InterOp Station Third-Party Developer Portal User Guide

11.01.23
Version 4

www.mihin.org

Table of Contents

Contents

Purpose of InterOp Station Third-Party Portal User Guide

The purpose of this guide is to help third-party developers register an application (app) as a client of the InterOp Station.

This guide focuses on issues connecting, testing, and/or adding a developer's privacy policy and security attestation documents.

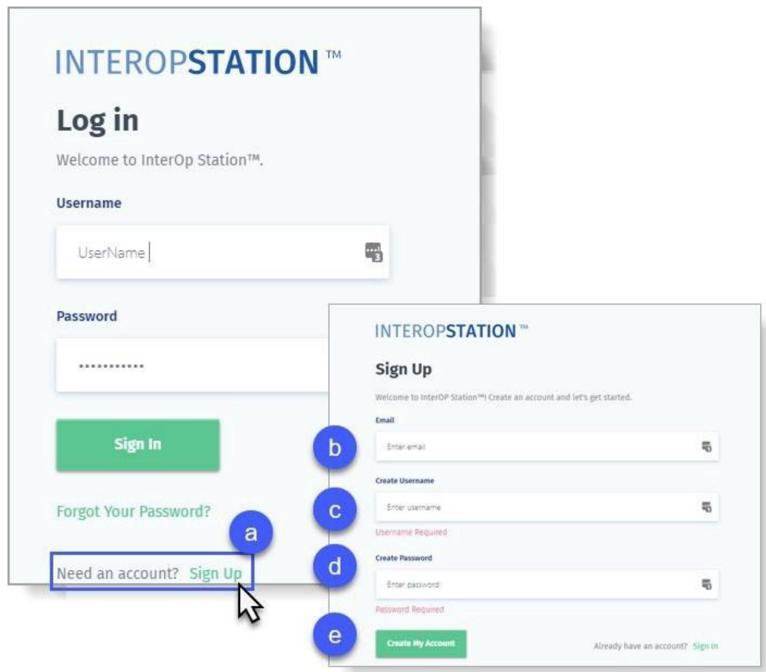
Note: *Third-party developers can contact the MiHIN Help Desk for assistance by email at help@mihin.org.*

Contact the MiHIN Help Desk if you experience any of the following issues while connecting your app:

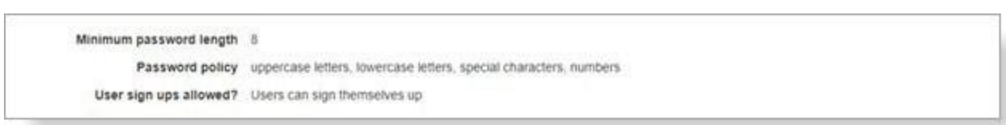
- Cannot submit a security attestation.
- Cannot get credentials in development.
- Tests are failing in development.
- Cannot get credentials for production.
- Tests are failing in production.

Creating an InterOp Station Third-Party Developer Portal Account

1. Navigate to https://www.interopstation.com/login
2. When your **Log in** menu displays:
   a) Select **Sign Up**.



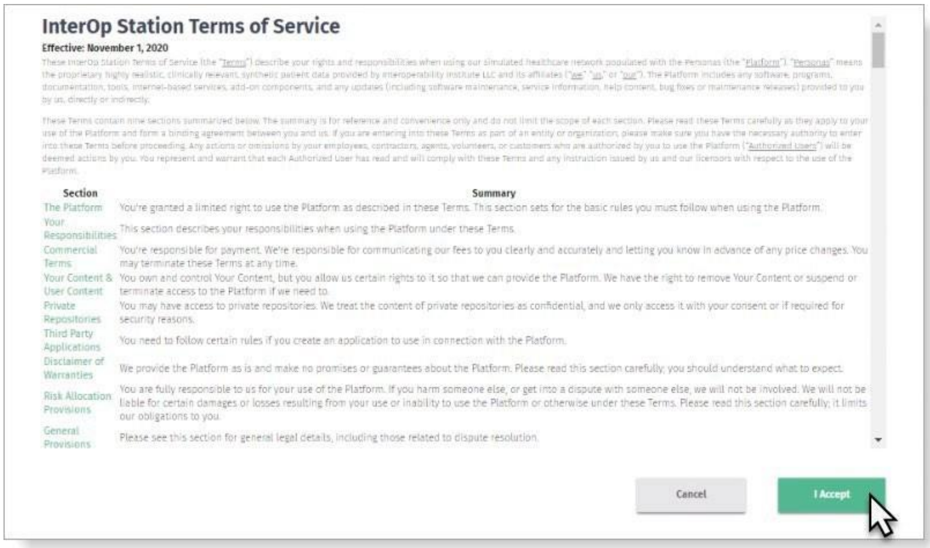
   b) Type your **Email** address.
   c) Create and type your **Username**.
   d) Create and type your **Password** using the password policy as shown here.



   e) Then select **Create My Account**.

3. An email will be sent to the email address provided to confirm your account.
4. Once confirmed, the third-party developer can sign in with the username and password created.
5. Click **I Accept** to agree to the **InterOp Station Terms of Service** and proceed.

Note: *Clicking **Cancel** returns you to the Log in window.*



Sign In Issues After Creating an Account

If a third-party developer has followed the steps appropriately and sign in still fails, refer to the [MiHIN Help Desk].

Customer Logo Request for API Registered Entities

Overview

This process is being designed for situations when a company has stated that they will not store and provide access to their logo.

Process

Assumption: One logo request per registered entity.

The logo disclaimer and the logo request required information below will be included in the third-party developer materials.

Disclaimer for use of logo:

Use of the [Customer] logo ([Customer] logo) is approved for limited use by third-party application vendors. The [Customer] logo is only to be used on the interface to connect with the Customer CMS Interoperability and Patient Access API solution, known as [Customer API name]. Use of the [Customer] logo for any other purpose including display on your company's website, on printed or electronic media, or other materials and/or products that will be distributed to the public requires submission of a separate request to the [Customer] [Representative entity].
Unauthorized use or distribution of the [Customer] logo may lead to civil and/or criminal penalties as allowed under applicable state and federal laws.

The user (entity representative) will request the logo via email and will supply the following information via email.

a. First and Last Name
b. Entity Name
c. Contact Phone Number*
d. Contact Email
e. Name of 3rd party app (app that will use the logo)

*Obtained for outreach if there is an undeliverable email notification for the contact email address.

