LastPass •••|

Table of Contents

Set Up Federated Login for LastPass Using Azure Active Directory

This guide provides setup instructions for using LastPass with Azure Active Directory (Azure AD) for your LastPass Enterprise or LastPass Identity account.

Summary

LastPass supports the following provisioning features:

- Create Users
- Update User Attributes
- Sync User Groups
- Deactivate or Disable Users

Federated login for LastPass Enterprise and LastPass Identity accounts allows users to log in to LastPass using their Azure AD account (instead of a username and separate Master Password) to access their LastPass Vault.

System Requirements

The enable federated login for LastPass using Azure AD, the following is required:

- An active Premium subscription to Microsoft Azure AD
- An active trial or paid LastPass Enterprise or LastPass Identity account
- An active LastPass Enterprise or LastPass Identity admin (required when activating your trial or paid account)

The LastPass Azure AD SCIM endpoint for federated login does not require any software installation.

Before you begin...

- It is **required** that you [enable the "Permit super admins to reset Master Passwords" policy](#) for at least 1 LastPass admin (who is also a non-federated admin) in the LastPass Admin Console. This ensures that all LastPass user accounts can still be recovered (via Master Password reset) if a critical setting is misconfigured or changed for federated login after setup is complete.
- It is helpful to open a text editor application so that you can copy and paste values that will be used between your LastPass Admin Console and the Azure AD Admin portal.

Step #1: Generate a Provisioning Token

1. Access the LastPass Admin Console by opening a web browser and navigating to either of the following:
    - For accounts using US data centers:
      https://lastpass.com/company/#!/dashboard
    - For accounts using EU data centers:
      https://lastpass.eu/company/#!/dashboard
2. Enter your administrator username and Master Password, then click **Log In**.
3. Select **Settings > Directory integrations** in the left navigation.
4. Click on the **Azure AD** tab.
5. Under Connection, copy the *URL* and paste it into your text editor application.
6. Click the **Create Provisioning Token** to generate it, then copy the *Token* and paste it into your text editor application.

   *Important: If you navigate away from the Azure AD tab within the Directory Integrations page, the Provisioning Token will no longer be accessible through the LastPass Admin Console. If the Token is lost, a new one can be generated, but this will invalidate the previous code. Any process that used the old Token will need to be updated with the new one. A new Provisioning Token can be generated by navigating back to the Azure AD tab and clicking **Reset Provisioning Token**.*
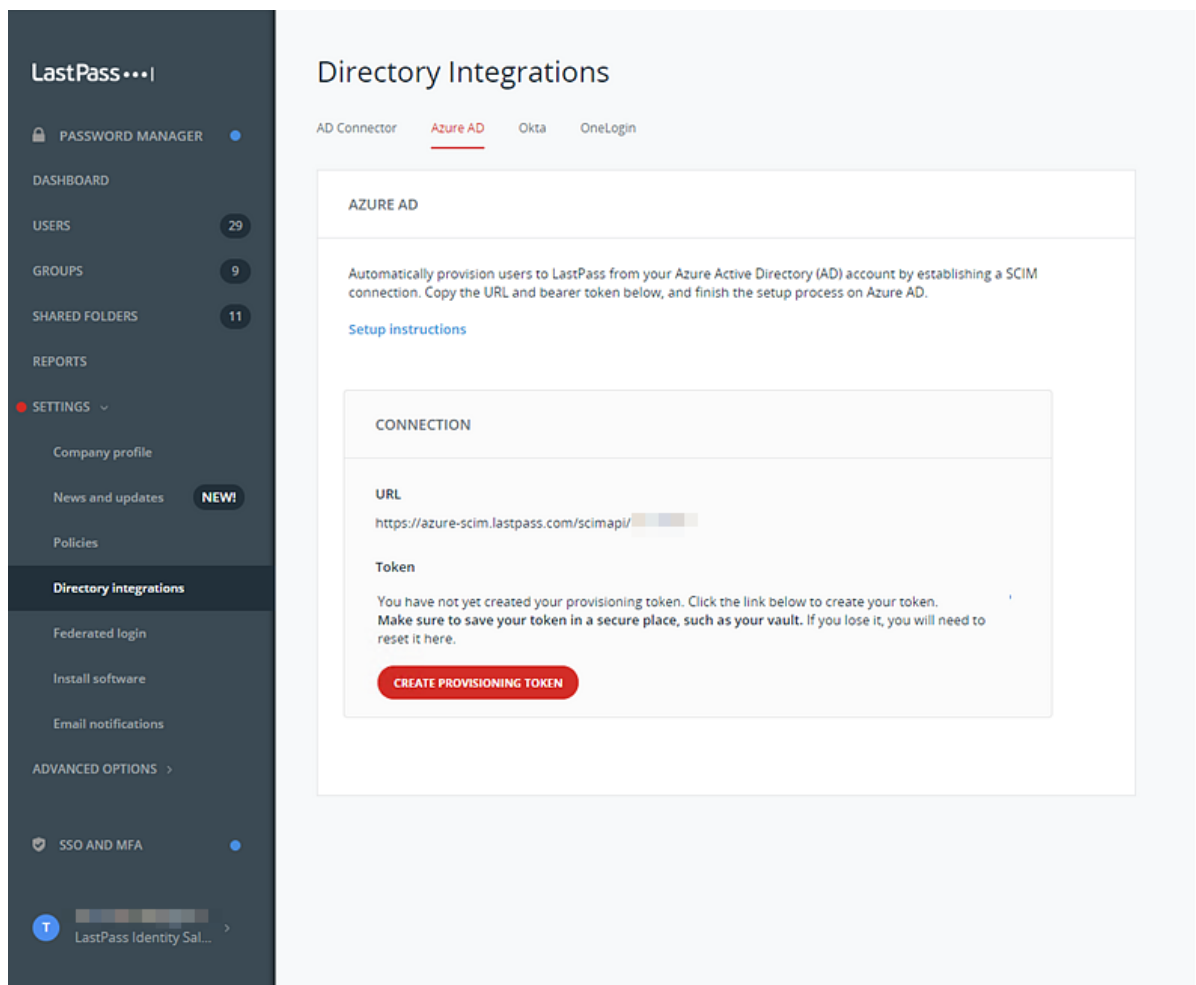
Step #2: Configure Azure AD with LastPass

