Project: Serverless E-Commerce Order Processing System on AWS

GitHub Repo: https://github.com/Mithra1995/serverless-project

Objective:

Develop a secure, scalable, and cost-efficient e-commerce order processing system using serverless AWS services, including a proper web-based UI and web application firewall (WAF) for protection against common web threats.

High-Level Architecture:

- Frontend: Hosted on Amazon S3 (static website) + CloudFront (CDN)
- API Layer: Managed through API Gateway
- Business Logic: Handled by AWS Lambda
- Database: Powered by DynamoDB
- Media Storage: Product images and frontend assets stored in S3
- Security: AWS WAF integrated with CloudFront & API Gateway

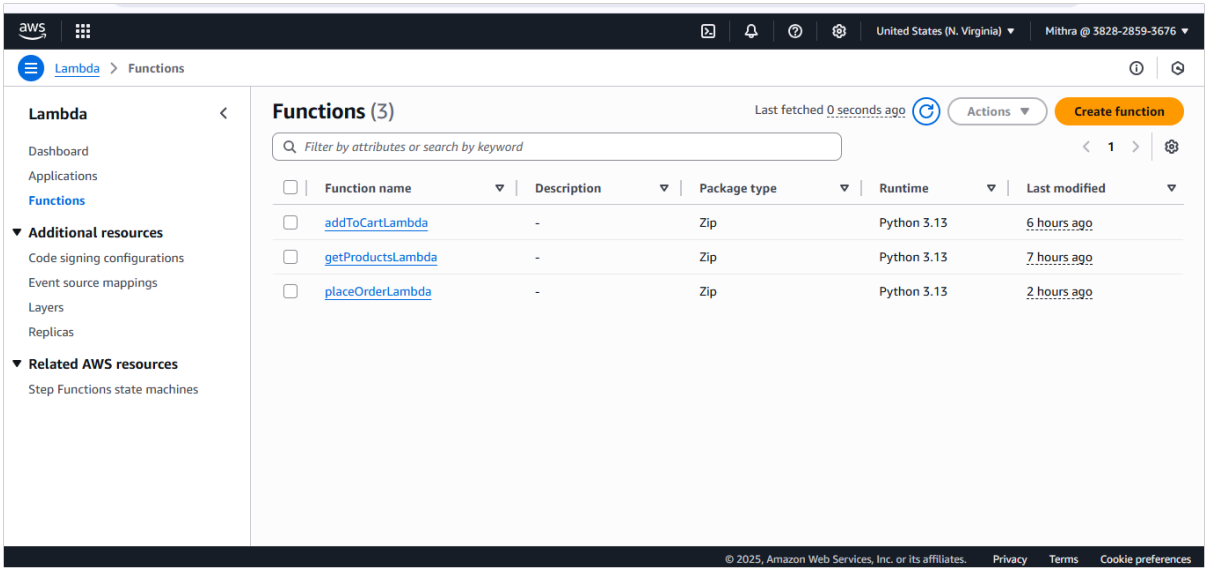
✨ Key Features

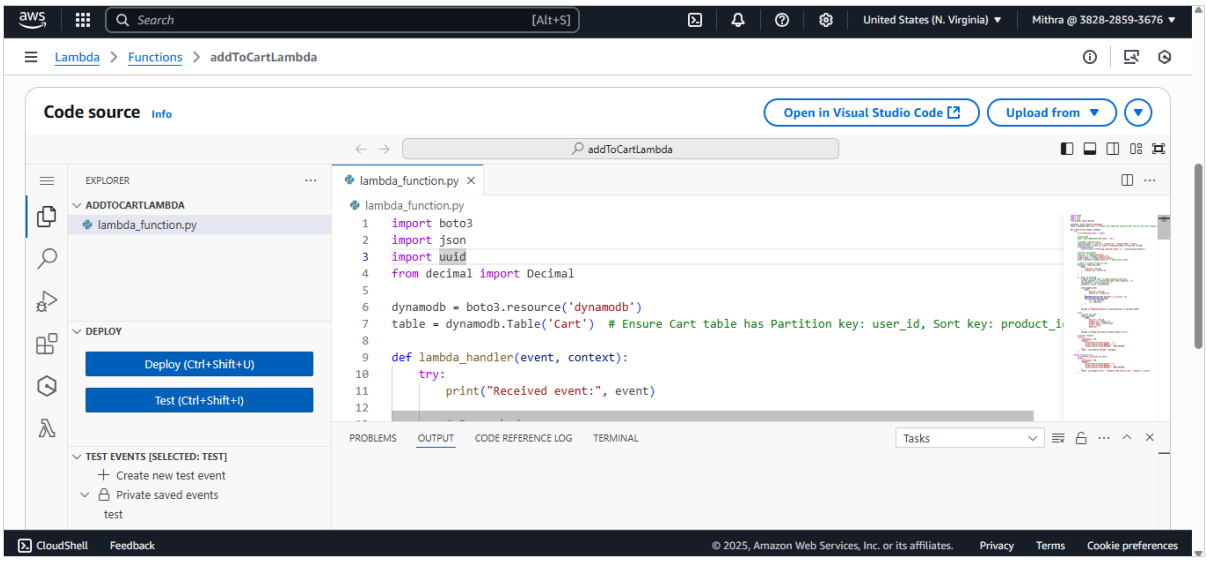
- Product listing UI
- Add to cart & order placement
- Order data stored securely in DynamoDB
- Secured API endpoints using AWS WAF

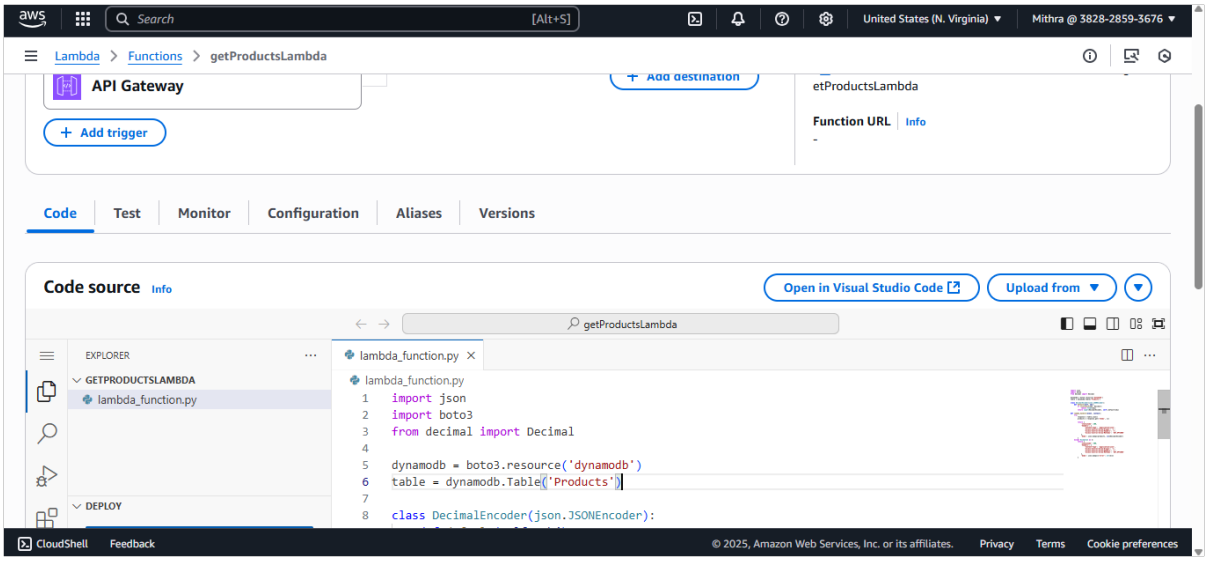
Step by Step Implementation:

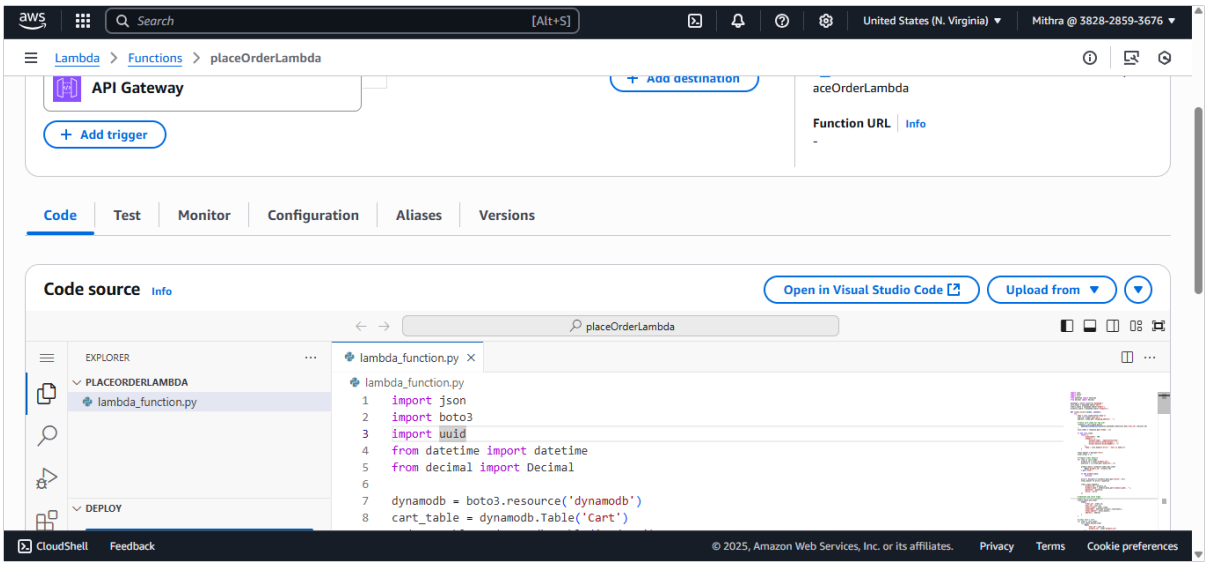
Step 1: Create Lambda function for Listing products, Add to cart and order placement

- `listProductsLambda`
- `addToCartLambda`
- `placeOrderLambda`

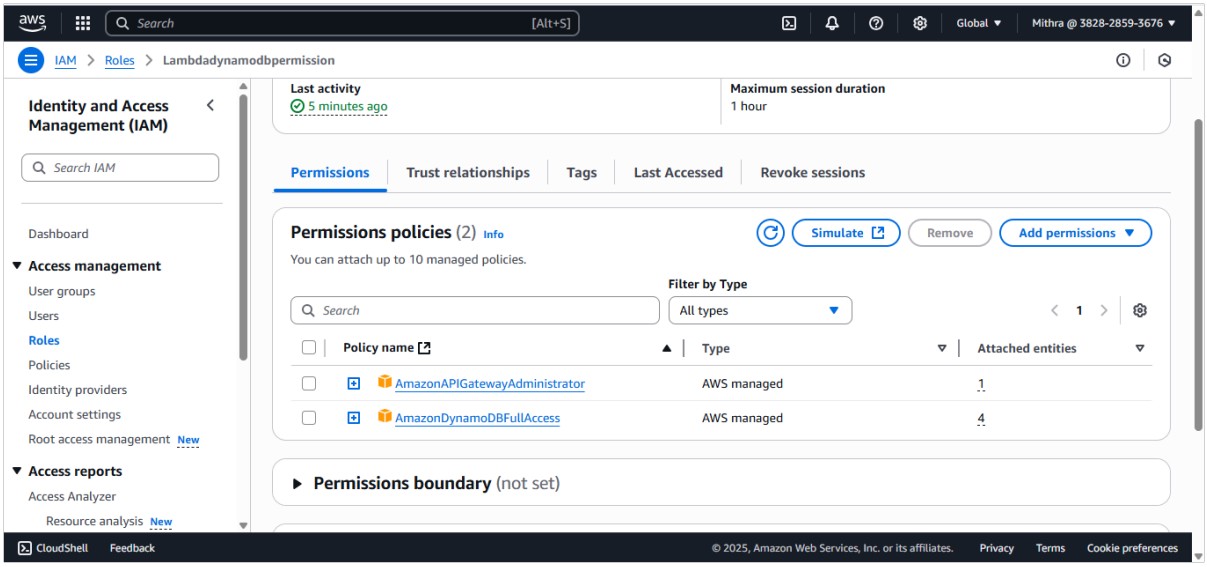

United States (N. Virginia)
Mithra @ 3828-2859-3676
Lambda > Functions
Lambda
Dashboard
Applications
Functions
Additional resources
Code signing configurations
Event source mappings
Layers
Replicas
Related AWS resources
Step Functions state machines
Functions (3)
Last fetched 0 seconds ago
Actions
Create function
Filter by attributes or search by keyword
Function name
Description
Package type
Runtime
Last modified
addToCartLambda - Zip Python 3.13 6 hours ago
getProductsLambda - Zip Python 3.13 7 hours ago
placeOrderLambda - Zip Python 3.13 2 hours ago
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Search [Alt+S]
United States (N. Virginia)
Mithra @ 3828-2859-3676
Lambda > Functions > addToCartLambda
Code source Info
Open in Visual Studio Code
Upload from
addToCartLambda
EXPLORER
ADDTOCARTLAMBDA
lambda_function.py
DEPLOY
Deploy (Ctrl+Shift+U)
Test (Ctrl+Shift+I)
TEST EVENTS [SELECTED: TEST]
Create new test event
Private saved events
test
lambda_function.py
import boto3
import json
import uuid
from decimal import Decimal
dynamodb = boto3.resource('dynamodb')
table = dynamodb.Table('Cart') # Ensure Cart table has Partition key: user_id, Sort key: product_i
def lambda_handler(event, context):
try:
print("Received event:", event)
PROBLEMS
OUTPUT
CODE REFERENCE LOG
TERMINAL
Tasks
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

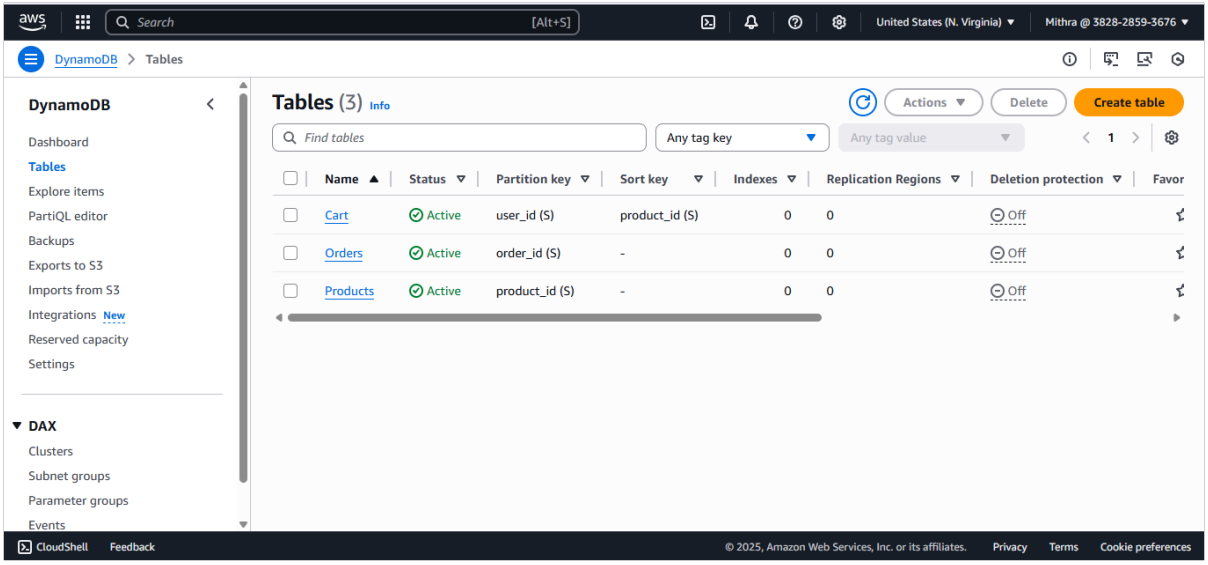




Step 2: Add IAM permission to lambda role to access DynamoDB



Step 3: Create DynamoDB table for cart, order and products table



Step 4: Add List or Products in DynamoDB table with productid, product name, price and Image URL