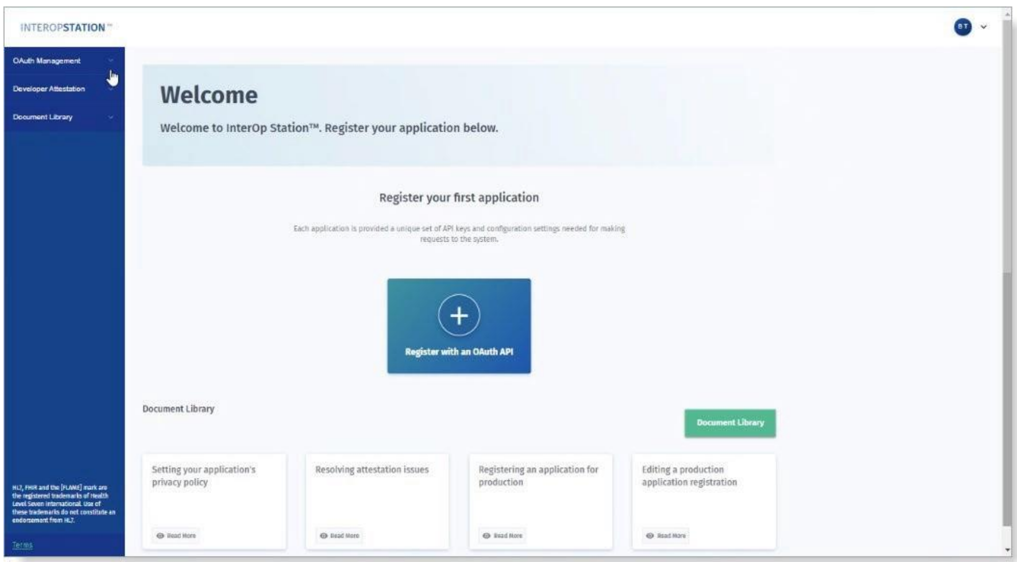
When the email request is reviewed and approved, and the logo is returned via email.

Connecting a Third-Party Developer App to InterOp Station

Welcome Page Navigation

The **Welcome** page allows you to register your app and view supporting information from the Document Library.

When you click **INTEROPSTATION**™ located above the sidebar navigation menu you will return to the Welcome page.



The left sidebar navigation menu provides links to view your OAuth Management including your **Application Dashboard, Developer Attestation,** and the **Document Library**.

Clicking any one of those links (from any page) will redirect you.

Register a SMART Application with the OAuth API Tool

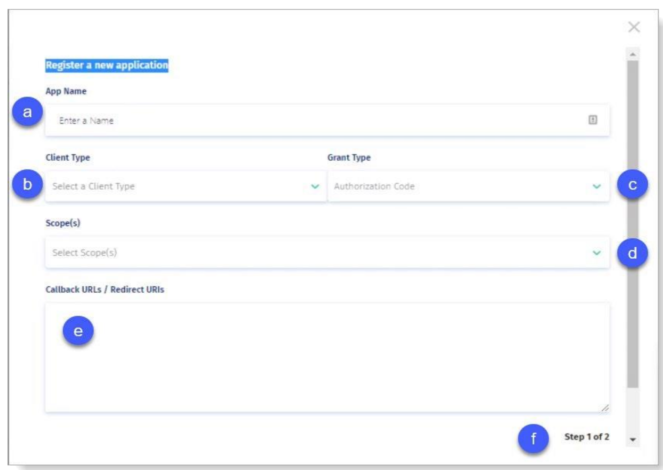
In the OAuth Credentials section of the Welcome page, the **Register with an OAuth API** tool displays. When you select this tool, you will be redirected to the **Register a new application** form.

1. Using the **Register a new application** form, enter the required information as follows:
   a. Type the **App Name** which identifies your SMART App.

   b. Use your **Client Type** arrow to select how you are configuring calls to the token endpoint. The Client ID (username) and secret (password) generated by IOL will be passed to the endpoint via this selection. **Confidential-Basic Auth** is your default and should work unless you know that another form of authentication is used by the app.

   c. Use your **Grant Type** arrow to choose how your app will request and receive the authorization token. (Note: Grant Type is defaulted to "Authorization Code" and is the only option)

   d. HL7 identifies the allowed scopes for your resources. Select your **Scope(s)** arrow to scroll to and select the scope of resources you are requesting for access, for example, CARIN Blue Button® FHIR Smart authorization. For more information on allowed Scopes visit [http://www.hl7.org/fhir/smar t-app-launch/scopes-and-launch- context/](http://www.hl7.org/fhir/smart-app-launch/scopes-and-launch-context/)



   Note: The list of scopes accepted are specified in the Metadata/Capability Statement here: https://api.interopstation.com/{tenant}/fhir/metadata

   (select the appropriate tenant's name to get tenant specific list). Also, each scope needs to be selected manually and added to the list. There is no multi-selection option.

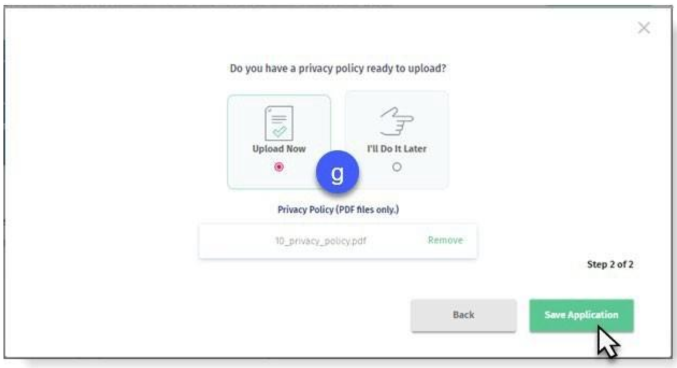
e. Type your **Callback URIs / Redirect URI** for the application you are connecting.

   Note: *To test this application with oauthdebugger.com, list your application's redirect URI and oauthdebugger.com/debug here separated by commas, for example:* https://yourapphere.com/ *or* https://oauthdebugger.com/debug

f. Click **Next** to complete **Step 1 of 2**.

g. The **Step 2 of 2** pop-up prompts you to upload a PDF of your Privacy Policy. Select **Upload Now** if your privacy policy is ready for upload and then click **Save Application**. The app is now connected with your policy.

   Note: *If you are not yet ready to upload your policy, select **I'll Do It Later** and then click **Save Application**. However, your privacy policy* must *be uploaded before your app can go to Production.*

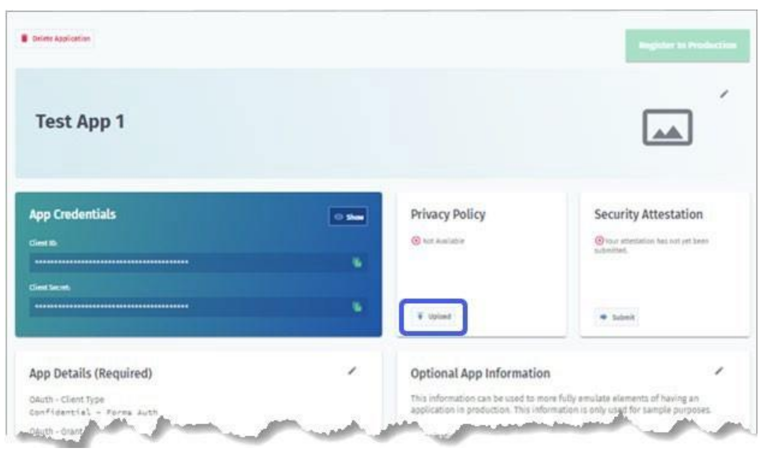


Navigating the Application Dashboard Page

Once the application has been registered with the OAuth API, the App Dashboard page will display. From this page you can:

- Modify the **App Details** you selected during the registration process.
- Upload and review your **Privacy Policy**.
- Complete or review your **Security Attestation**.
- Add **Optional App Information** such as your organization website, a description of the application, a point of contact, and an email address.
- Obtain your app credentials, i.e., Client ID and Client Secret, to complete the connection to the InterOp Station. The Client ID and Secret are also obtainable from the OAuth Credentials section of the Welcome page.

