# AWS WAF

**What is AWS WAF?**

AWS WAF is a web application firewall that lets you monitor the HTTP(S) requests that are forwarded to your protected web application resources. You can protect the following resource types:

* Amazon CloudFront distribution
* Amazon API Gateway REST API
* Application Load Balancer
* AWS AppSync GraphQL API
* Amazon Cognito user pool
* AWS App Runner service
* AWS Verified Access instance

AWS WAF lets you control access to your content. Based on criteria that you specify, such as the IP addresses that requests originate from or the values of query strings, the service associated with your protected resource responds to requests either with the requested content, with an HTTP 403 status code (Forbidden), or with a custom response.

**What is Web ACL?**

A web ACL is a collection of rules that AWS WAF uses to inspect web requests. Each rule includes one top-level statement that AWS WAF evaluates for each web request it receives. Each statement contains one or more predicates that AWS WAF uses to identify the web requests that you want to block or allow.

* AWS WAF has a default web ACL that is already configured to block common web attacks, such as SQL injection or cross-site scripting. You can customize this ACL to add your own rules, and you can create new web ACLs to protect your web applications. You can also use AWS Firewall Manager to centrally configure and manage AWS WAF rules across your accounts and applications.
* AWS WAF is integrated with AWS Shield, a managed DDoS protection service that safeguards web applications running on AWS. There is no additional charge for AWS WAF. You pay only for what you use.

**## Availability of AWS WAF**

AWS WAF is available in the following regions: US East (N. Virginia), US West (Oregon), US West (N. California), EU (Ireland), EU (Frankfurt), Asia Pacific (Singapore), Asia Pacific (Sydney), Asia Pacific (Tokyo), Asia Pacific (Seoul), Asia Pacific (Mumbai), South America (São Paulo), and AWS GovCloud (US).

**## Where it can be used?**

AWS WAF is available in three flavors: AWS WAF on Application Load Balancer, AWS WAF on Amazon CloudFront and AWS WAF on API Gateway.

AWS WAF basically works on the layer 7 of the OSI model.

**## How it works?**

AWS WAF allows you to set up rules to block, allow, or monitor (count) web requests based on conditions that you specify, such as the IP addresses that requests originate from or values in the requests. Conditions can include IP addresses, HTTP headers, HTTP body, or URI strings, and rules can be set up to apply to all requests or just those that match a pattern that you specify.

The Rule actions are:

- Allow (default)

- Block

- Count

**## How does Rules work?**

* A rule is a combination of one or more conditions. AWS WAF evaluates each rule against web requests based on the priority that you assign to the rule. AWS WAF processes only the rules that you assign the lowest priority, in order, starting from the lowest numbered priority. When a web request matches all the conditions in a rule, AWS WAF performs the action that is specified for that rule, either allow or block, and doesn’t evaluate the request against the rules with lower priority.
* Types of Rules:
  + Rate based rules
  + Regular rules
  + Custom rules
* Rate based rules: You can create rate based rules to block requests based on the rate of requests that come from individual IP addresses. You can also create rate based rules to block requests based on the rate of requests that come from individual IP addresses to specific URLs or query string parameters. For example, you can use rate based rules to protect your resources from bad bots that perform web scraping or slow down your site with automated vulnerability scans. You can also use rate based rules to protect against DDoS attacks by limiting the rate of incoming requests.
* Rate-based rules helps to protect against DDoS attacks by limiting the rate of incoming requests.
* Regular Rules: You can create regular rules to block or allow requests based on the values in the request. For example, you can create a rule that blocks requests that contain a specific string in the User-Agent header or requests that contain a specific value in a query string parameter. You can also create rules that block requests based on the length of a request, the length of a query string, or the length of any of the components of the request. You can also create rules that block requests based on the presence of SQL injection or cross-site scripting (XSS) strings in the request.
* Custom Rules: You can create custom rules to block or allow requests based on the values in the request. For example, you can create a rule that blocks requests that contain a specific string in the User-Agent header or requests that contain a specific value in a query string parameter. You can also create rules that block requests based on the length of a request, the length of a query string, or the length of any of the components of the request. You can also create rules that block requests based on the presence of SQL injection or cross-site scripting (XSS) strings in the request.

