

# Configuring OIDC as third-party IDP in Access (Auth0)

WorkspaceONE Access now supports acting as a relying party in an Open ID Connect flow of authentication. (SaaS 2012 release of Access)

In this lab we will be adding **Auth0** as our identity provider that we will integrate with WorkspaceONE Access.

Part 1: Sign up for Auth0 Trial

Part 2: Adding Auth0 as a third-party IDP in Workspace ONE

Part 3: Test authentication flow

## Part 1: Sign up for Auth0 Trial

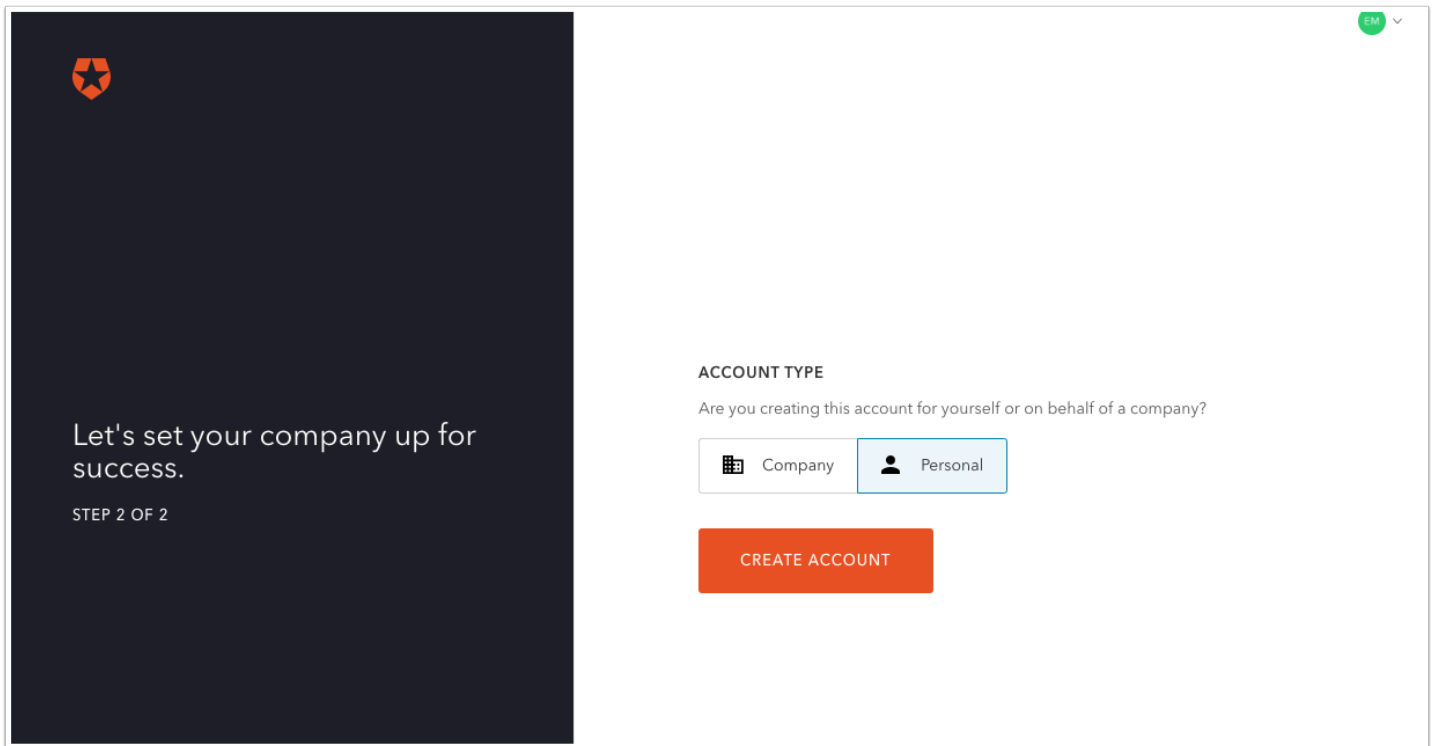
In this part we will be signing up for a free Auth0 trial account and setting up the the Auth0 environment.

1. Open a new browser window and navigate to <https://auth0.com/signup>

- Type a demo **email address** (Do **NOT** use your corporate e-mail address)
- And a **password**
- Click **SIGN UP**

2. Specify the **domain suffix** that you would like to use. Ensure it is easy to remember and that you make note of it.

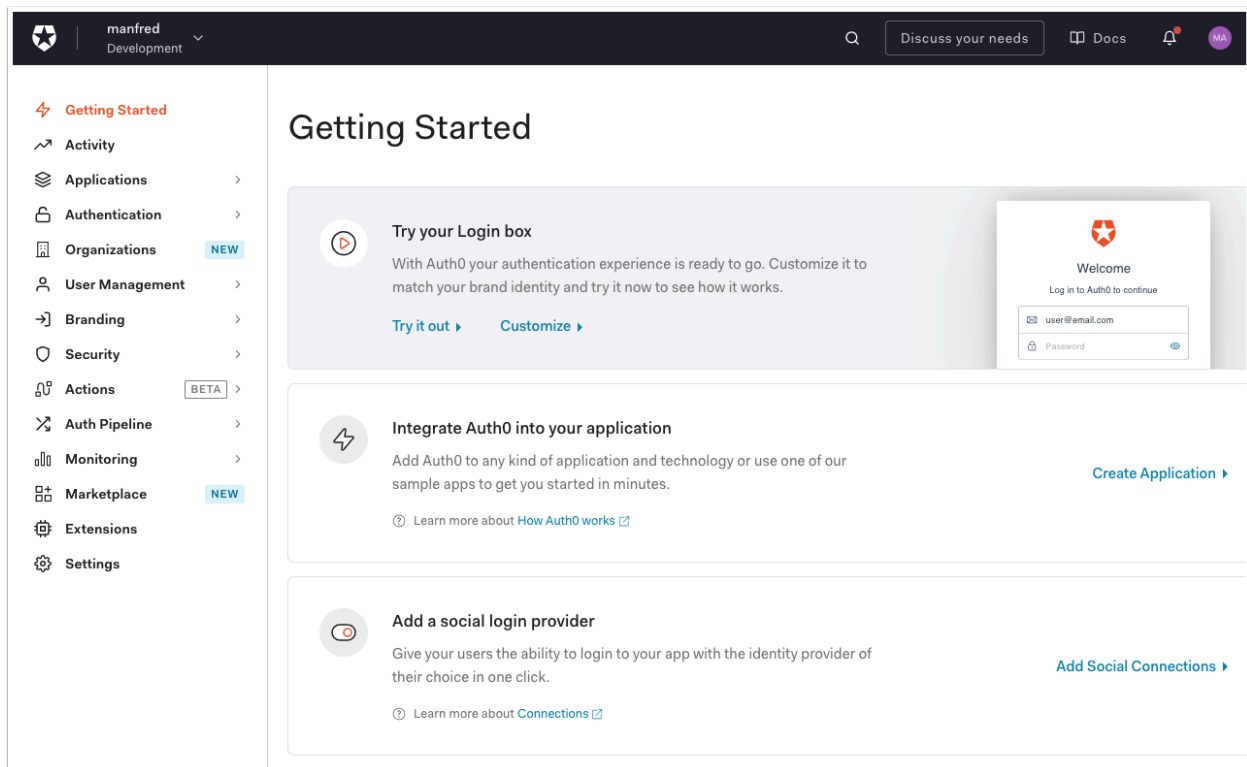
- Click **US** for Region.
- Click **NEXT**.



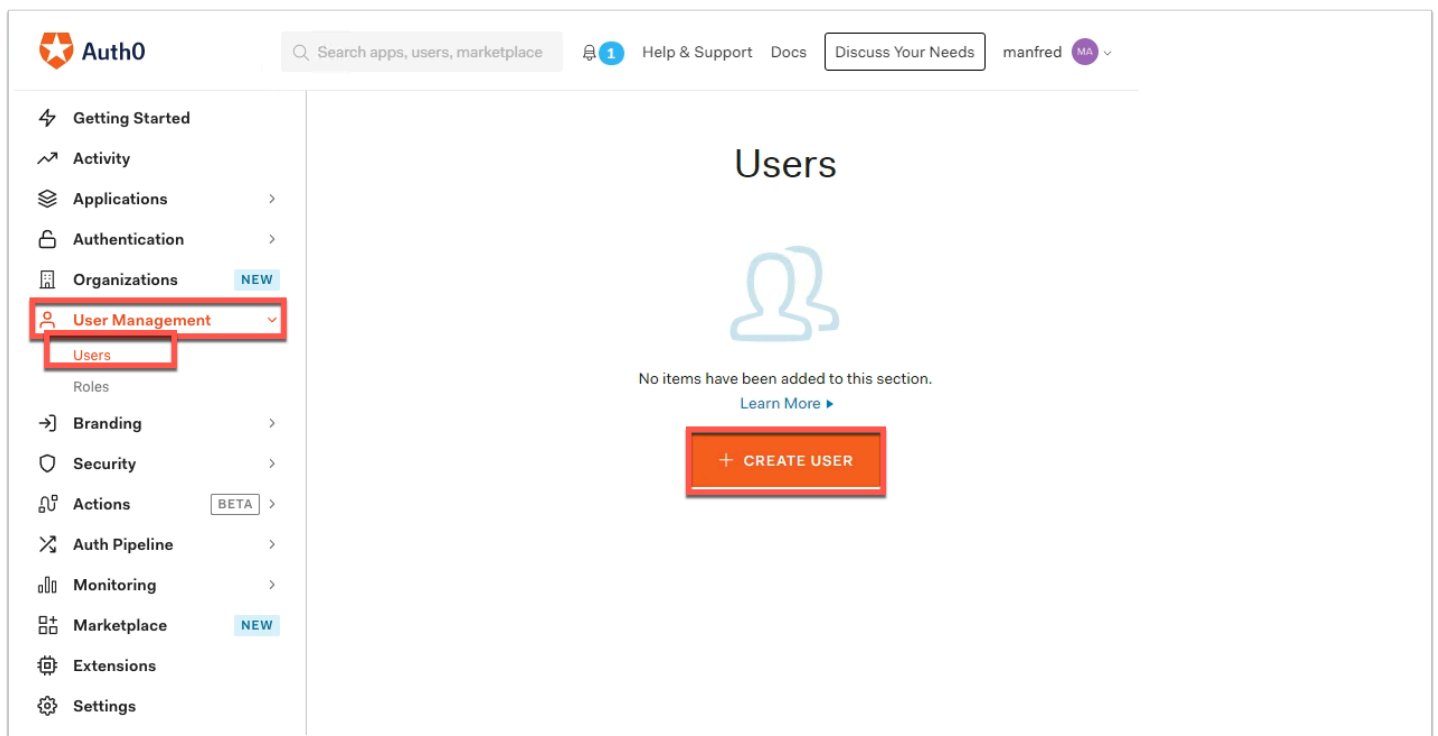
The screenshot displays a two-column layout. The left column has a dark background with an orange star icon at the top left. It contains the text "Let's set your company up for success." and "STEP 2 OF 2". The right column has a white background. At the top right, there is a green circular profile icon with the letters "EM" and a dropdown arrow. Below this, the heading "ACCOUNT TYPE" is followed by the question "Are you creating this account for yourself or on behalf of a company?". There are two selection buttons: "Company" with a building icon and "Personal" with a person icon. The "Personal" button is highlighted with a blue border. Below these buttons is a large orange button labeled "CREATE ACCOUNT".

