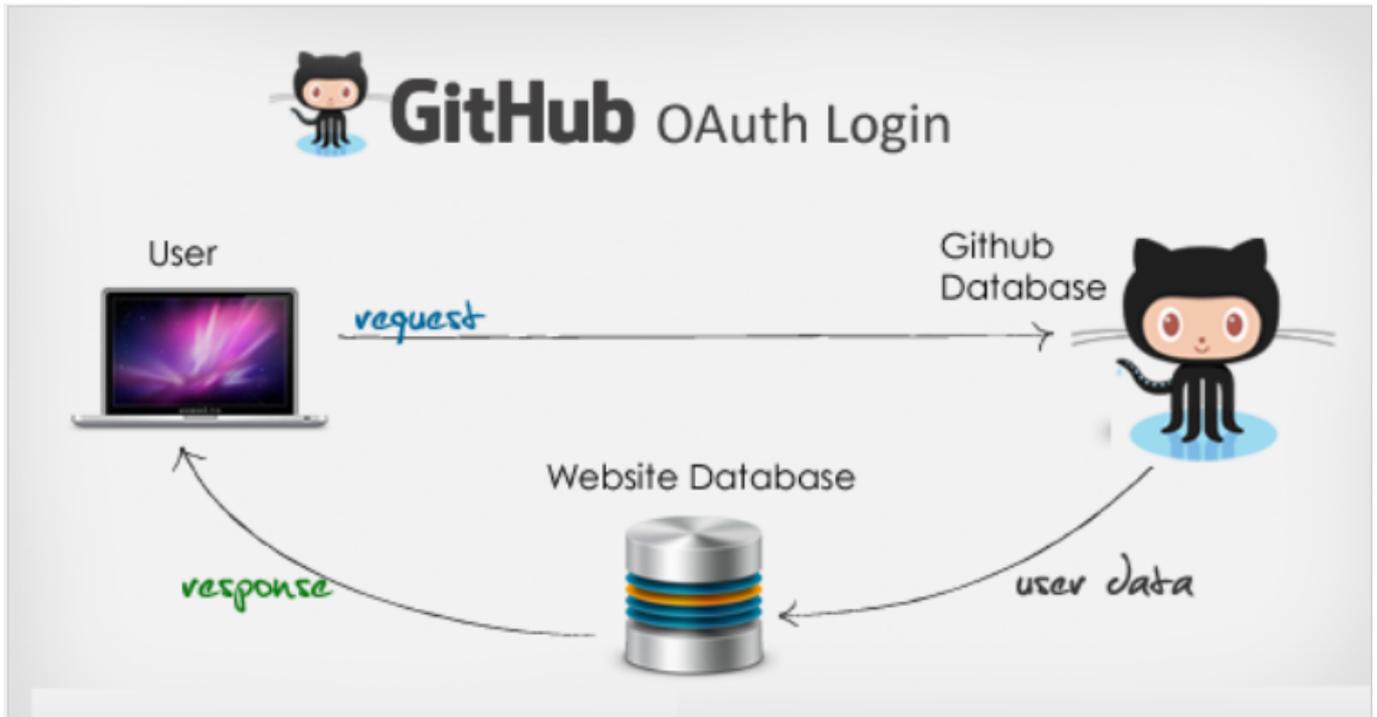


Artículo

[Muhammad Waseem](#) · 9 dic, 2021 Lectura de 3 min

[Open Exchange](#)

Autenticación OAuth2 con cuenta GitHub desde la aplicación web IRIS



En este artículo, mostraré los conceptos básicos de la autenticación OAuth2 con la cuenta de GitHub con la ayuda de la demostración online

<https://dappsecurity.demo.community.intersystems.com/csp/user/index.csp> usando SuperUser | SYS

Recomendaciones:

- Descripción general de OAuth2 : <https://learning.intersystems.com/course/view.php?id=252>
- Documentación: [Using an InterSystems IRIS Web Application as an OAuth2 Client](#)
- Artículos: [Implementación de la estructura Open Authorization \(OAuth 2.0\) en InterSystems IRIS - Parte 1](#)

A continuación, necesitamos 3 pasos:

