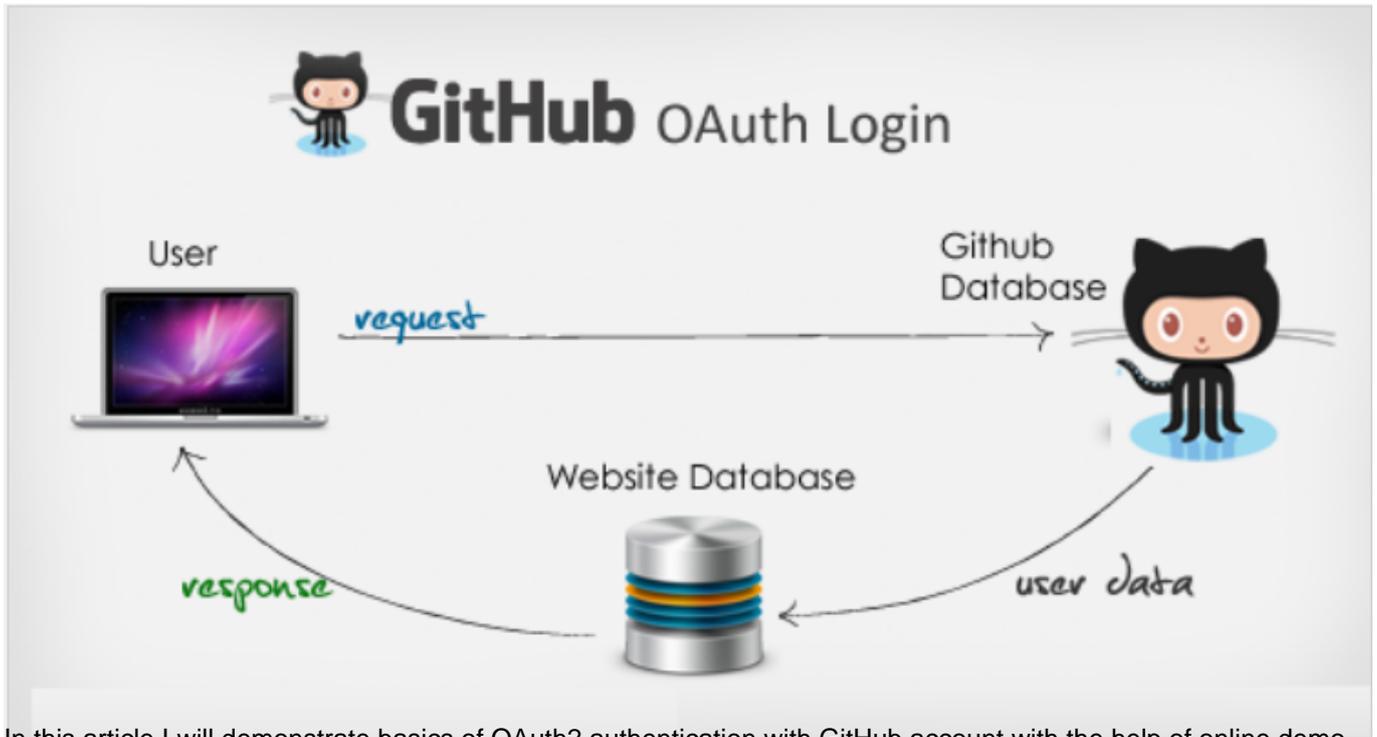


Article

[Muhammad Waseem](#) · Dec 4, 2021 3m read

[Open Exchange](#)

OAuth2 Authentication with GitHub account from IRIS Web Application



In this article I will demonstrate basics of OAuth2 authentication with GitHub account with the help of online demo <https://dappsecurity.demo.community.intersystems.com/csp/user/index.csp> by using SuperUser | SYS

Recommendations:

- OAuth2 Overview : <https://learning.intersystems.com/course/view.php?id=252>
- Documentations : [Using an InterSystems IRIS Web Application as an OAuth2 Client](#)
- Articles : <https://community.intersystems.com/post/intersystems-iris-open-authorized...>

