

AWS Cognito Set-up

Duke TBI

AWS Cognito Username Setup

Run cognito_aws_account_setup.py script attached in the email.

Install flask and authlib packages using the terminal:

```
pip install Flask authlib
```

Run the script:

```
Python cognito_aws_account_setup.py
```

```
python cognito_aws_account_setup.py
```

This should open a localhost app. Click on the URL and this should be the window you see:

Welcome! Please [Login](#).

Upon clicking Login, you should get one of two screens

The image displays two side-by-side screenshots of the AWS Cognito user interface, separated by the word "OR".

The left screenshot shows the "Sign in" screen. It has a title "Sign in" and a subtitle "Sign in to your account." Below this is a "Username" label and an input field with the placeholder text "Enter username". At the bottom of the input field is a blue button labeled "Next". Below the button is a link that says "New user? Create an account".

The right screenshot shows the "Sign up" screen. It has a title "Sign up" and a subtitle "Create a new account." Below this are four input fields: "Username" (placeholder "Enter username"), "Email address" (placeholder "name@host.com"), "Password" (placeholder "Enter password"), and "Confirm password" (placeholder "Reenter password"). Below the "Confirm password" field is a checkbox labeled "Show password". At the bottom of the form is a blue button labeled "Sign up". Below the button is a link that says "Have an account already? Sign in".

If you get the left screen, then please click create an account and you should automatically reach the right screen.

Please create a username, use the email address of your AWS account, create a new password and press sign up.

Please remember the username and the password. That is what we will be using in the next steps. Upon signing up, you should have gotten this landing page. If you see this, you have successfully created a username and password!

Successfully signed in

This is the default redirect page for Amazon Cognito user pools.

You're seeing this page because your Amazon Cognito app client doesn't have a return URL set.

Anonymizer - AWS Integration

Changes to the Anonymizer app

We have to add a secret hash key into the Anonymizer python script for the authentication part. Hence,

Locate the Anonymizer package codebase: a folder named “anonymizer”

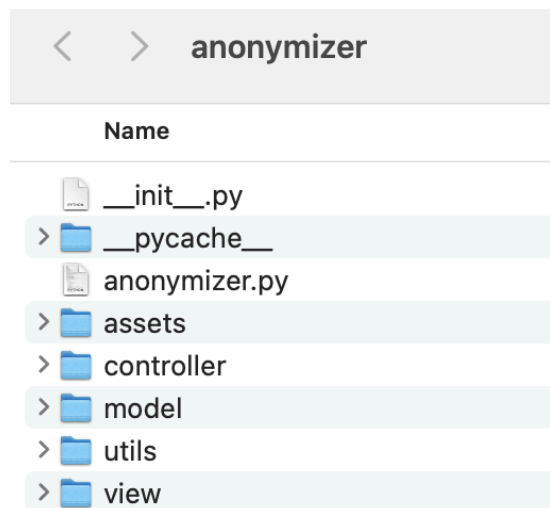
Usually in Windows, the folder would be located in these filepaths:

`C:\Users\<YourUserName>\.conda\pkgs` (conda installed)

Or

`C:\Program Files\Python\PythonXX\Lib\site-packages` (not conda installed)

The folder should have these files:



Open the controller folder and open the [project.py](#) file.

After all the import statements, add these new import statements and function:

```
import hmac
import hashlib
import base64
def get_secret_hash(username, client_id, client_secret):
    message = username + client_id
    dig = hmac.new(
        key=client_secret.encode('utf-8'),
        msg=message.encode('utf-8'),
        digestmod=hashlib.sha256
    ).digest()
    return base64.b64encode(dig).decode()
```

Find the `cognito_idp_client = boto3.client("cognito-idp",`
`region_name=self.model.aws_cognito.region_name)` line of code and right after it, add this
code chunk:

```
secret_hash = get_secret_hash(username=self.model.aws_cognito.username,  
    client_id=self.model.aws_cognito.app_client_id,  
    client_secret="XXXXXXXXXXXXXXXXXXXX")
```

Immediately after, you should see

```
response = cognito_idp_client.initiate_auth(
    ClientId=self.model.aws_cognito.app_client_id,
    AuthFlow="USER_PASSWORD_AUTH",
    AuthParameters={
        "USERNAME": self.model.aws_cognito.username,
        "PASSWORD": self.model.aws_cognito.password,
    },
)
```

