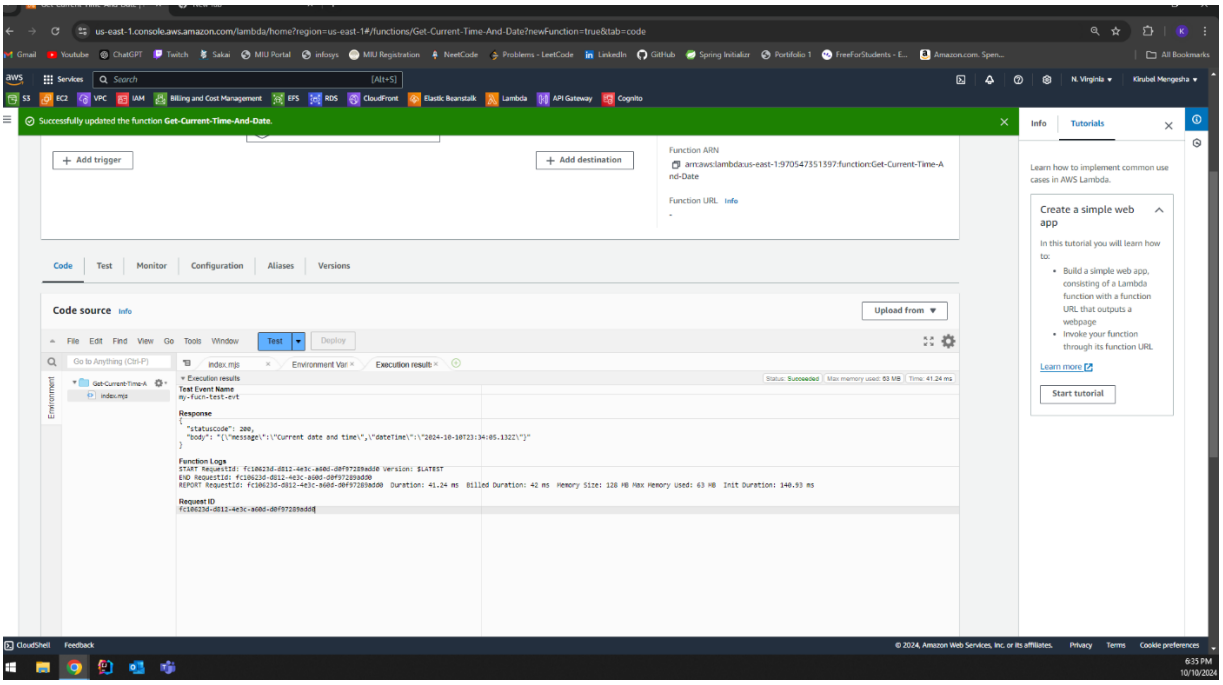
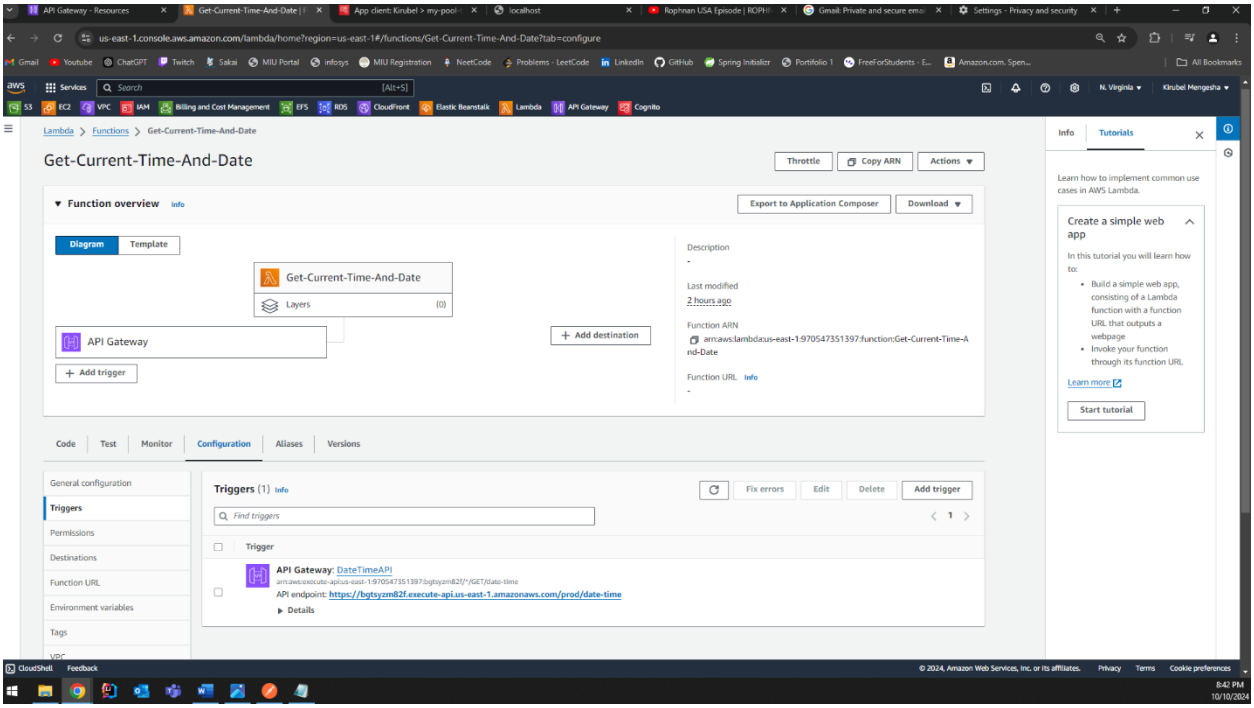
Task 1: Create and Invoke a Simple AWS Lambda Function