3. Set the **Account Type** option to **Personal**

- and click **CREATE ACCOUNT**



4. You should now be in the a **Auth0 Admin Console**.



5. In the **Auth0 Admin console** navigate to **User Management** on the left and click **Users**.

- Now click on **+ CREATE USER**

Create user

Email \*  
mark@euc-liveware.com

Password \*  
.....

Repeat Password \*  
.....

Connection \*  
Username-Password-Authentication

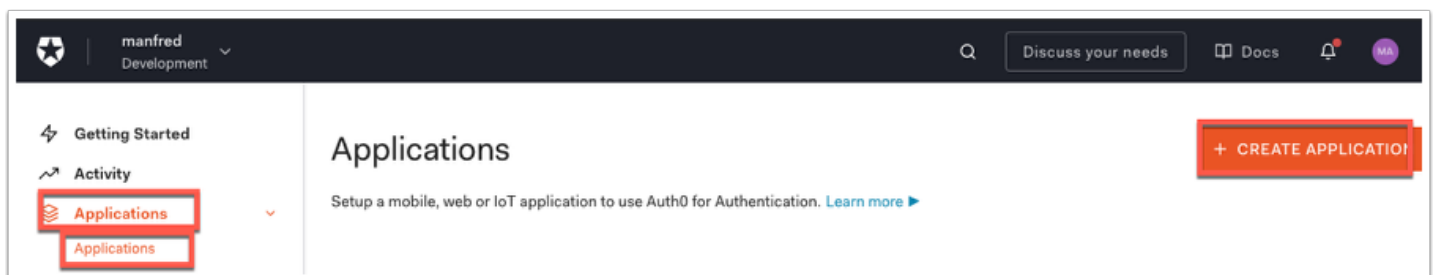
CANCEL CREATE

6. Fill in the **email** of the unique user you have been using for the labs. (example: mark@euc-liveware.com)

- Then type **VMware1!** as the password.
- Click **CREATE** at the bottom of the page.

## Part 2: Adding Auth0 as a third-party IDP in Workspace ONE

In this section you will configure Auth0 as a third-party OIDC IDP in Workspace ONE Access.



1. In the **Auth0 Admin Console** navigate to **Applications > Applications** and then click **+ CREATE APPLICATION**

Create application

×

Name \*

WorkspaceONE Access

You can change the application name later in the application settings.

Choose an application type

Native

Mobile, desktop, CLI and smart device apps running natively.

e.g.: iOS, Electron, Apple TV apps

Single Page Web Applications

A JavaScript front-end app that uses an API.

e.g.: Angular, React, Vue

Regular Web Applications

Traditional web app using redirects.

e.g.: Node.js Express, ASP.NET, Java, PHP

Machine to Machine Applications

CLIs, daemons or services running on your backend.

e.g.: Shell script

CREATE


CANCEL

2. In the Create Application pop-up window type **WorkspaceONE Access** as the Name for the application.

For the Application type select **Regular Web Applications**

Then click **CREATE**

← Back to Applications



WorkspaceONE Access

REGULAR WEB APPLICATION Client ID 2wTswzDCgBj1P18aQM1iUsDFdAX00n9Q

Quick Start

Settings

Addons

Connections

Basic Information

Name \*

WorkspaceONE Access

Domain

sfrank.us.auth0.com

Client ID

2wTswzDCgBj1P18aQM1iUsDFdAX00n9Q

Client Secret

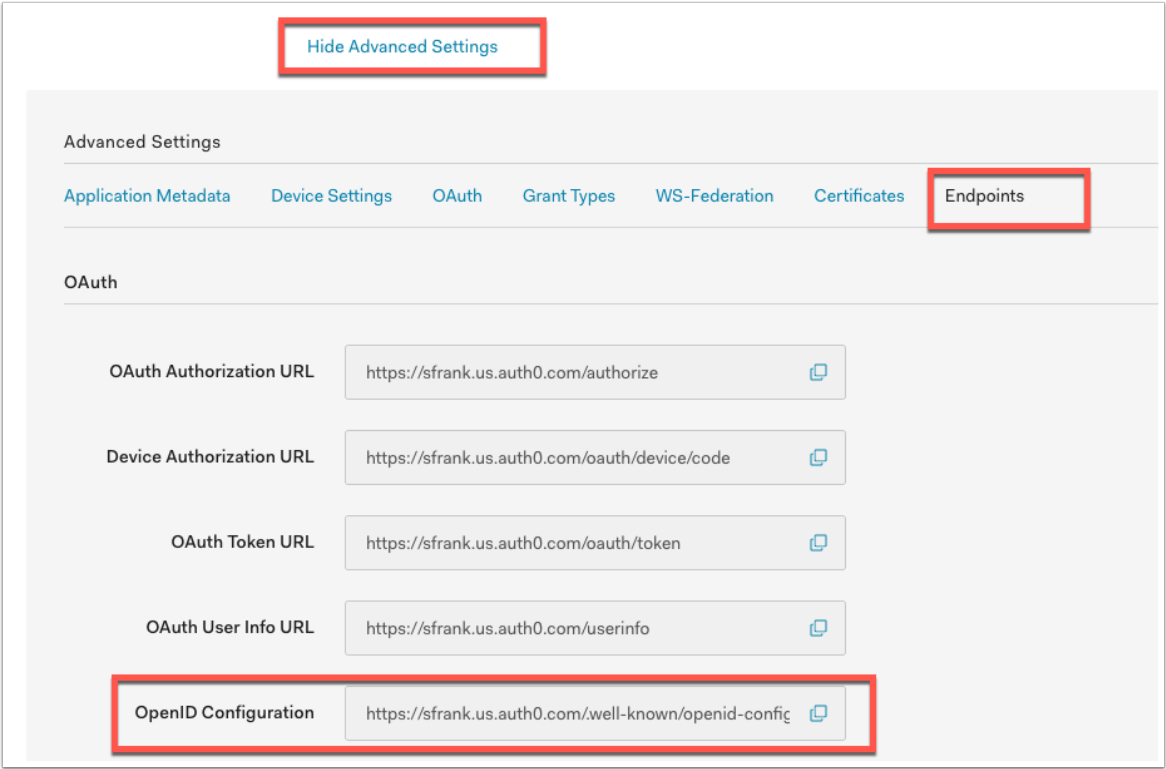
.....

The Client Secret is not base64 encoded.

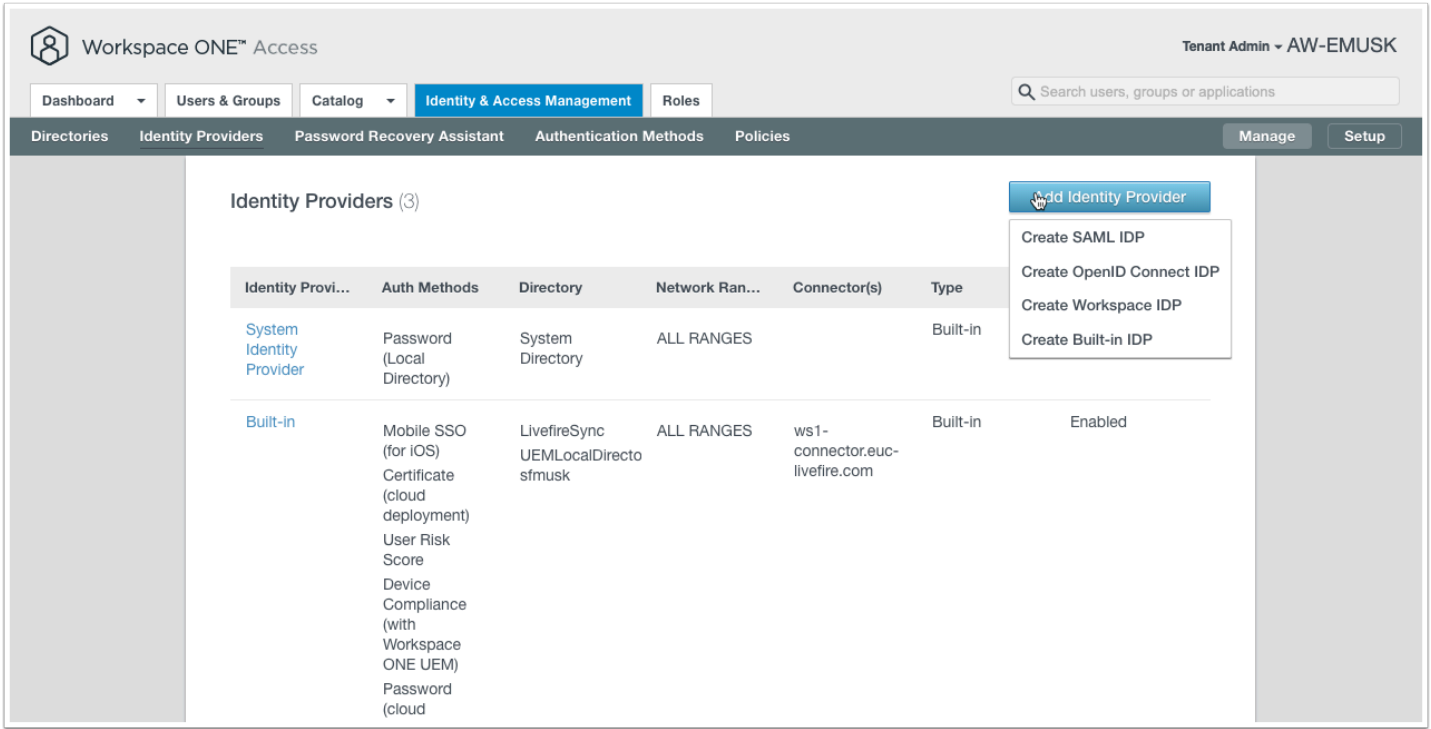
Description

Add a description in less than 140 characters

3. Click on the **Settings** tab and copy the **Client ID** and **Client Secret** to a notepad.

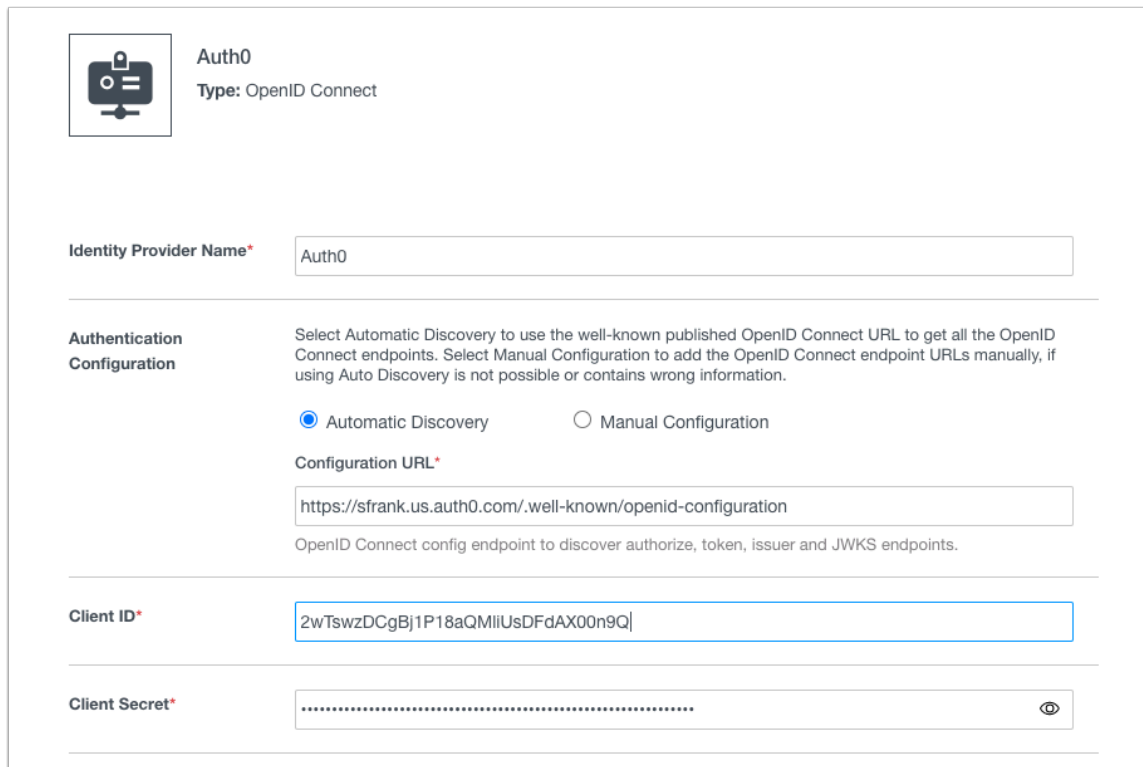


4. Still on the **settings** tab, scroll down until you see "**Show Advanced Settings**" Click it and click on **Endpoints**. The **OAuth** URLs will be displayed. Copy the **OpenID Configuration** into your notepad.



