



OIDC Plugin

Created: Jan 2, 2026
Updated:

Introduction.....	3
Why OIDC?.....	3
Configure with Google Auth Platform.....	4
Test Connection.....	5
Configure with Okta.....	6
Test Connection.....	9
After Issuer Configuration:.....	10
Configure Only Allow OIDC Login.....	10
Enable API Basic for Service Accounts.....	11
Configure Auto Provision User.....	12
Role source from OIDC claims.....	12

Introduction

OIDC is an **open, industry-standard identity protocol** built on OAuth 2.0. By adopting OIDC, the integration engine can seamlessly integrate with leading Identity Providers (IdPs) such as **Google Auth Platform and Okta**, avoiding vendor lock-in and ensuring long-term compatibility.

Why OIDC?

Centralized Identity Management

The OIDC plugin enables **centralized user authentication** through an external IdP rather than local user stores. This allows organizations to:

- Manage users, roles, and credentials in one place
- Enforce consistent authentication policies across systems
- Reduce administrative overhead within the integration engine

Enhanced Security Posture

- Eliminating local password storage within Mirth Connect / BridgeLink
- Supporting strong authentication mechanisms such as **Multi-Factor Authentication (MFA)**
- Using **short-lived tokens** and cryptographic signing (JWT) to **prevent credential leakage**

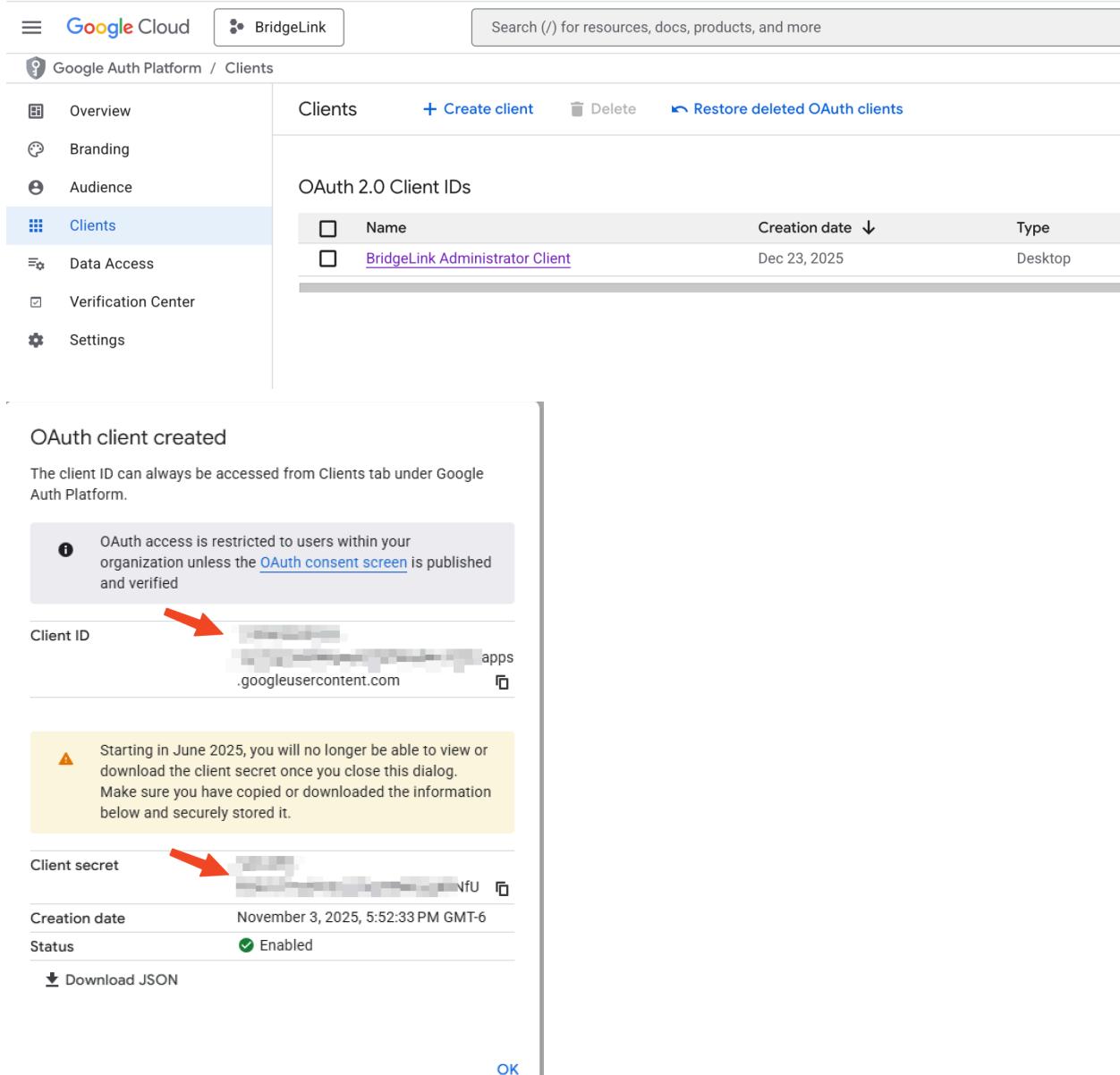
Improved User and Administrator Experience

