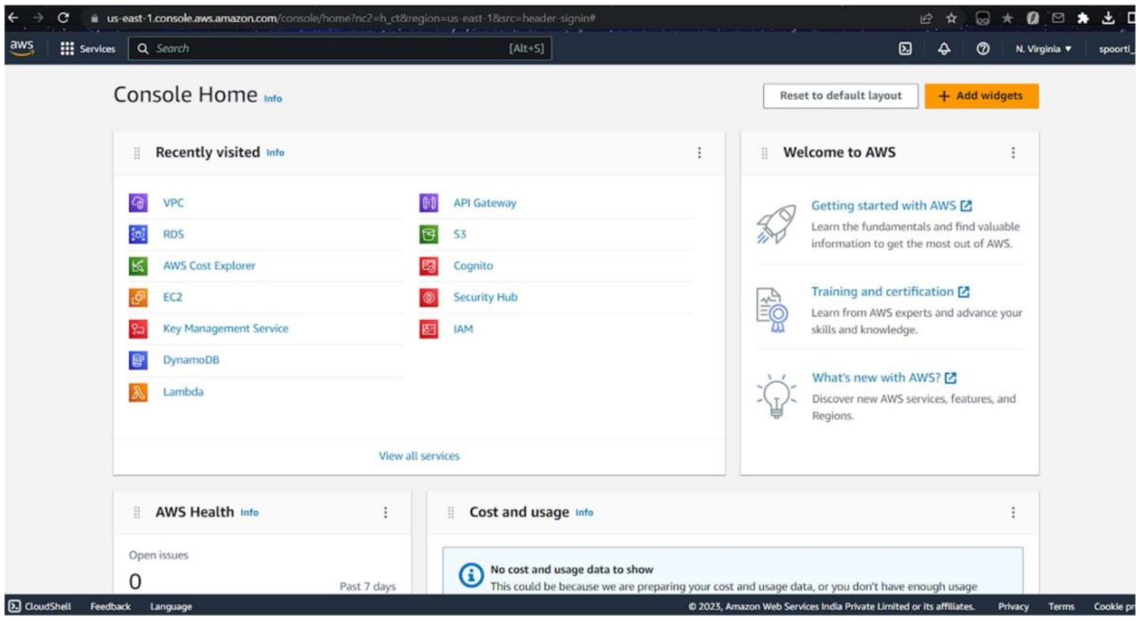
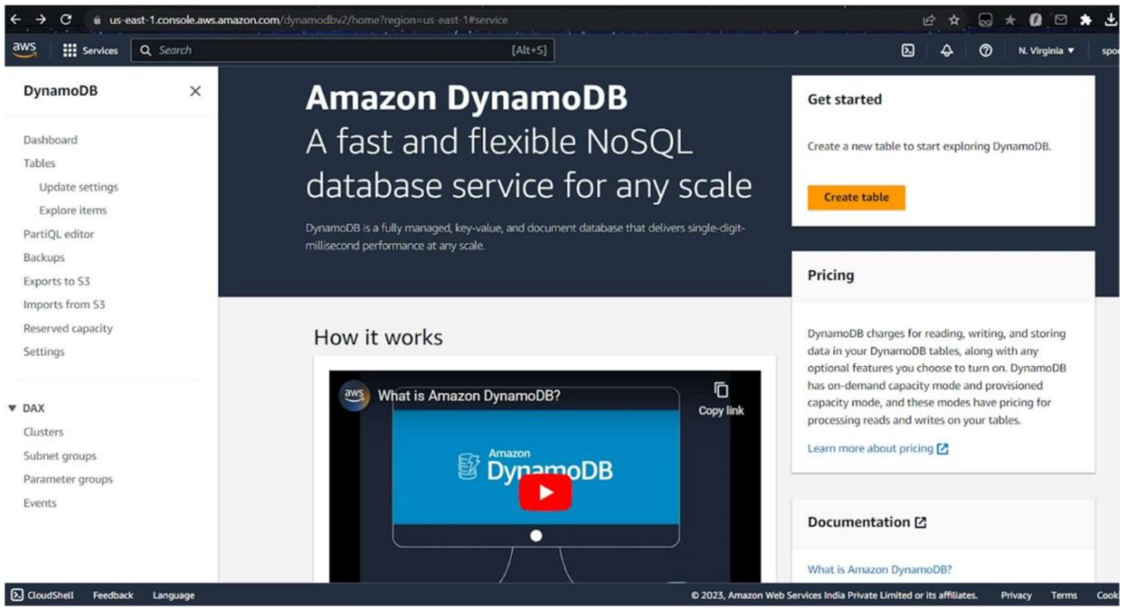
Snapshots

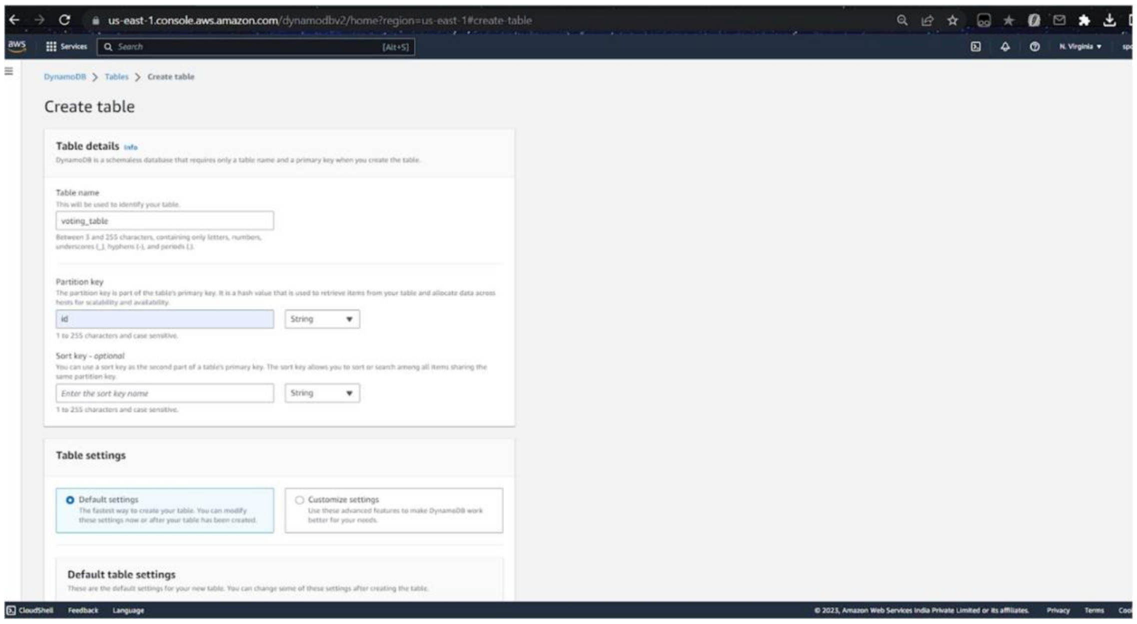
a) Open AWS console.



b) Navigate to DynamoDB and click on ‘Create table’.



c) Set table name as ‘voting table’ and Partition key (primary key) as ‘id’. Leave the other configurations as default.


us-east-1.console.aws.amazon.com/dynamodbv2/home?region=us-east-1#create-table

DynamoDB > Tables > Create table

Create table

Table details Info

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

Table name

This will be used to identify your table.

voting_table

Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.).

Partition key

The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.

id — String

1 to 255 characters and case sensitive.

Sort key - *optional*

You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.

Enter the sort key name — String

1 to 255 characters and case sensitive.

Table settings

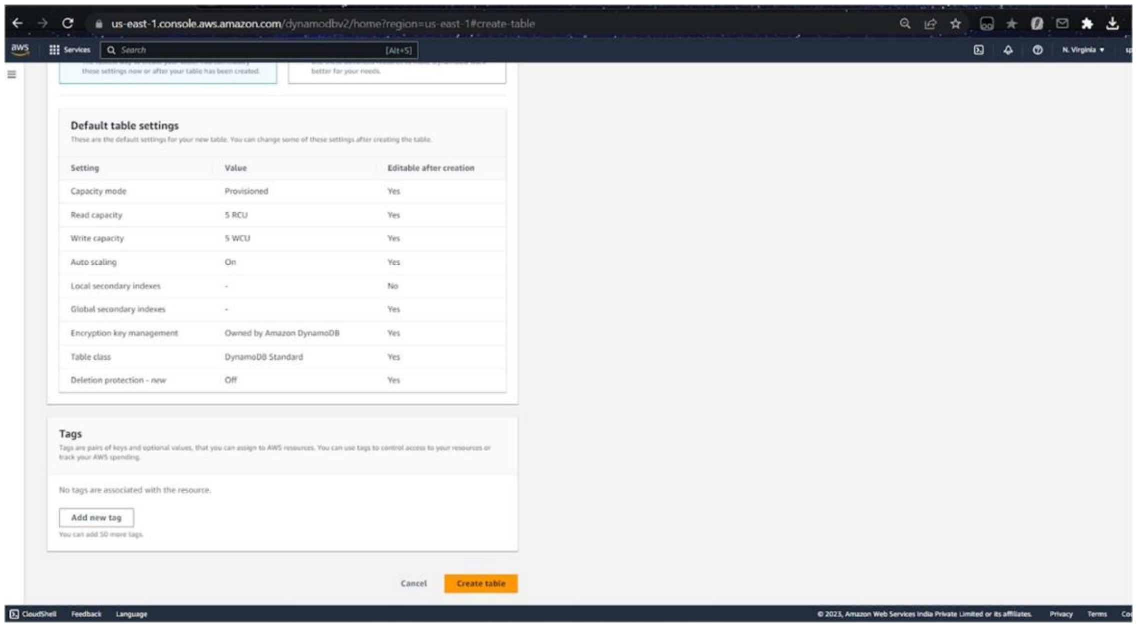
[✓ Default settings] The fastest way to create your table. You can modify these settings now or after your table has been created.

Customize settings: Use these advanced features to make DynamoDB work better for your needs.

Default table settings

These are the default settings for your new table. You can change some of these settings after creating the table.

CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms


us-east-1.console.aws.amazon.com/dynamodbv2/home?region=us-east-1#create-table

these settings now or after your table has been created. — better for your needs.

Default table settings

These are the default settings for your new table. You can change some of these settings after creating the table.

Setting	Value	Editable after creation
Capacity mode	Provisioned	Yes
Read capacity	5 RCU	Yes
Write capacity	5 WCU	Yes
Auto scaling	On	Yes
Local secondary indexes	-	No
Global secondary indexes	-	Yes
Encryption key management	Owned by Amazon DynamoDB	Yes
Table class	DynamoDB Standard	Yes
Deletion protection - *new*	Off	Yes

Tags

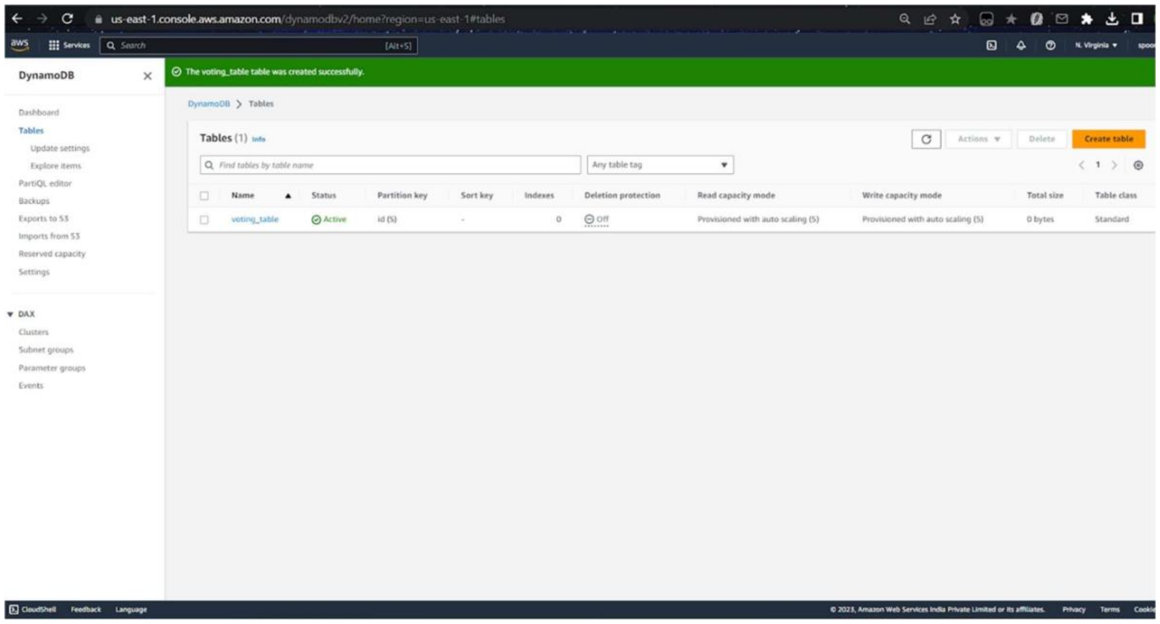
Tags are pairs of keys and optional values, that you can assign to AWS resources. You can use tags to control access to your resources or track your AWS spending.

No tags are associated with the resource.

You can add 50 more tags.

Add new tag
Cancel
Create table
CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms

d) Table has been created. Click on the newly created table and then go to ‘Explore Table items’.

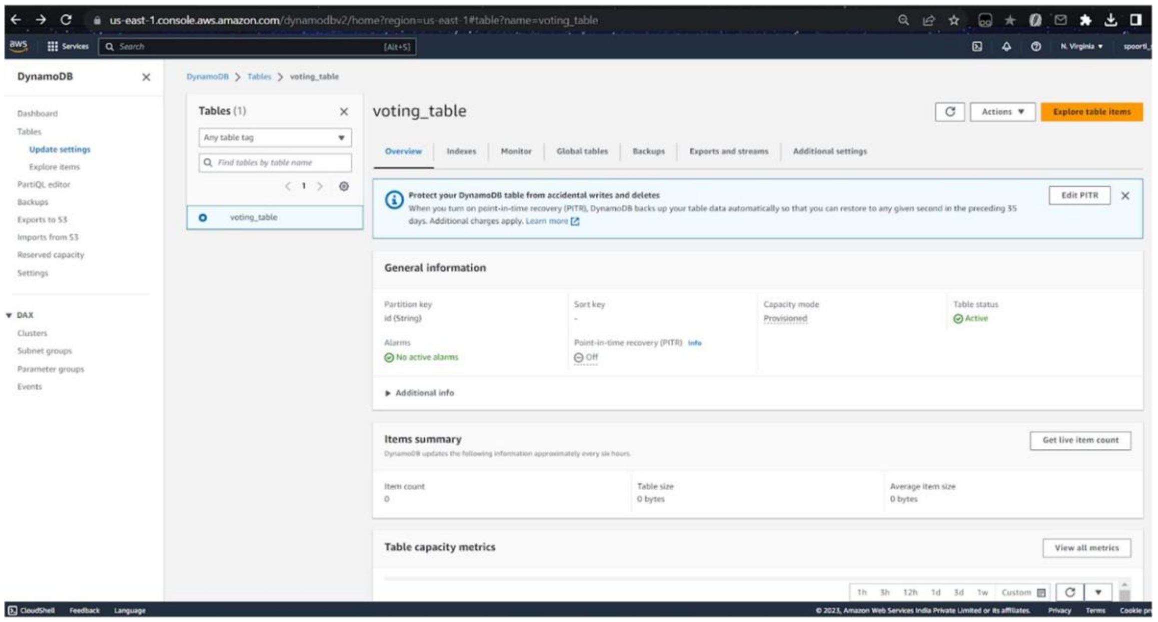


The voting_table table was created successfully.

DynamoDB > Tables

Tables (1)

Name	Status	Partition key	Sort key	Indexes	Deletion protection	Read capacity mode	Write capacity mode	Total size	Table class
voting_table	Active	id (S)	-	0	Off	Provisioned with auto scaling (5)	Provisioned with auto scaling (5)	0 bytes	Standard



voting_table

Protect your DynamoDB table from accidental writes and deletes
When you turn on point-in-time recovery (PITR), DynamoDB backs up your table data automatically so that you can restore to any given second in the preceding 35 days. Additional charges apply. Learn more

General information

Partition key: id (String)	Sort key: -	Capacity mode: Provisioned	Table status: Active
Alarms: No active alarms	Point-in-time recovery (PITR): Off		

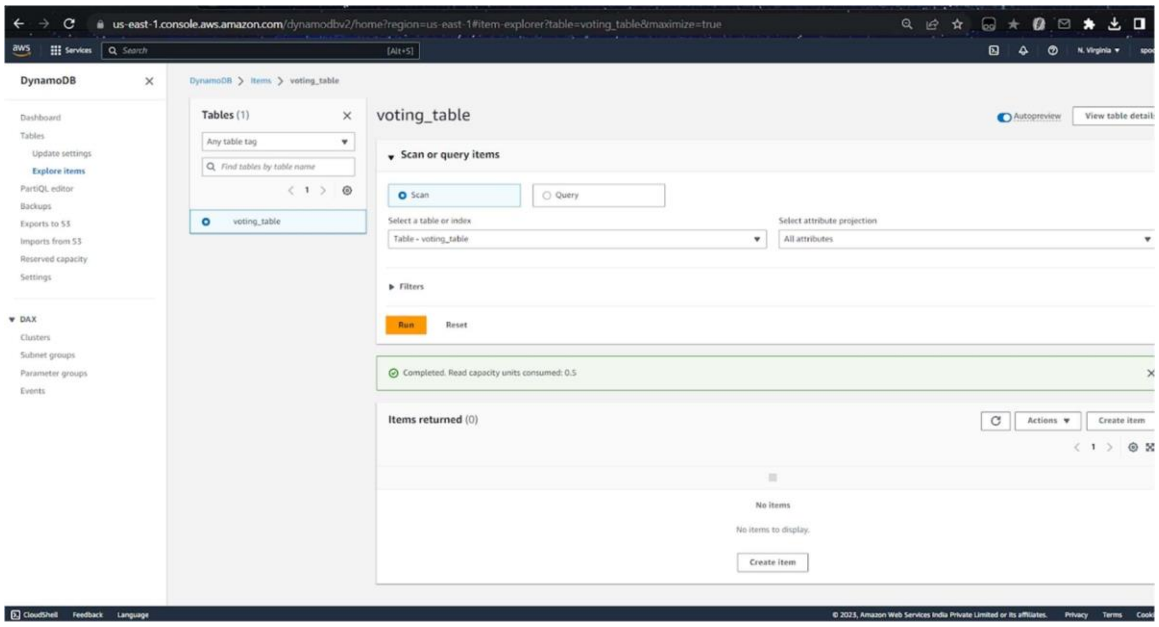
Additional info

Items summary
DynamoDB updates the following information approximately every six hours.

Item count	Table size	Average item size
0	0 bytes	0 bytes

Table capacity metrics

e) Scroll down and click on ‘Create item’.



voting_table

Select a table or index: Table - voting_table

Select attribute projection: All attributes

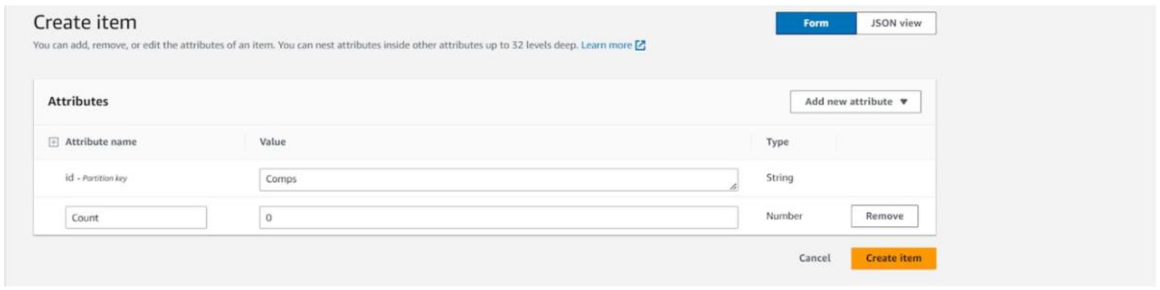
Completed. Read capacity units consumed: 0.5

Items returned (0)

No items

No items to display.

f) Create 3 items (Comps, IT, EXTC) each having ‘id’ and ‘Count’ as attributes.