## Configure Workspace ONE Access

5. Open a new tab in your browser and authenticate to your **Workspace ONE Access** tenant. Navigate to **Identity & Access Management > Identity Providers** and click **Add Identity Provider > Create OpenID Connect IDP**



The screenshot shows the 'Add Identity Provider' form in the Workspace ONE Access console. The form is titled 'Auth0' and 'Type: OpenID Connect'. It contains the following fields and options:

- Identity Provider Name\***: A text input field containing 'Auth0'.
- Authentication Configuration**: A section with a description: 'Select Automatic Discovery to use the well-known published OpenID Connect URL to get all the OpenID Connect endpoints. Select Manual Configuration to add the OpenID Connect endpoint URLs manually, if using Auto Discovery is not possible or contains wrong information.' Below this are two radio buttons: 'Automatic Discovery' (selected) and 'Manual Configuration'.
- Configuration URL\***: A text input field containing 'https://sfrank.us.auth0.com/.well-known/openid-configuration'. Below the field is a note: 'OpenID Connect config endpoint to discover authorize, token, issuer and JWKS endpoints.'
- Client ID\***: A text input field containing '2wTswzDCgBj1P18aQMIIUsDFdAX00n9Q|'.
- Client Secret\***: A text input field with masked characters (dots) and a toggle icon on the right.

5. Set the name of the **Identity Provider** to **Auth0**

Authentication Configuration: Automatic Discovery

**Paste** the **OpenID Configuration** URL from Auth0 into the Configuration URL

**Paste** the **Client ID** from your notepad

**Paste** the **Client Secret** from your notepad



User Lookup Attribute	Open ID User Identifier Attribute	Workspace ONE Access User Identifier Attribute
	email	email

---

**Just-in-Time User Provisioning** ☐ Enable

---

**Users** Select which users can authenticate using this IdP. Choose from the available directories from the list below.

☒ LivefireSync  
☐ UEMLocalDirectory\_sfmusk

---

**Network** Select which networks this IdP can be accessed from. Choose from the available network ranges from the list below.

☒ ALL RANGES  
☒ Internal

6. Now set the **User Lookup Attribute** to **email** for both OpenID User and Workspace ONE

Click **LivefireSync** for the **Users** directory

Click **All Ranges** for **Network**

**Authentication Method Name\*** Enter a name to identify the OpenID Connect authentication method. When you select this authentication method in the access policy, users are redirected to authenticate against the OpenID Connect authorization server.

Auth0

---

**Pass through Claims** ☐ Enable pass through claims to add non-standard OpenID Connect claims. Non-standard claims are sent by third party OpenID Connect identity provider and added to the token generated by Workspace ONE Access.

---

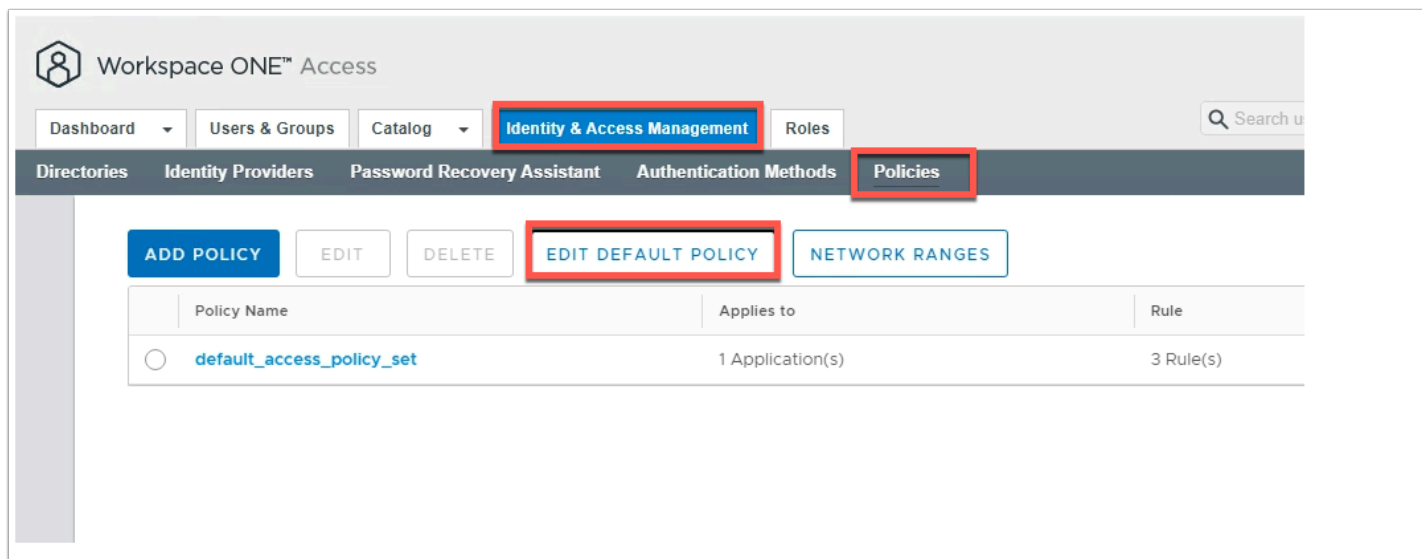
**Redirect URI** Integrate with Open ID Connect Provider using Redirect URI below

https://aw-emusk.vidmpreview.com/federation/auth/response/oauth2

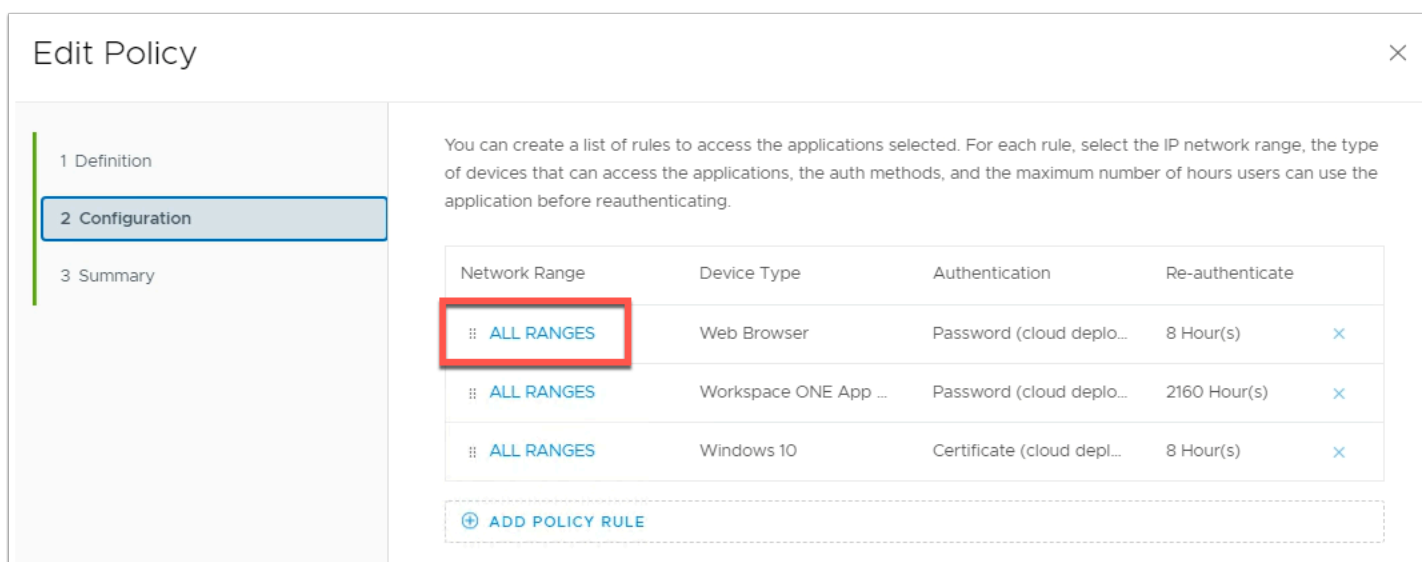
7. Set the Authentication Method name to **Auth0**

**Copy** the **Redirect URI** into your notepad.

Click **Add** to save the settings.



8. Click **Identity & Access Management** > **Policies** and click **EDIT DEFAULT POLICY**



9. In the **Edit Policy** window click **Configuration** then choose the policy with **Web Browser** Device Type and click **ALL RANGES**.

If a user's network range is \* ALL RANGES

and the user accessing content from \* Web Browser

and user belongs to group(s) Select Groups...

Rule applies to all users if no group(s) selected.

and user is registering FIDO2 authenticator \* No

Then perform this action Authenticate using...

then the user may authenticate using \* Auth0

If the preceding method fails or is not applicable, then Password (Local Directory)

+ ADD FALLBACK METHOD

CANCEL SAVE

10. In the **Edit Policy Rule** pop-up change the first authentication method to **Auth0** and leave the fall back method as **Password (Local Directory)**

Click **SAVE** at the bottom of the pop-up.

Edit Policy

1 Definition

2 Configuration

3 Summary

You can create a list of rules to access the applications selected. For each rule, select the IP network range, the type of devices that can access the applications, the auth methods, and the maximum number of hours users can use the application before reauthenticating.