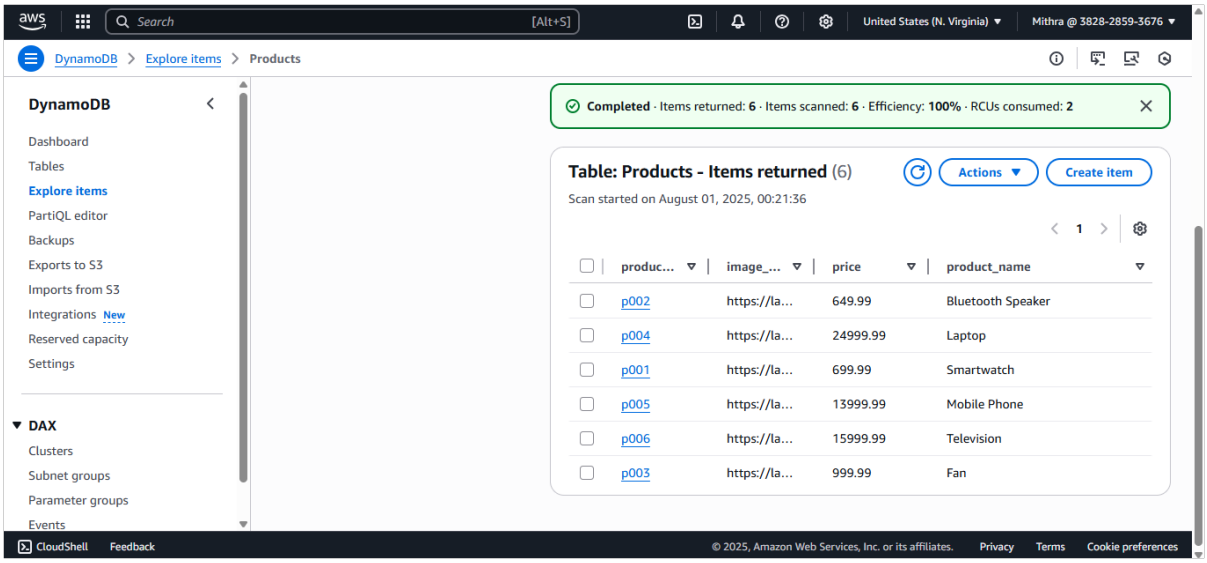
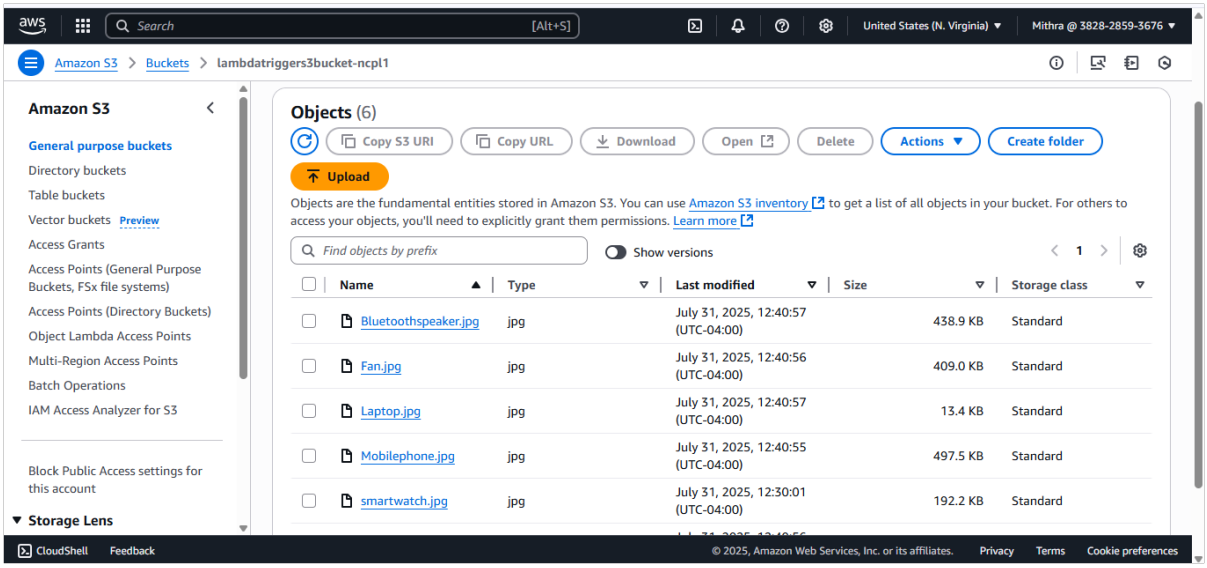
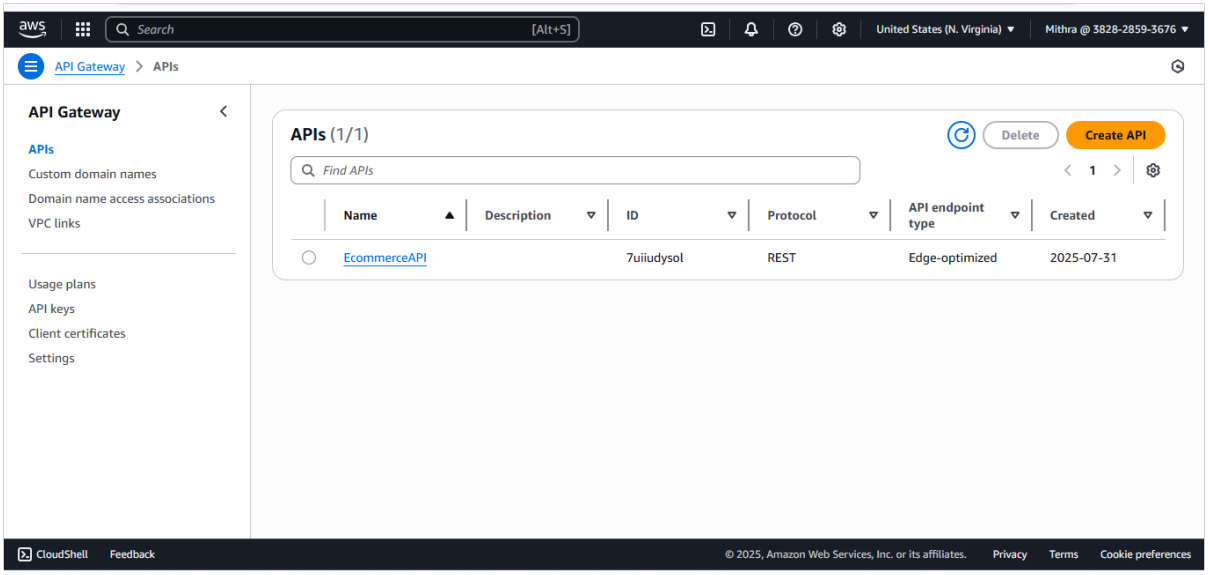
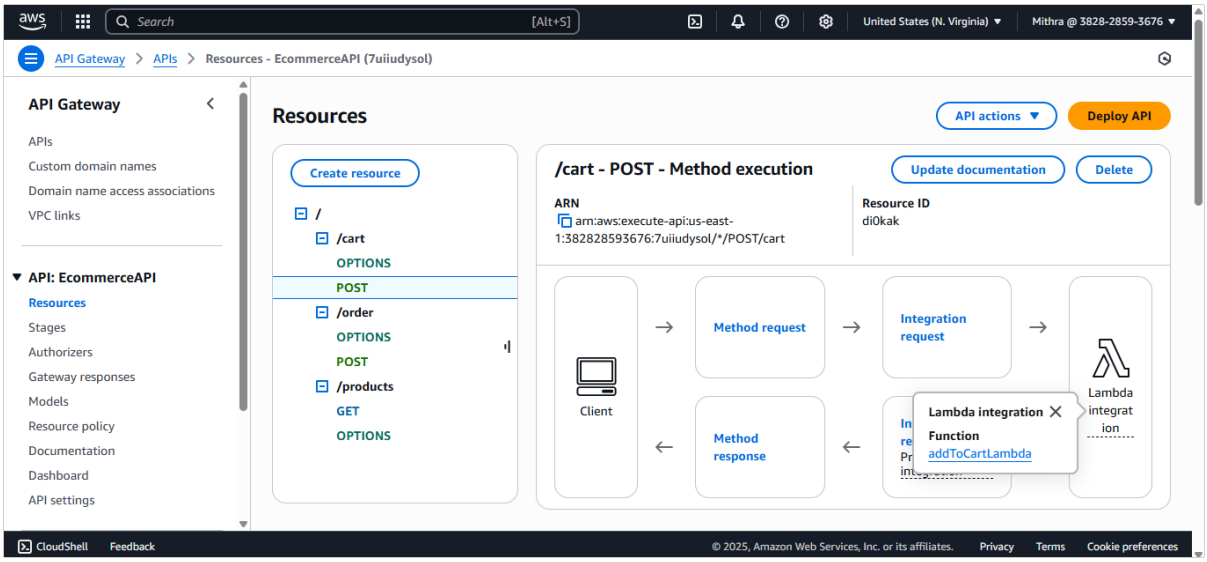


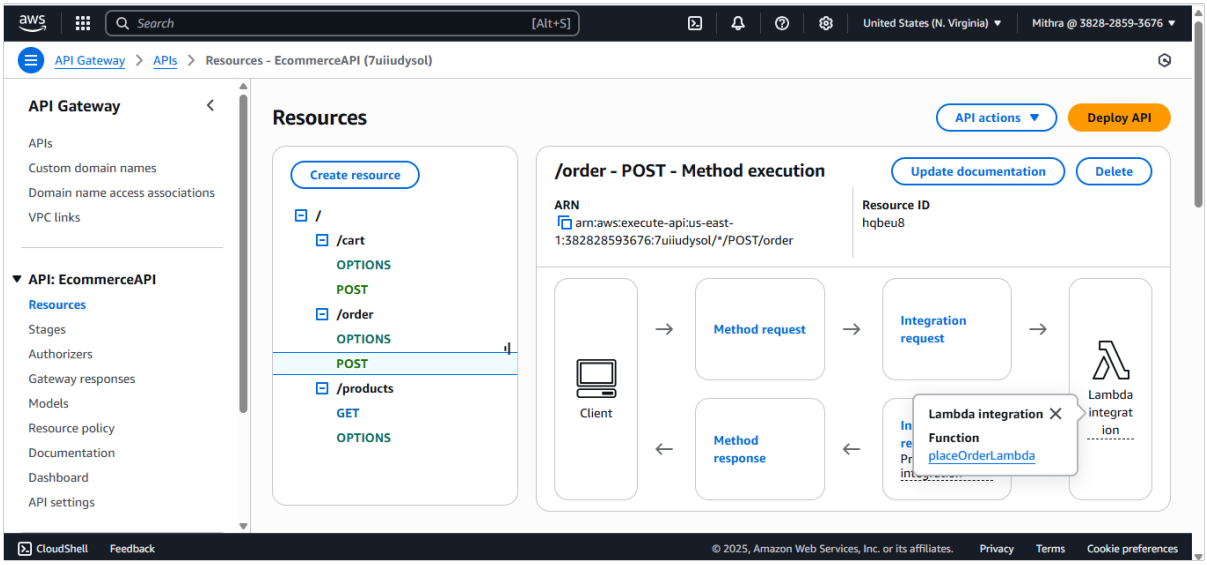
Image stored in s3

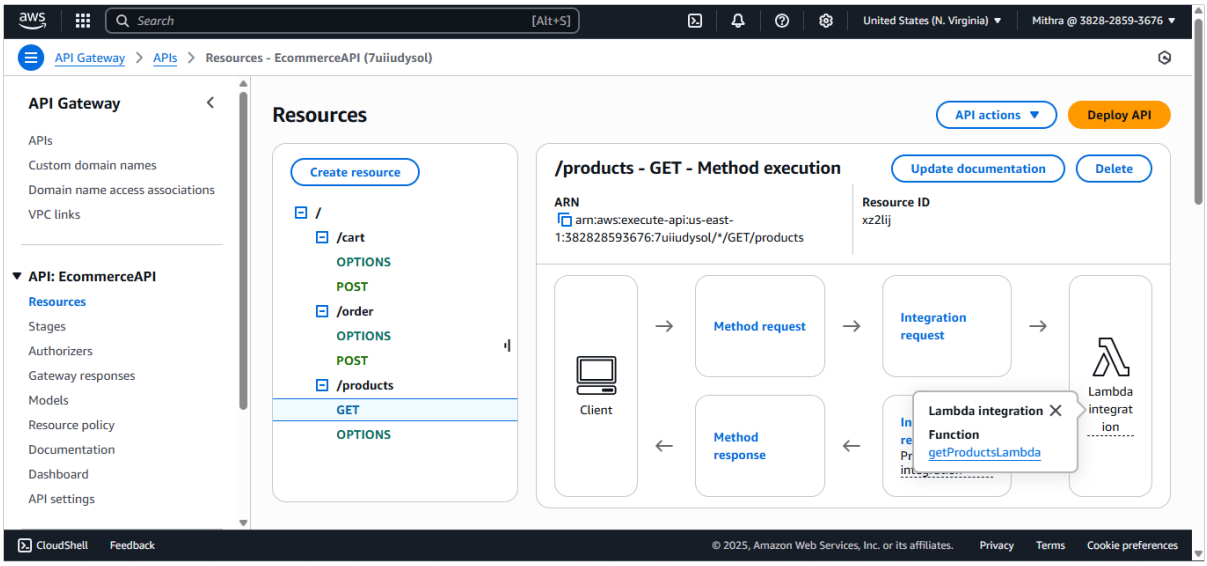


Step 5: Create API gateway as a Endpoints to trigger lambda function for cart, order and products