Note**: You can navigate back to this page at any time via **OAuth Management** on the sidebar Navigation Menu and then select **Edit.

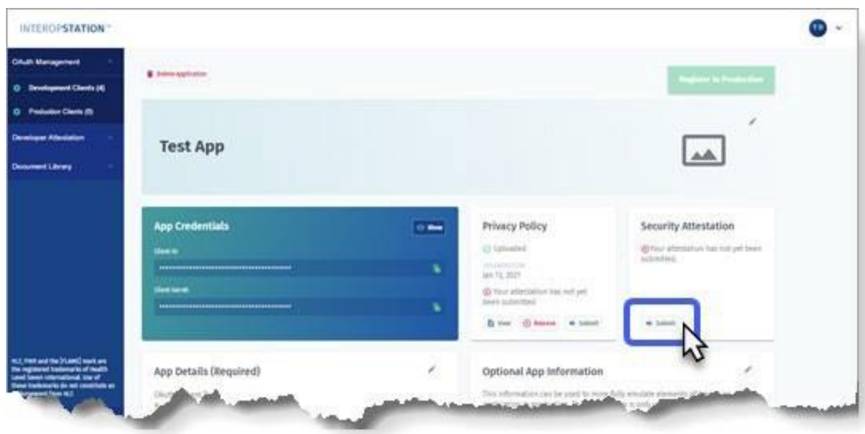


Security Attestation Requirement

Developers are required to submit a Security Attestation for their app. An automated MiHIN Help Desk ticket is generated after a Security Attestation review is completed. The MiHIN Security Team will review the third-party developer ticket and determine whether the submitted Security Attestation is accepted or needs to be resubmitted.

Submitting a Security Attestation

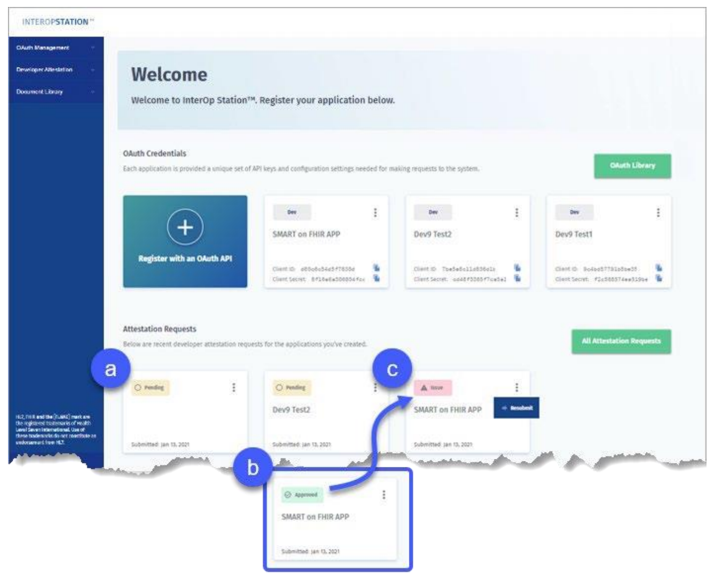
1. Security Attestations can be submitted from the Application Dashboard page by choosing **Submit** located on your **Security Attestation** tool.



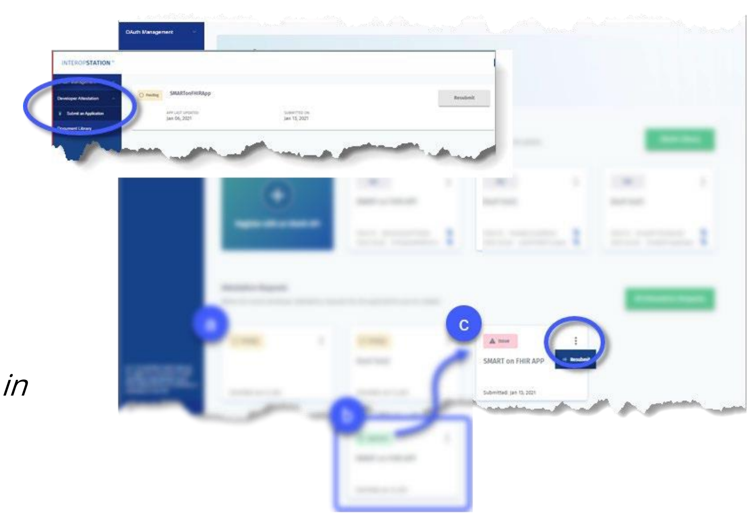
2. When the **Application Attestation** page displays, respond to each question and then click **Submit** to send to the MiHIN Security Team for review.

3. Navigate to and select your **Security Attestation,** which will be like the example shown below.
4. The status of your Security Attestation can be found on the **Welcome** page **Attestation Requests** dashboard or by clicking Attestation Requests located on your Sidebar Navigation Menu.

   *N**ote**: The Security Attestation must be in PDF format. If your Security Attestation is in PDF format and does not upload successfully, escalate to the MiHIN Help Desk at help@mihin.org.*



   a. **Approved**. The Security Attestation has been accepted by the MIHIN Security Team.
   b. **Pending**. The MIHIN Security Attestation has been submitted and is awaiting review.
   c. **Issue**. The Security Attestation has been denied by the MiHIN Security Team which will notify the third- party developers via email. Update your Security Attestation and resubmit for approval.

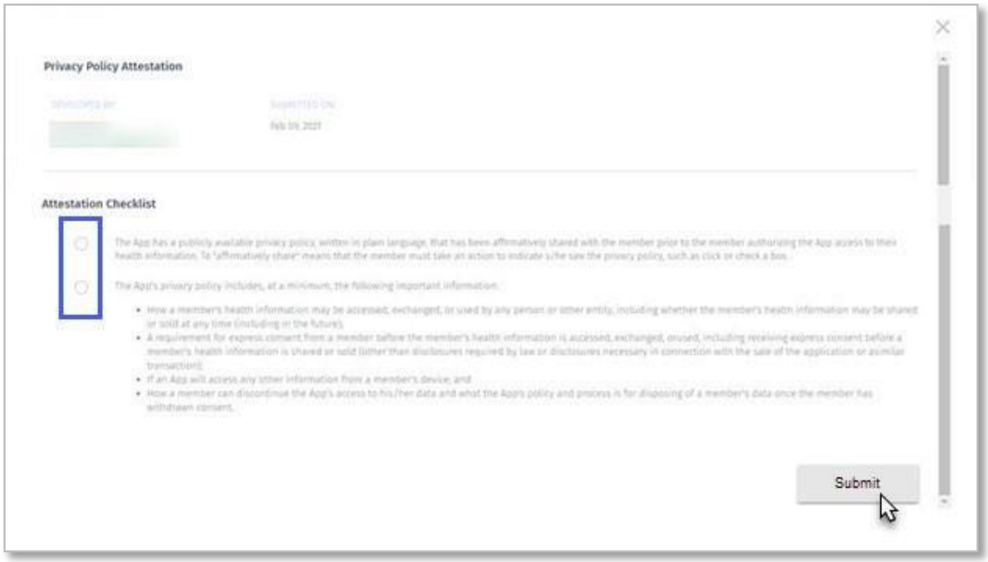


***Note:** To resubmit, select either **Attestation Requests** on the Sidebar Navigation menu or by clicking your **More** vertical ellipses tool on the Security Attestation tile.*

Additional information can be found in the [Upload a Privacy Policy](#) section.