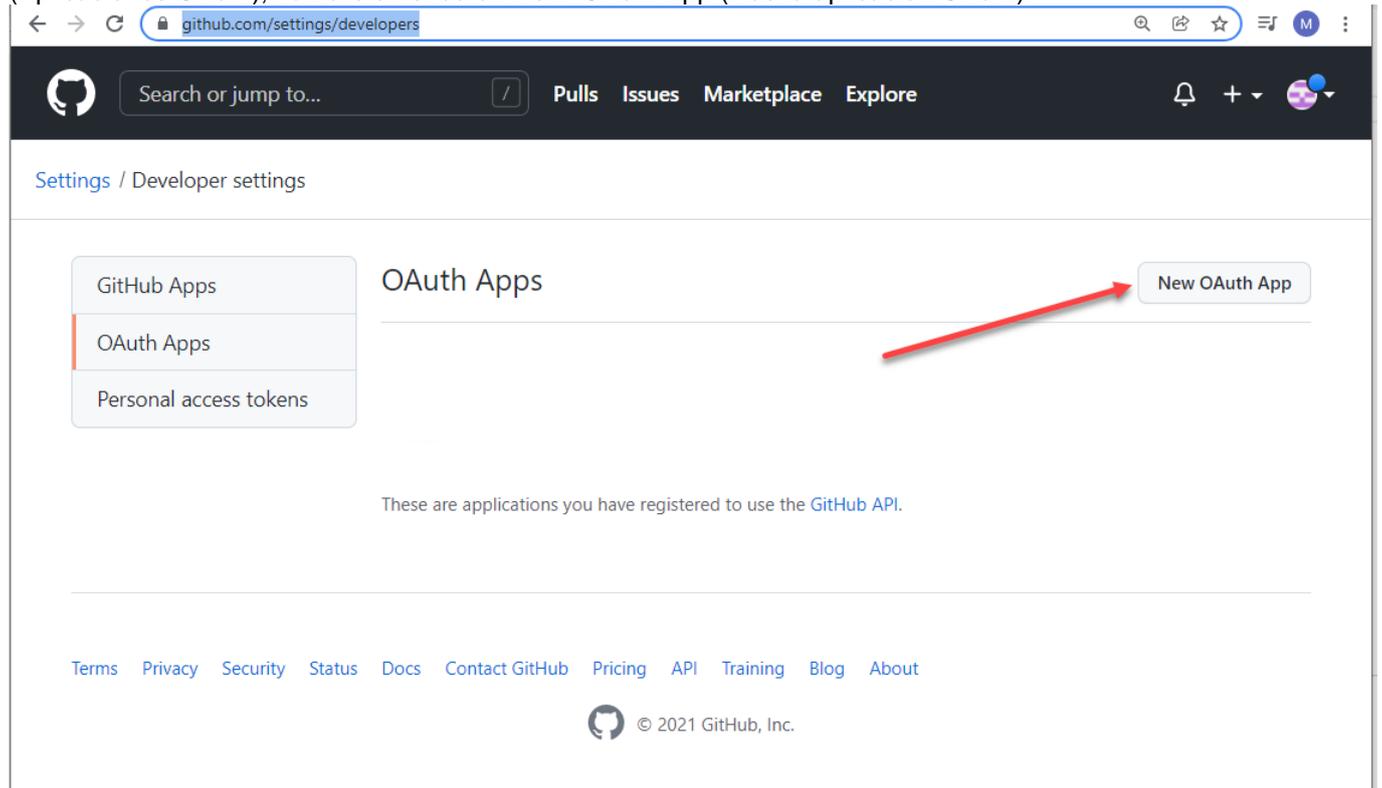
- Paso 1: Registrar la aplicación con el servidor de autenticación de GitHub
- Paso 2: Configurar el cliente OAuth 2.0 desde el portal de administración de InterSystems
- Paso 3: Llamar a la API para iniciar sesión con una cuenta de GitHub

Así que... ¡empecemos!

Paso 1: Registrar la aplicación con el servidor de autenticación de GitHub

Para registrar la aplicación con el servidor de autenticación de GitHub, necesitamos una cuenta de GitHub. Inicia sesión en la cuenta de GitHub, ve a <https://github.com/settings/developers> y, en la pestaña OAuth Apps

(Aplicaciones OAuth), haz clic en el botón New OAuth App (Nueva aplicación OAuth).



Escribe el nombre de la aplicación, la URL de la página de inicio, la descripción y la URL callback para la autorización.

