazon VPC IP Address Manager console](#).

##### Use IPAM pool for public IPv4 addresses

The IPAM pool you choose will be the preferred source of public IPv4 addresses. If the pool is depleted IPv4 addresses will be assigned by AWS.

#### Availability Zones and subnets Info

Select at least two Availability Zones and a subnet for each zone. A load balancer node will be placed in each selected zone and will automatically scale in response to traffic. The load balancer routes traffic to targets in the selected Availability Zones only.

##### us-west-1b (usw1-az3)

###### Subnet

Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.

subnet-067a70c26b82ba47c  
IPv4 subnet CIDR: 12.0.1.0/24

dhrupal-aws-waf-subnet1b

##### us-west-1c (usw1-az1)

###### Subnet

Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.

subnet-0cf48cad58b9f21c5  
IPv4 subnet CIDR: 12.0.2.0/24

dhrupal-aws-waf-subnet1c

## 11. Create a new security group to allow SSH and HTTP requests on port 80 for the application load balancer.

## 12. Name the security group, add a description, and the created VPC.

### Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

#### Basic details

##### Security group name Info

dhrupal-aws-waf-sg-ssh-http

Name cannot be edited after creation.

##### Description Info

Allow SSH and HTTPS requests

##### VPC Info

vpc-04ad72634e16955db (dhrupal-aws-waf)

## 13. Create SSH and HTTP inbound rules with anywhere access (0.0.0.0/0). Then create security group.

Inbound rules <small>Info</small>					
Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>	
SSH	TCP	22	Anywhere... <small>Info</small>	Q 0.0.0.0/0	<input type="button" value="Delete"/>
HTTP	TCP	80	Anywhere... <small>Info</small>	Q 0.0.0.0/0	<input type="button" value="Delete"/>
<input type="button" value="Add rule"/>					

14. Once the security group is created, then go back to create application load balancer tab and select the new security group created.

**Security groups Info**  
A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

**Security groups**

Select up to 5 security groups

Q |

<input checked="" type="checkbox"/> dhrupal-aws-waf-sg-ssh-http sg-00b115bae20224a54 VPC: vpc-04ad72634e16955db
<input checked="" type="checkbox"/> default sg-0350d891fb8274965 VPC: vpc-04ad72634e16955db
<input type="checkbox"/> launch-wizard-4 sg-03f4e458fc59b7dd VPC: vpc-04ad72634e16955db

15. For listeners and routing select the target group created in default action. Then leave everything else and create load balancer. Check the DNS name of load balancer to see if it loads after creating it

**Listeners and routing Info**  
A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests.

▼ Listener HTTP:80

Protocol: HTTP	Port: 80	Default action: <a href="#">Info</a>
Forward to: dhrupal-aws-waf-target-group		HTTP
Target type: Instance, IPv4		▼
<a href="#">Create target</a>		Q
dhrupal-aws-waf-target-group		HTTP ✓
Target type: Instance, IPv4		▼

**Listener tags - optional**  
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily.

## Create WAF (Web Application Firewall):

1. Search for WAF and Shield and from its dashboard select create Web ACL.
2. Name the Web ACL and check for the proper region.

**Web ACL details**

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

Global resources (CloudFront Distributions and AWS Amplify Applications)

Regional resources (Application Load Balancers, Amazon API Gateway REST APIs, AWS AppSync 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 West (N. California) ▼

**Name**

dhrupal-aws-waf-project

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

3. From add resources select application load balancer and choose the created load balancer. Add it and then click next.

**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     AWS AppSync API

Amazon Cognito user pool     AWS Verified Access

**Resources (1)**

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

Find AWS resources to associate

Name ▲  
dhrupal-aws-waf-lb

**Cancel** **Add**

4. Then open the left navigation, select IP sets and create a new IP set to add an IP in order to allow or block it. Name the IP set and add the device IP address with a range.

**Create IP set** Info

An IP set is a collection of IP addresses.

**IP set details**

**IP set name**  
my-ip-addr  
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.

**Region**  
Choose the AWS region to create this IP set in.  
US West (N. California)

**IP version**  
 IPv4     IPv6

**IP addresses**  
129.210.115.104/32

- Then back to the Web ACL tab, select add rules, and add my own rules and rule groups.
- Select IP set as rule type and name it.

**Add my own rules and rule groups [Info](#)**

**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

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

- For IP set select the IP set created and for action select block. Then add rule.