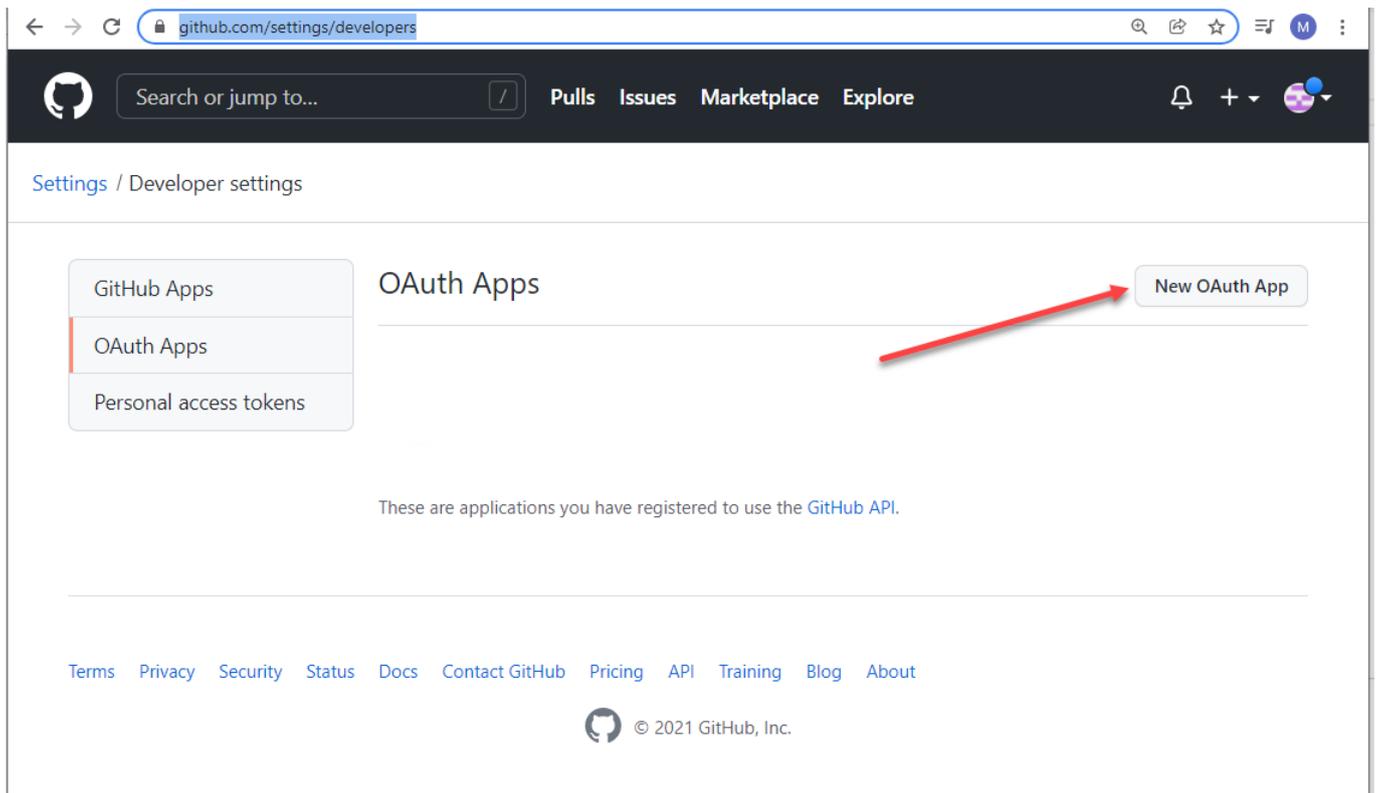
We need below 3 steps to achieve the desire :

- Step 1 : Register Application with GitHub Authentication Server
- Step 2 : Configure OAuth 2.0 Client from InterSystems Management portal
- Step 3 : Call API to login with GitHub account