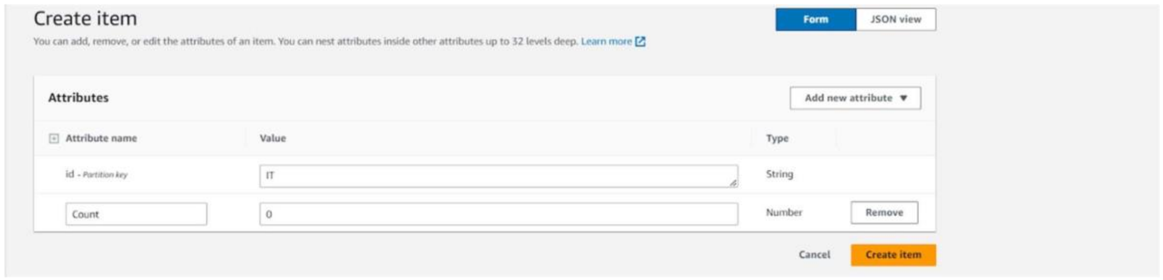


Create item

You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. Learn more

Attributes

Attribute name	Value	Type
id - Partition key	Comps	String
Count	0	Number

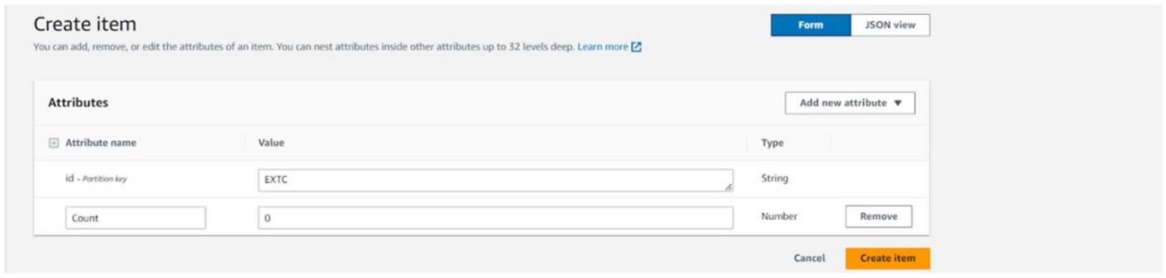


Create item

You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. Learn more

Attributes

Attribute name	Value	Type
id - Partition key	IT	String
Count	0	Number



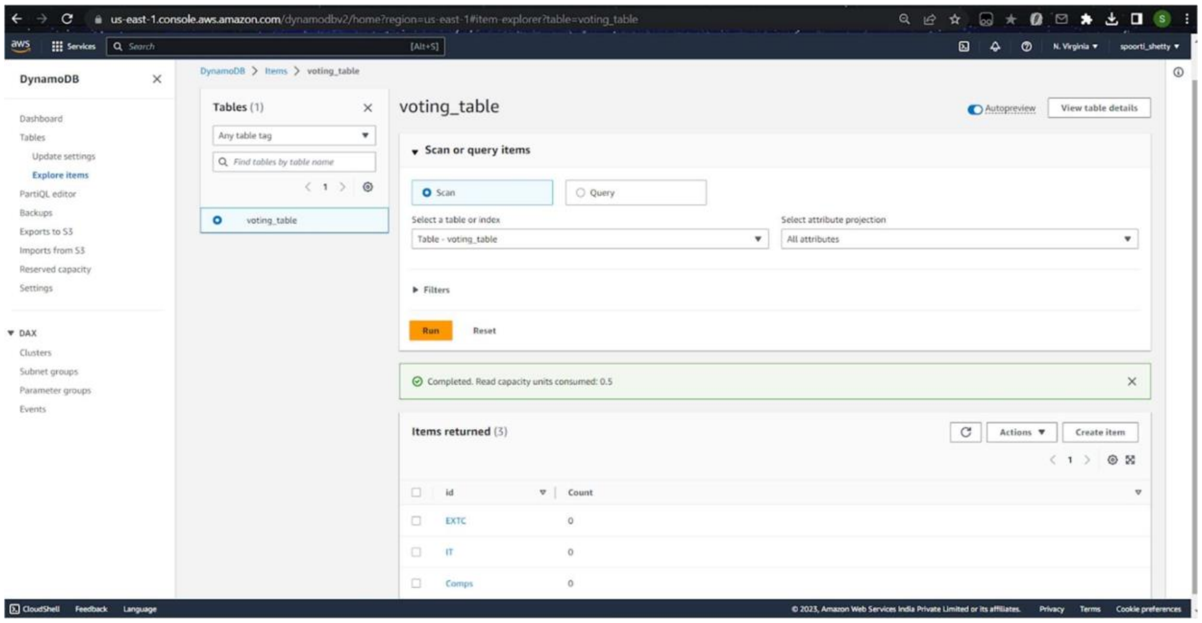
Create item

You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. Learn more

Attributes

Attribute name	Value	Type
id - Partition key	EXTC	String
Count	0	Number

g) DynamoDB table is ready.



Scan [✓] Query

Select a table or index: Table - voting_table

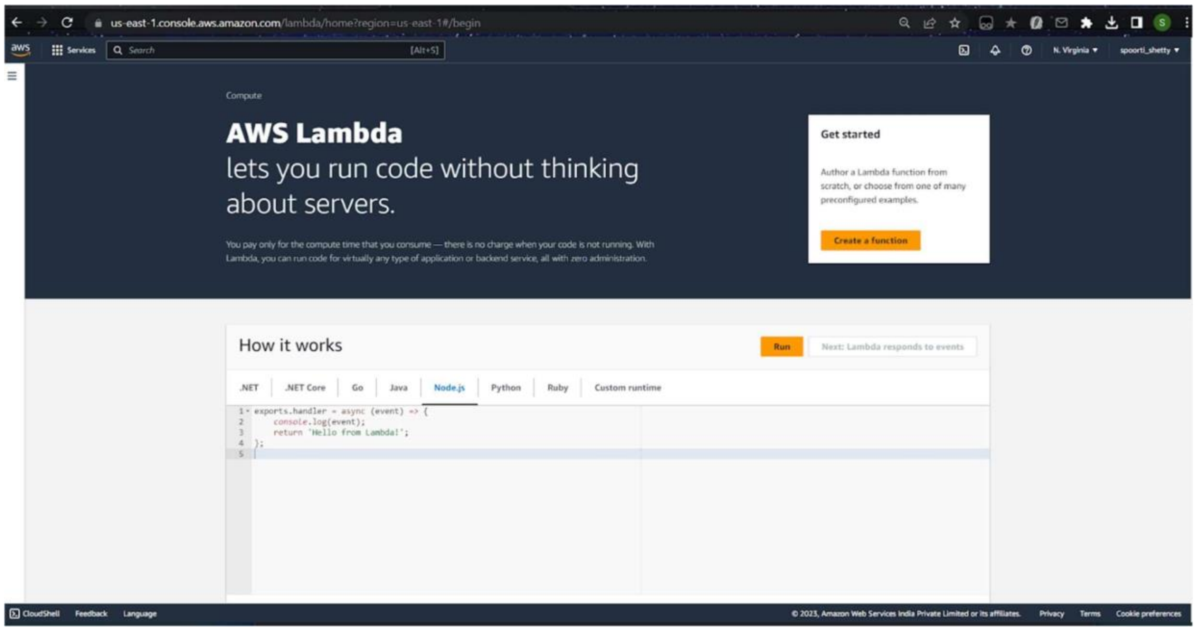
Select attribute projection: All attributes

Completed. Read capacity units consumed: 0.5

Items returned (3)

id	Count
EXTC	0
IT	0
Comps	0

h) Navigate to AWS Lambda and click on ‘Create a function’.



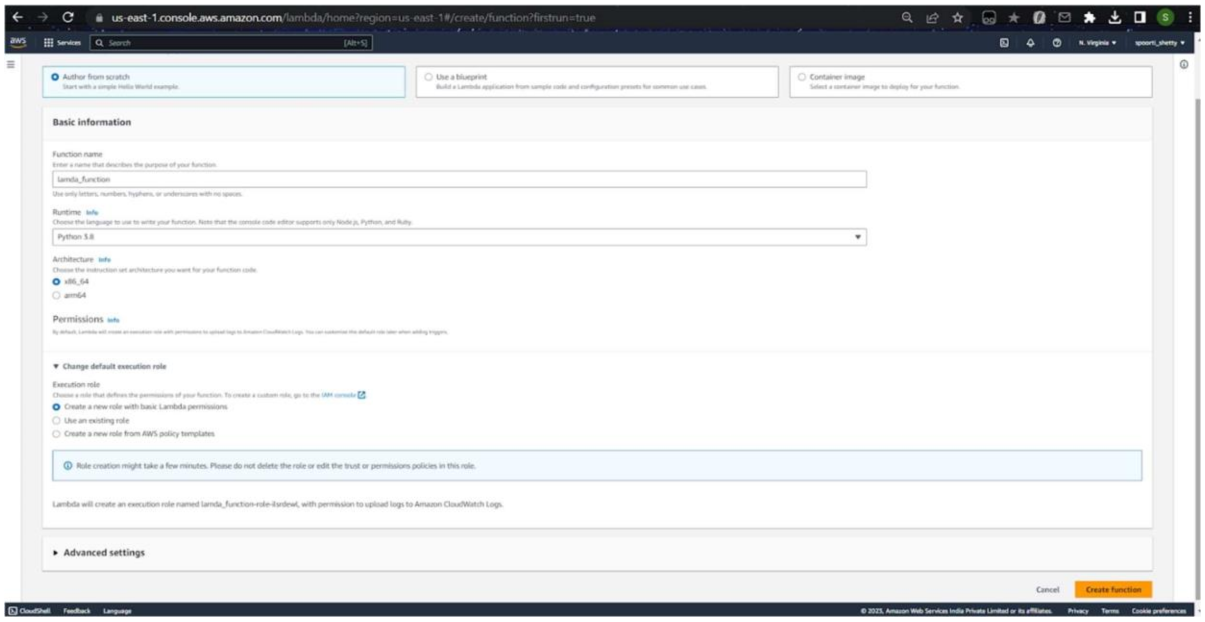
AWS Lambda
lets you run code without thinking about servers.

You pay only for the compute time that you consume — there is no charge when your code is not running. With Lambda, you can run code for virtually any type of application or backend service, all with zero administration.

Author a Lambda function from scratch, or choose from one of many preconfigured examples.

```
exports.handler = async (event) => {
    console.log(event);
    return 'Hello from Lambda!';
};
```

i) Let the function name be 'lamda_function', language for writing the function should be Python 3.8. Select 'Author from scratch'. Execution role should be 'Create a new role with basic Lambda permissions'.



Author from scratch — Start with a simple Hello World example.

Basic information

Function name: lamda_function

Runtime: Python 3.8

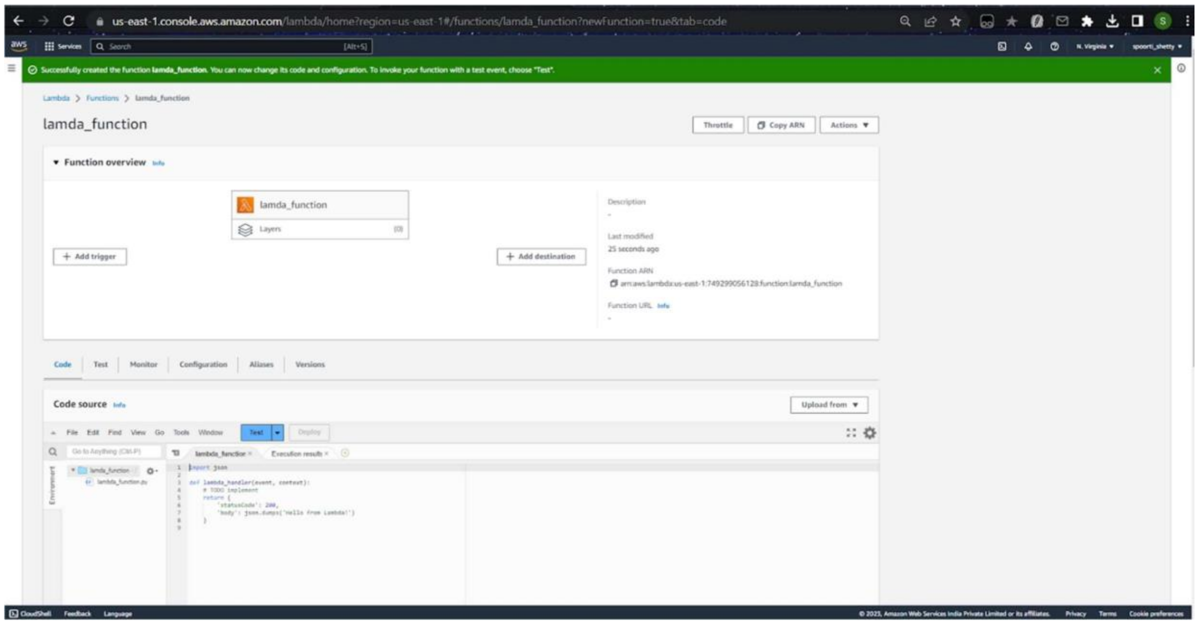
Architecture: x86_64

Execution role: Create a new role with basic Lambda permissions

Role creation might take a few minutes. Please do not delete the role or edit the trust or permissions policies in this role.

Lambda will create an execution role named lamda_function-role-ilsrdewl, with permission to upload logs to Amazon CloudWatch Logs.

j) 'lamda_function has been created. Now, scroll down and write the code for our new Lambda function and then configure test events.



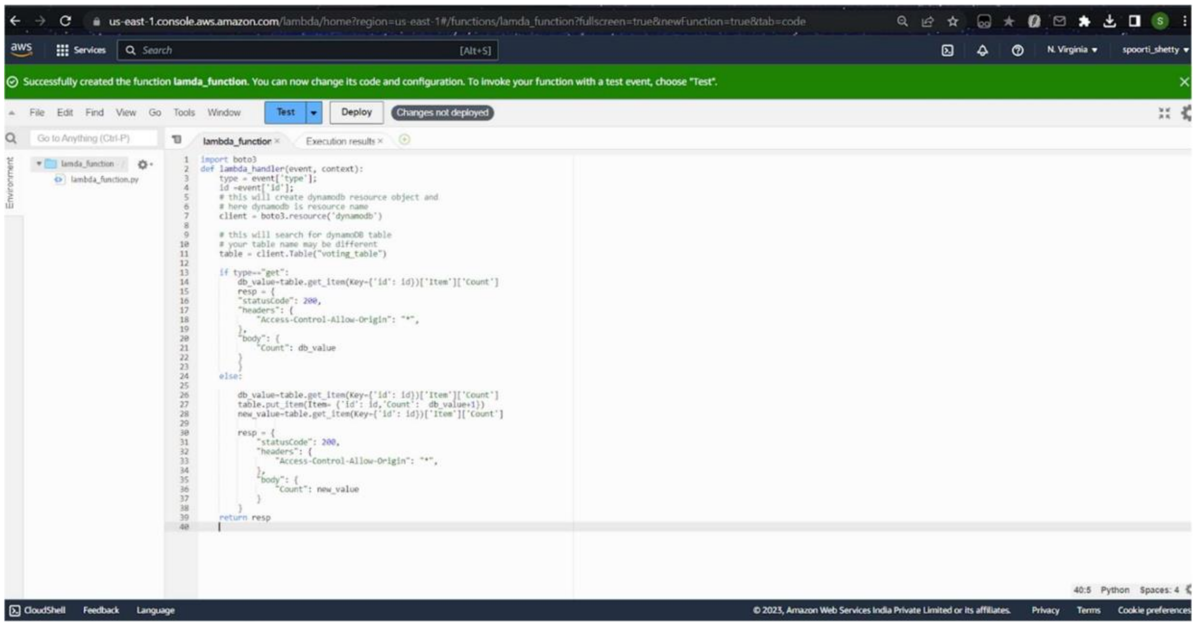
Successfully created the function **lamda_function**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

Last modified: 25 seconds ago

Function ARN: arn:aws:lambda:us-east-1:749299056128:function:lamda_function

```
import json

def lambda_handler(event, context):
    # TODO implement
    return {
        'statusCode': 200,
        'body': json.dumps('Hello from Lambda!')
    }
```



Successfully created the function **lamda_function**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

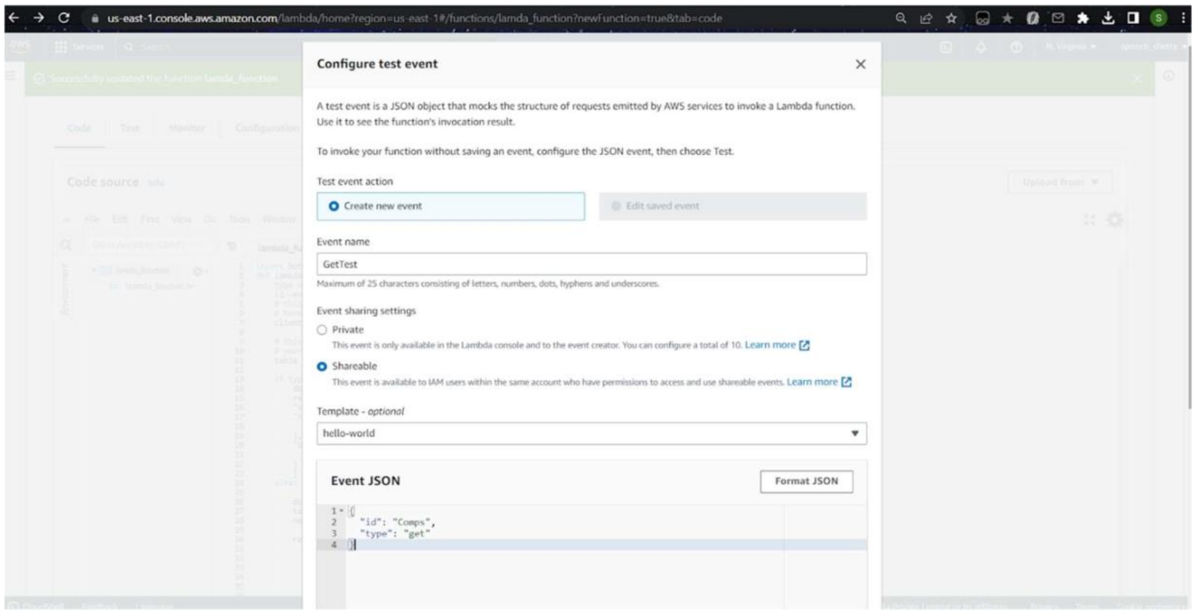
```python
import boto3
def lambda_handler(event, context):
    type = event['type'];
    id =event['id'];
    # this will create dynamodb resource object and
    # here dynamodb is resource name
    client = boto3.resource('dynamodb')

    # this will search for dynamoDB table
    # your table name may be different
    table = client.Table("voting_table")

    if type=="get":
        db_value=table.get_item(Key={'id': id})['Item']['Count']
        resp = {
        "statusCode": 200,
        "headers": {
            "Access-Control-Allow-Origin": "*",
        },
        "body": {
            "Count": db_value
        }
        }
    else:

        db_value=table.get_item(Key={'id': id})['Item']['Count']
        table.put_item(Item= {'id': id,'Count':  db_value+1})
        new_value=table.get_item(Key={'id': id})['Item']['Count']

        resp = {
            "statusCode": 200,
            "headers": {
                "Access-Control-Allow-Origin": "*",
            },
            "body": {
                "Count": new_value
            }
        }
    return resp
```



Configure test event

A test event is a JSON object that mocks the structure of requests emitted by AWS services to invoke a Lambda function. Use it to see the function's invocation result.

To invoke your function without saving an event, configure the JSON event, then choose Test.