Upload a Privacy Policy

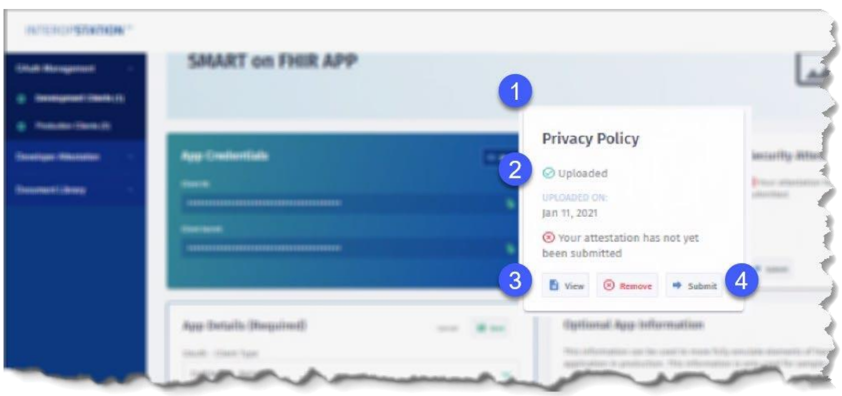
If you chose, *I'll Do It Later* on the *Do you have a privacy policy to upload?* pop up, you can upload it using your SMART on FHIR APP dashboard.



***Note**: The Privacy Policy must be in PDF format. If your Privacy Policy is in PDF format and does not upload successfully, escalate to the MiHIN Help Desk at [help@mihin.org.](mailto:help@mihin.org)*

2. Click **Upload ( )**.
3. Navigate to and select your Privacy Policy. When your PDF file successfully uploads, the options on the Privacy Policy tile change to either *View* or *Remove*.

   ***Note:** Now you can select **View** to preview your policy or select **Remove** if you are not ready to Submit your policy.*



4. Click **Submit** to complete your upload.

Privacy Policy Attestation

When the **Application Attestation** page displays, respond to each question, and then click **Submit**.

> ***Note:*** *How you answer questions on this attestation does not affect whether your application to register with InterOp Station is accepted.*

How to Debug and Validate an OAuth Connection

The Client ID and Client Secret are displayed on the Application Dashboard or on the Welcome page. Copy the credentials and enter them in the appropriate area of the third-party application to complete the connection to the InterOp Station.

The process to validate your OAuth connection is the same whether you are setting up in a Development or Production environment. The connection points for Development and Production vary, as noted in the third-party developer portal document library.

> ***Note:*** *The example below demonstrates how to simulate the OAuth 2.0 connection using the open source https://oauthdebugger.com and making calls via an API.*

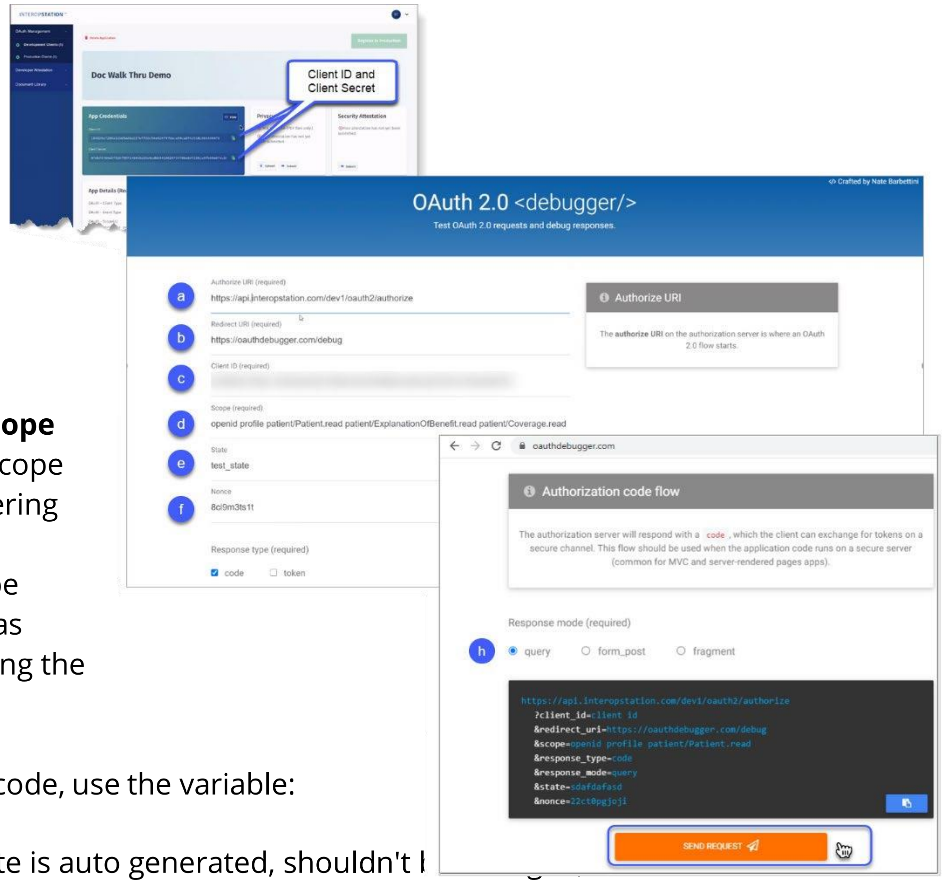
> ***Tip:*** *You will have to update your application to authenticate to interopstation.com using OAuth 2.0 and then API requests based on your application's scope.*

1. The OAuth debugger shown here is used to demonstrate how to enter your required app information such as Client ID and Scope. The image shown below is an example of how a tool like OAuth Debugger could display after you enter your information. An example of code follows this section.

   Note*: The names of the parameters listed below must be entered as shown, as they are case sensitive. All fields are required.*

   a. **Authorize URI** (required). Authorize URIs can be found on interopstation.com, Document Library, InterOp Station API Endpoints, OAuth 2 URL for the environment for which you are trying to connect.
   b. **Redirect URI** (required). From your application or the oauthdebugger.com/debug select **Redirect URI**. In your code, use the variable: **redirect_uri**

c. **Client ID** (required). In your code, use the variable:
   client_id You can get the client id from your application's page on InterOp Station under App Credentials.



d. **Scope** (required). In your code, use the variable: **scope**
   This is the application scope you chose while registering your application.
   Note: The scopes can be copied from list that was selected while registering the app.

e. **State** (required). In your code, use the variable: **state**
   Note: The value for state is auto generated, shouldn't b

f. **Nonce** (required). In your code, use the variable: **nonce**
   Note: This value must be unique for each request. Also, it is auto generated, shouldn't be changed, and cannot be left blank.

g. **Response type** (required). In your code, use the variable: **response_type**
   The default value is **code**. Select **token** if you have a Response type.