Once you have acquired the URL and Provisioning Token, you will need to enter them into the Azure AD Admin portal.

1. Log in to your Azure AD portal with your administrator account credentials at https://portal.azure.com.
2. Navigate to **Azure Active Directory** > **Enterprise Applications** > **New application** > **All** > **Non-gallery application**.
3. Enter a name for your application (LastPass) and click **Add** to create an app object. The application object created is intended to represent the target app (for which you would be provisioning and setting up single sign-on, not just as the SCIM endpoint).
4. Select the **Provisioning** tab in the left navigation.
5. For Provisioning Mode, use the drop-down menu and select **Automatic**.
6. Under Admin Credentials, enter the following:
   a. Locate the "Tenant URL" field and paste the **URL** you copied from the LastPass Admin Console.
   b. Locate the "Secret Token" field and paste the Provisioning **Token** you copied from the LastPass Admin Console.
7. Click **Test Connection** to have Azure AD attempt to connect to the SCIM endpoint. If the attempts fail, error information is displayed.
8. If the connection test succeeds, click **Save** to store the admin credentials.
9. Next, select **Mappings**.
10. Click **Synchronize Azure Active Directory Users to <app name>** to modify user object mappings.

11. Modify the User mappings with the following:
    a. Check the box for **Show advanced options** at the bottom of Attribute Mapping.
    b. Click **Edit attribute list for <app name>**.

Attribute Mapping   □ ✕

| 🖫 Save   ✕ Discard |

Source Object Scope
All records   >

Target Object (customappsso)
urn:ietf:params:scim:schemas:extension:enterprise:2.0:User   ⌄

Target Object Actions

☑ Create
☑ Update
☑ Delete

Attribute Mappings
Attribute mappings define how attributes are synchronized between Azure Active
Directory and customappsso

AZURE ACTIVE DIRECTORY ATTRIBUTE	CUSTOMAPPS...	MATCHING ...	
objectId	externalId	1	Delete
Switch([IsSoftDeleted], , "False", "True", "Tr	active		**Delete**
displayName	displayName		**Delete**
userPrincipalName	userName		Delete
Join(" ", [givenName], [surname])	name.forma...		**Delete**

Add New Mapping

☑ Show advanced options

Supported Attributes
View and edit the list of attributes that appear in the source and target attribute lists for
this application.

The attribute list for Azure Active Directory is up to date with all supported attributes.
Request additional attributes you would like to see supported here.

Edit attribute list for customappsso

    c. In the **Edit Attribute List**, make the following selections:
        • Name = id, Type = String – Check the boxes for **Primary** and **Required**
        • Name = userName, Type = String – Check the box for **Required**
        • Name = externalID, Type = String – Check the box for **Required**
    d. Click **Save** and return to Attribute Mapping.

Edit Attribute List

🖫 Save   ✕ Discard

customappsso User Attributes

NAME	TYPE	PRIMAR...	REQUIR...	MULTI-...	EXACT ...	API EXPRESSION	REFERENCED OBJE...	
id	String	☑	☑	☐	☐			Delete
active	Boolean	☐	☐	☐	☐			Delete
displayName	String	☐	☐	☐	☐			Delete
userName	String	☐	☑	☐	☐			Delete
externalId	String	☐	☑	☐	☐			Delete

12. Under the "Attribute Mapping" section, Azure may have created mappings already, but those can be modified and/or deleted.