Ten en cuenta que la URL callback para la autorización debe hacer referencia a la clase OAuth2.Response.cls (`{{dominio}} /csp/sys/oauth2/OAuth2.Response.cls`)

Haz clic en Register Application (Registrar aplicación):

← → ↻ github.com/settings/applications/new 🔍 📄 ☆ 🗨️ M ⋮

 Search or jump to... [/](#) [Pulls](#) [Issues](#) [Marketplace](#) [Explore](#) 🔔 + ▾ 

Register a new OAuth application

Application name *

Something users will recognize and trust.

Homepage URL *

The full URL to your application homepage.

Application description

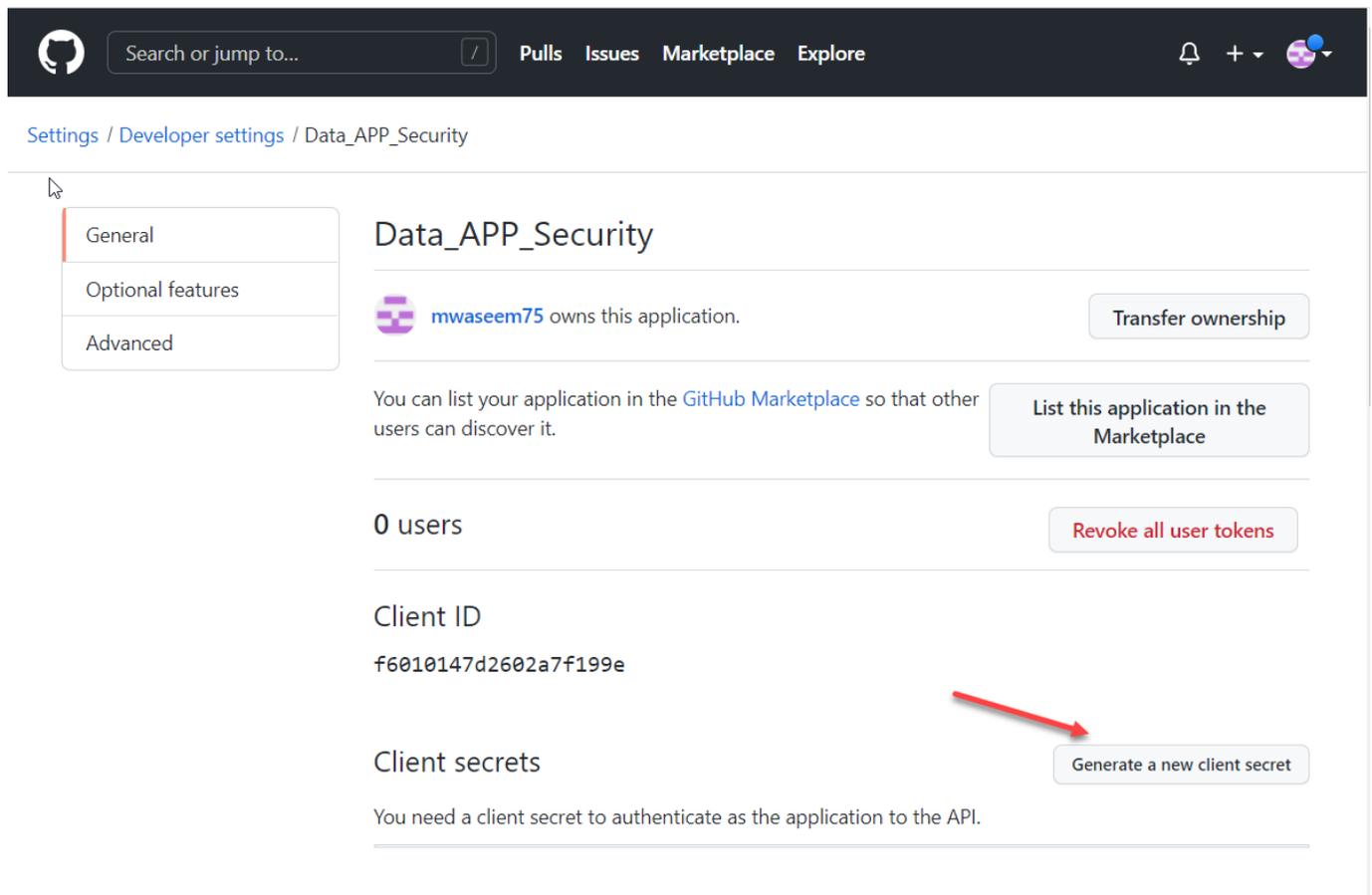
This is displayed to all users of your application.

Authorization callback URL *

Your application's callback URL. Read our [OAuth documentation](#) for more information.