Test event action: [✓ Create new event] Edit saved event

Event name: GetTest

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

Event sharing settings

Private [✓ Shareable]

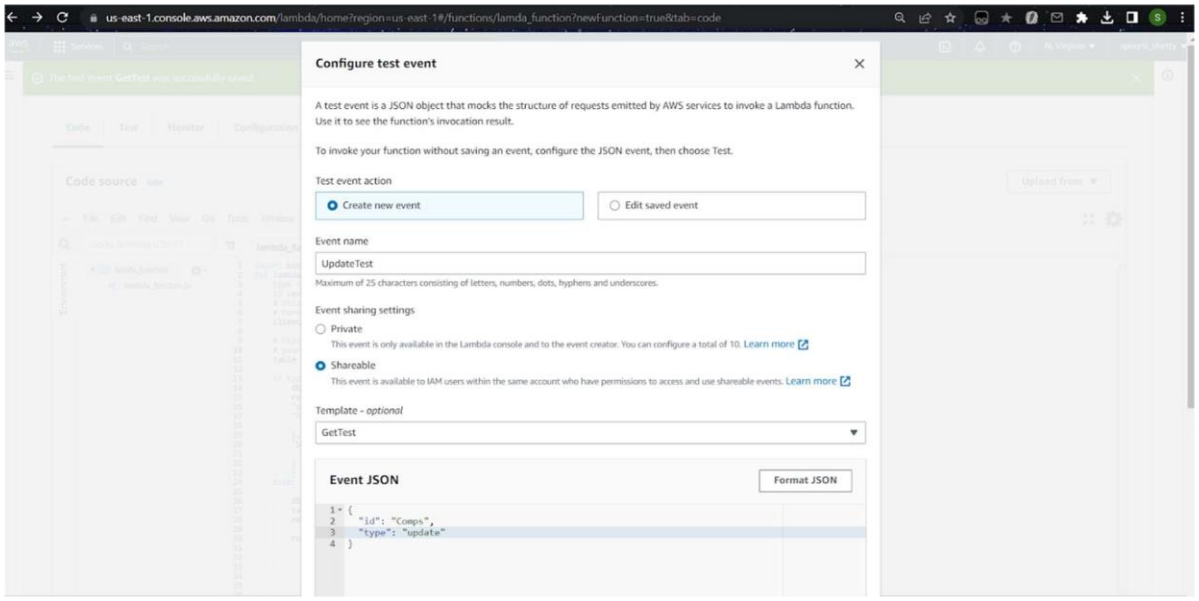
This event is only available in the Lambda console and to the event creator. You can configure a total of 10. Learn more

This event is available to IAM users within the same account who have permissions to access and use shareable events. Learn more

Template - *optional*: hello-world

Event JSON

```json
{
  "id": "Comps",
  "type": "get"
}
```


Configure test event

A test event is a JSON object that mocks the structure of requests emitted by AWS services to invoke a Lambda function. Use it to see the function's invocation result.

To invoke your function without saving an event, configure the JSON event, then choose Test.

Test event action: [✓ Create new event] Edit saved event

Event name: UpdateTest

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

Event sharing settings

Private: This event is only available in the Lambda console and to the event creator. You can configure a total of 10. Learn more

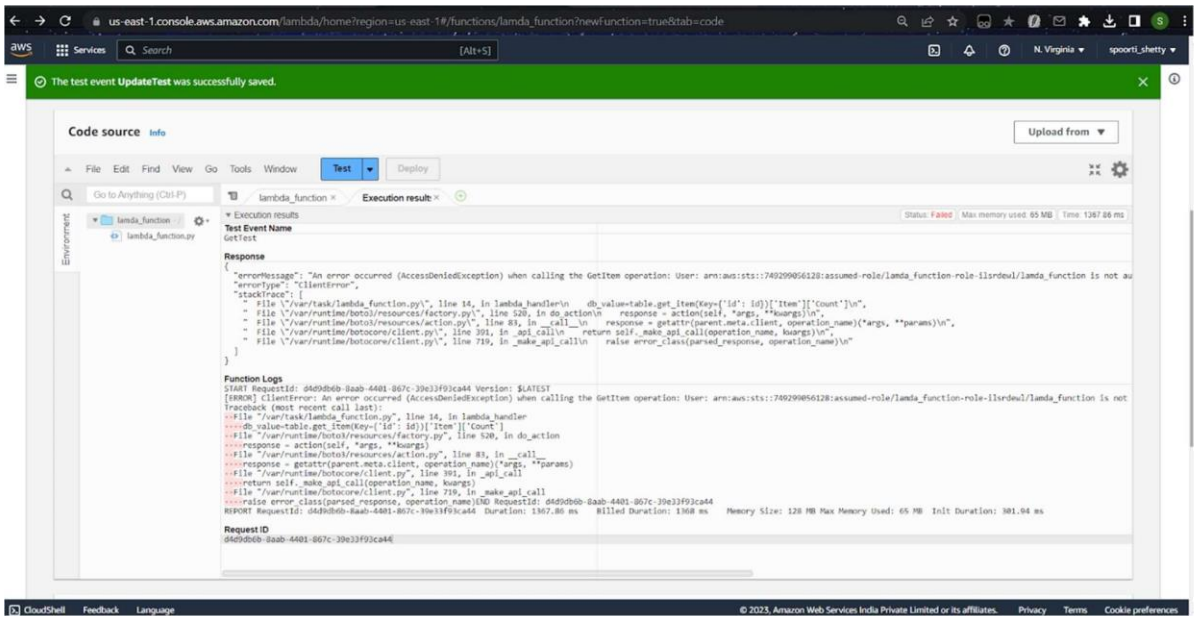
[✓ Shareable]: This event is available to IAM users within the same account who have permissions to access and use shareable events. Learn more

Template - *optional*: GetTest

Event JSON

```
{
  "id": "Comps",
  "type": "update"
}
```

k) When we deploy and run the test events, we get an error. This is because the lambda function role does not have enough permissions to access the DynamoDB database. Hence, we need to update the role to give full access of DynamoDB to Lambda function from IAM and re-run the test events.


The test event **UpdateTest** was successfully saved.

Code source

Execution results — Status: Failed | Max memory used: 65 MB | Time: 1367.86 ms

Test Event Name

GetTest

Response

```
{
  "errorMessage": "An error occurred (AccessDeniedException) when calling the GetItem operation: User: arn:aws:sts::749299056128:assumed-role/lamda_function-role-ilsrdewl/lamda_function is not au
  "errorType": "ClientError",
  "stackTrace": [
    "  File \"/var/task/lambda_function.py\", line 14, in lambda_handler\n    db_value=table.get_item(Key={'id': id})['Item']['Count']\n",
    "  File \"/var/runtime/boto3/resources/factory.py\", line 520, in do_action\n    response = action(self, *args, **kwargs)\n",
    "  File \"/var/runtime/boto3/resources/action.py\", line 83, in __call__\n    response = getattr(parent.meta.client, operation_name)(*args, **params)\n",
    "  File \"/var/runtime/botocore/client.py\", line 391, in _api_call\n    return self._make_api_call(operation_name, kwargs)\n",
    "  File \"/var/runtime/botocore/client.py\", line 719, in _make_api_call\n    raise error_class(parsed_response, operation_name)\n"
  ]
}
```

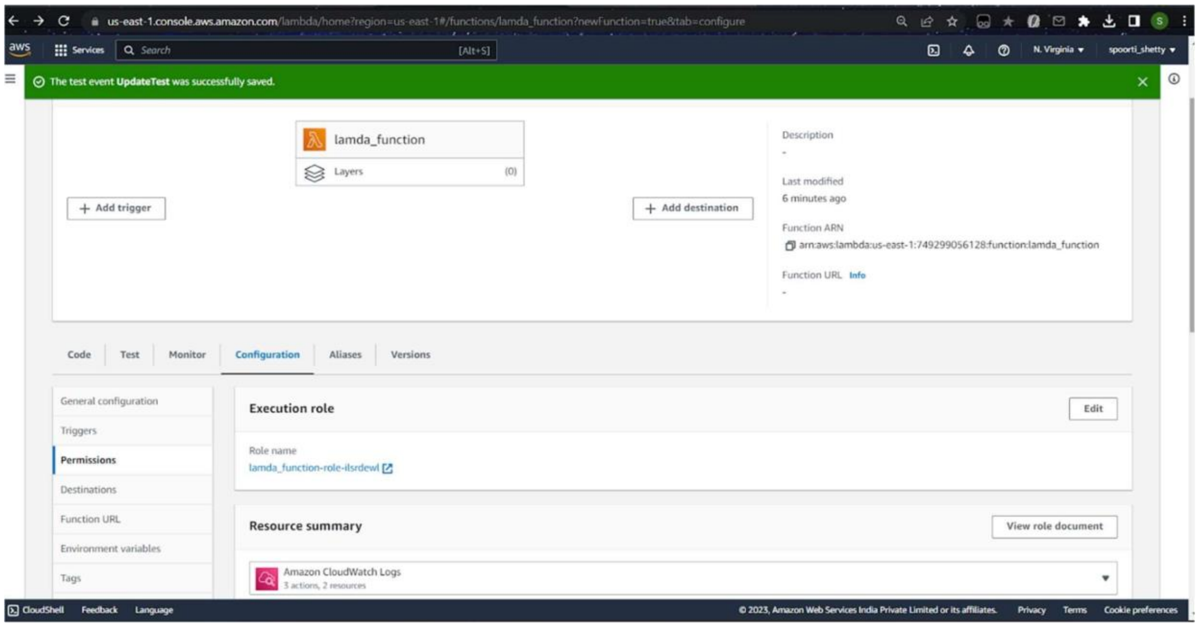
Function Logs

```
START RequestId: d4d9db6b-8aab-4401-867c-39e33f93ca44 Version: $LATEST
[ERROR] ClientError: An error occurred (AccessDeniedException) when calling the GetItem operation: User: arn:aws:sts::749299056128:assumed-role/lamda_function-role-ilsrdewl/lamda_function is not
Traceback (most recent call last):
  File "/var/task/lambda_function.py", line 14, in lambda_handler
    db_value=table.get_item(Key={'id': id})['Item']['Count']
  File "/var/runtime/boto3/resources/factory.py", line 520, in do_action
    response = action(self, *args, **kwargs)
  File "/var/runtime/boto3/resources/action.py", line 83, in __call__
    response = getattr(parent.meta.client, operation_name)(*args, **params)
  File "/var/runtime/botocore/client.py", line 391, in _api_call
    return self._make_api_call(operation_name, kwargs)
  File "/var/runtime/botocore/client.py", line 719, in _make_api_call
    raise error_class(parsed_response, operation_name)END RequestId: d4d9db6b-8aab-4401-867c-39e33f93ca44
REPORT RequestId: d4d9db6b-8aab-4401-867c-39e33f93ca44  Duration: 1367.86 ms    Billed Duration: 1368 ms    Memory Size: 128 MB Max Memory Used: 65 MB  Init Duration: 301.94 ms
```

Request ID

d4d9db6b-8aab-4401-867c-39e33f93ca44

© 2023, Amazon Web Services India Private Limited or its affiliates.



The test event **UpdateTest** was successfully saved.

Description
-

Last modified
6 minutes ago

Function ARN
arn:aws:lambda:us-east-1:749299056128:function:lamda_function

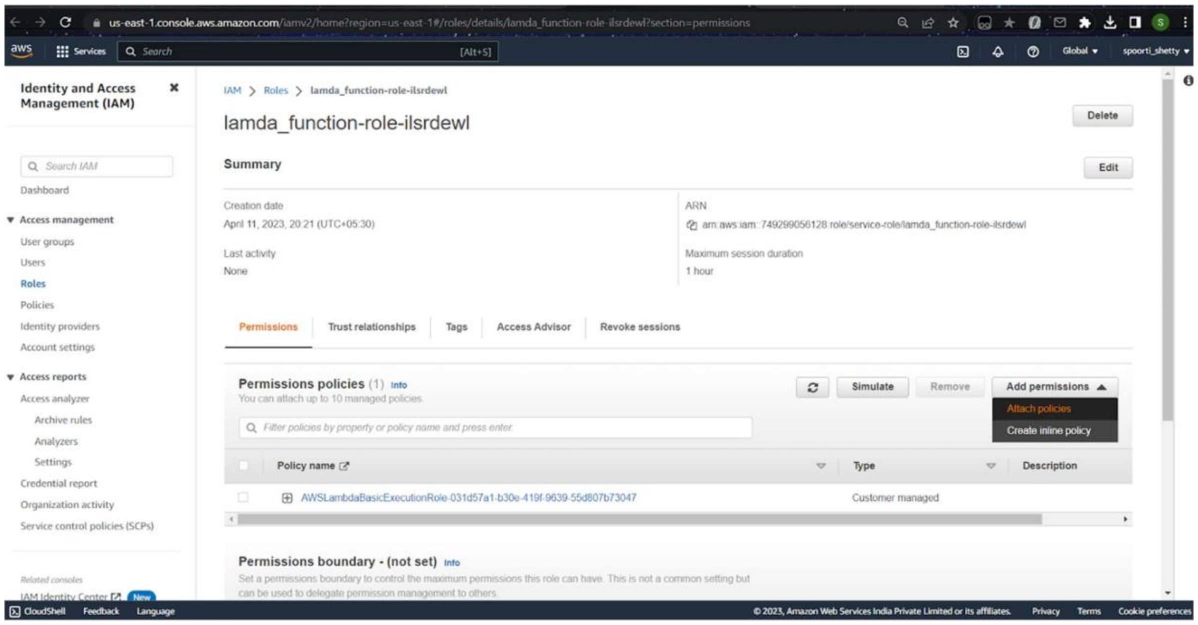
Function URL Info
-

Execution role

Role name
lamda_function-role-ilsrdewl

Resource summary

Amazon CloudWatch Logs
3 actions, 2 resources



lamda_function-role-ilsrdewl

Summary

Creation date
April 11, 2023, 20:21 (UTC+05:30)

Last activity
None

ARN
arn:aws:iam::749299056128:role/service-role/lamda_function-role-ilsrdewl

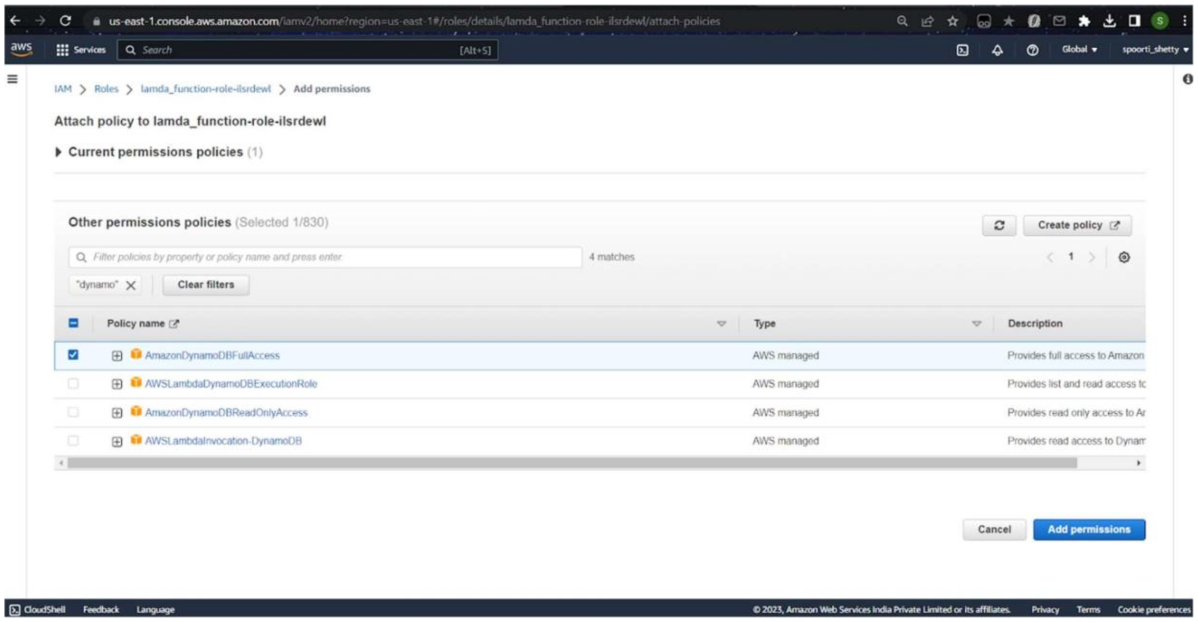
Maximum session duration
1 hour

Permissions policies (1) Info
You can attach up to 10 managed policies.

Filter policies by property or policy name and press enter

Policy name	Type	Description
AWSLambdaBasicExecutionRole-031d57a1-b30e-419f-9639-55d807b73047	Customer managed	

Permissions boundary - (not set) Info
Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting but can be used to delegate permission management to others.



IAM > Roles > lamda_function-role-ilsrdewl > Add permissions

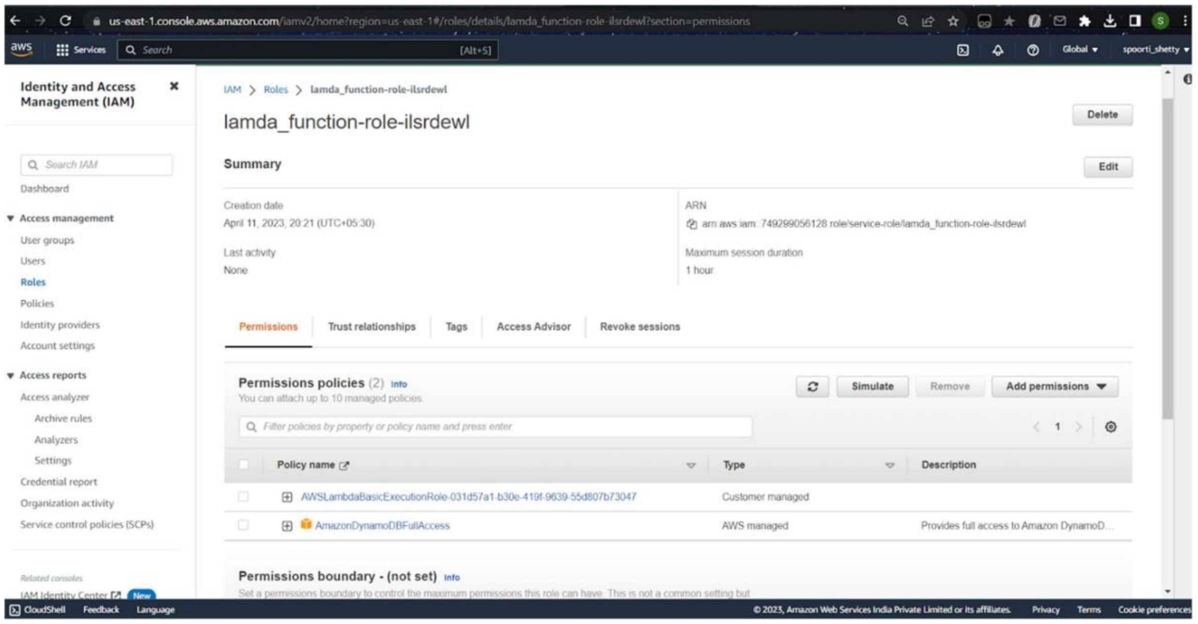
Attach policy to lamda_function-role-ilsrdewl

Current permissions policies (1)

Other permissions policies (Selected 1/830)

"dynamo" — 4 matches

	Policy name	Type	Description
✓	AmazonDynamoDBFullAccess	AWS managed	Provides full access to Amazon
	AWSLambdaDynamoDBExecutionRole	AWS managed	Provides list and read access to
	AmazonDynamoDBReadOnlyAccess	AWS managed	Provides read only access to Ar
	AWSLambdaInvocation-DynamoDB	AWS managed	Provides read access to Dynam



IAM > Roles > lamda_function-role-ilsrdewl

lamda_function-role-ilsrdewl

Summary

Creation date: April 11, 2023, 20:21 (UTC+05:30)

ARN: arn:aws:iam::749299056128:role/service-role/lamda_function-role-ilsrdewl

Last activity: None

Maximum session duration: 1 hour

Permissions policies (2)

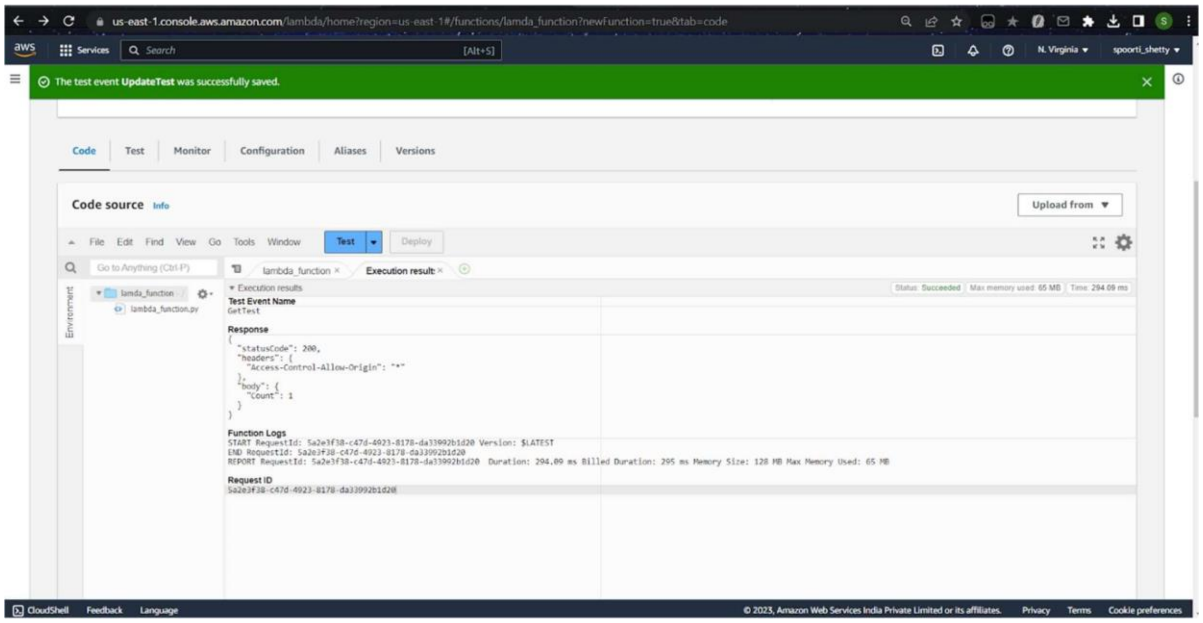
You can attach up to 10 managed policies.

