

How to Test Firely Server on Inferno

A Step-by-Step Guide to Demonstrate
100% Compliance with the 21st Century
Cures Act Final Rule



server@fire.ly

www.fire.ly

[@FirelyTeam](https://twitter.com/FirelyTeam)

How to Test Firely Server on Inferno



***The following guide, reference servers, and testing data are currently implemented for US Core 3.1.1 USCDI V1 and SMART 1.0 only.**

Are you considering using Firely Server to power your healthcare portal, app, or database? Or maybe you already do?

Either way, with the ‘Final Rule’ deadline fast approaching, it is good to know that Firely Server is 100% compliant with the § 170.315(g)(10) criterion.

This Step-by-Step Guide shows you how to test that compliance using Inferno **7.2.3**. It should take about 30 minutes to complete.

Contents

1	The 21st Century Cures Act Final Rule	4
2	Timelines	4
3	Introduction to Inferno	4
4	Now let’s start testing!	5
5	Standalone Patient App – Full Patient Access	8
6	Standalone Patient App – Limited Access	12
7	EHR Practitioner App	16
8	Single Patient API	21
9	Multi Patient API	25
10	Additional Tests	27
	10.1 Public Client Standalone Launch with OpenID Connect	27
	10.2 Token Revocation	31
	10.3 SMART App Launch Error: Invalid AUD Parameter	36
	10.4 SMART App Launch Error: Invalid Access Token Request	39
	10.5 EHR Launch with Patient Scopes	43
	10.6 Visual Inspection and Attestation	49
11	Wrapping things up	49



12	About Firely	49
13	Contact us	49

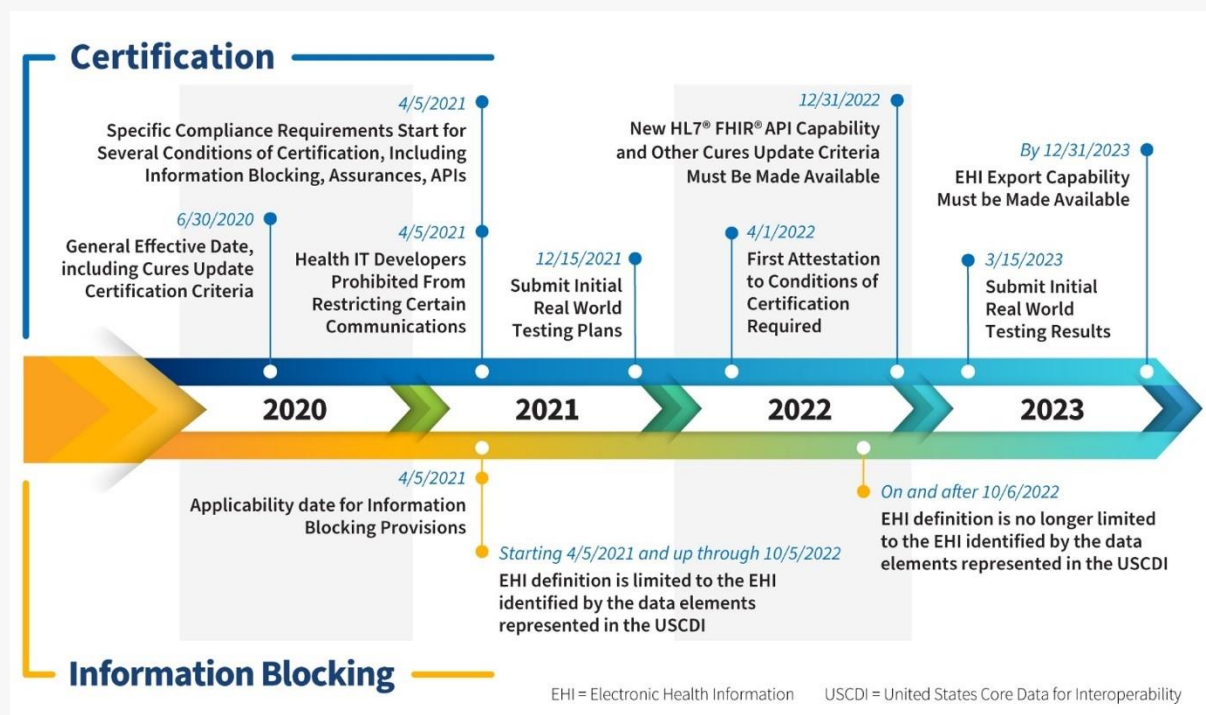
1 The 21st Century Cures Act Final Rule

ONC has adopted a new secure, standards-based API certification criterion in § 170.315(g)(10) to implement the 21st Century Cures Act's requirement that certified health IT APIs can be used "without special effort."

This new criterion requires standardized API access for single patient and population services and is limited to API-enabled "read" services using the HL7[®] Fast Healthcare Interoperability Resources (FHIR[®]) standard. The FHIR standard, in addition to a set of adopted implementation specifications, provides known and consistent technical requirements for software developers.

2 Timelines

The timelines for the Final Rule are shown below. The deadline for compliant FHIR APIs to be in place was 31 December 2022¹. And the regulation for EHI export capabilities - 170.315(b)(10) - is mandatory starting from December 31, 2023



Additional required updates can be found [here](#).

3 Introduction to Inferno

Inferno (<https://inferno.healthit.gov/onc-certification-g10-test-kit>) is an open-source tool to test if a FHIR server is compliant with the above-mentioned criteria as defined in § 170.315(g)(10). It tests for conformance to authentication, authorization, and FHIR content standards including:

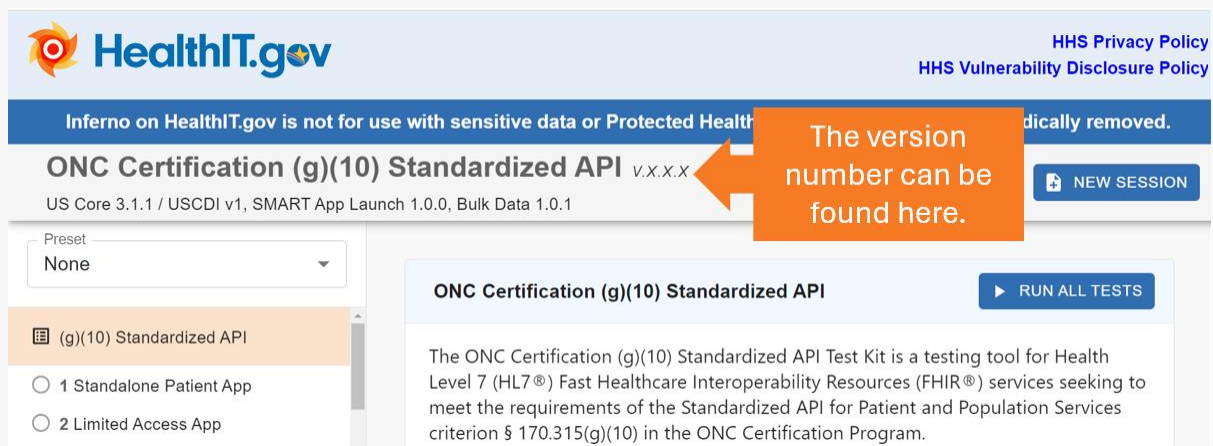
- FHIR Release 4.0.1
- FHIR US Core Implementation Guide (IG) STU 3.1.1
- SMART Application Launch Framework Implementation Guide Release 1.0.0

- HL7 FHIR Bulk Data Access (“Flat FHIR”) (v1.0.1: STU 1)

Firely Server ($\geq$ v4.0) is fully compliant with all these requirements. To demonstrate this, we have documented all the steps you should take to test Firely Server using Inferno.

There are six groups of tests with implicit dependencies between them, and we recommend you run them in the order we have used here. For each test we have described all the information you need to enter manually and provided screenshots with orange rectangles showing where it should be typed in. We also explain and show the overall flow of the test and the end result.

These instructions are based on Inferno version **7.2.3**. The Inferno version number can be found here:



If you’re using an older version of Inferno you should update to version **7.2.3** before using this guide.

Authentication and authorization are a key element of these tests. The tokens you get after logging in are valid for one hour which should be plenty of time to run all the tests. If a test fails because a token has expired you will need to start the process from the beginning.

The public Inferno server can be a bit slow, which can also result in a failing test. If all the data is entered correctly the test should succeed on the second attempt.

All tests should be green when complete. If you do encounter an unexplained failing test, feel free to reach out to server@fire.ly.

4 Now let’s start testing!

1. Navigate to https://inferno.healthit.gov/suites/g10_certification
2. Keep the default radio selections as US Core 3.1.1 / USCDI v1, SMART App Launch 1.0.0 and Bulk Data 1.0.1:





[Test Kits](#) [News & Updates](#) [Events](#) [About](#)

Inferno on HealthIT.gov is for demonstration only. Not for use with sensitive data or Protected Health Information (PHI). Data periodically removed. ⓘ

[HOME](#) / [TEST KITS](#) / [ONC CERTIFICATION \(G\)\(10\) STANDARDIZED API TEST KIT](#)

ONC Certification (g)(10) Standardized API Test Kit

Last Updated: June 10, 2025

Test Kit Version: 7.2.3

Maturity: High ⓘ

[SMART App Launch](#) [US Core](#) [Bulk Data](#)

The ONC Certification (g)(10) Standardized API Test Kit is a testing tool for Health IT systems seeking to meet the requirements of the Standardized API for Patient and Population Services criterion § 170.315(g)(10) in the ONC Certification Program. It is an approved test method for the [§ 170.315\(g\)\(10\) test procedure](#).

Systems may adopt later versions of standards than those named in the rule as approved by the ONC Standards Version Advancement Process (SVAP). Please note that US Core Implementation Guide v.7.0.0 should only be used with SMART App Launch Guide v2.0.0 or above due to granular scope support requirements within this version of US Core.

Please select which approved version of each standard to use, and click 'Create Test Session' to begin testing.

This test kit includes a [simulated conformant FHIR API](#) that can be used to demonstrate success for all tests. This simulated API is open source and is available on [GitHub](#). Visit the [walkthrough](#) for a demonstration of using these tests against the provided simulated FHIR API.

Status

The ONC Certification (g)(10) Standardized API is actively developed and updates are released monthly.

The test kit currently tests all requirements for the [Standardized API for Patient and Population Services criterion § 170.315\(g\)\(10\)](#), including updates from the [HTI-1 Final Rule](#). This includes:

- SMART App Standalone Launch with full system access
- SMART App Standalone Launch with limited system access
- SMART App Standalone Launch with OpenID Connect
- SMART App EHR Launch with user scopes
- SMART App EHR Launch with patient scopes
- SMART App Launch Invalid AUD Parameter
- SMART App Launch Invalid Access Token Request
- SMART App Launch Token Introspection
- SMART App Launch v1 and v2 scopes
- SMART App Launch finer-grained scope access
- Support for Capability Statement
- Support for all US Core Profiles
- Searches required for each resource
- Support for Must Support Elements
- Profile Validation
- Reference Validation
- Export of multiple patients using the FHIR Bulk Data Access IG

See the test descriptions within the test kit for detail on the specific validations performed as part of testing these requirements.

Repository and Resources

The ONC Certification (g)(10) Standardized API Test Kit can be [downloaded from its GitHub repository](#), where additional resources and documentation are also available to help users get started with the testing process. The repository [Wiki](#) provides a [FAQ](#) for testers, and the [Releases](#) page provides information about each new release.

Providing Feedback and Reporting Issues

We welcome feedback on the tests, including but not limited to the following areas:

- Validation logic, such as potential bugs, lax checks, and unexpected failures.
- Requirements coverage, such as requirements that have been missed, tests that necessitate features that the IG does not require, or other issues with the interpretation of the IG's requirements.
- User experience, such as confusing or missing information in the test UI.

Please report any issues with this set of tests in the [issues section](#) of the repository.

Start Testing

US Core Version

- ☒ US Core 3.1.1 / USCDI v1
☐ US Core 4.0.0 / USCDI v1
☐ US Core 6.1.0 / USCDI v3
☐ US Core 7.0.0 / USCDI v4

SMART App Launch Version

- ☒ SMART App Launch 1.0.0
☐ SMART App Launch 2.0.0
☐ SMART App Launch 2.2.0

Bulk Data Version

- ☒ Bulk Data 1.0.1
☐ Bulk Data 2.0.0

[+ Create Test Session](#)

Your Recent Sessions ⓘ

[\(g\)\(10\) Standardized API](#)

5/23/2025, 11:38:44 AM

[\(g\)\(10\) Standardized API](#)

5/9/2025, 8:41:37 PM

[\(g\)\(10\) Standardized API](#)

5/9/2025, 4:03:03 PM

[\(g\)\(10\) Standardized API](#)

4/28/2025, 3:04:26 PM

[\(g\)\(10\) Standardized API](#)

7/5/2024, 12:06:26 PM

3. Click button **Create Test Session**

5 Standalone Patient App – Full Patient Access

1. Click **1 Standalone Patient App** on the testing menu on the left of the screen and click **RUN TESTS**
2. Fill in these values:

Field	Value
FHIR Endpoint	https://secure.server.fire.ly
Standalone Client ID	Inferno
Standalone Client Secret	secret

Register Inferno as a standalone application using the following information:

- Redirect URI:
`https://inferno.healthit.gov/suites/custom/smart/redirect`

Enter in the appropriate scope to enable patient-level access to all relevant resources. In addition, support for the OpenID Connect (openid fhirUser), refresh tokens (offline_access), and patient context (launch/patient) are required.