* Create a simple AWS Lambda function using the AWS Management Console. The function can be written in Node.js and should perform a basic task (e.g., returning the current date and time) as the below:

```JavaScript
exports.Handler = async (event) => {
const response = {
statusCode: 200,
body: JSON.stringify({
message: 'Current date and time',
dateTime: new Date().toISOString()
}),
};
return response;
};
```

* Test the function by invoking it manually from the AWS Management Console.





* Set up an API Gateway to trigger the Lambda function via HTTP requests.

Task 2: Set Up API Gateway

1. Create an API:

* Go to the Amazon API Gateway console.

* Click on “Create API”.

* Select “REST API” and click “Build” under the REST API (not private).

* Choose “New API”, give it a name (e.g., DateTimeAPI), and a description if desired. Then click “Create API”.

2. Create a Resource:

* In the newly created API, under the "Actions" dropdown, select “Create Resource”.

* Enter a resource name and path (e.g., /date-time).

* Click "Create Resource".

3. Create a Method:

* With the new resource selected, under the "Actions" dropdown, select “Create Method”.

* Choose “GET” and click the check mark.

* For the integration type, select “Lambda Function”.

* Check “Use Lambda Proxy integration”.

* Select the region where your Lambda function is located and type the name of your Lambda function (GetCurrentDateTime).

* Click "Save" and then “OK” to give API Gateway permissions to invoke your Lambda function.