	Policy name	Type	Description
	AWSLambdaBasicExecutionRole-031d57a1-b30e-419f-9639-55d807b73047	Customer managed	
	AmazonDynamoDBFullAccess	AWS managed	Provides full access to Amazon DynamoD...

Permissions boundary - (not set)

Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting but

l) The tests now run successfully without any errors and also the database is getting updated as required.



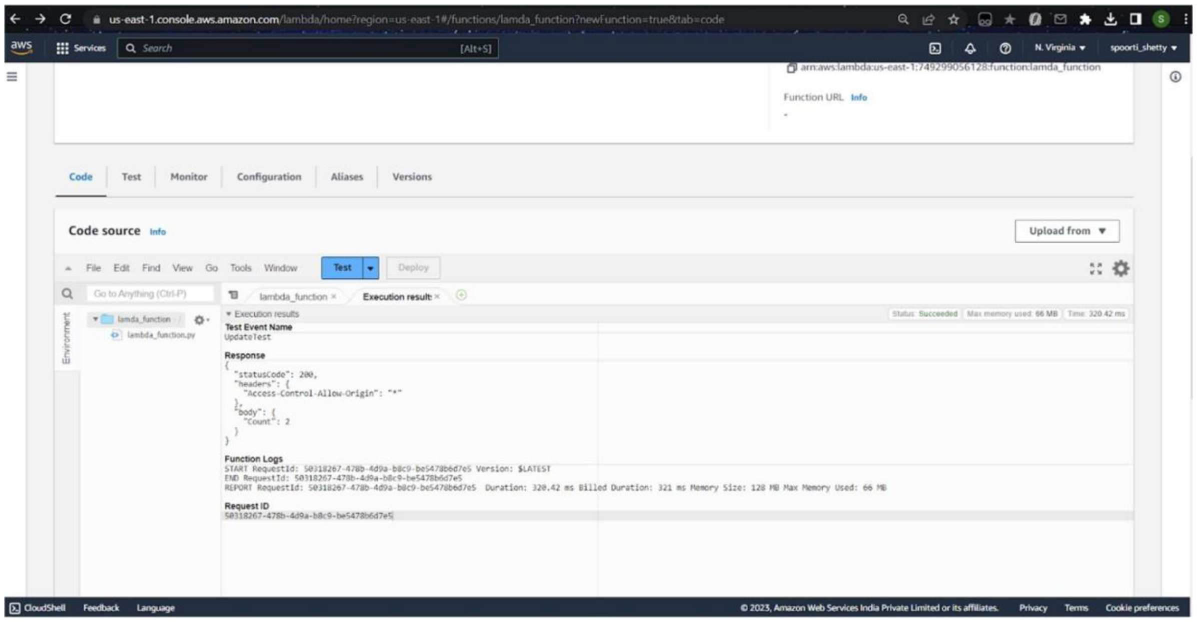
The test event UpdateTest was successfully saved.

Test Event Name
GetTest

Response
```
{
  "statusCode": 200,
  "headers": {
    "Access-Control-Allow-Origin": "*"
  },
  "body": {
    "Count": 1
  }
}
```

Function Logs
```
START RequestId: 5a2e3f38-c47d-4923-8178-da33992b1d20 Version: $LATEST
END RequestId: 5a2e3f38-c47d-4923-8178-da33992b1d20
REPORT RequestId: 5a2e3f38-c47d-4923-8178-da33992b1d20  Duration: 294.09 ms Billed Duration: 295 ms Memory Size: 128 MB Max Memory Used: 65 MB
```

Request ID
5a2e3f38-c47d-4923-8178-da33992b1d20

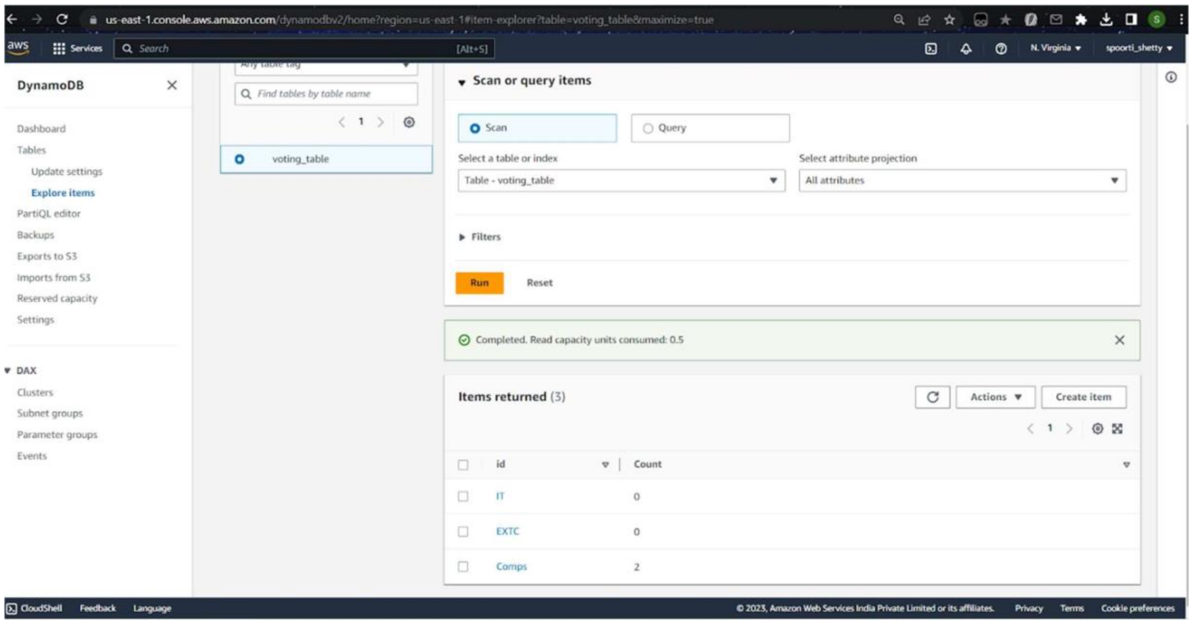


Test Event Name
UpdateTest

Response
```
{
  "statusCode": 200,
  "headers": {
    "Access-Control-Allow-Origin": "*"
  },
  "body": {
    "Count": 2
  }
}
```

Function Logs
```
START RequestId: 50318267-478b-4d9a-b8c9-be5478b6d7e5 Version: $LATEST
END RequestId: 50318267-478b-4d9a-b8c9-be5478b6d7e5
REPORT RequestId: 50318267-478b-4d9a-b8c9-be5478b6d7e5  Duration: 320.42 ms Billed Duration: 321 ms Memory Size: 128 MB Max Memory Used: 66 MB
```

Request ID
50318267-478b-4d9a-b8c9-be5478b6d7e5



Scan or query items

Scan [✓] / Query

Select a table or index: Table - voting_table

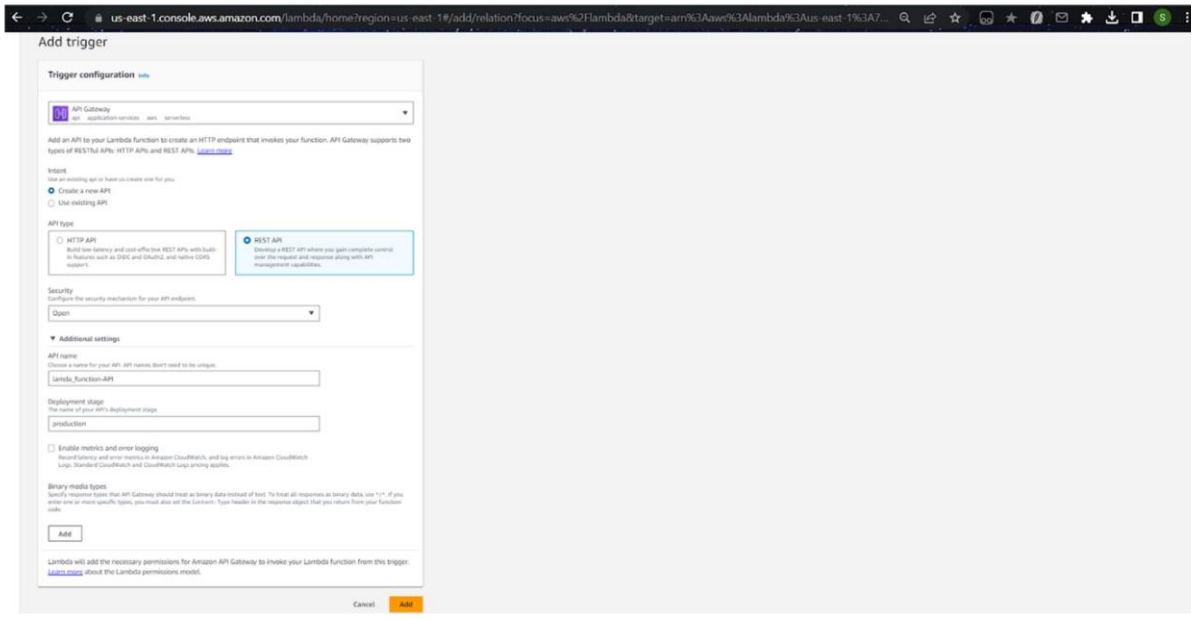
Select attribute projection: All attributes

Completed. Read capacity units consumed: 0.5

Items returned (3)

id	Count
IT	0
EXTC	0
Comps	2

m) Add a trigger for API gateway and access it.



Add trigger

Trigger configuration

API Gateway

Add an API to your Lambda function to create an HTTP endpoint that invokes your function. API Gateway supports two types of RESTful APIs: HTTP APIs and REST APIs. Learn more

Intent: [✓ Create a new API] / Use existing API

API type: HTTP API / [✓ REST API]

Security: Open

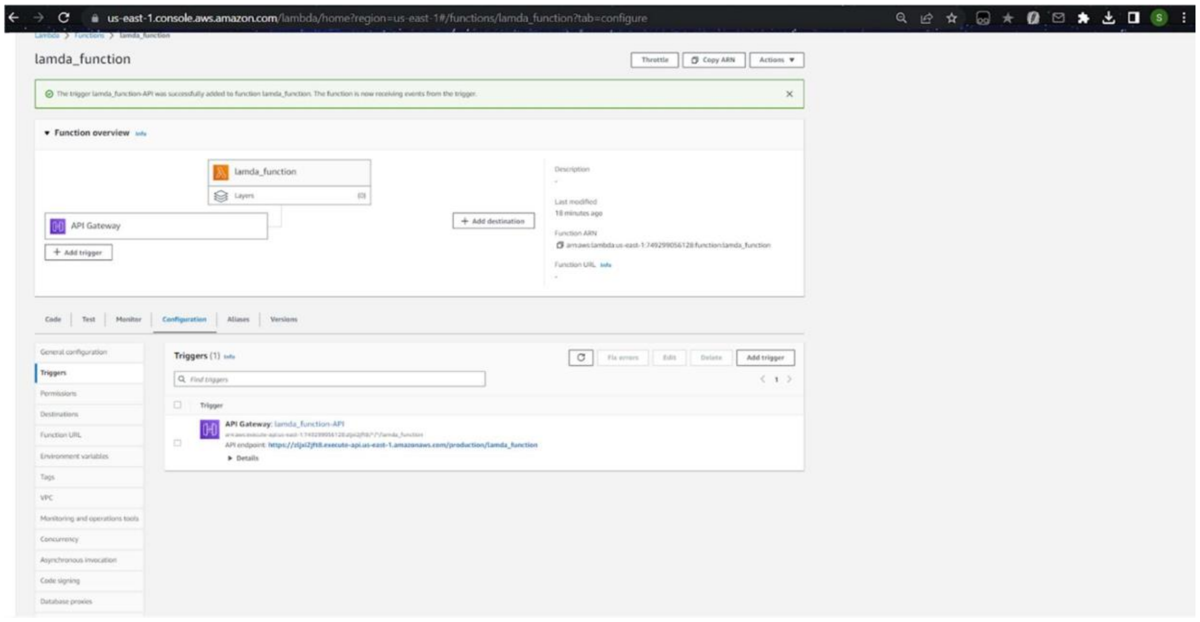
Additional settings

API name: lamda_function-API

Deployment stage: production

Enable metrics and error logging [ ]

Lambda will add the necessary permissions for Amazon API Gateway to invoke your Lambda function from this trigger. Learn more about the Lambda permissions model.



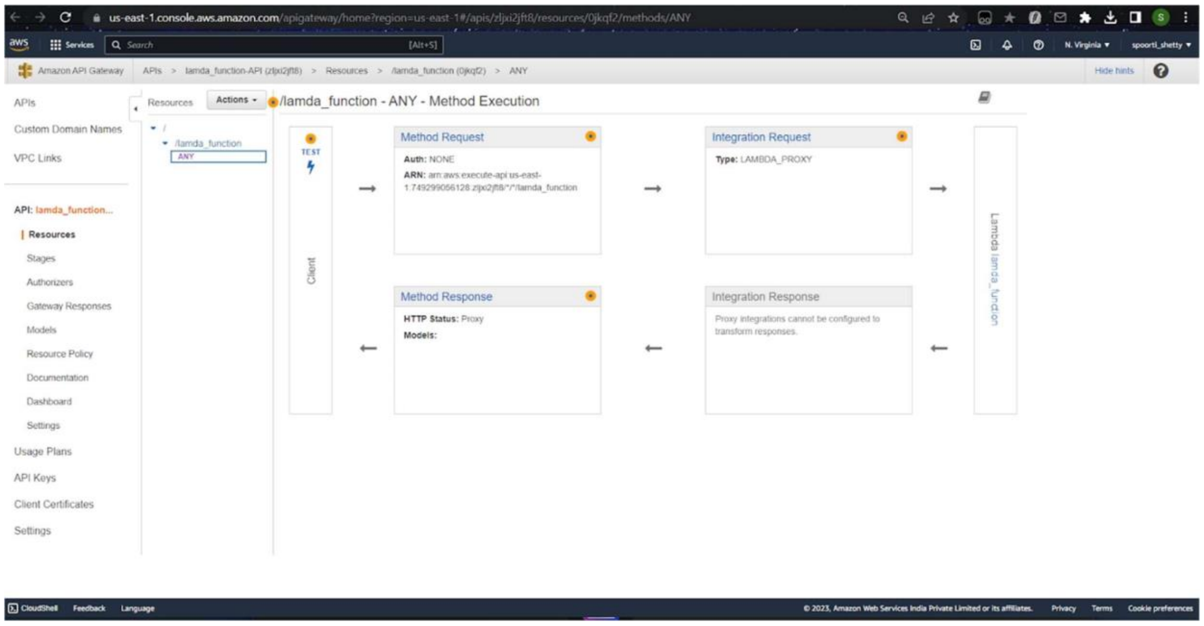
lamda_function

The trigger lamda_function-API was successfully added to function lamda_function. The function is now receiving events from the trigger.

Description: -
Last modified: 18 minutes ago
Function ARN: arn:aws:lambda:us-east-1:749299056128:function:lamda_function
Function URL: -

Triggers (1)

Trigger
API Gateway: lamda_function-API arn:aws:execute-api:us-east-1:749299056128:zljxi2jft8/*/*/lamda_function API endpoint: https://zljxi2jft8.execute-api.us-east-1.amazonaws.com/production/lamda_function



/lamda_function - ANY - Method Execution

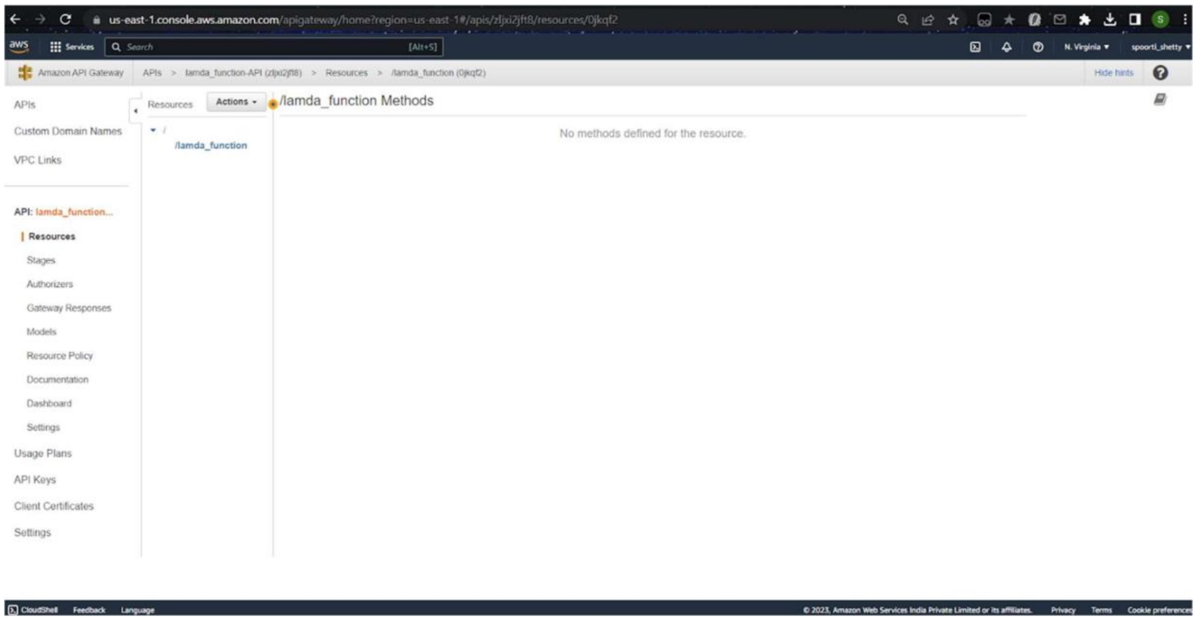
Method Request
Auth: NONE
ARN: arn:aws:execute-api:us-east-1:749299056128:zljxi2jft8/*/*/lamda_function

Integration Request
Type: LAMBDA_PROXY

Method Response
HTTP Status: Proxy
Models:

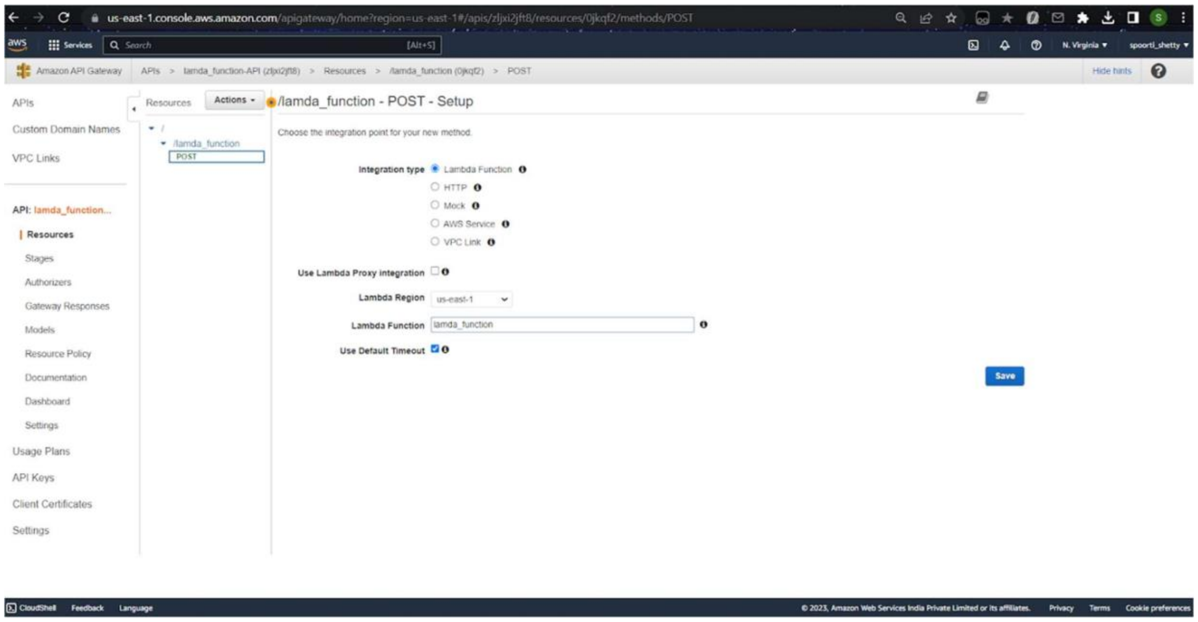
Integration Response
Proxy integrations cannot be configured to transform responses.

n) Delete the ANY method and create a new POST method.