Network Range	Device Type	Authentication	Re-authenticate
ALL RANGES	Web Browser	Auth0+1	8 Hour(s)
ALL RANGES	Workspace ONE App ...	Password (cloud depl...	2160 Hour(s)
ALL RANGES	Windows 10	Certificate (cloud depl...	8 Hour(s)

+ ADD POLICY RULE

CANCEL BACK NEXT

Edit Policy

1 Definition

2 Configuration

3 Summary

Definition

Name default\_access\_policy\_set

Description Default access policy set

Applications 1 Application(s)

Configuration

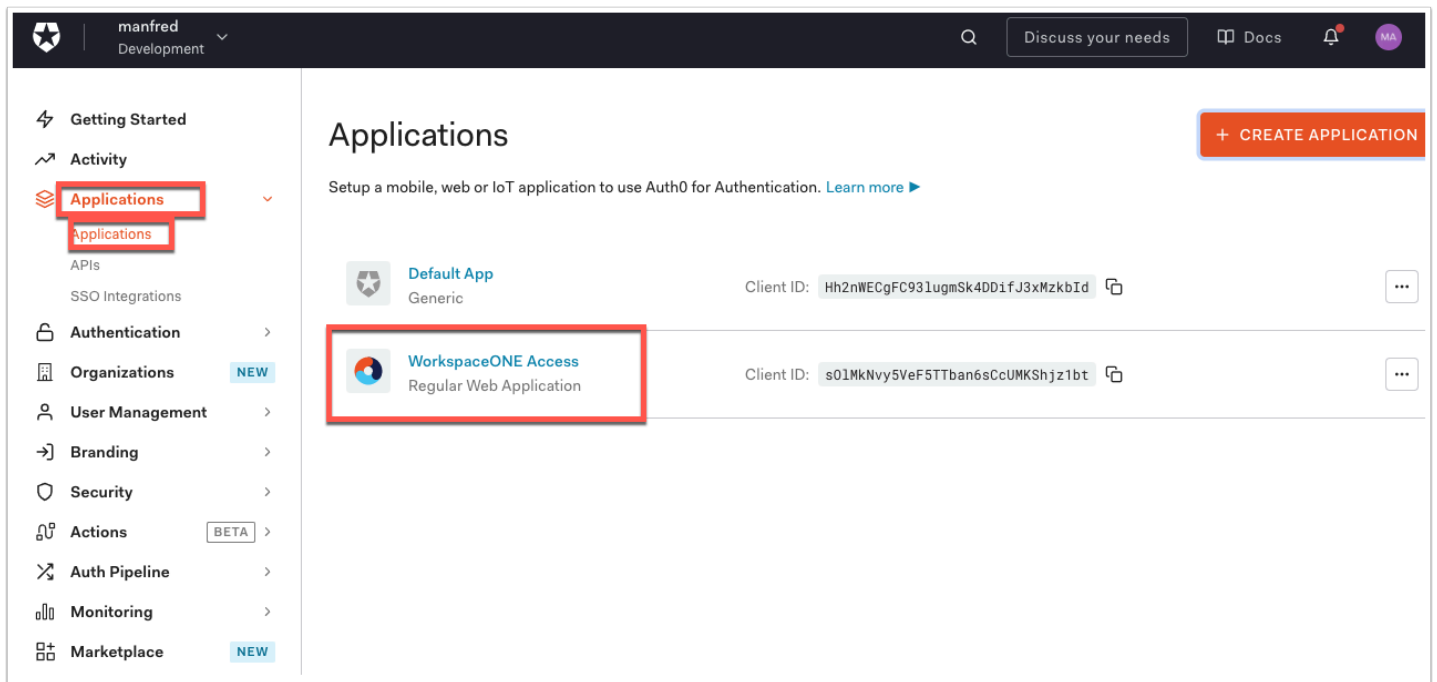
Policy Rule 1

If a user's network range is ALL RANGES and the user is accessing content from Web Browser and the user belongs to the group(s) All Users then the user may authenticate using Auth0

Fallback method: Password (Local Directory)

CANCEL BACK SAVE

11. Click **Next** on the **Configuration** page and **SAVE** on the **Summary** page of the **Edit Policy** window.



12. Flip back to the **Auth0** Admin Console and click **Applications > Applications**. Click on the **WorkspaceONE Access** application.

'Basic' (application uses HTTP Basic).

### Application URIs

Application Login URI	<input type="text" value="https://myapp.org/login"/>
<p>In some scenarios, Auth0 will need to redirect to your application's login page. This URI needs to point to a route in your application that should redirect to your tenant's <code>/authorize</code> endpoint. <a href="#">Learn more</a></p>	
Allowed Callback URLs	<input type="text" value="https://aw-emusk.vidmpreview.com/federation/auth/response/oauth2"/>
<p>After the user authenticates we will only call back to any of these URLs. You can specify multiple valid URLs by comma-separating them (typically to handle different environments like QA or testing). Make sure to specify the protocol ( <code>https://</code> ) otherwise the callback may fail in some cases. With the exception of custom URI schemes for native clients, all callbacks should use protocol</p>	

13. In the **Settings** tab of the application scroll down to Application URIs and **paste** the **Redirect URI** from Access into the **Allowed Callback URLs** in Auth0.

lifetime, after which the token can no longer be used. If rotation is enabled, an expiration lifetime must be set. [Learn More](#)

Absolute Lifetime

Sets the absolute lifetime of a `refresh_token` (in seconds).

Inactivity Expiration ☐ DISABLED

When enabled, a `refresh_token` will expire based on a specified inactivity lifetime, after which the token can no longer be used.

Inactivity Lifetime

Sets the inactivity lifetime of a `refresh_token` (in seconds).

[Show Advanced Settings](#)

**SAVE CHANGES**

**Danger Zone**

**Delete this application**

All your apps using this client will stop working. **DELETE**

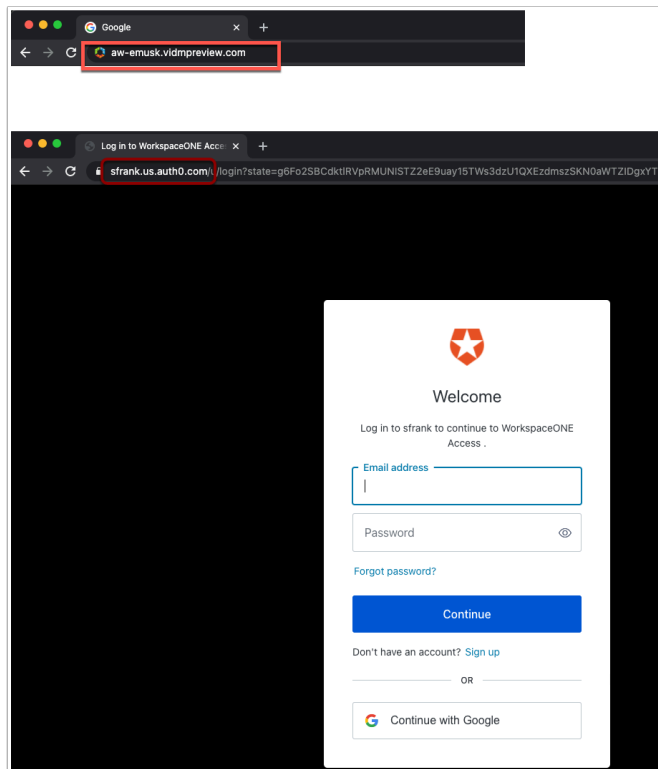
**Rotate secret**

All authorized apps will need to be updated with the new client secret. **ROTATE**

14. At the bottom of the page click **SAVE CHANGES**.

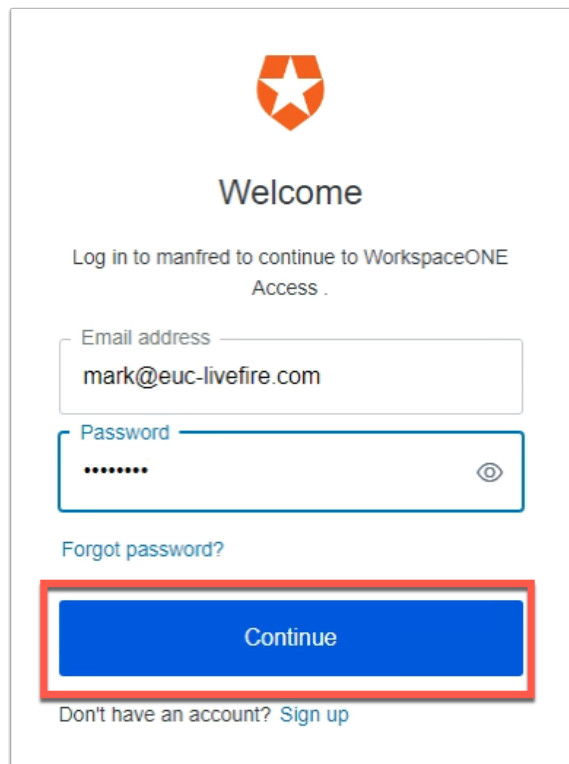
## Part 3: Test authentication flow

Let's now test the result of our integration with Auth0 as an OIDC IDP.

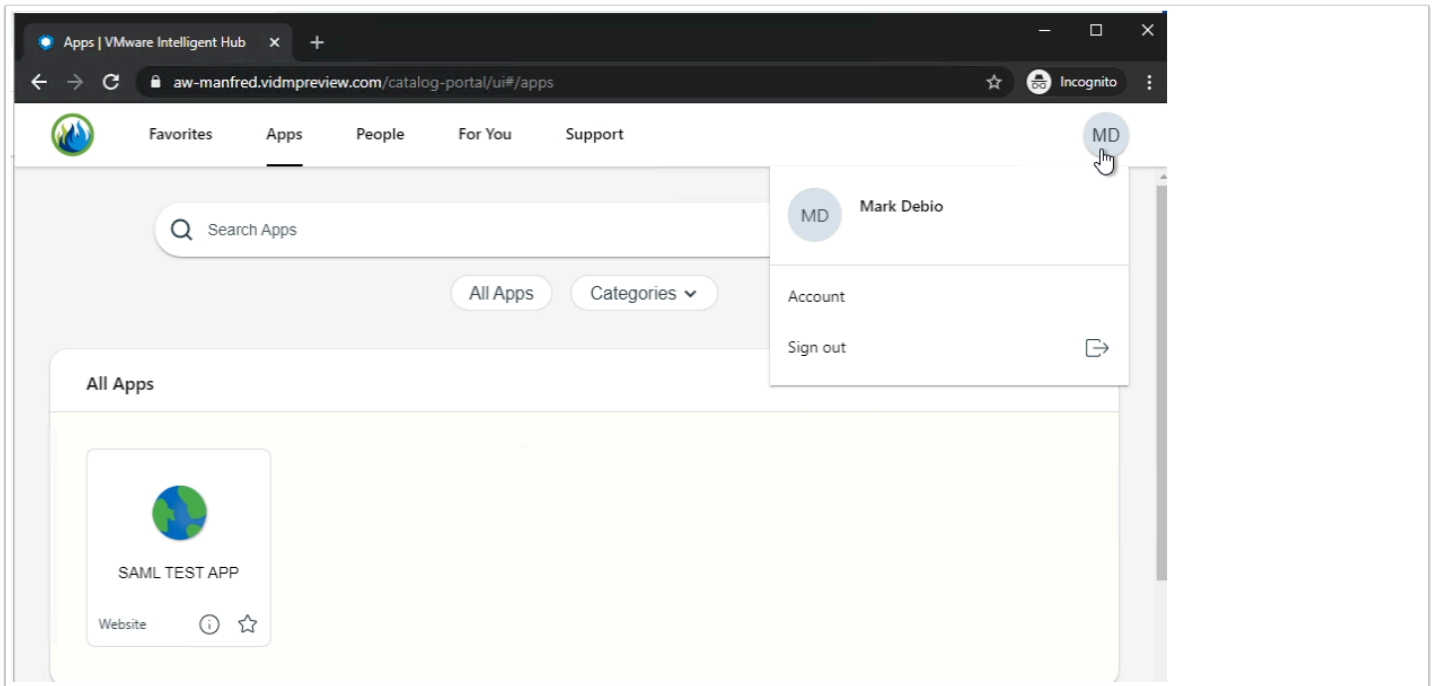


1. Open a new **Incognito window** and browse to your **Workspace ONE Access tenant**. As soon as you click **ENTER** you will be re-directed to Auth0

**Notice** the URL is your unique tenant for Auth0.



2. Enter the **E-mail address** of the user (**NOT** the admin account) and the password **VMware1!**  
Click **Continue**