4. Deploy the API:

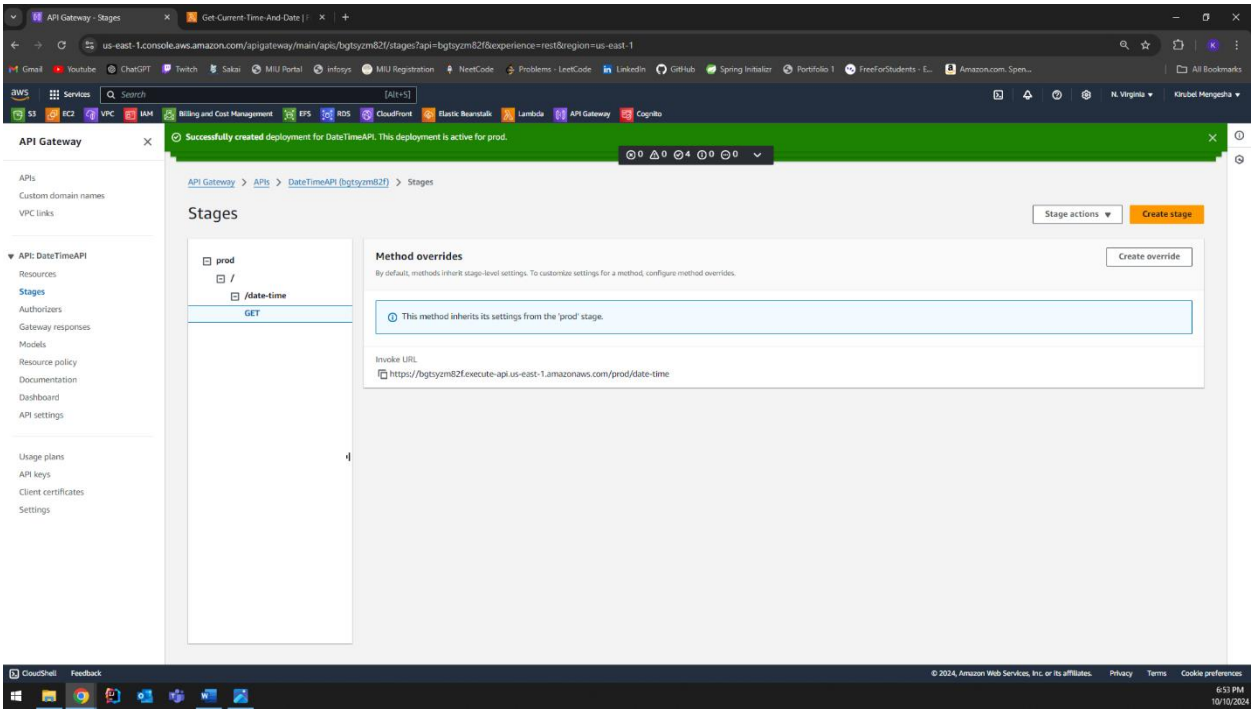
Under the "Actions" dropdown, select “Deploy API”.

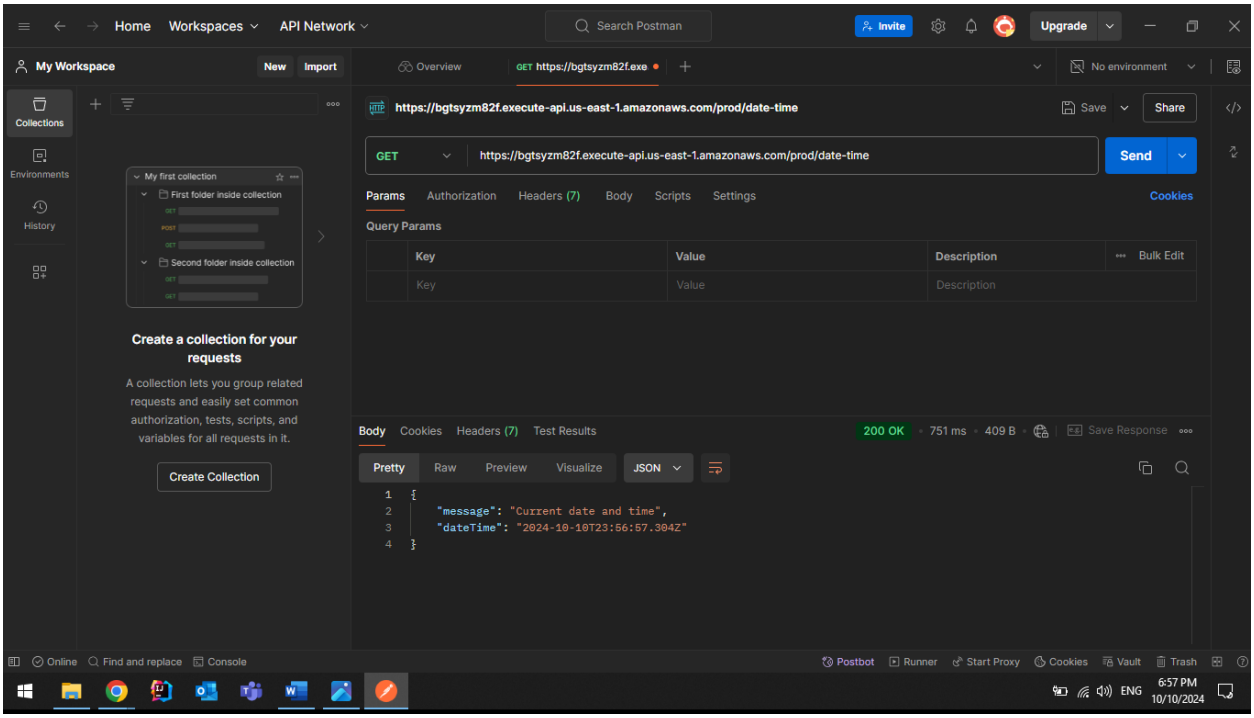
Create a new deployment stage (e.g., prod) and click "Deploy".