> *Important: Only the required 4 mappings should be present after editing, and must be configured correctly. You **MUST** delete all extra mappings except for the ones listed below, otherwise you will encounter synchronization issues.*

13. Modify the User Attributes as follows:
    a) **ExternalID** – Use the objectID attribute from Azure AD and set this as a matching attribute with Precedence set as **1**.
       - *Note: This should be the only mapping with any Precedence set. In order to change the ExternalID Precedence to 1, you may need to modify another attribute that might already have a Precedence set to 1. **After you find such attribute, you can change its precedence from 1 to 2, <u>then go back to ExternalID and set its Precedence to 1</u>**. Finally, to remove the Precedence entirely from the other attribute (now set to 2), you can now edit it once again and set the "Match objects using this attribute" to **No**.*
    b) **Active** – The default Azure AD mapping can be used, or a custom one which will be used to set the user as enabled/disabled in LastPass.
    c) **DisplayName** – Use any property from Azure AD. This should be a string which will be the synchronized user's name in LastPass.
    d) **UserName** – Map the user's email address from Azure AD. Please note that the userPrincipalName might not be equal to the email address. In this case, use an attribute from Azure AD which contains the email address the user will utilize and can read (e.g., Mail or in most cases, userPrincipalName should be fine).

    > *Important: If you already have users in LastPass, their email address **MUST** match the Azure AD attribute mapped to the userName value.*
    > *If this is not mapped correctly, a duplicate user will be created for every existing user in LastPass.*

14. Click **Save**, then return to the Provisioning settings and select **Mappings** (from **Step #9** above).
15. Click **Synchronize Azure Active Directory Groups to <app name>** to modify group object mappings.

16. Next, modify group object mappings as follows:
    a) Check the box for **Show advanced options** at the bottom of Attribute Mapping.
    b) Click **Edit attribute list for <app name>**.
    c) In the Edit Attribute List, make the following selections:
        - Name = id, Type = String – Check the boxes for **Primary** and **Required**
        - Name = externalID, Type = String – Check the box for **Required**
        - Name **=** displayName, Type = String – Check the box for **Required**
        - Name **=** members, Type = Reference – Check the box for **Multi-Valued**, then set referenced objects for:
            o **urn:ietf:params:scim:schemas:core:2.0:Group**
            o **urn:ietf:params:scim:schemas:extension:enterprise:2.0:User**
    d) Click **Save** and return to Attribute Mapping.
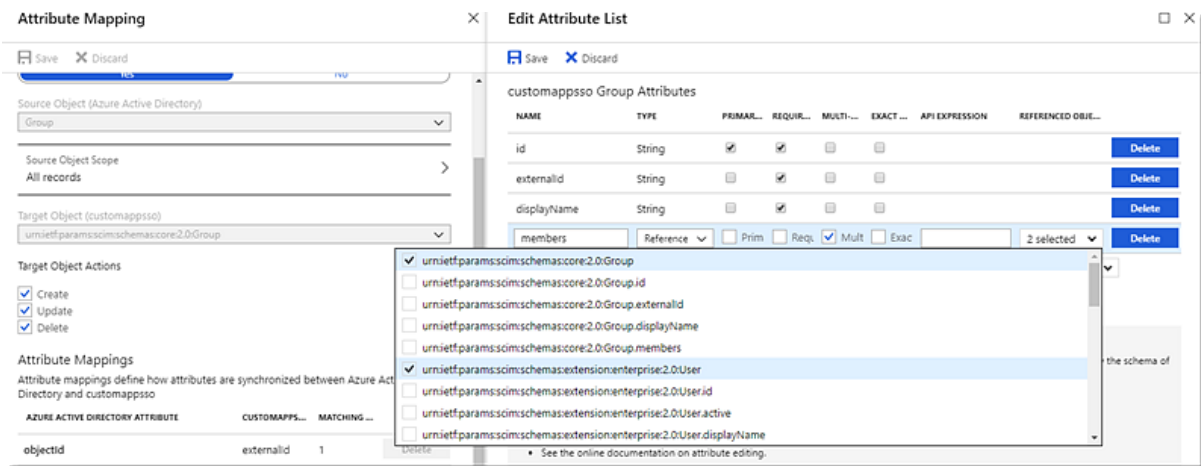
17. Under the "Attribute Mapping" section, Azure may have created mappings already, but those can be modified and/or deleted.

    *Important: Only the required 3 mappings should be present after editing, and must be configured correctly. You **MUST** delete all extra mappings except for the ones listed below, otherwise you will encounter synchronization issues.*

18. Modify the Group Attribute Mapping rules as follows:
    a) **ExternalID** – Use the objectID attribute from Azure AD and set this as a matching attribute with Precedence set as **1**. This should be the only mapping with any Precedence set.
    b) **DisplayName** – Use any attribute for group name.
    c) **Members** – User members from Azure AD.