- Users benefit from familiar enterprise login experiences
- Administrators gain simplified onboarding and offboarding

Configure with Google Auth Platform

Prerequisite

- Create an OAuth 2.0 Client ID and select the **Desktop** application type.
- Securely save the Client ID and Secret obtained from your Google Cloud Platform (GCP) account.



The screenshot shows the Google Cloud Platform interface for managing OAuth 2.0 Client IDs. The left sidebar is titled 'Google Auth Platform / Clients' and includes links for Overview, Branding, Audience, Clients (which is selected), Data Access, Verification Center, and Settings. The main content area is titled 'Clients' and shows a table of OAuth 2.0 Client IDs. The table has columns for 'Name', 'Creation date', and 'Type'. One row is listed: 'BridgeLink Administrator Client' (Creation date: Dec 23, 2025, Type: Desktop). A modal dialog box titled 'OAuth client created' is displayed, stating 'The client ID can always be accessed from Clients tab under Google Auth Platform.' It contains a note: 'OAuth access is restricted to users within your organization unless the [OAuth consent screen](#) is published and verified'. Below this, the 'Client ID' field is shown as a redacted string followed by '.googleusercontent.com'. A warning message below it says: 'Starting in June 2025, you will no longer be able to view or download the client secret once you close this dialog. Make sure you have copied or downloaded the information below and securely stored it.' The 'Client secret' field is also redacted. At the bottom of the modal, it shows 'Creation date: November 3, 2025, 5:52:33 PM GMT-6', 'Status: Enabled', and a 'Download JSON' button. An 'OK' button is at the bottom right of the modal.

Step 1: Configure OIDC Settings

- Open the **BridgeLink Administrator Console** and navigate to **Settings > OIDC**.
- Enable the OIDC feature.
- Enter the Issuer URL: <https://accounts.google.com>
- Enter the Client ID and Secret.
- Add at least 3 **Callback URLs**. You can enter idle ports on your local machine.

Settings

Server \ Administrator \ Tags \ Configuration Map \ Database Tasks \ Resources \ Certificate Manager \ Access Control \ Data Pruner \ Version History \ OIDC \

Enable OIDC

Enable: Yes No

OIDC Settings

Issuer: `https://accounts.google.com`

Client ID: `[REDACTED].apps.googleusercontent.com`

Client Secret: `[REDACTED]`

Redirect URIs:

`http://localhost:58244/callback`
`http://localhost:58245/callback`
`http://localhost:58246/callback`