[Register application](#) [Cancel](#)

Esto abrirá la página de detalles. Haz clic en Generate a new client secret (Generar un nuevo secreto de cliente) y guarda el ID de Cliente y la Clave Secreta que usaremos al configurar el cliente IRIS OAuth2.



The screenshot shows the GitHub Developer settings page for an application named "Data_APP_Security". The page is divided into three tabs: "General", "Optional features", and "Advanced". The "General" tab is selected. The application is owned by "mwaseem75". There are three main sections: "List this application in the Marketplace" (with a button), "0 users" (with a "Revoke all user tokens" button), and "Client secrets" (with a "Generate a new client secret" button). The Client ID is "f6010147d2602a7f199e". A red arrow points to the "Generate a new client secret" button.

Settings / Developer settings / Data_APP_Security

General
Optional features
Advanced

Data_APP_Security

 mwaseem75 owns this application. [Transfer ownership](#)

You can list your application in the [GitHub Marketplace](#) so that other users can discover it. [List this application in the Marketplace](#)

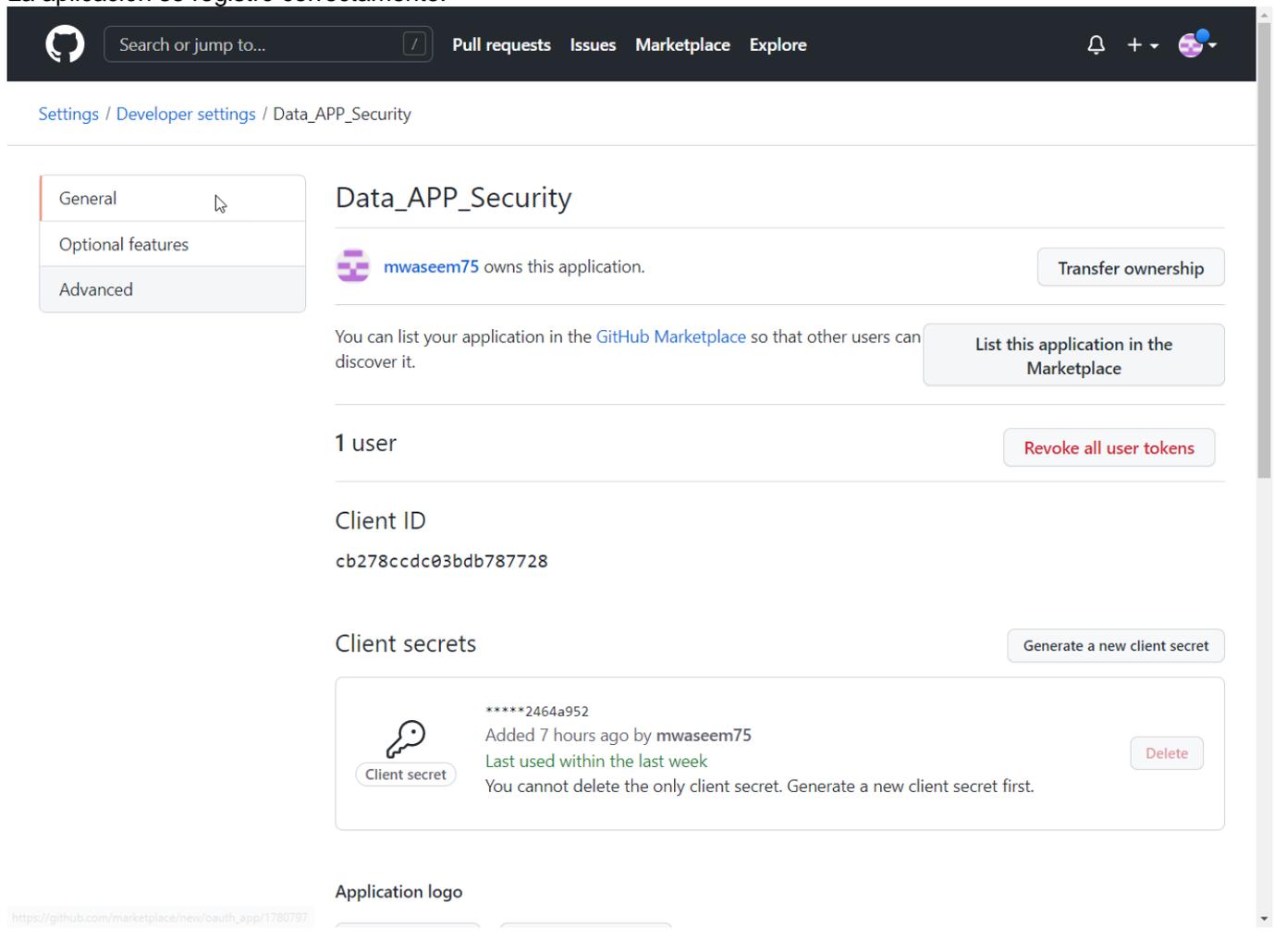
0 users [Revoke all user tokens](#)

