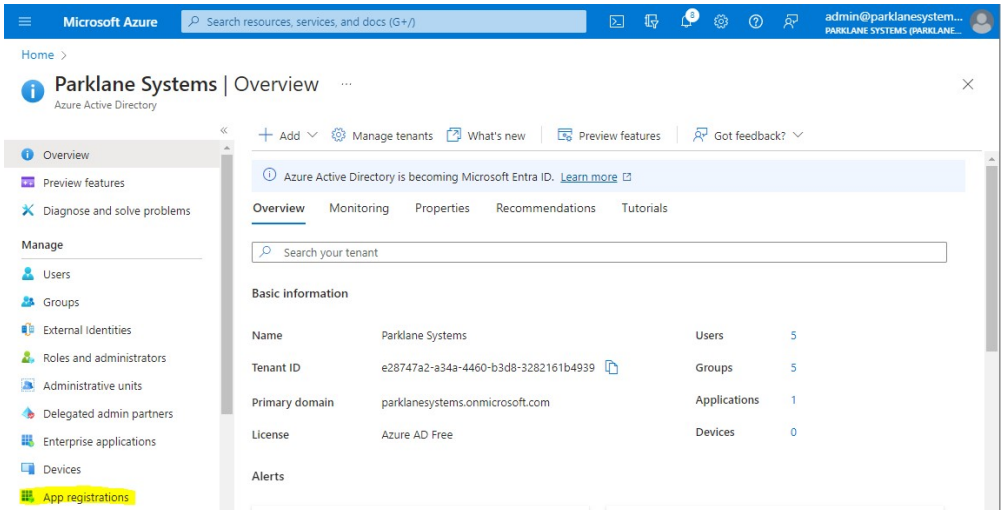
Registering KICS with Microsoft Entra ID (Azure AD)

In KICS 3.8.0, direct support was added for authentication using Microsoft Entra ID / Azure AD.

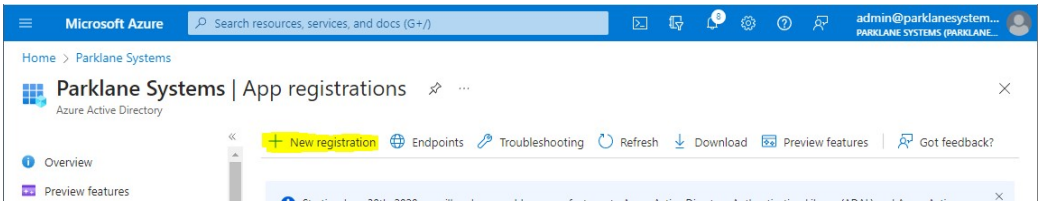
KICS utilizes SAML to authenticate with Microsoft Entra ID. To set up authentication, provisioning will need to be performed in both KICS and the Azure AD Portal.

Setting up Azure AD for KICS

Open the **Azure AD Portal** and select **App Registrations** on the left hand side



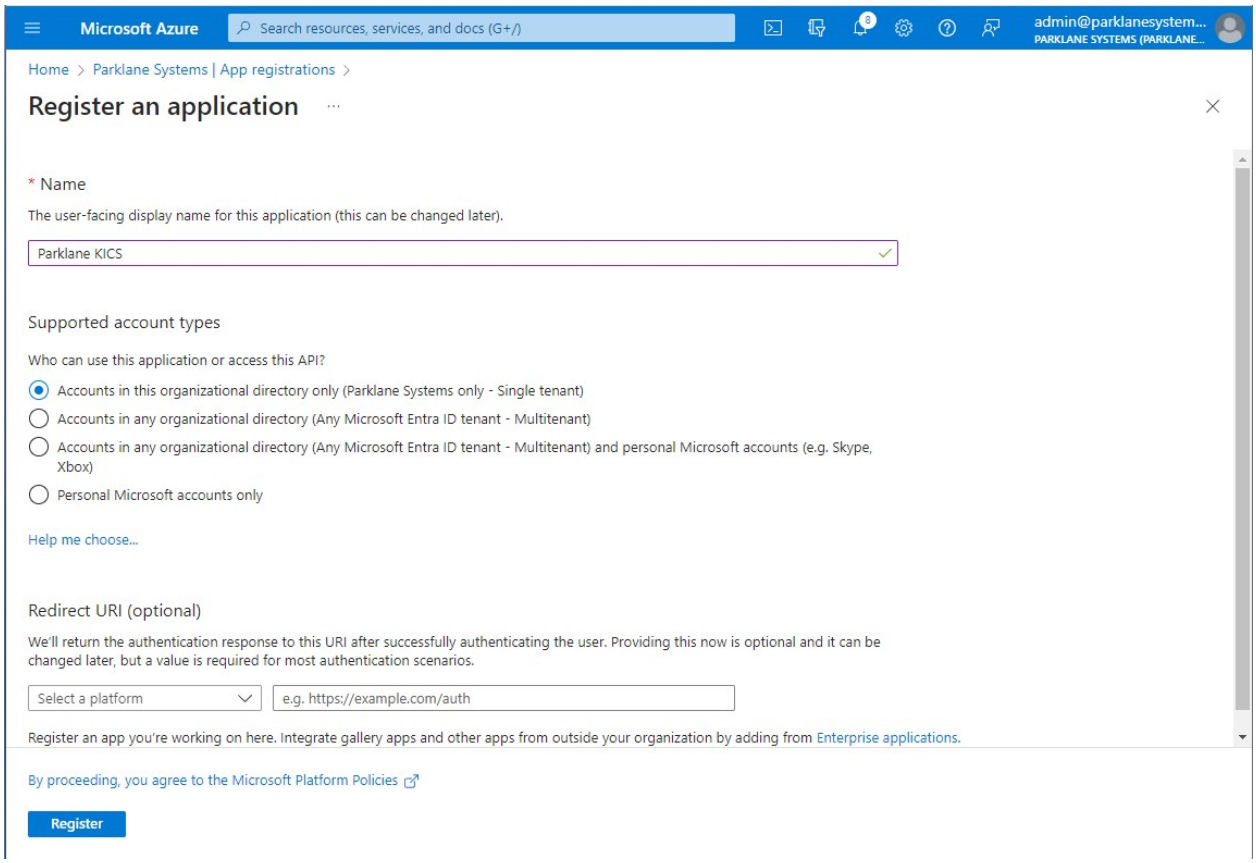
Select **New Registration**



Specify a name for the application such as **Parklane KICS**.