FHIR Endpoint (required) *

URL of the FHIR endpoint used by SMART applications

`https://secure.server.fire.ly`

Standalone Launch Credentials (required) *

Auth Type (required) *

Confidential Symmetric

☒ Populate fields from discovery 

Scopes (required) *

OAuth 2.0 scopes needed to enable all required functionality

launch/patient openid fhirUser offline_access
patient/Medication.read patient/AllergyIntolerance.read
patient/CarePlan.read patient/CareTeam.read
patient/Condition.read patient/Device.read
patient/DiagnosticReport.read
patient/DocumentReference.read patient/Encounter.read
patient/Goal.read patient/Immunization.read
patient/Location.read patient/MedicationRequest.read
patient/Observation.read patient/Organization.read
patient/Patient.read patient/Practitioner.read
patient/Procedure.read patient/Provenance.read
patient/PractitionerRole.read

Client ID (required) *

Client ID provided during registration of Inferno

`Inferno`

Client Secret (required) *

Client secret provided during registration of Inferno

`secret`

Proof Key for Code Exchange (PKCE)

☒ Enabled ☐ Disabled

PKCE Code Challenge Method

☒ S256 ☐ Plain

Authorization Request Method

☒ GET ☐ POST

3. Click **SUBMIT**
4. A **User Action Required** dialog opens:

User Action Required

Standalone Patient App - Full Access

[Follow this link to authorize with the SMART server.](#)

Tests will resume once Inferno receives a request at
<https://inferno.healthit.gov/suites/custom/smart/redirect> with
a state of f77913fc-20f2-4275-a616-90fec4913439.

CANCEL

5. Click on **Follow this link to authorize with the SMART server**. This will redirect you to Firely Identity Server
6. Log in to Firely Identity Server using these credentials:

Field	Value
Username	fa_alice@fire.ly
Password	AlicePassword1!



Log in

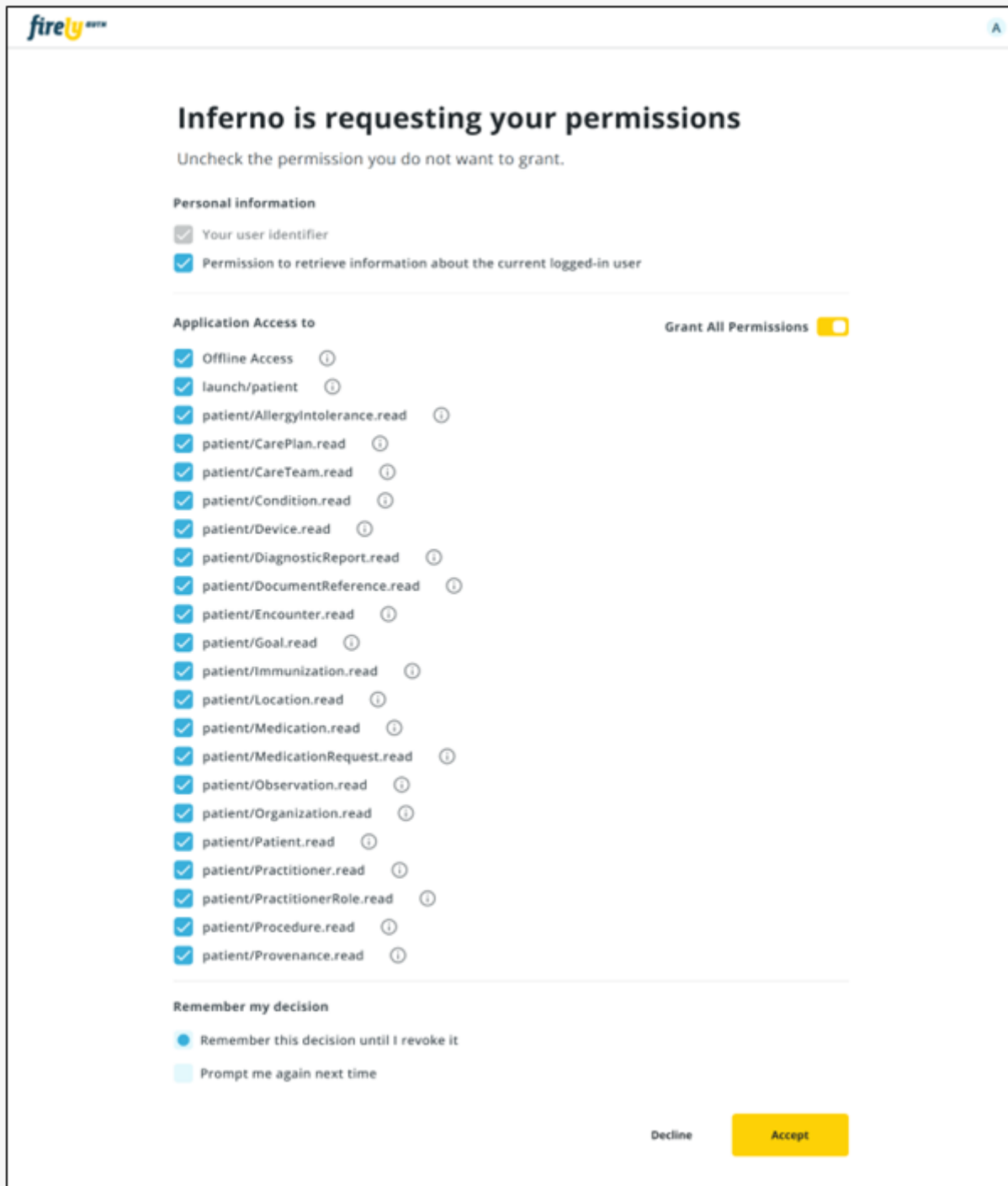
Please login to Firely Auth

👁

☐ Remember me
[Forgot Password?](#)

Log In

7. Press **Login**
8. A dialog **Inferno is requesting your permission** opens
9. Make sure all the boxes are checked:



Inferno is requesting your permissions

Uncheck the permission you do not want to grant.

Personal information

- ☒ Your user identifier
- ☒ Permission to retrieve information about the current logged-in user

Application Access to Grant All Permissions ☒

- ☒ Offline Access ⓘ
- ☒ launch/patient ⓘ
- ☒ patient/AllergyIntolerance.read ⓘ
- ☒ patient/CarePlan.read ⓘ
- ☒ patient/CareTeam.read ⓘ
- ☒ patient/Condition.read ⓘ
- ☒ patient/Device.read ⓘ
- ☒ patient/DiagnosticReport.read ⓘ
- ☒ patient/DocumentReference.read ⓘ
- ☒ patient/Encounter.read ⓘ
- ☒ patient/Goal.read ⓘ
- ☒ patient/Immunization.read ⓘ
- ☒ patient/Location.read ⓘ
- ☒ patient/Medication.read ⓘ
- ☒ patient/MedicationRequest.read ⓘ
- ☒ patient/Observation.read ⓘ
- ☒ patient/Organization.read ⓘ
- ☒ patient/Patient.read ⓘ
- ☒ patient/Practitioner.read ⓘ
- ☒ patient/PractitionerRole.read ⓘ
- ☒ patient/Procedure.read ⓘ
- ☒ patient/Provenance.read ⓘ

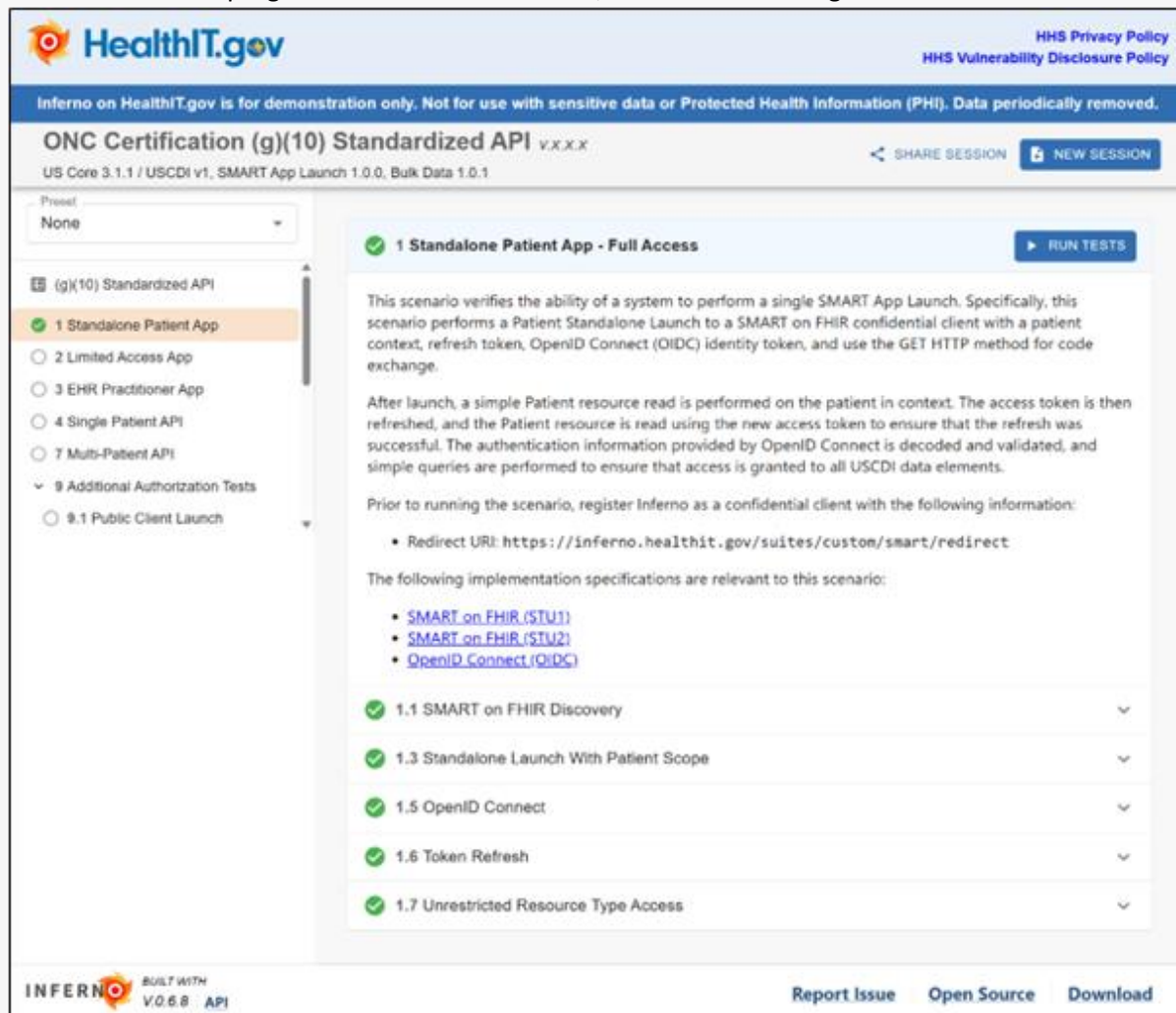
Remember my decision

- ☒ Remember this decision until I revoke it
- ☐ Prompt me again next time

Decline Accept

10. Click **Accept**
11. When you return to Inferno, a progress bar shows in the bottom right corner

12. When this progress bar finishes and closes, all tests should be green:



HealthIT.gov HHS Privacy Policy HHS Vulnerability Disclosure Policy

Inferno on HealthIT.gov is for demonstration only. Not for use with sensitive data or Protected Health Information (PHI). Data periodically removed.

ONC Certification (g)(10) Standardized API V.X.X.X

US Core 3.1.1 / USCDI v1, SMART App Launch 1.0.0, Bulk Data 1.0.1

SHARE SESSION NEW SESSION

Presets: None

- (g)(10) Standardized API
 - 1 Standalone Patient App**
 - 2 Limited Access App
 - 3 EHR Practitioner App
 - 4 Single Patient API
 - 7 Multi-Patient API
 - 9 Additional Authorization Tests
 - 9.1 Public Client Launch

1 Standalone Patient App - Full Access RUN TESTS

This scenario verifies the ability of a system to perform a single SMART App Launch. Specifically, this scenario performs a Patient Standalone Launch to a SMART on FHIR confidential client with a patient context, refresh token, OpenID Connect (OIDC) identity token, and use the GET HTTP method for code exchange.

After launch, a simple Patient resource read is performed on the patient in context. The access token is then refreshed, and the Patient resource is read using the new access token to ensure that the refresh was successful. The authentication information provided by OpenID Connect is decoded and validated, and simple queries are performed to ensure that access is granted to all USCDI data elements.