Client ID
f6010147d2602a7f199e

Client secrets [Generate a new client secret](#)

You need a client secret to authenticate as the application to the API.

La aplicación se registró correctamente.



The screenshot shows the GitHub Developer settings page for the same application "Data_APP_Security" after registration. The "General" tab is selected. The application is still owned by "mwaseem75". The "List this application in the Marketplace" button is now disabled. The "0 users" section now shows "1 user" and the "Revoke all user tokens" button is now active. The Client ID is "cb278ccdc03bdb787728". The "Client secrets" section now shows a list of secrets. The first secret is "*****2464a952", added 7 hours ago by "mwaseem75", and last used within the last week. A "Delete" button is visible next to the secret. A message below the secret states: "You cannot delete the only client secret. Generate a new client secret first." The "Generate a new client secret" button is still present.

Settings / Developer settings / Data_APP_Security

General
Optional features
Advanced

Data_APP_Security

 mwaseem75 owns this application. [Transfer ownership](#)

You can list your application in the [GitHub Marketplace](#) so that other users can discover it. [List this application in the Marketplace](#)

1 user [Revoke all user tokens](#)

Client ID
cb278ccdc03bdb787728

Client secrets [Generate a new client secret](#)

 *****2464a952
Added 7 hours ago by mwaseem75
Last used within the last week
[Delete](#)

[Client secret](#)

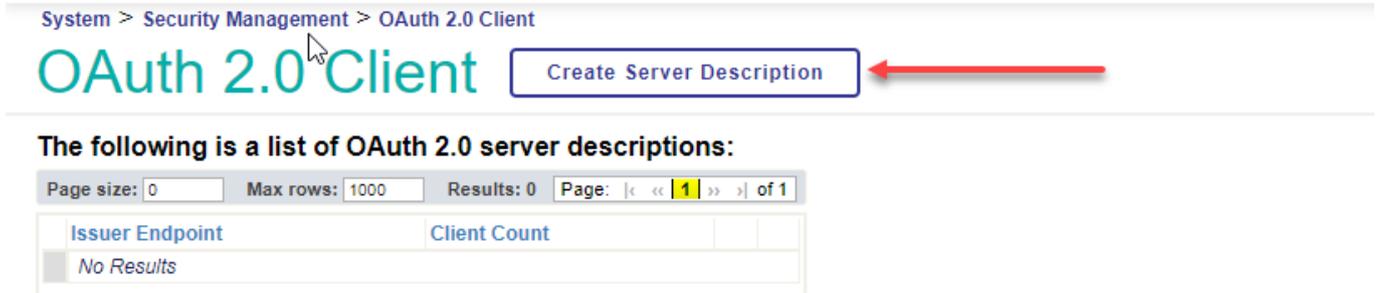
You cannot delete the only client secret. Generate a new client secret first.