We will not configure the **Redirect URI** at this point.

Click **Register**



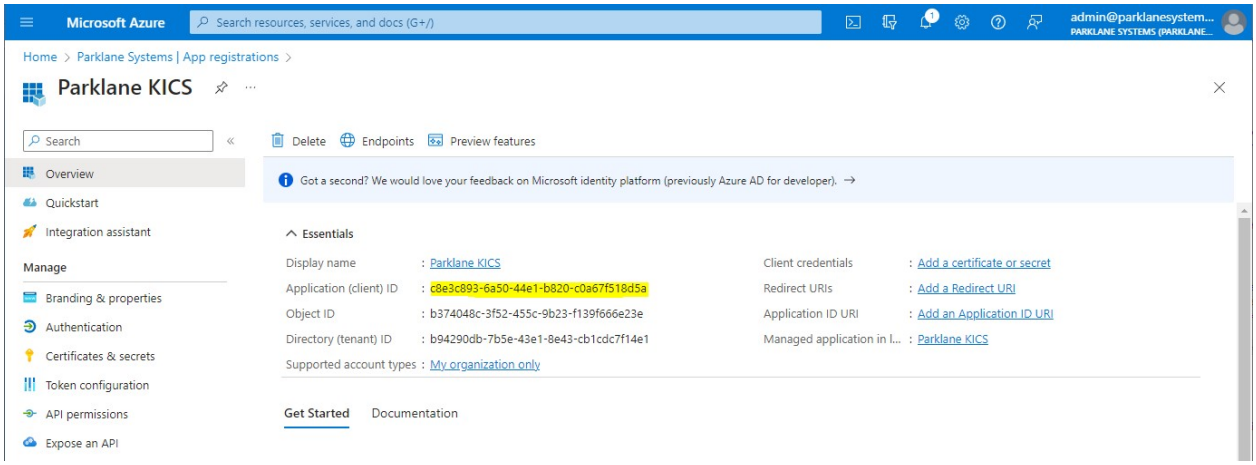
The **Parklane KICS** App Registration will now appear on the Azure AD Console

We will need the two identifiers from the Azure AD console to prepare KICS for authentication:

- **The Application (client) ID**
- **The Federation Metadata Document Endpoint URL**

To Obtain the Application ID:

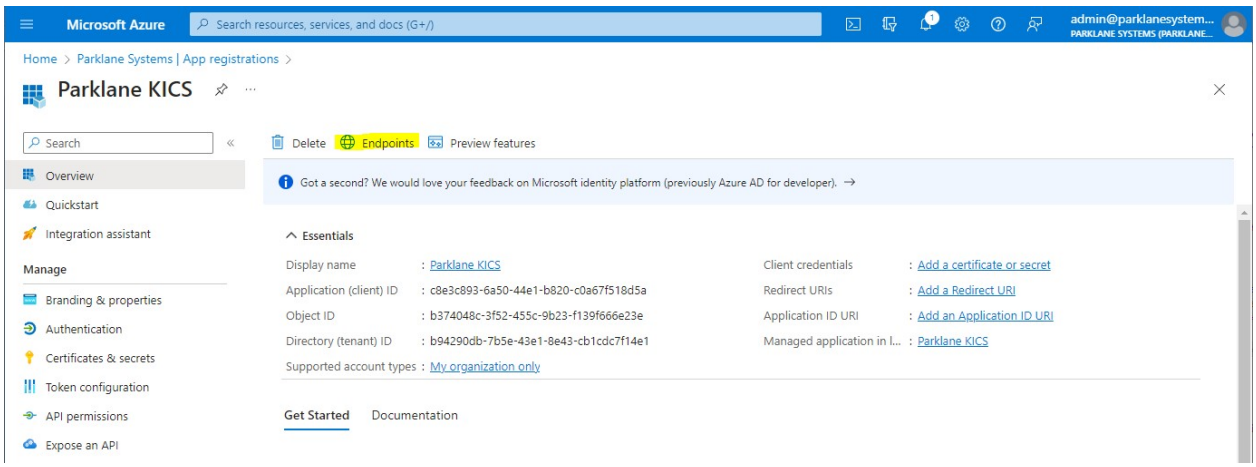
The **Application ID** is listed on the KICS Application Registration page under **Essentials**.



Make note of the Application ID. You will require this during Step 1 of setting up KICS.

