GoDocs®

SINGLE SIGN-ON

Client Configuration

OKTA

Embed the GoDocs document order process directly in our LOS to create a fully integrated client experience and access the GoDocs Platform by leveraging direct integration with Third Party Identitiy Providers and Automated User Provisioning.

Date: **July. 15, 2025**

Document Version: **1.2**

Table of Contents

Version History

Version	Changes	Date
V1.0	Initial Document Release	October 31, 2024
V1.1	Updates to step-by-step instructions for Token type setup	April 4, 2025
V1.2	Support for OKTA IDPs	July 15, 2025

Confidentiality Notice

Instructions for OpenId Connect setup for IDP configuration OKTA

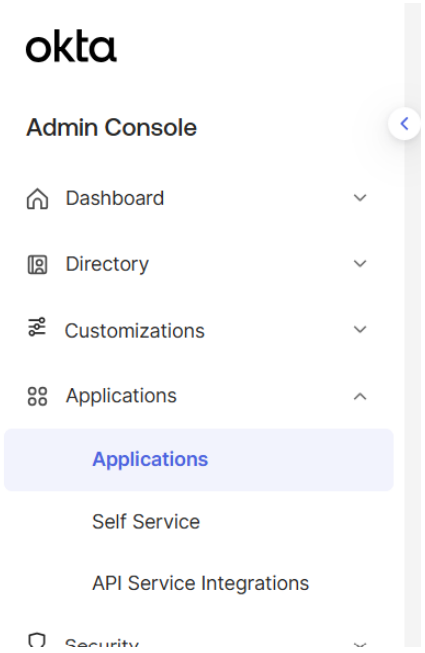
Setup Configurations for IDP Summary:

1. Create an “App Integration” in Okta tenant
2. Give name specific to GoDocs (i.e. godocs-sso)
3. Set Grant Type
    a. Authorization Code will be defaulted
    b. Add in “Implicit (hybrid)” and select “Allow ID Token with implicit grant type”
4. Configure Token to include:
    a. email
    b. family_name
    c. given_name
    d. display_name or name

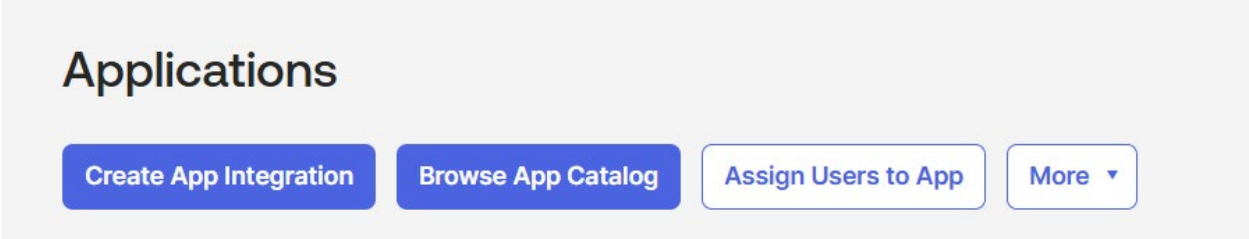
Step-by-step instructions:

Application General Integration Setup:

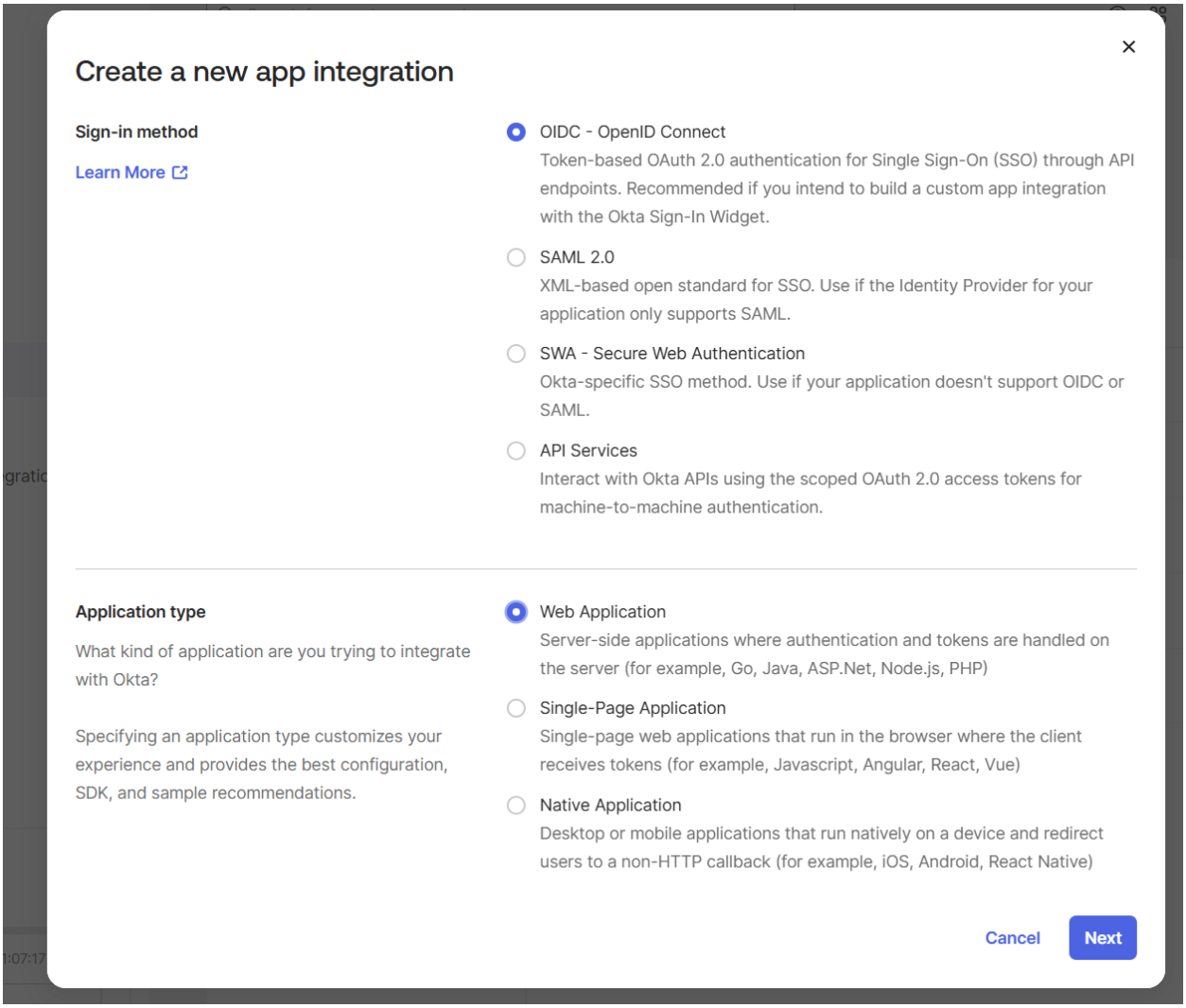
1. In your OKTA Admin Console Navigate “Applications > Applications” Menu option.