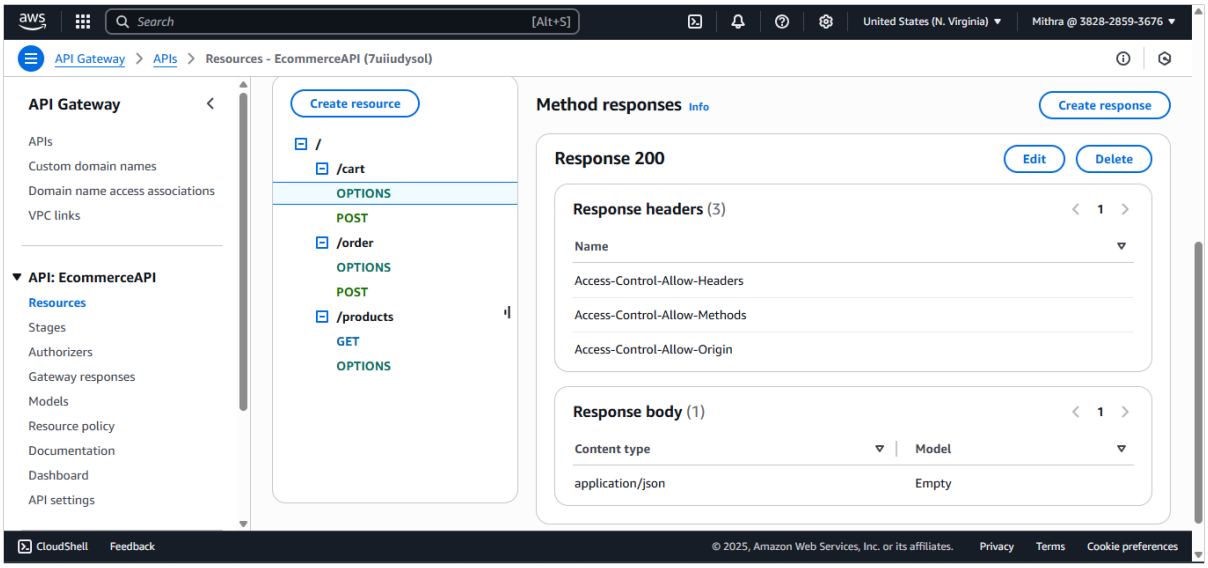




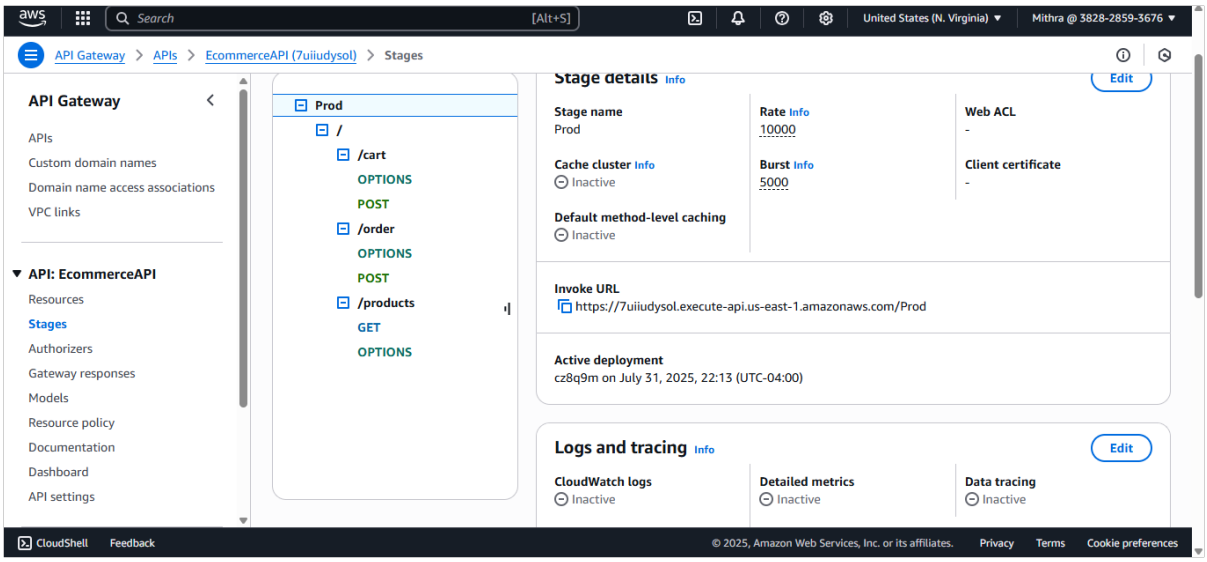




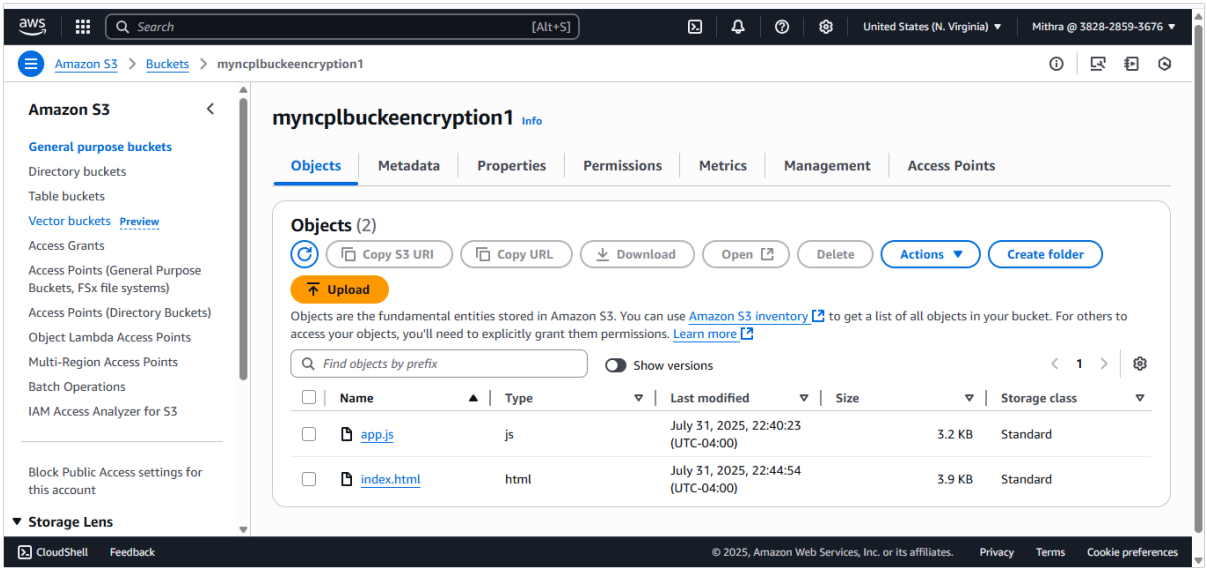
Step 6: Enable CORS for all the API Endpoints

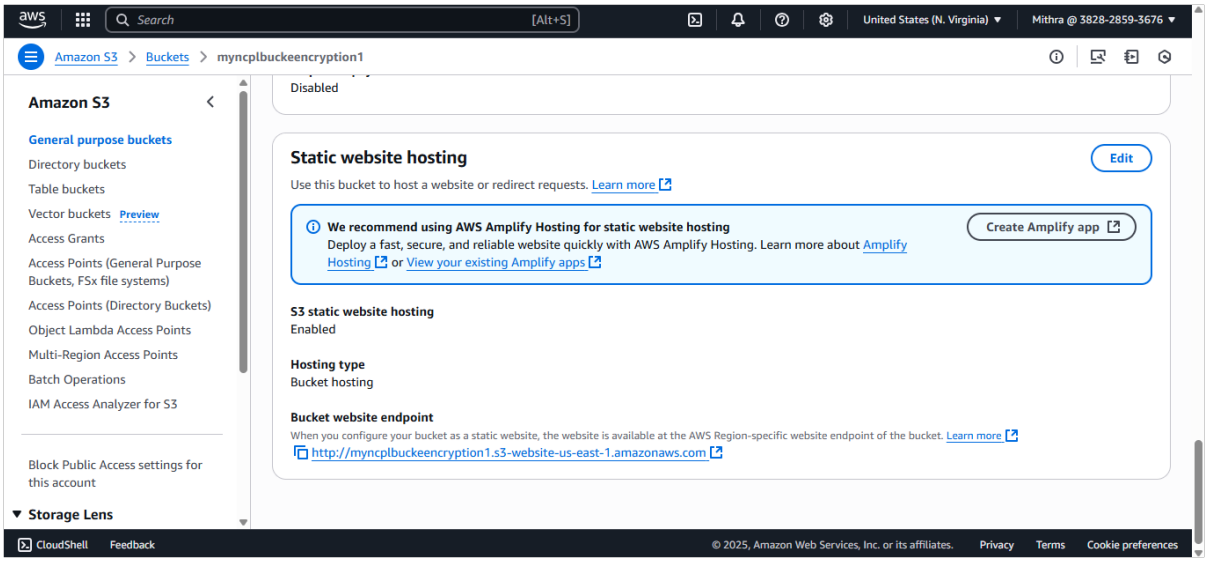


Step 7: Deploy API and name it as “Prod”

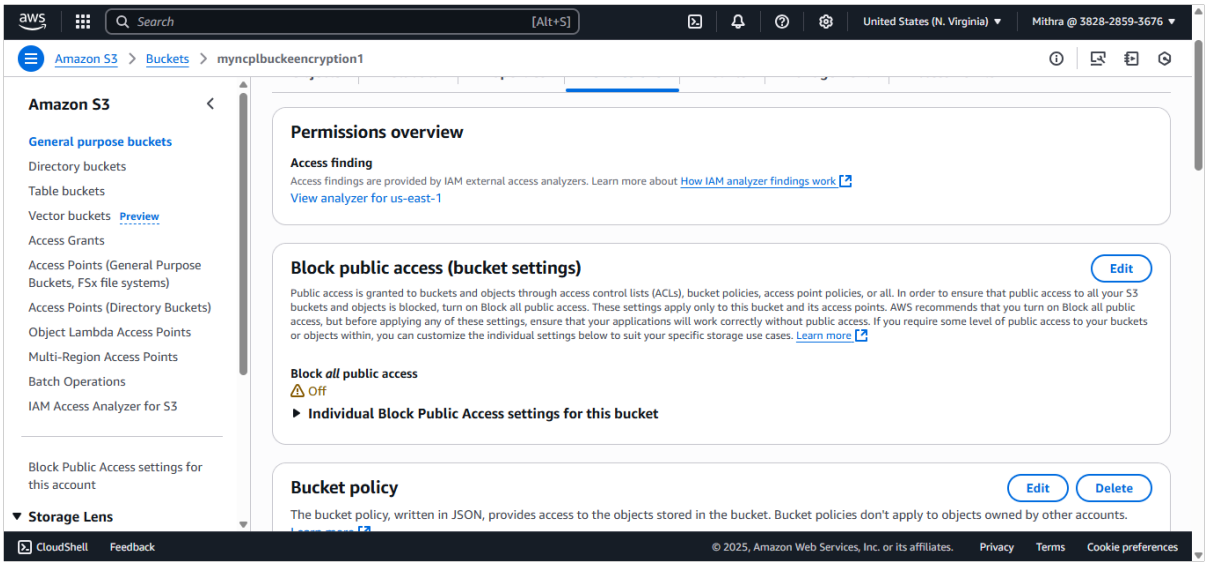


Step 8: Add index.html and app.js in another s3 bucket by enabling static web hosting

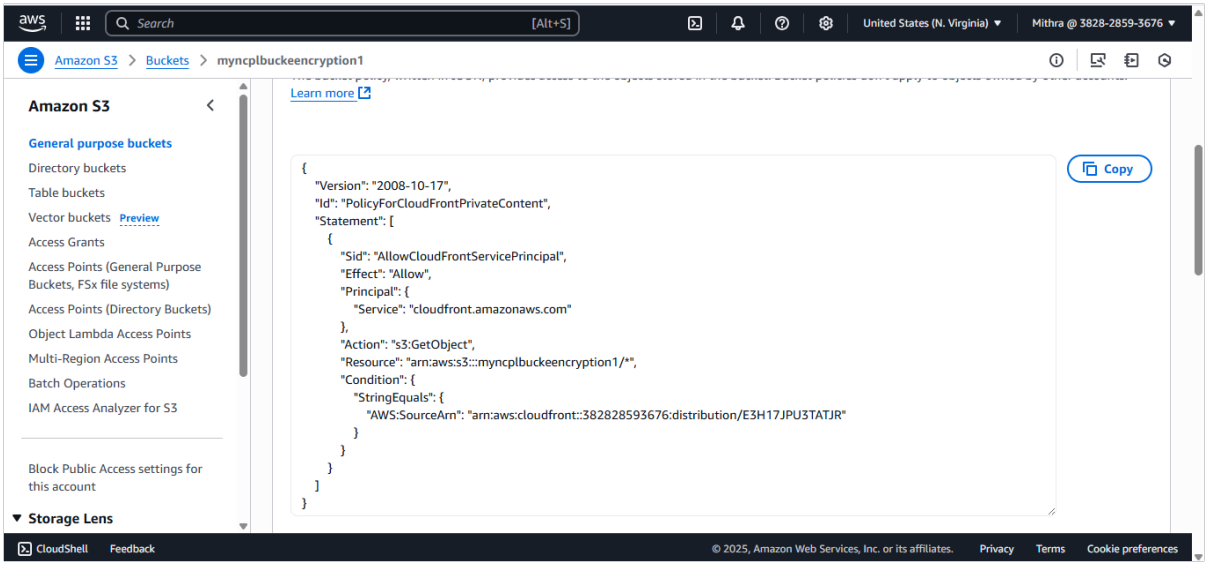


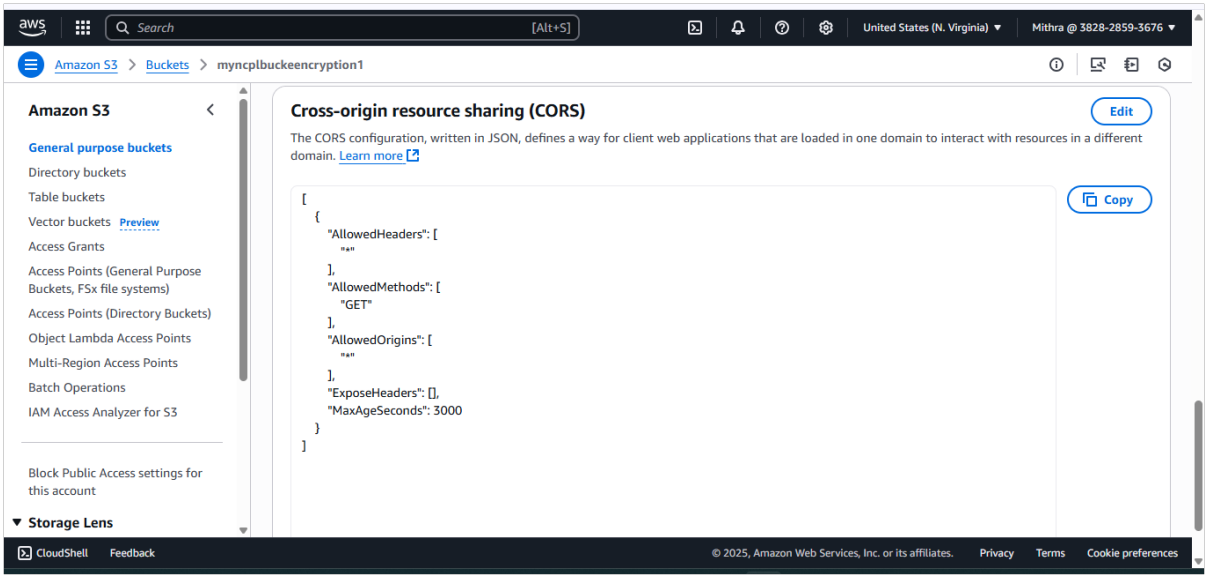


Step 9: Disable Block public access to access the website publicly

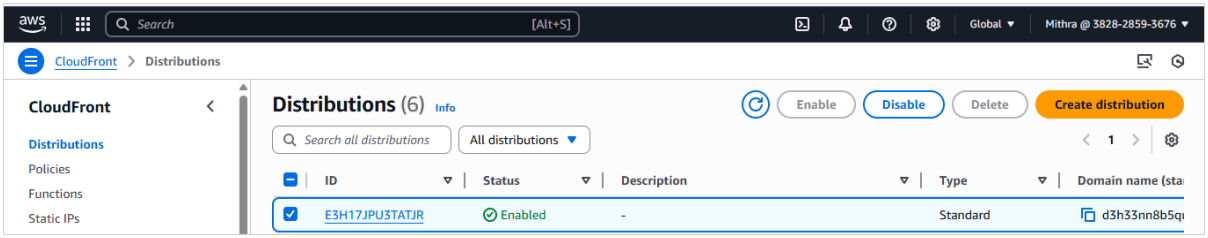


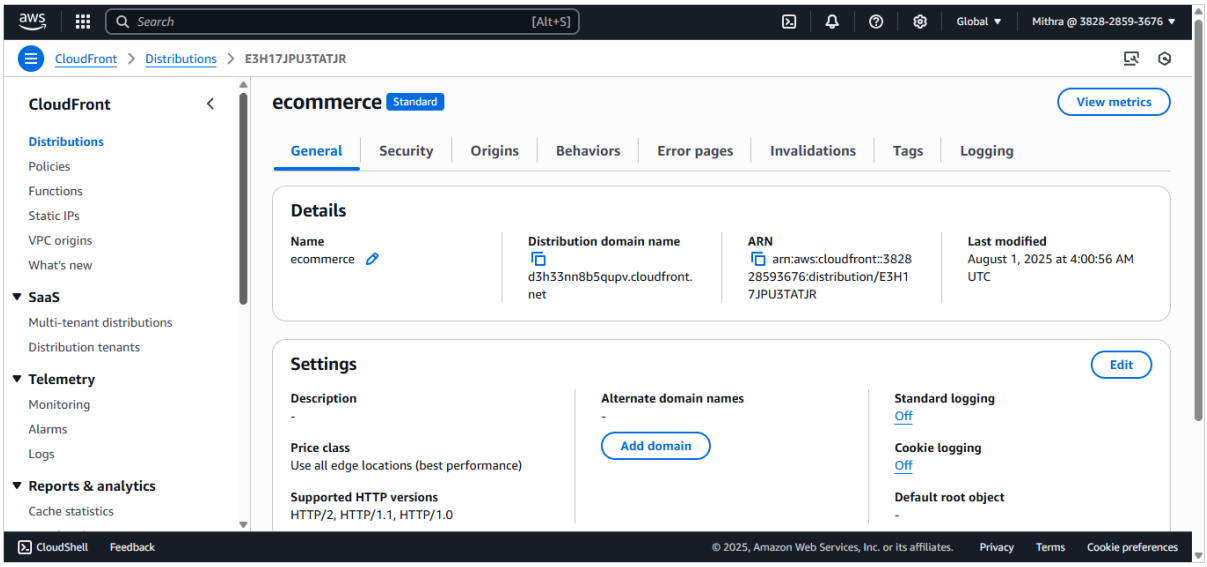
Step 10 : Add bucket policy to access S3 using CloudFront