To Obtain the Federation Metadata URL:

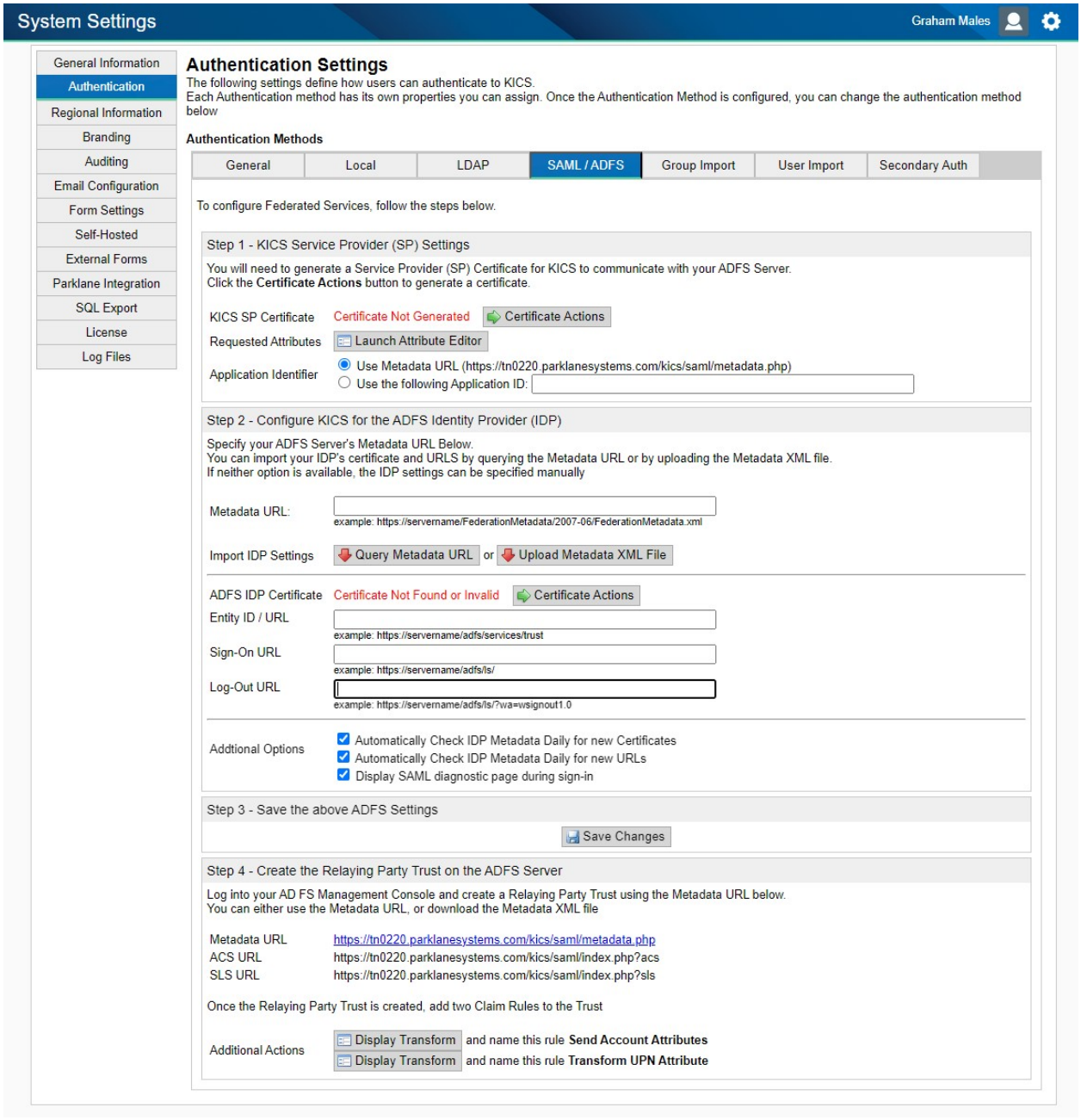
The **Federation Metadata Document** URL is listed on the KICS Application Registration page under **Endpoints**.



Make note of the Federation Metadata Document URL. You will require this during Step 2 of setting up KICS.

KICS Step 1 - Preparing KICS for Azure AD

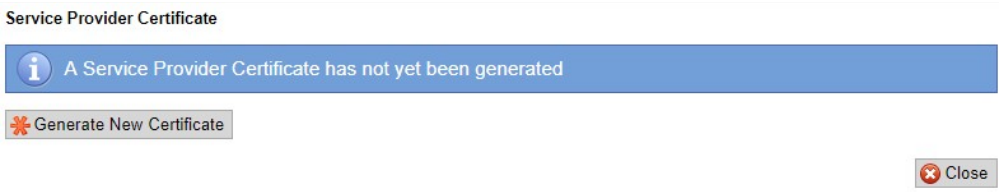
In KICS, go to **System Settings > Authentication > SAML / ADFS**



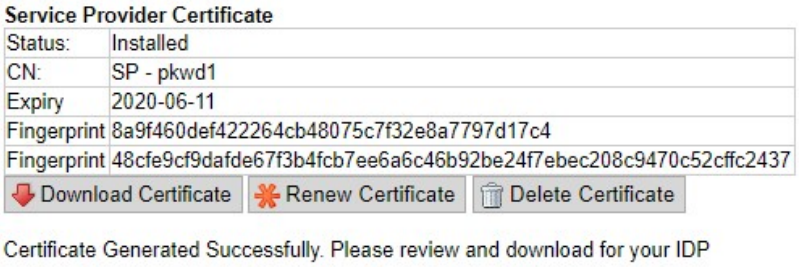
KICS Step 1.1 - Generate SP Certificate

Under **Step 1**, click **Certificate Actions** beside the **KICS SP Certificate** option

A Dialog for the Service Provider Certificate will appear



Click **Generate New Certificate**. Certificate Generation will take a couple seconds.