Prior to running the scenario, register Inferno as a confidential client with the following information:

- Redirect URI: <https://inferno.healthit.gov/suites/custom/smart/redirect>

The following implementation specifications are relevant to this scenario:

- [SMART on FHIR \(STU1\)](#)
- [SMART on FHIR \(STU2\)](#)
- [OpenID Connect \(OIDC\)](#)

- 1.1 SMART on FHIR Discovery
- 1.3 Standalone Launch With Patient Scope
- 1.5 OpenID Connect
- 1.6 Token Refresh
- 1.7 Unrestricted Resource Type Access

INFERNO BUILT WITH V.0.6.8 API

Report Issue Open Source Download

6 Standalone Patient App – Limited Access

1. Click **2 Limited Access App** on the testing menu on the left of the screen and click **RUN TESTS**
2. A dialog opens. All the required information should be prefilled:

The purpose of this test is to verify that patient app users can restrict access granted to apps to a limited number of resources. This test can only be run after the Standalone Patient App test, and all other inputs are locked to ensure the same launch configuration in both tests.

Enter which resources the user will grant access to below, and during the launch process only grant access to those resources. Inferno will verify that access granted matches these expectations.

Expected Resource Grant for Limited Access Launch (required) *

the user will only grant access to the following resources during authorization.

patient, condition, observation

Standalone FHIR Endpoint (required) 🔒 *

URL of the FHIR endpoint used by standalone applications

https://secure.server.fire.ly

Standalone Launch Credentials (required) 🔒 *

Auth Type (required) 🔒 *

Confidential Symmetric

☒ Populate fields from discovery 🔒

Scopes (required) 🔒 *

OAuth 2.0 scopes needed to enable all required functionality

launch/patient openid fhirUser offline_access
patient/Medication.read patient/AllergyIntolerance.read
patient/CarePlan.read patient/CareTeam.read
patient/Condition.read patient/Device.read
patient/DiagnosticReport.read
patient/DocumentReference.read patient/Encounter.read
patient/Goal.read patient/Immunization.read
patient/Location.read patient/MedicationRequest.read
patient/Observation.read patient/Organization.read
patient/Patient.read patient/Practitioner.read
patient/Procedure.read patient/Provenance.read
patient/PractitionerRole.read

Client ID (required) 🔒 *

Client ID provided during registration of Inferno

Inferno

Client Secret (required) 🔒 *

Client secret provided during registration of Inferno

secret

Proof Key for Code Exchange (PKCE) 🔒

☒ Enabled ☐ Disabled

PKCE Code Challenge Method 🔒

☒ S256 ☐ Plain

Authorization Request Method 🔒

☒ GET ☐ POST

3. Click **SUBMIT**
4. A **User Action Required** dialog opens:

User Action Required

Standalone Patient App - Limited Access

[Follow this link to authorize with the SMART server.](#)

Tests will resume once Inferno receives a request at
`https://inferno.healthit.gov/suites/custom/smart/redirect` with
a state of `d9421902-85b8-44c8-97f5-82760f33b71b`.

Access should only be granted to the following resources:

- patient
- condition
- observation

CANCEL

5. Click on **Follow this link to authorize with the SMART server**. This will redirect you to Firely Identity Server
6. A dialog **Inferno is requesting your permission** opens
7. Make sure to uncheck all *patient* scopes except **patient/Patient.read**, **patient/Condition.read** and **patient/Observation.read**:

firely

A

Inferno is requesting your permissions

Uncheck the permission you do not want to grant.

Personal information

☒ Your user identifier
 ☒ Permission to retrieve information about the current logged-in user

Application Access to

Grant All Permissions ☐

☐ Offline Access ⓘ
 ☐ launch/patient ⓘ
 ☐ patient/AllergyIntolerance.read ⓘ
 ☐ patient/CarePlan.read ⓘ
 ☐ patient/CareTeam.read ⓘ
 ☒ patient/Condition.read ⓘ
 ☐ patient/Device.read ⓘ
 ☐ patient/DiagnosticReport.read ⓘ
 ☐ patient/DocumentReference.read ⓘ
 ☐ patient/Encounter.read ⓘ
 ☐ patient/Goal.read ⓘ
 ☐ patient/Immunization.read ⓘ
 ☐ patient/Location.read ⓘ
 ☐ patient/Medication.read ⓘ
 ☐ patient/MedicationRequest.read ⓘ
 ☒ patient/Observation.read ⓘ
 ☐ patient/Organization.read ⓘ
 ☒ patient/Patient.read ⓘ
 ☐ patient/Practitioner.read ⓘ
 ☐ patient/PractitionerRole.read ⓘ
 ☐ patient/Procedure.read ⓘ
 ☐ patient/Provenance.read ⓘ

Remember my decision

☒ Remember this decision until I revoke it
 ☐ Prompt me again next time

Decline

Accept

8. Click **Accept**

9. When you return to Inferno, a progress bar shows in the bottom right corner

10. When this progress bar finishes and closes, all tests should be green:



[HHS Privacy Policy](#)
[HHS Vulnerability Disclosure Policy](#)

Inferno on HealthIT.gov is for demonstration only. Not for use with sensitive data or Protected Health Information (PHI). Data periodically

ONC Certification (g)(10) Standardized API V.X.X.X

US Core 3.1.1 / USCDI v1, SMART App Launch 1.0.0, Bulk Data 1.0.1

[NEW SESSION](#)

Preset
None

(g)(10) Standardized API

1 Standalone Patient App

2 Limited Access App

3 EHR Practitioner App

4 Single Patient API

7 Multi-Patient API

9 Additional Tests

9.1 SMART Public Client Launch

9.3 Token Revocation

9.4 SMART Invalid AUD Launch

9.5 SMART Invalid Token Request

9.8 EHR Launch with Patient Scopes

9.10 Visual Inspection

Report

Configuration Messages

2 Standalone Patient App - Limited Access

This scenario demonstrates the ability to perform a Patient Standalone Launch to a SMART on FHIR confidential client with limited access granted to the app based on user input. The tester is expected to grant the application access to a subset of desired resource types. The launch is performed using the same app configuration as in the Standalone Patient App test, demonstrating that the user has control over what scopes are granted to the app as required in the (g)(10) Standardized API criterion.

- SMART on FHIR (STU1)
- SMART on FHIR (STU2)

2.1 Standalone Launch With Limited Scope

2.3 Restricted Resource Type Access

[RUN TESTS](#)


BUILT WITH
V.0.4.38 API

[Report Issue](#)
[Open Source](#)
[Download](#)

7 EHR Practitioner App

- Click **3 EHR Practitioner App** on the testing menu on the left of the screen and click **RUN TESTS**
- Besides prefilled values, fill in these values:

Field	Value
EHR Launch Client ID	Inferno
EHR Launch Client Secret	secret

Register Inferno as an EHR-launched application using the following information:

- Launch URI:
`https://inferno.healthit.gov/suites/custom/smart/launch`
- Redirect URI:
`https://inferno.healthit.gov/suites/custom/smart/redirect`

Enter in the appropriate scope to enable user-level access to all relevant resources. If using SMART v2, v2-style scopes must be used. In addition, support for the OpenID Connect (openid fhirUser), refresh tokens (offline_access), and EHR context (launch) are required. This test expects that the EHR will launch the application with a patient context.

After submit is pressed, Inferno will wait for the system under test to launch the application.

FHIR Endpoint (required) *

URL of the FHIR endpoint used by SMART applications

`https://secure.server.fire.ly`

EHR Launch Credentials (required) *

Auth Type (required) *

Confidential Symmetric

☒ Populate fields from discovery 

Scopes (required) *

OAuth 2.0 scopes needed to enable all required functionality

launch openid fhirUser offline_access user/Medication.read
user/AllergyIntolerance.read user/CarePlan.read
user/CareTeam.read user/Condition.read user/Device.read
user/DiagnosticReport.read user/DocumentReference.read
user/Encounter.read user/Goal.read user/Immunization.read
user/Location.read user/MedicationRequest.read
user/Observation.read user/Organization.read
user/Patient.read user/Practitioner.read user/Procedure.read
user/Provenance.read user/PractitionerRole.read

Client ID (required) *

Client ID provided during registration of Inferno

Inferno

Client Secret (required) *

Client secret provided during registration of Inferno

secret

Proof Key for Code Exchange (PKCE)

☒ Enabled ☐ Disabled

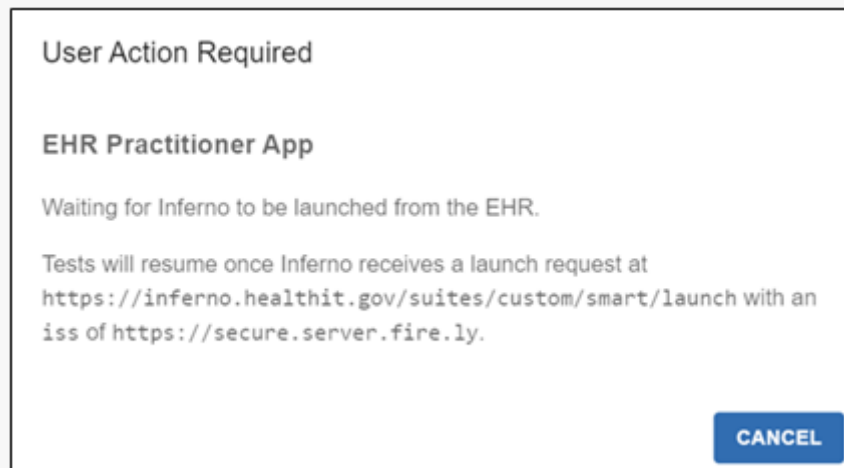
PKCE Code Challenge Method

☒ S256 ☐ Plain

Authorization Request Method

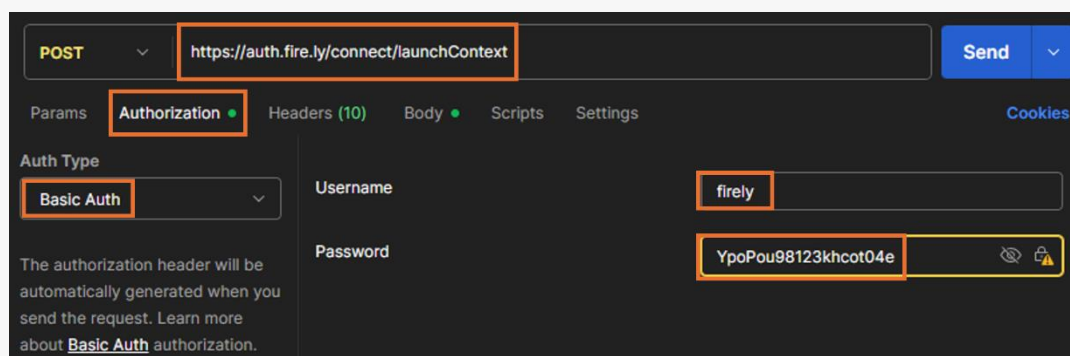
☒ GET ☐ POST

3. Click **SUBMIT**
4. A **User Action Required** dialog opens:



5. Now open Postman and open a new tab for making a request
6. Select the **Authorization** tab
7. Fill in/select the following values:

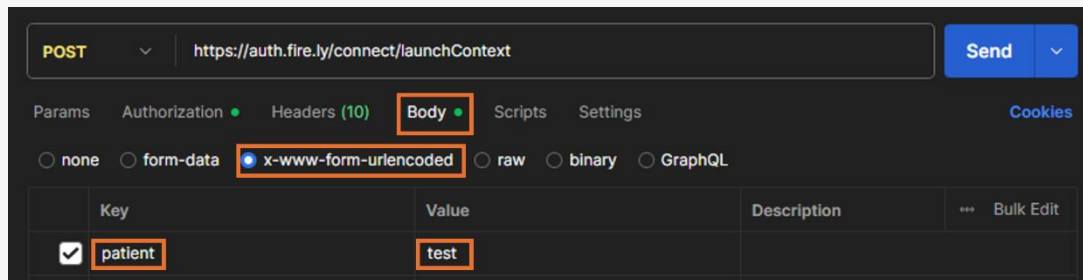
Field	Value
HTTP verb	POST
Request URL	https://auth.fire.ly/connect/launchContext
Type	Basic Auth
Username	firely
Password	YpoPou98123khcot04e



8. Select the **Body** tab
9. Select **x-www-form-urlencoded**

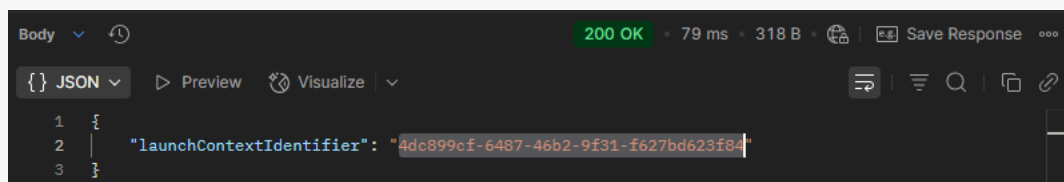
10. Fill in/select the following values:

KEY	VALUE
patient	test



11. Click **Send**

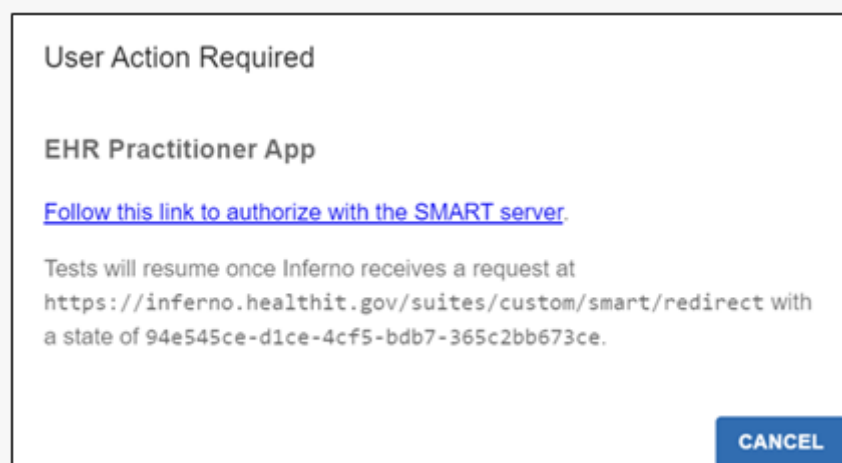
12. The response should be a **200 OK**



13. Open a new web browser tab and go to the following URL containing using the **launchContextIdentifier** value received (this value will be refreshed with each new request):

<https://inferno.healthit.gov/suites/custom/smart/launch?iss=https://secure.server.fire.ly&launch=4dc899cf-6487-46b2-9f31-f627bd623f84>

14. Another **User Action Required** dialog opens:



15. Click on **Follow this link to authorize with the SMART server**. This will redirect you to Firely Identity Server
16. A dialog **Inferno is requesting your permission** opens
17. Make sure all the boxes are checked:


A

Inferno is requesting your permissions

Uncheck the permission you do not want to grant.