5. Test Your API:

* Navigate to the Stages section of your API Gateway, select the stage you deployed to (e.g., prod), and you will see an Invoke URL.

* Access the Invoke URL appended with your resource path (e.g., https://your-api-id.execute-api.region.amazonaws.com/prod/date-time) in a browser or via a tool like Postman to test the API.





Task 3: Integrate Amazon Cognito with API Gateway

1. Create a Cognito User pool for the app. The main configs are, enabling the hosted UI to get tokens and selecting IMPLICIT GRANT. Step by step:

* Step 1: Select Email.

* Step 2: No MFA.

* Step 3: Nothing to do.

* Step 4: Select Send email with Cognito.

* Step 5: Configure multiple things.

- Give the user pool name.

- Select Use the Cognito Hosted UI.

- Give any domain prefix and give the app client name.

- Give http://localhost:3000 in callback URLs for testing.

- In Advanced app client settings: Implicit Grant to replace OAuth 2.0.

* Step 6: Create the user pool

* To Create a new user: Go into this user pool and create a user using email and password.

* To generate the token: Go to the user pool -> App Integration -> App Client -> View Hosted UI. Enter the registered email and password, Cognito will return the callbackURL containing the token.

2. Secure the API using tokens from the Cognito User pool.

* Go to Authorizers in the API Gateway

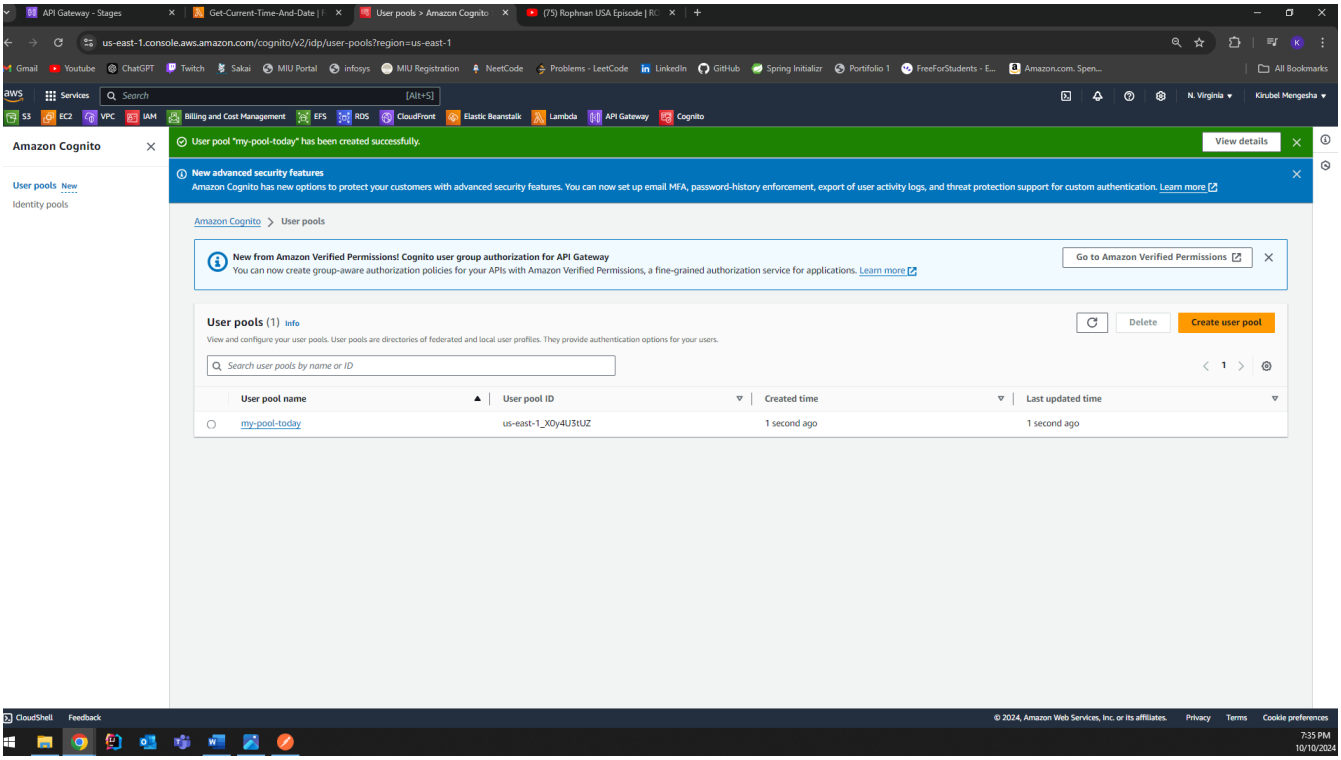
* Give it a name. Select Cognito as Authorizer type. Select the user pool. The token source is “Authorization”.