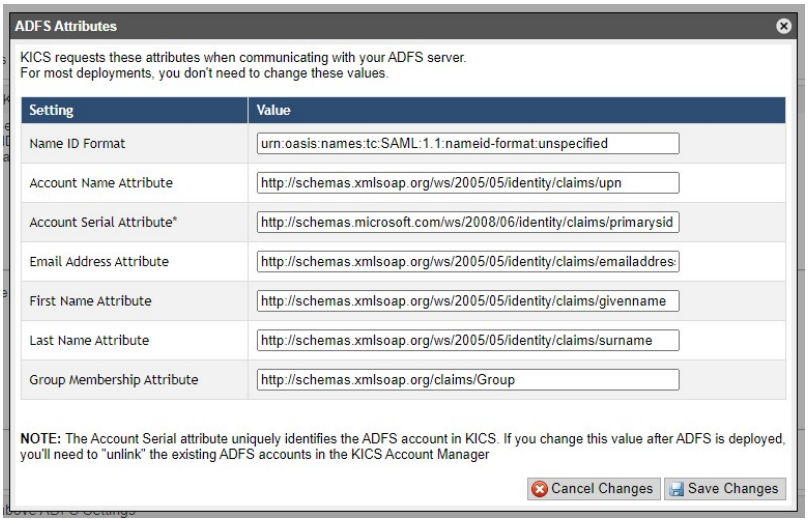
Click **Close**

KICS Step 1.2 - Setting up the Federated Services Attributes

You will need to set up KICS to use the Attributes that Azure provides during the sign-in process.

Launch the Attribute Editor

The Attribute Editor will display



KICS comes preconfigured for on-premises attributes. Please update the attributes listed below in red:

Attribute	Value
Name ID Format	urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified
Account Name	http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress
Account Serial	http://schemas.microsoft.com/identity/claims/objectidentifier
Email Address	http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress
First Name	http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname
Last Name	http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname
Group Membership	http://schemas.microsoft.com/ws/2008/06/identity/claims/groups

Click **Save Changes**

KICS Step 1.3 - Setting the Azure Application ID

Continuing down the page, you will see the configuration parameter for the KICS application Identifier

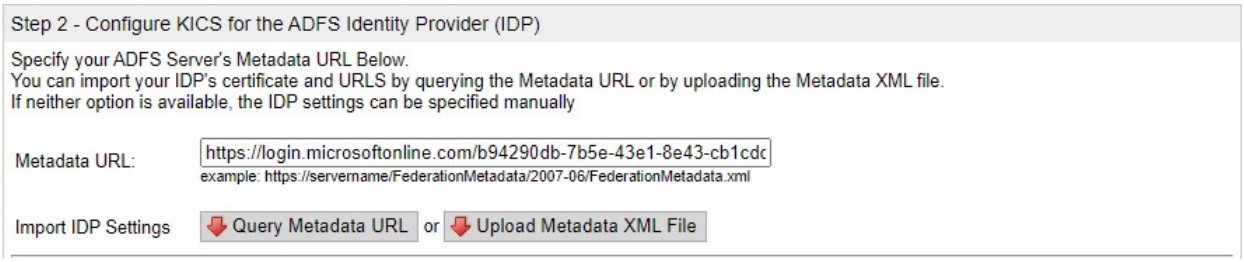
Select **Use the following Application ID**

Paste in the **Application ID** you obtained from the Azure Console

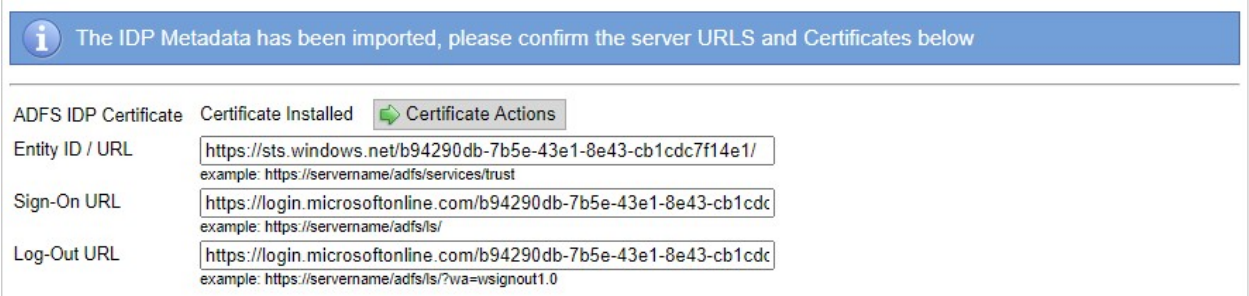
KICS Step 2 - Configuring Azure as the identity provider

On the **System Settings > Authentication > SAML** page, **Step 2** focuses on setting up the URLS and Certificates for your Azure Environment. KICS can import this data from the Federation Metadata URL copied from your KICS App registration page above.

Paste the Federation Metadata Document Endpoint URL you obtained from the Azure Console into the **Metadata URL** field and then click **Query Metadata URL**. The Azure URLs and Certificates will be imported.



You will see the Azure IDP Certificate and Endpoint URLS configured in KICS.



Most SAML server certificates will last for 1 year. You have the option to configure KICS to query the Metadata once a day to check for certificate updates and URLS.

