NextGen Patient Access API Authentication Guide

Version 1.2 – Last Updated June 22nd, 2021

Patient API OAuth2 Authentication

R4 Patient API (**FHIR R4**) Authentication Endpoints & Data Route Base URLs:

- **Authorize**: https://fhir.nextgen.com/nge/prod/patient-oauth/authorize
- **Token**: https://fhir.nextgen.com/nge/prod/patient-oauth/token
- **Conformance**: https://fhir.nextgen.com/nge/prod/fhir-api-r4/fhir/r4/metadata
- **Data Path** *(base URL per endpoint)*: https://fhir.nextgen.com/nge/prod/**fhir-api-r4/fhir/r4/**

DSTU2 Patient API (**FHIR DSTU2**) Authentication Endpoints & Data Route Base URLs:

- **Authorize**: https://fhir.nextgen.com/nge/prod/patient-oauth/authorize
- **Token**: https://fhir.nextgen.com/nge/prod/patient-oauth/token
- **Data Path** *(base URL per endpoint)*: https://fhir.nextgen.com/nge/prod/**fhir-api/fhir/dstu2/**

Note: Patient Access API Route endpoints and usage details are documented in both the Developer Portal (in the API Explorer tool accessed via API Hub) and in the API-specific "Swagger" JSON available here (*Note:* Use the drop-down in the upper right of the Swagger UI to toggle between **DSTU2** & **R4** API documentation.)

Both the DSTU2 and R4 versions of NextGen Patient Access APIs support the following OAuth 2.0 Grant Types; usage examples follow.

- **Authorization Code Grant** (grant_type=authorization_code)
- **Refresh Token Grant** (grant_type=refresh_token)

Note: `YOUR_CLIENT_ID` and `YOUR_CLIENT_SECRET` are issued *provisionally* upon requesting an application in the NextGen API Developer Portal, and become active only when your request to "Add Application" (as detailed in the **How to Register & Configure Your Application** document) is approved by a NextGen Administrator.

Obtaining a Patient API Token via Authorization Code Grant

Patients will login via your redirect page, which will result in the creation of an authorization code your app will then exchange for an `access_code` at the POST /token auth endpoint. To begin, your app will need to launch a browser frame/window so the patient can enter their Patient API credentials.
The initial Call from your app to display the NextGen Patient API login UI for the Authorization Code Grant is shown below.

Example Patient API Authorization Call Using Authorization Code Grant

```
GET https://fhir.nextgen.com/nge/prod/patient-
oauth/authorize?response_type=code&client_id=YOUR_CLIENT_ID
&redirect_uri=https://YOUR_CALLBACK_URL
```

- The patient enters their Patient Access API credentials *(created via the Patient API Enrollment Workflow process, **which is facilitated by the patient's NextGen Practice**)* and authenticates.
- The authorization code issued under the grant must then be POSTed to the /token endpoint with the parameters & headers shown below, and the token is returned in response.
- **Please note the required application/x-www-form-urlencoded Content-Type header.**

Request to Token (after obtaining the Authorization Code):
*(**Note**: The FHIR R4 data path is shown in the example for this endpoint)*

```
curl -X POST \
  'https://fhir.nextgen.com/nge/prod/patient-
oauth/token?grant_type=authorization_code&redirect_uri=https://YOUR_CALLBACK_URL
&client_id=YOUR_CLIENT_ID&client_secret=YOUR_CLIENT_SECRET&code=ISSUED_CODE' \
  -H 'Content-Type: application/x-www-form-urlencoded' \
```

Obtaining a Refresh Token via Refresh Token Grant

- The FHIR DSTU2 and FHIR R4 Patient APIs both support refresh tokens.
- You can exchange a refresh token for a new access token via the /token endpoint by changing the `grant_type` to equal `refresh_token` and adding a `refresh_token` query parameter and value as shown in the following examples.

Note: The Refresh Token examples below use inconsistent endpoints to illustrate the differences in the base URL between the DSTU2 and R4 versions of Patient API. Your app's use of base URL must be consistent for success with the Patient API version you elect to use.

- Refresh Tokens must be exchanged within 48 hours of issue.