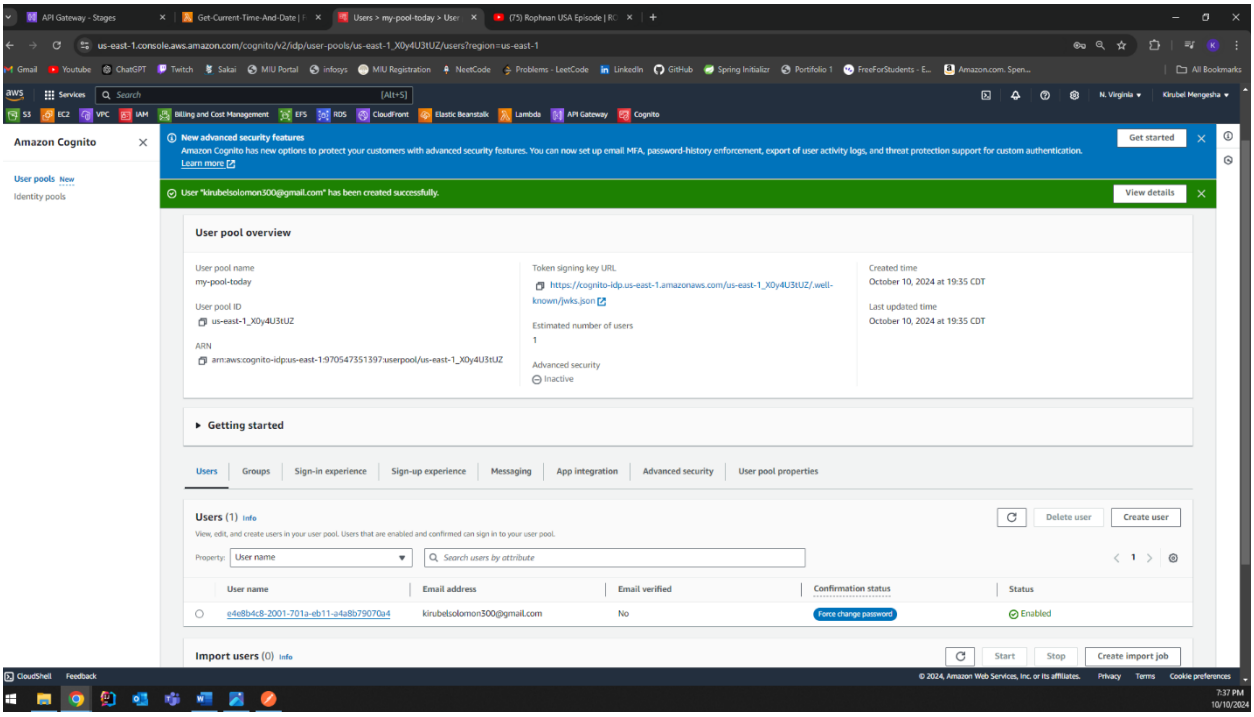
* Go to the resources. Click on the method and click on the “Method request”. Click “Edit” in the Method request settings. In the Authorization select the authorizer. Deploy the API.