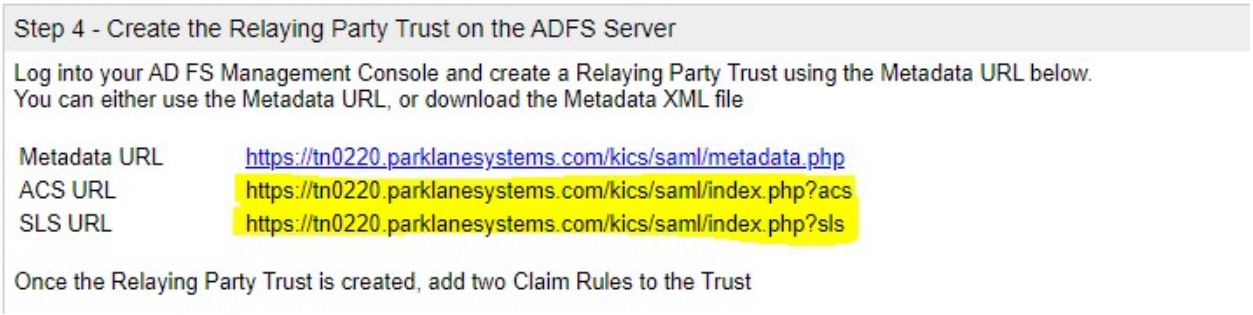
The **Display SAML Diagnostic Page during sign-in** option allows you to review the SAML attributes returned from Azure and make sure they line up during the authentication process. It's highly recommended to enable this feature during the initial testing of the authentication component.

Click **Save** to save the current settings into KICS

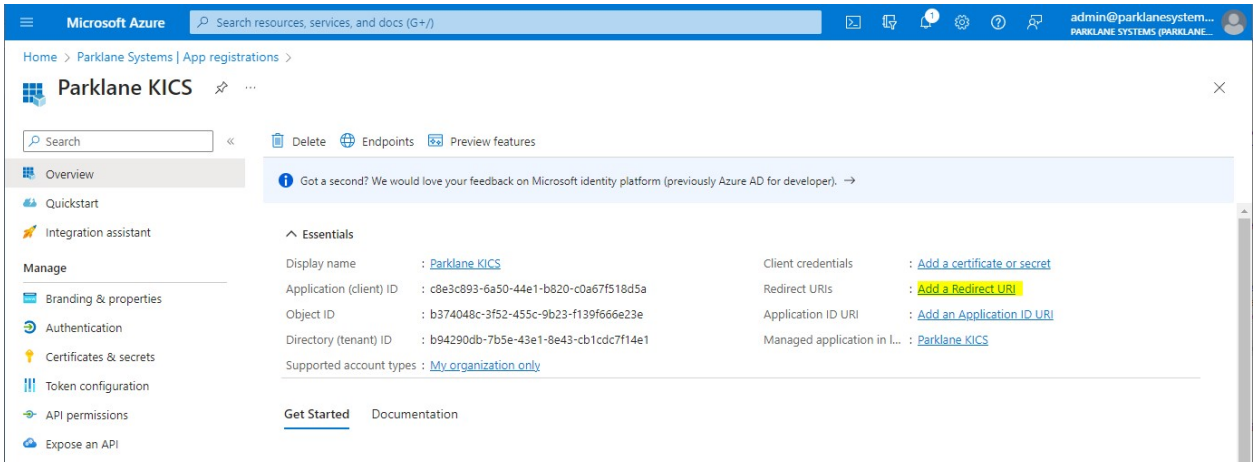
Setting up URLS and Attributes for KICS within Azure

KICS can now send authentication requests to Azure, however we need to configure Azure to provide the proper attributes and send clients back to the proper URL.

On the KICS **System Settings > Authentication > SAML page**, make note of the **ACS** and **SLS** URLS.

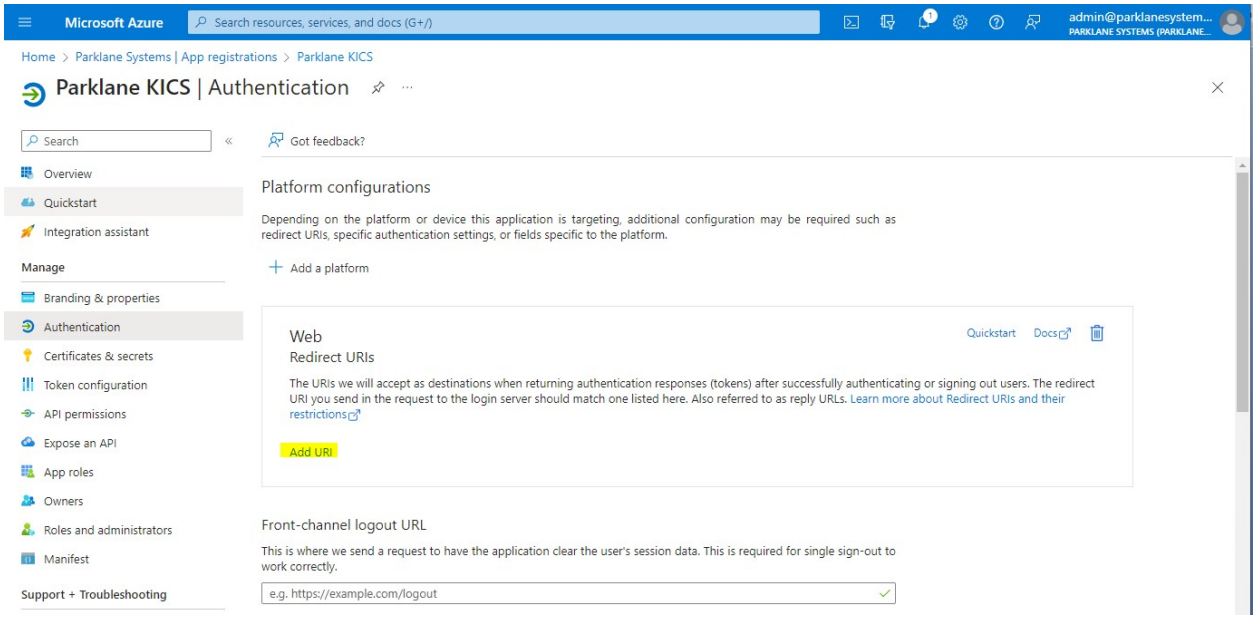


Open the **Azure AD Portal** > **App Registrations > Parklane KICS** configuration page.