19. Click **Save**, then return to the Provisioning settings.
20. Under Settings, the Scopefield defines which users and or groups are synchronized. Selecting **Sync only assigned users and groups** (recommended) will only sync users and/or groups assigned in the Users and groups tab.
    - ***Note:*** *If syncing only assigned users and groups (recommended), be sure to select the **Users and groups** tab and assign the users and/or groups you wish to sync.*
21. Once your configuration is complete, enable the Provisioning Status by clicking **On**.
22. Click **Save** to start the Azure AD provisioning service.
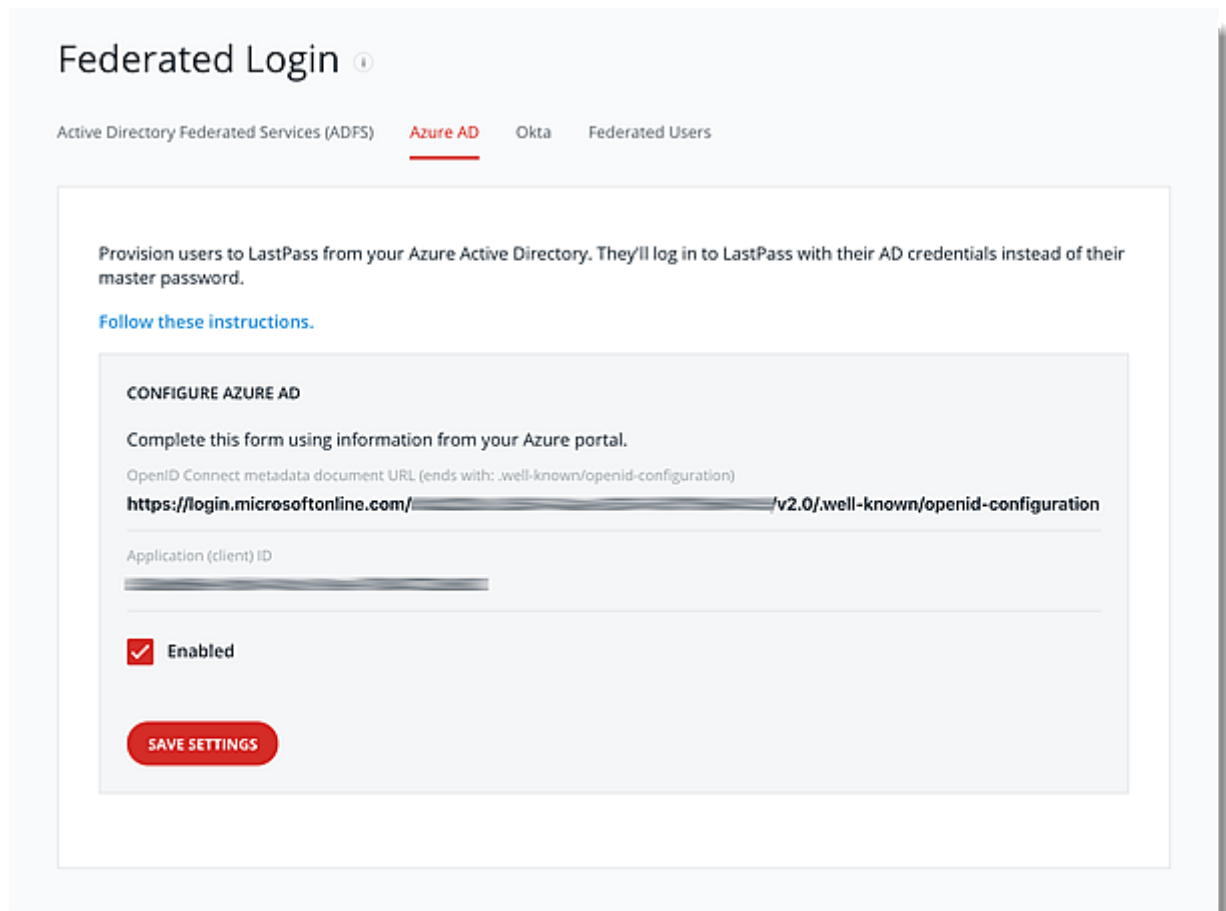
Step #3: Capture the Application ID and OpenID Connect from Azure AD

1. In the Azure AD portal, navigate to your directory.
2. Select **App registrations** in the left navigation, then select **the name of your app**. You will then be redirected to the **Overview** page.
3. With **Overview** selected in the left navigation, copy the *Application (client) ID* field contents and paste it into your open text editor.
4. Click **Endpoints** in the top navigation to expand the menu on the right.
5. Copy *OpenID Connect metadata document* field contents and paste it into your open text editor.
6. Proceed to the next step below where these items will be used.
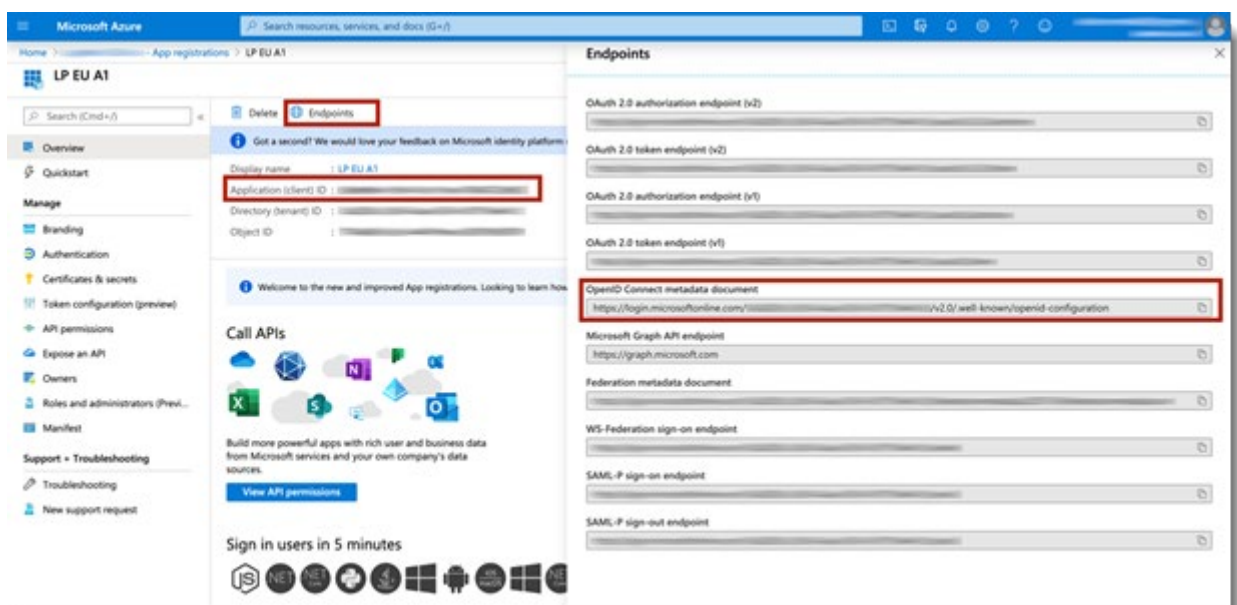
# Step #4: Configure Federated login settings in LastPass