Personal information

☒ Your user identifier

☒ Permission to retrieve information about the current logged-in user

Application Access to Grant All Permissions ☒

- ☒ Offline Access ⓘ
- ☒ launch
- ☒ user/AllergyIntolerance.read ⓘ
- ☒ user/CarePlan.read ⓘ
- ☒ user/CareTeam.read ⓘ
- ☒ user/Condition.read ⓘ
- ☒ user/Device.read ⓘ
- ☒ user/DiagnosticReport.read ⓘ
- ☒ user/DocumentReference.read ⓘ
- ☒ user/Encounter.read ⓘ
- ☒ user/Goal.read ⓘ
- ☒ user/Immunization.read ⓘ
- ☒ user/Location.read ⓘ
- ☒ user/Medication.read ⓘ
- ☒ user/MedicationRequest.read ⓘ
- ☒ user/Observation.read ⓘ
- ☒ user/Organization.read ⓘ
- ☒ user/Patient.read ⓘ
- ☒ user/Practitioner.read ⓘ
- ☒ user/PractitionerRole.read ⓘ
- ☒ user/Procedure.read ⓘ
- ☒ user/Provenance.read ⓘ

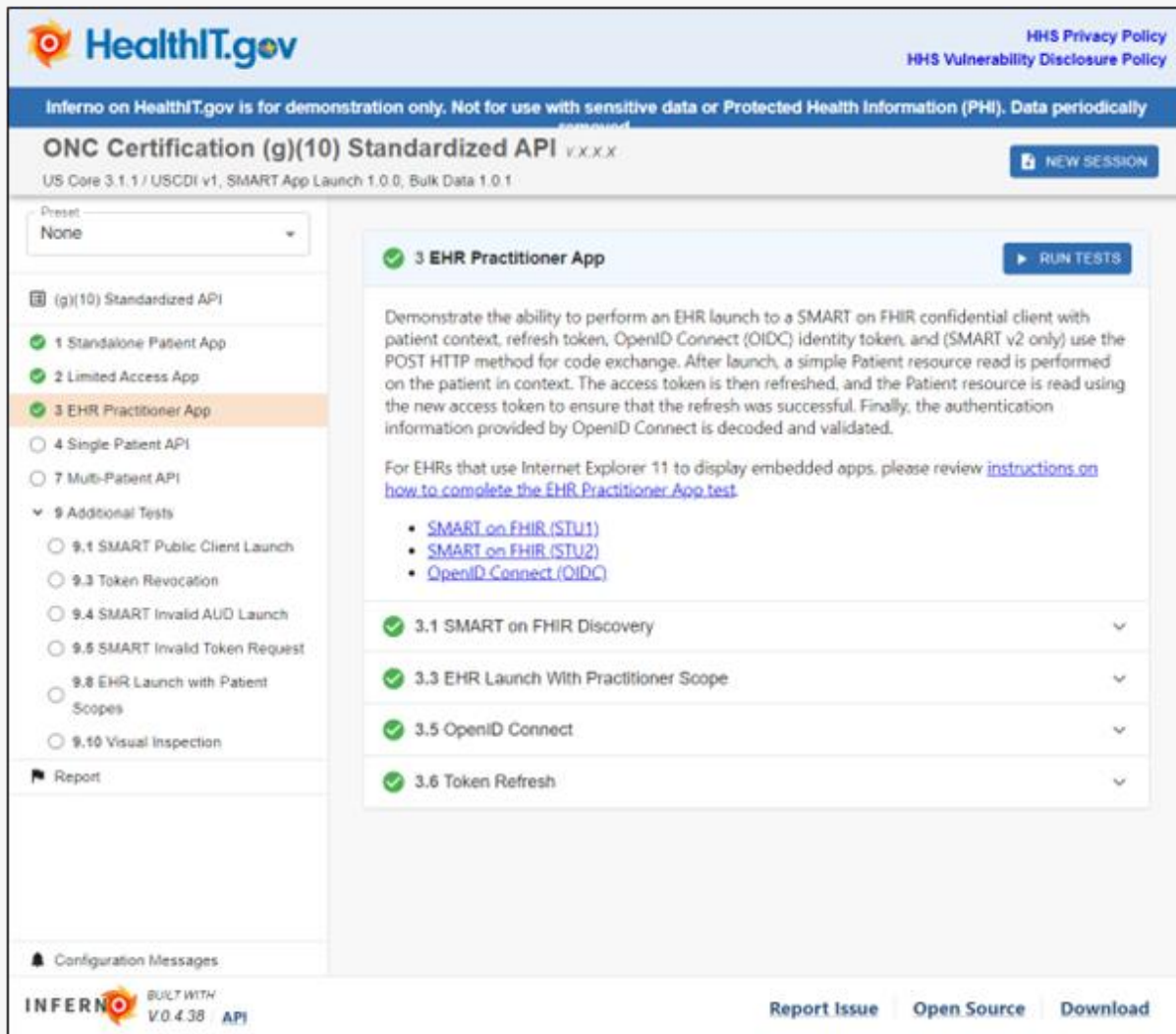
Remember my decision

☒ Remember this decision until I revoke it

☐ Prompt me again next time

Decline
Accept

18. Click **Accept**
19. When you return to Inferno, a progress bar shows in the bottom right corner
20. When this progress bar finishes and closes, all tests should be green:



The screenshot shows the HealthIT.gov Inferno interface for testing the ONC Certification (g)(10) Standardized API. The interface is divided into several sections:

- Header:** HealthIT.gov logo, HHS Privacy Policy, and HHS Vulnerability Disclosure Policy.
- Disclaimer:** Inferno on HealthIT.gov is for demonstration only. Not for use with sensitive data or Protected Health Information (PHI). Data periodically updated.
- Test Suite:** ONC Certification (g)(10) Standardized API V.X.X.X. US Core 3.1.1 / USCDI v1, SMART App Launch 1.0.0, Bulk Data 1.0.1. A "NEW SESSION" button is available.
- Left Sidebar:** A list of tests under the "(g)(10) Standardized API" category. The tests are:
 - 1 Standalone Patient App
 - 2 Limited Access App
 - 3 EHR Practitioner App (selected)
 - 4 Single Patient API
 - 7 Multi-Patient API
 - 9 Additional Tests
 - 9.1 SMART Public Client Launch
 - 9.3 Token Revocation
 - 9.4 SMART Invalid AUD Launch
 - 9.5 SMART Invalid Token Request
 - 9.8 EHR Launch with Patient Scopes
 - 9.10 Visual Inspection
 - Report
 - Configuration Messages
- Main Content Area:**
 - 3 EHR Practitioner App:** A green checkmark indicates it is ready for testing. A "RUN TESTS" button is present.
 - Description:** Demonstrate the ability to perform an EHR launch to a SMART on FHIR confidential client with patient context, refresh token, OpenID Connect (OIDC) identity token, and (SMART v2 only) use the POST HTTP method for code exchange. After launch, a simple Patient resource read is performed on the patient in context. The access token is then refreshed, and the Patient resource is read using the new access token to ensure that the refresh was successful. Finally, the authentication information provided by OpenID Connect is decoded and validated.
 - Instructions:** For EHRs that use Internet Explorer 11 to display embedded apps, please review [instructions on how to complete the EHR Practitioner App test](#).
 - Related Tests:**
 - SMART on FHIR (STU1)
 - SMART on FHIR (STU2)
 - OpenID Connect (OIDC)
 - Test Results:** A list of related tests with green checkmarks:
 - 3.1 SMART on FHIR Discovery
 - 3.3 EHR Launch With Practitioner Scope
 - 3.5 OpenID Connect
 - 3.6 Token Refresh
- Bottom Bar:**
 - INFERNO:** BUILT WITH V.0.4.38 API
 - Buttons:** Report Issue, Open Source, Download

8 Single Patient API

1. Click **4 Single Patient API** on the testing menu on the left of the screen and click **RUN TESTS**
2. All relevant information should be prefilled. (The value in the **Bearer Token** field will differ from our screenshot.)

Please fill out required fields in order to run the test group.

FHIR Endpoint (required) *

URL of the FHIR endpoint used by SMART applications

`https://secure.server.fire.ly`

Patient ID from SMART App Launch (required) 🔒 *

`test`

Additional Patient IDs

Comma separated list of Patient IDs that together with the Patient ID from the SMART App Launch contain all MUST SUPPORT elements.

Implantable Device Type Code

Enter the code for an Implantable Device type, or multiple codes separated by commas. If blank, Inferno will validate all Device resources against the Implantable Device profile

SMART App Launch Credentials (required) 🔒 *

Auth Type (required) 🔒 *

`Confidential Symmetric`

Access Token (required) 🔒 *

`69442ECCADB3A9BA470AF951EE5C35F3B0E455AD5748A8`

Refresh Token (will automatically refresh if available) 🔒

`0E7AA52270F7734BC04FC0AB21986D359ABBB07C08BF49E`

Client ID 🔒

Client ID provided during registration of Inferno

`Inferno`

Client Secret 🔒

Client secret provided during registration of Inferno

`secret`

Token URL 🔒

URL of the authorization server's token endpoint

`https://auth.fire.ly/connect/token`

Access Token Issue Time 🔒

The time that the access token was issued in iso8601 format

`2025-06-12T11:23:38+00:00`

Token Lifetime 🔒

The lifetime of the access token in seconds

`3600`

3. Click **SUBMIT**
4. A progress bar shows in the bottom right corner
5. These tests take about 5 minutes to complete. Once the progress bar finishes and closes, all tests should be green:



[HHS Privacy Policy](#)
[HHS Vulnerability Disclosure Policy](#)

Info on HealthIT.gov is not for use with sensitive data or Protected Health Information (PHI). Data periodically removed.

ONC Certification (g)(10) Standardized API V.X.X.X

US Core 3.1.1 / USCDI v1, SMART App Launch 1.0.0, Bulk Data 1.0.1

[NEW SESSION](#)

Presets

None

(g)(10) Standardized API

- 1 Standalone Patient App
- 2 Limited Access App
- 3 EHR Practitioner App
- 4 Single Patient API**
- 7 Multi-Patient API
- 9 Additional Tests
 - 9.1 SMART Public Client Launch
 - 9.3 Token Revocation
 - 9.4 SMART Invald AUI Launch
 - 9.9 SMART Invald Token Request
 - 9.8 EHR Launch with Patient Scopes
 - 9.10 Visual Inspection

Report

Configuration Messages

4 Single Patient API (US Core 3.1.1) [RUN TESTS](#)

For each of the relevant USCDI data elements provided in the CapabilityStatement, this test executes the [required/supported searches](#) as defined by the US Core Implementation Guide v3.1.1.

The test begins by searching by one or more patients, with the expectation that the Bearer token provided to the test grants access to all USCDI resources. It uses results returned from that query to generate other queries and checks that the results are consistent with the provided search parameters. It then performs a read on each Resource returned and validates the response against the relevant [profile](#) as currently defined in the US Core Implementation Guide.

All MUST SUPPORT elements must be seen before the test can pass, as well as Data Absent Reason to demonstrate that the server can properly handle missing data. Note that Encounter, Organization and Practitioner resources must be accessible as references in some US Core profiles to satisfy must support requirements, and those references will be validated to their US Core profile. These resources will not be tested for FHIR search support.