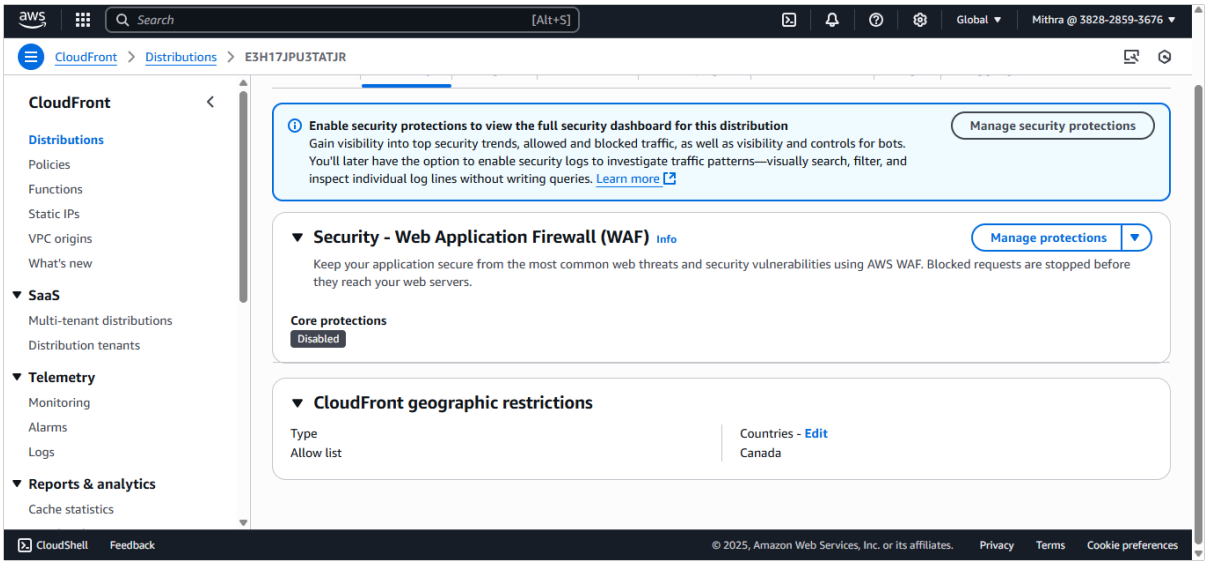




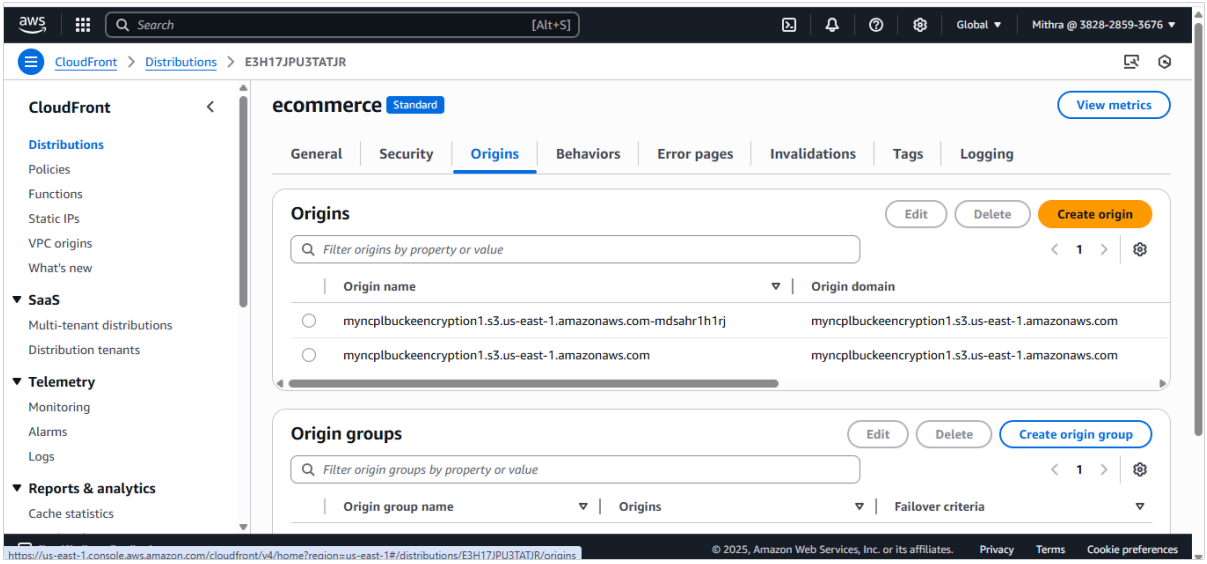
Step 11: Create CloudFront for S3





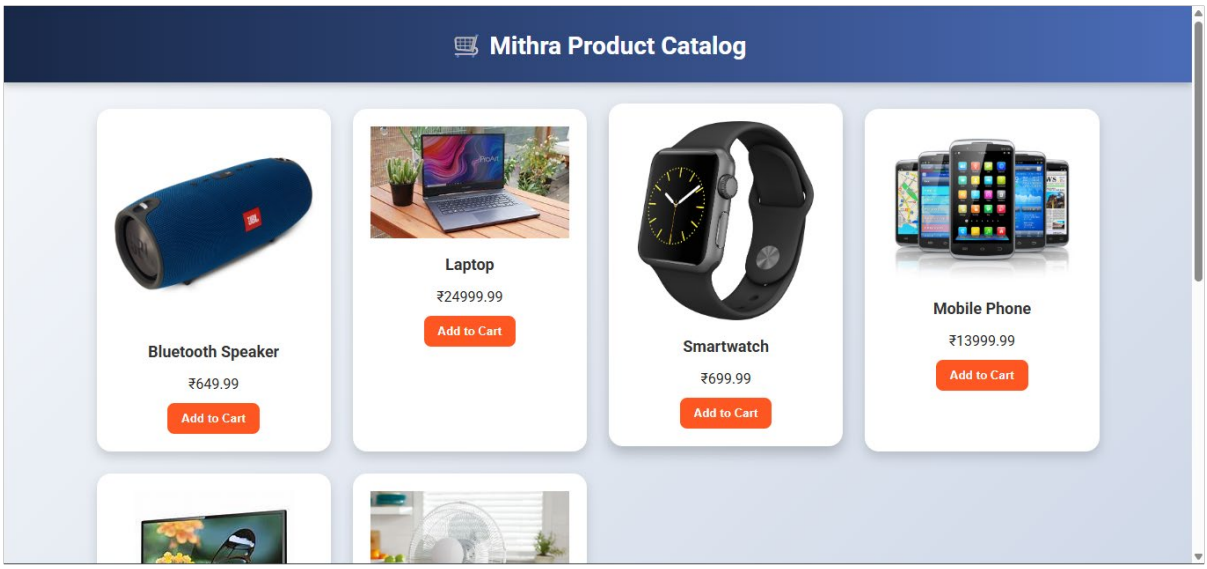


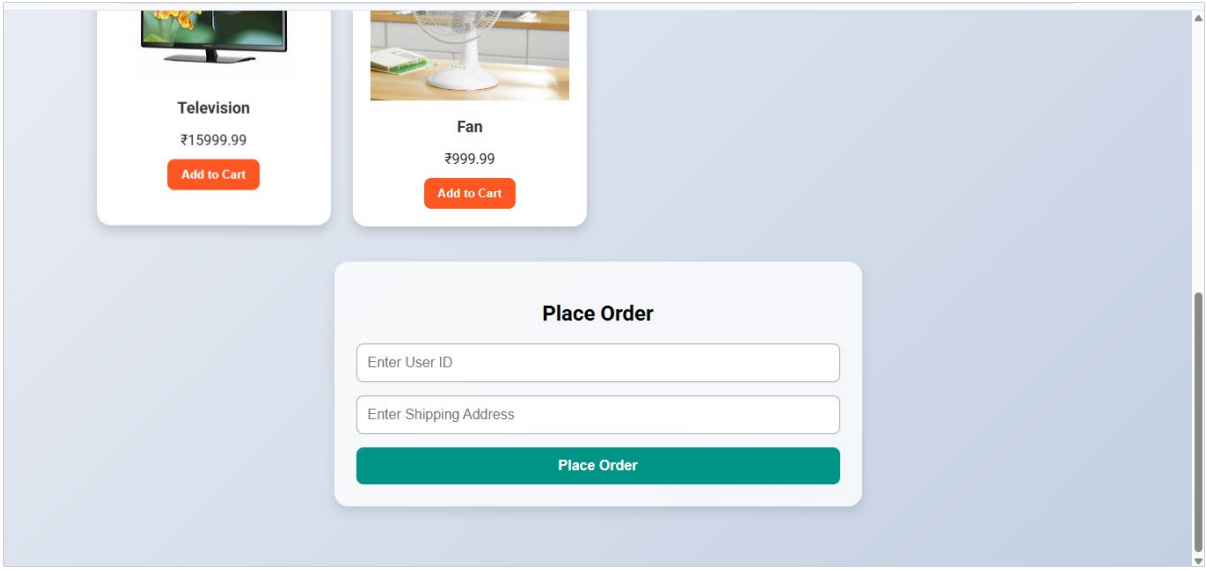
Step 12: Add OAC for the CloudFront and update the same in S3 bucket policy



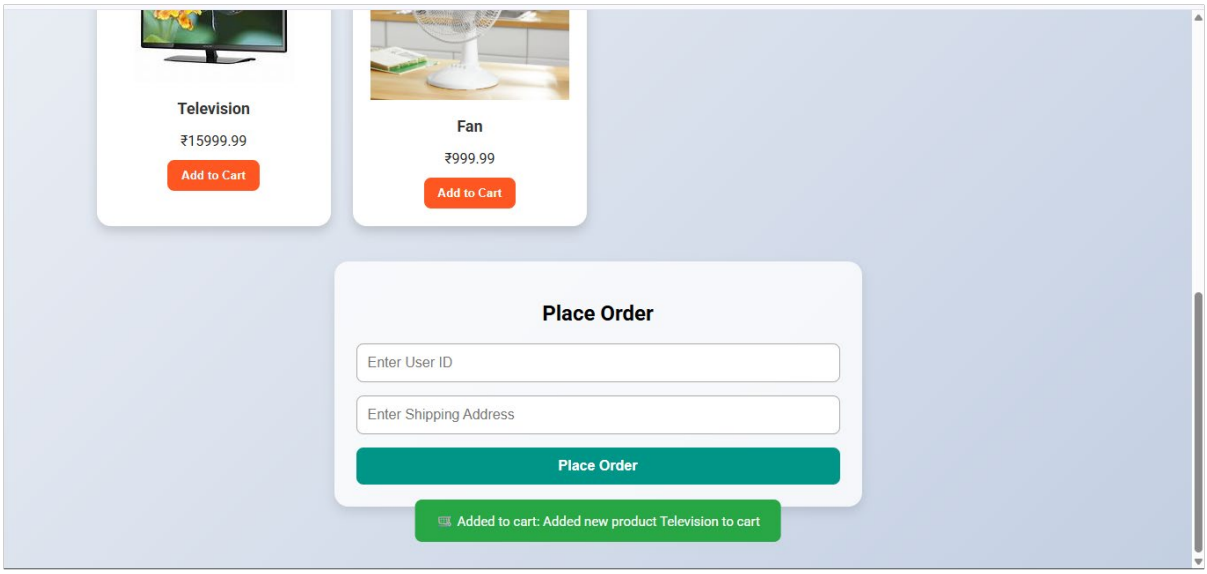
Step 13: Access the CloudFront webpage using below URL

https://d3h33nn8b5qupv.cloudfront.net/index.html and list of products will get displayed

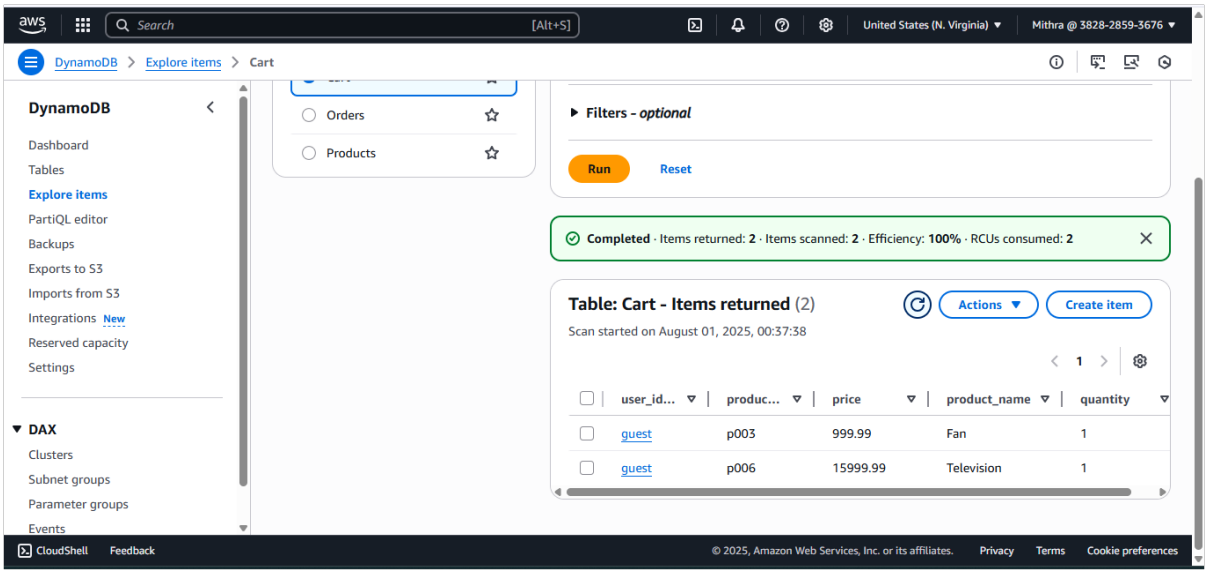




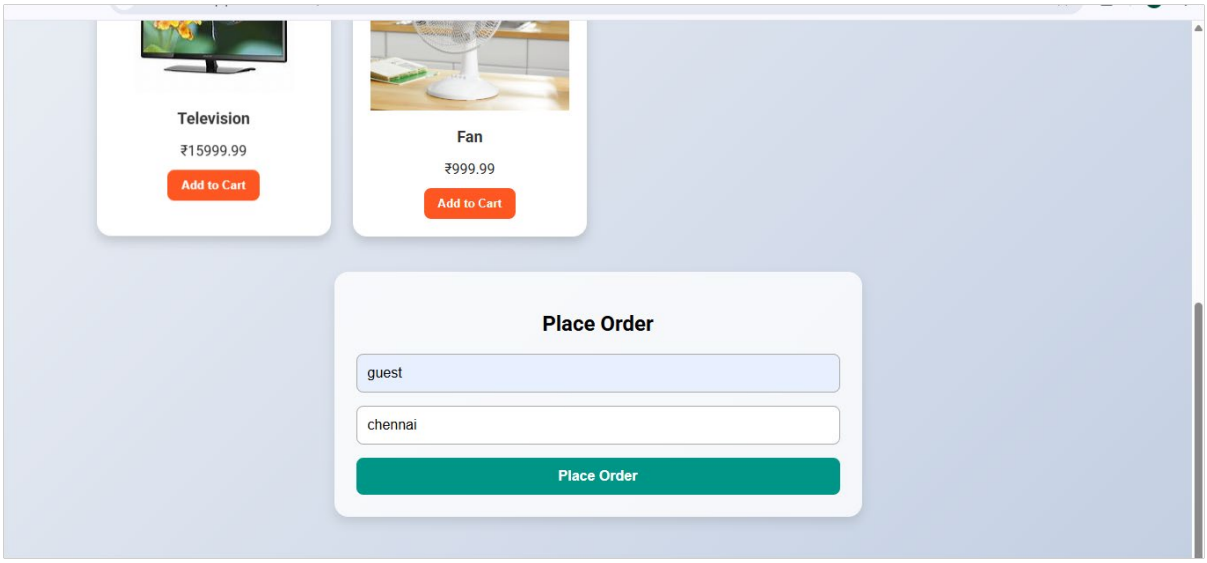
Step 14: Now add the products to cart and check the products are getting added in DynamoDB and also message will get poped-up once the product is added to cart

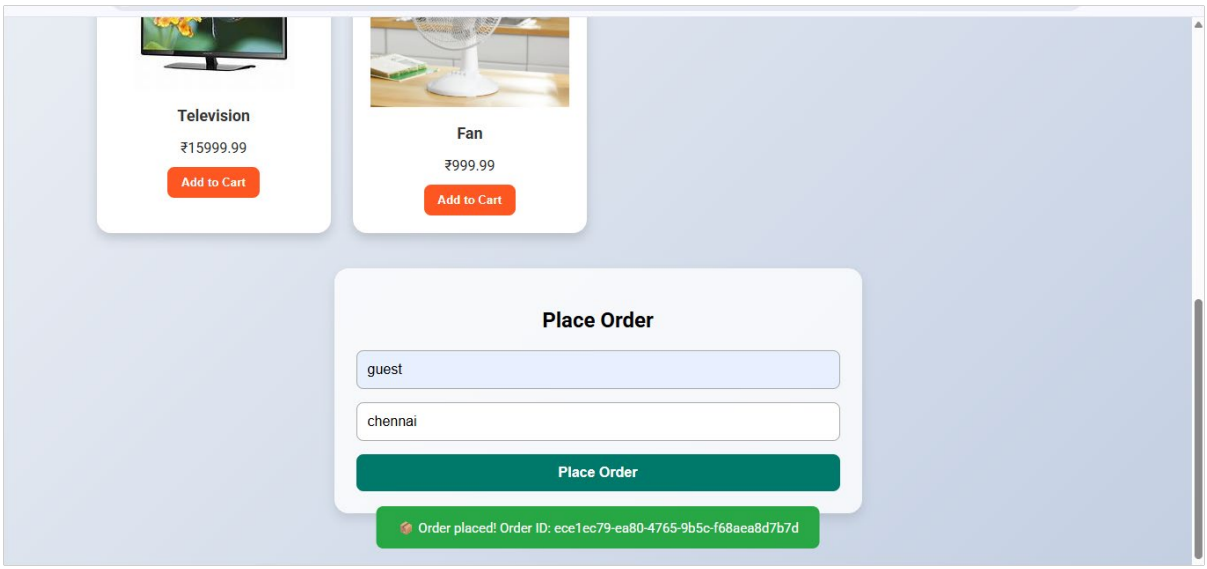


Step 15: And the products are added in DynamoDB cart table



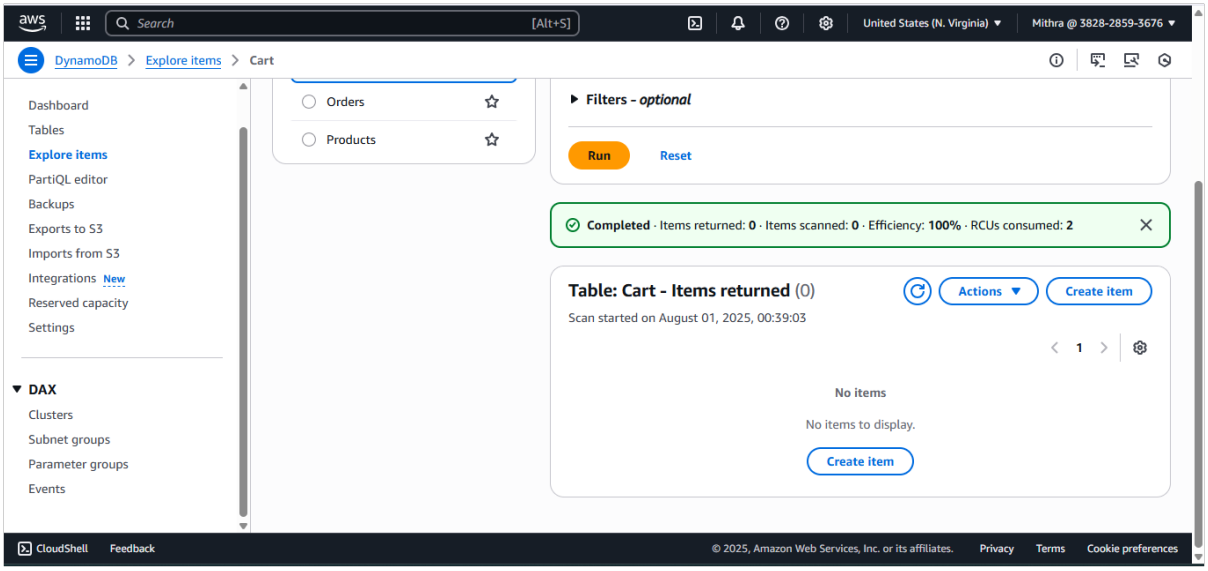
Step 16: Now place the order using the user_id: guest and Address as “Chennai” and once the order is placed successfully the message will be poped-up



