# Tutorial: Manually deploy and troubleshoot WebAuthnKit frontend at AWS

- Introduction
- Prerequisites
- · Clone or download the GitHub project
- · Creating the aws-exports.js file
- · Install and build the WebAuthnKit React web app
- · Zip and upload the React web app files
- Deploy WebAuthnKit React web app at AWS Amplify
- · Checking the React web app at AWS Amplify
- Run the WebAuthnKit React web app locally

#### Introduction

The complete and automated deployment of the WebAuthn Starter Kit backend at AWS is described in the tutorial Automated WebAuthnKit deployment at AWS, which describes the pre-requisites, installation packages, source code and scripts needed for the deployment of the WebAuthn Starter Kit at AWS.

This guide, however, describes how to **manually** deploy the WebAuthn Starter Kit frontend at AWS. Furthermore, it contains a troubleshooting section for identifying and fixing issues that may occur at the AWS frontend.

#### **Prerequisites**

See the prerequisities section in the tutorial Automated WebAuthnKit deployment at AWS.

Before proceeding with deploying the WebAuthnKit frontend, make sure to perform the steps in the tutorial Manually deploy and troubleshoot WebAuthnKit backend at AWS.

#### Clone or download the GitHub project

See the section on how to Clone or download the GitHub project in the tutorial Automated WebAuthnKit deployment at AWS.

## Creating the aws-exports.js file

This section describes the manual steps that are needed for creating the aws-exports.js file for the React web app configuration.

The file aws-exports.js must be created at the workstation in the folder ~\WebAuthnKit\clients\web\react\src\. In order to do this manually, follow the steps below.

Step 1.1. Login to the AWS console, select the Service CloudFormation, and view Stacks.

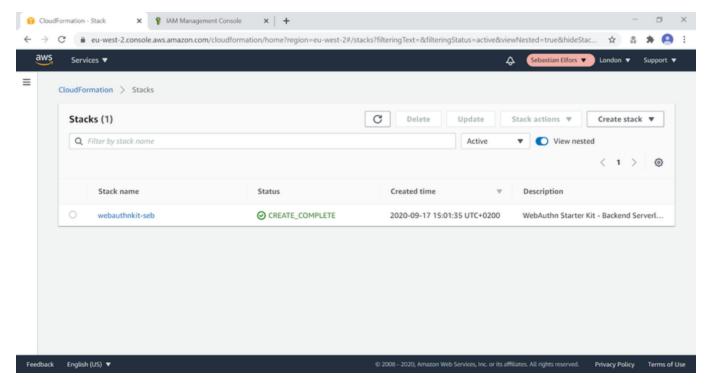


Figure 1 - Select CloudFormation Stack

Step 1.2. Click on the Stack Name in the screen above. In this example, it is called "webauthnkit-seb".

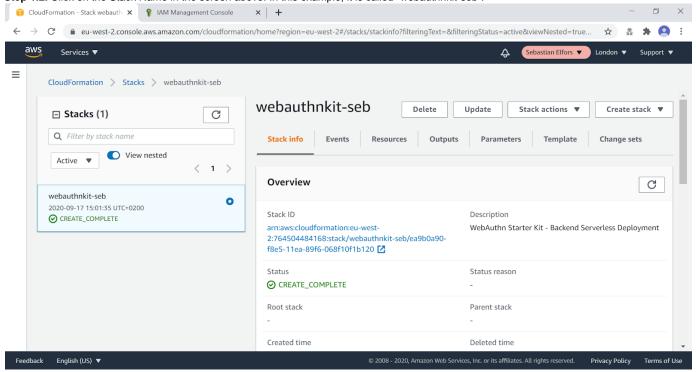


Figure 2 - View CloudFormation Stack details

Step 1.3. Select the option "Outputs".