**## How does it evaluate the rules?**

If you have multiple rules in a web ACL, AWS WAF evaluates each request against them in order, starting from the lowest priority rule. The first rule that matches the request is the rule that AWS WAF uses to determine how to respond. If none of the rules in a web ACL match, AWS WAF applies the default action for the web ACL.

# Step 1 AWS WAF --->Create web ACL 🡪 select the region -🡪 name

# step2: Select the Associated AWS resources – optional in the below click next

# Step3: Add the rules are create Owen rules

# Step 4: select the paid rule, and free rules and Aws rules and third-party rules

# 

# Free rules in the select any one In my case selecting the admin protection

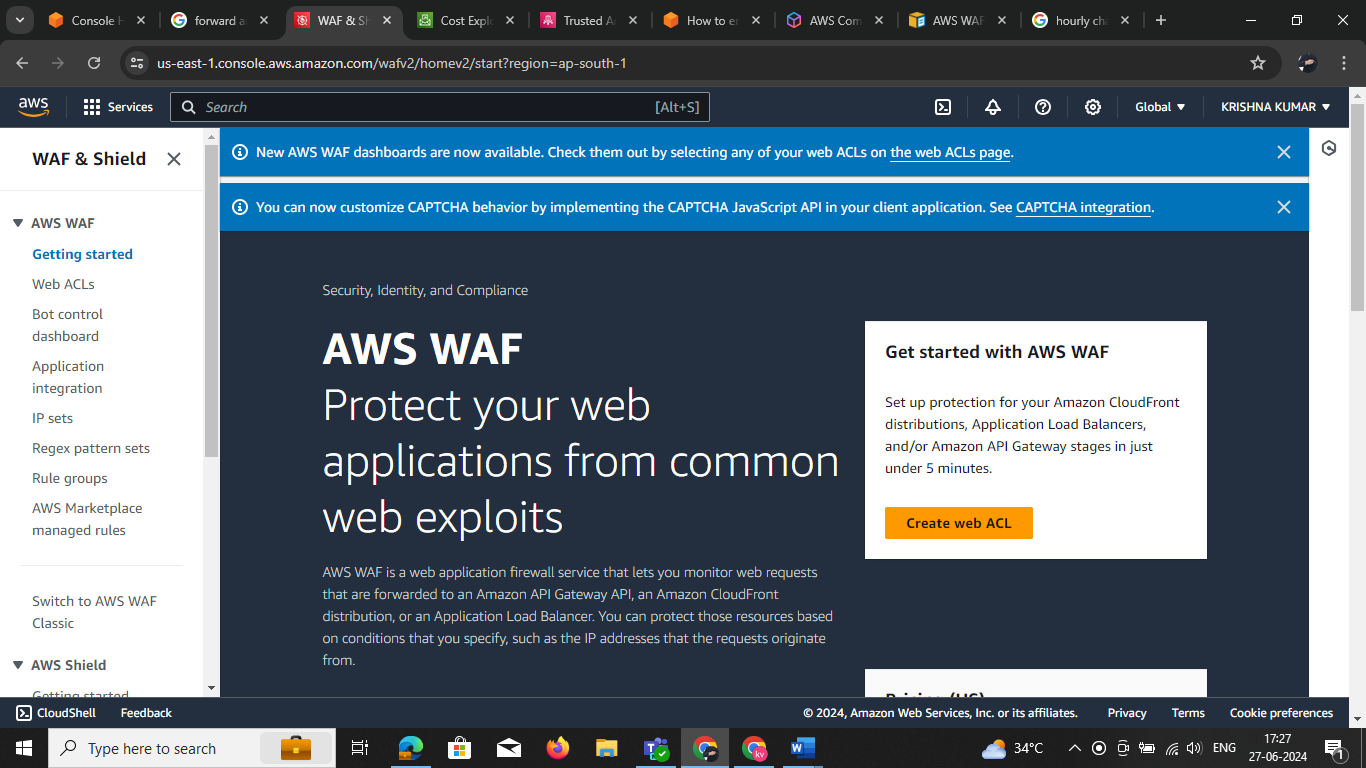
# Step 5 : Set rule priority and click on next

