■ Deploy OITH User Sync to AWS Lambda

This guide will help you deploy the user sync function to AWS so user profiles are automatically saved to the cloud.

Step 1: Create DynamoDB Table

1. Go to AWS DynamoDB Console
2. Click **"Create table"**
3. Fill in:
- **Table name:** `oith-users`
- **Partition key:** `pk` (String)
- **Sort key:** `sk` (String)
4. Click **"Create table"**

Step 2: Create Lambda Function

1. Go to AWS Lambda Console
2. Click **"Create function"**
3. Choose **"Author from scratch"**
4. Fill in:
- **Function name:** `oith-user-sync`
- **Runtime:** `Node.js 20.x`
- **Architecture:** `x86_64`
5. Click **"Create function"**

Step 3: Add Lambda Code

1. In the Lambda function page, scroll to **"Code source"**
2. Replace the code with the contents of `userSync.js`
3. Click **"Deploy"**

Step 4: Configure Environment Variables

1. Go to **Configuration** → **Environment variables**
2. Click **"Edit"**
3. Add:
- **Key:** `DYNAMODB_TABLE`
- **Value:** `oith-users`
4. Click **"Save"**

Step 5: Add DynamoDB Permissions

1. Go to **Configuration** → **Permissions**
2. Click on the **Role name** link
3. Click **"Add permissions"** → **"Attach policies"**
4. Search for and select **"AmazonDynamoDBFullAccess"**
5. Click **"Add permissions"**

Step 6: Create API Gateway

1. Go to AWS API Gateway Console
2. Click **"Create API"**
3. Choose **"HTTP API"** → **"Build"**
4. Click **"Add integration"**
- **Integration type:** Lambda
- **Lambda function:** `oith-user-sync`
5. **API name:** `oith-api`
6. Click **"Next"**

Step 7: Configure Routes

1. Add these routes:
- `POST /users`
- `GET /users`
- `GET /health`
2. All routes should integrate with your Lambda function
3. Click **"Next"** → **"Next"** → **"Create"**

Step 8: Get Your API URL

1. After creating, you'll see an **"Invoke URL"** like:
`https://xxxxxxxxxx.execute-api.us-east-1.amazonaws.com`
2. Copy this URL

Step 9: Configure Your App

Option A: Set via Browser Console

1. Open your app (https://main.d3cpep2ztx08x2.amplifyapp.com/prototype/index.html)
2. Open browser Developer Tools (F12)
3. In Console, run:
`javascript setAWSApiUrl('https://xxxxxxxxxx.execute-api.us-east-1.amazonaws.com')`

Option B: Set in Admin Dashboard

1. Open the Admin Dashboard (manager.html)
2. Go to Settings
3. Enter your API URL

Step 10: Test It!

1. Create a new user profile on your AWS app
2. Check DynamoDB to see the data saved
3. Open Admin Dashboard to see the user

■ Done!

Now when users create profiles on your AWS Amplify app, their data is automatically saved to DynamoDB and visible in your admin dashboard!

Troubleshooting

CORS Errors

The Lambda function already includes CORS headers. If you still get errors:
1. Go to API Gateway → Your API → CORS
2. Enable CORS for all origins (`*`)

Permission Denied

Make sure your Lambda function has the `AmazonDynamoDBFullAccess` policy attached.

Data Not Showing

1. Check CloudWatch Logs for your Lambda function
2. Verify the DynamoDB table name matches your environment variable