1. Go back to the LastPass Admin Console, then select **Settings** > **Federated login** in the left navigation.
2. Select the **Azure AD** tab, then enter the following:
   - In the "Directory (tenant) ID" field, paste the **OpenID Connect metadata document** from **Step #3** (in the previous section).
   - In the "Application (client) ID" field, paste the **Application (client) ID** from **Step #3** (in the previous section).
3. Check the box for **Enabled**.
4. Click **Save Settings** when finished.

Step #5: Configure a Redirect URI in Azure AD

1. In the Azure AD portal, select **App Registrations**, then select your **<app name>.**
2. Click **Authentication**, then click **Switch to the Old Experience**.
3. Select **Overview** in the left navigation.
4. Under Redirect URI in the upper-right navigation, click **Add a Redirect URI**.
5. Add the first Redirect URI, as follows:
   - For the Type column, use the drop-down menu and select **Web**
   - For the Redirect URI column, enter either of the following:
     i. For accounts using **US data centers**:
        https://lastpass.com/passwordreset.php
     ii. For accounts using **EU data centers**:
        https://lastpass.eu/passwordreset.php
6. Add the second Redirect URI, as follows:
   - For the Type column, use the drop-down menu and select **Web**
   - For the Redirect URI column, enter
     https://accounts.lastpass.com/federated/oidcredirect.html
7. Under the Implicit Grant settings, check the boxes to enable the following settings:
   - **Access tokens**
   - **ID tokens**
8. Click **Save** when finished.
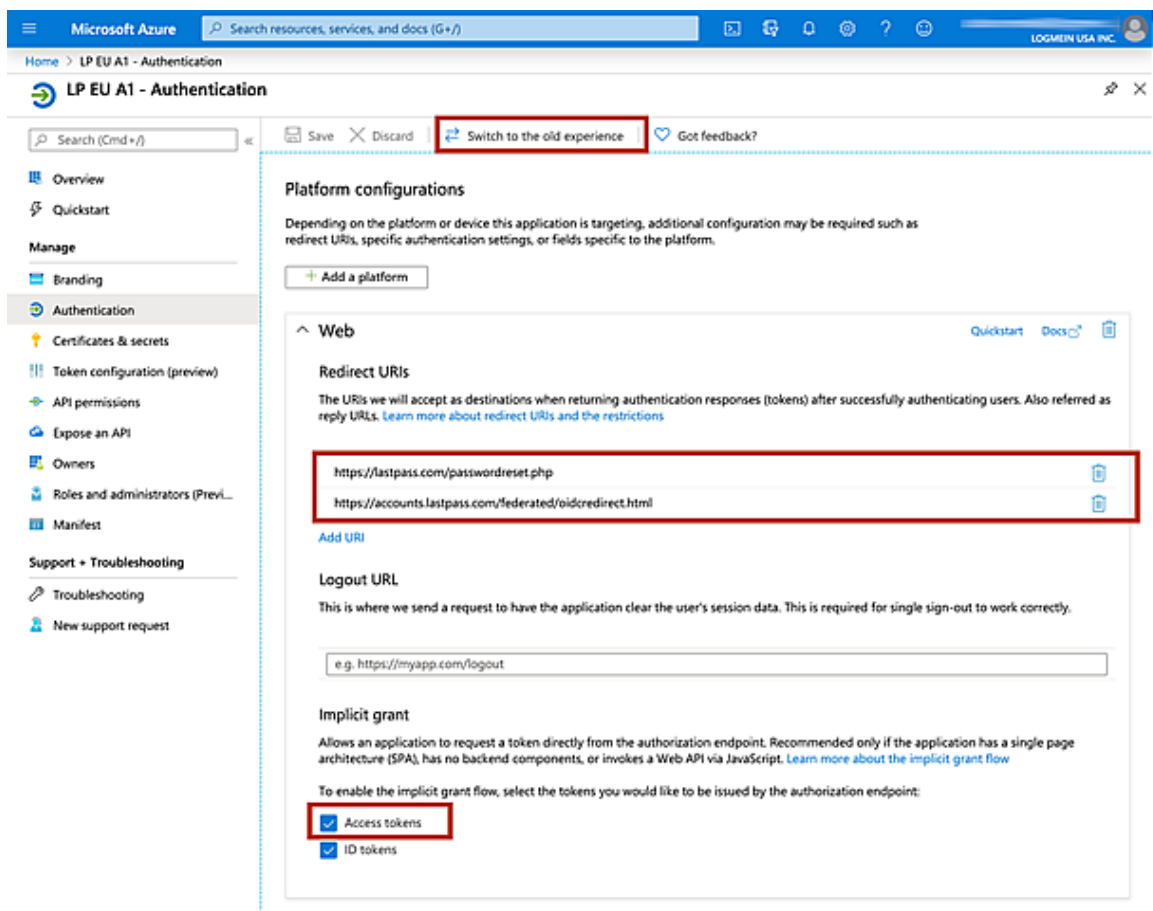
Step #6: Configure API permissions in Azure AD