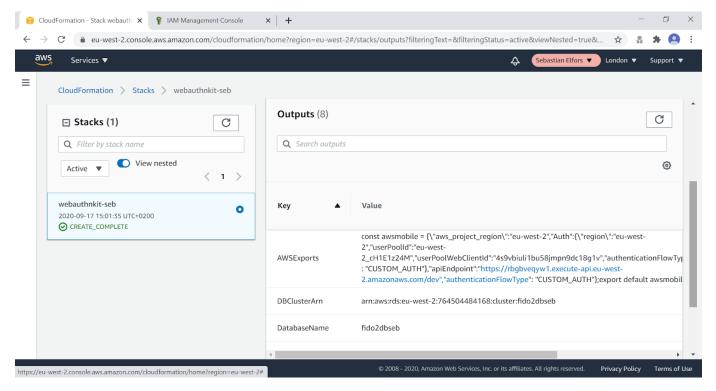


Figure 3 - CloudFormation Stack Output in the AWS Console

Step 1.4. Scroll down in the list of outputs and copy the text next to "AWSExports" from the AWS console.

Another option is to copy the same information from the output in the PowerShell prompt when building and deploying the AWS backend:

```
\times
 Select Windows PowerShell
                                                                                                                 П
                    AWSExports
Kev
Description
                    aws-exports.js configuration for deploying React client to AWS Amplify hosting
/alue const awsmobile = {"aws_project_region":"eu-west-2", "Auth": {"region":"eu-west-2", 'userPoolId":"eu-west-2_uqlidCfbO", "userPoolWebClientId":"14q3sgv5fkkbrm6brdgvt7ro8h", "authenticationFlowType":
Value
'authenticationFlowType": "CUSTOM_AUTH"}; export default awsmobile;
Key
                    DatabaseName
Description
                    Aurora Serverless Database Name
Value
                    fido2dhseb15
                    AWSConfiguration
Key
Description
                    Mobile awsconfiguration.json config
                    {"CognitoUserPool": {"Default": {"PoolId":"eu-
Value
west-2_uqlidCfb0","AppClientId":"14q3sgv5fkkbrm6brdgvt7ro8h","Region":"eu-west-2"}},"Auth": {"Default":
 "authenticationFlowType": "CUSTOM_AUTH"}}}
Successfully created/updated stack - webauthnkit-seb15 in eu-west-2
Re-building the JavaWebAuthnLib.jar file...
      Scanning for projects...
       Building webauthn 1.0-SNAPSHOT
```

Figure 4 - CloudFormation Stack Output in the AWS CLI

- Step 1.5. At the workstation, navigate to the folder ~/WebAuthnKit/clients/web/react/src/ and create a new file called aws-exports.js.
- Step 1.6. Paste the text from "AWSExports" in the AWS console to the file aws-exports.js. Remove the backslashes (if any) in the aws-exports.js.
- Step 1.7. Finally, save the file aws-exports.js in the folder ~/WebAuthnKit/clients/web/react/src/.

Also, make a note of the AppClientId, since that will be used in the section Deploy WebAuthnKit React web app at AWS Amplify.

#### Install and build the WebAuthnKit React web app

Step 2.1. Start a Windows PowerShell prompt, or a MacOS/Linux Terminal, and navigate to the React project folder ~\WebAuthnKit\clients\web\react\. This will be the default folder of the PowerShell prompt throughout the rest of this document.

Step 2.2. In the PowerShell command prompt, run the following commands to install and build the React web application binaries and dependencies:

```
npm install
npm run build
```

#### Zip and upload the React web app files

**Step 3.1.** Navigate to the React distribution folder ~\WebAuthnKit\clients\web\react\dist\ and zip all files in this directory. Call the zip-file Archive. zip, for example.

**Step 3.2.** Use the PowerShell or Terminal command prompt to upload the zip-file to the AWS S3 bucket (which is designated for the WebAuthnKit deployment):