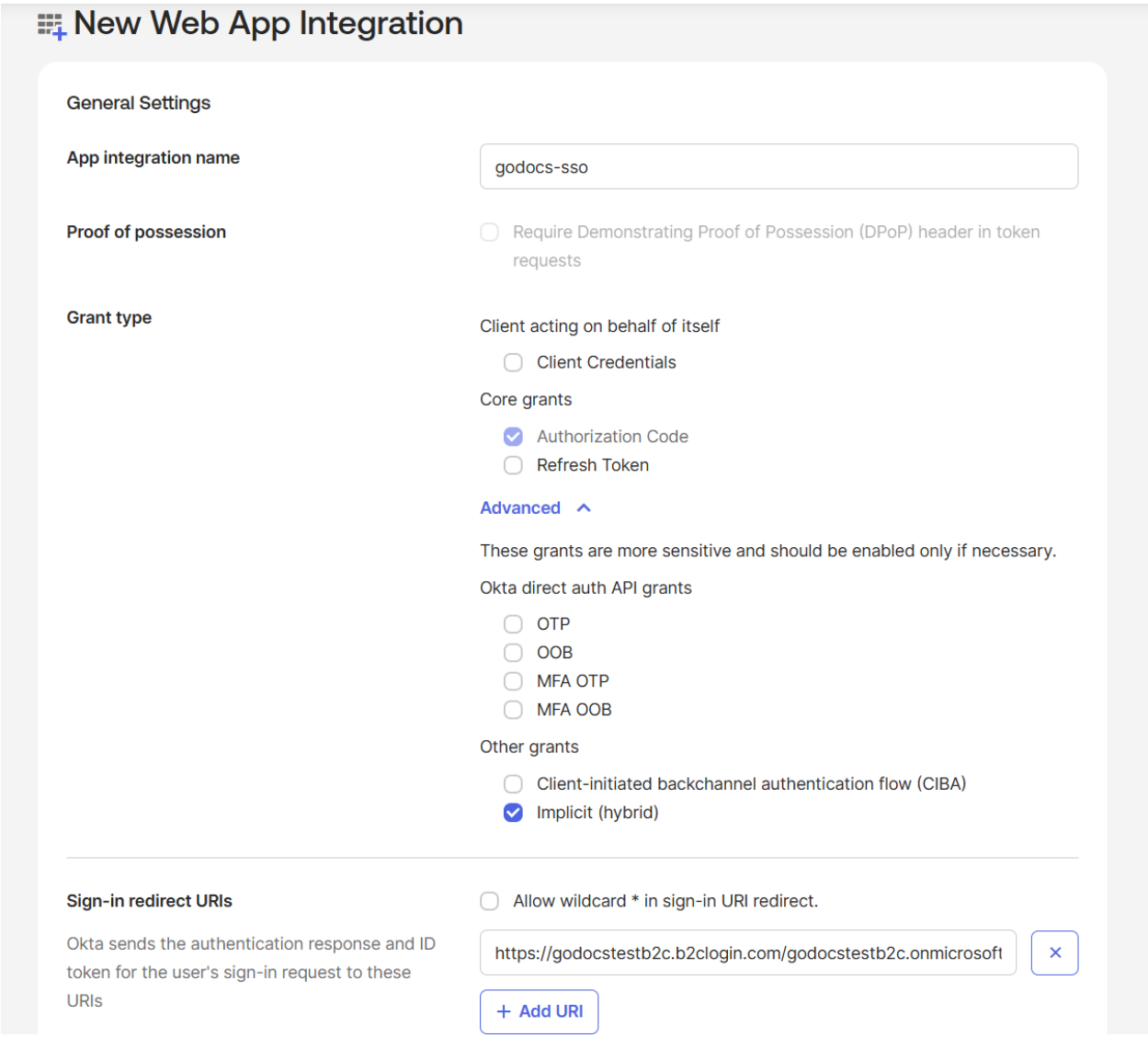
2. Click the “Create App Integration” button



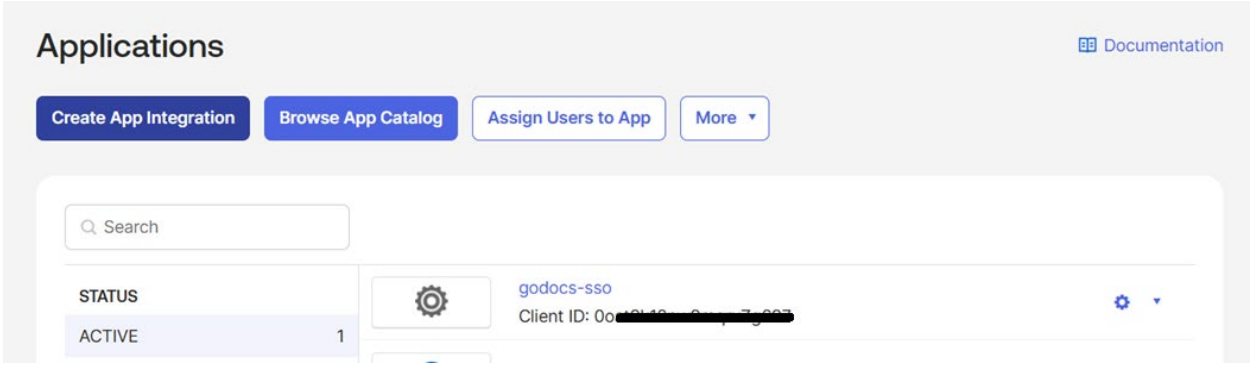
3. Upon clicking the button a dialog will appear.

