# **SECURE API ACCESS USING**

# **AMAZON COGNITO** Icon Description automatically generated

Hello! folks its awesome to have you here working towards understanding how to secure your APIs using **COGNITO**. So, let’s get started right away with this tutorial.

Objective: Is To provide secure access to your application users, you define who can access the application resources and what kind of access can be granted. Access is based on identity controls that can confirm authentication (AuthN) and authorization (AuthZ), which are different concepts.

According to Wikipedia: In simple terms

* *Authorization:* *is the process of verifying that “you are permitted to do what you are trying to do.”*
* *Authentication:  is the process of verifying that “you are who you say you are,”*

To know details regarding Amazon Cognito, click [ [here](https://docs.aws.amazon.com/cognito/latest/developerguide/what-is-amazon-cognito.html) ], If you have basic understanding of what it is then let’s get to implementation.

# **Tutorial**: **Secure your API Gateway with Amazon Cognito User Pools**

to know more about **User Pools,** click [[here](https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-user-identity-pools.html)].

Login to your AWS account and then Choose Cognito in the search bar and then select **CREATE USER POOLS**

Give these settings

Graphical user interface, text, application, email

Description automatically generated

Let the default selections be as they are and in MFA just provide an option which you desire, I have given Authenticator apps

Graphical user interface, text, application

Description automatically generated

Choose Next and even in Configuring sign-up experience let default options be as they are. Now when you come into **Configure message delivery** In IAM role name give **Cognito-tutorial-role**

And In Email Section select **Send Email with Cognito**

Text

Description automatically generated

Choose **Next** in Integrate your App section give the User Pool Name and App client Name and then choose **Next.**

In **Review** section Create the User Pool. You’d have a similar screen

Graphical user interface, text, application, email

Description automatically generated

Choose the Switch back to Old Console and then you’d have a similar screen

Graphical user interface

Description automatically generated

Select your user pool and then in App integrations section go to **APP CLIENT SETTINGS**

We will enable our user pool to be used by the app and then we’ll provide the callback URL and signout URL.

Text, application

Description automatically generated with medium confidence

Then click on **Save Changes.**

Now click on Domain Name, for ease lets go with Cognito Domain, If you your own domain you can use it as well. For now, I’ll be going with Cognito Domain. Make sure you give a unique name or else when checking domain availability, it would state domain not available. You’d have a similar screen

Graphical user interface, text, application

Description automatically generated

Click on **Save Changes.** Now when you use that URL it would act as our login page which would ask our username and password.

Now go back to App Client Settings and in Hosted UI section before it used to show that we din’t have any hosted UI but now

A picture containing chart

Description automatically generated

**Launch Hosted UI.** Well now try sign up and give your username and password and register yourself once it’s done and enter then you’d have a similar screen.



Once its done you’d receive a mail from Cognito, similar like this

Graphical user interface, text, application

Description automatically generated

Once you put the code then you’d see the domain.

Graphical user interface, text, application

Description automatically generated

Now as you can see in URL there is a new information that got appended in the headers [code], this information in that encoded value would be later leveraged during invocation of API gateway endpoint and it will serve a JWT that authorises the request.

**NOW LET’S BUILD OUR REST API**

Search for API Gateway and build REST API and follow the usual procedure like creating the resources and the methods.

You’d have a similar screen

Graphical user interface, application

Description automatically generated

Now go to Authorisers section and create

Graphical user interface, text, application

Description automatically generated

Now take a look at the **Token Source** and hover over it you might find some information written so basically what it means is when you are doing this from the web app perspective, you set up a user and then login the user ,then you get a token and then you provide that token to API Gateway in the headers and behind the scenes API Gateway is going to call Cognito and makes sure that the person is legit and the token is valid an has permissions if so then it will forward the request to Lambda function to return the information.