- 4.1 Capability Statement
- 4.2 Patient Tests
- 4.3 AllergyIntolerance Tests
- 4.4 CarePlan Tests
- 4.5 CareTeam Tests
- 4.6 Condition Tests
- 4.7 Implantable Device Tests
- 4.8 DiagnosticReport for Report and Note exchange Tests
- 4.9 DiagnosticReport for Laboratory Results Reporting Tests
- 4.10 DocumentReference Tests
- 4.11 Goal Tests
- 4.12 Immunization Tests
- 4.13 MedicationRequest Tests
- 4.14 Smoking Status Observation Tests
- 4.15 Pediatric Weight for Height Observation Tests
- 4.16 Laboratory Result Observation Tests
- 4.17 Pediatric BMI for Age Observation Tests
- 4.18 Pulse Oximetry Tests
- 4.19 Pediatric Head Occipital-Frontal Circumference Percentile Tests
- 4.20 Observation Body Height Tests
- 4.21 Observation Body Temperature Tests
- 4.22 Observation Blood Pressure Tests
- 4.23 Observation Body Weight Tests
- 4.24 Observation Heart Rate Tests
- 4.25 Observation Respiratory Rate Tests
- 4.26 Procedure Tests
- 4.27 Encounter Tests
- 4.28 Organization Tests
- 4.29 Practitioner Tests
- 4.30 Provenance Tests
- 4.31 Clinical Notes Guidance
- 4.32 Missing Data Tests

9 Multi Patient API

1. Click **7 Multi-Patient API** on the testing menu on the left of the screen and click **RUN TESTS**
2. Besides the prefilled values, fill in these values:

Field	Value
Bulk Data FHIR URL	https://secure.server.fire.ly
Backend Services Token Endpoint	https://auth.fire.ly/connect/token
Bulk Data Client ID	Inferno-Bulk
Group ID	test
Patient IDs in exported Group	test, test2
Scopes	system/*.read

Register Inferno as a bulk data client with the following information, and enter the client id and client registration in the appropriate fields. This set of tests only checks the Group export. Enter the group export information in the appropriate box.

Register Inferno with the following JWK Set Uri:

- https://inferno.healthit.gov/suites/custom/g10_certificate/.well-known/jwks.json

Bulk Data FHIR URL (required) *

The URL of the Bulk FHIR server.

<https://secure.server.fire.ly>

Group ID (required) *

The Group ID associated with the group of patients to be exported.

test

Patient IDs in exported Group

Comma separated list of every Patient ID that is in the specified Group. This information is provided by the system under test to verify that data returned matches expectations. Leave blank to not verify Group inclusion.

test, test2

Implantable Device Type Codes in exported Group

Comma separated list of every Implantable Device type that is in the specified Group. This information is provided by the system under test to verify that data returned matches expectations. Leave blank to verify all Device resources against the Implantable Device profile.

Limit validation to a maximum resource count

To validate all, leave blank.

Export Times Out after (1-600) (required) *

While testing, Inferno waits for the server to complete the exporting task. If the calculated totalTime is greater than the timeout value specified here, Inferno bulk client stops testing. Please enter an integer for the maximum wait time in seconds. If timeout is less than 1, Inferno uses default value 180. If the timeout is greater than 600 (10 minutes), Inferno uses the maximum value 600.

180

Multi-Patient API Credentials (required) *

Auth Type (required) *

Backend Services

☐ Populate fields from discovery 

Token URL (required) *

URL of the authorization server's token endpoint

<https://auth.fire.ly/connect/token>

Scopes (required) *

OAuth 2.0 scopes needed to enable all required functionality

system.read

Client ID (required) *

Client ID provided during registration of Inferno

Inferno-Bulk

Encryption Algorithm

☒ ES384 ☐ RS384

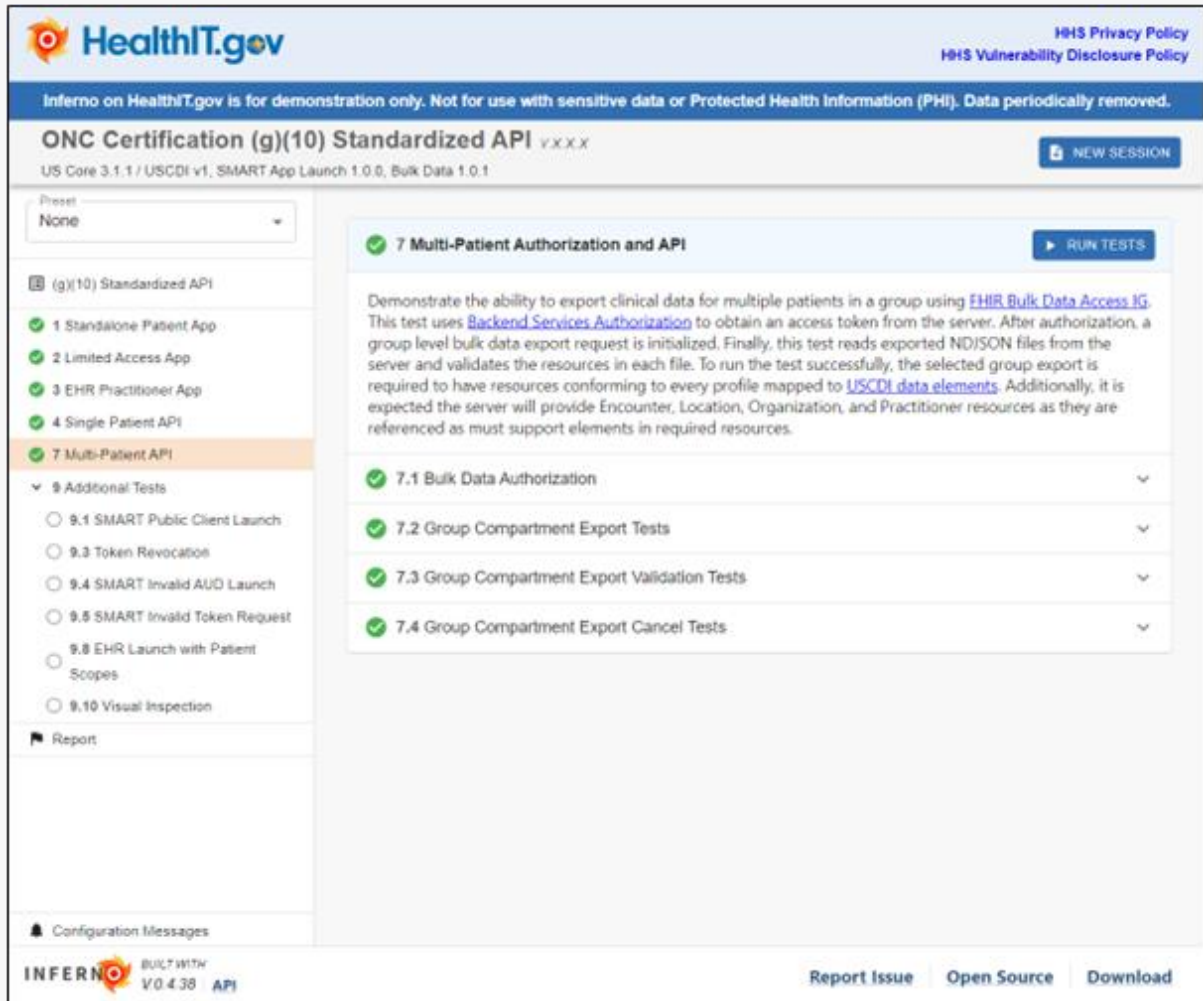
Key ID (kid)

Key ID of the JWKS private key used to sign the client assertion. If blank, the first key for the selected encryption algorithm will be used.

JWKS

The JWKS (including private keys) which will be used to sign the client assertion. If blank, Inferno's default JWKS will be used.

3. Click **SUBMIT**
4. A progress bar shows in the bottom right corner
5. These tests take about 4 minutes to complete. Once the progress bar finishes and closes, all tests should be green:



HealthIT.gov HHS Privacy Policy HHS Vulnerability Disclosure Policy

Inferno on HealthIT.gov is for demonstration only. Not for use with sensitive data or Protected Health Information (PHI). Data periodically removed.

ONC Certification (g)(10) Standardized API V.X.X.X

US Core 3.1.1 / USCDI v1, SMART App Launch 1.0.0, Bulk Data 1.0.1

NEW SESSION

7 Multi-Patient Authorization and API RUN TESTS

Demonstrate the ability to export clinical data for multiple patients in a group using [FHIR Bulk Data Access IG](#). This test uses [Backend Services Authorization](#) to obtain an access token from the server. After authorization, a group level bulk data export request is initialized. Finally, this test reads exported NDJSON files from the server and validates the resources in each file. To run the test successfully, the selected group export is required to have resources conforming to every profile mapped to [USCDI data elements](#). Additionally, it is expected the server will provide Encounter, Location, Organization, and Practitioner resources as they are referenced as must support elements in required resources.

- 7.1 Bulk Data Authorization
- 7.2 Group Compartment Export Tests
- 7.3 Group Compartment Export Validation Tests
- 7.4 Group Compartment Export Cancel Tests

9.1 SMART Public Client Launch

9.2 Token Revocation

9.3 SMART Invalid AUD Launch

9.4 SMART Invalid Token Request

9.5 EHR Launch with Patient Scopes

9.6 Visual Inspection

Report

Configuration Messages

INFERNO BUILT WITH V0.4.38 API

Report Issue Open Source Download

10 Additional Tests

10.1 Public Client Standalone Launch with OpenID Connect

1. Click **9.1 SMART Public Client Launch** on the testing menu on the left of the screen and click **RUN TESTS**
2. Besides the prefilled values, fill in this value:

Field	Value
Public Launch Client ID	Inferno-Public

Register Inferno as a standalone application using the following information:

- Redirect URI:
https://inferno.healthit.gov/suites/custom/smart/redirect

Enter in the appropriate scope to enable patient-level access to all relevant resources. If using SMART v2, v2-style scopes must be used. In addition, support for the OpenID Connect (openid fhirUser), refresh tokens (offline_access), and patient context (launch/patient) are required.

Public Launch FHIR Endpoint (required) *
URL of the FHIR endpoint used by standalone applications
https://secure.server.fire.ly

Public Launch Credentials (required) *

Auth Type (required)  *
Public

☒ **Populate fields from discovery **

Scopes (required) *
OAuth 2.0 scopes needed to enable all required functionality
launch/patient openid fhirUser offline_access
patient/Medication.read patient/AllergyIntolerance.read
patient/CarePlan.read patient/CareTeam.read
patient/Condition.read patient/Device.read
patient/DiagnosticReport.read
patient/DocumentReference.read patient/Encounter.read
patient/Goal.read patient/Immunization.read
patient/Location.read patient/MedicationRequest.read
patient/Observation.read patient/Organization.read
patient/Patient.read patient/Practitioner.read
patient/Procedure.read patient/Provenance.read
patient/PractitionerRole.read

Client ID (required) *
Client ID provided during registration of Inferno
Inferno-Public

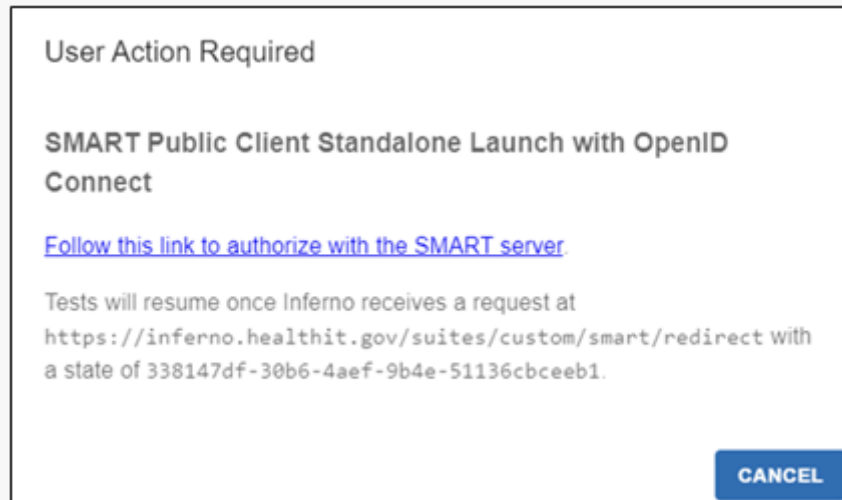
Proof Key for Code Exchange (PKCE)
☒ Enabled ☐ Disabled

PKCE Code Challenge Method
☒ S256 ☐ Plain

Authorization Request Method 
☒ GET ☐ POST

3. Click **SUBMIT**

4. A **User Action Required** dialog opens



5. Click on **Follow this link to authorize with the SMART server**. This will redirect you to Firely Identity Server
6. A dialog **Inferno-Public is requesting your permission** opens
7. Make sure all the boxes are checked:

firely

A

Inferno-Public is requesting your permissions

Uncheck the permission you do not want to grant.