/lamda_function Methods

No methods defined for the resource.



/lamda_function - POST - Setup

Choose the integration point for your new method.

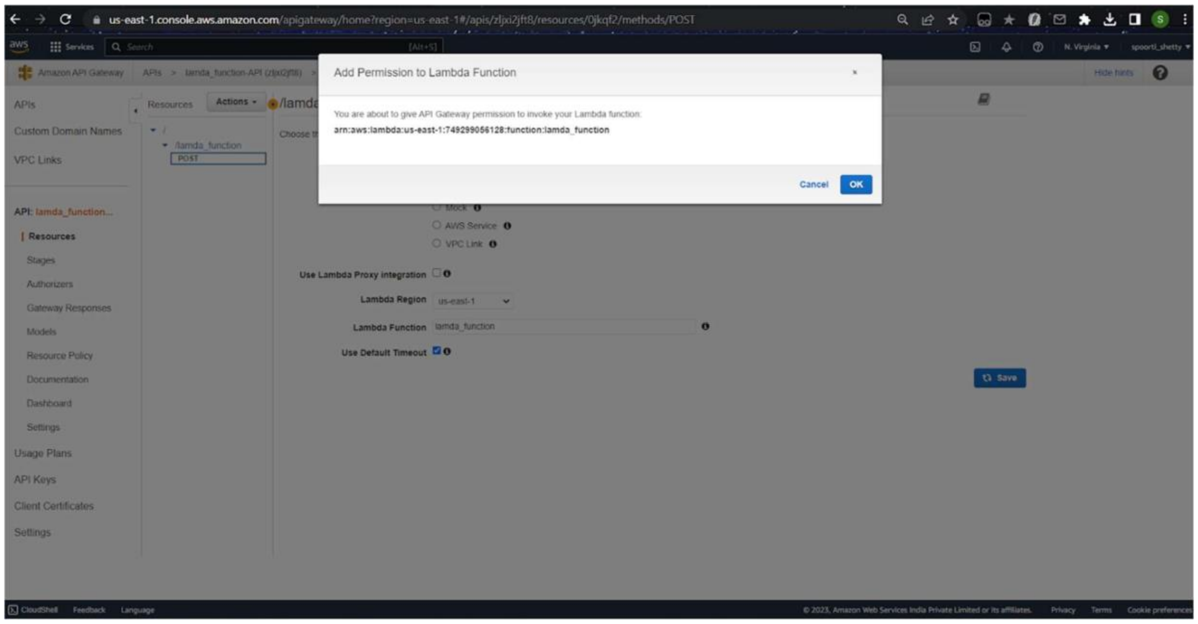
Integration type: [✓ Lambda Function] HTTP Mock AWS Service VPC Link

Use Lambda Proxy integration: ☐

Lambda Region: us-east-1

Lambda Function: lamda_function

Use Default Timeout: ✓



us-east-1.console.aws.amazon.com/apigateway/home?region=us-east-1#/apis/zljxi2jft8/resources/0jkqf2/methods/POST

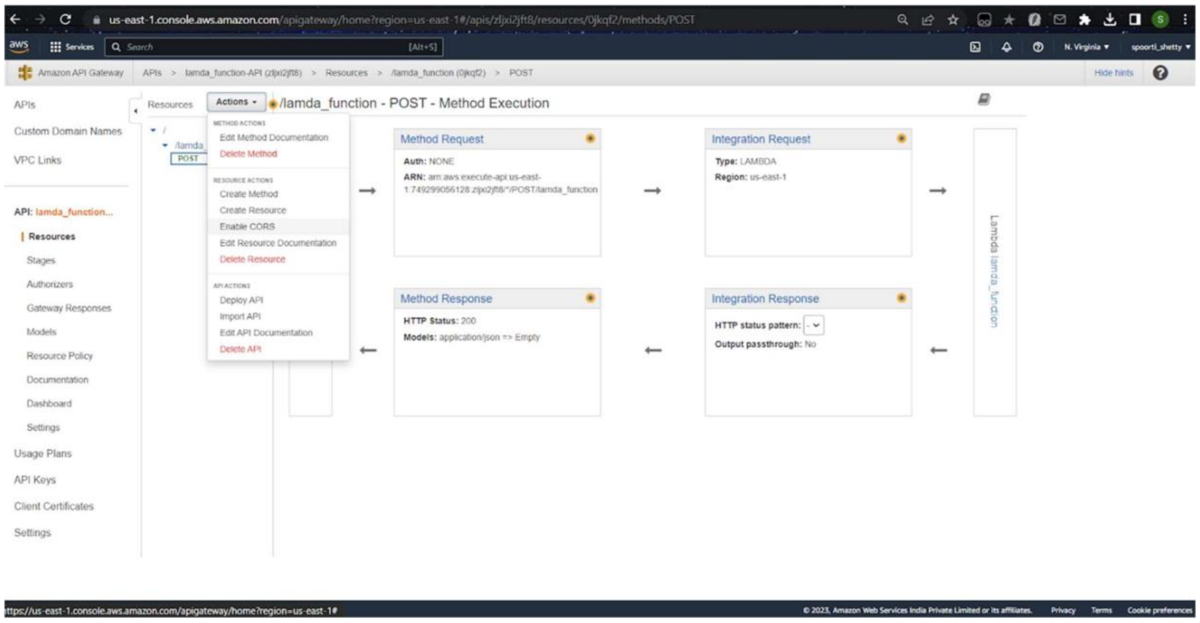
Add Permission to Lambda Function

You are about to give API Gateway permission to invoke your Lambda function:
arn:aws:lambda:us-east-1:749299056128:function:lamda_function

Use Lambda Proxy integration ☐
Lambda Region: us-east-1
Lambda Function: lamda_function
Use Default Timeout ☑

© 2023, Amazon Web Services India Private Limited or its affiliates.

o) Enable CORS.



us-east-1.console.aws.amazon.com/apigateway/home?region=us-east-1#/apis/zljxi2jft8/resources/0jkqf2/methods/POST

APIs > lamda_function-API (zljxi2jft8) > Resources > /lamda_function (0jkqf2) > POST

/lamda_function - POST - Method Execution

Method Request
Auth: NONE
ARN: arn:aws:execute-api:us-east-1:749299056128:zljxi2jft8/*/POST/lamda_function

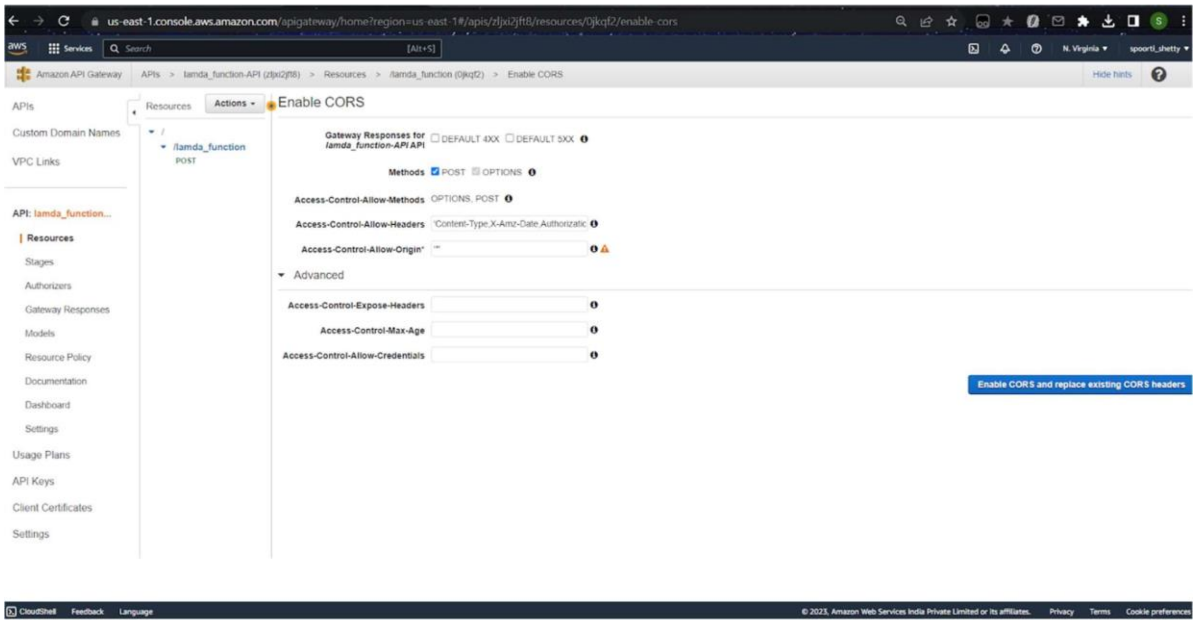
Integration Request
Type: LAMBDA
Region: us-east-1

Method Response
HTTP Status: 200
Models: application/json => Empty

Integration Response
HTTP status pattern: -
Output passthrough: No

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

© 2023, Amazon Web Services India Private Limited or its affiliates.


us-east-1.console.aws.amazon.com/apigateway/home?region=us-east-1#/apis/zljxi2jft8/resources/0jkqf2/enable-cors

APIs > lamda_function-API (zljxi2jft8) > Resources > /lamda_function (0jkqf2) > Enable CORS

Enable CORS

Gateway Responses for *lamda_function-API* API: ☐ DEFAULT 4XX ☐ DEFAULT 5XX

Methods: ☑ POST ☐ OPTIONS

Access-Control-Allow-Methods: OPTIONS, POST

Access-Control-Allow-Headers: 'Content-Type,X-Amz-Date,Authorizatic

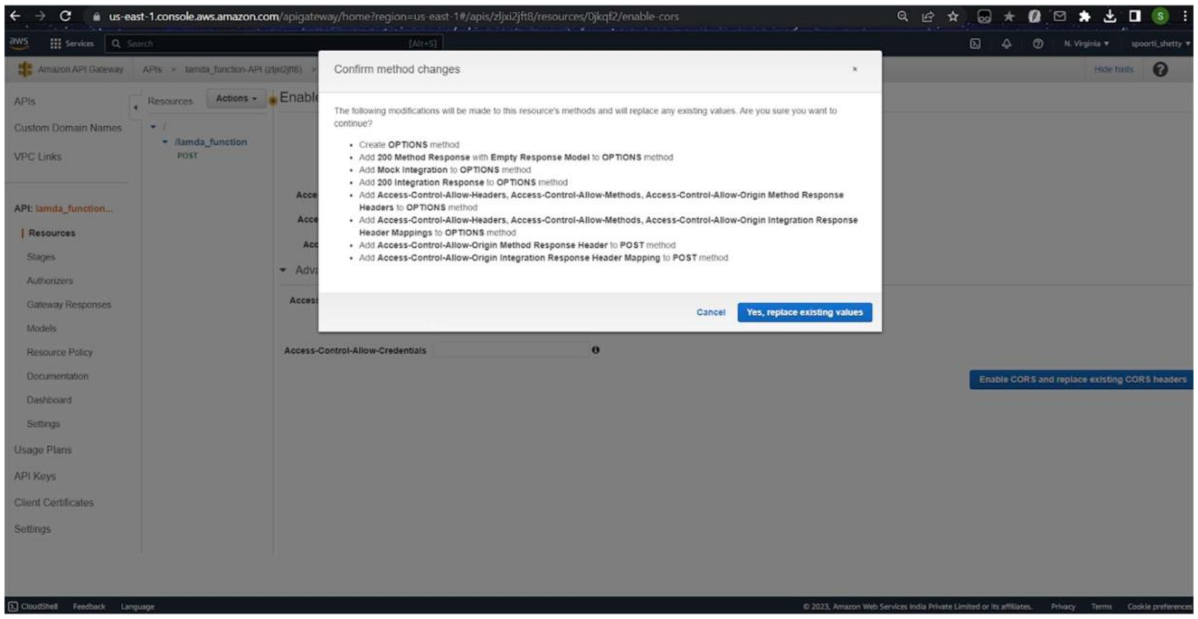
Access-Control-Allow-Origin*: '*'

Advanced

Access-Control-Expose-Headers:

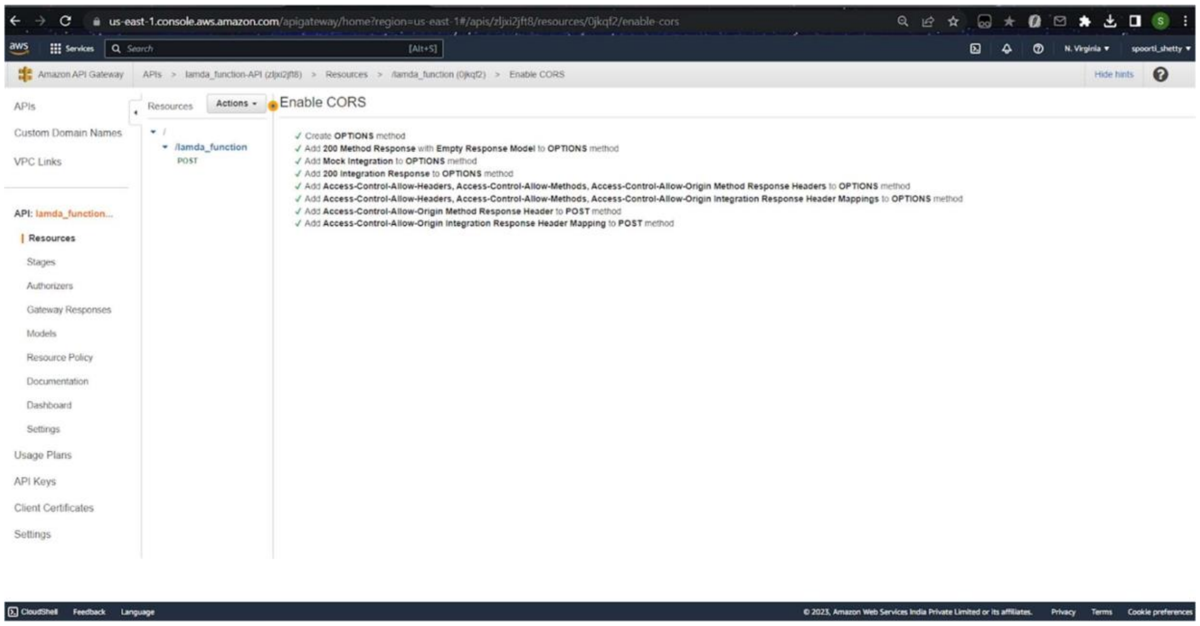
Access-Control-Max-Age:

Access-Control-Allow-Credentials:


Confirm method changes

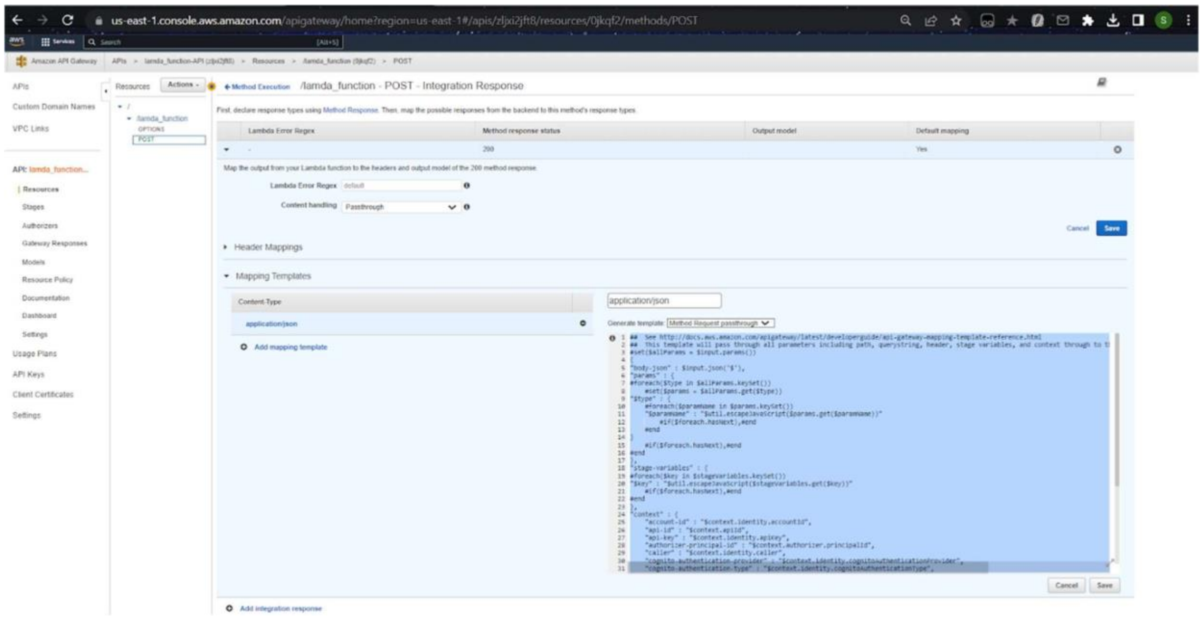
The following modifications will be made to this resource's methods and will replace any existing values. Are you sure you want to continue?

- Create **OPTIONS** method
- Add **200 Method Response** with **Empty Response Model** to **OPTIONS** method
- Add **Mock Integration** to **OPTIONS** method
- Add **200 Integration Response** to **OPTIONS** method
- Add **Access-Control-Allow-Headers, Access-Control-Allow-Methods, Access-Control-Allow-Origin Method Response Headers** to **OPTIONS** method
- Add **Access-Control-Allow-Headers, Access-Control-Allow-Methods, Access-Control-Allow-Origin Integration Response Header Mappings** to **OPTIONS** method
- Add **Access-Control-Allow-Origin Method Response Header** to **POST** method
- Add **Access-Control-Allow-Origin Integration Response Header Mapping** to **POST** method


Enable CORS

✓ Create **OPTIONS** method
✓ Add **200 Method Response** with **Empty Response Model** to **OPTIONS** method
✓ Add **Mock Integration** to **OPTIONS** method
✓ Add **200 Integration Response** to **OPTIONS** method
✓ Add **Access-Control-Allow-Headers, Access-Control-Allow-Methods, Access-Control-Allow-Origin Method Response Headers** to **OPTIONS** method
✓ Add **Access-Control-Allow-Headers, Access-Control-Allow-Methods, Access-Control-Allow-Origin Integration Response Header Mappings** to **OPTIONS** method
✓ Add **Access-Control-Allow-Origin Method Response Header** to **POST** method
✓ Add **Access-Control-Allow-Origin Integration Response Header Mapping** to **POST** method

p) Add mapping templates in Integration Response of POST method with content type as ‘application/json’.


/lamda_function - POST - Integration Response

First, declare response types using Method Response. Then, map the possible responses from the backend to this method's response types.

Lambda Error Regex	Method response status	Output model	Default mapping
-	200		Yes

Map the output from your Lambda function to the headers and output model of the 200 method response.