3. Test the API

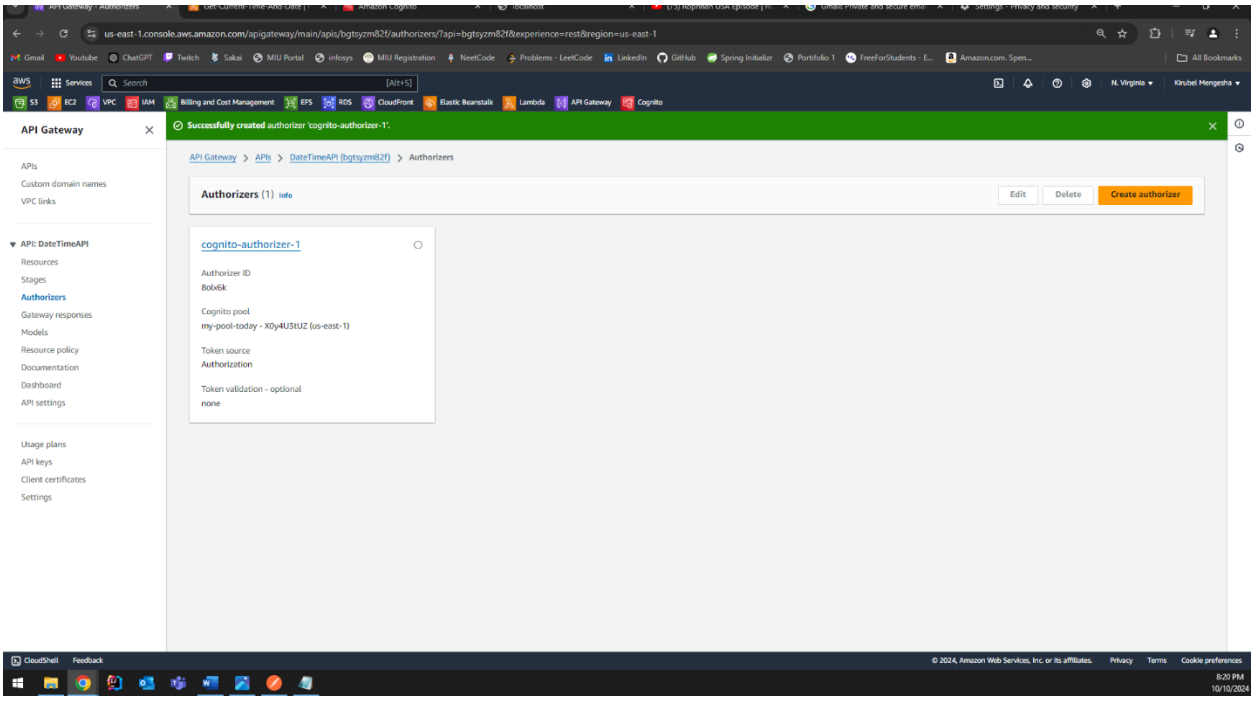
* Go to the Cognito pool and click on “App integration”. Scroll all the way down and click on the app client. Click on the “View Hosted UI” in the hosted UI section. Sign up and enter the code in your email. It will redirect you to the callback URL you entered and, in the URL, you will see `id_token` as a hash parameter.