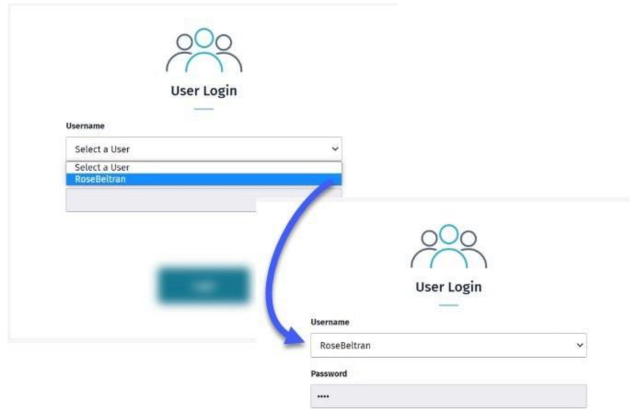
h. **Response mode** (required). In your code, use the variable:
   response_mode=query

An example of the URL after the parameters above have been updated can be found at

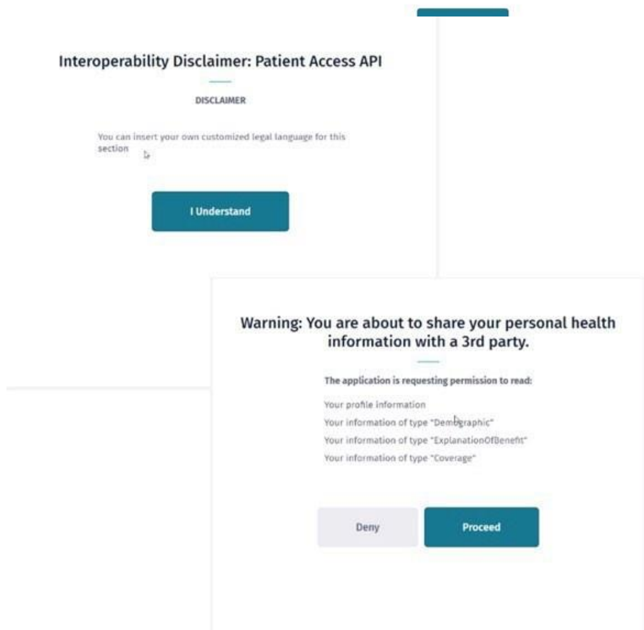
Debugging and validating an OAuth connection in the appendix.

2. After the application connects, you will be redirected to the patient **User Login**. Once you log in, your test patient will display along with the password.

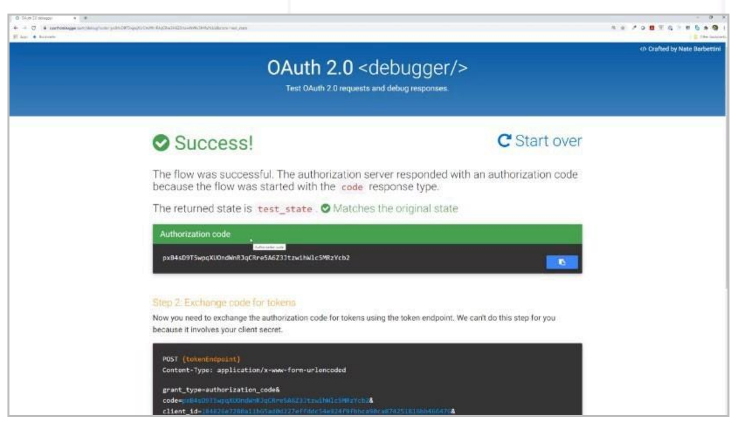
   Note: *"Marian Benton" is a patient in the Development environment. Verify the username and password match the environment you are working in, for example, Development or Production.*



3. The following notifications display using the language that the Payer inserts indicating that the patient will be providing their personal health information (PHI) to a third-party.



4. The **Success!** message will display with your **Authorization code** for Postman.

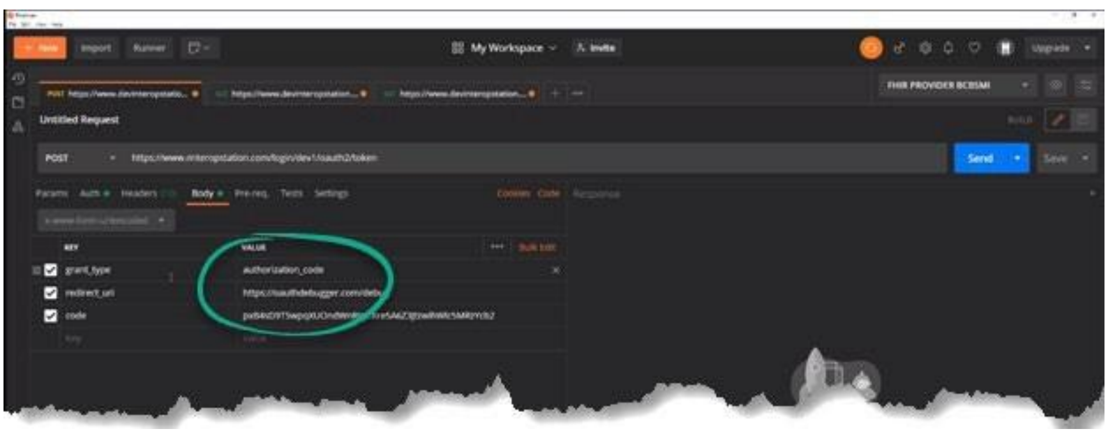


Connecting to InterOp Station

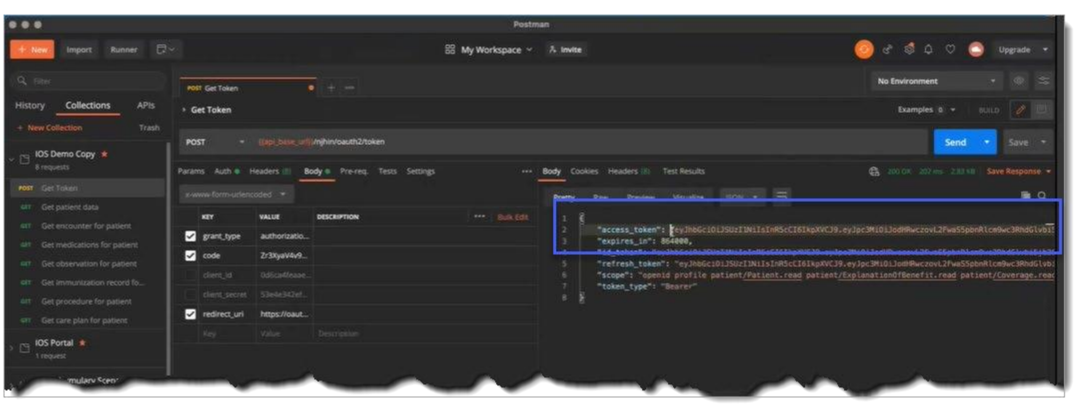
1. Copy your token and then navigate to and open **Postman**.