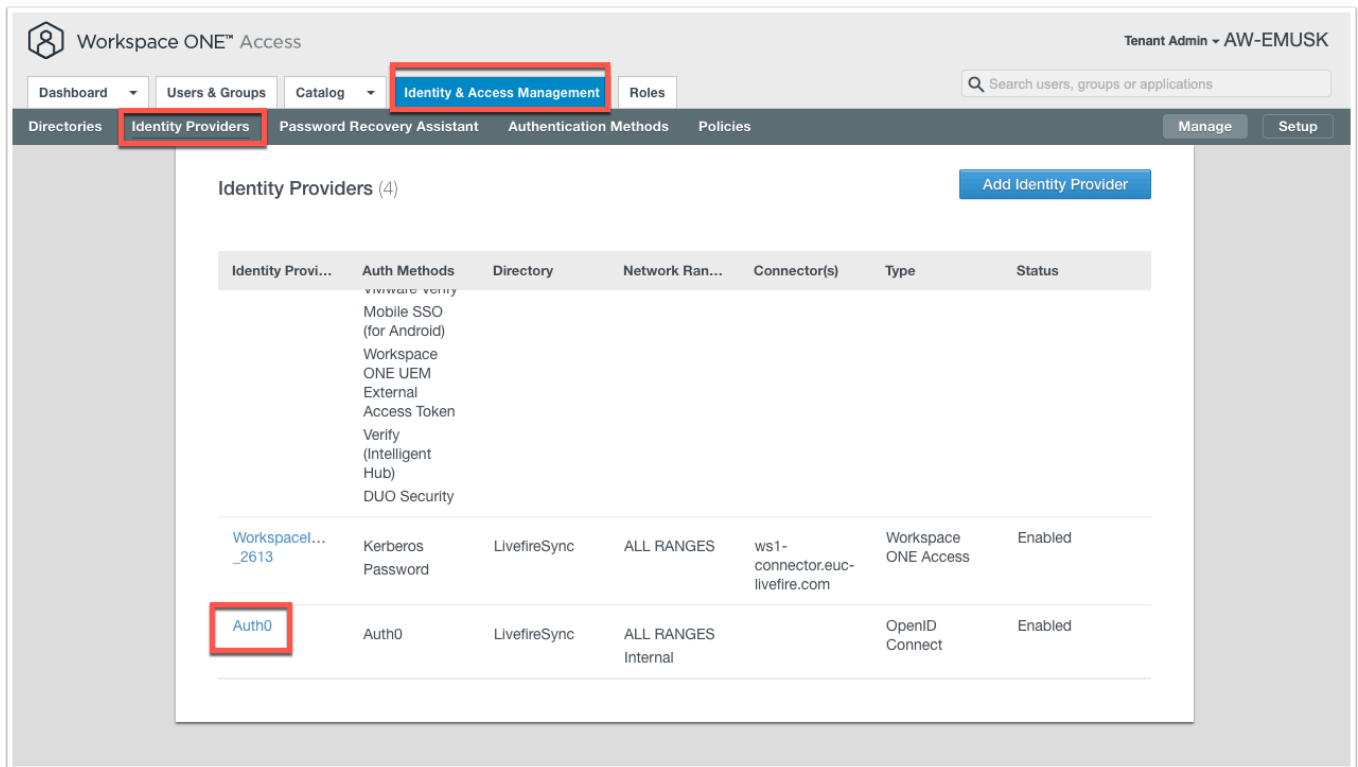
3. Now you should have been authenticated to your **Workspace**, double check your user in the top right hand corner of the Hub.

**VERY IMPORTANT IF** you are not planning on continuing to the bonus material please change your default access policy at this point. In order to ensure you can log back in as local administrator into Workspace ONE Access you will have to append your Access URL with **/SAAS/auth/0** (Example: [aw-rfederer.vidmepreview.com/SAAS/auth/0](#)) you will be able to log on and change the **default access policy** from Auth0 to Certificate (**cloud deployment**) with a fallback to **Password (cloud deployment)** with a fallback to **Password (Local Directory)**.

This concludes the integration with Auth0 as a third party OIDC Identity Provider.

## Bonus Material 1: Just-In-Time Provisioning

In addition to the above integration Auth0 can act as a source of identity for your users and can create a user database on Access using JIT (Just-In-Time provisioning). The users get provisioned in Access as part of the initial authentication process.



1. In the **Workspace ONE Access Admin** console navigate to **Identity & Access Management** > **Identity Providers** and click on **Auth0** the Identity Provider you created above.

**Just-in-Time User Provisioning** ☒ Enable

Just-in-Time Directory: Auth0-Directory

User Attribute Mappings\* Map the OpenID Connect attributes to the Workspace ONE Access attributes that are listed. These are required attributes that are added when the user account is created.

Open ID User Identifier A...	Workspace ONE Access ...	
name	firstName	✗ +
nickname	lastName	✗ +
sub	ExternalID	✗ +
email	distinguishedName	✗ +
email	userName	✗ +
email	userPrincipalName	✗ +
email	email	✗ +

Redirect URI Integrate with Open ID Connect Provider using Redirect URI below

<https://aw-emusk.vidmpreview.com/federation/auth/response/oauth2>

[Save](#) [Cancel](#)

2. In the **Auth0 IDP** settings scroll down to **Just-in-Time User Provisioning** and click **Enable**.

- Set your Directory Name to **Auth0-Directory**



- Domains: **Auth0**
- User Attribute Mappings:
- **email - userName**
- **name - firstName**
- **nickname - lastName**
- **email - email**
- **email - userPrincipalName**
- **sub - ExternalID**
- **email - distinguishedName**
- Scroll down to the bottom of the page and click **Save**.

**NOTE:** Some of these values may need to be typed manually

The screenshot shows the 'User Attributes' configuration page. On the left, the 'Users & Roles' menu is visible, with 'Users' highlighted. The main content area displays a list of attributes with checkboxes for 'Required'. The 'email' attribute is checked. Below the list, the 'Raw JSON' tab is selected, showing a JSON object for a user named 'emusk.aw@gmail.com'.

```

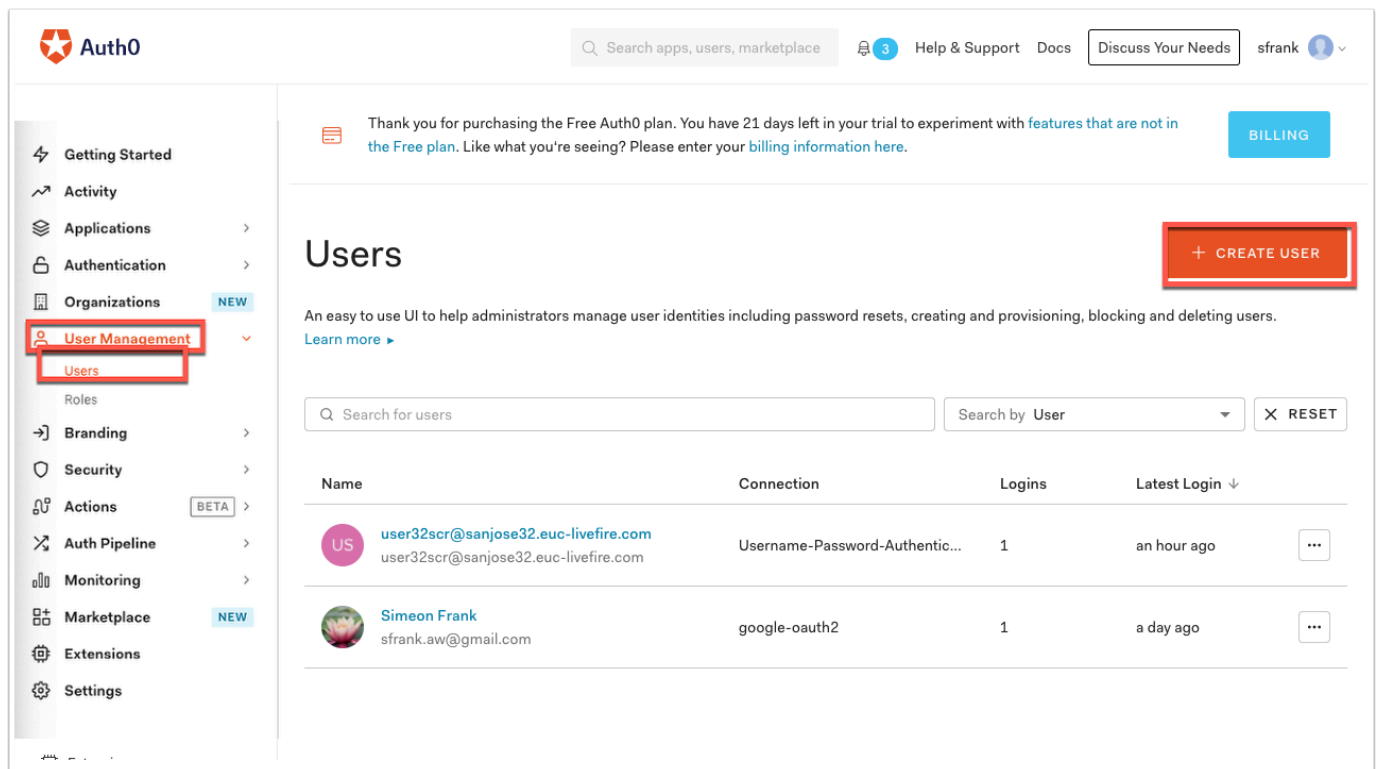
{
  "created_at": "2021-02-17T07:58:28.943Z",
  "email": "emusk.aw@gmail.com",
  "email_verified": true,
  "identities": [
    {
      "connection": "Username-Password-Authentication",
      "provider": "auth0",
      "user_id": "682ccc9dc534b0b7e0832a",
      "isSocial": false
    }
  ],
  "name": "emusk.aw@gmail.com",
  "nickname": "emusk.aw",
  "picture": "https://s.gravatar.com/avatar/15b1e2ec5ad83d6b486fcd1f9589097e~4808r+pg",
  "updated_at": "2021-02-17T08:16:51.486Z",
  "user_id": "auth0|682ccc9dc534b0b7e0832a",
  "last_ip": "88.228.201.86",
  "last_login": "2021-02-17T08:13:38.238Z",
  "login_count": 2,
  "blocked_for": [],
  "guardian_authenticators": []
}

```

**NOTE:** You must ensure here that you have all the attributes that are set to required in Workspace ONE Access mapped to an attribute otherwise the user provisioning will fail.

In order to find out which attributes you can use coming from **Auth0** click on the specific user in **Users** and click on the **Raw JSON** tab.