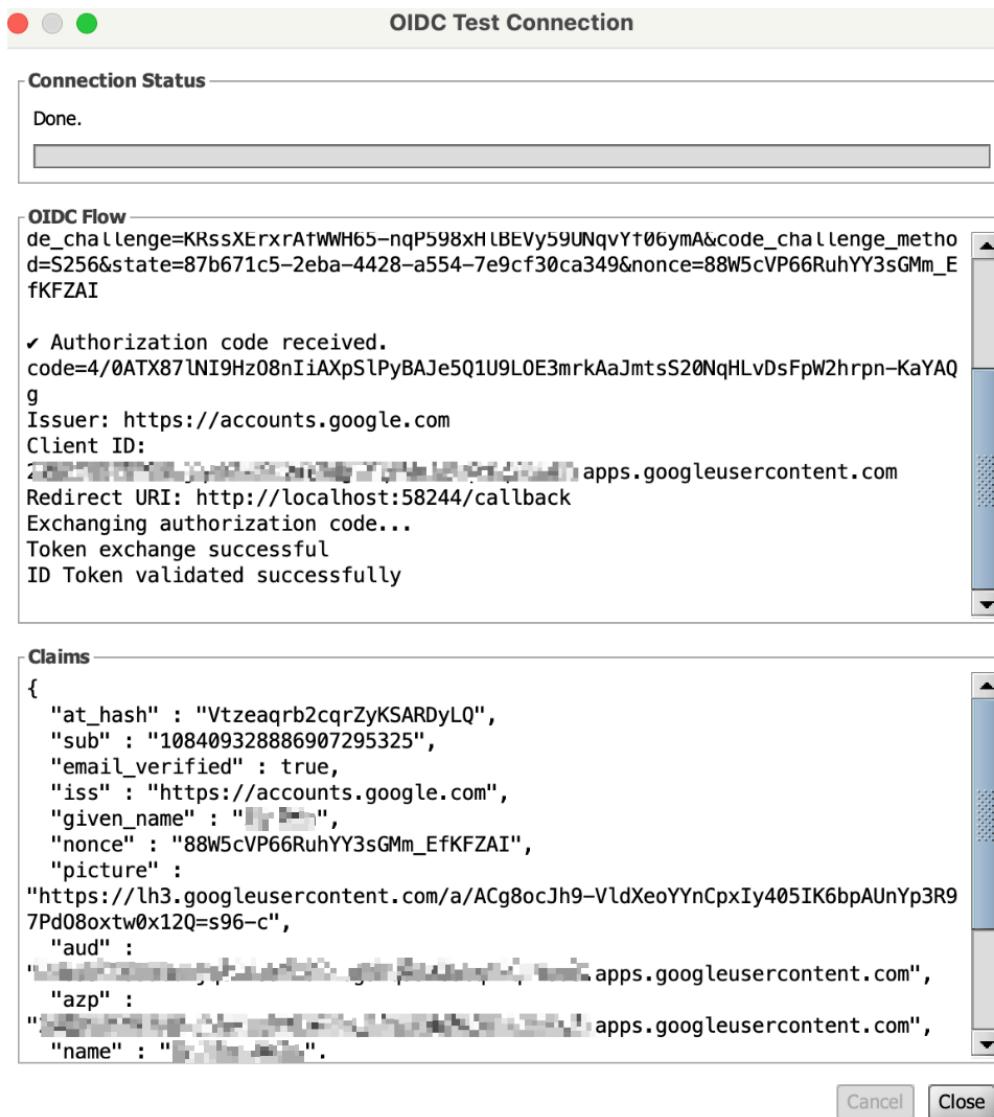
Test Connection

Click “Test Connection” to run a test with your OIDC server

Scopes: `openid profile email`

Test Connection

You can see the claims response from OIDC server in the prompt dialog window

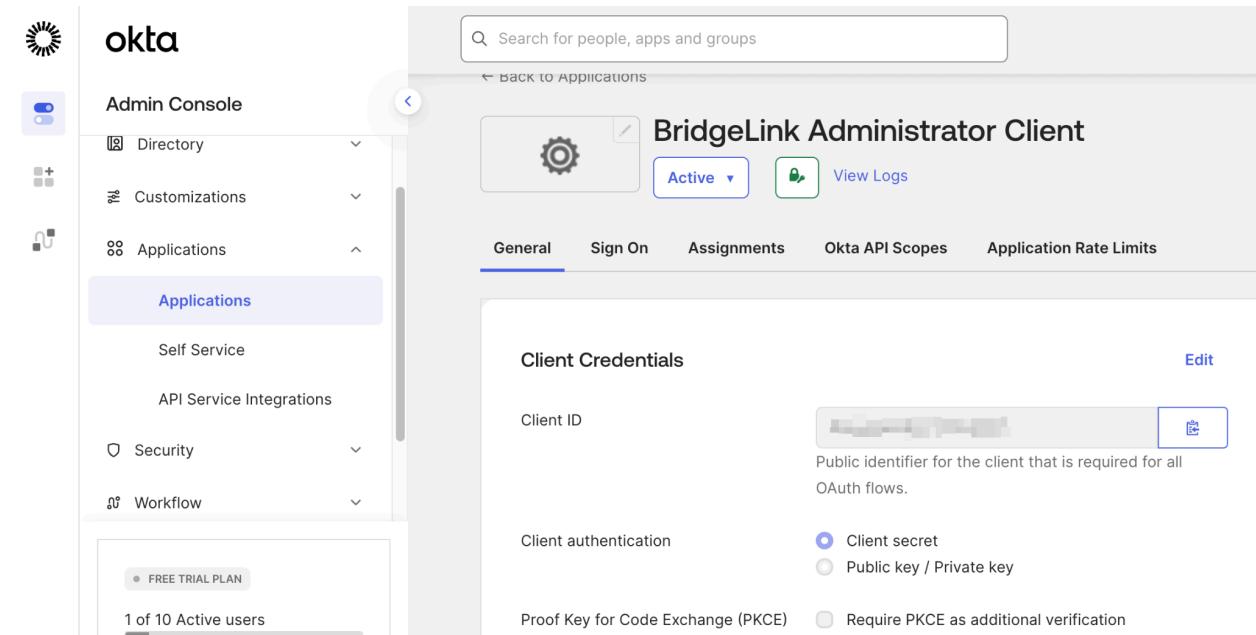


Configure with Okta

Prerequisite:

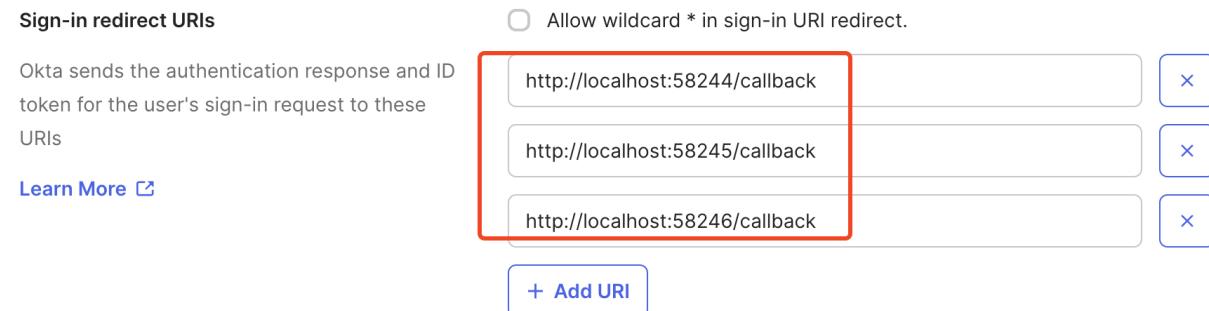
Please create an App Integration in your Okta account.

EX:



The screenshot shows the Okta Admin Console interface. On the left, the sidebar has 'Admin Console' selected, with 'Applications' highlighted. The main area shows the 'BridgeLink Administrator Client' application details. The 'General' tab is selected, showing the 'Client Credentials' section. It includes a 'Client ID' field (redacted), a 'Client authentication' section (radio button selected for 'Client secret'), and a 'Proof Key for Code Exchange (PKCE)' section (radio button selected for 'Require PKCE as additional verification'). A 'View Logs' button is also present.

Make sure you enter at least 3 callback urls, these will be used in the BridgeLink OIDC setting.



The screenshot shows the 'Sign-in redirect URIs' configuration in the Okta Admin Console. It includes a note that Okta sends the authentication response and ID token for the user's sign-in request to these URIs. There is a checkbox for 'Allow wildcard * in sign-in URI redirect'. Three callback URLs are listed: 'http://localhost:58244/callback', 'http://localhost:58245/callback', and 'http://localhost:58246/callback'. The first two URLs are highlighted with a red box. A 'Learn More' link is also present.