   Postman Demo bundle is available to download at: https://mihin.org/wp-content/uploads/2021/02/IOS-Demo.postman_collection.json_.zip

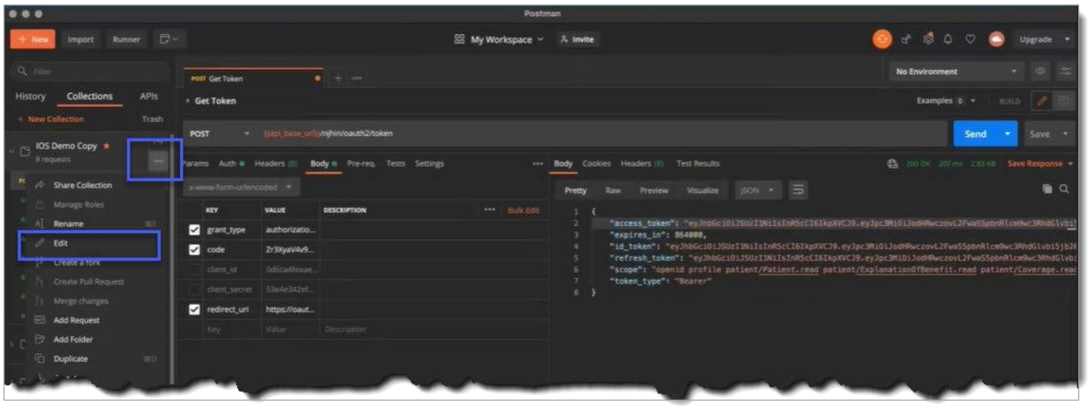
2. Using your **Body** tab:
   - enter your **Client ID** and **Secret**
     (can be obtained from the Developer Console on interopstation.com)
   - enter your **grant_type** key value
     (default value to be given as: “authorization_code”)
   - enter your **redirect_uri** key value
     (default value to be given as: https://oauthdebugger.com/debug)
   - and then Paste your authorization code as your **code** key value
     (this is the Authorization Code received during “How to debug and validate an OAuth connection” section above)



3. Copy the **Access** token string in the **Response** window.



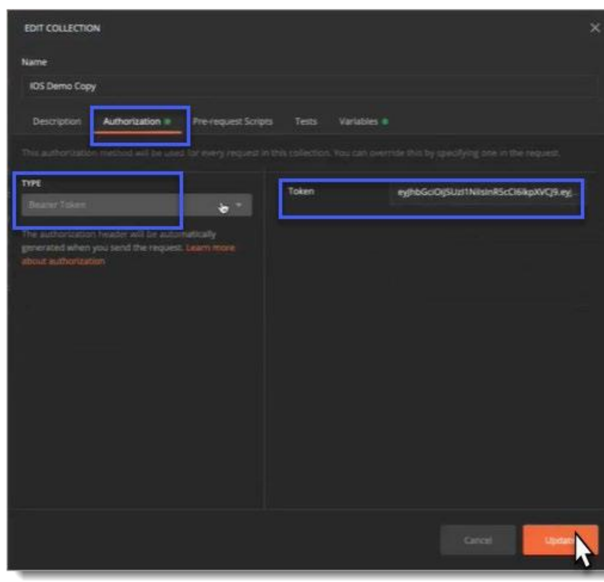
4. On the left navigation menu, click on the **More** horizontal ellipses for options to manage your collection.
5. Click on **Edit** to bring up the **Edit Collection** form.



6. Click on the **Authorization** tab and paste the token in the **Token** field.

   Note: *The* ***Type*** *should be set to* ***Bearer Token****.*

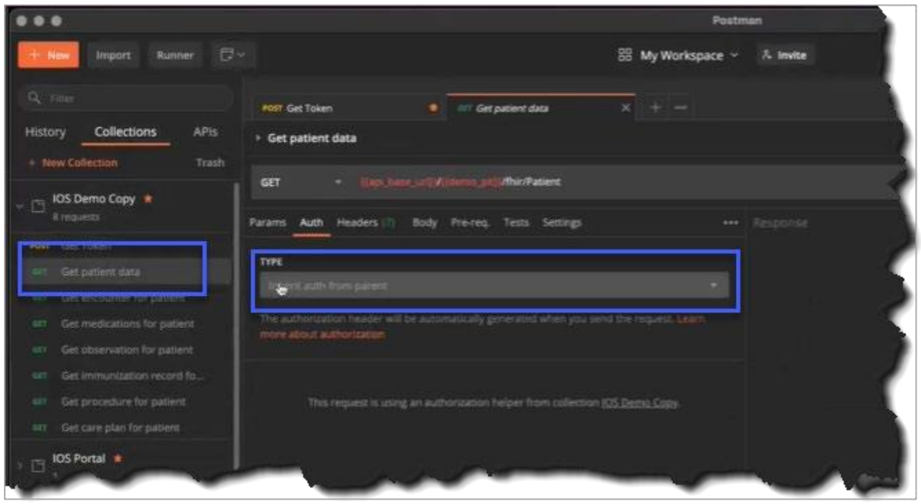
7. Click **Update**.



Testing a Third-Party App Connection to InterOp Station for Development

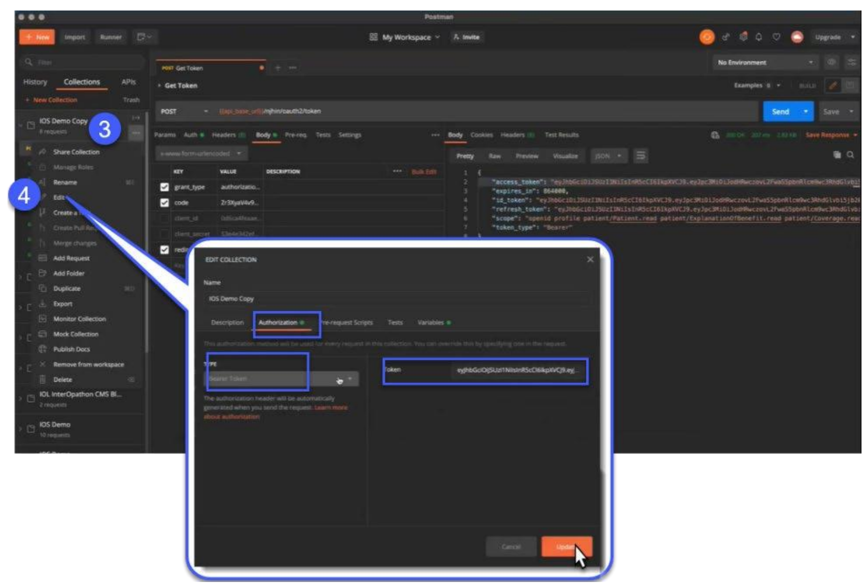
Using the information below, you will be able to test whether you can connect and test for data.

1. On the left side menu, click **Get patient data** to open the **Get patient data** form.
2. On the **Auth** tab, select **inherit auth from parent** in **Type** dropdown menu.