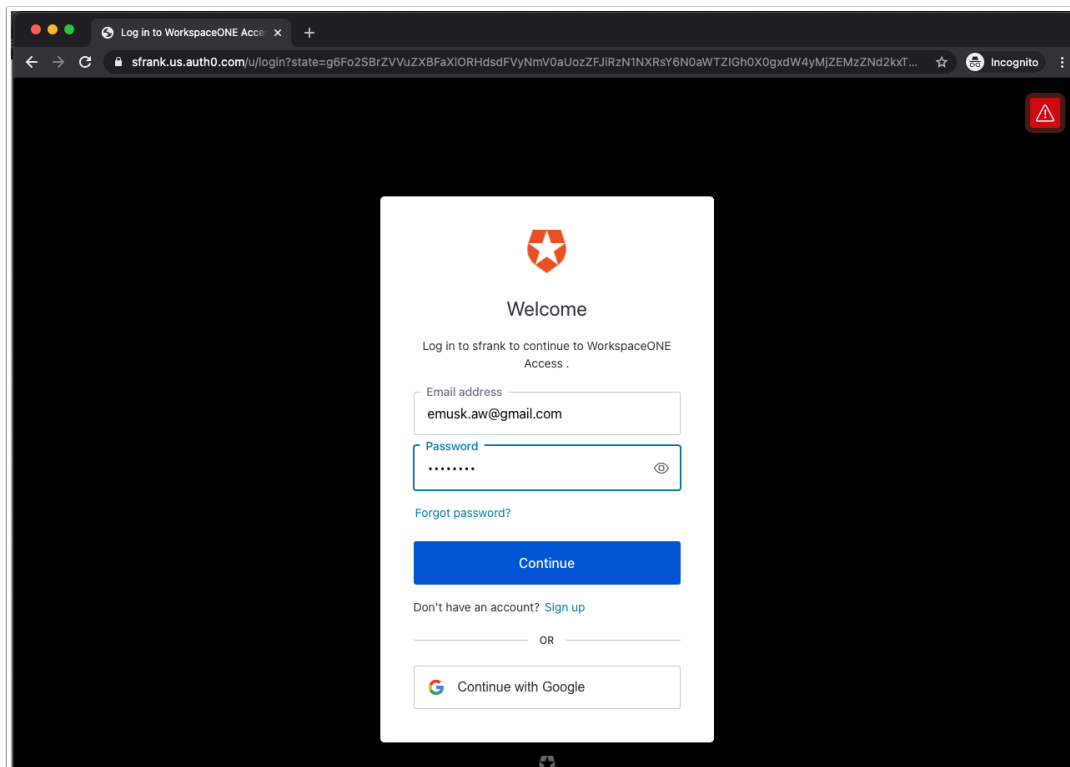
Here you will see the attributes listed.



3. Now flip back to the **Auth0 Admin Console** and click on **User Management > Users** and click **+ Create User**. We will now create a user that is non-existent in the Workspace ONE directory.

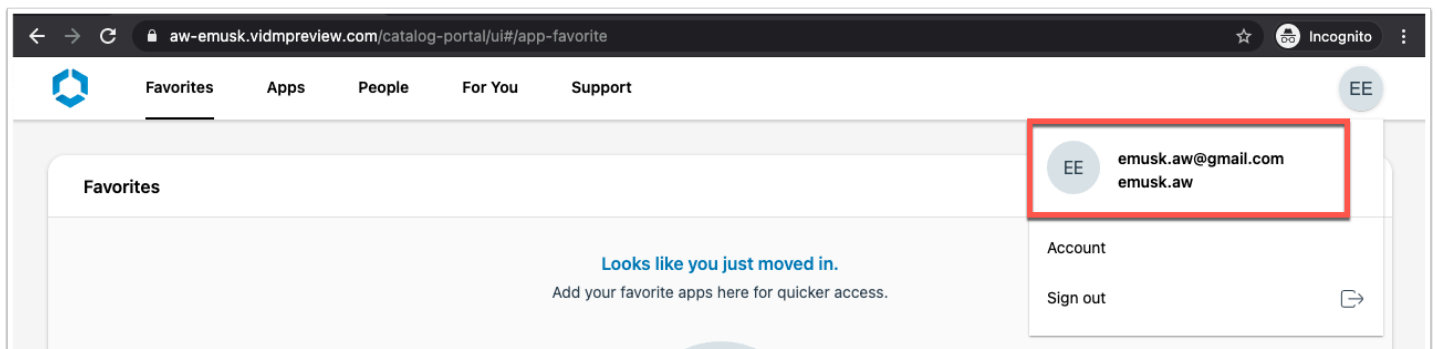
The screenshot shows the 'Create user' form in the Auth0 Admin Console. The form has a title bar 'Create user' with a close button. It contains four main input fields: 'Email \*' with the value 'emusk.aw@gmail.com', 'Password \*' (masked with dots), 'Repeat Password \*' (masked with dots), and 'Connection \*' (a dropdown menu set to 'Username-Password-Authentication'). At the bottom, there are two buttons: 'CREATE' and 'CANCEL'.

4. Create a random user that is not in the Workspace ONE Access directory. Use the password **VMware1!** Click on **CREATE**



5. Open a **new incognito window**. (If a previous one was open close and re-open)

Browse to the **Workspace ONE Access URL** and you will be redirect to your **Auth0** URL. Authenticate using the **email** created for the user you created above and **VMware1!** click **Continue**



6. You should now be authenticated using that **unique user**.

**NOTE:** If you are seeing errors after attempting to authenticate go back to **STEP 2** in order to look at the attributes that may not be lining up correctly.

Workspace ONE™ Access Tenant Admin - AW-EMUSK

Navigation: Dashboard, Users & Groups, Catalog, **Identity & Access Management**, Roles

Sub-navigation: **Directories**, Identity Providers, Password Recovery Assistant, Authentication Methods, Policies, Manage, Setup

Directories (4) [Add Directory](#)

Directo...	Type	Domains	Synced Groups	Synced Users	Last S...	Alerts
<a href="#">System Directory</a>	Local Directory	1	0	1		
<a href="#">LivefireSync</a>	Active Directory over LDAP	1	60	12	Feb 15, 2021 10:55:16 AM	8 <a href="#">Sync Now</a>
<a href="#">UEMLocalDirectory_sfmsuk</a>	Other Directory	1	0	0		
<b><a href="#">Auth0-Directory</a></b>	Just-in-Time Directory	1	0	1		

7. Switch back to the Workspace ONE Access admin console and click on **Identity & Access Management** > **Directories** and notice you now have **Auth0-Directory** and the type is **Just-in-Time Directory**.

Workspace ONE™ Access

Navigation: Dashboard, **Users & Groups**, Catalog, Identity & Access Management, Roles

Sub-navigation: **Users**, Groups

Users (11)

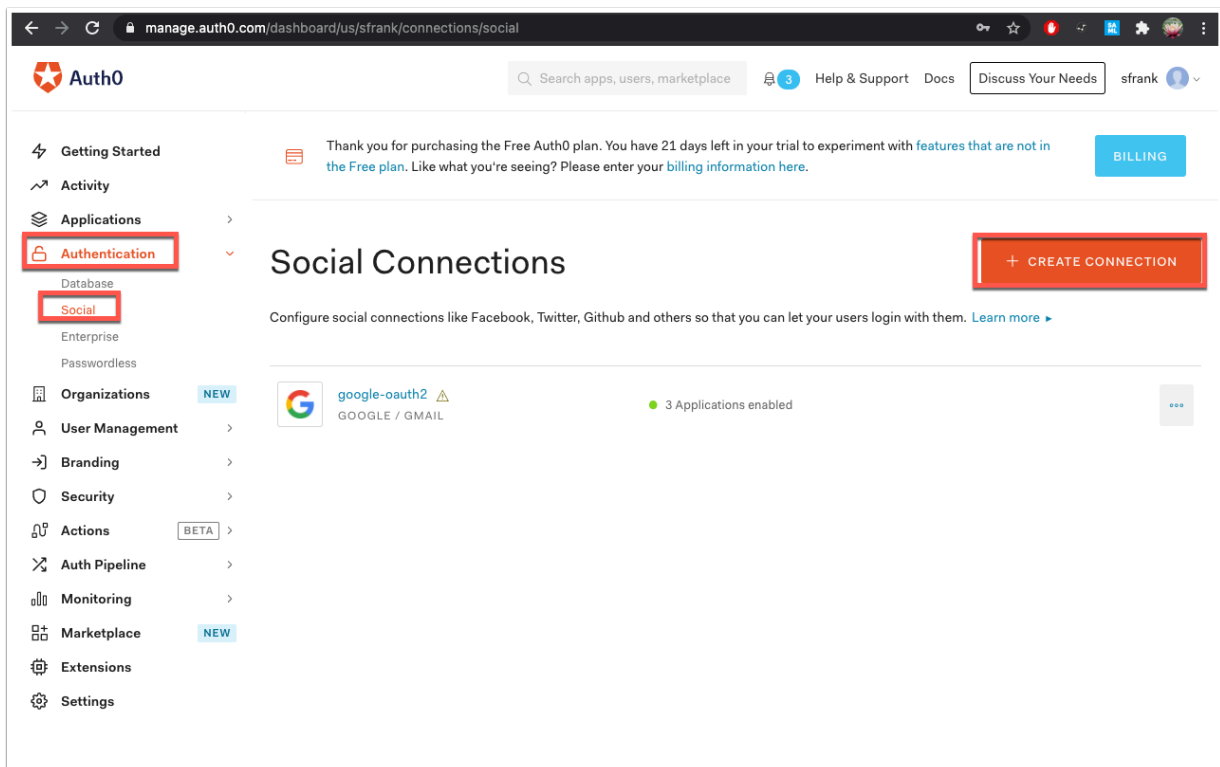
User Name	User ID	Domain	Directory	VMware Verify P...	Groups
<a href="#">Admin, Tenant</a>	manfred	System Domain	System Directory	N/A	ALL USERS
<a href="#">Debio, Mark</a>	Mark	euc-livfire.com	LivefireSync	N/A	ALL USERS, M
<a href="#">Dusello, Fernando</a>	Fernando	euc-livfire.com	LivefireSync	N/A	ALL USERS, M
<a href="#">Ikin, Kevin</a>	Kevin	euc-livfire.com	LivefireSync	N/A	ALL USERS, Dc
<b><a href="#">manfred, manf...</a></b>	<b>manfred@euc-liv...</b>	<b>Auth0</b>	<b>Auth0-Directory</b>	N/A	ALL USERS
<a href="#">Marios, Tom</a>	Tom	euc-livfire.com	LivefireSync	N/A	ALL USERS, M
<a href="#">mark, mark@e...</a>	mark@euc-livfir...	Auth0	Auth0-Directory	N/A	ALL USERS
<a href="#">Markez, Kim</a>	kim	euc-livfire.com	LivefireSync	N/A	ALL USERS, Dc

8. At the top navigation click on **Users & Groups** and click **Users**.

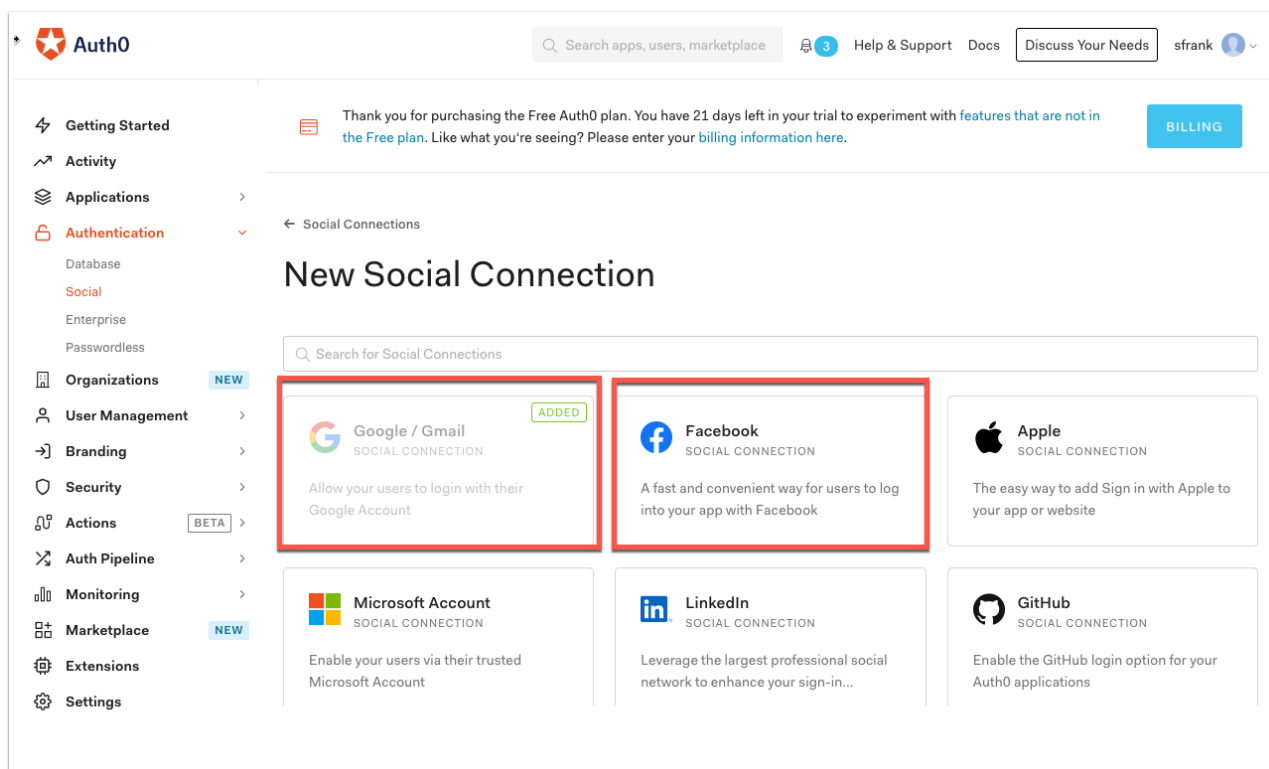
You should now be able to see the user that has been created as a result of JIT.