Text

Description automatically generated

After creating the authorizer, now launch the hosted UI and this time in the URL change the **response\_type** from **response\_type=code to response\_type=token.** After a successful sign-in, Amazon Cognito returns user pool tokens to your web browser's address bar.

https://<your\_domain>/login?**response\_type=code**&client\_id=<your\_app\_client\_id>&redirect\_uri=<your\_callback\_url>

https://<your\_domain>/login?**response\_type=token**&client\_id=<your\_app\_client\_id>&redirect\_uri=<your\_callback\_url>

Now after you login you would find a ginormous header copy that from the URL

You’d have a similar info

id\_token=eyJraWQiOiJDalNJUWVBN1wvNTZreUt4YWNyXC83dzgyalJxR1hkdHlUVjFPYVFNWks0elk9IiwiYWxnIjoiUlMyNTYifQ..lIWB9cDobYjY47bM0SEZ3cEgSWxQlLwZPxHaQ1EPNAomP4qImjImPJMWWqNmfyFq58EruWBlQdJctkEpZaq1ueo8td\_nwvpIYklq9QYK5bVfrYuP6uqF0SJjYvyBKiig8v2dMeCOB1JrNR9XqC1Q31PxAqh3Mqnr0VOVUxOSbUXJGDx7QCejpcIQRyx2iwSE6W7OJZUNK5Bs8CPHYQnpUG9xlJoa6RrxKh8pcUFLU8RMTFhHvThXeTb9ih86LYzyCOHlCxSEE9mPA5xIcEiAer\_dhkOWUNnx8NisugO004Z7BYYvk5z57QpiO7T6RwAVqghLZjvtyyfpnaZmBptpMg

&access\_token=eyJraWQiOiIrNmVkUE5wTU9CQVhCeitNamdWSXNVNmJBbGlLRjhwUU1hU3I4UjNsVmM0PSIsImFsZyI6IlJTMjU2In0..c1ATkqmuvBs4WyXO0SOi67ItrMovcrj37KP9JGDD6ZVXRUuHPKeDSIy86Y77QRBYrrnbYhvLlsEVFT5VoSQelgZxjy4bpuwCDu1vX6U0DolyFbg53ydYlS7S0Hx5p00lF8PrOtxqL7W4UGLHuCqXzKyD2RSr3sXIMVOLssUGRNvSYHE\_8oh0RlqNo6BXvu-1CdeCYkGm960gx-IcI58vNCDACXPzFtwZu7EovXD0m-08JFq3QIvTYZG37wVOTY8oL2lAw26x6jqG3LnZ8Bje-tOpLC3mBKOU3VtHKyuMfLW5yBMq8ZwhAn6KQ9SgziqhBATASpPjN401CDhZjNnA6Q&expires\_in=3600&token\_type=Bearer

We would require the id\_token which is the upper information so grab that info and put it in the authorization. Go to the Authorizer that you created and choose test

Then a screen would pop up asking you for a value for the authorization header and so you paste the id\_token information into that and hit test, you’d have a similar screen

**Graphical user interface, text, application, email

Description automatically generated**

**Try :** putting some token value in the authorization header, you’d get a message stating **Unauthorized request**

**Now** lets go to the Resources and select the resource you created In our case its GET follow it up by going to **Method Request** you’ll find **Authorization**

Graphical user interface, text, application

Description automatically generated

**Note**: Please refresh the page and then try selecting the Authorization that’s when you’d find the Cognito user pool that you created. Once you have selected it the OAuth Scopes would show up, If you remember we have already given a lot of scopes such as phone, email, etc. So give any as you please.

Graphical user interface, text, application, chat or text message

Description automatically generated

Now go to Actions and Deploy the Api, Now our API is updated with the new authorization code.

Graphical user interface, application

Description automatically generated

Once this is done try invoking the invoke URL by appending the resource name.



Shape

Description automatically generated with medium confidence

**TEST YOUR API**

Use Postman to test your API

Add a new header Authorization and paste the **access\_token** value into it and hit the endpoint. You’d have a similar screen

Text

Description automatically generated

This marks the end of the tutorial, thank you for sticking till the end and finishing the tutorial, hope you understood the concepts of Cognito and how to implement and integrate with an API. **Happy Learning!!**

Resources:

* <https://docs.aws.amazon.com/>
* <https://aws.amazon.com/blogs/compute/secure-api-access-with-amazon-cognito-federated-identities-amazon-cognito-user-pools-and-amazon-api-gateway/>