So Let's start

Step 1 : Register Application with GitHub Authentication Server

In order to register application with GitHub authentication server we need GitHub account. Log in to GitHub account and navigate to <https://github.com/settings/developers> and under OAuth Apps tab click New OAuth App button



Enter Application name, Homepage URL, Description and Authorization call back URL
Please note that Authorization call back URL must refer to OAuth2.Response.cls class
({domain}/csp/sys/oauth2/OAuth2.Response.cls)
Click Register Application

← → ↻ 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](#)

This will open detail page. Click Generate a new client secret and save Client ID and Secret Key which we will use while configuring IRIS OAuth2 client

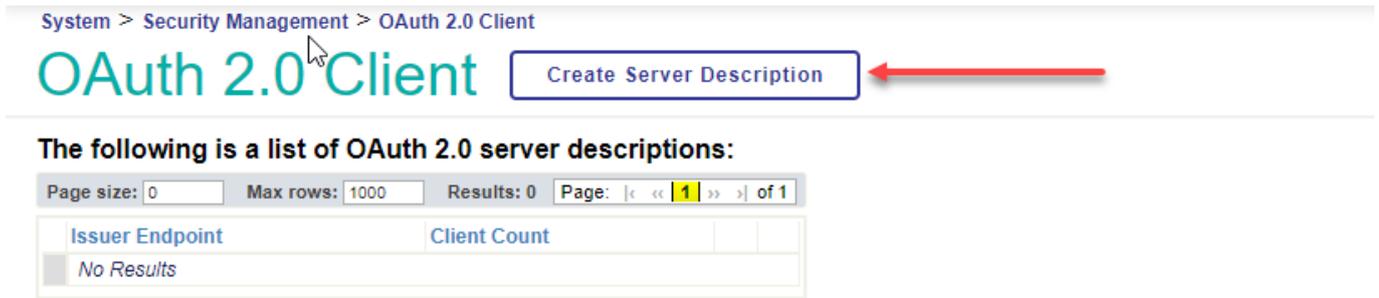
The screenshot shows the GitHub Developer settings page for an application named "Data_APP_Security". The page is divided into three tabs: "General", "Optional features", and "Advanced". The "General" tab is selected. The application is owned by "mwaseem75". There are three main sections: 1. Ownership: "mwaseem75 owns this application." with a "Transfer ownership" button. 2. Marketplace: "You can list your application in the GitHub Marketplace so that other users can discover it." with a "List this application in the Marketplace" button. 3. Users: "0 users" with a "Revoke all user tokens" button. Below these is the "Client ID" field with the value "f6010147d2602a7f199e". The "Client secrets" section shows "You need a client secret to authenticate as the application to the API." and a "Generate a new client secret" button. A red arrow points to this button.