## Bonus Material 2: Auth0 and Social Integrations

One of the benefits of using Auth0 is that it allows for integrations with Social platforms such as Google & Facebook. In this exercise you will create the Google and & Facebook connection with Auth0 and authenticate to Workspace ONE Access using one of these social connections as your source of identity.

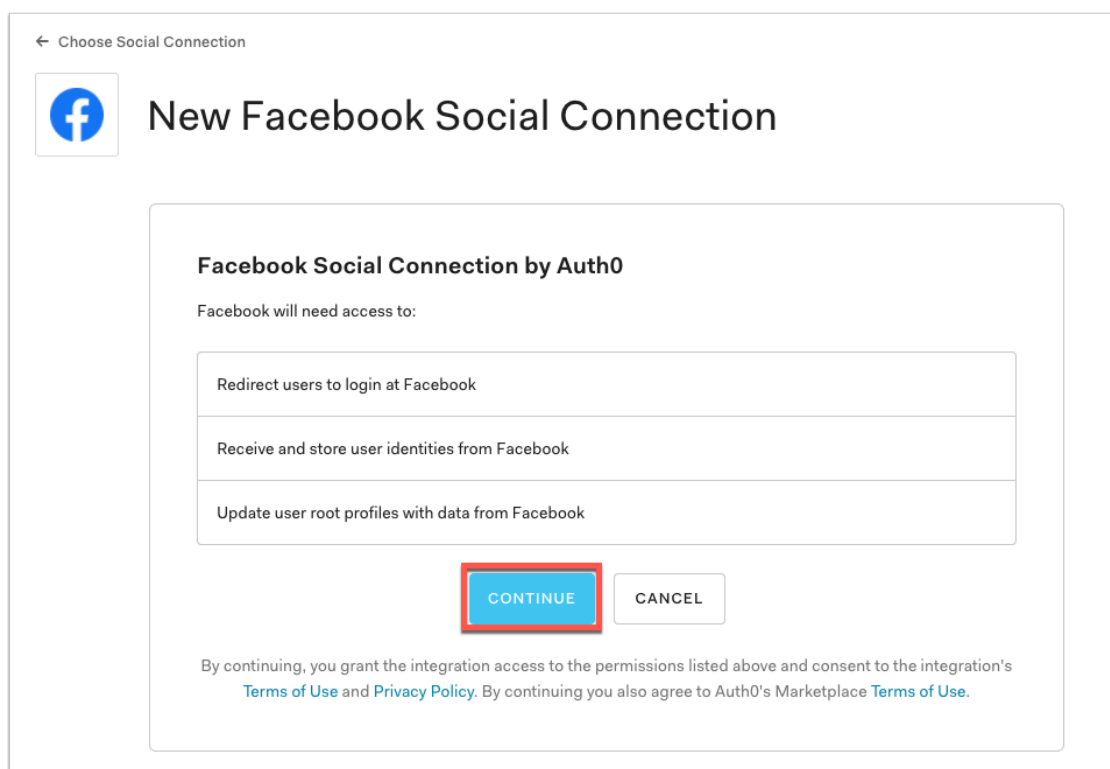


1. In the **Auth0 admin console** navigate to **Authentication** > **Social** and click **CREATE CONNECTION**.



2. If its not already added add **Google / Gmail** and **Facebook**. (In my case Google has already been added automatically)

- Notice the other available connections on this page - Dropbox, Paypal, Microsoft etc...



3. Proceed with adding **Facebook** as a social connection and click **Continue** to grant access to the information listed above.

Choose Social Connection

### New Facebook Social Connection

[SETUP GUIDE](#)

**Name** facebook

If you are triggering a login manually, this is the identifier you would use on the connection parameter.

**App ID** Leave blank to use Auth0 dev keys  
[How to obtain a App ID?](#)

**App Secret** Leave blank to use Auth0 dev keys  
For security purposes, we don't show your existing App Secret.

**User Data**

- ☒ **Public Profile** ⓘ  
public\_profile
- ☒ **Email** ⓘ  
email
- ☐ **Group Access Member Info** ⓘ  
groups\_access\_member\_info
- ☐ **Publish to Groups** ⓘ  
publish\_to\_groups
- ☐ **Age Range** ⓘ  
user\_age\_range
- ☐ **Birthday** ⓘ  
user\_birthday
- ☐ **Events** ⓘ  
user\_events
- ☐ **Friends** ⓘ  
user\_friends
- ☐ **Gender** ⓘ  
user\_gender
- ☐ **Hometown** ⓘ  
user\_hometown

**Deprecated Permissions**

- ☐ **manage\_notifications** ⓘ  
manage\_notifications
- ☐ **publish\_actions** ⓘ  
publish\_actions
- ☐ **read\_stream** ⓘ  
read\_stream
- ☐ **read\_mailbox** ⓘ  
read\_mailbox
- ☐ **user\_groups** ⓘ  
user\_groups
- ☐ **user\_managed\_groups** ⓘ  
user\_managed\_groups
- ☐ **user\_status** ⓘ  
user\_status

These permissions are only available for applications using Graph API version v2.3 or older.

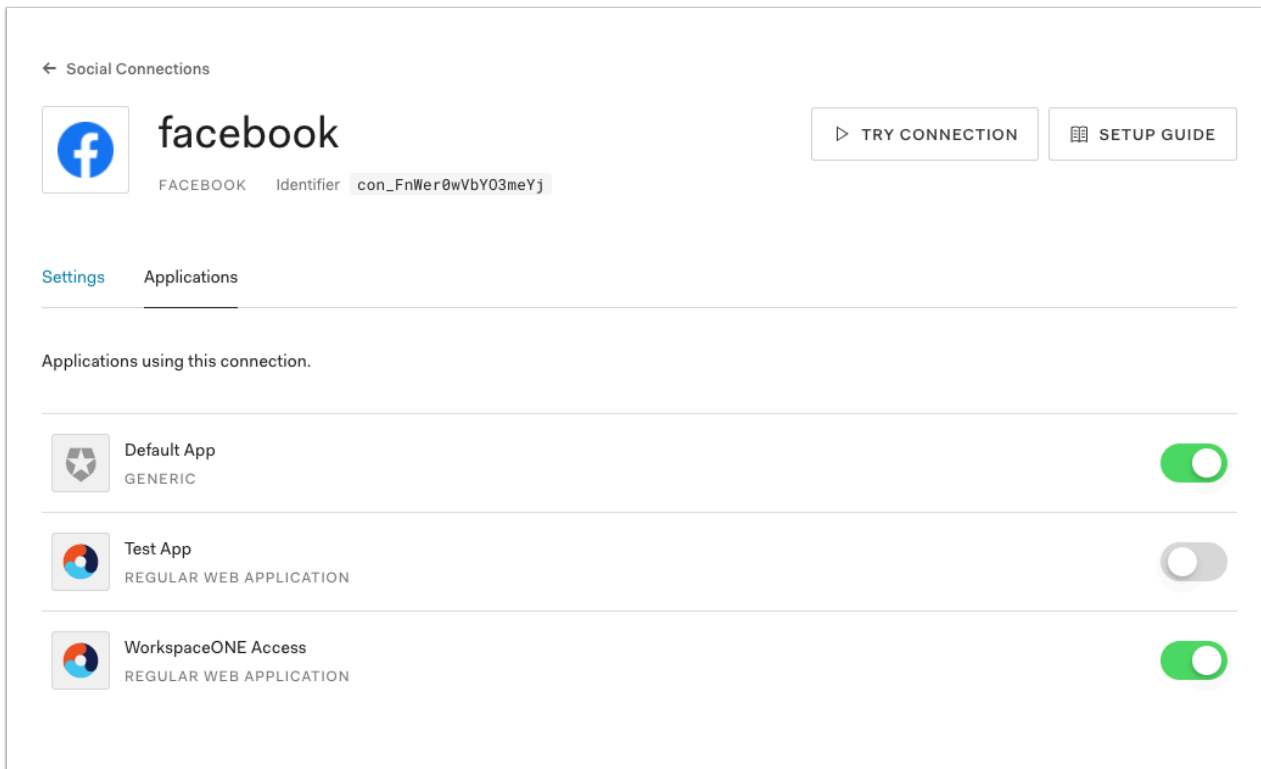
**Advanced**

Sync user profile attributes at each login ☒ **ENABLED**

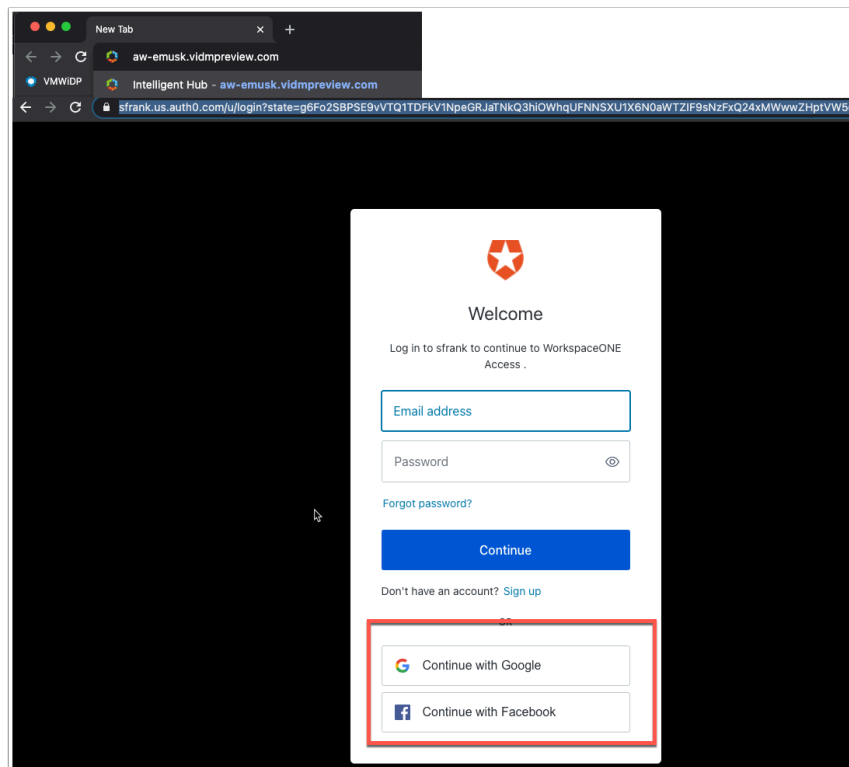
**CREATE**

4. You will be presented with options for linking Facebook. This could either be a private Facebook for enterprise or you can use the public Facebook accounts. For our example we will use the public Facebook Accounts.