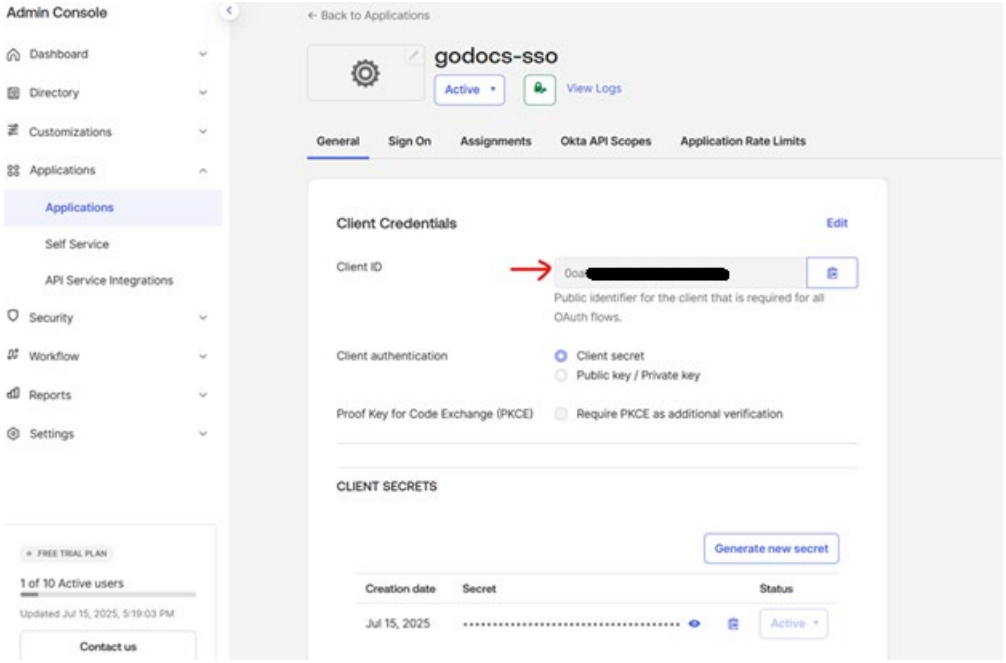


4.
5. Select the radio button “OIDC – OpenId Connect”
6. Select the radio button “Web Application”
7. Click “Next”
8. You will be navigated to the “New Web App Integration” page

9. 

10. Enter a name for the app registration (anything you would like to call it or follow any naming convention your company follows. We do not see this name. i.e. godocs-sso)
11. In Grant Type Section
    a. Authorized code will be defaulted
    b. Click “Advanced”
    c. Check “Implicit (hybrid)”
12. In “Sign-In redirect URIs” section
    a. In the textbox paste in the URL:
        i. https://godocstestb2c.b2clogin.com/godocstestb2c.onmicrosoft.com/oauth2/authresp
        ii. Note: This is our testing tenant URL. We will have a production tenant URL we will send you when ready to go live.
13. Click the “Save” Button

14. In your Applications Dashboard you should see your new App Integration
15. 

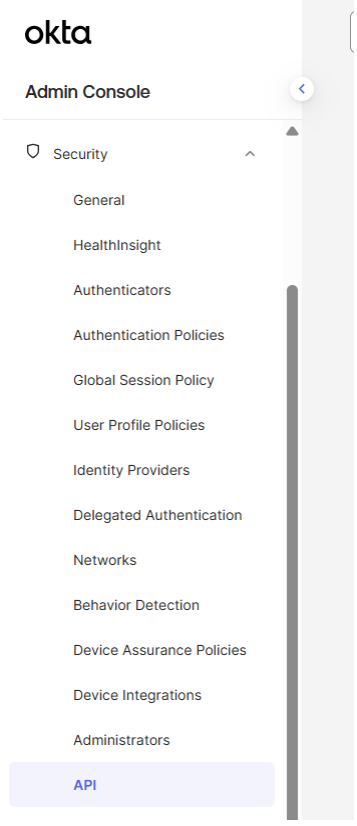
16. Click the App Integration
    a. In the “Client Credential” Section
    b. Copy the Client Id and save for later
    c. 