Personal information

☒ Your user identifier

☒ Permission to retrieve information about the current logged-in user

Application Access to Grant All Permissions ☒

- ☒ Offline Access ⓘ
- ☒ launch/patient ⓘ
- ☒ patient/AllergyIntolerance.read ⓘ
- ☒ patient/CarePlan.read ⓘ
- ☒ patient/CareTeam.read ⓘ
- ☒ patient/Condition.read ⓘ
- ☒ patient/Device.read ⓘ
- ☒ patient/DiagnosticReport.read ⓘ
- ☒ patient/DocumentReference.read ⓘ
- ☒ patient/Encounter.read ⓘ
- ☒ patient/Goal.read ⓘ
- ☒ patient/Immunization.read ⓘ
- ☒ patient/Location.read ⓘ
- ☒ patient/Medication.read ⓘ
- ☒ patient/MedicationRequest.read ⓘ
- ☒ patient/Observation.read ⓘ
- ☒ patient/Organization.read ⓘ
- ☒ patient/Patient.read ⓘ
- ☒ patient/Practitioner.read ⓘ
- ☒ patient/PractitionerRole.read ⓘ
- ☒ patient/Procedure.read ⓘ
- ☒ patient/Provenance.read ⓘ

Remember my decision

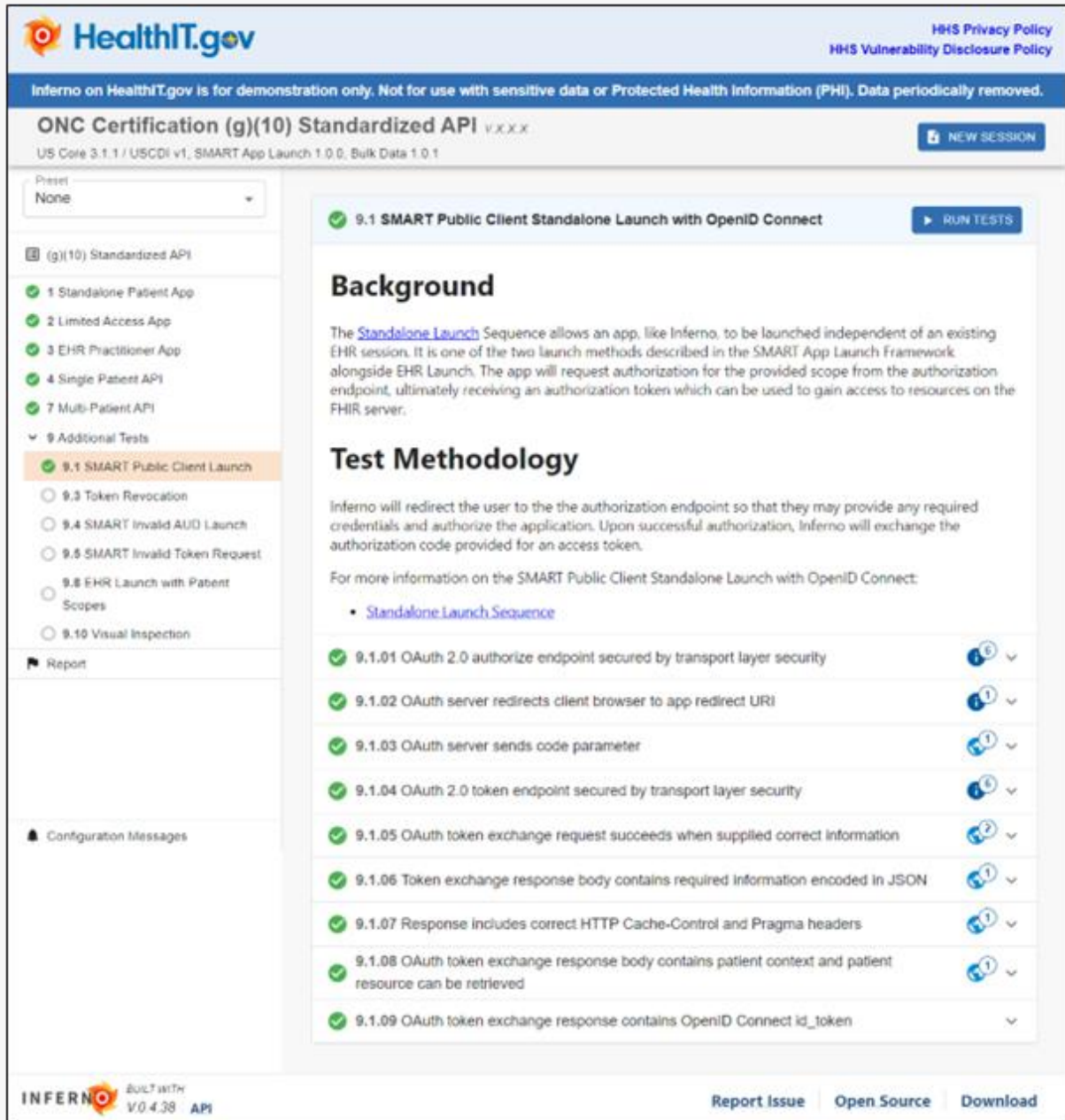
☒ Remember this decision until I revoke it

☐ Prompt me again next time

Decline

Accept

8. Click **Accept**
9. When you return to Inferno, a progress bar shows in the bottom right corner
10. When this progress bar finishes and closes, all tests should be green:



HealthIT.gov HHS Privacy Policy HHS Vulnerability Disclosure Policy

Inferno on HealthIT.gov is for demonstration only. Not for use with sensitive data or Protected Health Information (PHI). Data periodically removed.

ONC Certification (g)(10) Standardized API V.X.X.X

US Core 3.1.1 / USCDI v1, SMART App Launch 1.0.0, Bulk Data 1.0.1 [NEW SESSION](#)

Preset: None

(g)(10) Standardized API

- 1 Standalone Patient App
- 2 Limited Access App
- 3 EHR Practitioner App
- 4 Single Patient API
- 7 Multi-Patient API
- 9 Additional Tests
 - 9.1 SMART Public Client Launch**
 - 9.3 Token Revocation
 - 9.4 SMART Invalid AUD Launch
 - 9.5 SMART Invalid Token Request
 - 9.8 EHR Launch with Patient Scopes
 - 9.10 Visual Inspection

Report

Configuration Messages

9.1 SMART Public Client Standalone Launch with OpenID Connect [RUN TESTS](#)

Background

The [Standalone Launch](#) Sequence allows an app, like Inferno, to be launched independent of an existing EHR session. It is one of the two launch methods described in the SMART App Launch Framework alongside EHR Launch. The app will request authorization for the provided scope from the authorization endpoint, ultimately receiving an authorization token which can be used to gain access to resources on the FHIR server.

Test Methodology

Inferno will redirect the user to the the authorization endpoint so that they may provide any required credentials and authorize the application. Upon successful authorization, Inferno will exchange the authorization code provided for an access token.

For more information on the SMART Public Client Standalone Launch with OpenID Connect:

- [Standalone Launch Sequence](#)

9.1.01 OAuth 2.0 authorize endpoint secured by transport layer security	1/5
9.1.02 OAuth server redirects client browser to app redirect URI	1/1
9.1.03 OAuth server sends code parameter	1/1
9.1.04 OAuth 2.0 token endpoint secured by transport layer security	1/5
9.1.05 OAuth token exchange request succeeds when supplied correct information	1/2
9.1.06 Token exchange response body contains required information encoded in JSON	1/1
9.1.07 Response includes correct HTTP Cache-Control and Pragma headers	1/1
9.1.08 OAuth token exchange response body contains patient context and patient resource can be retrieved	1/1
9.1.09 OAuth token exchange response contains OpenID Connect id_token	

INFERNO BUILT WITH **V.0.4.38 API** [Report Issue](#) [Open Source](#) [Download](#)

10.2 Token Revocation

This test requires you to perform two POSTs to revoke both the bearer and refresh token.

The explanation and screenshots here are based on using [Postman](#).

1. Click **9.3 Token Revocation** on the testing menu on the left of the screen and click **RUN TESTS**
2. Select **Yes** for the 'Prior to executing ... patient standalone launch' option:



Please fill out required fields in order to run the test group.

The Health IT developer demonstrated a patient's request for revoking the tokens provided during the patient standalone launch within the last hour

☒ Yes ☐ No

Notes, if applicable:

Patient ID (required) *

Patient ID associated with revoked tokens provided as context in the patient standalone launch. This will be used to verify access is no longer granted using the revoked token.

test

FHIR Endpoint (required) *

URL of the FHIR endpoint used by standalone applications

https://secure.server.fire.ly

Revoked Bearer Token (required) *

Prior to the test, please revoke this bearer token from patient standalone launch.

Auth Type (required) *

Prior to the test, please revoke this bearer token from patient standalone launch.

Confidential Symmetric

Access Token (required) *

074EDED22FC89E6D869D78E8E0DE3A1575CAB149E65E93

Refresh Token (will automatically refresh if available) (required) *

0E7AA52270F7734BC04FC0AB21986D359ABBB07C08BF49E

Client ID

Client ID provided during registration of Inferno

Inferno

Client Secret

Client secret provided during registration of Inferno

secret

Token URL (required) *

URL of the authorization server's token endpoint

https://auth.fire.ly/connect/token

Access Token Issue Time

The time that the access token was issued in iso8601 format

2025-06-12T12:22:54+00:00

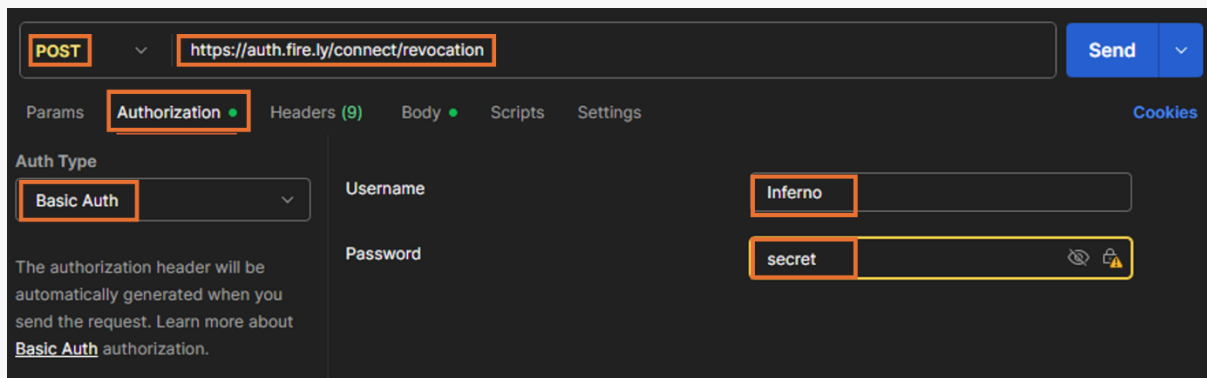
Token Lifetime

The lifetime of the access token in seconds

3600

3. Now open Postman and open a new tab for making a request
4. Select the **Authorization** tab
5. Fill in/select the following values:

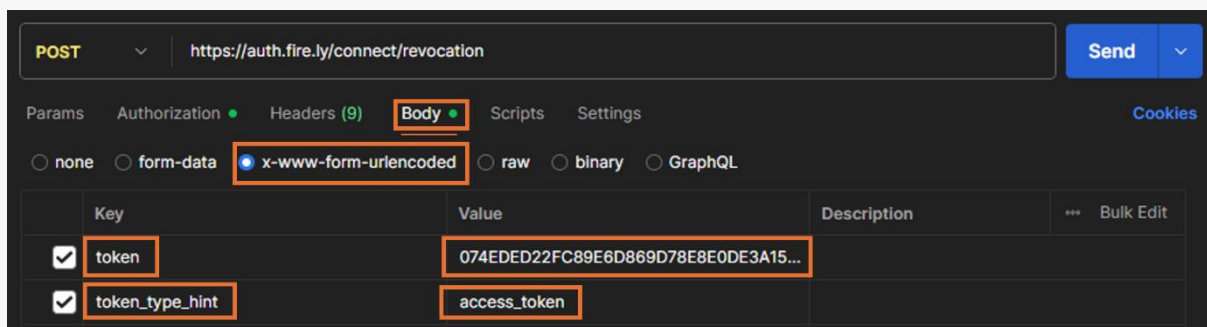
Field	Value
HTTP verb	POST
Request URL	https://auth.fire.ly/connect/revocation
Type	Basic Auth
Username	Inferno
Password	secret



The screenshot shows the Postman interface with the **Authorization** tab selected. The **Auth Type** is set to **Basic Auth**. The **Username** field is filled with **Inferno** and the **Password** field is filled with **secret**. The **Send** button is visible in the top right corner.

6. Select the **Body** tab
7. Select **x-www-form-urlencoded**
8. Fill in/select the following values:

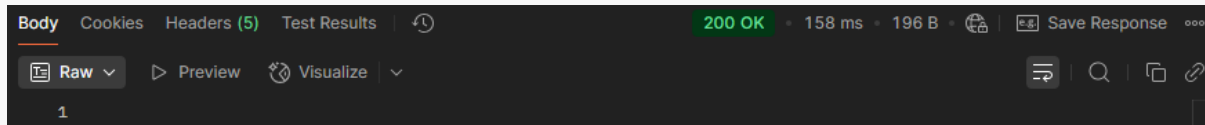
KEY	VALUE
token	Copy/paste the value from the Revoked Bearer Token field in the Inferno dialog
token_type_hint	access_token



The screenshot shows the Postman interface with the **Body** tab selected. The **x-www-form-urlencoded** radio button is selected. The **token** and **token_type_hint** keys are added to the body, with their respective values pasted into the **Value** field. The **Send** button is visible in the top right corner.