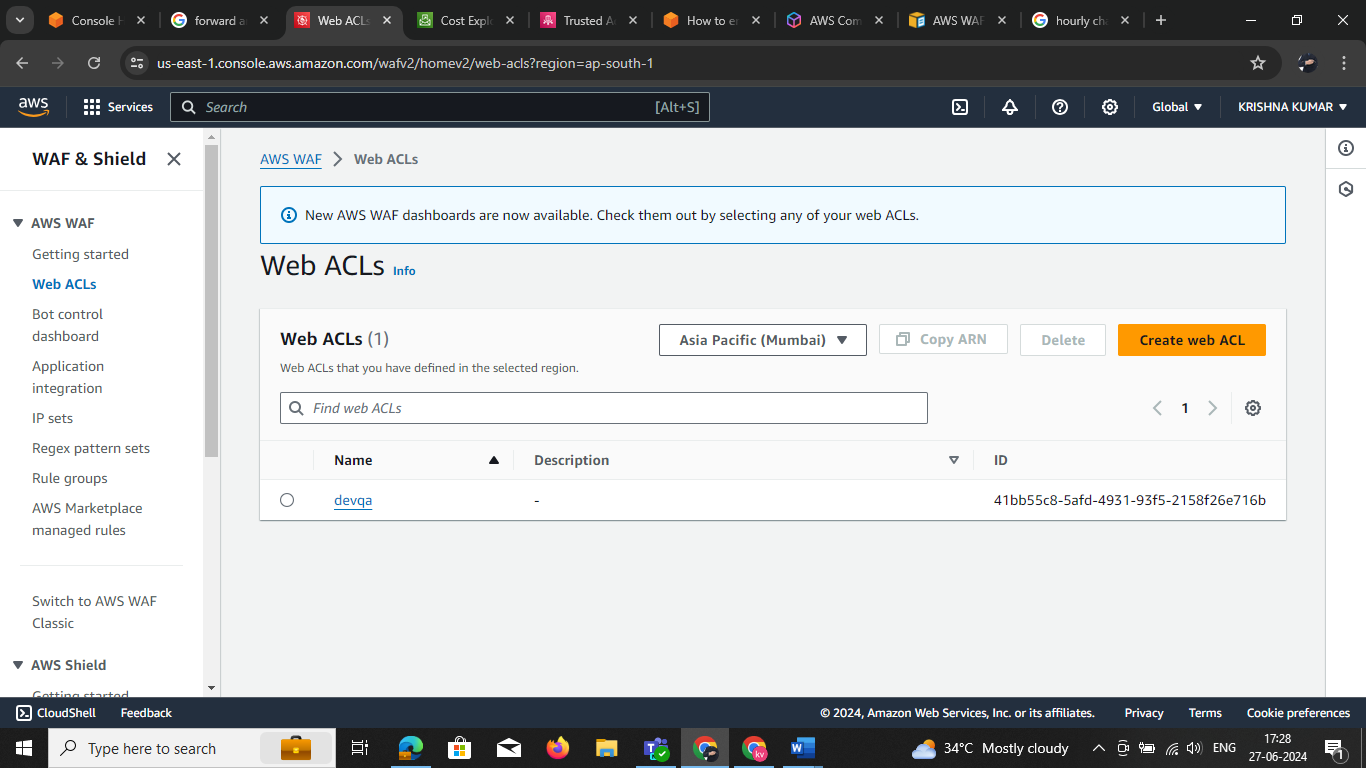
# Step 6 Configure metrics

# Step 7 : Review and create web ACL click on the create

# 

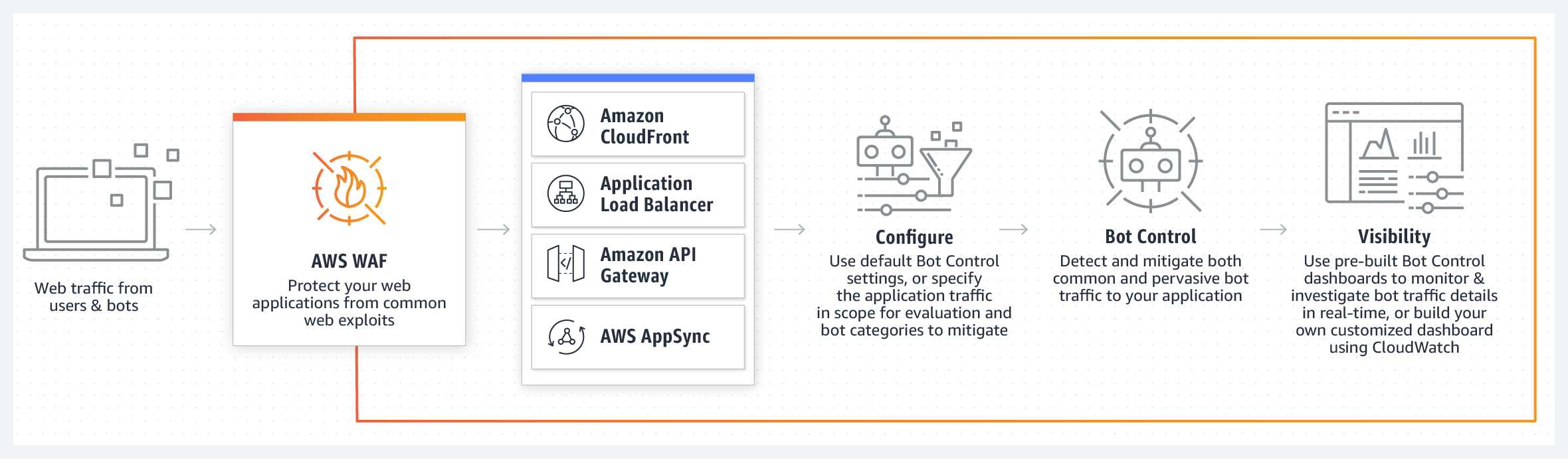
Ones create the

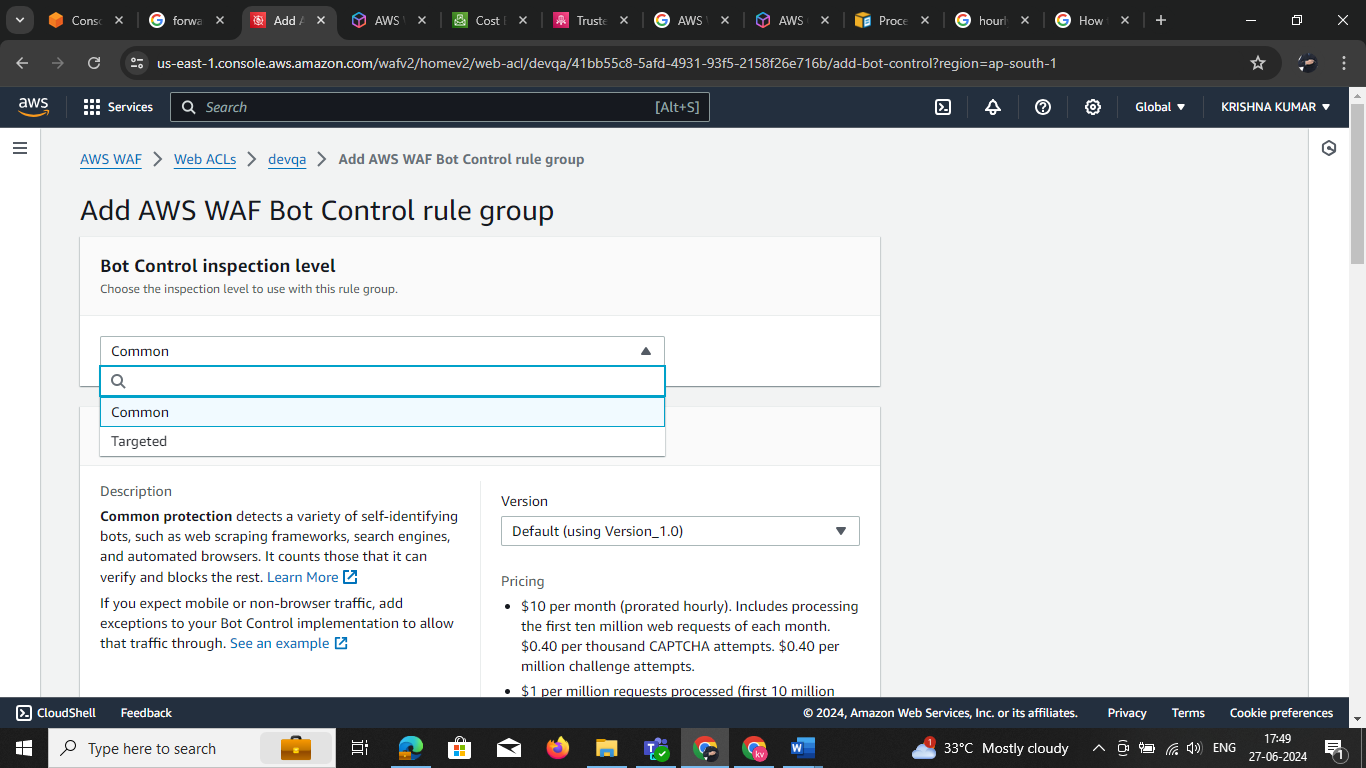




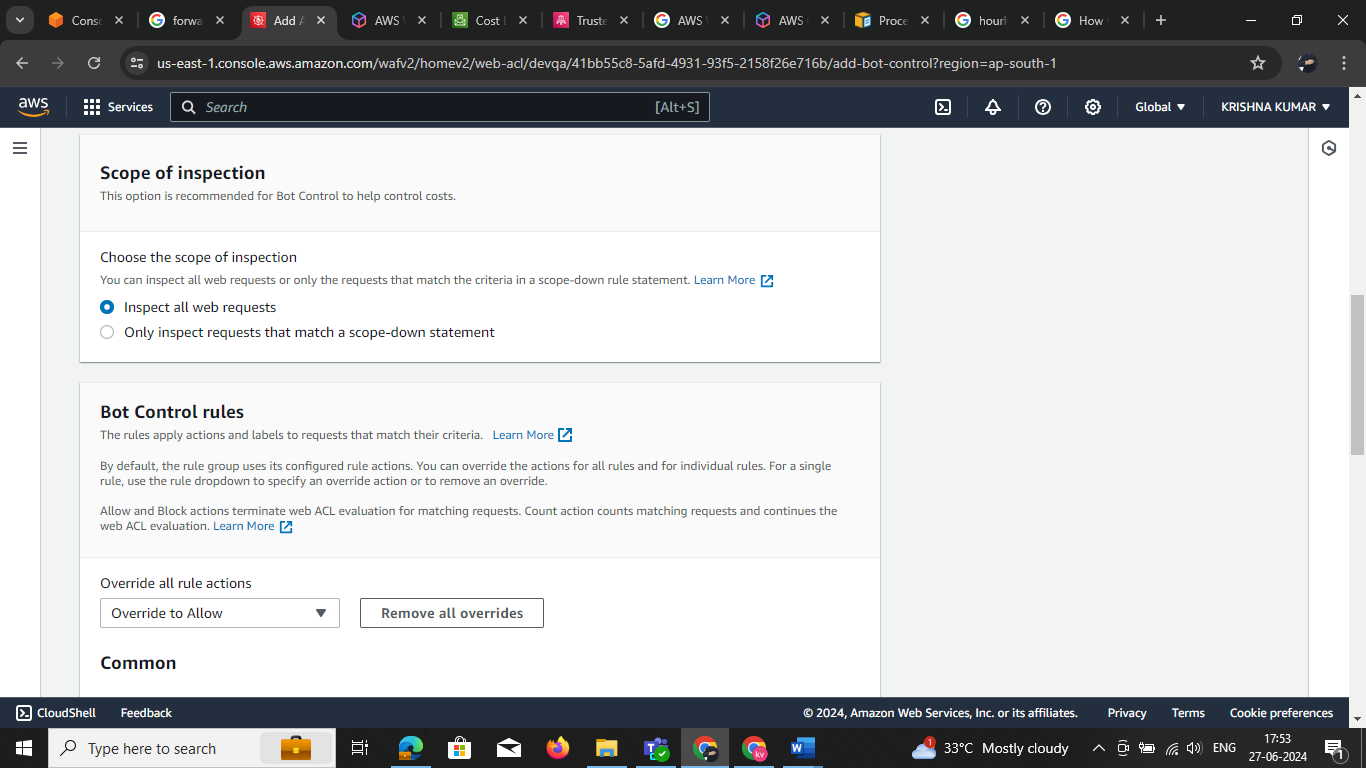
# AWS WAF Bot Control

* AWS WAF Bot Control gives you visibility and control over common and pervasive bot traffic that can consume excess resources, skew metrics, cause downtime, or perform other undesired activities. With just a few clicks, you can use the Bot Control managed rule group to block or rate-limit pervasive bots, such as scrapers, scanners, and crawlers, or you can allow common bots, such as status monitors and search engines. The Bot Control managed rule group can be used alongside other Managed Rules for WAF or with your own custom WAF rules to protect your applications.
* Bot Control enables you to monitor bot traffic activity with dashboards that provide detailed, real-time visibility into bot categories, identities, and other bot traffic details. You can use AWS Firewall Manager to deploy Bot Control for your web applications across multiple accounts in your AWS Organization.



step 1: select the bot select any one

* select the scope action



## **IP sets**

The table of IP sets shows all of your existing sets. You can search for a set by name, and you can also filter by Region. To view the details for a specific set in the table, choose the name of the set to open the details page.

To delete an IP set, choose the option to the left of the set, and then choose **Delete**