17. Next we will need this OpenId Connect Metadata URL
    a. It will generally be in this format:
        i. https://{your-company-domain-in-okta} /.well-known/openid-configuration
    b. if you are unsure what your domain is in okta:
        i. Click your name at the top right of the admin console
        ii. In the drop down the you will see your domain
18. Reminder Save Client ID and OpenId Connect Metadata URL you will be providing us with that information.

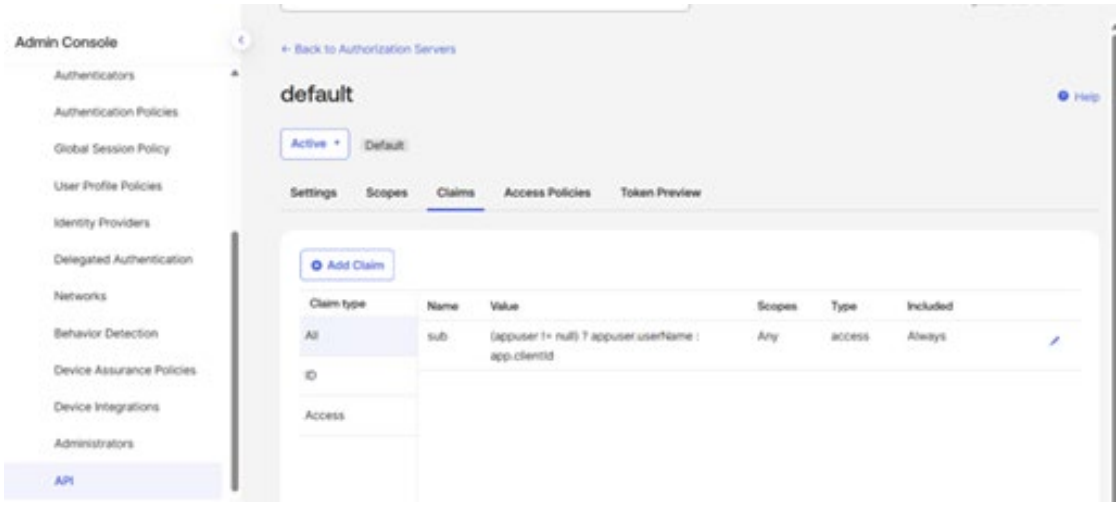
Grant Type Setup:

1. Your Authentication Platform request id_token to authenticate the user during SSO. You need to set up your app integration to be able to send back the correct id_token
2. In you Application Integration Page Scroll down to the “Grant Type” Section
    a. Click “Advanced”
    b. Ensure “Implicit (hybrid)” is checked. If not, please check it
    c. Check “Allow ID Token with implicit grant type”
    d. Click “Save” Button

Token Configuration:

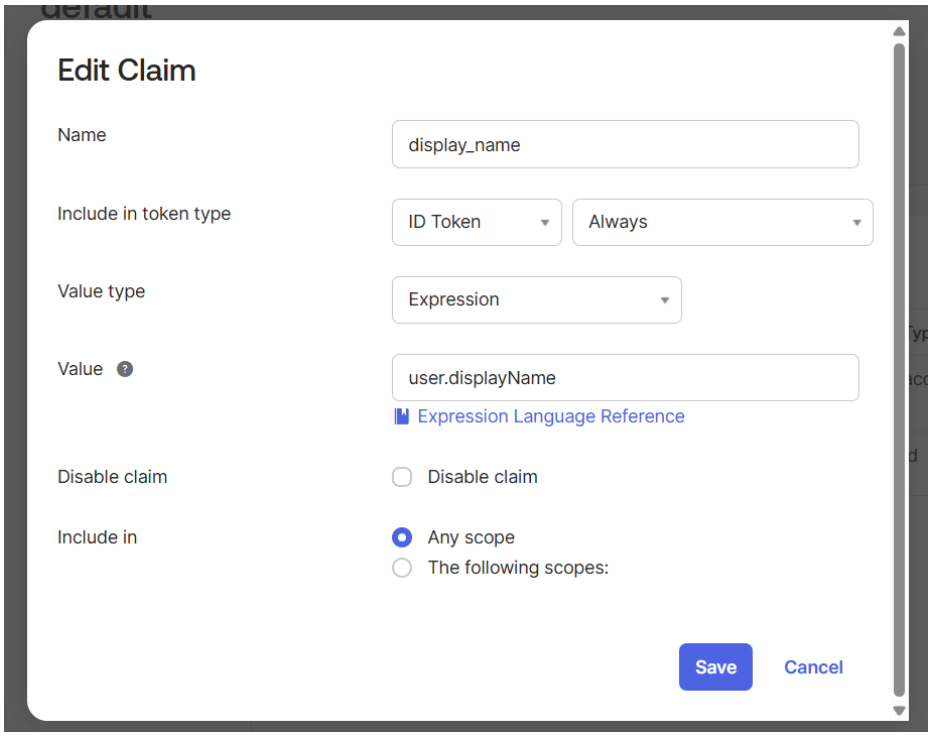
1. GoDocs Authentication Platform requires 4 claims to authenticate the user.
    a. First Name
    b. Last Name
    c. Email
    d. Display Name
2. You need to configure your token that gets generated in OKTA to provide those claims during the token exchange process.
3. To do this/Verify this Navigate to “Security > API”
4. 