aws s3 cp Archive.zip s3://<s3BucketName> --profile <awsCliProfile>

It is also possible to upload the zip-file manually in the AWS console by navigating to Services Amazon S3 and pressing the button "Upload". S3 Management Console C s3.console.aws.amazon.com/s3/buckets/webauthnkit-seb42/?region=eu-west-2&tab=overview Q # \* Amazon S3 > webauthnkit-seb42 webauthnkit-seb42 Overview Q Type a prefix and press Enter to search. Press ESC to clear EU (London) 2 Viewing 1 to 11 Storage class -276a4022898ee57f1c704b9493bde89c Oct 15, 2020 7:08:54 PM GMT+0200 24.3 KB Standard 88faa248fd8c2b4b108dcd9c6be6a20 Oct 15, 2020 7:08:51 PM GMT+0200 8bb6d7bf586bbc7e11c26fd936854d95 Oct 15, 2020 7:08:47 PM GMT+0200 4.9 KB Standard 9b567b7c39b8f25f095037086b766f0f.template Oct 15, 2020 7:09:43 PM GMT+0200 37.9 KB Archive.zip Oct 15, 2020 7:15:14 PM GMT+0200 195 4 KB Standard c416d3db2cfd4521ff47ab481c4e034c Oct 15 2020 7:08:47 PM GMT+0200 1.3 KB Standard ☐ d41d8cd98f00b204e9800998ecf8427e Oct 15, 2020 7:08:47 PM GMT+0200 0 B Standard Feedback English (US) ▼

Figure 5 - Amazon S3 bucket in the AWS Console

### Deploy WebAuthnKit React web app at AWS Amplify

In this section the WebAuthn Starter Kit React web application will be deployed at AWS Amplify.

Step 4.1. In the PowerShell command prompt, navigate to the folder ~/WebAuthnKit/clients/web/react/.

Step 4.2. In the PowerShell command prompt, run the following command to deploy the WebAuthnKit React client web app:

```
aws amplify start-deployment --app-id <amplifyAppId> --branch-name
<amplifyBranchName> --source-url s3://<s3BucketName>/Archive.zip --
profile <awsCliProfile>
```

If the React app is successfully published to AWS Amplify, the public URL to the app will be displayed at the bottom of the PowerShell prompt and in the AWS Amplify console. For example: <a href="https://dev.d3sohe8raj3jl8.amplifyapp.com">https://dev.d3sohe8raj3jl8.amplifyapp.com</a>.

```
Select Windows PowerShell
                                                                                                                       \times
  main (746 KiB)
      main.js
WARNING in webpack performance recommendations:
You can limit the size of your bundles by using import() or require.ensure to lazy load some parts of your application.
For more info visit https://webpack.js.org/guides/code-splitting/
Child html-webpack-plugin for "index.html":
    Entrypoint undefined = index.html
    [0] ./node_modules/html-webpack-plugin/lib/loader.js!./src/index.html 838 bytes {0} [built]
    [2] (webpack)/buildin/global.js 475 bytes {0} [built]
    [3] (webpack)/buildin/module.js 546 bytes {0} [built]
        + 1 hidden module
  Zipping artifacts completed.
 Deployment complete!
https://dev.d1qpkj8xjrob1i.amplifyapp.com
PS C:\MyFiles\Projects\WebAuthnStarterKit\Deployments\Seb11\WebAuthnKit-0.0.1\WebAuthnKit-0.0.1\clients\web\react>
```

Figure 6 - AWS Amplify URL in the AWS CLI

Step 4.3. Copy the URL at the bottom of the output and paste into a web browser to ensure that the React app can be accessed at AWS Amplify.

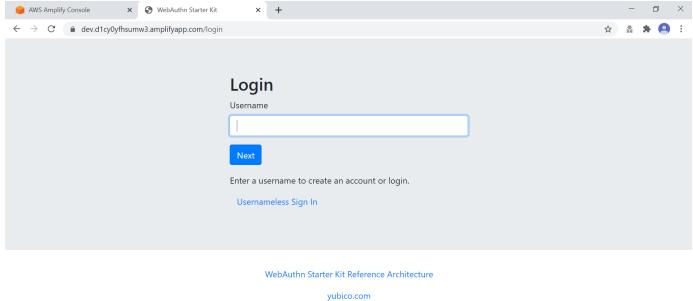


Figure 7 - Accessing the WebAuthnKit React web app running at AWS Amplify

This completes the deployment of the WebAuthn Starter Kit React web app at AWS Amplify.