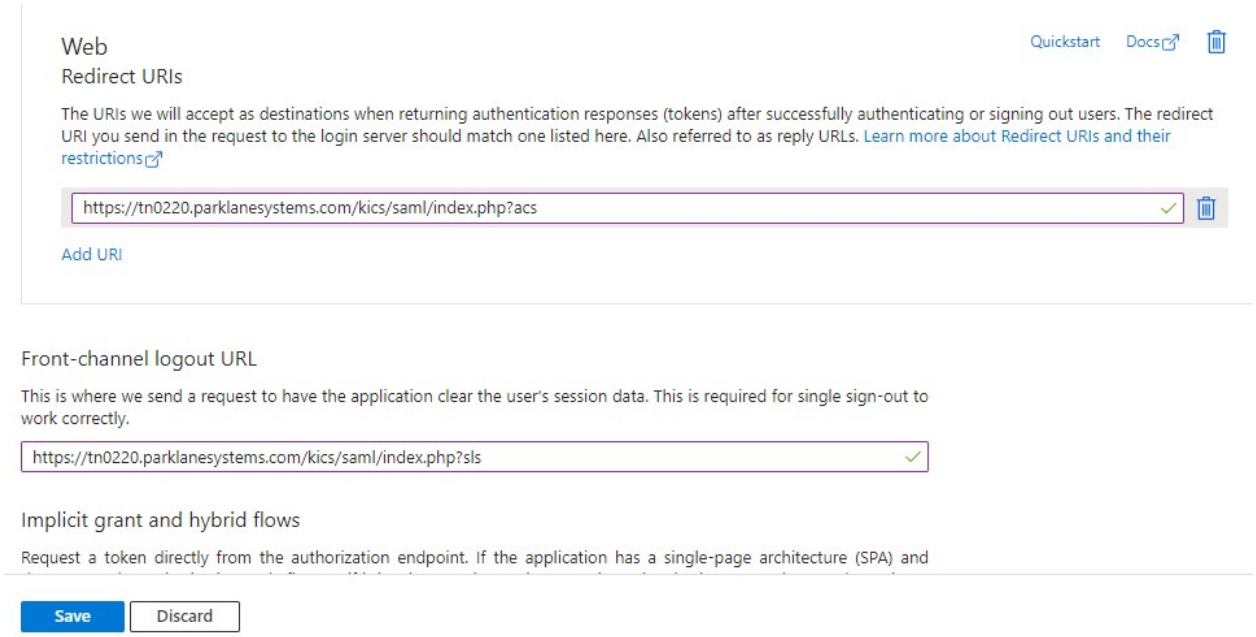
Select **Add a Redirect URI**

The **Authentication** page will open.



Under **Web** - **Redirect URIs**, select **add URI** and paste in the ACS URL obtained from KICS.

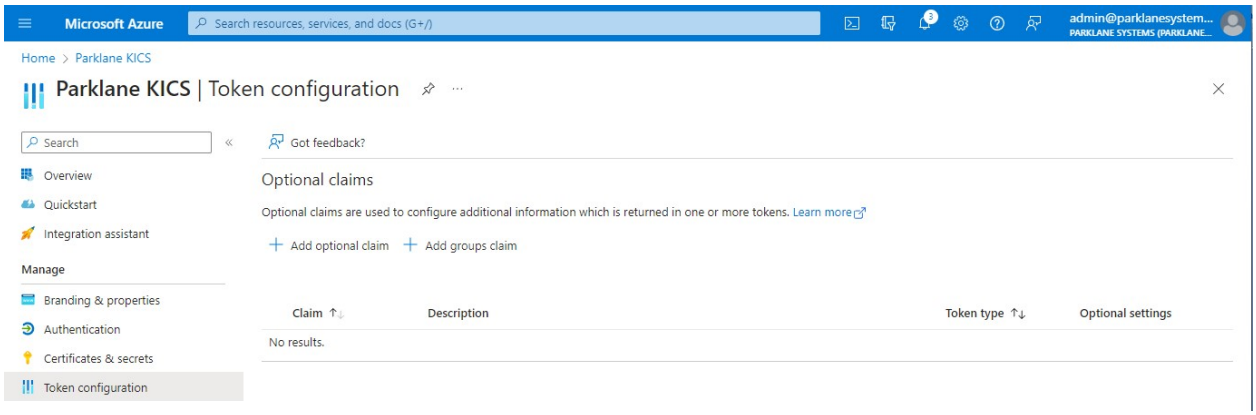
Under **Front-channel logout URL**, paste the SLS URL obtained from KICS.



Click **Save**

Next, we will set up the Attribute Tokens for KICS. KICS accepts the default tokens that Azure AD provides, however if you wish to pass Group Membership, you will need to configure the Group Token Attribute.