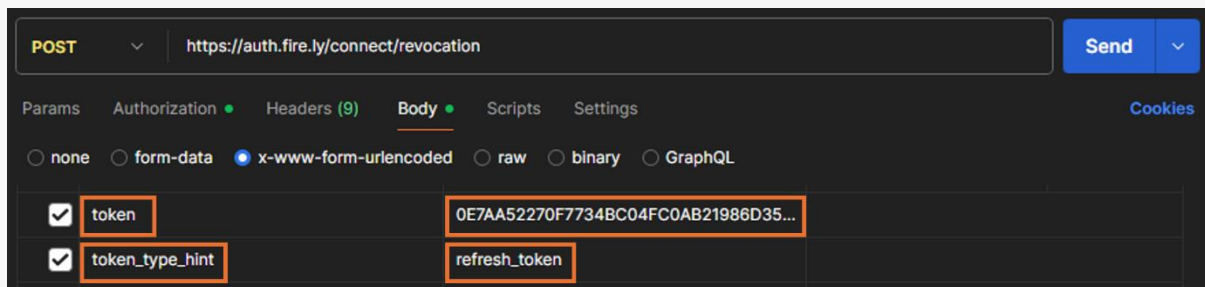
9. Click **Send**

10. The response should be a **200 OK**



11. Change the following values:

KEY	VALUE
token	Copy/paste the value from the Revoked Refresh Token field in the Inferno dialog
token_type_hint	refresh_token



12. Click **Send**. The response should again be a **200 OK**

13. Go back to the Inferno dialog and click **SUBMIT**

14. A progress bar shows in the bottom right corner

15. When this progress bar finishes and closes, all tests should be green:





[HHS Privacy Policy](#)
[HHS Vulnerability Disclosure Policy](#)

Inferno on HealthIT.gov is for demonstration only. Not for use with sensitive data or Protected Health Information (PHI). Data periodically removed.

ONC Certification (g)(10) Standardized API V.X.X.X

[SHARE SESSION](#) [NEW SESSION](#) [Help](#)

Preset
None

(g)(10) Standardized API

1 Standalone Patient App

2 Limited Access App

3 EHR Practitioner App

4 Single Patient API

7 Multi-Patient API

9 Additional Authorization Tests

9.1 Public Client Launch

9.3 Token Revocation

9.4 Invalid AUD Launch

9.5 Invalid Token Request

9.8 EHR Launch with Patient Scopes

11 Visual Inspection

Report

Configuration Messages

9.3 Token Revocation

RUN TESTS

This scenario verifies the ability of the system to revoke access granted to an application at the direction of a patient. Access to the application must be revoked within one hour of the patient's request.

9.3.01 Health IT developer demonstrated the ability of the Health IT Module to revoke tokens within one hour of a patient's request.

9.3.02 Access to Patient resource returns unauthorized after token revocation.

9.3.03 Token refresh fails after token revocation.

INFERNO BUILT WITH V.0.6.11 API

[Report Issue](#) [Open Source](#) [Download](#)

10.3 SMART App Launch Error: Invalid AUD Parameter

1. Click **9.4 SMART Invalid AUD Launch** on the testing menu on the left of the screen and click **RUN TESTS**
2. A dialog opens with all relevant information prefilled:

Register Inferno as a standalone application using the following information:

- Redirect URI:
https://inferno.healthit.gov/suites/custom/smart/redirect

Standalone FHIR Endpoint (required) *
URL of the FHIR endpoint used by standalone applications
https://secure.server.fire.ly

Standalone Launch Credentials (required) *

Auth Type (required)  *
Confidential Symmetric

☒ Populate fields from discovery 

Scopes (required) *
OAuth 2.0 scopes needed to enable all required functionality

launch/patient openid fhirUser offline_access
patient/Medication.read patient/AllergyIntolerance.read
patient/CarePlan.read patient/CareTeam.read
patient/Condition.read patient/Device.read
patient/DiagnosticReport.read
patient/DocumentReference.read patient/Encounter.read
patient/Goal.read patient/Immunization.read
patient/Location.read patient/MedicationRequest.read
patient/Observation.read patient/Organization.read
patient/Patient.read patient/Practitioner.read
patient/Procedure.read patient/Provenance.read
patient/PractitionerRole.read

Client ID (required) *
Client ID provided during registration of Inferno
Inferno

Client Secret (required) *
Client secret provided during registration of Inferno
secret

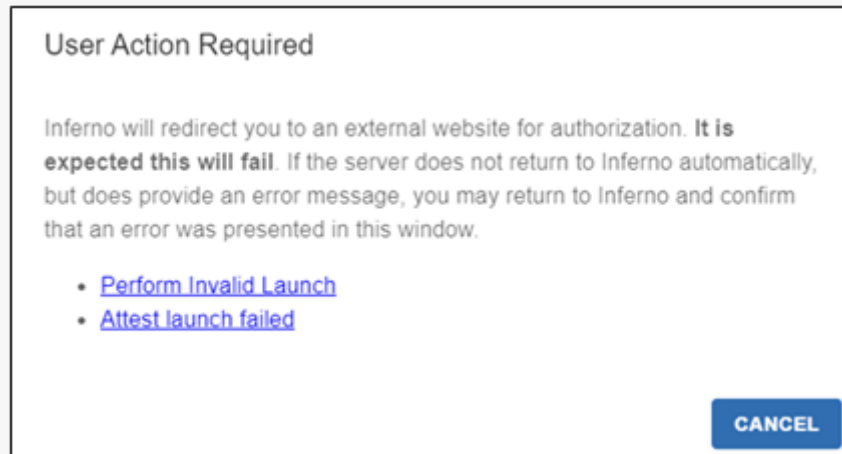
Proof Key for Code Exchange (PKCE)
☒ Enabled ☐ Disabled

PKCE Code Challenge Method
☒ S256 ☐ Plain

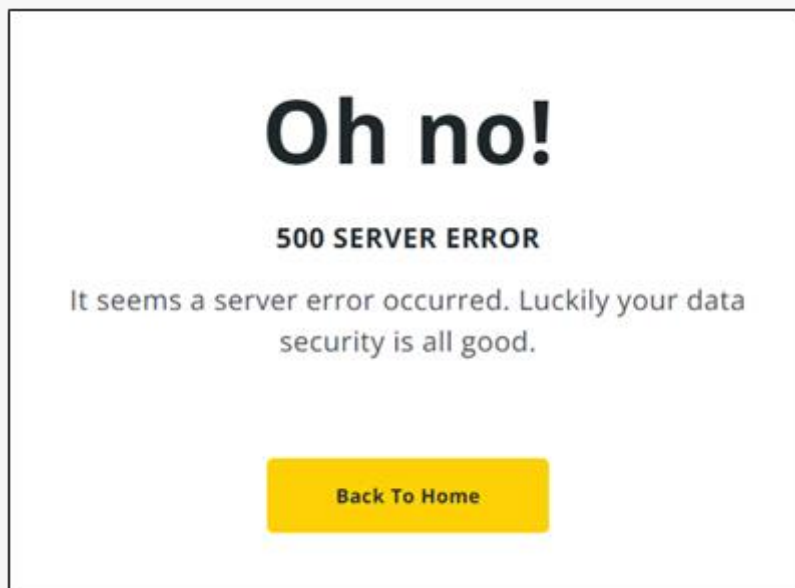
Authorization Request Method 
☒ GET ☐ POST

3. Click **SUBMIT**

4. A **User Action Required** dialog opens:



5. Right click **Perform Invalid Launch** and select **Open link in new tab**
6. A login attempt to Firely Identity Server will open in a new tab, this should fail:



7. Return to the Inferno browser tab from step 4
8. Click **Attest Launch Failed**
9. A progress bar shows in the bottom right corner
10. When this progress bar finishes and closes, all tests should be green:



[HHS Privacy Policy](#)
[HHS Vulnerability Disclosure Policy](#)

Inferno on HealthIT.gov is not for use with sensitive data or Protected Health Information (PHI). Data periodically removed.

ONC Certification (g)(10) Standardized API V.X.X.X
US Core 3.1.1 / USCDI v1, SMART App Launch 1.0.0, Bulk Data 1.0.1

NEW SESSION

Preset
None

(g)(10) Standardized API

1 Standalone Patient App
2 Limited Access App
3 EHR Practitioner App
4 Single Patient API
7 Multi-Patient API
9 Additional Tests
9.1 SMART Public Client Launch
9.3 Token Revocation
9.4 SMART Invalid AUD Launch
9.5 SMART Invalid Token Request
9.8 EHR Launch with Patient Scopes
9.10 Visual Inspection

Report

Configuration Messages

9.4 SMART App Launch Error: Invalid AUD Parameter

RUN TESTS

Background

The Invalid AUD Sequence verifies that a SMART Launch Sequence, specifically the Standalone Launch Sequence, does not work in the case where the client sends an invalid FHIR server as the aud parameter during launch. This must fail to ensure that a genuine bearer token is not leaked to a counterfeit resource server.

This test is not included as part of a regular SMART Launch Sequence because it requires the browser of the user to be redirected to the authorization service, and there is no expectation that the authorization service redirects the user back to Inferno with an error message. The only requirement is that Inferno is not granted a code to exchange for a valid access token. Since this is a special case, it is tested independently in a separate sequence.

Note that this test will launch a new browser window. The user is required to 'Attest' in the Inferno user interface after the launch does not succeed, if the server does not return an error code.

- [Standalone Launch Sequence \(STU1\)](#)
- [Standalone Launch \(STU2\)](#)

9.4.01 OAuth server redirects client browser to app redirect URI

9.4.03 Inferno client app does not receive code parameter redirect URI
Tester attested that the authorization service did not succeed due to invalid AUD parameter


BUILT WITH
V.0.4.38 API

[Report Issue](#)
[Open Source](#)
[Download](#)

10.4 SMART App Launch Error: Invalid Access Token Request

- Click **9.5 SMART Invalid Token Request** on the testing menu on the left of the screen and click **RUN TESTS**
- A dialog opens with all relevant information prefilled:

Register Inferno as a standalone application using the following information:

- Redirect URI:
https://inferno.healthit.gov/suites/custom/smart/redirect

Standalone FHIR Endpoint (required) *
URL of the FHIR endpoint used by standalone applications
https://secure.server.fire.ly

Standalone Launch Credentials (required) *

Auth Type (required)  *
Confidential Symmetric 

☒ **Populate fields from discovery **

Scopes (required) *
OAuth 2.0 scopes needed to enable all required functionality

launch/patient openid fhirUser offline_access
patient/Medication.read patient/AllergyIntolerance.read
patient/CarePlan.read patient/CareTeam.read
patient/Condition.read patient/Device.read
patient/DiagnosticReport.read
patient/DocumentReference.read patient/Encounter.read
patient/Goal.read patient/Immunization.read
patient/Location.read patient/MedicationRequest.read
patient/Observation.read patient/Organization.read
patient/Patient.read patient/Practitioner.read
patient/Procedure.read patient/Provenance.read
patient/PractitionerRole.read

Client ID (required) *
Client ID provided during registration of Inferno

Inferno

Client Secret (required) *
Client secret provided during registration of Inferno

secret

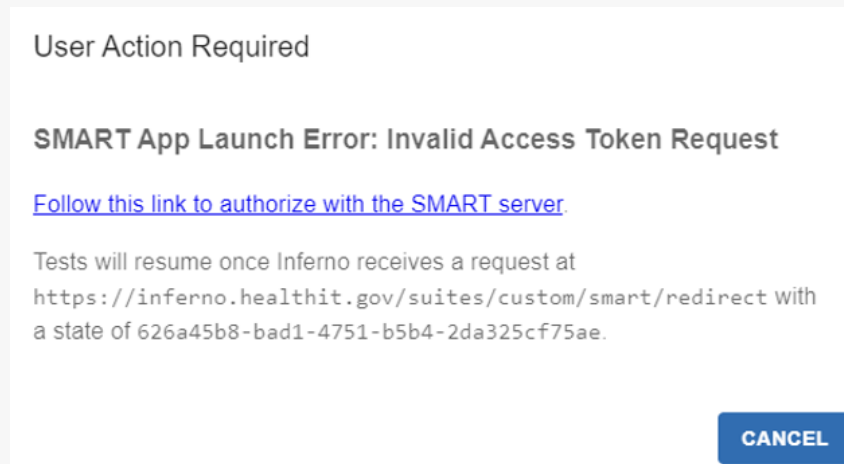
Proof Key for Code Exchange (PKCE)
☒ Enabled ☐ Disabled

PKCE Code Challenge Method
☒ S256 ☐ Plain

Authorization Request Method 
☒ GET ☐ POST

3. Click **SUBMIT**

4. A **User Action Required** dialog opens



5. Click on **Follow this link to authorize with the SMART server**. This will redirect you to Firely Identity Server
6. A dialog **Inferno is requesting your permission** opens
7. Make sure all the boxes are checked:

firely

A

Inferno is requesting your permissions

Uncheck the permission you do not want to grant.