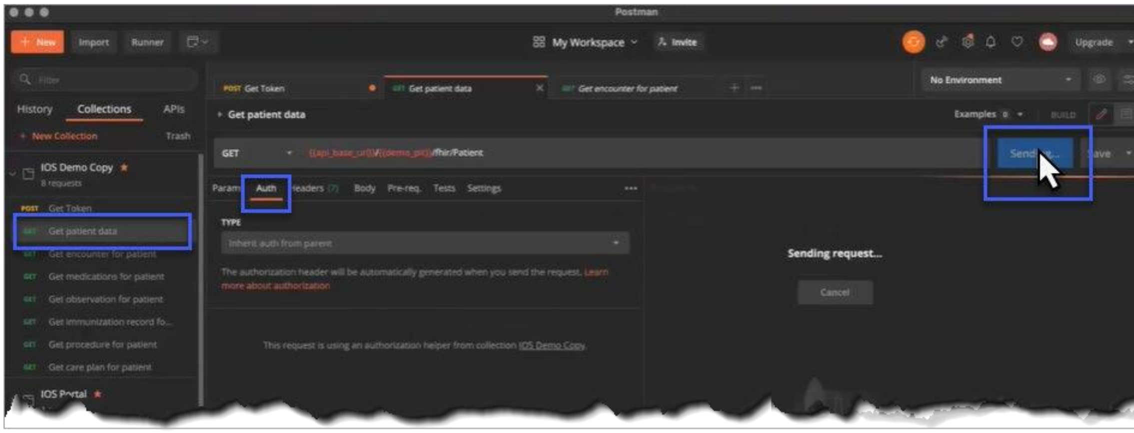


3. Click on the **More** horizontal ellipses for options to manage your collection.
4. Click on **Edit**. The **Edit Collection** form appears.

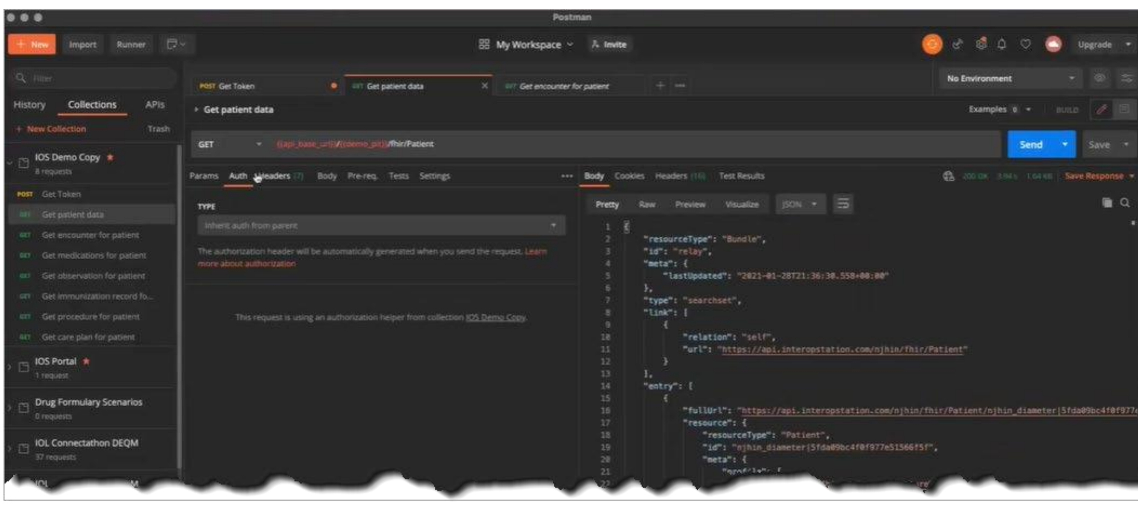
Note: *Confirm* ***Bearer Token*** *is the selected token type on the* ***Auth*** *tab.*



5. On the left side menu, click **Get patient data**.
6. On the **Get patient data** form, click **Send** to retrieve patient data.



7. Patient data displays in the **Response** section of the **Get patient data** form.



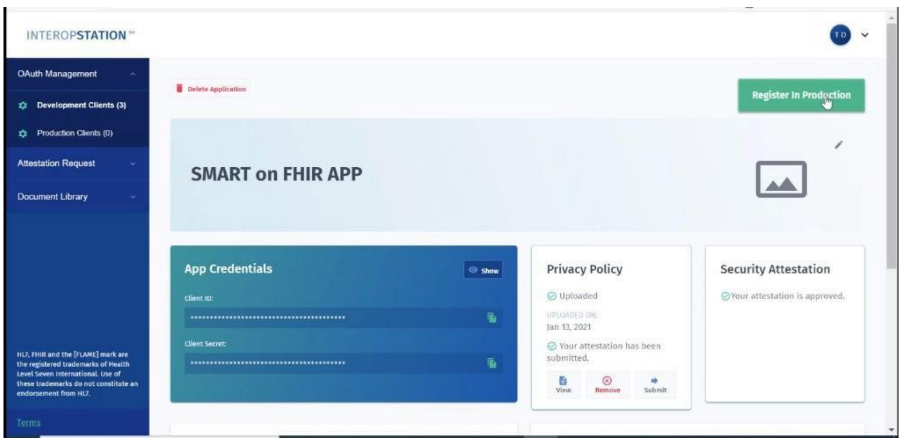
8. Repeat Steps 1 through 7 to retrieve other patient data categories from your collection.

Registering a Third-Party App for Production Clients in InterOp Station

Caution! *When you register an app in Production you will be accessing HIPAA- protected data.*

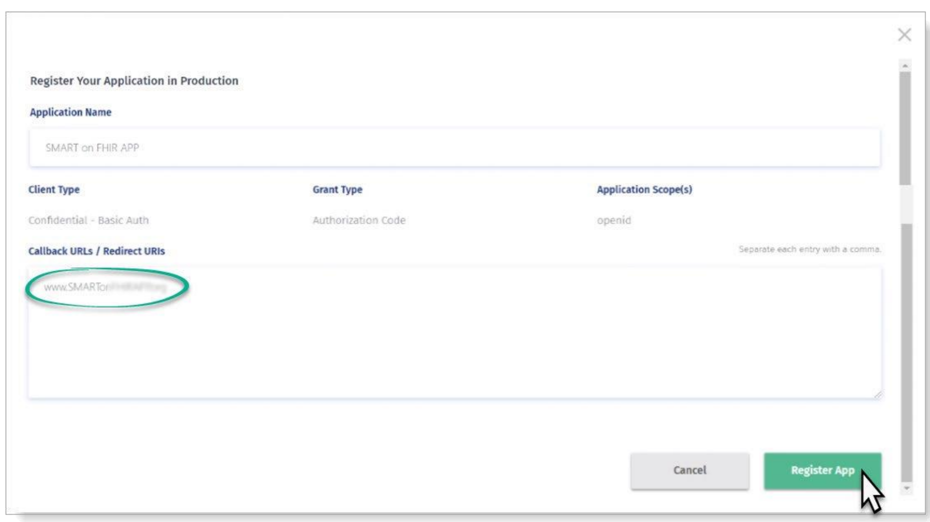
After successfully uploading your Security Attestation and Privacy Policy, navigate to the **Application Dashboard.**