- In the User Data section you can select which attributes you would like to sync across to Auth0. Enable **Email**.
- At the bottom of the page click **CREATE**.



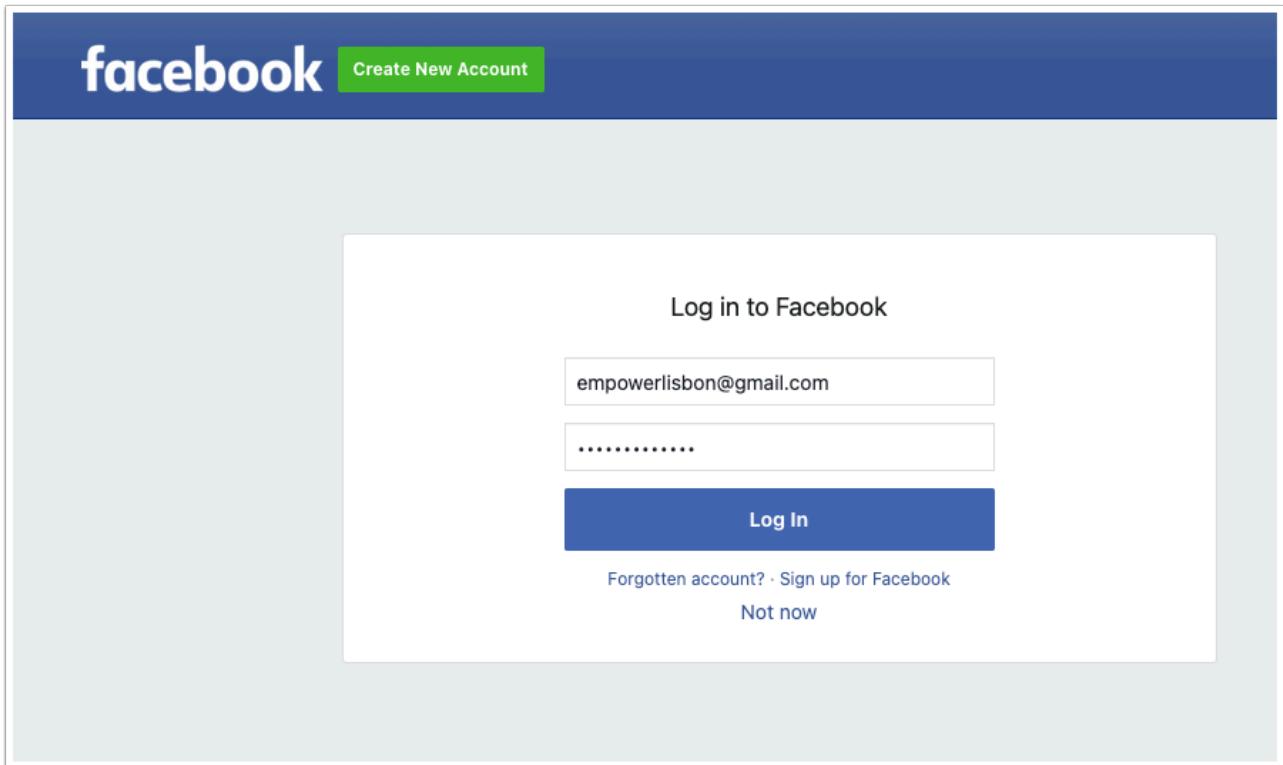
5. Now give the **Default App** and **WorkspaceONE Access** applications the permission to use this connection.



6. Now open an incognito window. (Make sure previous incognito windows have been closed). Browse to your **Workspace ONE Access tenant URL**.

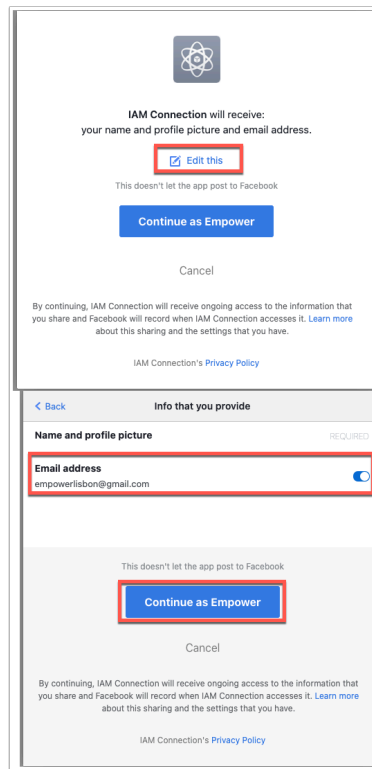


- You will be redirected to your unique instance of **Auth0**.
- Now you will see **Continue with Google** and **Continue with Facebook** as available options here on the authentication page.
- Click **Continue with Facebook**



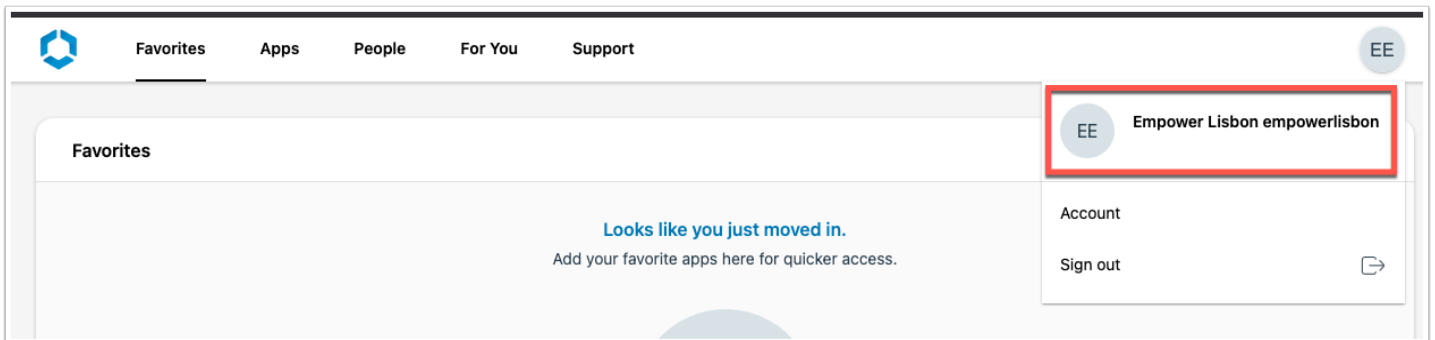
7. Log into **Facebook** (preferably a test facebook account)

- Click **Log In**

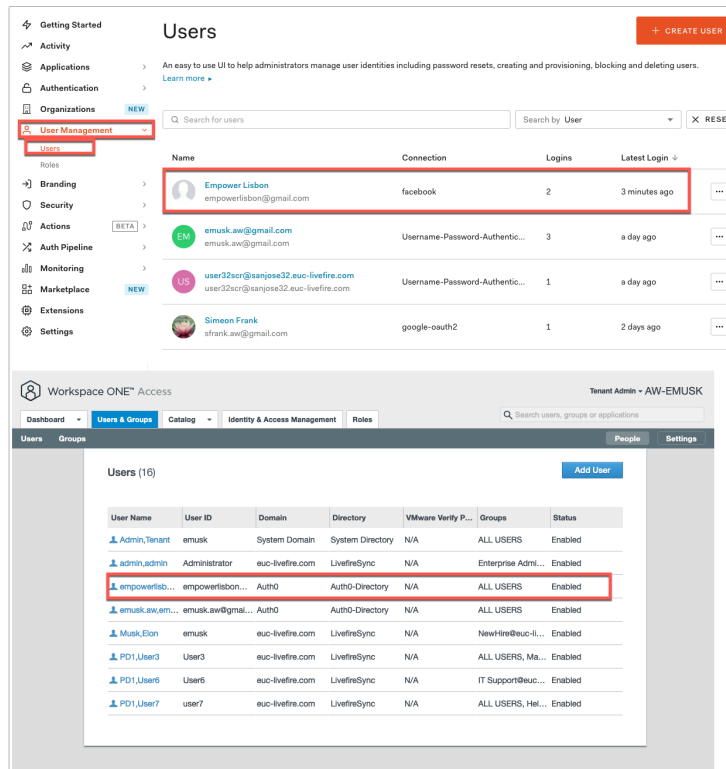


8. You will now see the permission window. **Click on Edit** this and you will see that **name and profile picture** and **e-mail address** are the only attributes being sent as configured above.

Click **Continue as Empower**.

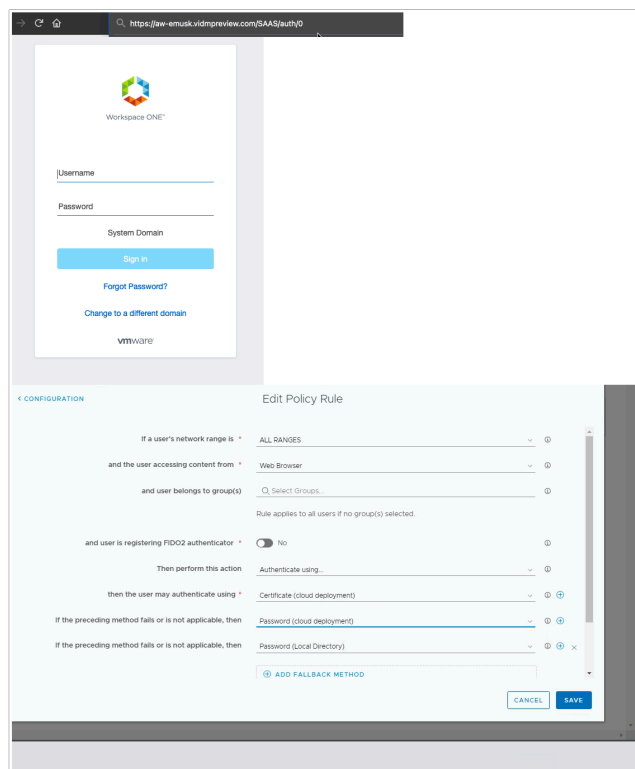


9. You should now be authenticated to the Workspace ONE Hub with the facebook user. At authentication the user get created in Auth0 then through JIT gets created in Workspace ONE Access.



10. You will note in the Auth0 Admin Console in the **User Management > Users** that the new **user** from Facebook has been created.

- Additionally on the Workspace ONE Access side you will see the user has been created.



11. **VERY IMPORTANT** In order to ensure you can log back in as local administrator into Workspace ONE Access you will have to append your Access URL with [/SAAS/auth/0](#) you will be able to log on and change the **default access policy** from Auth0 to Certificate (**cloud deployment**) with a fallback to **Password (cloud deployment)** with a fallback to **Password (Local Directory)**.

In this lab you saw the power of leveraging existing identity databases (both corporate and personal) in order to provide central authentication and single-sign-on to various corporate applications using Workspace ONE Access.

This concludes the Bonus Material for the Auth0 integration as Third-party OIDC Identity Provider.

Author: Simeon Frank