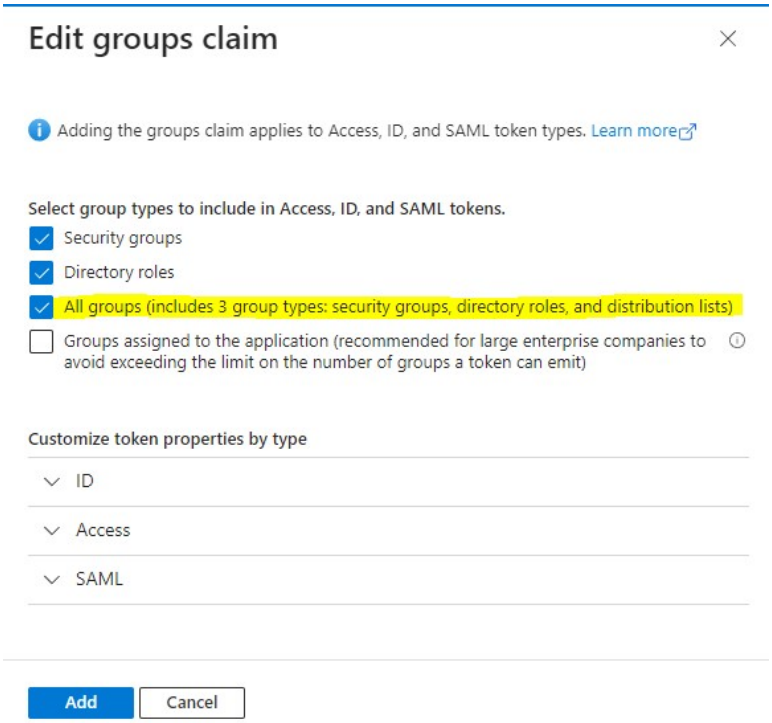
On the left side of the Azure Portal, select **Token Configuration**



Select **Add Groups Claim**

The **Edit Groups Claim** dialog will appear on the right side of the screen.

Choose which groups would like to pass into KICS. If you're unsure, select **All Groups**

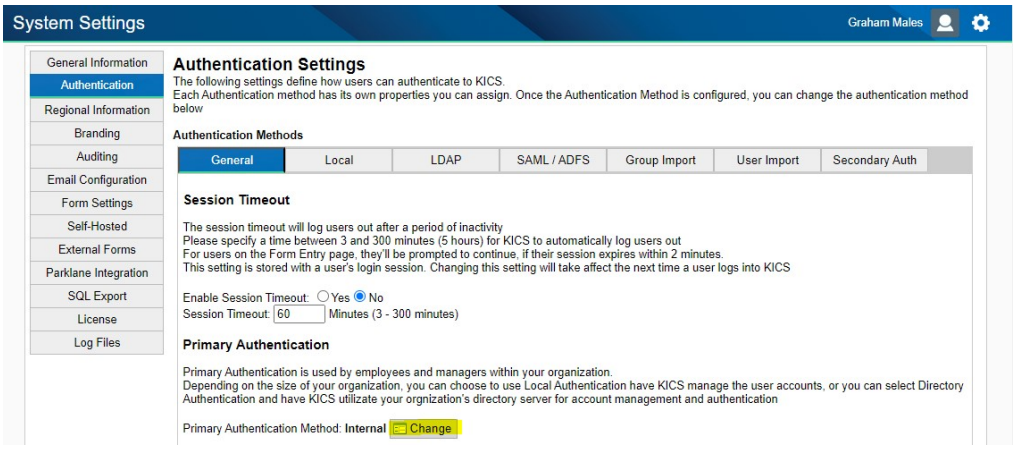


Click **Add**

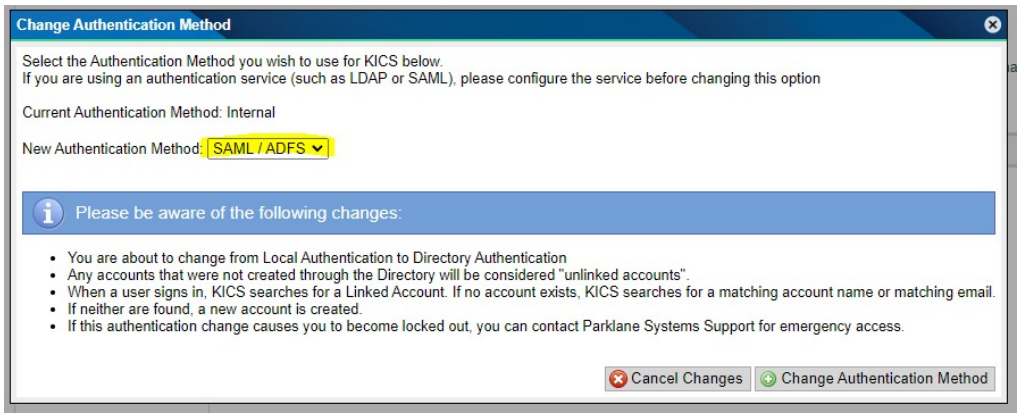
Configure KICS to Authenticate with Azure

At this point, the provisioning of SAML in KICS and Azure is complete. To configure KICS to send login requests to Azure, we need to change the primary login method in KICS.

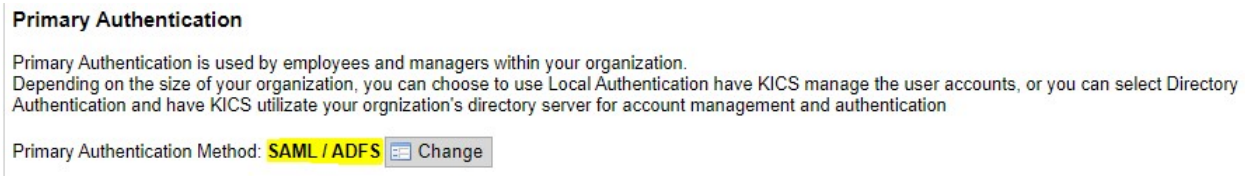
In KICS, go to the **System Settings - Authentication - General** page



Under the **Primary Authentication** section, click **Change**

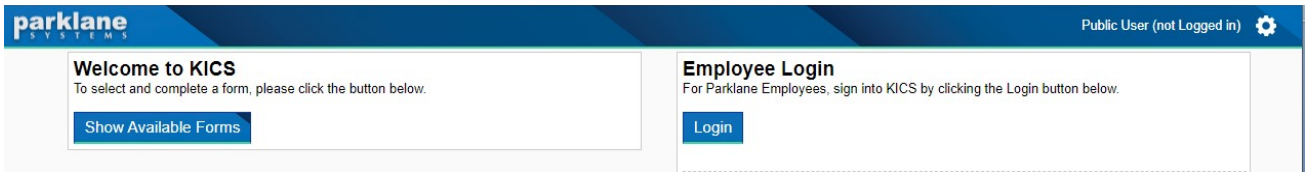


Select **SAML/ADFS** and then click **Change Authentication Method**



The primary authentication method will now be listed as **SAML/ADFS**.