Application logo

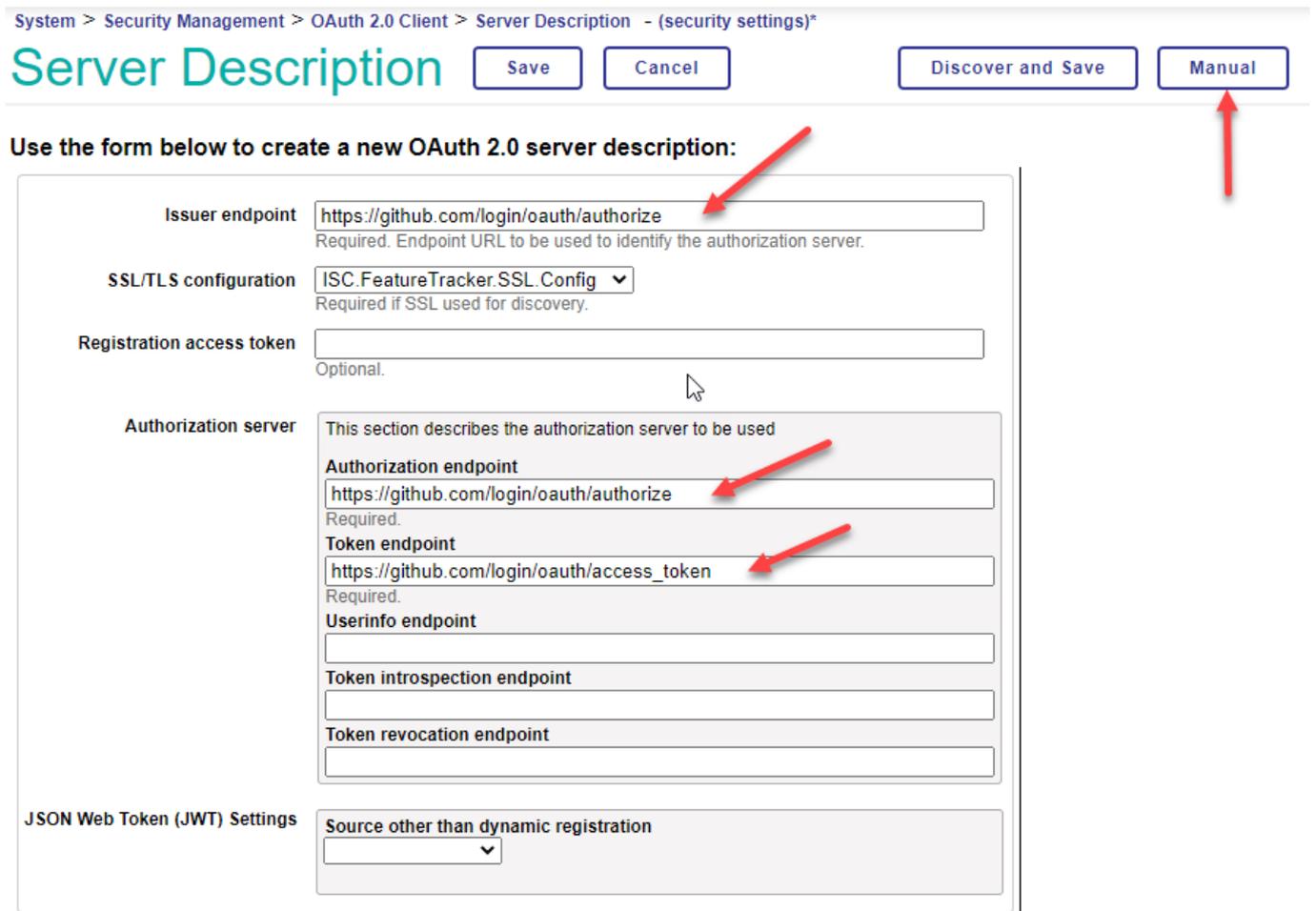
https://github.com/marketplace/new/oauth_app/1780797

Paso 2: Configurar el cliente OAuth 2.0 desde el portal de administración de InterSystems

Ve a [System](#) > [Security Management](#) > [OAuth 2.0 Client](#) y haz clic en el botón Create Server Description (Crear descripción del servidor):



Haz clic en el botón Manual en la parte superior, introduce los detalles en el formulario y guarda la descripción del servidor:



Vuelve a [System](#) > [Security Management](#) > [OAuth 2.0 Client](#) y haz clic en Client Configurations (Configuraciones del cliente):

System > Security Management > OAuth 2.0 Client

OAuth 2.0 Client

Create Server Description

The following is a list of OAuth 2.0 server descriptions:

Page size: 0 Max rows: 1000 Results: 1 Page: |< << 1 >> >| of 1

Issuer Endpoint	Client Count	
https://github.com/login/oauth/authorize	0	Client Configurations Delete

En la pestaña General, completa los detalles. Ten en cuenta que la URL de redireccionamiento del cliente es la misma que pusimos en la URL callback para la autorización al registrar nuestra aplicación en GitHub.

General Client Information JWT Settings Client Credentials

Application name Data_APP_Security
Required. Local name of the client application.

Client name Data_APP_Security
Global name to be used for dynamic registration.

Description Data App Security Application for Authorization, Authentication and Auditing

Enabled

Client Type Confidential Public Resource server

SSL/TLS configuration ISC.FeatureTracker.SSL.Config
Required.