Navigate to Okta -> Directory -> Groups -> “Add group”, and add user to the group
EX:

okta

Admin Console

Groups

People

BridgeLink DevOps Role

Created: 12/24/2025 Last modified: 12/24/2025 View logs

People Applications Profile Directories Admin roles

People

Assign people

Person & username Status

Zi-Min Weng Active

zweng@innovarhealthcare.com

Navigate to Okta -> Security -> API -> Select the Authorizer server -> Claims -> Add Claim

okta

Admin Console

API

default

Active Default

Claims

Add Claim

Claim type	Name	Value	Scopes	Type	Included
All	sub	(appuser != null) ? appuser.userName :	Any	access	Always
ID	groups	groups: matches regex .*	Any	id	Always

In the Claim edit window, please enter “groups”, select the ID Token for token type, Groups for Value type. And save.

Edit Claim

Name: groups

Include in token type: ID Token (Always)

Value type: Groups

Filter: Only include groups that meet the following condition.
Matches regex: .*

Disable claim: Disable claim

Include in: Any scope
 The following scopes:

Save Cancel

Step 1: Configure OIDC Settings

- Open the **BridgeLink Administrator Console** and navigate to **Settings > OIDC**.
- Enable the OIDC feature.
- Enter the Issuer URL: `https://yourOktaDomain/oauth2/default`
- Enter the Client ID and Secret.
- Add the same **Callback URLs as on Okta**. You can enter idle ports on your local machine.

Enable OIDC: Yes No

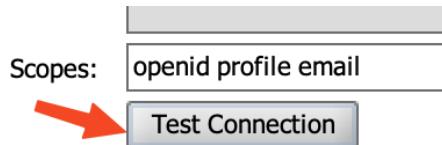
OIDC Settings

Issuer:	https://trial- REDACTED .okta.com/oauth2/default
Client ID:	REDACTED
Client Secret:	*****
Redirect URIs:	http://localhost:58244/callback http://localhost:58245/callback http://localhost:58246/callback