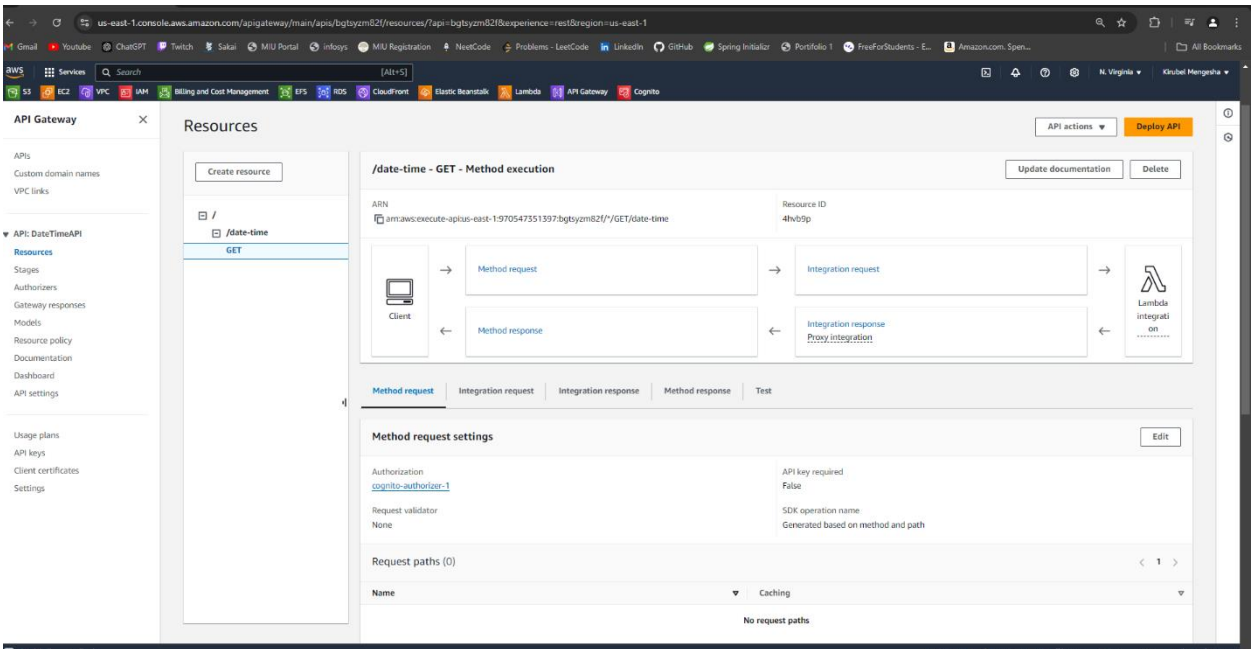
* Copy the ID Token. Provide it in the header as Authorization.