5. In the API Dashboard click the “Authorization Servers” tab
6. In this dashboard you will see all your Authorization Servers. Click the one that pertains to us. (Could be default or however your admin console is set up)
7. In this example we use “default”
8. Click the name of the authorization server.

9. You will navigate the authorization server page of that specific authorization server.
10. Click the “Claims” tab
    a. 

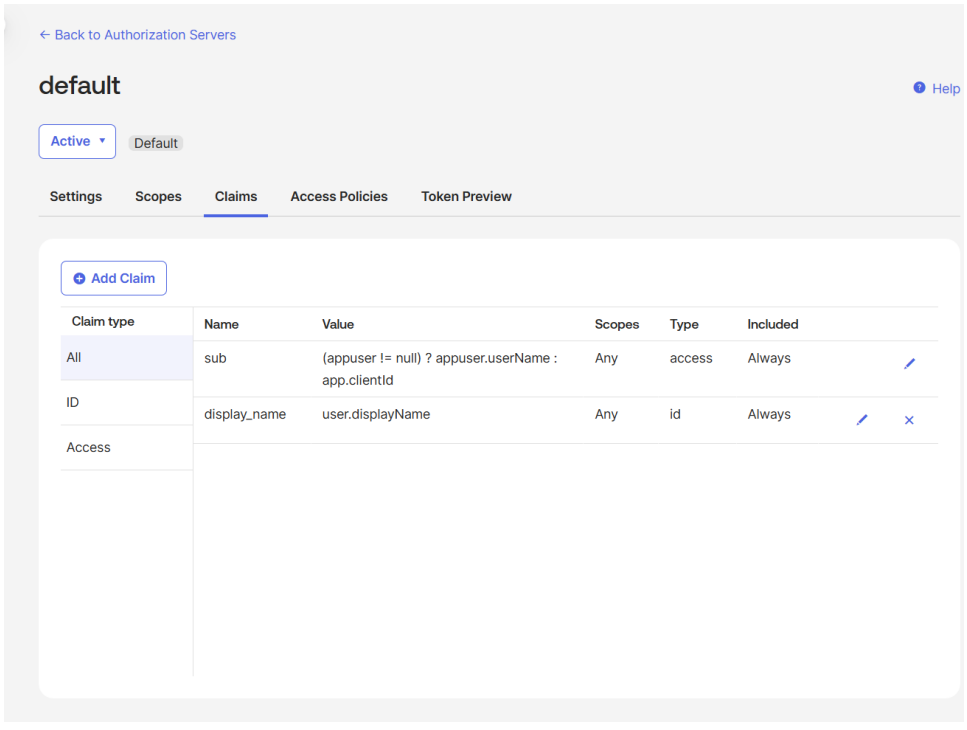
    b. OPTIONAL: (If your system handles the “name” claim differently proceed to adding a custom claim)
    c. In here we will add a custom claim “display_name” (we use this claim to store the display name of the user in GoDocs’ Authentication Platform)
    d. To Add click “Add Claim”
    e. A dialog will appear add the following data:
        i. Name -> display_name
        ii. Include in Token type -> ID Token, Always
        iii. Value type -> Expression
        iv. Value -> user.displayName
        v. Include in -> Any Scope (configure how your system is setup for scoping)
        vi. Click “Save” Button

vii.