Application is registered successfully

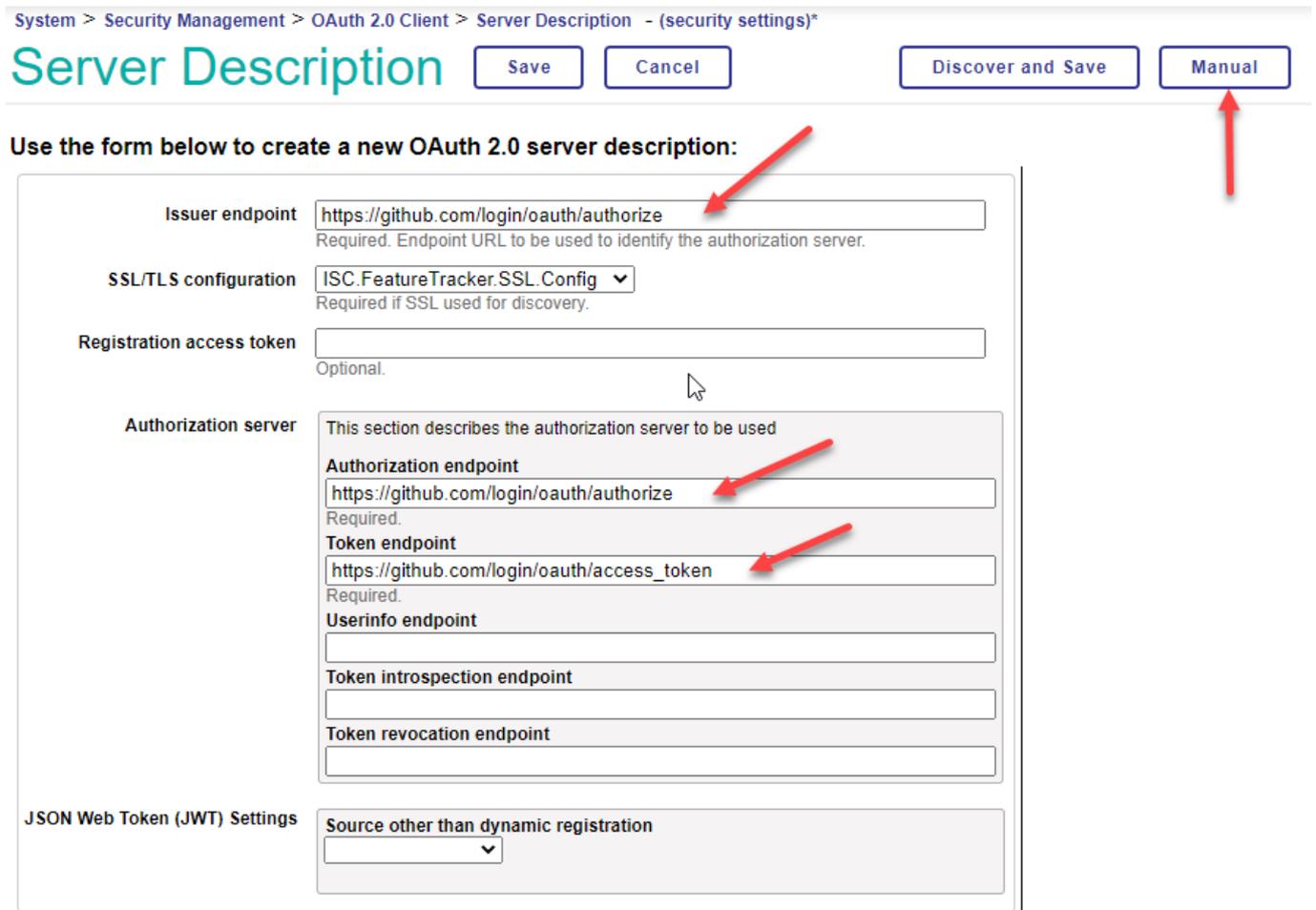
This screenshot shows the same GitHub Developer settings page, but now with one user and a client secret. The "Optional features" tab is selected. The "Users" section shows "1 user" and a "Revoke all user tokens" button. The "Client ID" is "cb278ccdc03bdb787728". The "Client secrets" section shows a "Generate a new client secret" button and a list of secrets. One secret is shown: "*****2464a952", added 7 hours ago by "mwaseem75", last used within the last week. A "Delete" button is next to it, with a message: "You cannot delete the only client secret. Generate a new client secret first." At the bottom, there is an "Application logo" section with a URL: https://github.com/marketplace/new/oauth_app/1780797.

Step 2 : Configure OAuth 2.0 Client from InterSystems Management portal

Navigate to [System](#) > [Security Management](#) > [OAuth 2.0 Client](#) and click Create Server Description button



Click Manual button from top and enter below details and Save the server description



Navigate back to [System](#) > [Security Management](#) > [OAuth 2.0 Client](#) and click Client Configurations

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

In General tab fill the details. Please note that Client redirect URL is same what we entered in Authorization callback URL while registering our application at 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://dapsecurity.demo.community.intersystems.com/csp/sys/oauth2/OAuth2.Response.cls
Use TLS/SSL
Host name dapsecurity.demo.communi **Port** **Prefix**
Required. Optional. 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

Under Client Credentials enter Client ID and Client Secret which we generated while registering our application at GitHub and save the client configuration

General Client Information JWT Settings Client Credentials

This client's credentials

Client ID
cb278ccdc03bdb787728
Required.