#### Checking the React web app at AWS Amplify

In order to check the React web app in the AWS Amplify Console, run the following command in the PowerShell command prompt:

amplify console

This will launch the AWS Amplify Console, which allows the user to check the React app settings and status. It is also possible to access the AWS Amplify Console by using the AWS console, and navigating to Services AWS Amplify and selecting the created AWS Amplify web app.

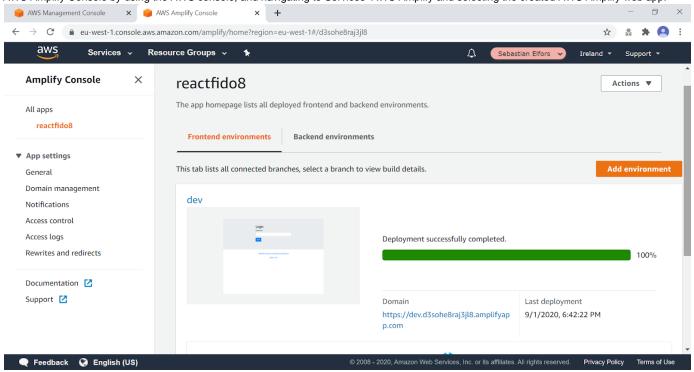


Figure 8 - Checking the React web app at the AWS Amplify console

#### Run the WebAuthnKit React web app locally

It is possible to run the WebAuthnKit React web app locally at the workstation.

**Step 5.1.** First, update the environment variables in the AWS Console for the Lambda function JavaWebAuthnLib<suffix> as described in this section. The environment variables should be changed to the following values:

YUBICO\_WEBAUTHN\_ALLOWED\_ORIGINS: https://localhost:8080 YUBICO\_WEBAUTHN\_RP\_ID: localhost

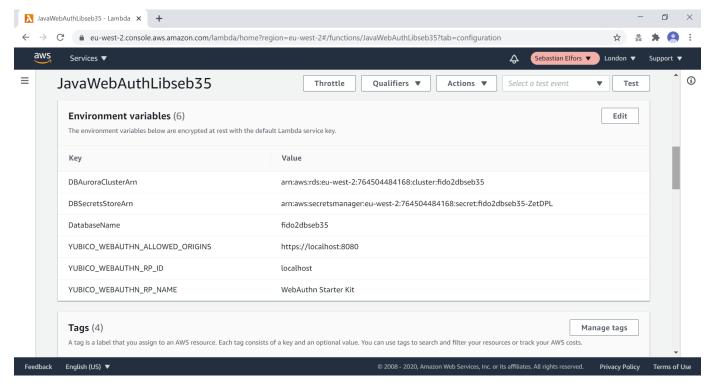
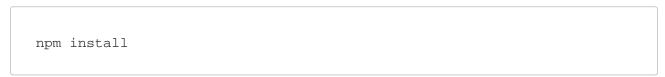


Figure 9 - Changing the environment variables at the AWS console

Step 5.2. In the PowerShell command prompt at the workstation, navigate to the folder ~/WebAuthnKit/clients/web/react/. Then run the following command to install the React web app at the local machine:



Step 5.3. In the PowerShell command prompt at the workstation, run the following command to run the React web app at the local machine:

npm start

**Step 5.4.** If this is successful, a web browser should be launched with the WebAuthn Starter Kit login page. The URL for this locally running instance of the WebAuthnKit React web app is https://localhost:8080.

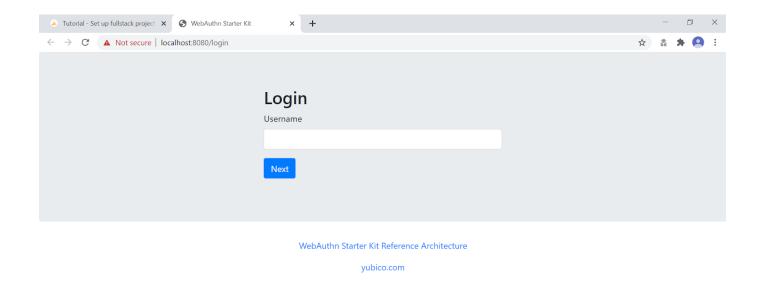


Figure 10 - Accessing the WebAuthnKit React web app running locally