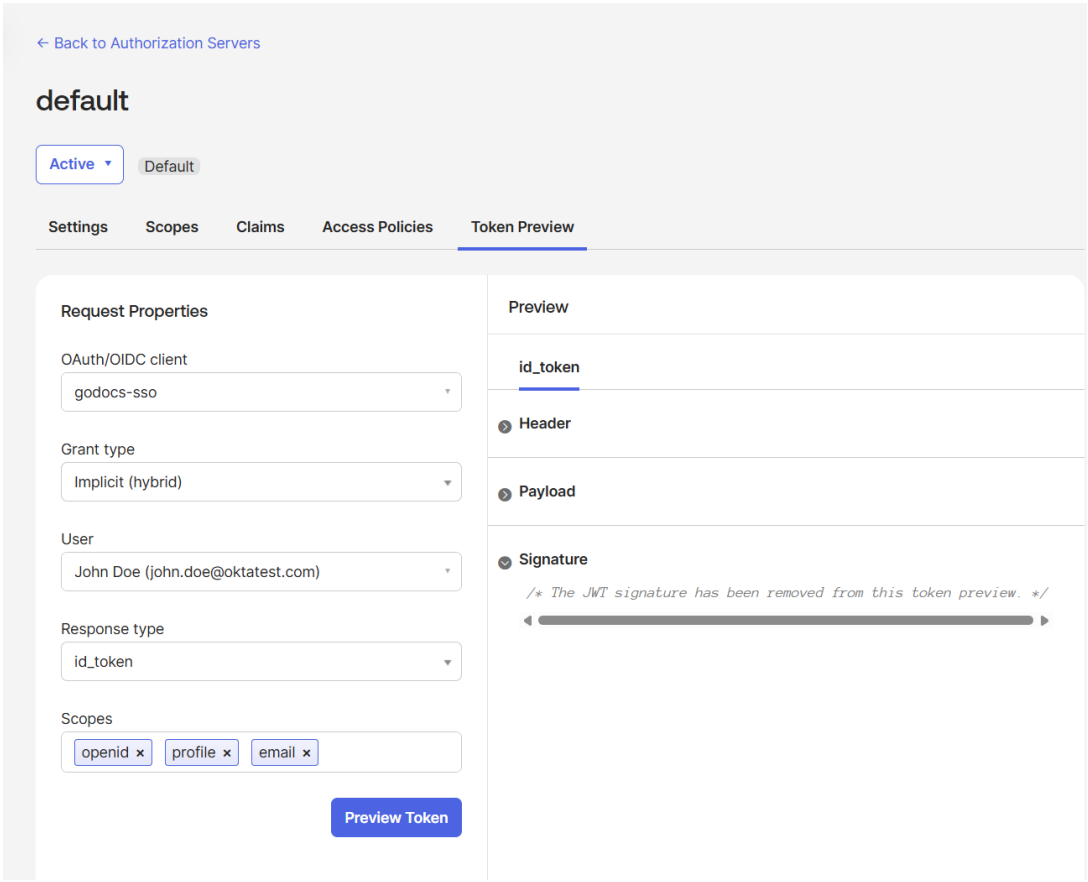


viii. You should now see the claim added to your claims tab

ix.



Token Testing:

1. To test your new set up and see what your system will be sending back to GoDocs' Authentication platform you can test a toke generation.
2. In the same Authorization Server click the "Token Preview" tab
3. Here you will enter information that you set up for GoDocs SSO
4. In the Request Properties Section
    a. OAuth/OIDC client -> set to the app integration you created for godocs
    b. Grant Type -> Implicit (hybrid)
    c. User -> select any user in your IDP
    d. Response Type -> id_token
    e. Scopes -> openid, profile, email (these will be the scope GoDocs will be requesting in is authentication request)
        i. In Okta these Scope will provide us with the user's email, family_name, given_name and name
    f. Click "Preview Token"
    g. On the right hand side your token will generate.
    h. NOTE: if your token preview does not work you will have to update your access policies in the "Access Policies" Tab
    i. 

    j. If your token includes the 4 claims GoDocs needs (email, given_name, family_name, display_name or name) then you are complete.

k. Please save a sample of the token output save for later.

19. That's it. Send GoDocs the "client id" and "OpenId Connect metadata document" uri and sample of the token output you saved earlier.