Client ID Issued At

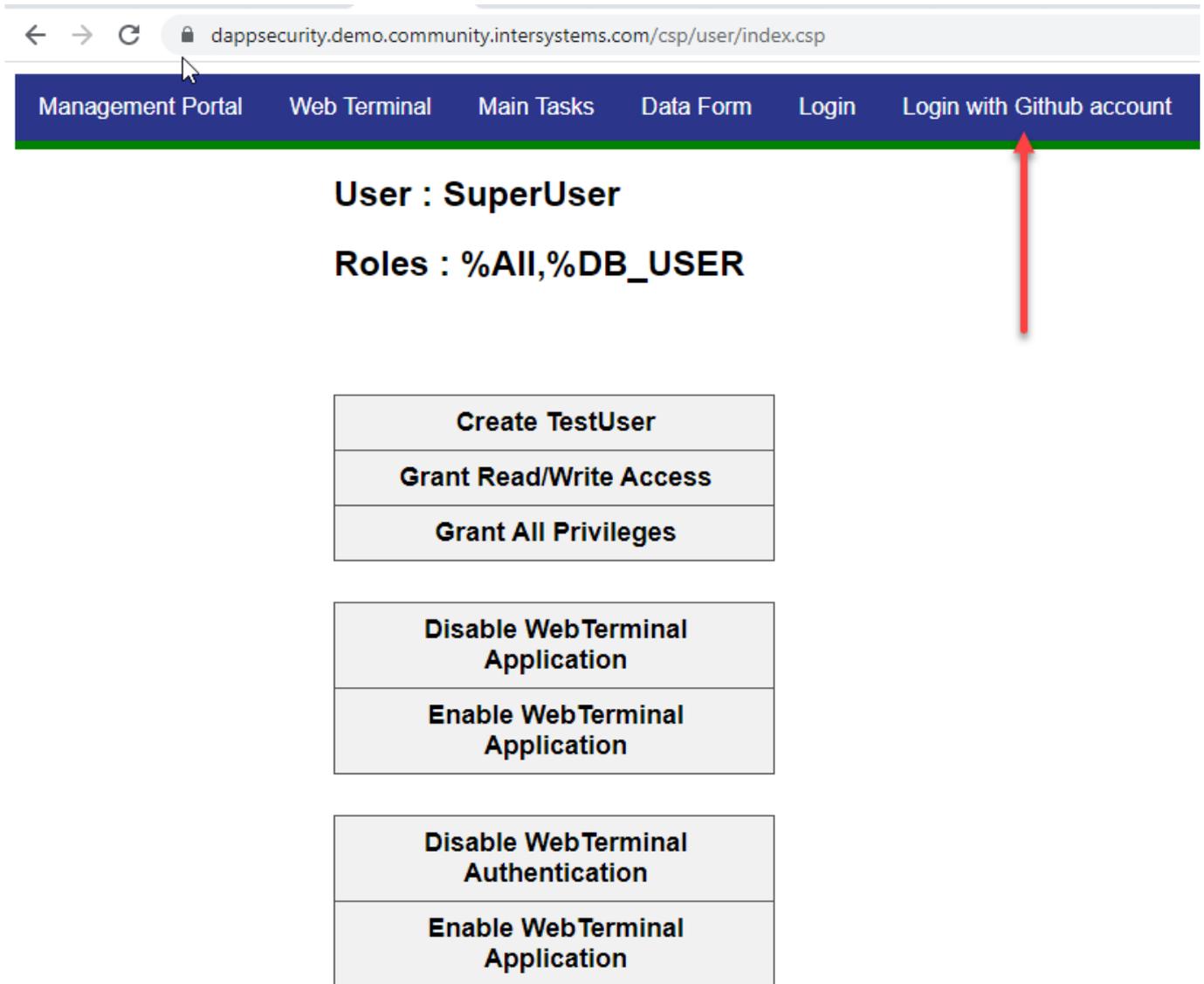
Client secret
22b8136d37516c39ad9b0c334b395cb32464a952
Required if client type is confidential.