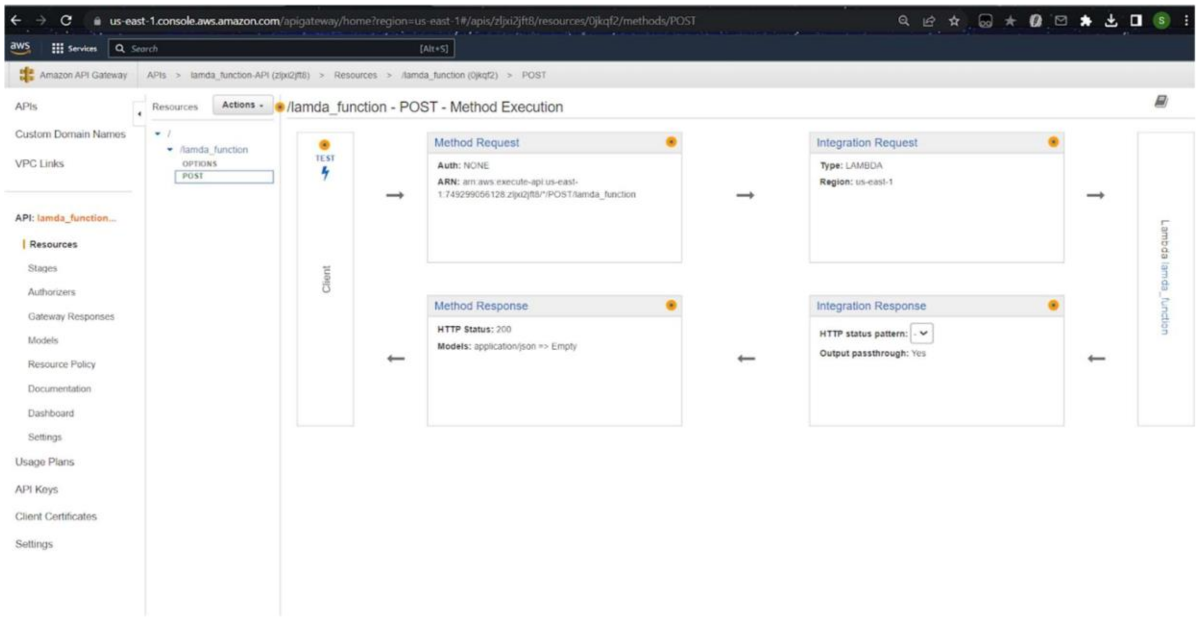
Lambda Error Regex: default
Content handling: Passthrough

Mapping Templates

Content-Type: application/json

Generate template: Method Request passthrough

```
##  See http://docs.aws.amazon.com/apigateway/latest/developerguide/api-gateway-mapping-template-reference.html
##  This template will pass through all parameters including path, querystring, header, stage variables, and context through to t
#set($allParams = $input.params())
{
"body-json" : $input.json('$'),
"params" : {
#foreach($type in $allParams.keySet())
    #set($params = $allParams.get($type))
"$type" : {
    #foreach($paramName in $params.keySet())
    "$paramName" : "$util.escapeJavaScript($params.get($paramName))"
        #if($foreach.hasNext),#end
    #end
}
    #if($foreach.hasNext),#end
#end
},
"stage-variables" : {
#foreach($key in $stageVariables.keySet())
"$key" : "$util.escapeJavaScript($stageVariables.get($key))"
    #if($foreach.hasNext),#end
#end
},
"context" : {
    "account-id" : "$context.identity.accountId",
    "api-id" : "$context.apiId",
    "api-key" : "$context.identity.apiKey",
    "authorizer-principal-id" : "$context.authorizer.principalId",
    "caller" : "$context.identity.caller",
    "cognito-authentication-provider" : "$context.identity.cognitoAuthenticationProvider",
    "cognito-authentication-type" : "$context.identity.cognitoAuthenticationType",
```



/lamda_function - POST - Method Execution

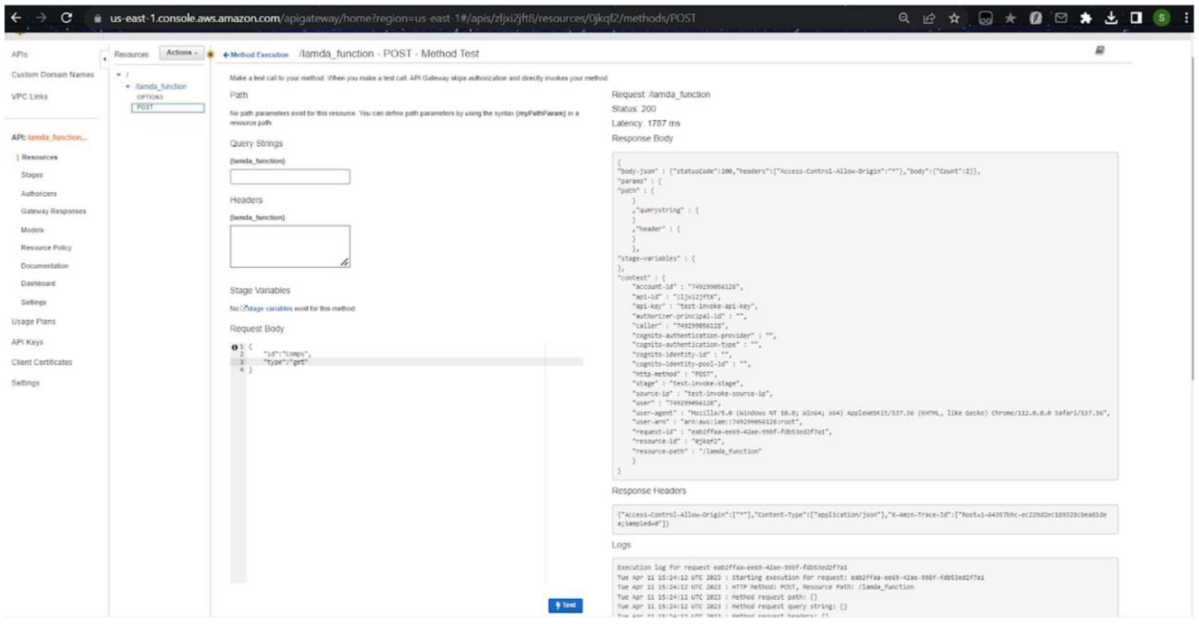
Method Request
Auth: NONE
ARN: arn:aws:execute-api:us-east-1:749299056128:zljxi2jft8/*/POST/lamda_function

Integration Request
Type: LAMBDA
Region: us-east-1

Method Response
HTTP Status: 200
Models: application/json => Empty

Integration Response
HTTP status pattern: -
Output passthrough: Yes

q) Test the POST method. It gives the required output.



/lamda_function - POST - Method Test

Make a test call to your method. When you make a test call, API Gateway skips authorization and directly invokes your method

Path

No path parameters exist for this resource. You can define path parameters by using the syntax {myPathParam} in a resource path.

Query Strings

{lamda_function}

Headers

{lamda_function}

Stage Variables

No stage variables exist for this method

Request Body

```
{
    "id":"comps",
    "type":"get"
}
```

Request: /lamda_function
Status: 200
Latency: 1787 ms
Response Body

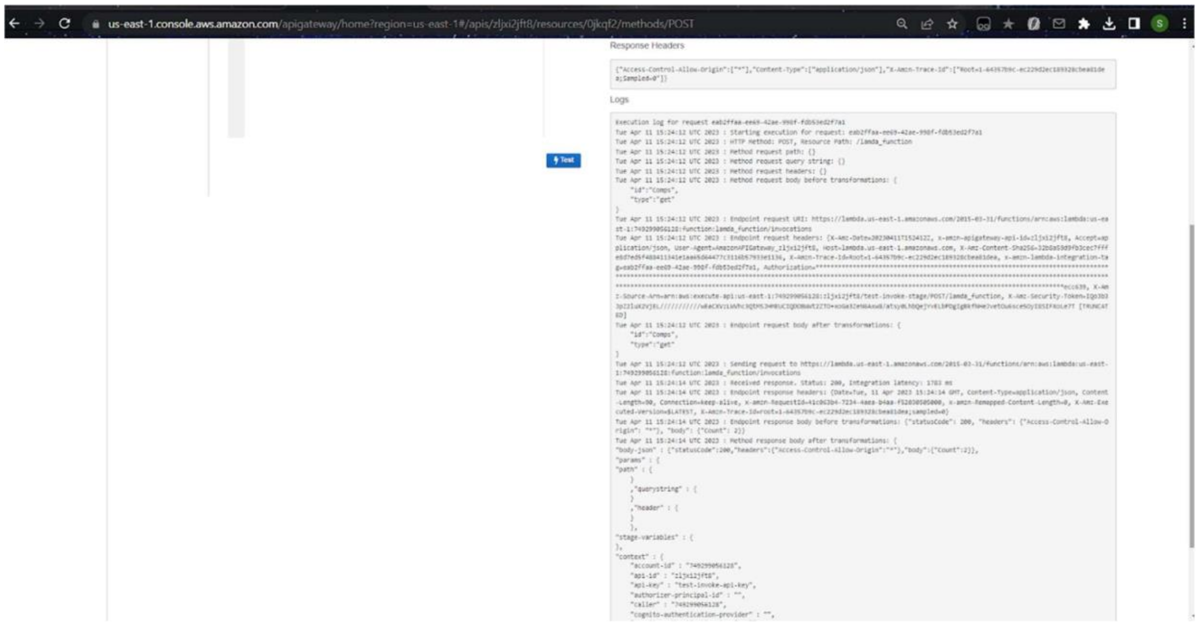
```
{
"body-json" : {"statusCode":200,"headers":{"Access-Control-Allow-Origin":"*"},"body":{"Count":2}},
"params" : {
"path" : {
    }
    ,"querystring" : {
    }
    ,"header" : {
    }
    },
"stage-variables" : {
},
"context" : {
    "account-id" : "749299056128",
    "api-id" : "zljxi2jft8",
    "api-key" : "test-invoke-api-key",
    "authorizer-principal-id" : "",
    "caller" : "749299056128",
    "cognito-authentication-provider" : "",
    "cognito-authentication-type" : "",
    "cognito-identity-id" : "",
    "cognito-identity-pool-id" : "",
    "http-method" : "POST",
    "stage" : "test-invoke-stage",
    "source-ip" : "test-invoke-source-ip",
    "user" : "749299056128",
    "user-agent" : "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.0.0 Safari/537.36",
    "user-arn" : "arn:aws:iam::749299056128:root",
    "request-id" : "eab2ffaa-ee69-42ae-998f-fdb53ed2f7a1",
    "resource-id" : "0jkqf2",
    "resource-path" : "/lamda_function"
    }
}
```

Response Headers

```
{"Access-Control-Allow-Origin":["*"],"Content-Type":["application/json"],"X-Amzn-Trace-Id":["Root=1-64357b9c-ec229d2ec189328cbea81de
a;Sampled=0"]}
```

Logs

```
Execution log for request eab2ffaa-ee69-42ae-998f-fdb53ed2f7a1
Tue Apr 11 15:24:12 UTC 2023 : Starting execution for request: eab2ffaa-ee69-42ae-998f-fdb53ed2f7a1
Tue Apr 11 15:24:12 UTC 2023 : HTTP Method: POST, Resource Path: /lamda_function
Tue Apr 11 15:24:12 UTC 2023 : Method request path: {}
Tue Apr 11 15:24:12 UTC 2023 : Method request query string: {}
Tue Apr 11 15:24:12 UTC 2023 : Method request headers: {}
```



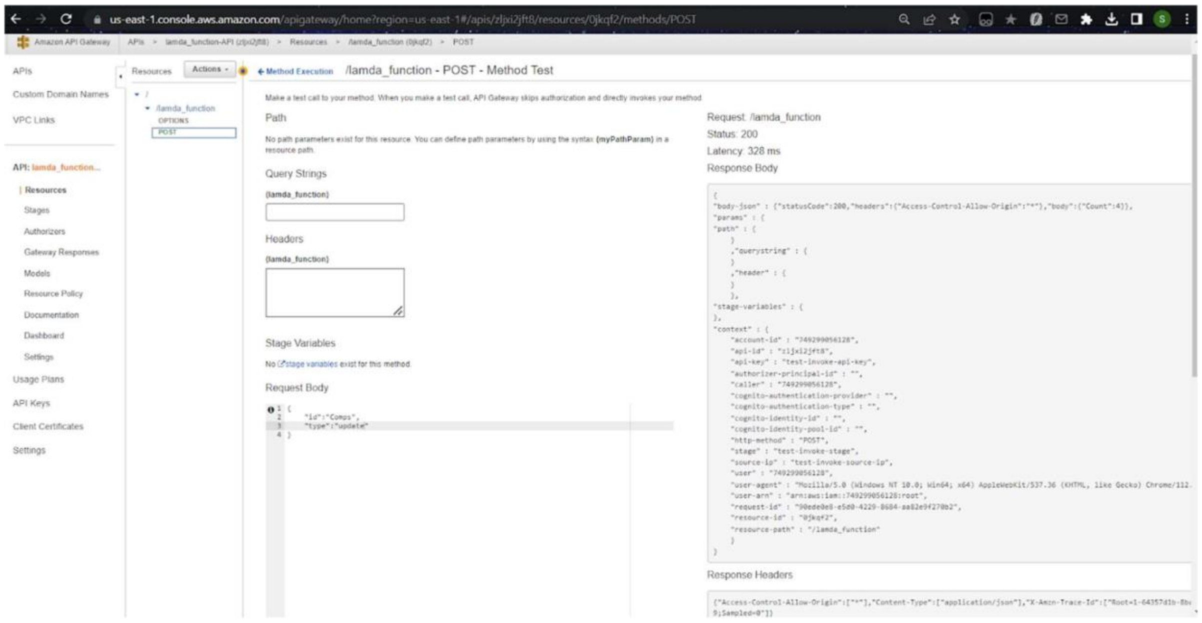
us-east-1.console.aws.amazon.com/apigateway/home?region=us-east-1#/apis/zljxi2jft8/resources/0jkqf2/methods/POST

Response Headers

```
{"Access-Control-Allow-Origin":["*"],"Content-Type":["application/json"],"X-Amzn-Trace-Id":["Root=1-64357b9c-ec229d2ec189328cbea81de
a;Sampled=0"]}
```

Logs

```
Execution log for request eab2ffaa-ee69-42ae-990f-fdb53ed2f7a1
Tue Apr 11 15:24:12 UTC 2023 : Starting execution for request: eab2ffaa-ee69-42ae-990f-fdb53ed2f7a1
Tue Apr 11 15:24:12 UTC 2023 : HTTP Method: POST, Resource Path: /lamda_function
Tue Apr 11 15:24:12 UTC 2023 : Method request path: {}
Tue Apr 11 15:24:12 UTC 2023 : Method request query string: {}
Tue Apr 11 15:24:12 UTC 2023 : Method request headers: {}
Tue Apr 11 15:24:12 UTC 2023 : Method request body before transformations: {
    "id":"Comps",
    "type":"get"
}
Tue Apr 11 15:24:12 UTC 2023 : Endpoint request URI: https://lambda.us-east-1.amazonaws.com/2015-03-31/functions/arn:aws:lambda:us-ea
st-1:749299056128:function:lamda_function/invocations
Tue Apr 11 15:24:12 UTC 2023 : Endpoint request headers: {X-Amz-Date=20230411T152412Z, x-amzn-apigateway-api-id=zljxi2jft8, Accept=ap
plication/json, User-Agent=AmazonAPIGateway_zljxi2jft8, Host=lambda.us-east-1.amazonaws.com, X-Amz-Content-Sha256=32b6a59d9fb3cec7fff
e8d7ed5f483411341e1aa65d64477c3116b57933e1136, X-Amzn-Trace-Id=Root=1-64357b9c-ec229d2ec189328cbea81dea, x-amzn-lambda-integration-ta
g=eab2ffaa-ee69-42ae-990f-fdb53ed2f7a1, Authorization=**********************************************************************************
**********************************************************************************************************************************
****************************************************************************************************************ecc639, X-Am
z-Source-Arn=arn:aws:execute-api:us-east-1:749299056128:zljxi2jft8/test-invoke-stage/POST/lamda_function, X-Amz-Security-Token=IQoJb3
JpZ2luX2VjEL//////////[illegible] [TRUNCAT
ED]
Tue Apr 11 15:24:12 UTC 2023 : Endpoint request body after transformations: {
    "id":"Comps",
    "type":"get"
}
Tue Apr 11 15:24:12 UTC 2023 : Sending request to https://lambda.us-east-1.amazonaws.com/2015-03-31/functions/arn:aws:lambda:us-east-
1:749299056128:function:lamda_function/invocations
Tue Apr 11 15:24:14 UTC 2023 : Received response. Status: 200, Integration latency: 1783 ms
Tue Apr 11 15:24:14 UTC 2023 : Endpoint response headers: {Date=Tue, 11 Apr 2023 15:24:14 GMT, Content-Type=application/json, Content
-Length=90, Connection=keep-alive, x-amzn-RequestId=41c063b4-7234-4aea-b4aa-f52030505000, x-amzn-Remapped-Content-Length=0, X-Amz-Exe
cuted-Version=$LATEST, X-Amzn-Trace-Id=root=1-64357b9c-ec229d2ec189328cbea81dea;sampled=0}
Tue Apr 11 15:24:14 UTC 2023 : Endpoint response body before transformations: {"statusCode": 200, "headers": {"Access-Control-Allow-O
rigin": "*"}, "body": {"Count": 2}}
Tue Apr 11 15:24:14 UTC 2023 : Method response body after transformations: {
"body-json" : {"statusCode":200,"headers":{"Access-Control-Allow-Origin":"*"},"body":{"Count":2}},
"params" : {
"path" : {
    }
    ,"querystring" : {
    }
    ,"header" : {
    }
    },
"stage-variables" : {
},
"context" : {
    "account-id" : "749299056128",
    "api-id" : "zljxi2jft8",
    "api-key" : "test-invoke-api-key",
    "authorizer-principal-id" : "",
    "caller" : "749299056128",
    "cognito-authentication-provider" : "",
```



us-east-1.console.aws.amazon.com/apigateway/home?region=us-east-1#/apis/zljxi2jft8/resources/0jkqf2/methods/POST

/lamda_function - POST - Method Test

Make a test call to your method. When you make a test call, API Gateway skips authorization and directly invokes your method

Path

No path parameters exist for this resource. You can define path parameters by using the syntax {myPathParam} in a resource path.

Query Strings

{lamda_function}

Headers

{lamda_function}

Stage Variables

No stage variables exist for this method.

Request Body

```
{
    "id":"Comps",
    "type":"update"
}
```

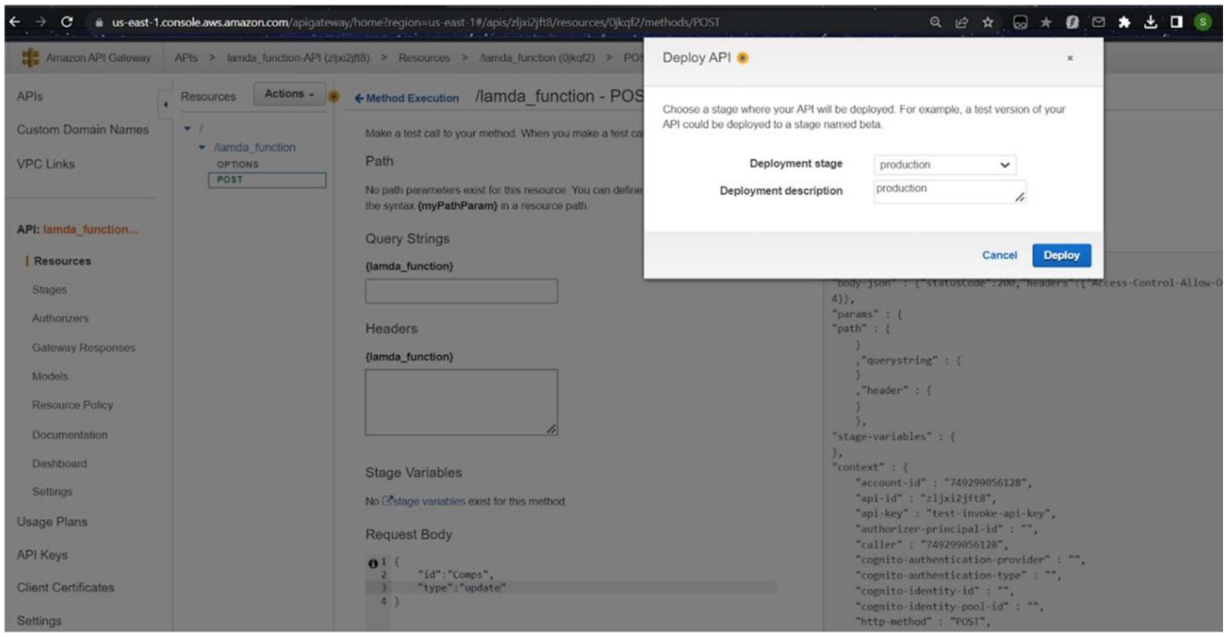
Request: /lamda_function
Status: 200
Latency: 328 ms
Response Body

```
{
"body-json" : {"statusCode":200,"headers":{"Access-Control-Allow-Origin":"*"},"body":{"Count":4}},
"params" : {
"path" : {
    }
    ,"querystring" : {
    }
    ,"header" : {
    }
    },
"stage-variables" : {
},
"context" : {
    "account-id" : "749299056128",
    "api-id" : "zljxi2jft8",
    "api-key" : "test-invoke-api-key",
    "authorizer-principal-id" : "",
    "caller" : "749299056128",
    "cognito-authentication-provider" : "",
    "cognito-authentication-type" : "",
    "cognito-identity-id" : "",
    "cognito-identity-pool-id" : "",
    "http-method" : "POST",
    "stage" : "test-invoke-stage",
    "source-ip" : "test-invoke-source-ip",
    "user" : "749299056128",
    "user-agent" : "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.
    "user-arn" : "arn:aws:iam::749299056128:root",
    "request-id" : "90ede0e8-e5d0-4229-8684-aa82e9f270b2",
    "resource-id" : "0jkqf2",
    "resource-path" : "/lamda_function"
    }
}
```

Response Headers

```
{"Access-Control-Allow-Origin":["*"],"Content-Type":["application/json"],"X-Amzn-Trace-Id":["Root=1-64357d1b-8b
9;Sampled=0"]}
```

r) Deploy the API. Make sure to copy the ‘Invoke url’ and paste it into the HTML code to link it to the website.



us-east-1.console.aws.amazon.com/apigateway/home?region=us-east-1#/apis/zljxi2jft8/resources/0jkqf2/methods/POST

Deploy API

Choose a stage where your API will be deployed. For example, a test version of your API could be deployed to a stage named beta.