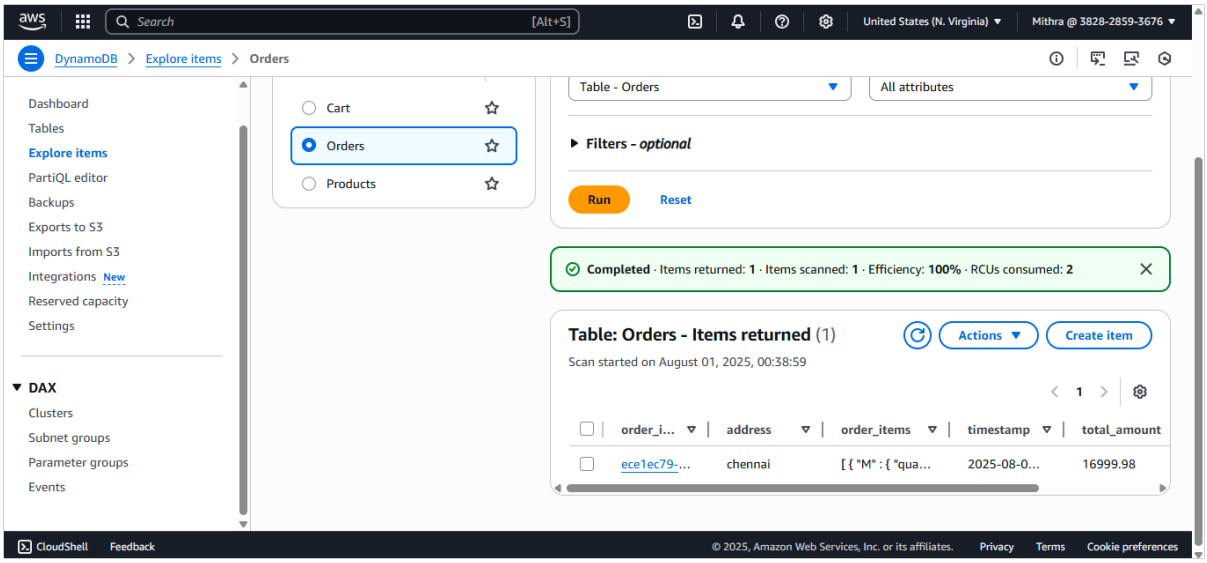


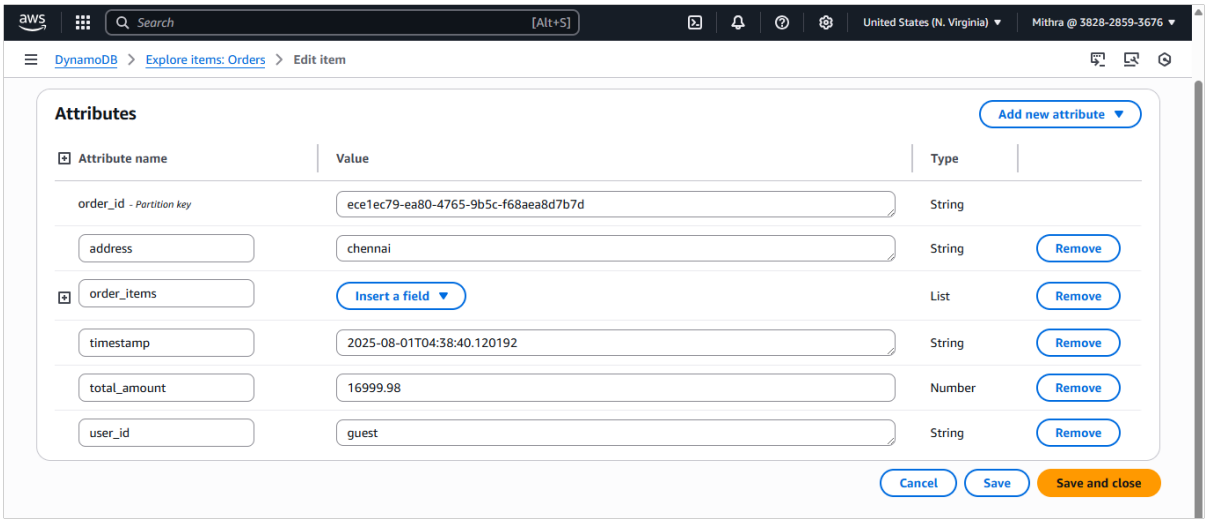
Output:

Step 17: Once the order is placed, we can see the cart becomes empty

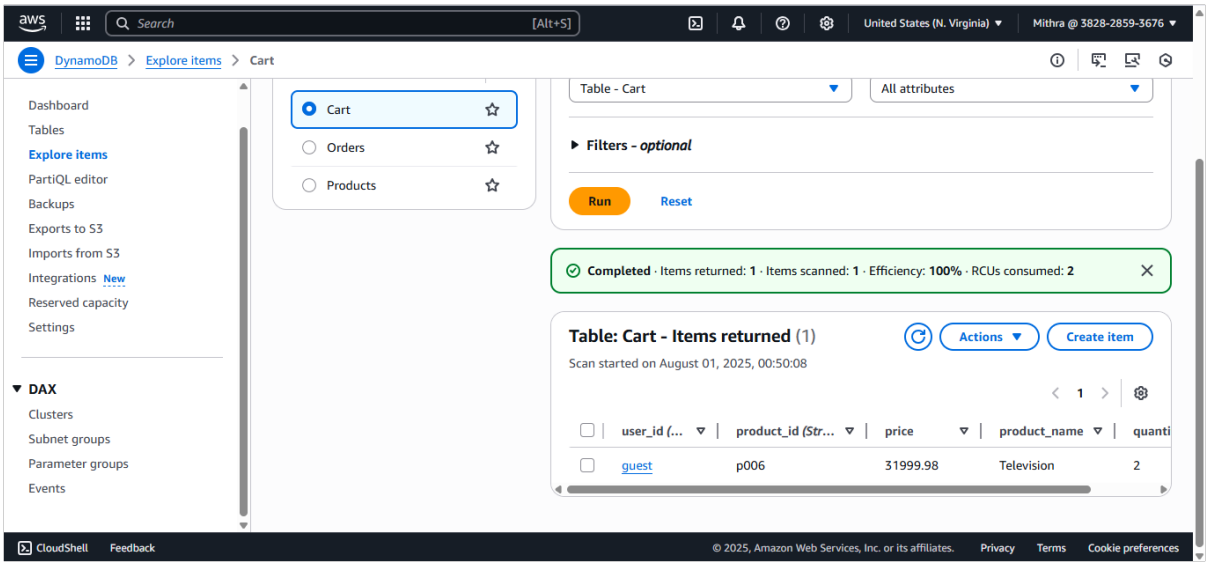


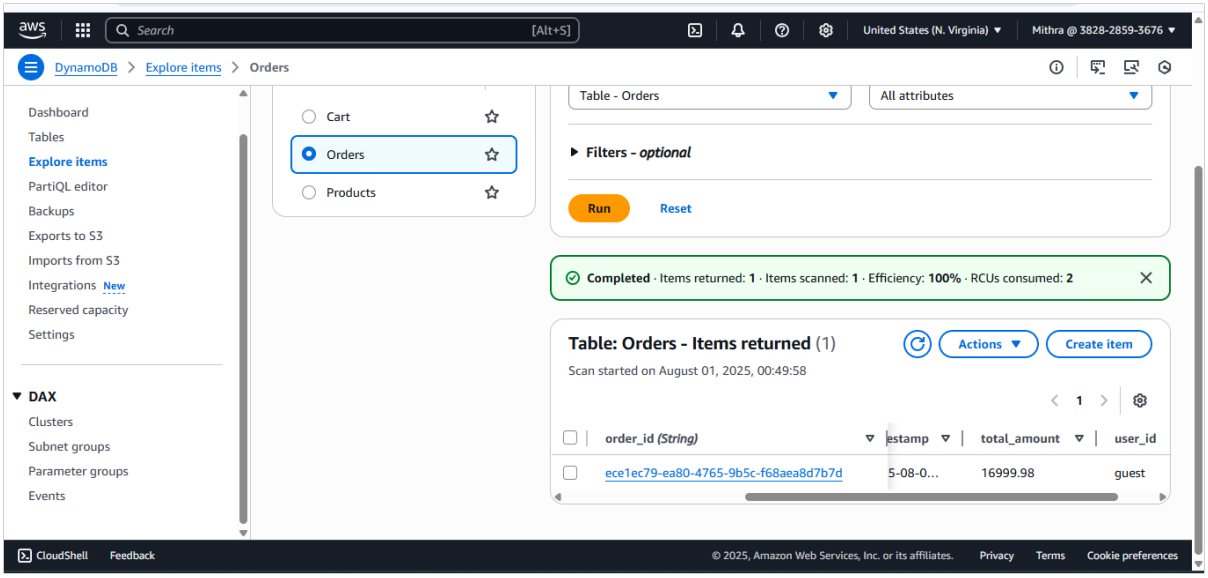
Step 18: Then the orders are placed successfully is added in Orders DynamoDB table



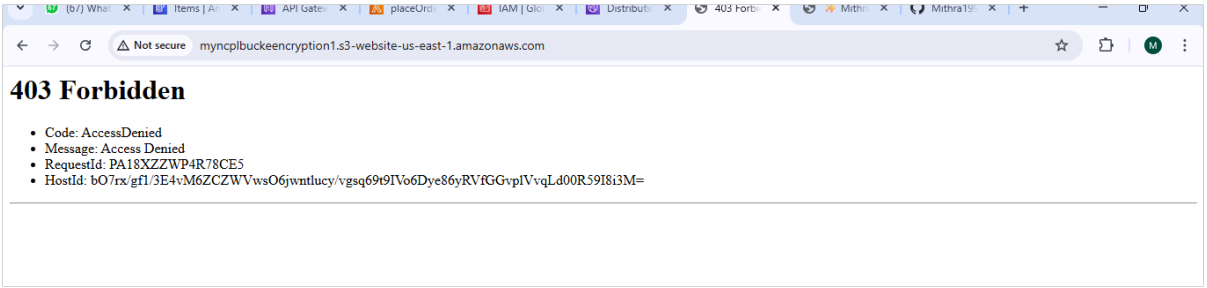


Step 19: If we add the same product twice the product quantity will get increased and price also will be updated

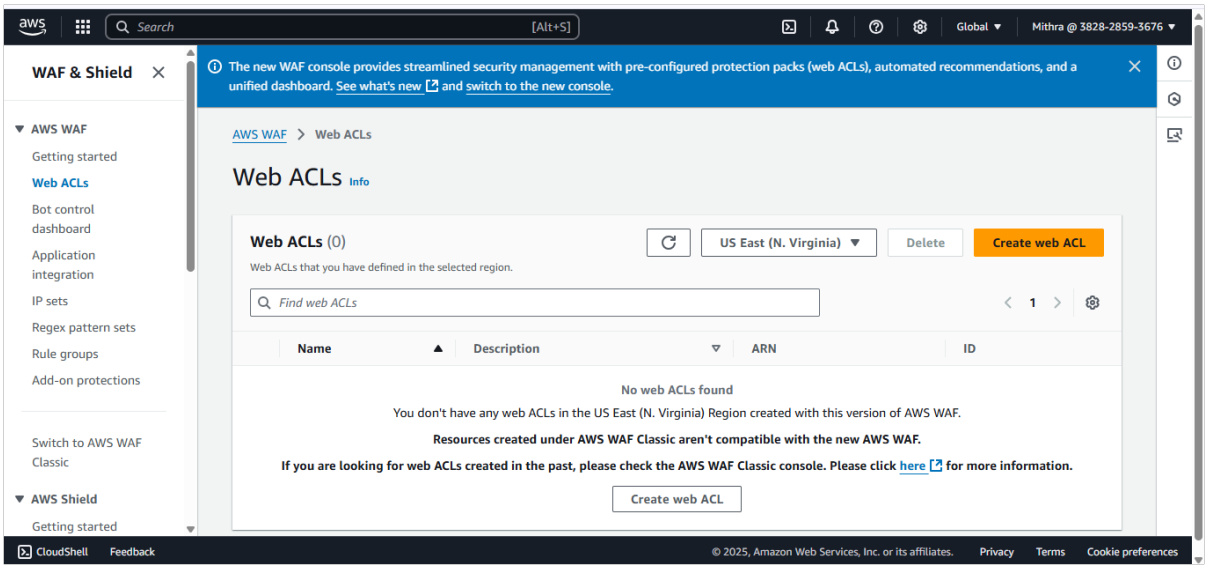


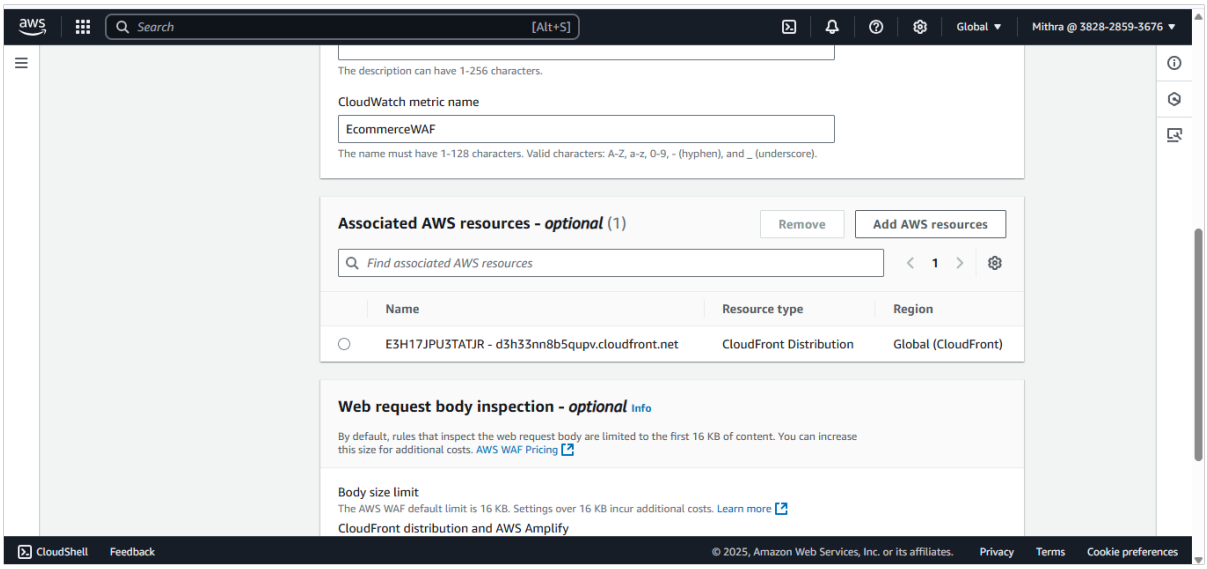


Step 20: Try accessing the S3 static website hosting URL it will show as “Access denied” since the bucket policy is updated to CloudFront , so the website will only be accessible through CloudFront URL