Client Secret Expires At

Registration Client Uri

Step 3 : Call API to login with GitHub account

Navigate to online demo at <https://dappsecurity.demo.community.intersystems.com/csp/user/index.csp> and login with SuperUser | SYS
Press Login with Github account from top menu



This will open Oauth.cls page. Make sure to logout from GitHub before trying sign in. Click Github Sign In

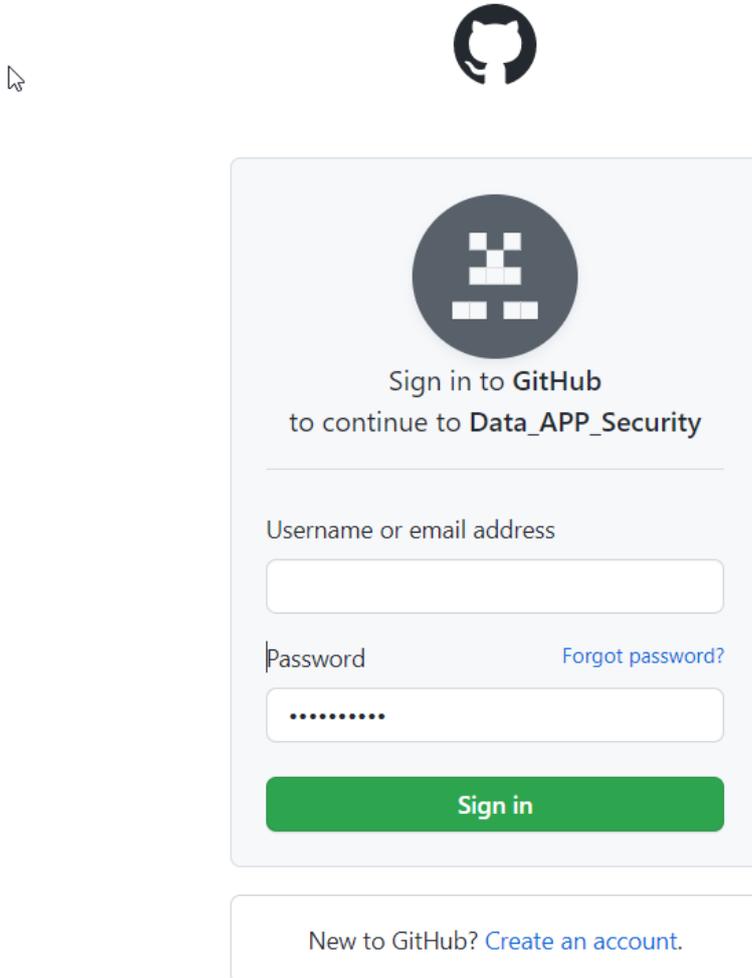


System will navigate to GitHub authentication server by using GetAuthorizationCodeEndpoint method

of %SYS.OAuth2.Authorization class

<https://docs.intersystems.com/latest/csp/documatic/%25CSP.Documatic.cls?...>

github.com/login?client_id=cb278ccdc03bdb787728&return_to=%2Flogin%2Foauth%2Fauthorize%3Fappr...



The image shows a GitHub login page. At the top center is the GitHub logo. Below it is a light blue box containing a circular profile picture placeholder with a grid pattern. The text reads "Sign in to GitHub to continue to Data_APP_Security". There are two input fields: "Username or email address" and "Password". A "Forgot password?" link is next to the password field. A green "Sign in" button is at the bottom of the box. Below the box is a link: "New to GitHub? [Create an account.](#)"

Upon successful login system will redirect to OauthRe.cls page

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

Code is available at this repository <https://github.com/mwaseem75/DataAPPSecurity>

Thanks

[#Authentication](#) [#GitHub](#) [#OAuth2](#) [#InterSystems](#) [IRIS](#)

[Check the related application on InterSystems Open Exchange](#)

Source URL: <https://community.intersystems.com/post/oauth2-authentication-github-account-iris-web-application>