The login page will now show a login button to redirect users to the Azure AD login page.

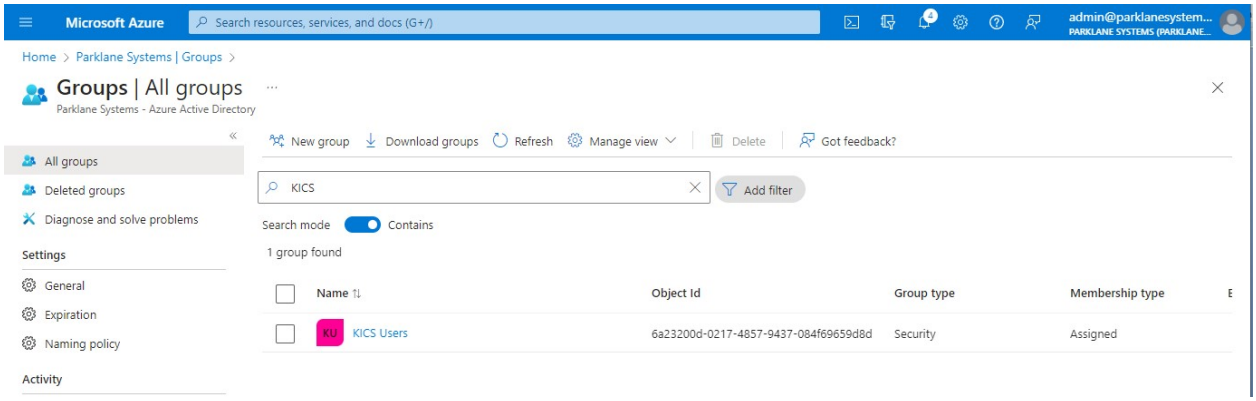


Importing Groups into KICS from Azure

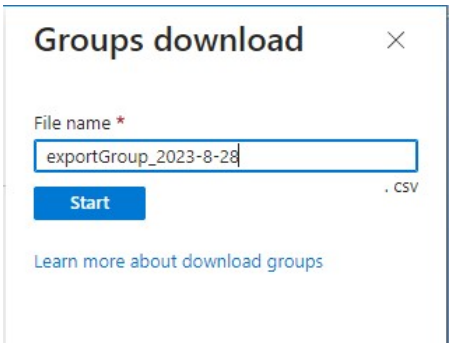
In KICS, we can import a list of groups that are exported from the Azure AD console. These groups can be assigned role permissions in KICS.

When users authenticate, they can be automatically granted permissions to templates and functions within KICS if they are a member of the appropriate group.

From the **Azure AD Console**, open the **Groups** Panel



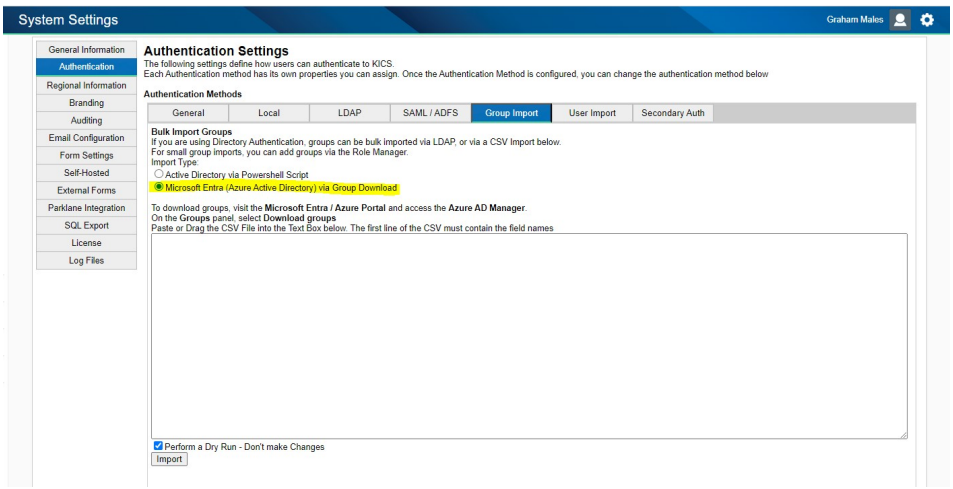
Select **Download Groups**.



Click **Start**. Azure will generate an export of Azure AD groups as a CSV file.

In KICS, go to the **System Settings - Authentication - Group Import** page

Select **Microsoft Entra (Azure Active Directory) via Group Download**



Paste the contents of the CSV file into the text box.

You have the option to perform a **dry run** of the import to ensure the import will be successful without making changes.

Click **Import**

You will see a status result of the import

Line Number	Import Success	Group Name	Import Details
1	Yes	KICS Users	Group not found, creating

If this was a dry run and everything looks successful, uncheck the **dry run** option and re-run the import to add the groups to KICS.

The Group Import is now successful. If you access the Role Manager, you can now search for and see the groups you imported.