Client redirect URL
The client URL to be specified to the authorization server to receive responses.
https://dappsecurity.demo.community.intersystems.com/csp/sys/oauth2/OAuth2.Respons
e.cls
Use TLS/SSL
Host name dappsecurity.demo.communi
Required. **Port**
Optional. **Prefix**
Optional.

Required grant types (check at least one) Authorization code
 Implicit
 Resource owner password credentials
 Client credentials
 JWT authorization

Authentication type none basic form encoded body client secret JWT private key JWT

En la pestaña Client Credentials (Credenciales de cliente), introduce la ID de cliente y el secreto del cliente que generamos al registrar nuestra aplicación en GitHub y guarda la configuración del cliente.

General Client Information JWT Settings Client Credentials

This client's credentials

Client ID
cb278ccdc03bdb787728
Required.

Client ID Issued At

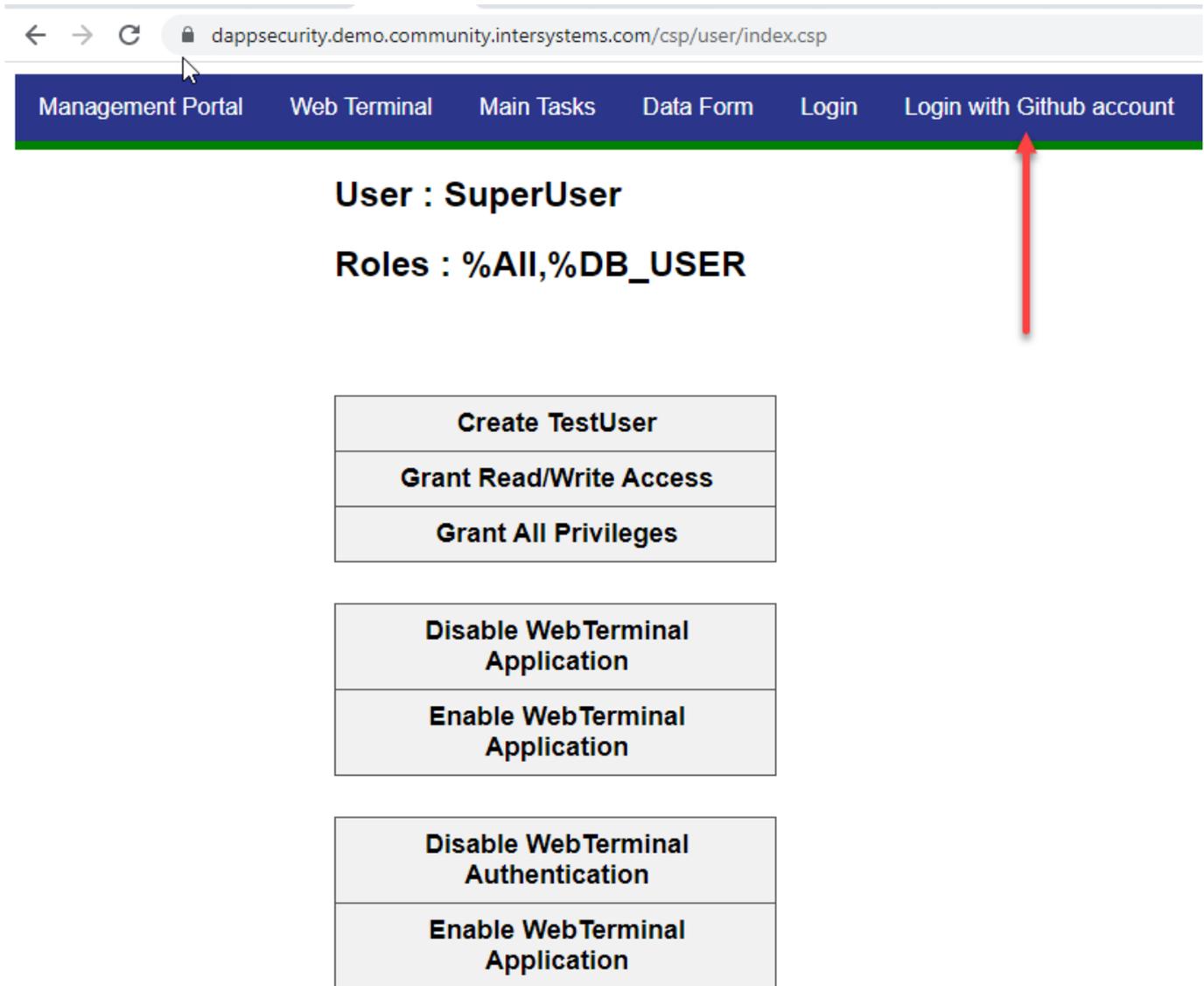
Client secret
22b8136d37516c39ad9b0c334b395cb32464a952
Required if client type is confidential.