Deployment stage: production

Deployment description: production

Cancel | Deploy

Make a test call to your method. When you make a test ca

Path

No path parameters exist for this resource. You can define the syntax {myPathParam} in a resource path.

Query Strings

{lamda_function}

Headers

{lamda_function}

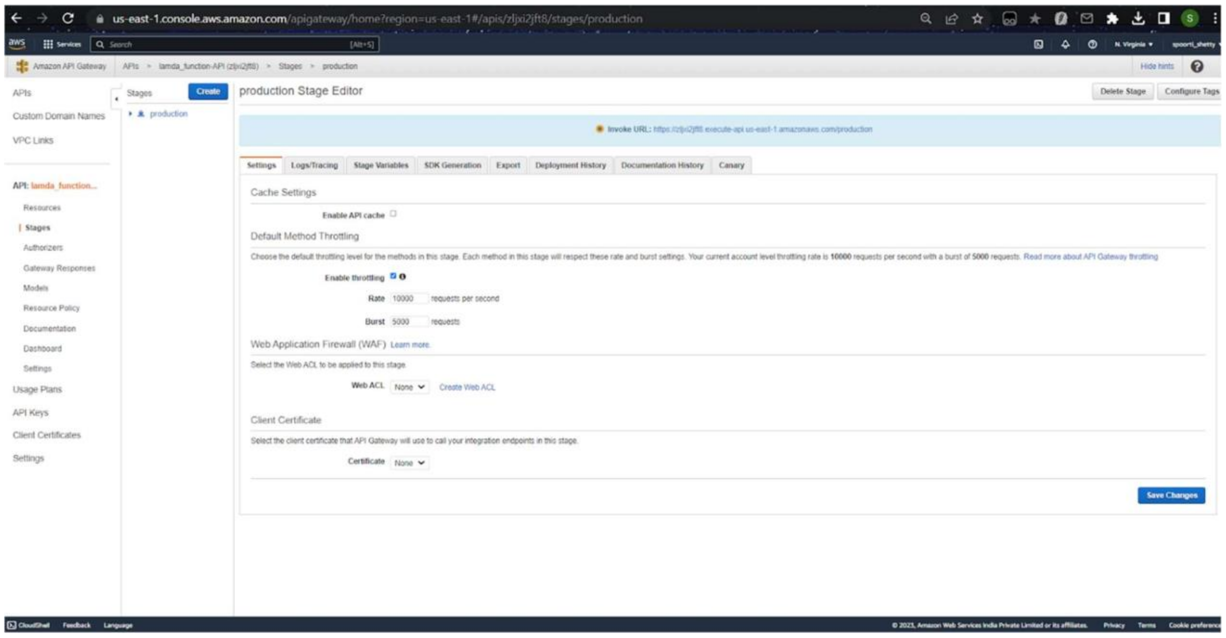
Stage Variables

No stage variables exist for this method.

Request Body

```
{
    "id":"Comps",
    "type":"update"
}
```

```
"body-json" : {"statusCode":200,"headers":{"Access-Control-Allow-O
4}},
"params" : {
"path" : {
    }
    ,"querystring" : {
    }
    ,"header" : {
    }
    },
"stage-variables" : {
},
"context" : {
    "account-id" : "749299056128",
    "api-id" : "zljxi2jft8",
    "api-key" : "test-invoke-api-key",
    "authorizer-principal-id" : "",
    "caller" : "749299056128",
    "cognito-authentication-provider" : "",
    "cognito-authentication-type" : "",
    "cognito-identity-id" : "",
    "cognito-identity-pool-id" : "",
    "http-method" : "POST",
```



us-east-1.console.aws.amazon.com/apigateway/home?region=us-east-1#/apis/zljxi2jft8/stages/production

production Stage Editor

Invoke URL: https://zljxi2jft8.execute-api.us-east-1.amazonaws.com/production

Cache Settings

Enable API cache ☐

Default Method Throttling

Choose the default throttling level for the methods in this stage. Each method in this stage will respect these rate and burst settings. Your current account level throttling rate is **10000** requests per second with a burst of **5000** requests. Read more about API Gateway throttling

Enable throttling ☑

Rate 10000 requests per second

Burst 5000 requests

Web Application Firewall (WAF) Learn more.

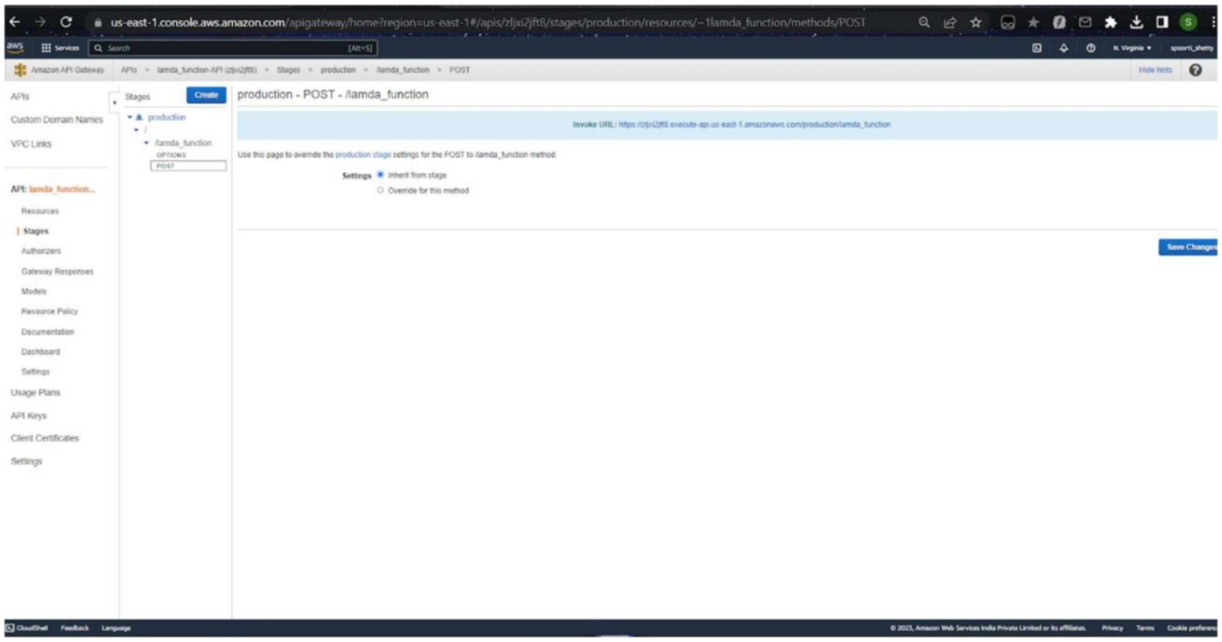
Select the Web ACL to be applied to this stage.

Web ACL: None   Create Web ACL

Client Certificate

Select the client certificate that API Gateway will use to call your integration endpoints in this stage.

Certificate: None


us-east-1.console.aws.amazon.com/apigateway/home?region=us-east-1#/apis/zljxi2jft8/stages/production/resources/~1lamda_function/methods/POST

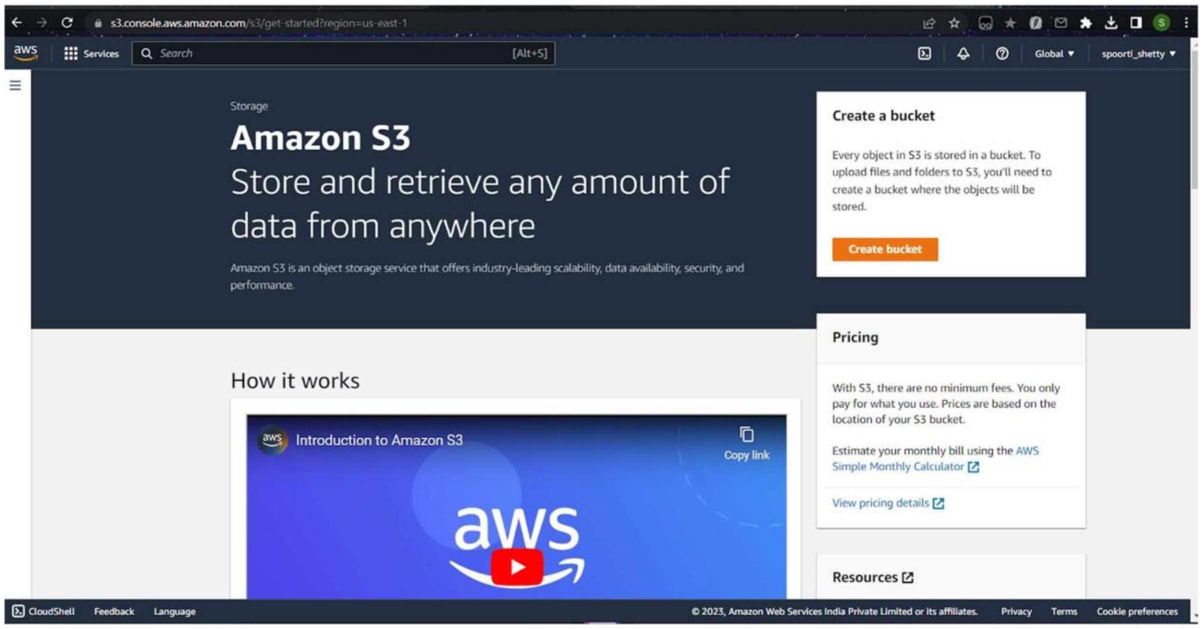
production - POST - /lamda_function

Invoke URL: https://zljxi2jft8.execute-api.us-east-1.amazonaws.com/production/lamda_function

Use this page to override the production stage settings for the POST to /lamda_function method.

Settings: Inherit from stage (selected) / Override for this method

s) Navigate to Amazon S3 and click on ‘Create bucket’.


s3.console.aws.amazon.com/s3/get-started?region=us-east-1

Storage

Amazon S3

Store and retrieve any amount of data from anywhere

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.

Create a bucket

Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.

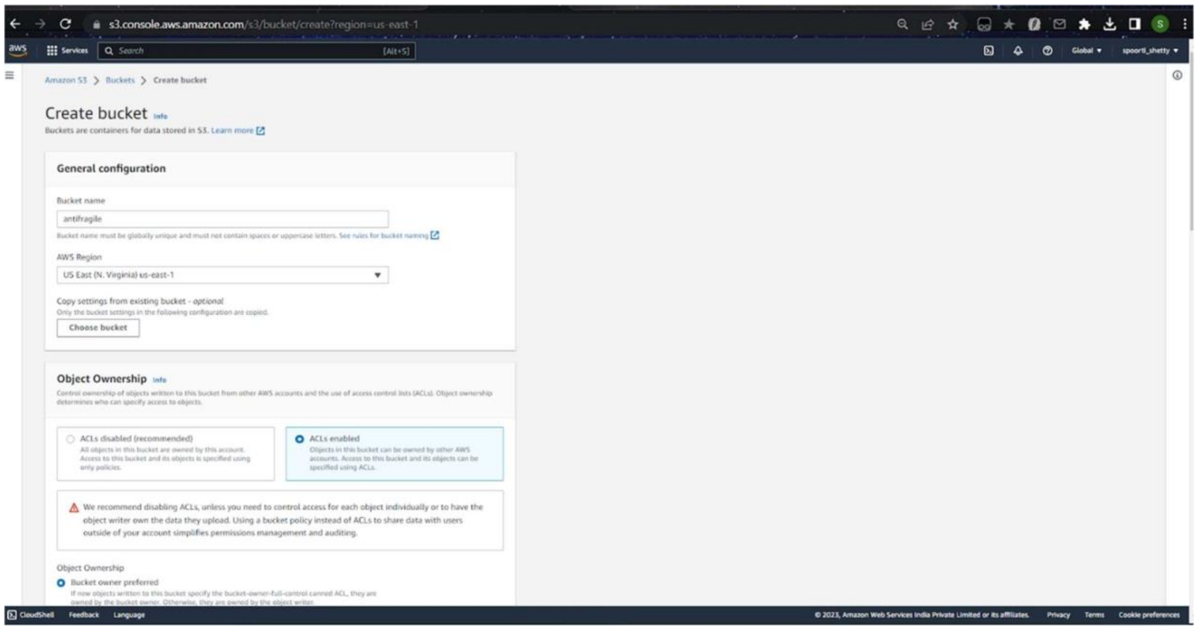
Pricing

With S3, there are no minimum fees. You only pay for what you use. Prices are based on the location of your S3 bucket.

Estimate your monthly bill using the AWS Simple Monthly Calculator

t) Set a bucket name along with the following configurations:

- ACLs enabled
- Uncheck ‘Block all public access’
- Enable bucket versioning



s3.console.aws.amazon.com/s3/bucket/create?region=us-east-1

Amazon S3 > Buckets > Create bucket

Create bucket Info

Buckets are containers for data stored in S3. Learn more

General configuration

Bucket name: antifragile

Bucket name must be globally unique and must not contain spaces or uppercase letters. See rules for bucket naming

AWS Region: US East (N. Virginia) us-east-1

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

ACLs disabled (recommended) [✓ ACLs enabled]

ACLs disabled (recommended): All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

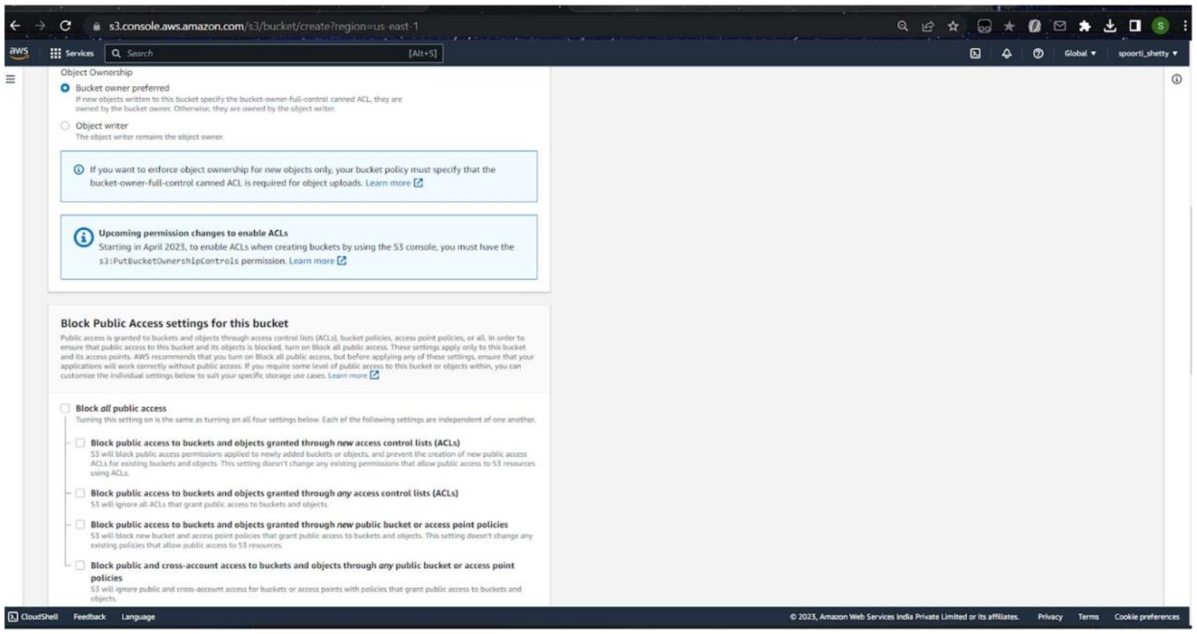
ACLs enabled: Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

We recommend disabling ACLs, unless you need to control access for each object individually or to have the object writer own the data they upload. Using a bucket policy instead of ACLs to share data with users outside of your account simplifies permissions management and auditing.

Object Ownership

[✓ Bucket owner preferred]

If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.



s3.console.aws.amazon.com/s3/bucket/create?region=us-east-1

Object Ownership

[✓ Bucket owner preferred] Object writer

Bucket owner preferred: If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.

Object writer: The object writer remains the object owner.

If you want to enforce object ownership for new objects only, your bucket policy must specify that the bucket-owner-full-control canned ACL is required for object uploads. Learn more

Upcoming permission changes to enable ACLs

Starting in April 2023, to enable ACLs when creating buckets by using the S3 console, you must have the `s3:PutBucketOwnershipControls` permission. Learn more

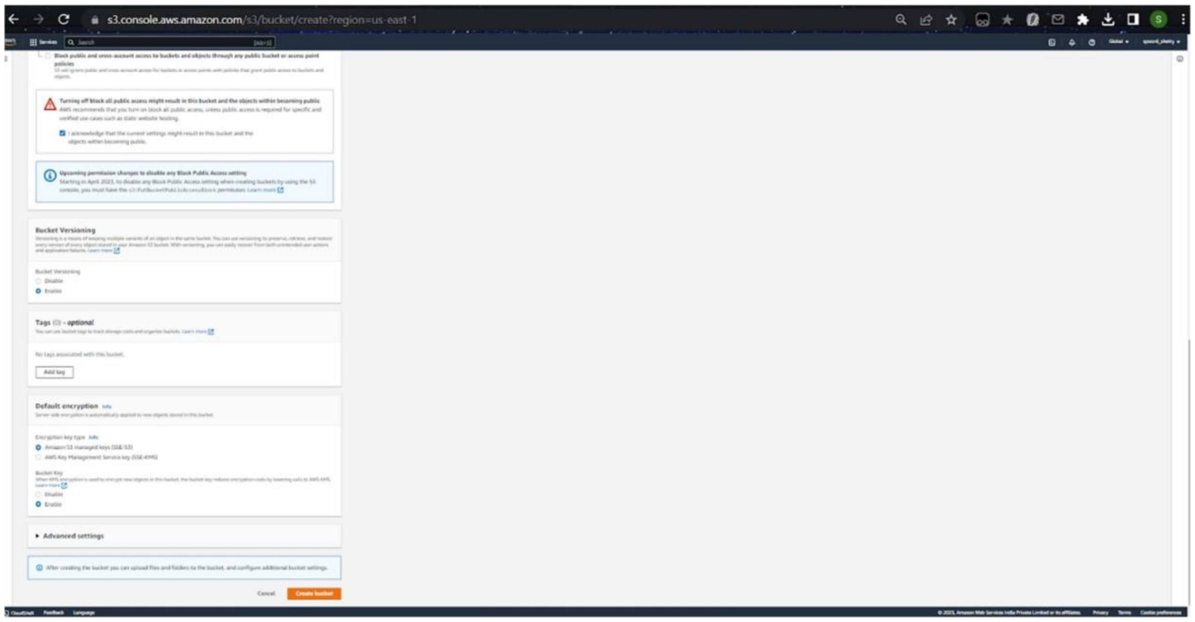
Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

☐ **Block *all* public access**

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- ☐ **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
  S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- ☐ **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
  S3 will ignore all ACLs that grant public access to buckets and objects.
- ☐ **Block public access to buckets and objects granted through *new* public bucket or access point policies**
  S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- ☐ **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
  S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.



s3.console.aws.amazon.com/s3/bucket/create?region=us-east-1

- [ ] Block public and cross-account access to buckets and objects through any public bucket or access point policies
  S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Turning off block all public access might result in this bucket and the objects within becoming public
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

- [x] I acknowledge that the current settings might result in this bucket and the objects within becoming public.