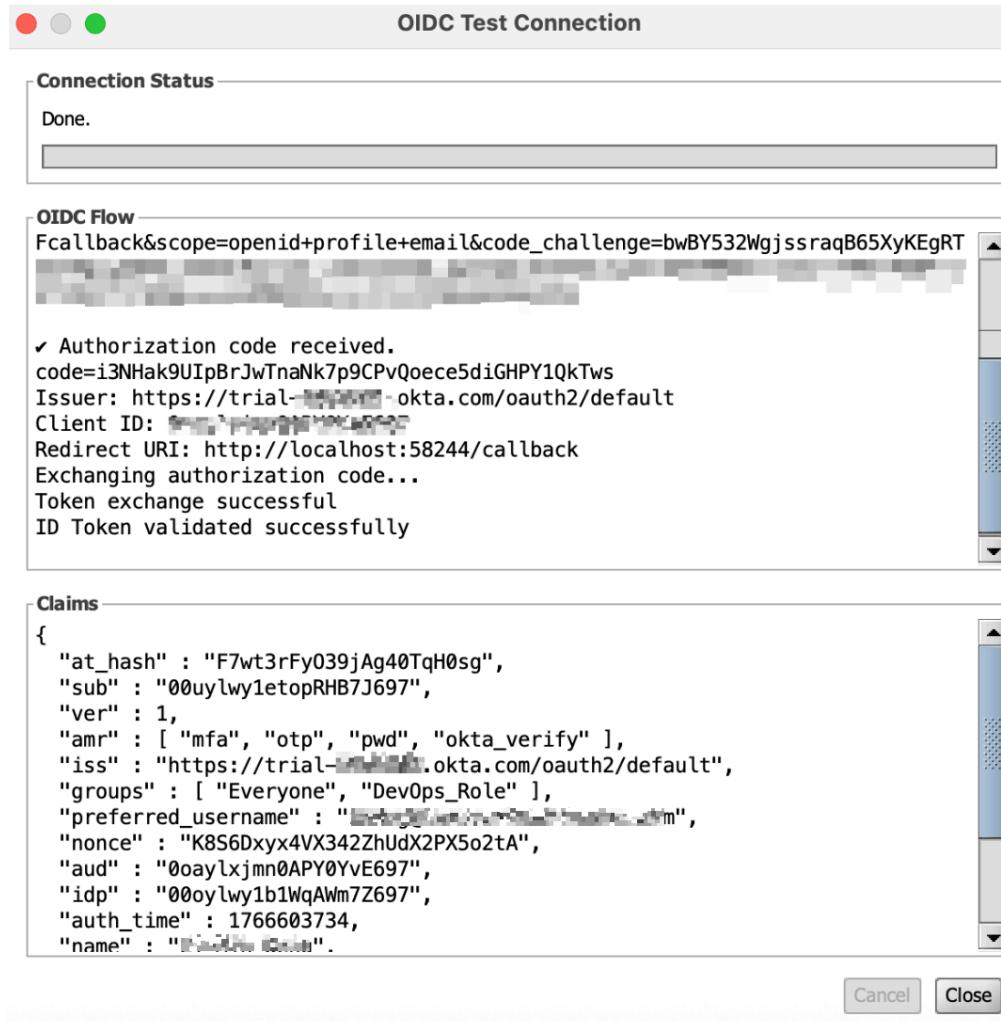
Save Cancel

Test Connection

Click “Test Connection” to run a test with your OIDC server



You can see the claims response from OIDC server in the prompt dialog window
EX:



After Issuer Configuration:

Configure Only Allow OIDC Login

If "Only allow OIDC login" is selected, all BridgeLink users **must** sign in through the OIDC server. An exception user can be added to bypass OIDC authentication for emergency fallback.

Access Control

Exclusive Mode: Only allow OIDC login

Emergency Fallback: Allow Emergency Admin fallback

Emergency Admin Username:

Service Accounts (API):

Service Account Users:

Enable API Basic for Service Accounts

When "Enable API Basic for Service Accounts" is enabled and a user is chosen, only the selected user can access and utilize the BridgeLink APIs (<https://<BridgeLink IP>:8443/api>).

Access Control

Exclusive Mode: Only allow OIDC login

Emergency Fallback: Allow Emergency Admin fallback

Emergency Admin Username:

Service Accounts (API): Enable API Basic for Service Accounts

Service Account Users:

Configure Auto Provision User

This feature creates a new user in BridgeLink if the username from the OIDC login does not exist. The new user will be assigned the role defined in the Innovar Role-Based Control plugin.

Role Source from OIDC Claims

If you check **"Read roles from token claims,"** the OIDC plugin reads the role name from the claim payload and assigns the matched role name in the Innovar Role-Based Control plugin. You must define the corresponding role privilege in the Innovar Role-Based plugin first.