Client Secret Expires At

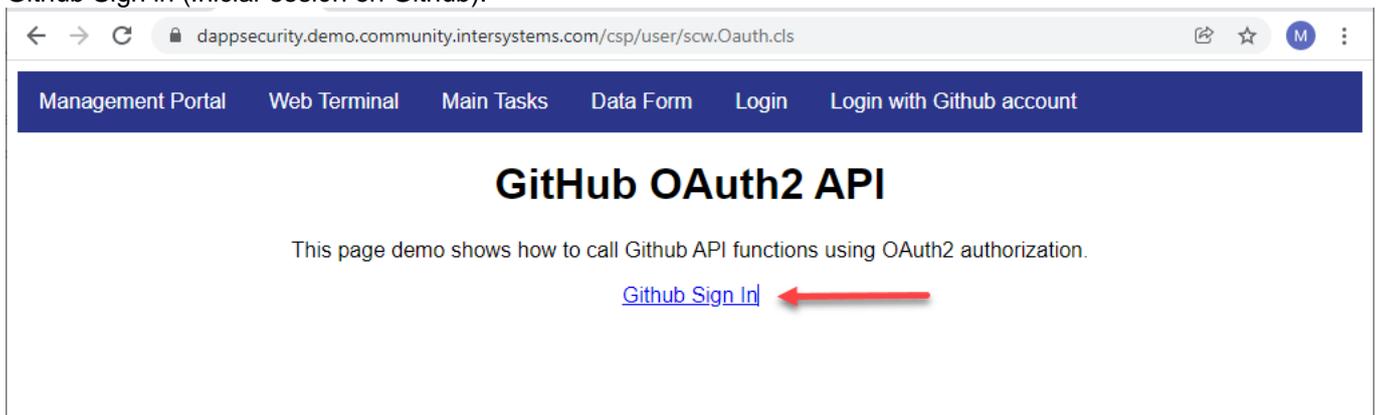
Registration Client Uri

Paso 3: Llamar a la API para iniciar sesión con una cuenta de GitHub

Ve a la demostración online: <https://dappsecurity.demo.community.intersystems.com/csp/user/index.csp> e inicia sesión con SuperUser | SYS
Haz clic en Login with Github account (Iniciar sesión con la cuenta de Github) en el menú superior:



Esto abrirá la página Oauth.cls. Asegúrate de cerrar sesión en GitHub antes de intentar iniciar sesión. Haz clic en Github Sign in (Iniciar sesión en Github):



El sistema navegará al servidor de autenticación de GitHub usando el método GetAuthorizationCodeEndpoint de la clase% SYS.OAuth2.Authorization
<https://docs.intersystems.com/latest/csp/documatic/%25CSP.Documatic.cls?&LIBRARY=%25SYS&PRIVATE=1&CLASNAME=%25SYS.OAuth2.Authorization>

← → ↻ github.com/login?client_id=cb278ccdc03bdb787728&return_to=%2Flogin%2Foauth%2Fauthorize%3Fappr... 🔑 🔍 📄 ☆ M ⋮





Sign in to GitHub
to continue to Data_APP_Security

Username or email address

Password [Forgot password?](#)

Sign in

New to GitHub? [Create an account.](#)

Cuando hayas iniciado sesión correctamente, el sistema lo redireccionará a la página OauthRe.cls

← → ↻ dappsecurity.demo.community.intersystems.com/csp/user/scw.OauthRe.cls 🗑️

Management Portal Web Terminal Main Tasks Data Form Login **Login with Github account**

Successfully Authorize in IRIS via Github OAuth2

El código está disponible en este repositorio <https://github.com/mwaseem75/DataAPPSecurity>

[#Autenticación](#) [#GitHub](#) [#OAuth2](#) [#InterSystems IRIS](#)
[Ir a la aplicación en InterSystems Open Exchange](#)

URL de
fuente: <https://es.community.intersystems.com/post/autenticaci%C3%B3n-oauth2-con-cuenta-github-desde-la-aplicaci%C3%B3n-web-iris>