**Step 1 : create ip**

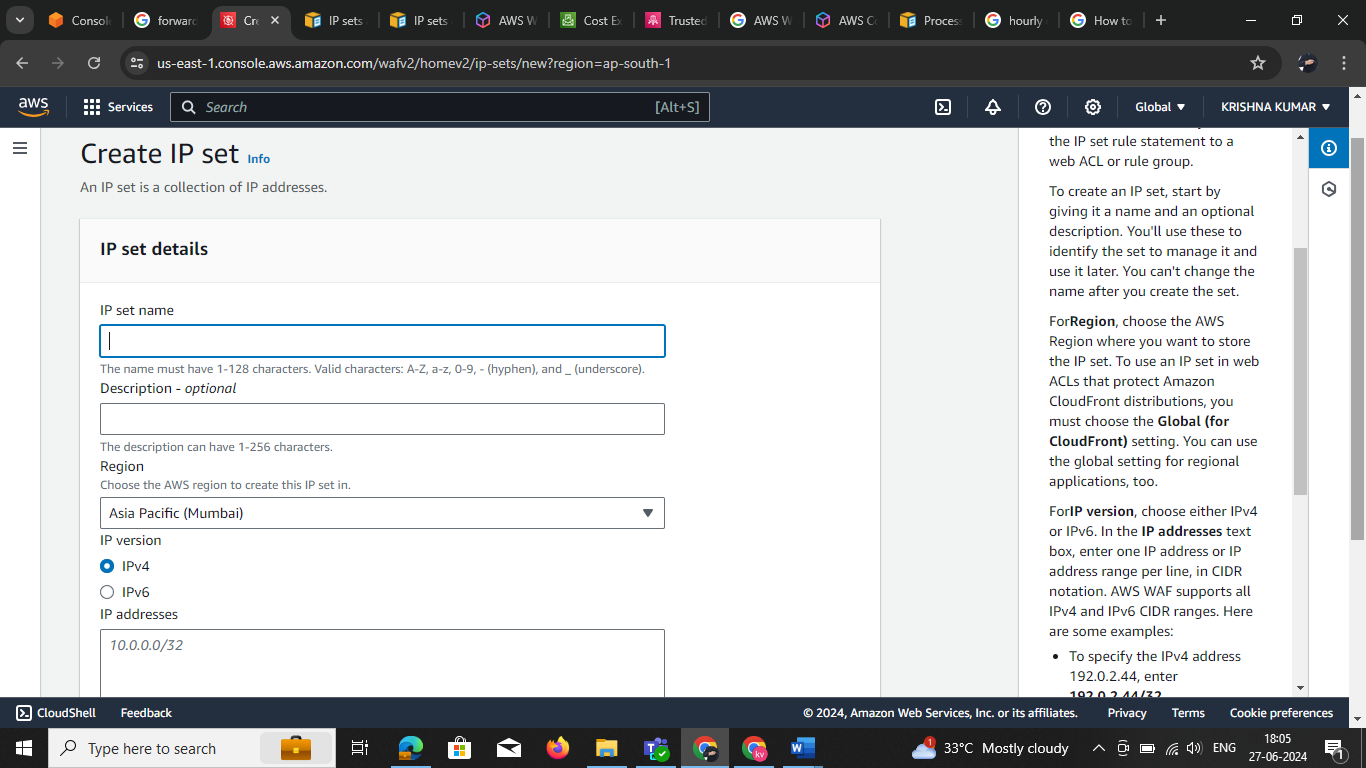
An IP set is a collection of IP addresses and IP address ranges that you use in a rule statement. To use an IP set in a web ACL or rule group, you first create an IP set with your IP address specifications. Then you reference the set when you add the IP set rule statement to a web ACL or rule group.

To create an IP set, start by giving it a name and an optional description. You'll use these to identify the set to manage it and use it later. You can't change the name after you create the set.

For**Region**, choose the AWS Region where you want to store the IP set. To use an IP set in web ACLs that protect Amazon CloudFront distributions, you must choose the **Global (for CloudFront)** setting. You can use the global setting for regional applications, too.

For**IP version**, choose either IPv4 or IPv6. In the **IP addresses** text box, enter one IP address or IP address range per line, in CIDR notation. AWS WAF supports all IPv4 and IPv6 CIDR ranges. Here are some examples:

* To specify the IPv4 address 192.0.2.44, enter **192.0.2.44/32**.
* To specify the IPv6 address 0:0:0:0:0:ffff:c000:22c, enter **0:0:0:0:0:ffff:c000:22c/128**.
* To specify the range of IPv4 addresses from 192.0.2.0 to 192.0.2.255, enter **192.0.2.0/24**.
* To specify the range of IPv6 addresses from 2620:0:2d0:200:0:0:0:0 to 2620:0:2d0:200:ffff:ffff:ffff:ffff, enter **2620:0:2d0:200::/64**.