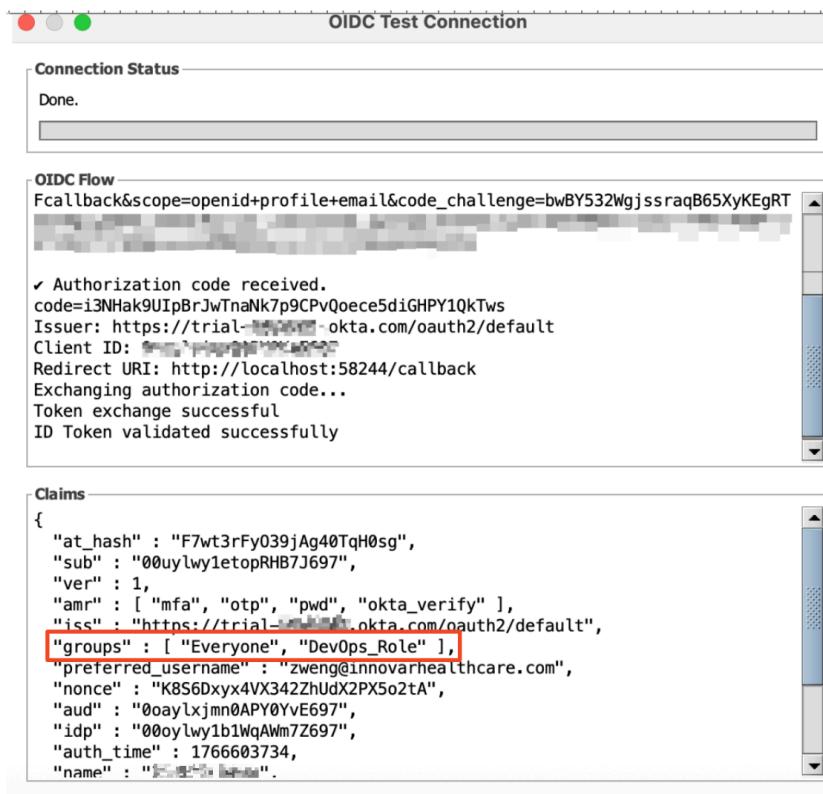
User Provisioning

Auto Provision Users:	<input checked="" type="checkbox"/> Automatically create user if missing
Default Role:	<input type="button" value="DevOps_Role"/>
Use roles from token:	<input checked="" type="checkbox"/> Read roles from token claims
Role source (claim):	<input checked="" type="radio"/> Groups <input type="radio"/> Roles <input type="radio"/> Custom: <input type="text"/>

Role source from OIDC claims

EX: the claim payload from Okta with group name “DevOps_Role”, please select the role source based on the OIDC claim response. For example, select “Groups” if you see the claims like the

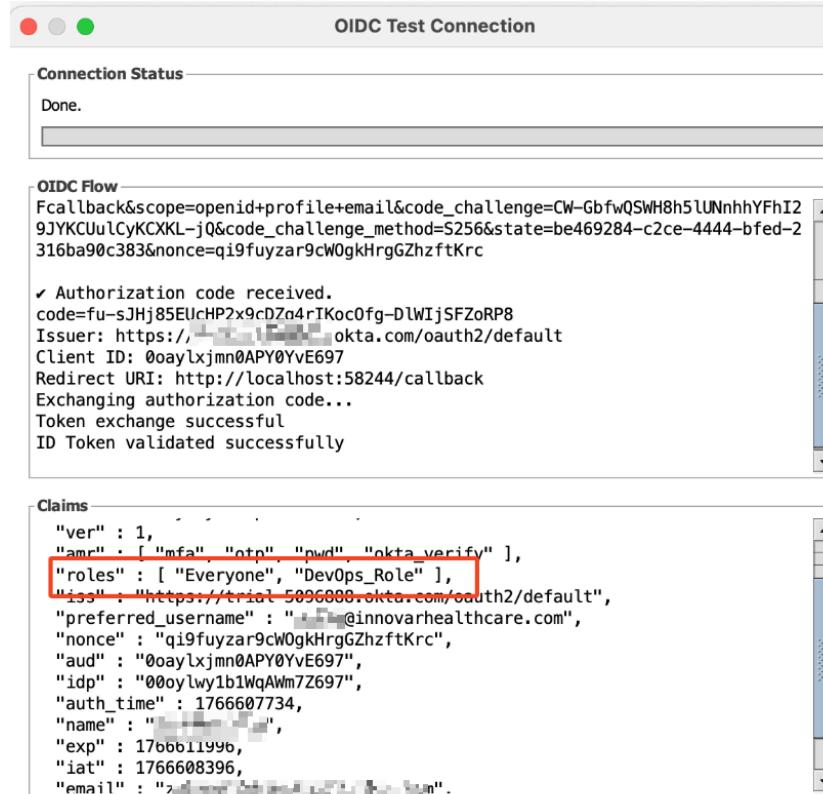
below picture:



The screenshot shows the 'OIDC Test Connection' interface. The 'Claims' section displays a JSON object with various attributes. The 'groups' attribute is highlighted with a red box, showing its value as an array containing 'Everyone' and 'DevOps_Role'.

```
{  
  "at_hash" : "F7wt3rFy039jAg40TqH0sg",  
  "sub" : "00uylw1etopRHB7J697",  
  "ver" : 1,  
  "amr" : [ "mfa", "otp", "pwd", "okta_verify" ],  
  "iss" : "https://trial-[REDACTED].okta.com/oauth2/default",  
  "groups" : [ "Everyone", "DevOps_Role" ],  
  "preferred_username" : "zweng@innovarhealthcare.com",  
  "nonce" : "K8S6Dxyx4VX342ZhUdX2PX5o2tA",  
  "aud" : "0oaylxjmn0APY0YvE697",  
  "idp" : "00oylw1b1WqAwm7Z697",  
  "auth_time" : 1766603734,  
  "name" : "[REDACTED]"}
```

Select “Roles” if you see the claim like the below picture:



The screenshot shows the 'OIDC Test Connection' interface. The 'Claims' section displays a JSON object with various attributes. The 'roles' attribute is highlighted with a red box, showing its value as an array containing 'Everyone' and 'DevOps_Role'.