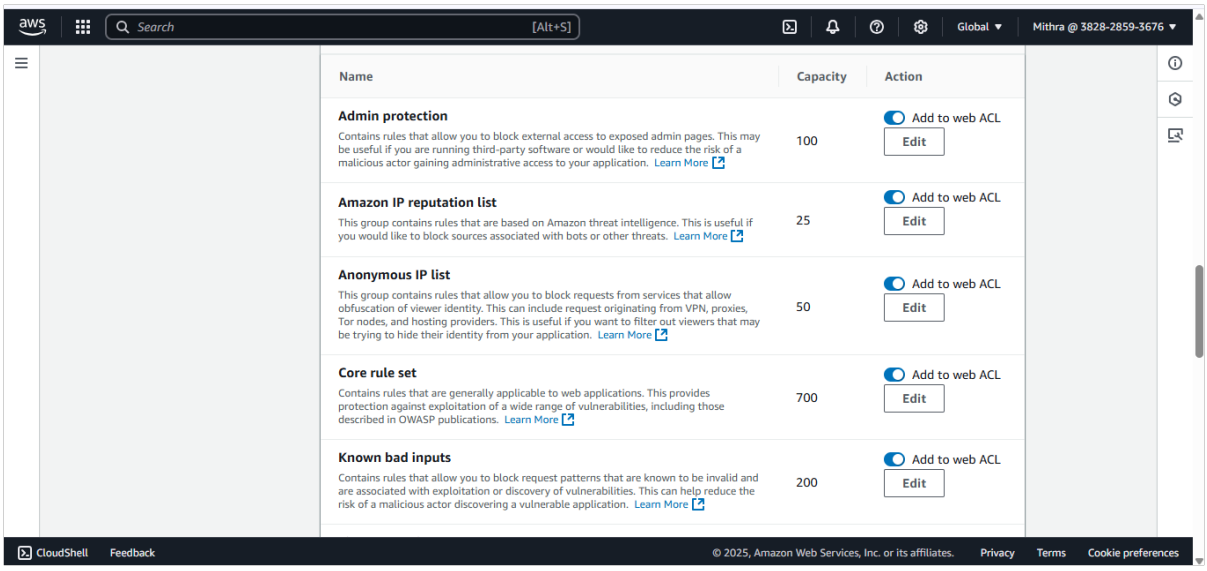


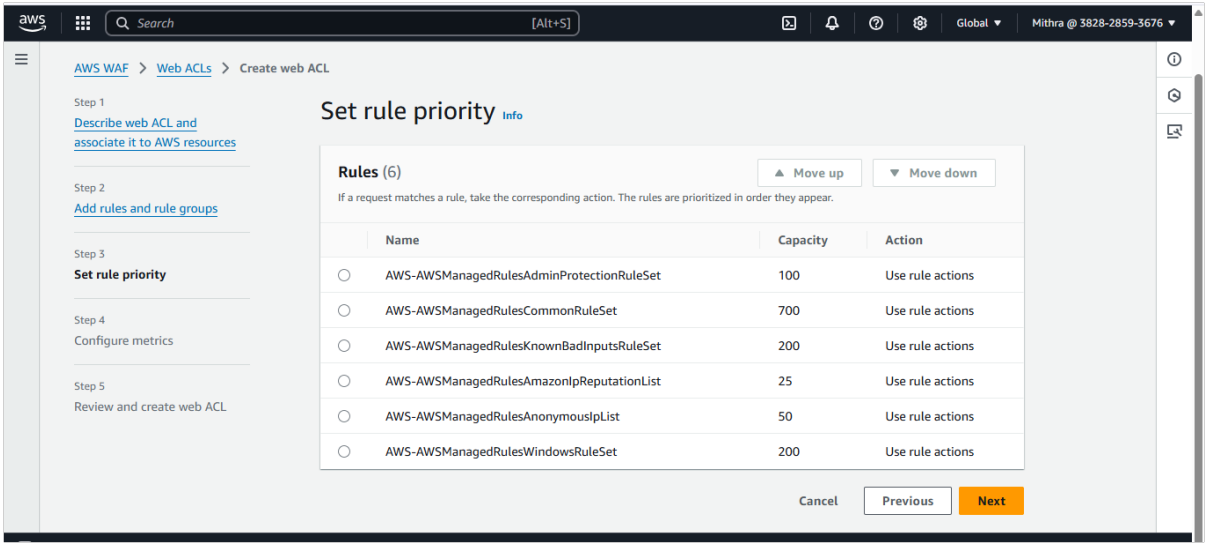
Step 21: Now add the WAF config rules for CloudFront(Global) and API endppint(Regional)

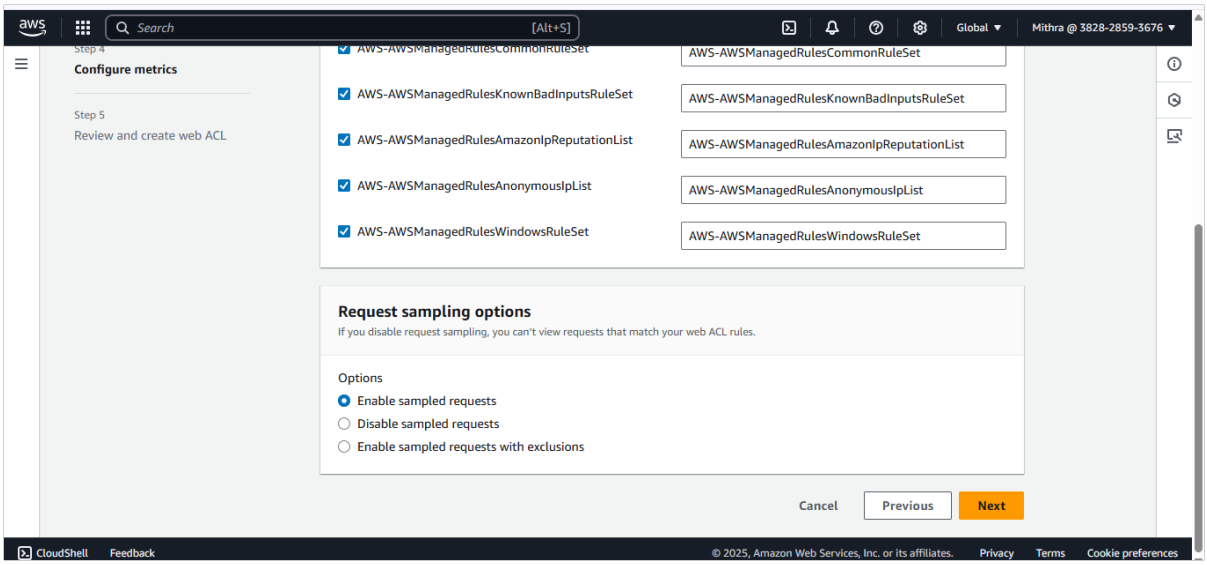


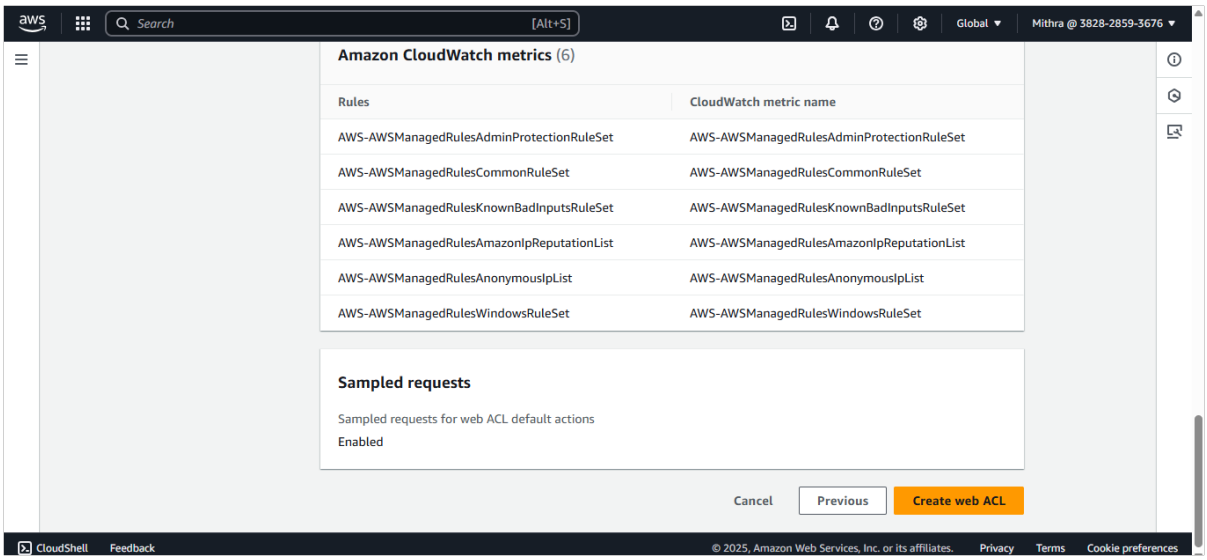


Step 22: Enable free AWS rules for CloudFront

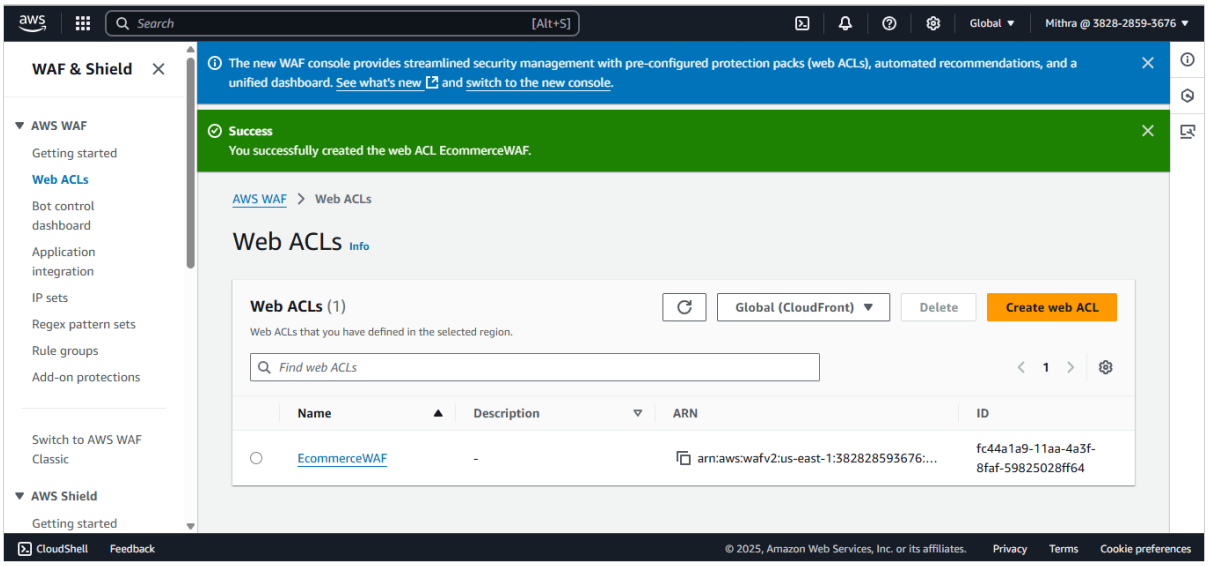




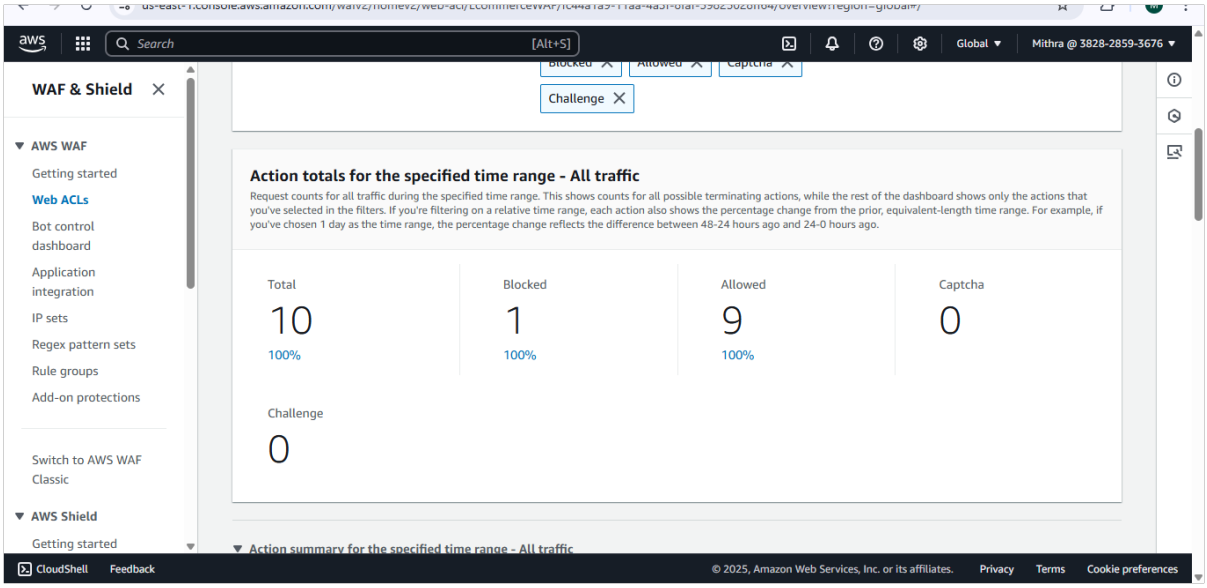

Search
[Alt+S]
Global
Mithra @ 3828-2859-3676
Step 4
Configure metrics
Step 5
Review and create web ACL
AWS-AWSManagedRulesCommonRuleSet
AWS-AWSManagedRulesKnownBadInputsRuleSet
AWS-AWSManagedRulesAmazonIpReputationList
AWS-AWSManagedRulesAnonymousIpList
AWS-AWSManagedRulesWindowsRuleSet
Request sampling 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
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

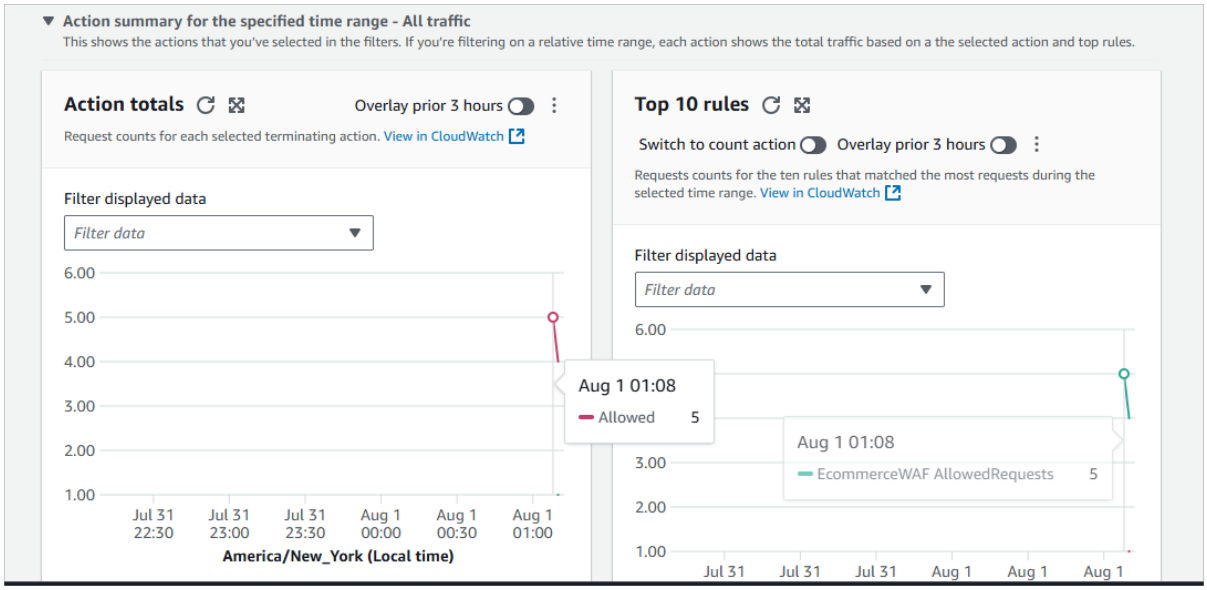

Search
[Alt+S]
Global
Mithra @ 3828-2859-3676
Amazon CloudWatch metrics (6)
Rules
CloudWatch metric name
AWS-AWSManagedRulesAdminProtectionRuleSet
AWS-AWSManagedRulesCommonRuleSet
AWS-AWSManagedRulesKnownBadInputsRuleSet
AWS-AWSManagedRulesAmazonIpReputationList
AWS-AWSManagedRulesAnonymousIpList
AWS-AWSManagedRulesWindowsRuleSet
Sampled requests
Sampled requests for web ACL default actions
Enabled
Cancel
Previous
Create web ACL
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