1. In the Azure AD portal, select **API permissions** in the left navigation.
2. Click the **Add a permission** button, then select **Microsoft Graph**.
3. In the right navigation, select **Delegated permissions**.
4. Under the Permission menu, check the boxes to enable the following permission settings:
   - **email**
   - **openid**
   - **profile**
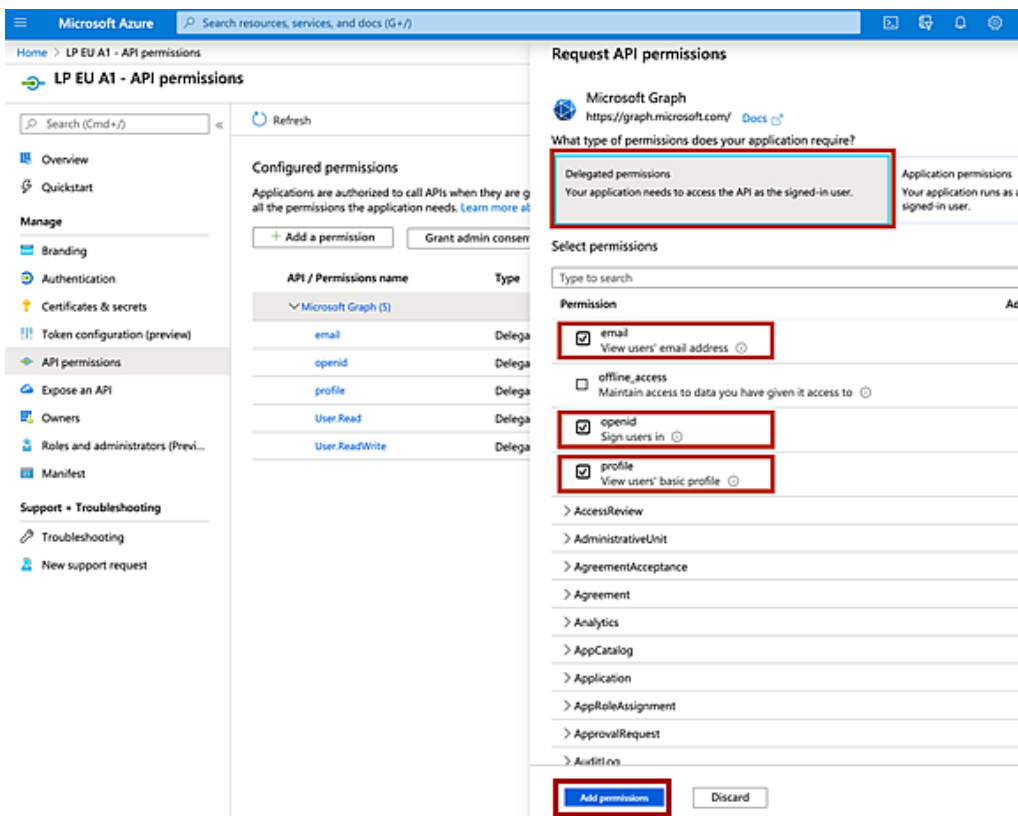
5. Under the User menu, check the boxes to enable the following user settings:
   - **User.Read**
   - **User.ReadWrite**
6. When finished, click **Add permissions**.
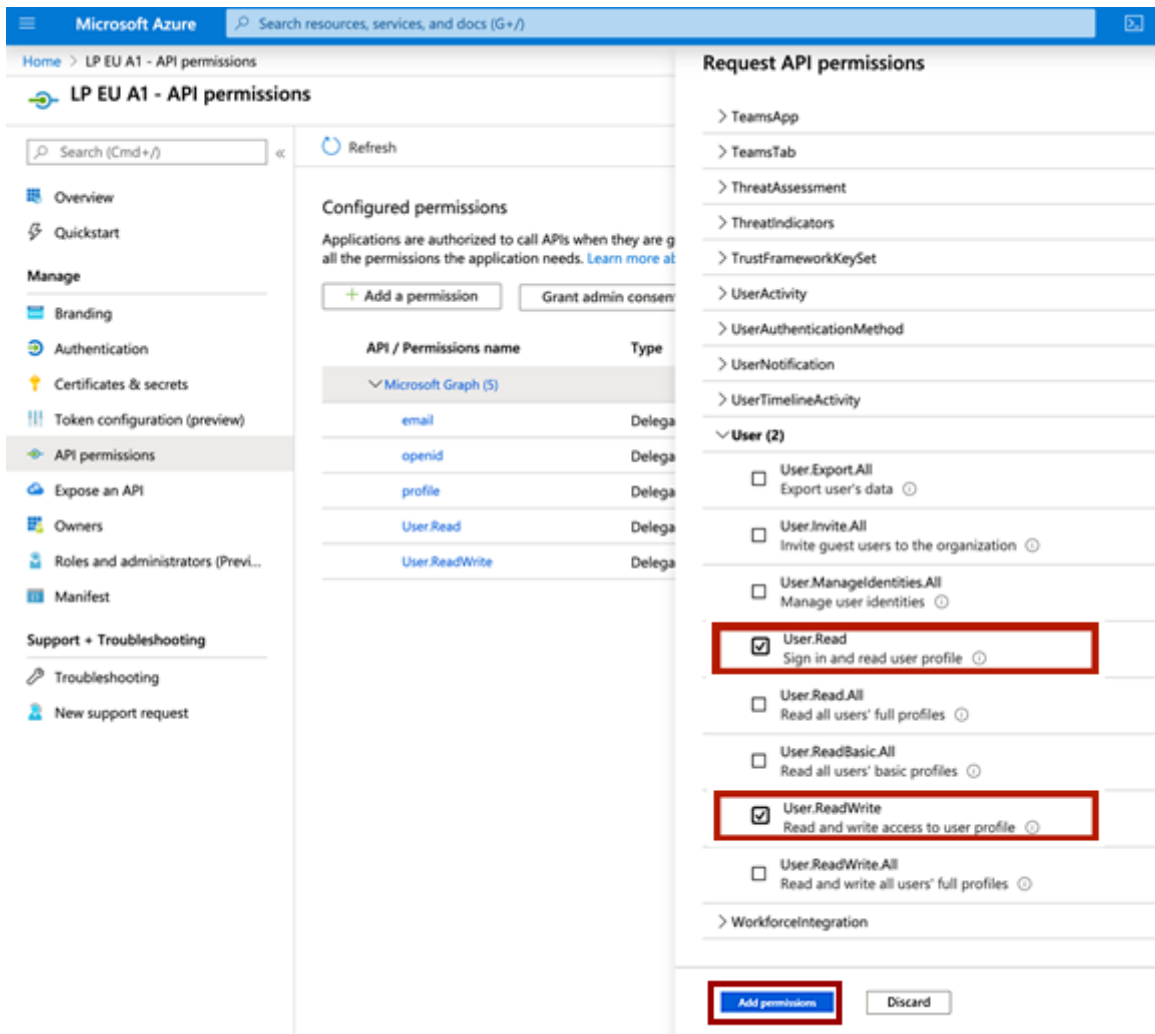
7. Under Configured Permissions, click **Grant <your LastPass application name>** to finish configuring API permissions for your LastPass app.
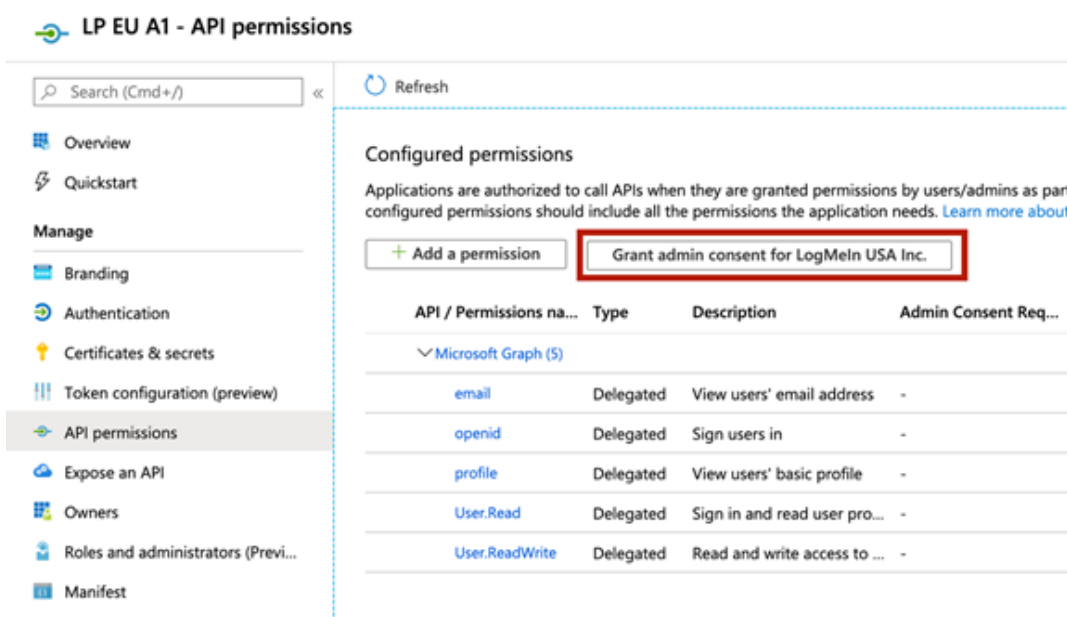
Step #7: Add users to the LastPass app in Azure AD