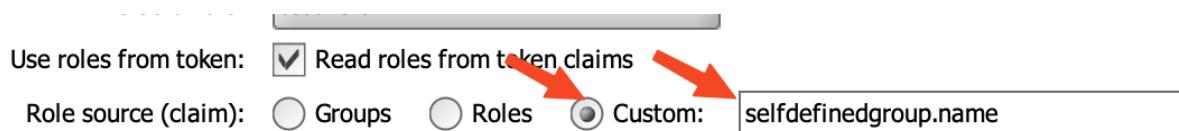
```
{  
  "ver" : 1,  
  "amr" : [ "mfa", "otp", "pwd", "okta_verify" ],  
  "roles" : [ "Everyone", "DevOps_Role" ],  
  "iss" : "https://trial-5096000.okta.com/oauth2/default",  
  "preferred_username" : "[REDACTED]@innovarhealthcare.com",  
  "nonce" : "qi9fuyzar9cW0gkHrgGZhztKrc",  
  "aud" : "0oaylxjmn0APY0YvE697",  
  "idp" : "00oylw1b1WqAwm7Z697",  
  "auth_time" : 1766607734,  
  "name" : "[REDACTED]",  
  "exp" : 1766611996,  
  "iat" : 1766608396,  
  "email" : "[REDACTED]"}
```

Select “custom” and enter the custom pattern in claim

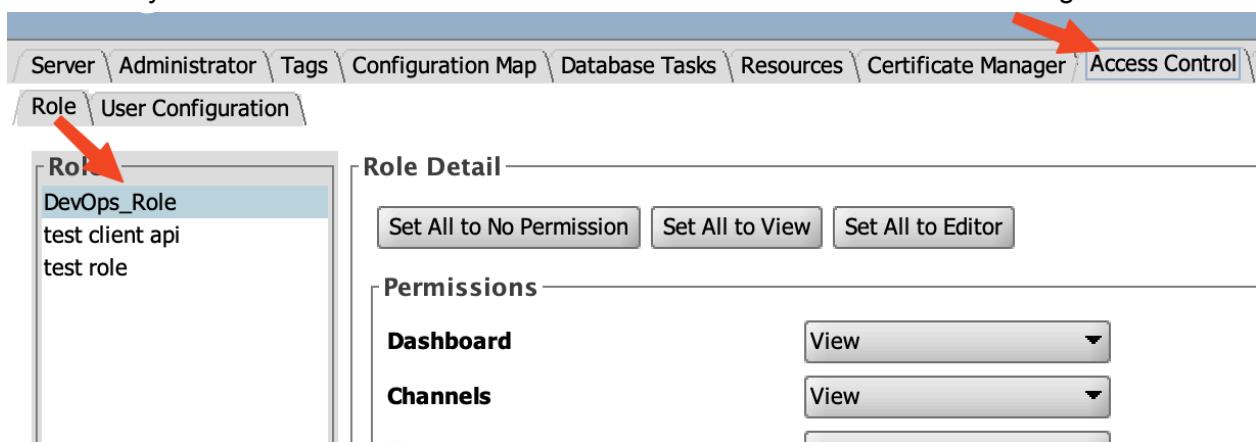
EX: claim response

```
{  
  "selfdefinedgroup": {  
    "name": "DevOps_Role"  
  }  
}
```

Custom Role source claim:



Make sure you have the same role name in the Innovar Role-based Control setting.



This configuration **will automatically create a new user** in BridgeLink—if they do not already exist—upon sign-in with OIDC, assigning them the **DevOps_Role** privilege.

Now you are all set!