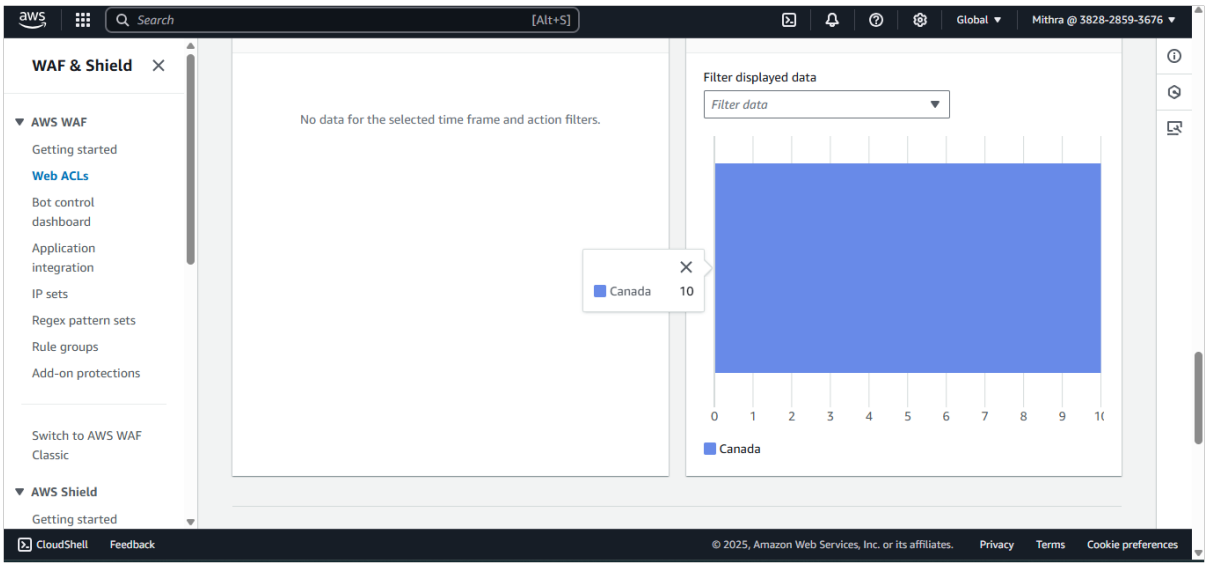
Step 23 : Now the WAF is enabled for CloudFront



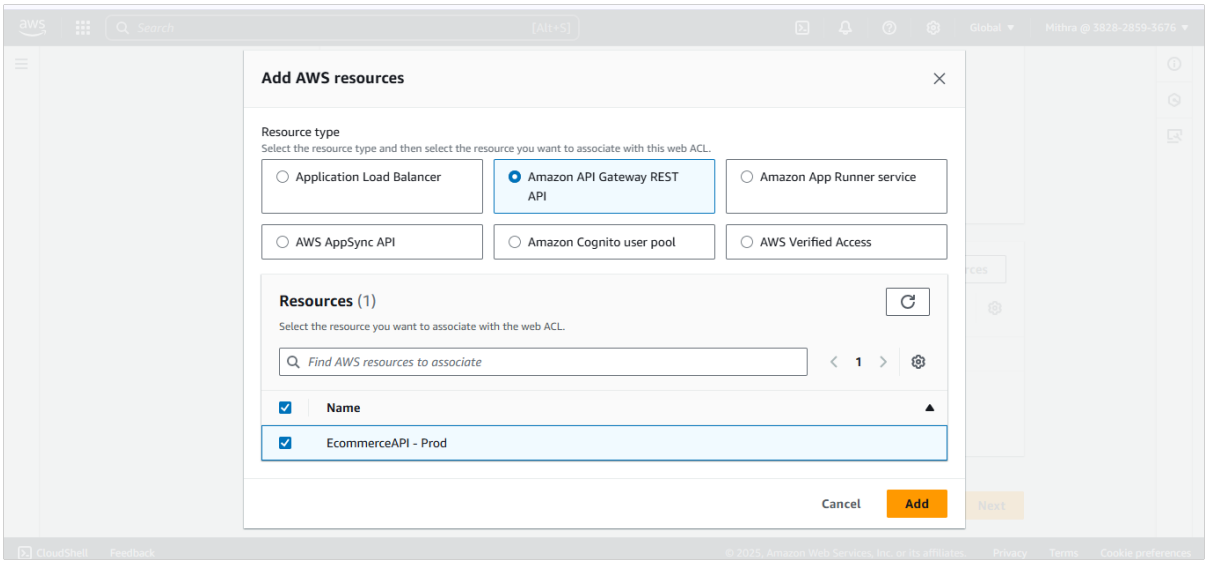
Step 24 : Now we can see the traffic displayed for CloudFront

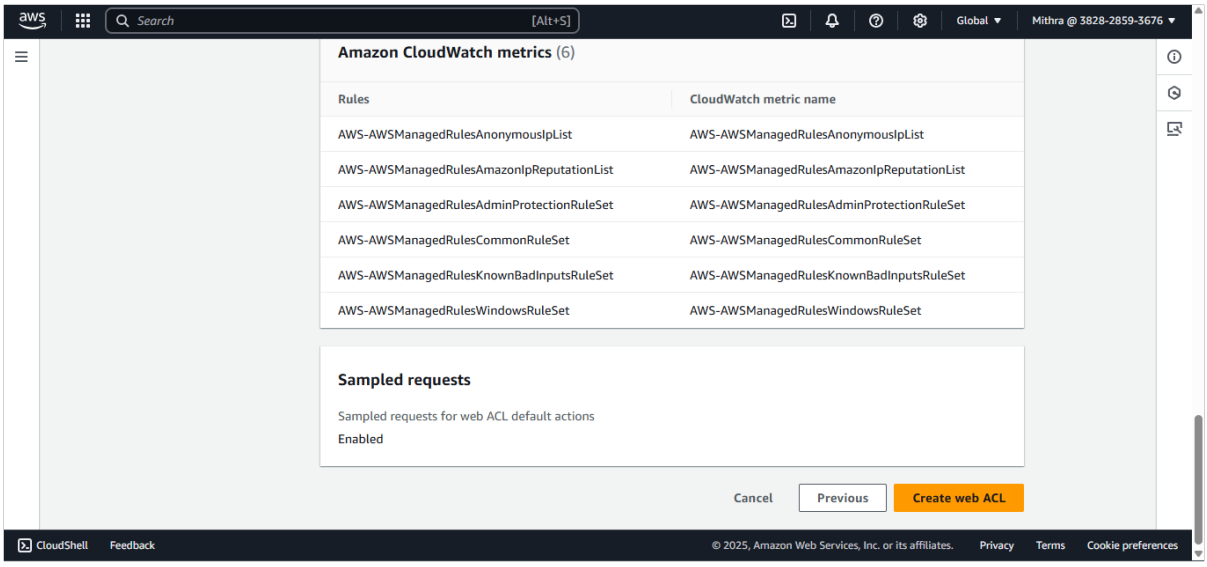




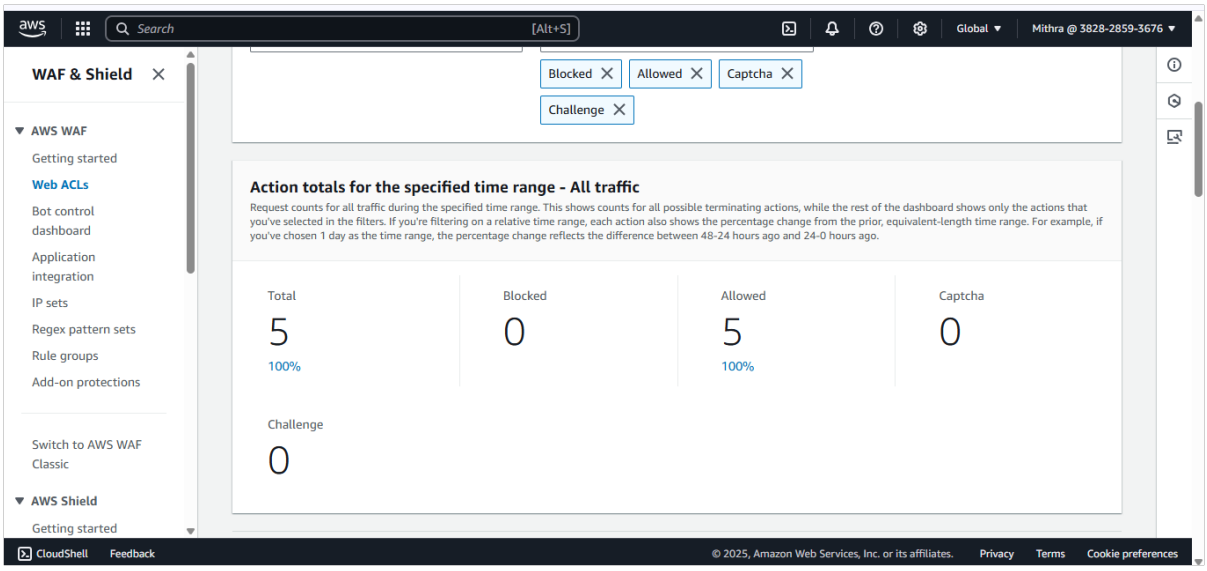


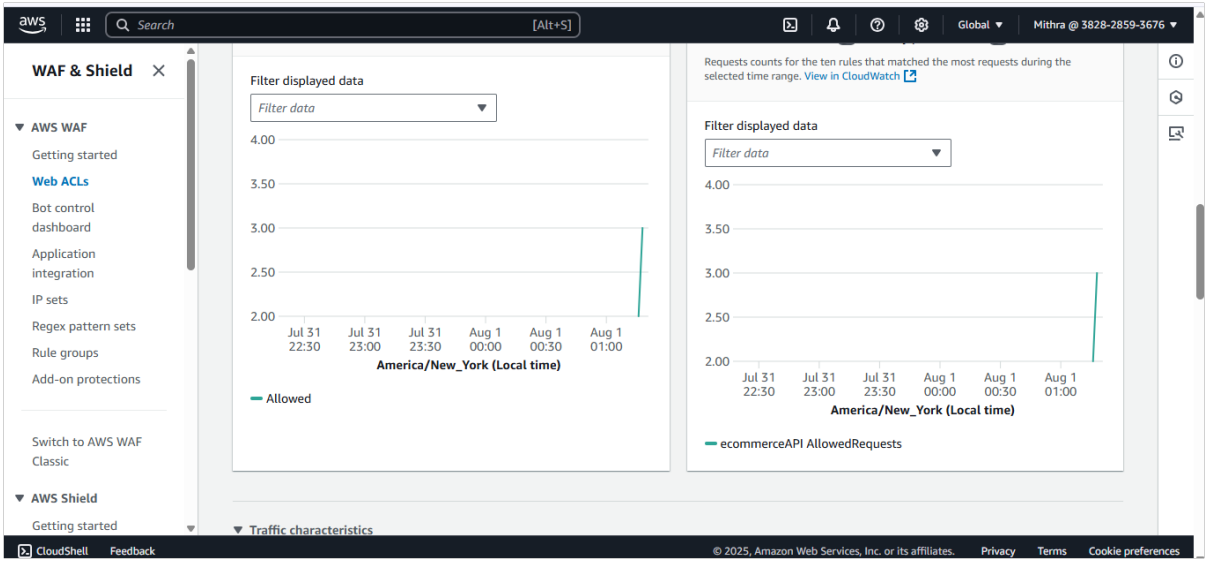
Step 25: Add WAF for API Endpoint

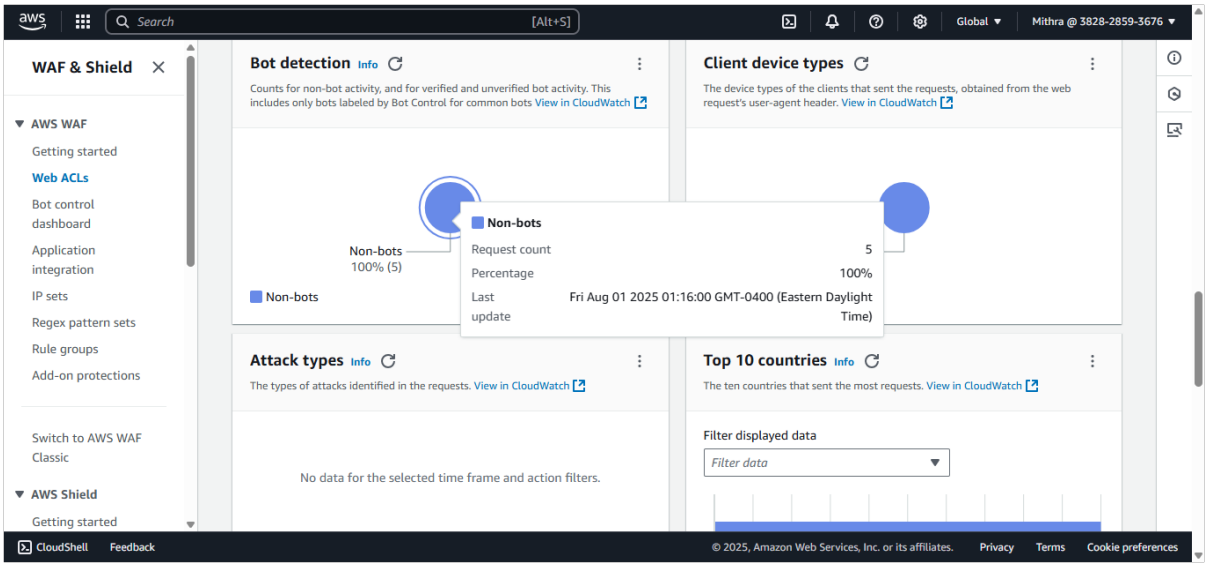

Add AWS resources
Resource type
Select the resource type and then select the resource you want to associate with this web ACL.
Application Load Balancer
Amazon API Gateway REST API
Amazon App Runner service
AWS AppSync API
Amazon Cognito user pool
AWS Verified Access
Resources (1)
Select the resource you want to associate with the web ACL.
Find AWS resources to associate
Name
EcommerceAPI - Prod
Cancel
Add

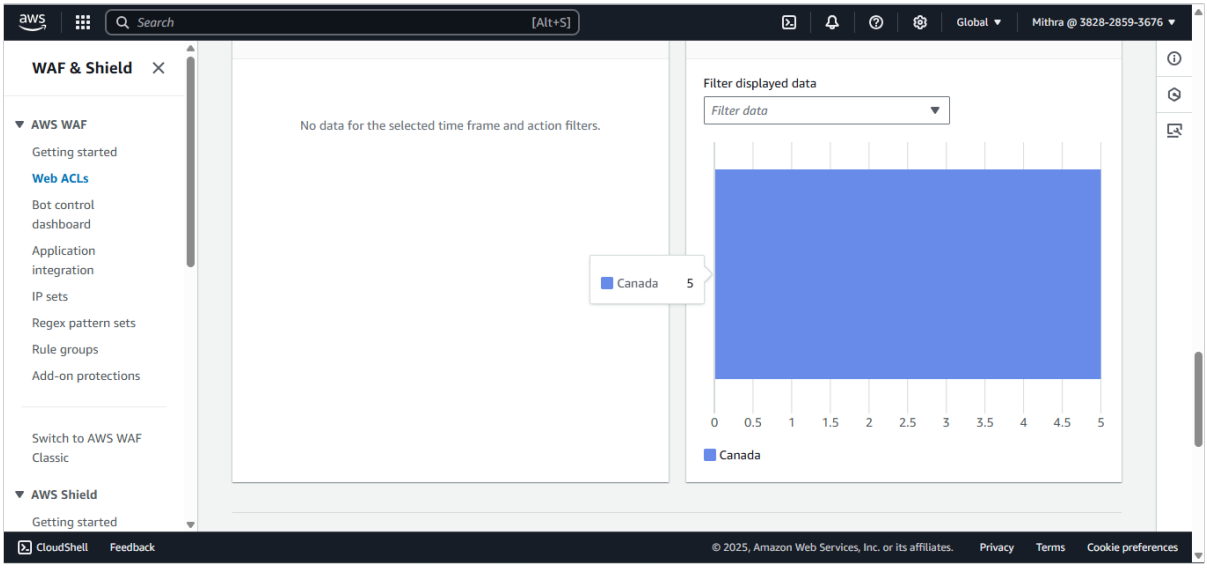

Search
[Alt+S]
Global
Mithra @ 3828-2859-3676
Amazon CloudWatch metrics (6)
Rules
CloudWatch metric name
AWS-AWSManagedRulesAnonymousIpList
AWS-AWSManagedRulesAmazonIpReputationList
AWS-AWSManagedRulesAdminProtectionRuleSet
AWS-AWSManagedRulesCommonRuleSet
AWS-AWSManagedRulesKnownBadInputsRuleSet
AWS-AWSManagedRulesWindowsRuleSet
Sampled requests
Sampled requests for web ACL default actions
Enabled
Cancel
Previous
Create web ACL
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