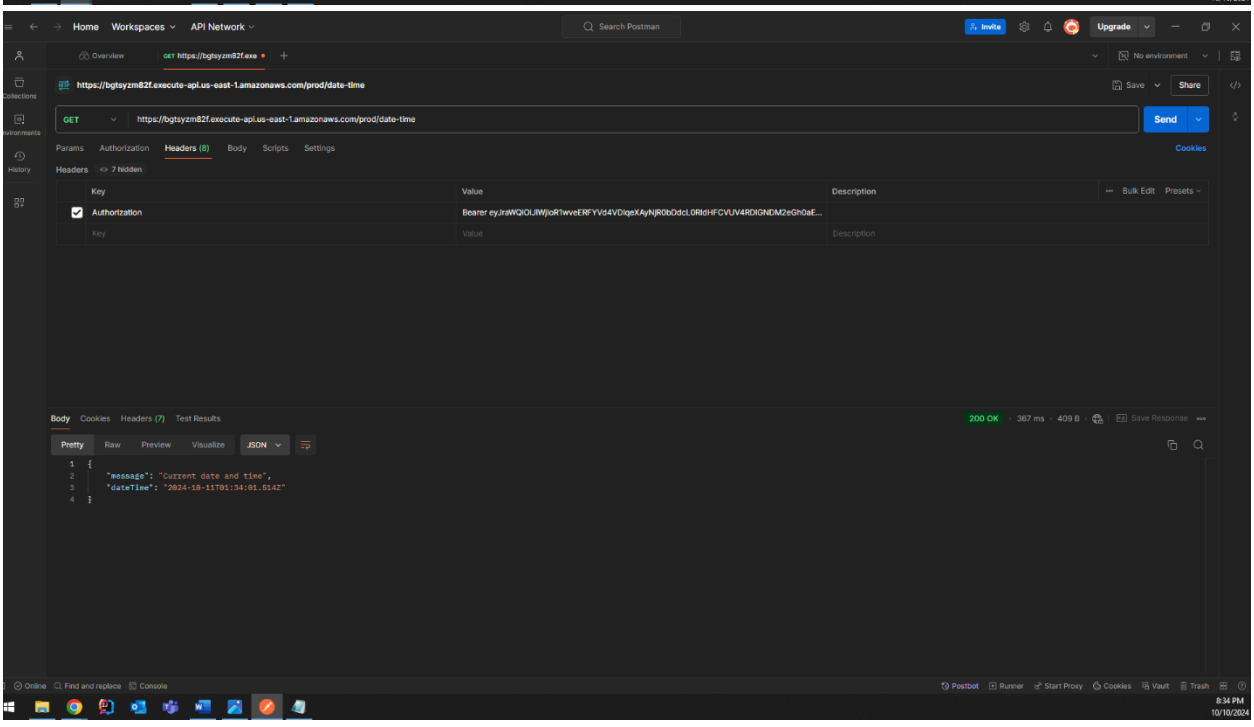



Amazon Cognito
User pools
Identity pools
New advanced security features
Amazon Cognito has new options to protect your customers with advanced security features. You can now set up email MFA, password-history enforcement, export of user activity logs, and threat protection support for custom authentication.
Learn more
Get started
User "kirubelsolomon300@gmail.com" has been created successfully.
View details
User pool overview
User pool name
my-pool-today
User pool ID
us-east-1_XOy4U3tUZ
ARN
arn:aws:cognito-idp:us-east-1:970547351397:userpool/us-east-1_XOy4U3tUZ
Token signing key URL
https://cognito-idp.us-east-1.amazonaws.com/us-east-1_XOy4U3tUZ/.well-known/jwks.json
Estimated number of users
1
Advanced security
Inactive
Created time
October 10, 2024 at 19:35 CDT
Last updated time
October 10, 2024 at 19:35 CDT
Getting started
Users
Groups
Sign-in experience
Sign-up experience
Messaging
App integration
Advanced security
User pool properties
Users (1) Info
View, edit, and create users in your user pool. Users that are enabled and confirmed can sign in to your user pool.
Delete user
Create user
Property: User name
Search users by attribute
User name
Email address
Email verified
Confirmation status
Status
e4e8b4c8-2001-701a-eb11-a4a8b79070a4
kirubelsolomon300@gmail.com
No
Force change password
Enabled
Import users (0) Info
Start
Stop
Create import job
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
7:37 PM
10/10/2024


localhost:3000/#id_token=eyJraWQiOiJlWjloR1weveRFYVd4VDlqeXAyNjR0bDdcL0RldHFCVUV4RDlGNDM2eGh0aEU9IiwiYWxnIjoiUlMyNTYifQ.eyJhdF9oYXNoIjoiLThsS0FITU5DTzBaSUliLW94b3gzUSIsInN1YiI6ImU0ZThiNGM4LTIwMDEtNzAxYS1lYj...


API Gateway
APIs
Custom domain names
VPC links
API: DateTimeAPI
Resources
Stages
Authorizers
Gateway responses
Models
Resource policy
Documentation
Dashboard
API settings
Usage plans
API keys
Client certificates
Settings
Successfully created authorizer 'cognito-authorizer-1'.
API Gateway > APIs > DateTimeAPI (bgtsyzm82f) > Authorizers
Authorizers (1) Info
Edit
Delete
Create authorizer
cognito-authorizer-1
Authorizer ID
8olx6k
Cognito pool
my-pool-today - XOy4U3tUZ (us-east-1)
Token source
Authorization
Token validation - optional
none
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
8:20 PM
10/10/2024


API Gateway
Resources
API actions
Deploy API
Create resource
/date-time - GET - Method execution
Update documentation
Delete
ARN
arn:aws:execute-api:us-east-1:970547351397:bgtsyzm82f/*/GET/date-time
Resource ID
4hvb9p
Client
Method request
Integration request
Method response
Integration response
Proxy integration
Lambda integration
Method request
Integration request
Integration response
Method response
Test
Method request settings
Edit
Authorization
cognito-authorizer-1
API key required
False
Request validator
None
SDK operation name
Generated based on method and path
Request paths (0)
Name
Caching
No request paths


Home
Workspaces
API Network
Search Postman
https://bgtsyzm82f.execute-api.us-east-1.amazonaws.com/prod/date-time
GET
https://bgtsyzm82f.execute-api.us-east-1.amazonaws.com/prod/date-time
Send
Params
Authorization
Headers (8)
Body
Scripts
Settings
Key
Value
Description
Authorization
Bearer ey.IraWQiOiJIWjJoRTwveERFYVd4VDIqeXAyNjR0bDdcL0RIdHFCVUV4RDlGNDM2eGh0aE...
Body
Cookies
Headers (7)
Test Results
200 OK
367 ms
409 B
Pretty
Raw
Preview
Visualize
JSON
{
"message": "Current date and time",
"dateTime": "2024-10-11T01:34:01.514Z"
}