Create regex pattern set

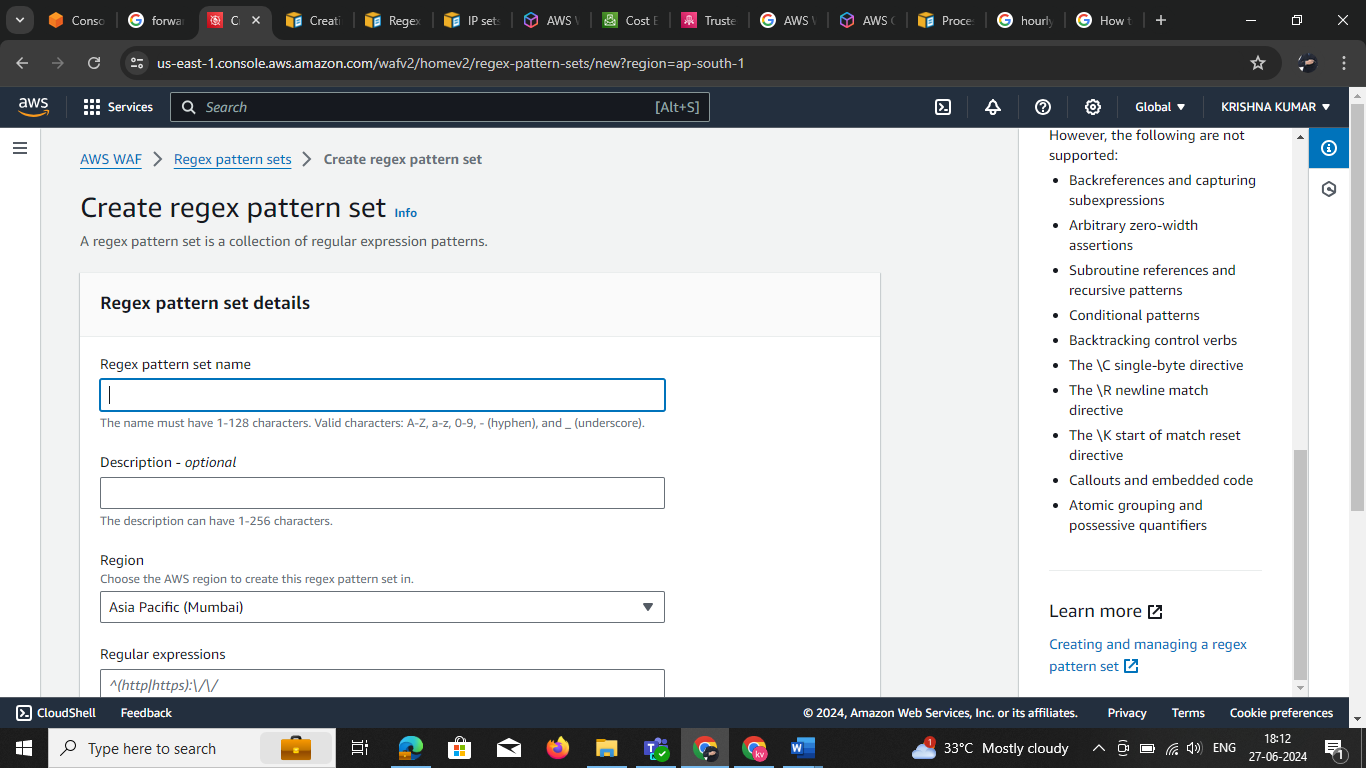
A regex pattern set is a collection of regular expressions. If your regex pattern set contains more than one regex pattern, when it's used in a rule the pattern matching is combined with an OR. That is, a web request will match the pattern set rule statement if the request component matches any of the patterns in the set.

To create an regex pattern set, start by giving it a name and an optional description. You'll use these to identify the set to manage it and use it later. You can't change the name after you create the set.

For**Region**, choose the AWS Region where you want to store the regex pattern set. To use a regex pattern set in web ACLs that protect Amazon CloudFront distributions, you must choose the **Global (for CloudFront)** setting. You can use the global setting for regional applications, too.

AWS WAF supports most standard Perl Compatible Regular Expressions (PCRE). However, the following are not supported:

* Backreferences and capturing subexpressions
* Arbitrary zero-width assertions
* Subroutine references and recursive patterns
* Conditional patterns
* Backtracking control verbs
* The \C single-byte directive
* The \R newline match directive
* The \K start of match reset directive
* Callouts and embedded code
* Atomic grouping and possessive quantifiers



## **Rule groups**

The table of rule groups shows all of your existing groups. You can search for a group by name, and you can also filter by region. To view the details for a specific group in the table, choose the name of the group to open the details page.

To delete a rule group, choose the option to the left of the group, and then choose **Delete**.

.