1. In the Azure AD portal, with your LastPass application selected, go to **Overview** > **Enterprise applications** in the left navigation.
2. Select your newly created LastPass application.
3. Select **Users and groups** in the left navigation.
4. Click **Add user**.
5. Locate each of the users and/or groups in the list, then click **Select** to grant access to the LastPass app.
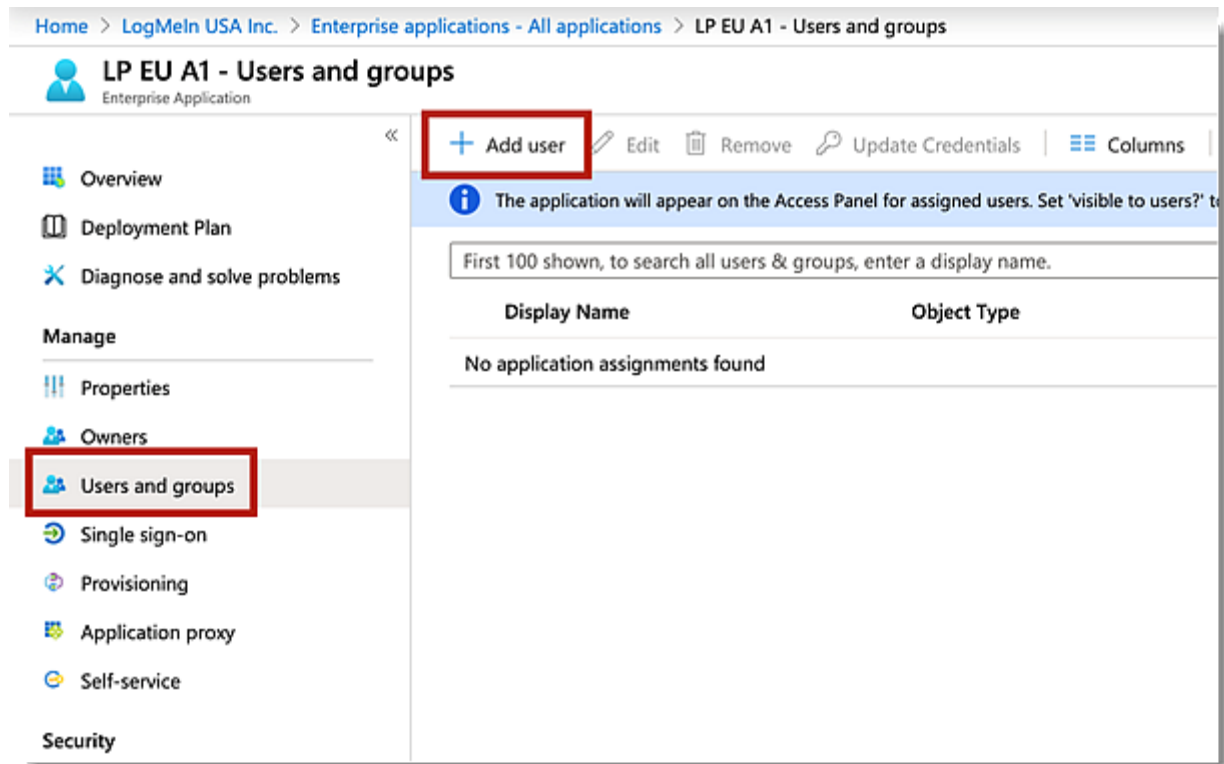
Step #8: Set up Multifactor Authentication on Azure AD (optional)