Upcoming permission changes to disable any Block Public Access setting
Starting in April 2023, to disable any Block Public Access setting when creating buckets by using the S3 console, you must have the s3:PutBucketPublicAccessBlock permission. Learn more

Bucket Versioning
Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. Learn more

Bucket Versioning
Disable [✓ Enable]

Tags (0) - *optional*
You can use bucket tags to track storage costs and organize buckets. Learn more

No tags associated with this bucket.

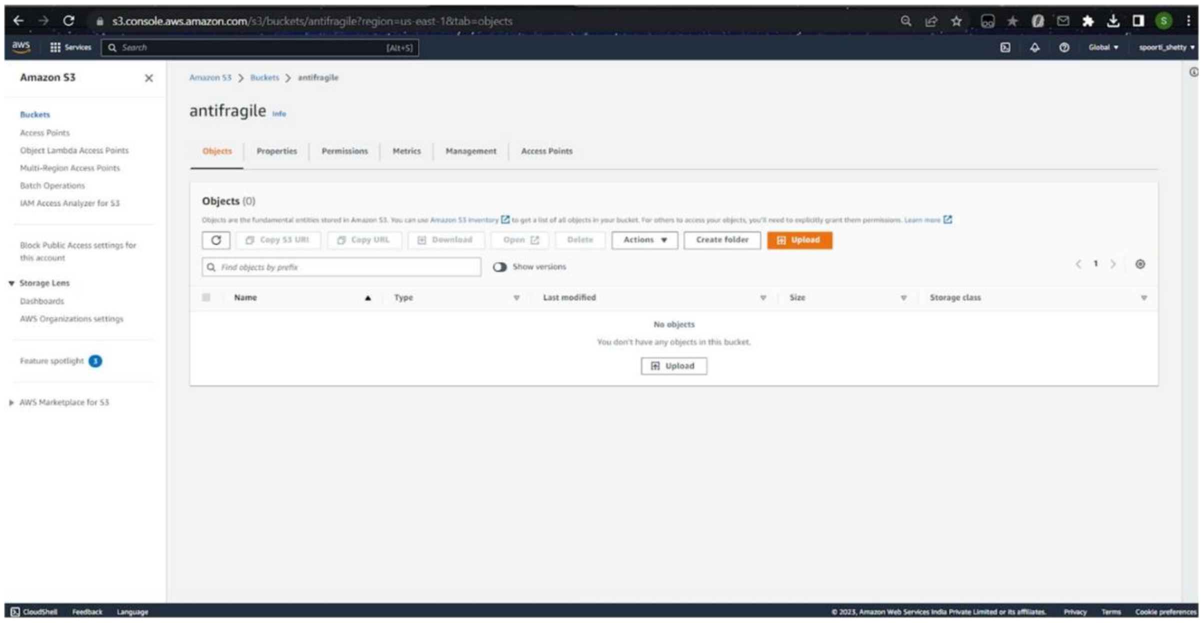
Default encryption Info
Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption key type Info
[✓ Amazon S3 managed keys (SSE-S3)] AWS Key Management Service key (SSE-KMS)

Bucket Key
When KMS encryption is used to encrypt new objects in this bucket, the bucket key reduces encryption costs by lowering calls to AWS KMS. Learn more
Disable [✓ Enable]

After creating the bucket you can upload files and folders to the bucket, and configure additional bucket settings.

u) Click on the newly created bucket and upload the HTML file. Make sure to grant public access.



s3.console.aws.amazon.com/s3/buckets/antifragile?region=us-east-1&tab=objects

Amazon S3 > Buckets > antifragile

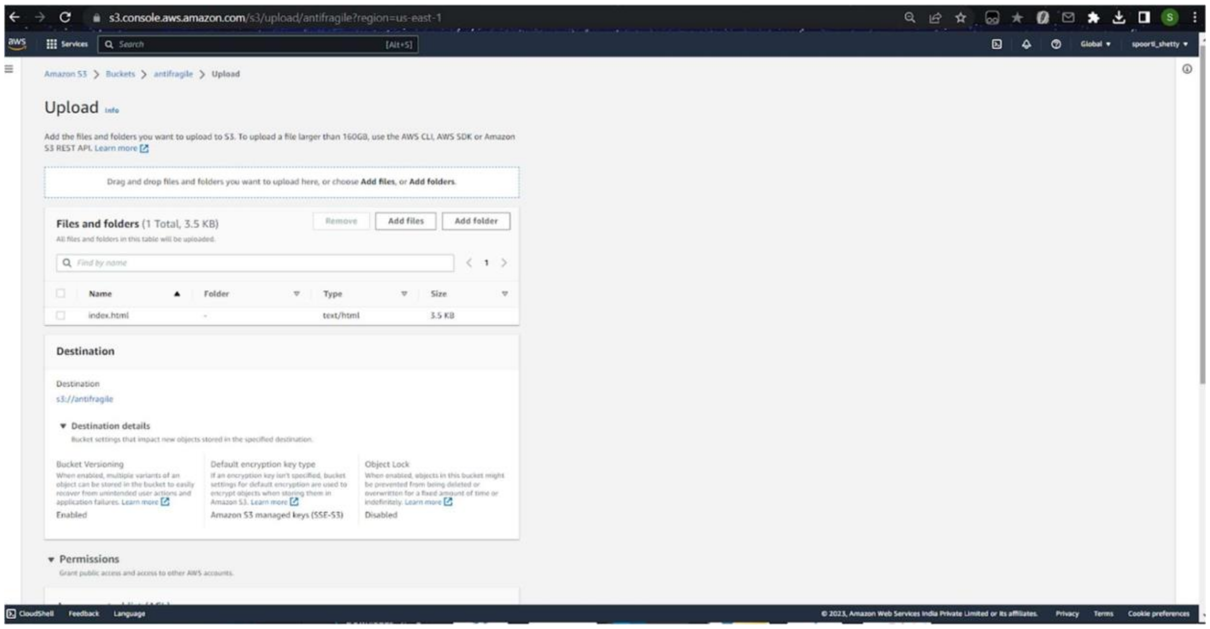
antifragile

Objects (0)
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified	Size	Storage class

No objects
You don't have any objects in this bucket.

© 2023, Amazon Web Services India Private Limited or its affiliates.


s3.console.aws.amazon.com/s3/upload/antifragile?region=us-east-1

Amazon S3 > Buckets > antifragile > Upload

Upload

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more

Drag and drop files and folders you want to upload here, or choose **Add files**, or **Add folders**.

Files and folders (1 Total, 3.5 KB)

All files and folders in this table will be uploaded.

Name	Folder	Type	Size
index.html	-	text/html	3.5 KB

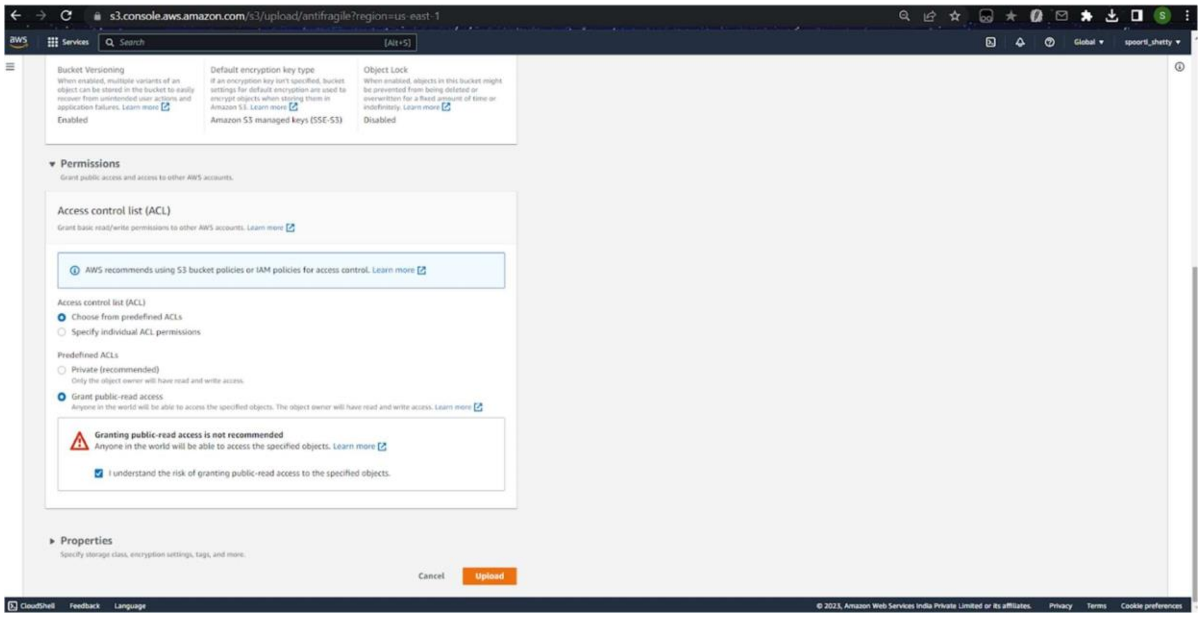
Destination

Destination
s3://antifragile

Destination details
Bucket settings that impact new objects stored in the specified destination.

Bucket Versioning	Default encryption key type	Object Lock
When enabled, multiple variants of an object can be stored in the bucket to easily recover from unintended user actions and application failures. Learn more	If an encryption key isn't specified, bucket settings for default encryption are used to encrypt objects when storing them in Amazon S3. Learn more	When enabled, objects in this bucket might be prevented from being deleted or overwritten for a fixed amount of time or indefinitely. Learn more
Enabled	Amazon S3 managed keys (SSE-S3)	Disabled

Permissions
Grant public access and access to other AWS accounts.


s3.console.aws.amazon.com/s3/upload/antifragile?region=us-east-1

Bucket Versioning	Default encryption key type	Object Lock
When enabled, multiple variants of an object can be stored in the bucket to easily recover from unintended user actions and application failures. Learn more	If an encryption key isn't specified, bucket settings for default encryption are used to encrypt objects when storing them in Amazon S3. Learn more	When enabled, objects in this bucket might be prevented from being deleted or overwritten for a fixed amount of time or indefinitely. Learn more
Enabled	Amazon S3 managed keys (SSE-S3)	Disabled

Permissions
Grant public access and access to other AWS accounts.

Access control list (ACL)
Grant basic read/write permissions to other AWS accounts. Learn more

AWS recommends using S3 bucket policies or IAM policies for access control. Learn more

Access control list (ACL)
- [✓ Choose from predefined ACLs]
- Specify individual ACL permissions

Predefined ACLs
- Private (recommended)
  Only the object owner will have read and write access.
- [✓ Grant public-read access]
  Anyone in the world will be able to access the specified objects. The object owner will have read and write access. Learn more

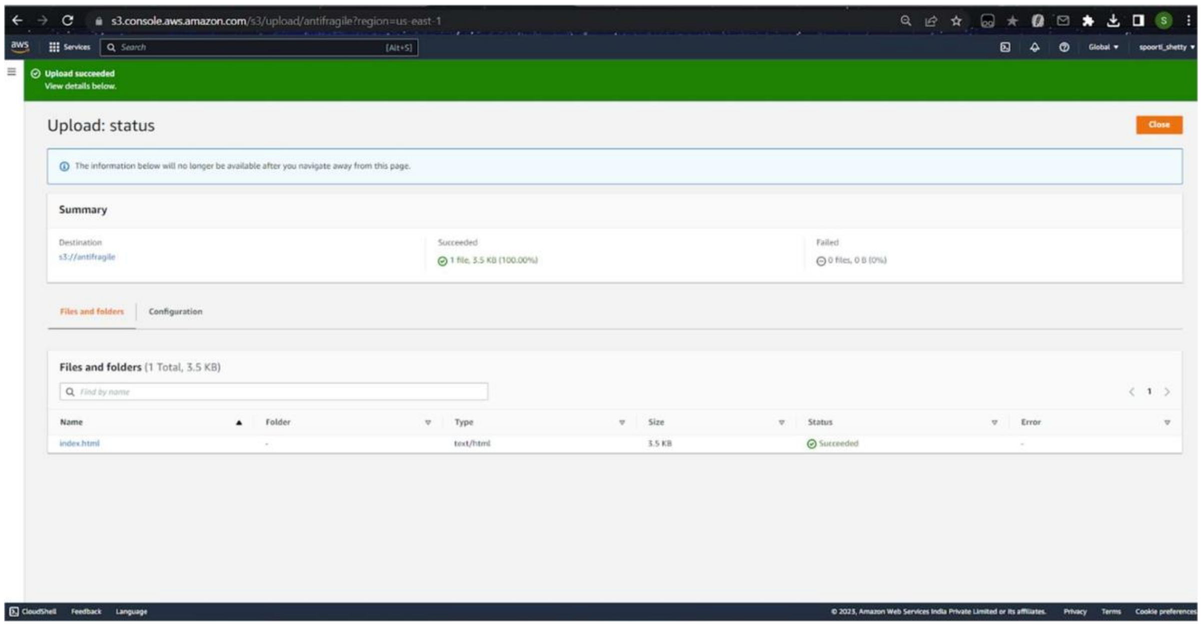
Granting public-read access is not recommended
Anyone in the world will be able to access the specified objects. Learn more

[✓] I understand the risk of granting public-read access to the specified objects.

Properties
Specify storage class, encryption settings, tags, and more.

Cancel | Upload

v) The file has been uploaded successfully. Now, click on the file.



Upload succeeded
View details below.

Upload: status

The information below will no longer be available after you navigate away from this page.

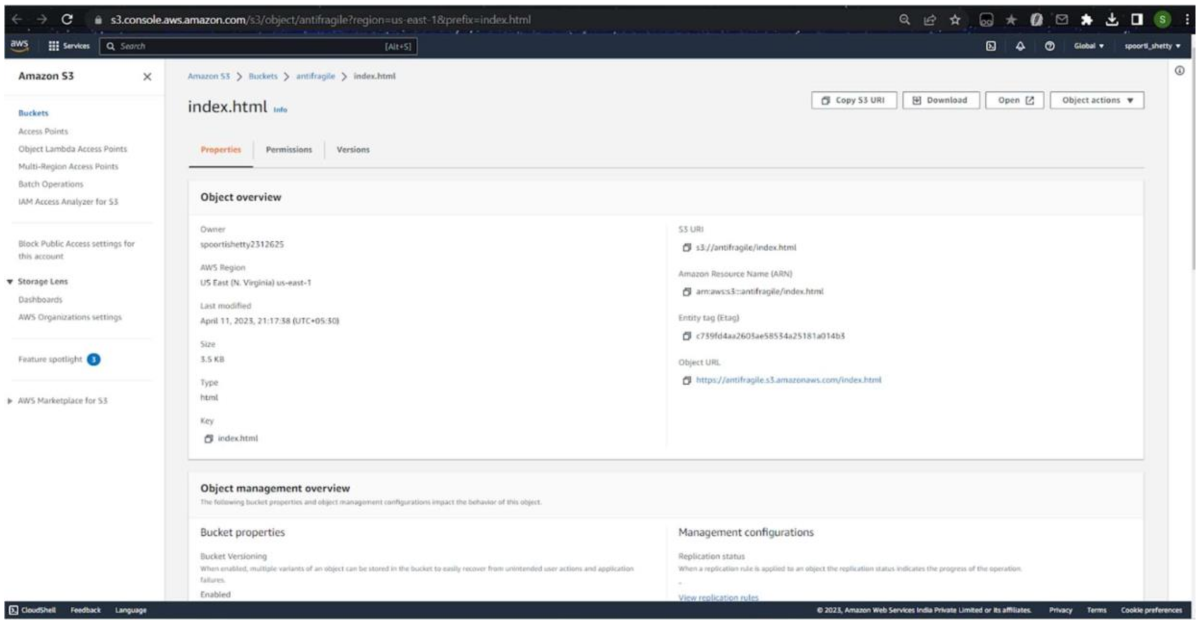
Summary

Destination	Succeeded	Failed
s3://antifragile	1 file, 3.5 KB (100.00%)	0 files, 0 B (0%)

Files and folders (1 Total, 3.5 KB)

Name	Folder	Type	Size	Status	Error
index.html	-	text/html	3.5 KB	Succeeded	-

w) Click on the ‘Object URL’ to view the website.



Amazon S3 > Buckets > antifragile > index.html

index.html

Object overview

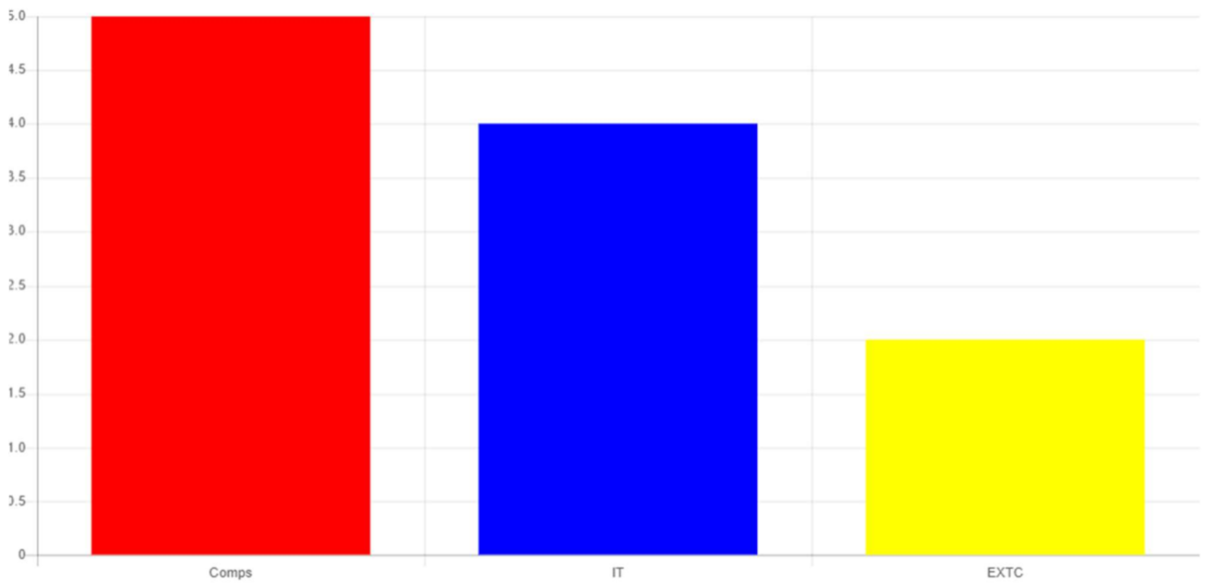
Owner	spoortishetty2312625
AWS Region	US East (N. Virginia) us-east-1
Last modified	April 11, 2023, 21:17:38 (UTC+05:30)
Size	3.5 KB
Type	html
Key	index.html
S3 URI	s3://antifragile/index.html
Amazon Resource Name (ARN)	arn:aws:s3:::antifragile/index.html
Entity tag (Etag)	c739fd4aa2605ae58534a25181a014b3
Object URL	https://antifragile.s3.amazonaws.com/index.html

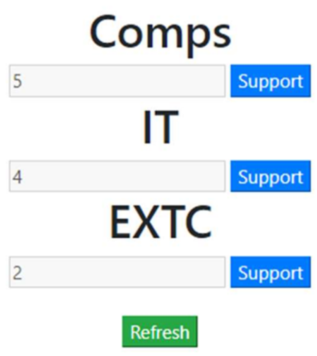
Object management overview
The following bucket properties and object management configurations impact the behavior of this object.

Bucket properties
Bucket Versioning
When enabled, multiple variants of an object can be stored in the bucket to easily recover from unintended user actions and application failures.
Enabled

Management configurations
Replication status
When a replication rule is applied to an object the replication status indicates the progress of the operation.
-
View replication rules

x) The website is live and available for public access.


Comps
IT
EXTC


Comps
5
Support
IT
4
Support
EXTC
2
Support
Refresh