**IP set**

IP set

▼

**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 in a header. Use caution with the option, IP address in header, because headers can be handled inconsistently by proxies and they can be modified to bypass inspection.

- Source IP address**
- IP address in header

**Action**  
Choose an action to take when a request originates from one of the IP addresses in this IP set.

- Allow
- Block**
- Count
- CAPTCHA [customize](#)
- Challenge

► **Custom response - optional**

[Cancel](#) **Add rule**

- Then select the rule created.

**Rules (1/1)**

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

<input checked="" type="checkbox"/> Name	Capacity	Action
<input checked="" type="checkbox"/> block-my-ip-addr	1	Block

- Then select the rule again, keep everything else default and then create Web ACL.
- Verify the associated AWS resources with the load balancer and rules in the created Web ACL.

**dhruval-aws-waf-project**

Download web ACL as JSON

Traffic overview | Rules | **Associated AWS resources** | Custom response bodies | Logging and metrics | Sampled requests | CloudWatch Log Insights

**Associated AWS resources (1)**

dhruval-aws-waf-lb Application Load Balancer US West (N. California)

**dhruval-aws-waf-project**

Download web ACL as JSON

Traffic overview | **Rules** | Associated AWS resources | Custom response bodies | Logging and metrics | Sampled requests | CloudWatch Log Insights

**Rules (1)**

block-my-ip-addr Block 0 -

### Test the WAF:

- Go back to EC2, select load balancer, copy its url and open in a new tab and it should be blocked with a 403 forbidden message.

Not Secure dhruval-aws-waf-lb-423977278.us-west-1.elb.amazonaws.com

403 Forbidden

2. Check the created Web ACL sampled requests logs to check for blocked access.

Sampled requests (2)					
Samples of requests from the past 3 hours.					
Metric name	Source IP	URI	Rule inside rule group	Action	Time
block-my-ip-addr	129.210.115.104 (US)	/favicon.ico	-	BLOCK	Tue Apr 22 2025 19:16:15 GMT-0700 (Pacific Daylight Time)
block-my-ip-addr	129.210.115.104 (US)	/	-	BLOCK	Tue Apr 22 2025 19:16:15 GMT-0700 (Pacific Daylight Time)

Note: the rule can be edited to Allow or CAPTCHA as well for testing purposes. The logs below show allow and CAPTCHA traffic as well after editing the rule.

Sampled requests (11)					
Samples of requests from the past 3 hours.					
Metric name	Source IP	URI	Rule inside rule group	Action	Time
dhrupal-aws-waf-project	129.210.115.104 (US)	/	-	ALLOW	Tue Apr 22 2025 19:24:02 GMT-0700 (Pacific Daylight Time)
dhrupal-aws-waf-project	129.210.115.104 (US)	/	-	ALLOW	Tue Apr 22 2025 19:24:05 GMT-0700 (Pacific Daylight Time)
block-my-ip-addr	129.210.115.104 (US)	/	-	CAPTCHA	Tue Apr 22 2025 19:24:02 GMT-0700 (Pacific Daylight Time)
block-my-ip-addr	129.210.115.104 (US)	/favicon.ico	-	ALLOW	Tue Apr 22 2025 19:23:49 GMT-0700 (Pacific Daylight Time)
block-my-ip-addr	129.210.115.104 (US)	/	-	CAPTCHA	Tue Apr 22 2025 19:23:52 GMT-0700 (Pacific Daylight Time)
block-my-ip-addr	129.210.115.104 (US)	/favicon.ico	-	BLOCK	Tue Apr 22 2025 19:16:15 GMT-0700 (Pacific Daylight Time)
block-my-ip-addr	129.210.115.104 (US)	/	-	ALLOW	Tue Apr 22 2025 19:23:49 GMT-0700 (Pacific Daylight Time)
block-my-ip-addr	129.210.115.104 (US)	/	-	ALLOW	Tue Apr 22 2025 19:21:54 GMT-0700 (Pacific Daylight Time)
block-my-ip-addr	129.210.115.104 (US)	/favicon.ico	-	BLOCK	Tue Apr 22 2025 19:21:54 GMT-0700 (Pacific Daylight Time)
block-my-ip-addr	129.210.115.104 (US)	/	-	BLOCK	Tue Apr 22 2025 19:16:15 GMT-0700 (Pacific Daylight Time)

Successfully configured an AWS WAF with an Application Load Balancer to access an EC2 instance with allow, deny, or CAPTCHA actions.