Refresh Token Grant Request Example (*DSTU2 endpoint shown*):

```
curl -X POST \
  'https://fhir.nextgen.com/nge/prod/patient-
oauth/token?grant_type=refresh_token&refresh_token=REFRESH_TOKEN_VALUE&client_id=
YOUR_CLIENT_ID&client_secret=YOUR_CLIENT_SECRET' \
  -H 'Content-Type: application/x-www-form-urlencoded' \
```

Subsequent Patient API Call *(R4 endpoint shown)*:

```
curl -X GET \
  https://fhir.nextgen.com/nge/prod/fhir-api-r4/fhir/r4/Patient \
  -H 'Authorization: Bearer YOUR_ACCESS_TOKEN' \
  -H 'Accept: application/json'
```

The "id" in the response payload returned by the GET /patient route in the example above will be the patient ID required as a query parameter in many other routes in the Patient API set.

The route-level documentation includes details specifying the routes that will expect this Patient ID value as the value for the {patient} query parameter (which is required by most Patient API routes). The same {id} value returned by GET /Patient is also expected as the **{id}** path variable for the GET /Patient/**{id}** route.

How to Make Patient Access API Calls

- The Patient Access APIs (named **FHIR DSTU2** & **FHIR R4**) enable patients to access their data using compatible applications as required by various regulations.
- After having obtained your app's **client_id** & **client_secret** (provided upon completion of the process documented in **How to Register & Configure Your Application**), your app will be able to obtain an authentication token and use the token to make subsequent data requests on behalf of the authenticated patient. Your app may also use the `refresh_token` grant to obtain a new `access_token`), provided you exchange the `refresh_token` for a new `access_token` within 48 hours of the initial issue of the `refresh_token`. The `access_token` is required to make any & all Patient Access API calls, excepting calls to the OAuth endpoints and to the GET /metadata conformance routes.
- *Testing Notes*:
    - If you are using another tool such as Postman, the base URL variant (either **DSTU2** or **R4**) from the top of this document) you use to make data calls will invoke the corresponding version of Patient Access API. Use the Authentication Grant detailed in the prior section to obtain an access_token.

- Remember, the API(s) named **FHIR DSTU2** and/or **FHIR R4** must have first been added to your app in the Developer Portal as described in the **How to Register & Configure Your Application** document prior to success.

- Upon obtaining an `access_token` for either version of Patient API, you can proceed to making data calls to individual Patient API routes documented in the conformance (GET /metadata) route response and in the online Patient Access API Documentation. The Base URLs are unique to each FHIR version; please consult the lists of Authentication & Data Route Endpoints in the prior **Patient API OAuth2 Authentication** section as needed.
- Additional guidance is available in the Patient Access API FAQ.

Expected Headers

Always ensure your GET calls to data endpoints include the following header values:

```
-H 'Authorization: Bearer YOUR_ACCESS_TOKEN' \
-H 'Accept: application/json' \
```

Patient API Sandbox Credentials

The following Patient API credentials (also shown in prior examples) are valid for testing both versions of Patient API (**nge.prod.fhir-api** & **nge.prod.fhir-api-r4**):

- Sandbox Username: **patientapitest**
- Sandbox Password: **Password1!**

FAQ - Patient Access API

- **What is the workflow for obtaining Patient ID?**
    - After completing the appropriate authentication protocol for the **nge.prod.fhir-api** *(DSTU2)* or **nge.prod.fhir-api-r4** *(R4)* Patient APIs, call the GET /Patient route *without using any parameters*.
    - If using the API Explorer tool, enter the sandbox test patient credentials (U:) testsurgery_592 (P:) password1 in the authorization login pop-up and select "Next", then select "Grant" on the subsequent screen to complete authentication.
    - If using Postman (or your app), obtain your authentication token via the processes detailed on the **Authentication Details** wiki page under "Patient API OAuth2 Authentication".
    - The GET /Patient response will include the {patient} ID guid in the response as "id": "{patient}".