Add the `secret_hash` as another parameter to `AuthParameters`, as shown below:

```
response = cognito_idp_client.initiate_auth(
    ClientId=self.model.aws_cognito.app_client_id,
    AuthFlow="USER_PASSWORD_AUTH",
    AuthParameters={
        "USERNAME": self.model.aws_cognito.username,
        "PASSWORD": self.model.aws_cognito.password,
        "SECRET_HASH":secret_hash,
    },
)
```

Find this line of code: `user_attribute_1 = response["UserAttributes"][0]`

And change it to `user_attribute_1 = response["UserAttributes"][2]`

Save the changes made to the script. Now load the Anonymizer GUI.

Anonymizer GUI

When creating a new project, go to project settings and Click on AWS Cognito Credentials

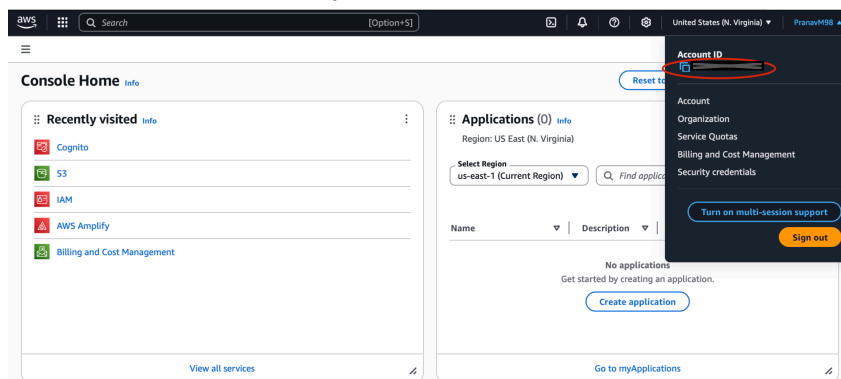


The screenshot shows the 'Project Settings' window with the following fields and buttons:

- Site ID: 972149
- Project Name: MY_PROJECT
- UID Root: 1.2.826.0.1.3680043.10.474.2
- Remove Pixel PHI: ☐
- DICOM Servers: Local Server, Query Server, Export Server
- AWS S3 Server: AWS Cognito Credentials <- Click Here
- Network Timeouts: Network Timeouts
- Storage Directory: .../pranavmanjunath/Documents/RSNA Anonymizer/MY_PROJECT
- Modalities: Select Modalities
- Storage Classes: Select Storage Classes
- Transfer Syntaxes: Select Transfer Syntaxes
- Script File: assets/scripts/default-anonymizer.script
- Logging Levels: Set Logging Levels
- Update Project button

And use these credentials:

AWS Account ID: Usually found here (circled):



Region Name: us-east-1

Cognito Application Client ID: [REDACTED]
Cognito User Pool ID: [REDACTED]
Cognito Identity Pool ID: [REDACTED]
S3 bucket: amplify-cognitoapp-dev-fc8c3-deployment
Username: Use username created in step 1
Password: Use password created in step 1

Check Export to AWS and press Ok

It should look like this:



The screenshot shows a macOS-style dialog box titled "AWS Cognito Credentials for Export to S3". It contains the following fields and controls:

- AWS Account ID:** A text field with a redacted value and a red prompt "Use your AWS Account ID".
- Region Name:** A text field containing "us-east-1".
- Cognito Application Client ID:** A text field with a redacted value.
- Cognito User Pool ID:** A text field with a redacted value.
- Cognito Identity Pool ID:** A text field with a redacted value.
- S3 Bucket:** A text field containing "amplify-cognitoapp-dev-fc8c3-deployment".
- S3 Prefix:** An empty text field.
- Username:** A text field containing "pranav.manjunath1" and a red prompt "Enter your username you created in the first step".
- Password:** A text field with masked characters "*****" and a red prompt "Enter your password you created in the first step".
- Export to AWS:** A checkbox that is checked with a blue checkmark.
- Ok:** A blue button at the bottom right.

Continue to upload the DICOM files and press export and it should work! 😊