1. Click **Register in Production**.



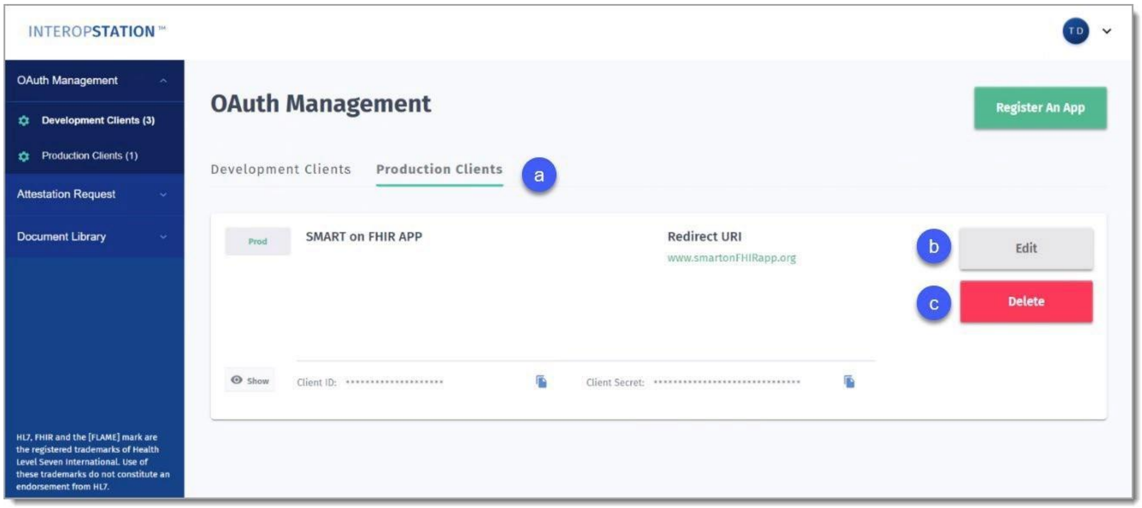
2. In the **Register Your Application in Production** form, type the **Callback URLS/ Redirect URIs** for each application as shown in the example.

3. Click **Register App**.

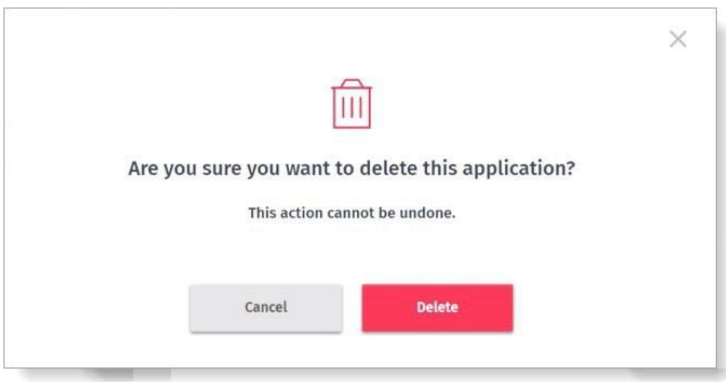


4. In **OAuth Management,** click **Production Clients** on your sidebar navigation menu.
   a. Click the **Production Clients** tab to view a list of your registered apps in production.
   b. Use your **Edit** tool as noted in the *Development Clients* section above.
   c. Use your **Delete** tool to remove an app from Production. If you choose to delete your Sandbox version, you must navigate to the **Development Clients** tab and delete it there as well.

> ***Tip:*** *A best practice is to query test records to confirm your app is registered correctly. Use the Postman App for querying records. To query the test Payer record, you must have an associated test Patient record.*



5. When the **Are you sure you want to delete this application?** message displays, click **Delete** to remove your app from Production.



Testing a Third-Party App Connection in InterOp Station Production

Patient Access API

Follow the same steps as outlined in the section *Testing a third-party app connection to InterOp Station for development* above.

Instead of a patient name and password as shown in Step 2, you will need to use the credentials for a synthetic user.

> ***Note:*** *Production testing uses credentials for a synthetic user. The Development environment will only connect to Development client third-party applications in InterOp Station. The Production environments, for example, BCBSM and NJHIN, will only connect to Production client third-party applications in InterOp Station.*

The synthetic user credentials for testing are

Environment: Development

Username: MarianBenton

Password: [Autofilled in UI]

Note: If the user needs test credentials for Production environment, please contact the MiHIN Help Desk.

Provider Directory API

Third-party app developers should use the following Provider Directory endpoints to connect to the InterOp Station production environment:

- https://api.interopstation.com/[tenant]/fhir/Endpoint
- https://api.interopstation.com/[tenant]/fhir/HealthcareService
- https://api.interopstation.com/[tenant]/fhir/InsurancePlan
- https://api.interopstation.com/[tenant]/fhir/Location
- https://api.interopstation.com/[tenant]/fhir/OrganizationAffiliation
- https://api.interopstation.com/[tenant]/fhir/Organization
- https://api.interopstation.com/[tenant]/fhir/PractitionerRole
- https://api.interopstation.com/[tenant]/fhir/Practitioner

Where [tenant] is the tenant/payer that is being queried.

See the following table for the tenant's name for each customer.

Customer Name	Tenant
Blue Cross Blue Shield of Michigan	bcbsm
McLaren Health Plan	mhp
McLaren MDwise	mdw
Upper Peninsula Health Plan (UPHP)	uphp
Michigan Department of Health and Human Services	mdhhs
New Jersey Medicaid / Family Care	njios

Debugging and validating an OAuth connection

Here is an example of the URL after the parameters above have been updated:

https://api.interopstation.com/dev1/oauth2/authorize?
redirect_uri=https://oauthdebugger.com/debug &**client_id**=client_id&**scope**=openid profile patient/Patient.read patient/ExplanationOfBenefit.read patient/Encounter.read patient/Procedure.read patient/Observation.read patient/Condition.read patient/Immunization.read patient/DiagnosticReport.read patient/ServiceRequest.read&**state**=test&**nonce**=kbbuk9mhz2n &**response_type**=code&**response_mode=query**

Note: ***dev1*** *is an example tenant. Please use the tenant you are targeting, if not* ***dev1***.

Frequently Asked Questions (FAQs)

Q: What is the Refresh Token?

A: The Refresh Token is located below the Access Token in the response. This token expires after 30 minutes. The user can obtain another Access token using the same refresh token for its duration.

Q: What is an Access Token?

A: An Access Token expires after 5 minutes from the time it was issued. Thereafter, the user must obtain a new access token (using the same refresh token) once it expires.

Q: What is a Capability Statement?

A: The types of resources available can be found in metadata for the respective tenant when searched for the meta data in FHIR. Below is the sample query that can be used to do the same: https://api.interopstation.com/{tenant}/fhir/metadata

The general capability statement is also located at:

https://mihin.org/wp-content/uploads/2021/04/IOSCapabilityStatement-typical.zip

Q: What are the requirements of Nonce being a unique value?

A: Nonce is auto generated and nothing needs to be done.

Q: What is the required "Response_mode=query" used for?

A: This is set to get the response back as a query.

Q: Is the patient id in the Authorization Token for the response?

A: Yes

Q: How are we able to get the FHIR patient id and patient data without id?

A: A third-party app cannot have direct access to the patient data without an Authorization Token, which is prompted by the member/enrollee. Once an Authorization Token is obtained, it already has the patient ID in it; it gives back the respective data when queried. There is no need to insert the patient ID separately.

Q: Why don't all Organization resources have Type 2 NPIs?

A: In the IOStation Provider Directory, Organizations have an OID (Object Identifier) as their primary and unique identifiers. Hence, NPI is not a required field here. We suggest using an OID when querying for an organization.

Q: Do we support Bulk API Queries?

A: Although Bulk API Queries are on our road map, they are **not** currently supported by Interopstation.