If desired, you can set up Multifactor Authentication at the Azure AD (Identity Provider) level.

You're all set!

You have successfully set up your LastPass Enterprise or LastPass Identity account to use federated login with your Azure Active Directory. All of your newly populated federated users will receive a Welcome email informing them that they can now log in to use LastPass. Please note that your LastPass users must log in using the LastPass web browser extension in order to use federated login for their Azure AD account with LastPass.

- To learn more about deploying the LastPass web browser extension to your organization, please see Install LastPass Software Using the Admin Console.
- To see your end users' experience, please see Federated Login Experience for LastPass Users.
- If your end users have linked personal accounts associated with their federated login account, please see How do I verify my linked personal account?
- To convert a non-federated user to a federated user, please see How do I convert an existing LastPass user to a federated (Azure AD) user?

Troubleshooting & Tips

- It is **required** that you enable the "Permit super admins to reset Master Passwords" policy for at least 1 LastPass admin (who is also a non-federated admin) in the LastPass Admin Console. This ensures that all LastPass user accounts can still be recovered (via Master Password reset) if a critical setting is misconfigured or changed for federated login after setup is complete.

Contact Us

If you have not started a LastPass Enterprise or LastPass Identity trial, please contact our Sales team at lastpass.com/contact-sales for more information.

For additional help, please see Set Up Federated Login for LastPass Using Azure Active Directory, and if desired, select a contact option at the bottom of the article.