- **What is expected as the value of {id} in routes with an {id} Path Variable (e.g. GET /Endpoint/{id})?**
    - Calls to routes with **{id}** path variables expect the value you provide for **{id}** to match an **{id}** value of an entry that you must have already obtained in the response payload from a prior request (*without* an **{id}**) to the same endpoint.
    - For example, you cannot call **`GET /AllergyIntolerance/{id}`** until you have first called **`GET /AllergyIntolerance?Patient={patient_id}`** to obtain one or more AllergyIntolerance **{id}** values for the authenticated Patient.
    - You would then use an AllergyIntolerance **{id}** value from the desired entry in the response payload from **`GET /AllergyIntolerance?Patient={patient_id}`** as the value of the **{id}** path variable (e.g. **`GET /AllergyIntolerance/{your_desired_allegyintolerance_id}`**).

- **What are the available values for routes that require a "Category" query parameter?**

Patient API Version(s)	**Endpoint**	Acceptable Values for **Category**
DSTU2 & R4	/CarePlan	**assess-plan**
DSTU2 *Only*	/CarePlan	**careteam** *(R4 includes an equivalent /CareTeam endpoint)*
DSTU2 & R4	/Condition	**encounter-diagnosis**
		health-concern
		Problem-list-item
	/DiagnosticReport	**lab**
	/Observation	**laboratory**
		vital-signs
		social-history

- **What is the integration process for a NextGen Client Practice who want to use our Patient Access API App?**

  - NextGen does not gate or otherwise control the progression from Development > Beta > Production for Patient Access API apps since (per ONC-CMS regulations) we do not require your organization to have a partnership agreement with NextGen as a condition of your App's usage of Patient Access APIs. The choice to use your app is made by each Patient of a NextGen Enterprise EHR® Client Practice.
  - Prior to going to production, NextGen requests that each Patient Access API developer communicate the following:
    - Notify NextGen (via APIpartners@nextgen.com) of their intention to begin support of production integrations of their application;
    - Provide NextGen (via APIpartners@nextgen.com) with any & all requirements that individual NextGen Healthcare client practices must satisfy (e.g. practice registration, etc.) before patients of that practice will be able to utilize your application to obtain their data via Patient Access API.
    - Communicating this information (and any related details (e.g. practice registration URL, etc.) to NextGen will allow us to list your app among those known to support NextGen Patient Access APIs in both Client-facing & Patient-facing information.
  - NextGen Practices do not need to provide you with a unique Base URL or any other kind of identifier for authentication or data access prior to Patient Access API usage. This is because the logic that returns the correct patient's data to your app is handled entirely by the NextGen API platform as a function of the unique patient credentials passed to NextGen during the authentication process for Patient Access API.
  - Patient API credentials are issued by a Practice to a patient (which the Patient must use to log into your app) via a Patient Enrollment Workflow Process. These Patient API credentials are unique to each patient for each specific NextGen Practice. Documentation for the enrollment process that establishes Patient API credentials is included in the API Installation details referenced in the next FAQ item below.

- **Do NextGen Clients need to complete any prerequisites to use a Patient Access API App?**

    - To enable Patient use of a Patient Access API app, a NextGen practice only needs to follow the "**How do I get started?**" steps detailed near the end of the article in NextGen Success Community (*our client portal*) entitled "**NextGen Patient Access API**".
    - Developers should be aware that the required steps for NextGen® Enterprise EHR Clients to enable Patient Access API are:
        1. License & Install (at no charge) the API Suite Manager software in their NextGen® Enterprise EHR production server environment;
        2. Enroll individual patients in API via the "**Patient Enrollment Workflow**" to establish the Patient API credentials each patient will use to enable Patient Access API App authentication and personal data retrieval.

- **What information should we provide to NextGen® Enterprise EHR Clients to help them verify they are prepared for Patient Access API integration?**

    - You may provide the following link (pointing to the **NextGen Patient Access API** Success Community Article) to clients as a courtesy: https://www.community.nextgen.com/articles/Hot_Topic/NextGen-Patient-Access-API
    - Keep in mind that access to the NGE Customer Success Community is solely for current NextGen Healthcare clients.
    - The client-facing article linked above contains an introduction to Patient API, a Patient Access API FAQ, detailed process instructions (API licensing & installation, links to the *Patient Enrollment Workflow* documentation, etc.), and more.