Personal information

☒ Your user identifier
 ☒ Permission to retrieve information about the current logged-in user

Application Access to

☒ Offline Access ⓘ
 ☒ launch/patient ⓘ
 ☒ patient/AllergyIntolerance.read ⓘ
 ☒ patient/CarePlan.read ⓘ
 ☒ patient/CareTeam.read ⓘ
 ☒ patient/Condition.read ⓘ
 ☒ patient/Device.read ⓘ
 ☒ patient/DiagnosticReport.read ⓘ
 ☒ patient/DocumentReference.read ⓘ
 ☒ patient/Encounter.read ⓘ
 ☒ patient/Goal.read ⓘ
 ☒ patient/Immunization.read ⓘ
 ☒ patient/Location.read ⓘ
 ☒ patient/Medication.read ⓘ
 ☒ patient/MedicationRequest.read ⓘ
 ☒ patient/Observation.read ⓘ
 ☒ patient/Organization.read ⓘ
 ☒ patient/Patient.read ⓘ
 ☒ patient/Practitioner.read ⓘ
 ☒ patient/PractitionerRole.read ⓘ
 ☒ patient/Procedure.read ⓘ
 ☒ patient/Provenance.read ⓘ

Grant All Permissions

☐

Remember my decision

☒ Remember this decision until I revoke it
 ☐ Prompt me again next time

Decline

Accept

8. Click **Accept**

9. When you return to Inferno, a progress bar shows in the bottom right corner

10. When this progress bar finishes and closes, all tests should be green:



[HHS Privacy Policy](#)
[HHS Vulnerability Disclosure Policy](#)

Inferno on HealthIT.gov is not for use with sensitive data or Protected Health Information (PHI). Data periodically removed.

ONC Certification (g)(10) Standardized API V.X.X.X
US Core 3.1.1 / USCDI v1, SMART App Launch 1.0.0, Bulk Data 1.0.1

NEW SESSION

Preset
None

(g)(10) Standardized API

1 Standalone Patient App
2 Limited Access App
3 EHR Practitioner App
4 Single Patient API
7 Multi-Patient API
9 Additional Tests
9.1 SMART Public Client Launch
9.3 Token Revocation
9.4 SMART Invalid AUD Launch
9.5 SMART Invalid Token Request
9.8 EHR Launch with Patient Scopes
9.10 Visual Inspection

Report

Configuration Messages

9.5 SMART App Launch Error: Invalid Access Token Request

RUN TESTS

Background

The Invalid Access Token Request Sequence verifies that a SMART Launch Sequence, specifically the [Standalone Launch](#) Sequence, does not work in the case where the client sends an invalid Authorization code or client ID during the code exchange step. This must not result in a successful launch.

This test is not included as part of a regular SMART Launch Sequence because some servers may not accept an authorization code after it has been used unsuccessfully in this manner.

9.5.01 OAuth server redirects client browser to app redirect URI
9.5.02 OAuth server sends code parameter
9.5.03 OAuth token exchange fails when supplied invalid code
9.5.04 OAuth token exchange fails when supplied invalid client ID


BUILT WITH
V.0.4.38 | API

[Report Issue](#)
[Open Source](#)
[Download](#)

10.5 EHR Launch with Patient Scopes

- Click **9.8 EHR Launch with Patient Scopes** on the testing menu on the left of the screen and click **RUN TESTS**
- Besides prefilled values, fill in these values:

Field	Value
EHR Launch Client ID	Inferno
EHR Launch Client Secret	secret

Please fill out required fields in order to run the test group.

EHR Launch FHIR Endpoint (required) *
 URL of the FHIR endpoint used by EHR launched applications
 https://secure.server.fire.ly

EHR Launch with Patient Scopes Credentials (required) *

Auth Type (required) 🔒 *
 Confidential Symmetric

☒ Populate fields from discovery 🔒

Scopes (required) 🔒 *
 OAuth 2.0 scopes needed to enable all required functionality
 launch openid fhirUser offline_access patient/Patient.read

Client ID (required) *
 Client ID provided during registration of Inferno
 Inferno

Client Secret (required) *
 Client secret provided during registration of Inferno
 secret

Proof Key for Code Exchange (PKCE)
☒ Enabled ☐ Disabled

PKCE Code Challenge Method
☒ S256 ☐ Plain

Authorization Request Method 🔒
☒ GET ☐ POST

3. Click **SUBMIT**
4. A **User Action Required** dialog opens

User Action Required

Additional Tests

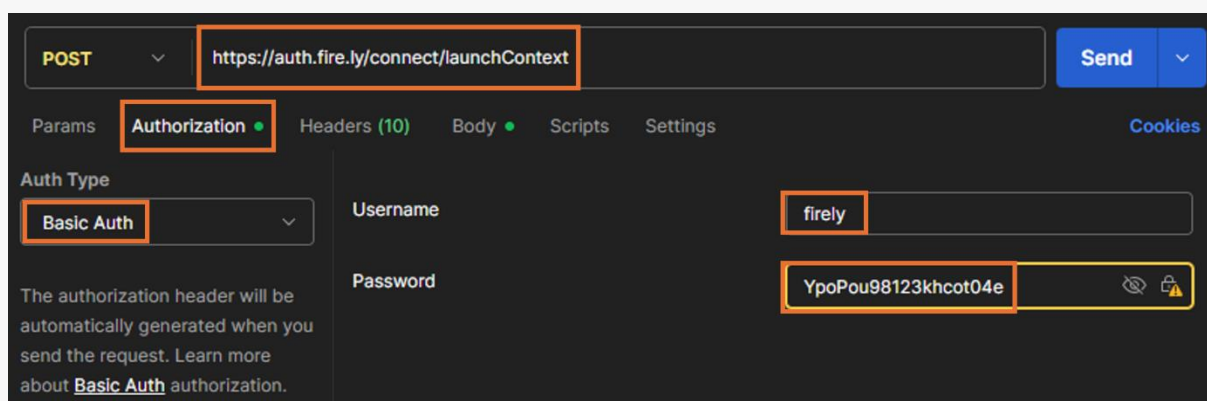
Waiting for Inferno to be launched from the EHR.

Tests will resume once Inferno receives a launch request at <https://inferno.healthit.gov/suites/custom/smart/launch> with an iss of <https://secure.server.fire.ly>.

CANCEL

5. Now open Postman and open a new tab for making a request
6. Select the **Authorization** tab
7. Fill in/select the following values:

Field	Value
HTTP verb	POST
Request URL	https://auth.fire.ly/connect/launchContext
Type	Basic Auth
Username	firely
Password	YpoPou98123kxcot04e



The screenshot shows the Postman interface with the following configuration:

- Method:** POST
- URL:** <https://auth.fire.ly/connect/launchContext>
- Tab:** Authorization
- Auth Type:** Basic Auth
- Username:** firely
- Password:** YpoPou98123kxcot04e

The authorization header will be automatically generated when you send the request. Learn more about [Basic Auth](#) authorization.

8. Select the **Body** tab
9. Select **x-www-form-urlencoded**
10. Fill in/select the following values:

KEY	VALUE
-----	-------

patient	test
---------	------

POST ▼ https://auth.fire.ly/connect/launchContext Send ▼

Params Authorization Headers (10) **Body** • Scripts Settings Cookies

☐ none ☐ form-data ☒ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

	Key	Value	Description	...	Bulk Edit
☒	patient	test			

11. Press **Send**

12. The response should be a **200 OK**

Body Cookies Headers (6) Test Results ↺ 200 OK 64 ms 318 B 🌐 📄 Save Response ...

{} **JSON** ▼ ▶ Preview 🔍 Visualize ▼

```

1  {
2    "launchContextIdentifier": "3fc36f3a-aa4d-4ce1-a16d-dd59dccaee1"
3  }
  
```

13. Open a new web browser tab and go to the following URL containing using the **launchContextIdentifier** value received (this value is refreshed with each new request):

<https://inferno.healthit.gov/suites/custom/smart/launch?iss=https://secure.server.fire.ly&launch=3fc36f3a-aa4d-4ce1-a16d-dd59dccaee1>

Another **User Action Required** dialog opens:

User Action Required

Additional Tests

[Follow this link to authorize with the SMART server.](#)

Tests will resume once Inferno receives a request at
<https://inferno.healthit.gov/suites/custom/smart/redirect> with
 a state of 8525c18d-cc78-4fa8-819b-93ad0551d49e.

CANCEL

14. Click on **Follow this link to authorize with the SMART server**. This will redirect you to Firely Identity Server

15. A dialog **Inferno is requesting your permission** opens

16. Make sure all the boxes are checked:


A

Inferno is requesting your permissions

Uncheck the permission you do not want to grant.

Personal information

☒ Your user Identifier

☒ Permission to retrieve information about the current logged-in user

Application Access to Grant All Permissions ☒

☒ Offline Access ⓘ

☒ launch

☒ patient/Patient.read ⓘ

Remember my decision

☒ Remember this decision until I revoke it

☐ Prompt me again next time

Decline
Accept

17. Click **Accept**
18. When you return to Inferno, a progress bar shows in the bottom right corner
19. When this progress bar finishes and closes, all tests should be green:



[HHS Privacy Policy](#)
[HHS Vulnerability Disclosure Policy](#)

Inferno on HealthIT.gov is not for use with sensitive data or Protected Health Information (PHI). Data periodically removed.

ONC Certification (g)(10) Standardized API V.X.X.X

US Core 3.1.1 / USCDI v1, SMART App Launch 1.0.0, Bulk Data 1.0.1

[NEW SESSION](#)

Preset
None

(g)(10) Standardized API

- ✓ 1 Standalone Patient App
- ✓ 2 Limited Access App
- ✓ 3 EHR Practitioner App
- ✓ 4 Single Patient API
- ✓ 7 Multi-Patient API
- ▼ 9 Additional Tests
 - ✓ 9.1 SMART Public Client Launch
 - ✓ 9.3 Token Revocation
 - ✓ 9.4 SMART Invalid AUD Launch
 - ✓ 9.5 SMART Invalid Token Request
 - ✓ 9.8 EHR Launch with Patient Scopes
 - 9.10 Visual Inspection

Report

Configuration Messages

✓ 9.8 EHR Launch with Patient Scopes

[RUN TESTS](#)

Background

Systems are required to support the permission-patient capability as part of the [Clinician Access for EHR Launch Capability Set](#).

Additionally, if an application launched from an EHR requests and is granted a clinical scope restricted to a single patient, the EHR SHALL establish a patient in context.

Register Inferno as an EHR-launched application using patient-level scopes and the following URIs:

- Launch URI: <https://inferno.healthit.gov/suites/custom/smart/launch>
- Redirect URI: <https://inferno.healthit.gov/suites/custom/smart/redirect>

Test Methodology

Inferno will attempt an EHR Launch with a clinical scope restricted to a single patient and verify that a patient-level scope is granted and a patient id is received.

For more information on the EHR Launch with Patient Scopes

- [Apps that launch from the EHR](#)

- ✓ 9.8.01 EHR server redirects client browser to Inferno app launch URI
- ✓ 9.8.02 EHR server sends launch parameter
- ✓ 9.8.03 OAuth 2.0 authorize endpoint secured by transport layer security
- ✓ 9.8.04 OAuth server redirects client browser to app redirect URI
- ✓ 9.8.05 OAuth server sends code parameter
- ✓ 9.8.06 OAuth 2.0 token endpoint secured by transport layer security
- ✓ 9.8.07 OAuth token exchange request succeeds when supplied correct information
- ✓ 9.8.08 Token exchange response body contains required information encoded in JSON
- ✓ 9.8.09 Response includes correct HTTP Cache-Control and Pragma headers
- ✓ 9.8.10 OAuth token exchange response body contains patient context and patient resource can be retrieved
- ✓ 9.8.11 Patient-level scopes were granted

INFERNO BUILT WITH
V.0.4.38 | API

[Report Issue](#) | [Open Source](#) | [Download](#)

10.6 Visual Inspection and Attestation

This does not contain automated tests and a number of these tests are not applicable to Firely Server. You can still 'run' these tests yourself based on the results you have so far.

11 Wrapping things up

Congratulations, you ran all the tests! Thanks for taking the time to check that Firely Server is compliant with ONC regulations relevant to FHIR servers. So, what's next?

You could [download](#) a trial version of Firely Server and continue testing it with your own set of tools, such as Postman. We have two blog posts to help you get started with that:

- [Installing your first FHIR server with Firely Server](#)
- [Your first interactions with your FHIR server](#)

Or maybe you've hit a blocker with your own FHIR server, and you're wondering how we created the infrastructure to pass all the Inferno tests? It took Firely Server, an authorization server, some extra configuration, and a lot of hours to make it all work seamlessly together, and we'd love to share that story with you. Let us know by sending an email to server@fire.ly.

12 About Firely

Firely provides all the software, training, and expertise needed to bring FHIR to life.

We're the authors of one of the two main open source FHIR libraries: Firely .NET SDK. Our software powers FHIR APIs and systems around the world, and many governments, hospitals, payers, and MedTech companies rely on our software for their FHIR capabilities.

Firely also provides FHIR training courses and workshops, and is the organizer of FHIR DevDays, the world's leading FHIR event. In 2019, we became a founding member of the FHIR Business Alliance.

You can find out more about Firely solutions, services, and education on our website (<https://fire.ly>).

13 Contact us

Your feedback is very welcome. Please send us your comments, compliments, and questions – and if you're interested in the commercial side of things, we'd be more than happy to schedule a meeting.

Just send us an email at info@fire.ly and we'll get back to you asap.



Made with TLC in Amsterdam



Firely Amsterdam

Westerdok 442
1013 BH Amsterdam
The Netherlands
+31 20 2461 241
info@fire.ly