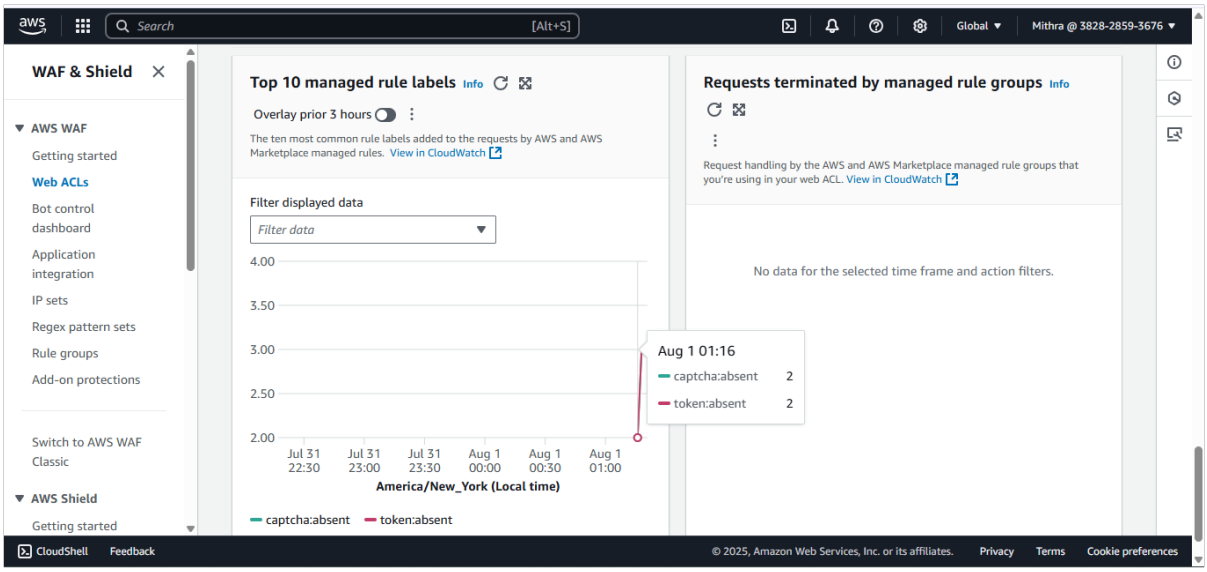
Step 26: Now we can see the traffic us displayed for API

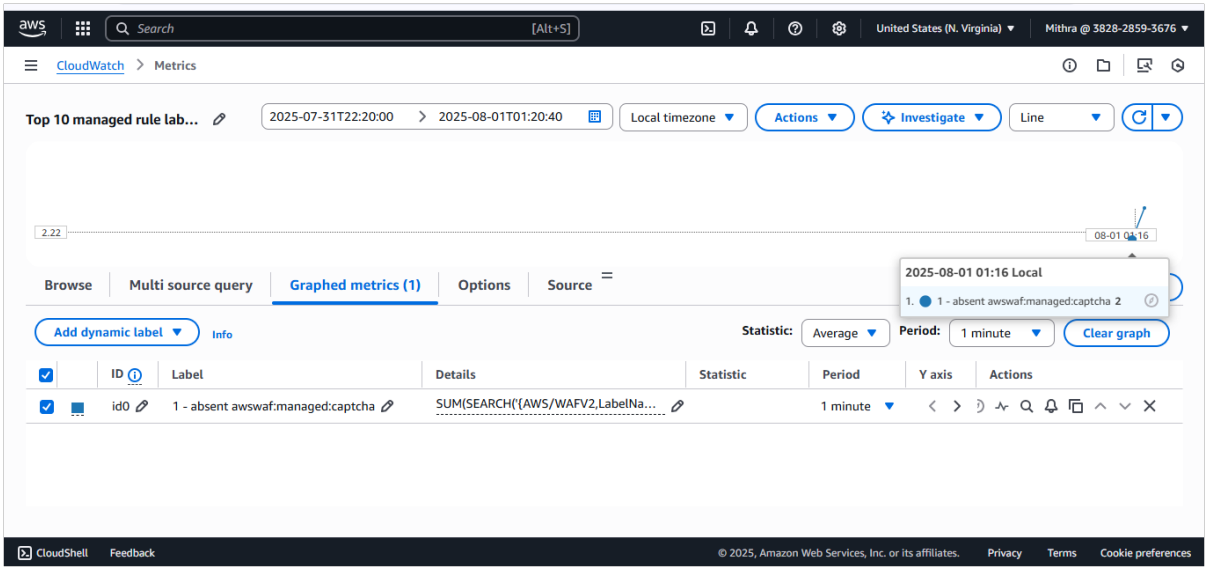


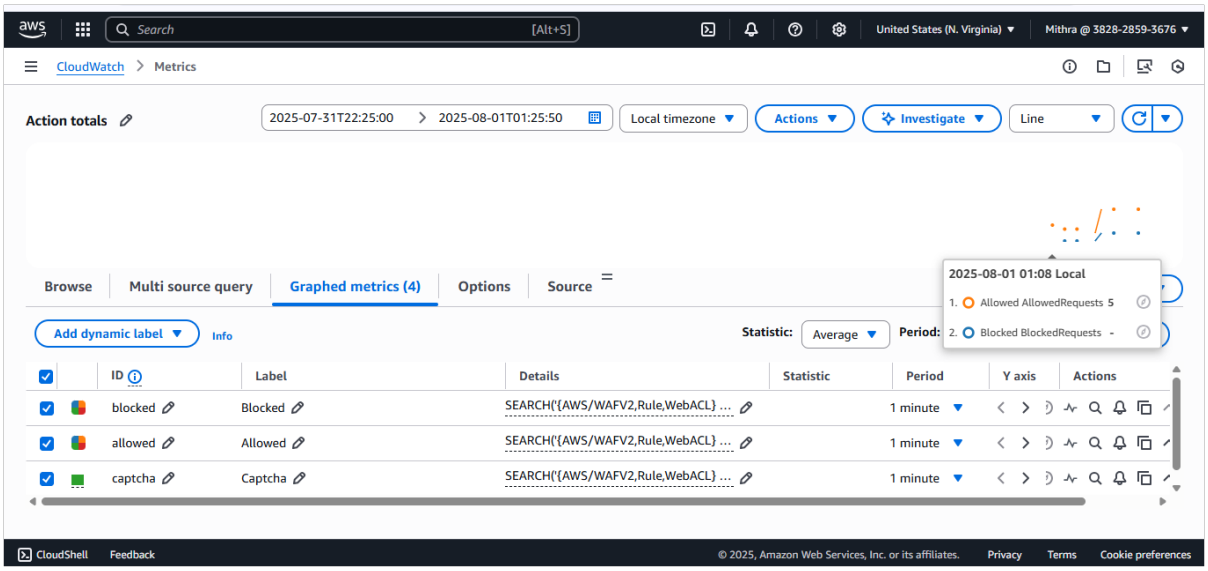


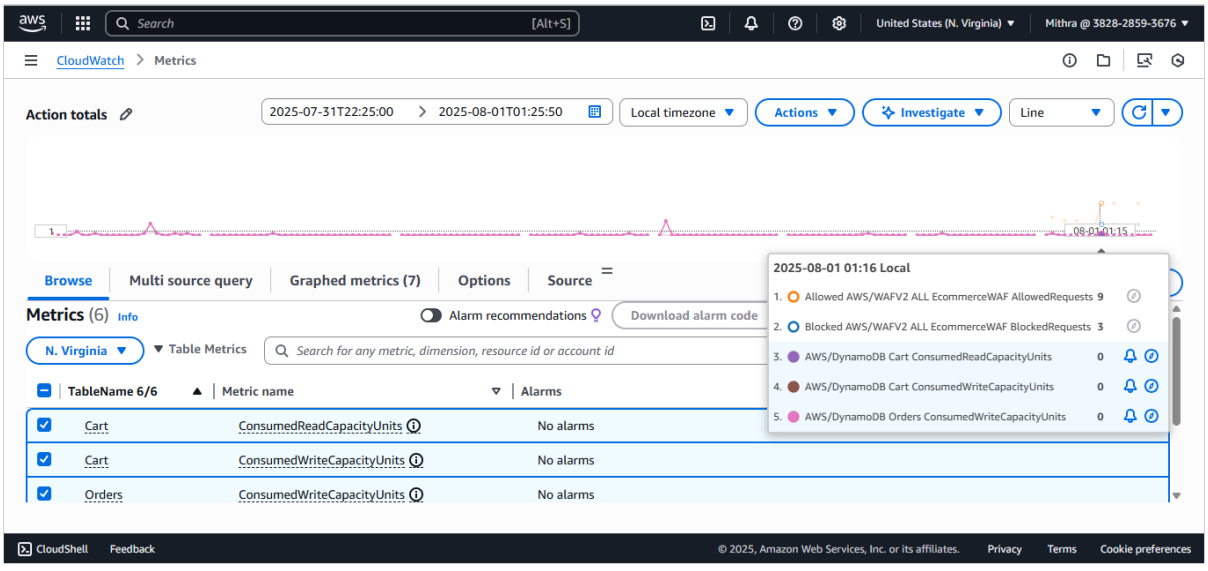

Search
[Alt+S]
Global
Mithra @ 3828-2859-3676
WAF & Shield
AWS WAF
Getting started
Web ACLs
Bot control dashboard
Application integration
IP sets
Regex pattern sets
Rule groups
Add-on protections
Switch to AWS WAF Classic
AWS Shield
Getting started
Bot detection Info
Counts for non-bot activity, and for verified and unverified bot activity. This includes only bots labeled by Bot Control for common bots View in CloudWatch
Client device types
The device types of the clients that sent the requests, obtained from the web request's user-agent header. View in CloudWatch
Non-bots
100% (5)
Non-bots
Request count 5
Percentage 100%
Last update Fri Aug 01 2025 01:16:00 GMT-0400 (Eastern Daylight Time)
Attack types Info
The types of attacks identified in the requests. View in CloudWatch
No data for the selected time frame and action filters.
Top 10 countries Info
The ten countries that sent the most requests. View in CloudWatch
Filter displayed data
Filter data
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

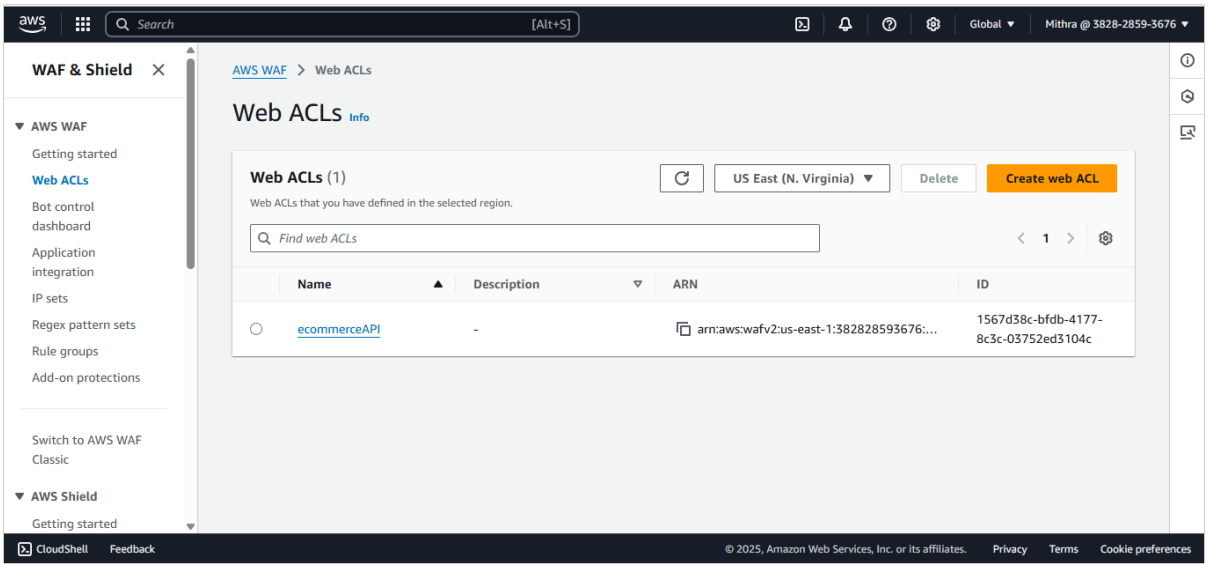

Search
[Alt+S]
Global
Mithra @ 3828-2859-3676
WAF & Shield
AWS WAF
Getting started
Web ACLs
Bot control dashboard
Application integration
IP sets
Regex pattern sets
Rule groups
Add-on protections
Switch to AWS WAF Classic
AWS Shield
Getting started
No data for the selected time frame and action filters.
Filter displayed data
Filter data
Canada 5
0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5
Canada
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Search
[Alt+S]
Global
Mithra @ 3828-2859-3676
WAF & Shield
AWS WAF
Getting started
Web ACLs
Bot control dashboard
Application integration
IP sets
Regex pattern sets
Rule groups
Add-on protections
Switch to AWS WAF Classic
AWS Shield
Getting started
Top 10 managed rule labels Info
Overlay prior 3 hours
The ten most common rule labels added to the requests by AWS and AWS Marketplace managed rules. View in CloudWatch
Filter displayed data
Filter data
4.00
3.50
3.00
2.50
2.00
Jul 31 22:30
Jul 31 23:00
Jul 31 23:30
Aug 1 00:00
Aug 1 00:30
Aug 1 01:00
America/New_York (Local time)
Aug 1 01:16
captcha:absent 2
token:absent 2
captcha:absent
token:absent
Requests terminated by managed rule groups Info
Request handling by the AWS and AWS Marketplace managed rule groups that you're using in your web ACL. View in CloudWatch
No data for the selected time frame and action filters.
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Search
[Alt+S]
United States (N. Virginia)
Mithra @ 3828-2859-3676
CloudWatch
Metrics
Top 10 managed rule lab...
2025-07-31T22:20:00
2025-08-01T01:20:40
Local timezone
Actions
Investigate
Line
2.22
08-01 01:16
2025-08-01 01:16 Local
1. 1 - absent awswaf:managed:captcha 2
Browse
Multi source query
Graphed metrics (1)
Options
Source
Add dynamic label
Info
Statistic:
Average
Period:
1 minute
Clear graph
ID
Label
Details
Statistic
Period
Y axis
Actions
id0
1 - absent awswaf:managed:captcha
SUM(SEARCH('{AWS/WAFV2,LabelNa...
1 minute
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Search [Alt+S]
United States (N. Virginia)
Mithra @ 3828-2859-3676
CloudWatch > Metrics
Action totals
2025-07-31T22:25:00 > 2025-08-01T01:25:50
Local timezone
Actions
Investigate
Line
2025-08-01 01:08 Local
1. Allowed AllowedRequests 5
2. Blocked BlockedRequests -
Browse
Multi source query
Graphed metrics (4)
Options
Source
Add dynamic label
Info
Statistic:
Average
Period:
ID
Label
Details
Statistic
Period
Y axis
Actions
blocked
Blocked
SEARCH('{AWS/WAFV2,Rule,WebACL} ...
1 minute
allowed
Allowed
SEARCH('{AWS/WAFV2,Rule,WebACL} ...
1 minute
captcha
Captcha
SEARCH('{AWS/WAFV2,Rule,WebACL} ...
1 minute
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Search [Alt+S]
United States (N. Virginia)
Mithra @ 3828-2859-3676
CloudWatch > Metrics
Action totals
2025-07-31T22:25:00 > 2025-08-01T01:25:50
Local timezone
Actions
Investigate
Line
2025-08-01 01:16 Local
1. Allowed AWS/WAFV2 ALL EcommerceWAF AllowedRequests 9
2. Blocked AWS/WAFV2 ALL EcommerceWAF BlockedRequests 3
3. AWS/DynamoDB Cart ConsumedReadCapacityUnits 0
4. AWS/DynamoDB Cart ConsumedWriteCapacityUnits 0
5. AWS/DynamoDB Orders ConsumedWriteCapacityUnits 0
Browse
Multi source query
Graphed metrics (7)
Options
Source
Metrics (6)
Info
Alarm recommendations
Download alarm code
N. Virginia
Table Metrics
Search for any metric, dimension, resource id or account id
TableName 6/6
Metric name
Alarms
Cart
ConsumedReadCapacityUnits
No alarms
Cart
ConsumedWriteCapacityUnits
No alarms
Orders
ConsumedWriteCapacityUnits
No alarms
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Search
[Alt+S]
Global
Mithra @ 3828-2859-3676
WAF & Shield
AWS WAF
Getting started
Web ACLs
Bot control dashboard
Application integration
IP sets
Regex pattern sets
Rule groups
Add-on protections
Switch to AWS WAF Classic
AWS Shield
Getting started
AWS WAF > Web ACLs
Web ACLs Info
Web ACLs (1)
Web ACLs that you have defined in the selected region.
US East (N. Virginia)
Delete
Create web ACL
Find web ACLs
1
Name
Description
ARN
ID
ecommerceAPI
-
arn:aws:wafv2:us-east-1:382828593676:...
1567d38c-bfdb-4177-8c3c-03752ed3104c
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences