25/01/2021

# MS AzureAD: Manage users in your Administrative Unit

English

Bosbec AB
Färögatan 33
164 51 Kista

Phone: +46 (8) 22 25 25
E-Mail: info@bosbec.se
www.bosbec.se

# Table of Contents

# Introduction

This documentation will guide you through the set-up of adding and removing Microsoft users from your AzureAD Administrative Unit, externally from the Bosbec interface. With this functionality you can manage all your organizations Microsoft Users and Administrative Units directly from the Bosbec platform.

In this tutorial you will find information about creating an “Administrative Unit”, “AzureAD application” and importing the Workflow Template to manage your units from your Bosbec account.

**Requirements**

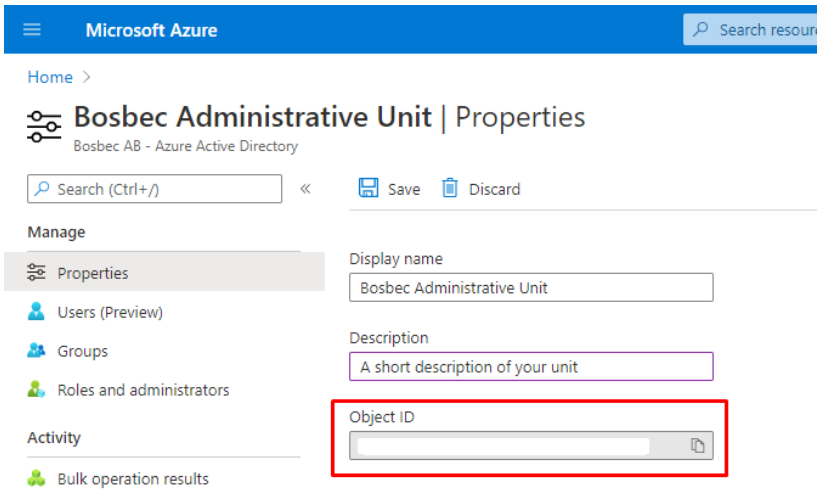
- Microsoft AzureAD Tenant (MS AzureAD account and organization team)
- Privileged Role Administrator for your MS AzureAD account in your organization

# Set up MS Azure Active Directory

This section provides you with the required steps to access your AzureAD externally. Remember to save any IDs and secrets in a safe place. These IDs will be used later in the Bosbec Workflow Builder to complete the requests to Microsoft from the Bosbec platform.

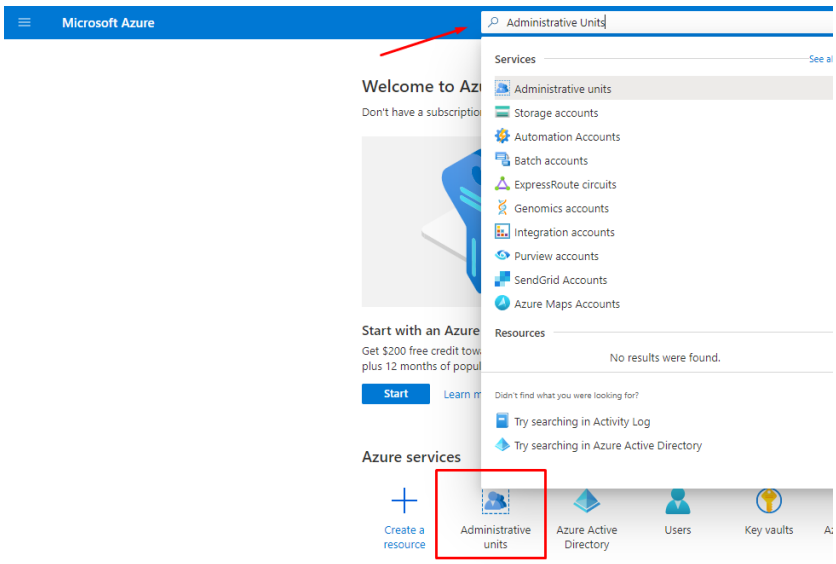
## Creating an Administrative Unit

If you already have an Administrative Unit on your AzureAD account, you can skip this step. Just remember to acquire the Administrative Units Object-ID in the properties-menu of your Administrative Unit (Administrative Units -> Your Unit -> Properties -> Object-ID).




*Copy the Object-ID of your Administrative Unit.*

Start by logging in to your MS AzureAD account. Navigate to your “Administrative Units” either by locating the directory in your “Azure Services” or search for “Administrative Units” in the top search bar.

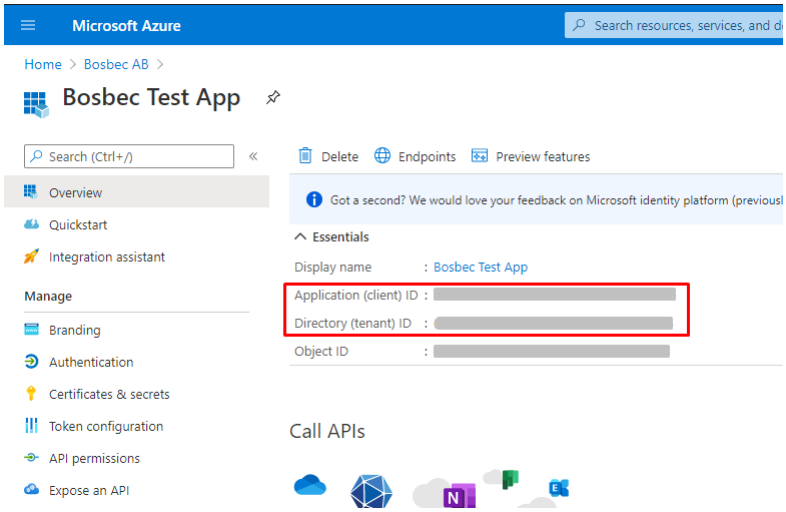



*Navigate to your Administrative units in the AzureAD interface.*

Once you have navigated to the directory, click “Add”. Give your Administrative Unit a name, description and assign any roles you want to assign to the unit. Save the Object-ID.

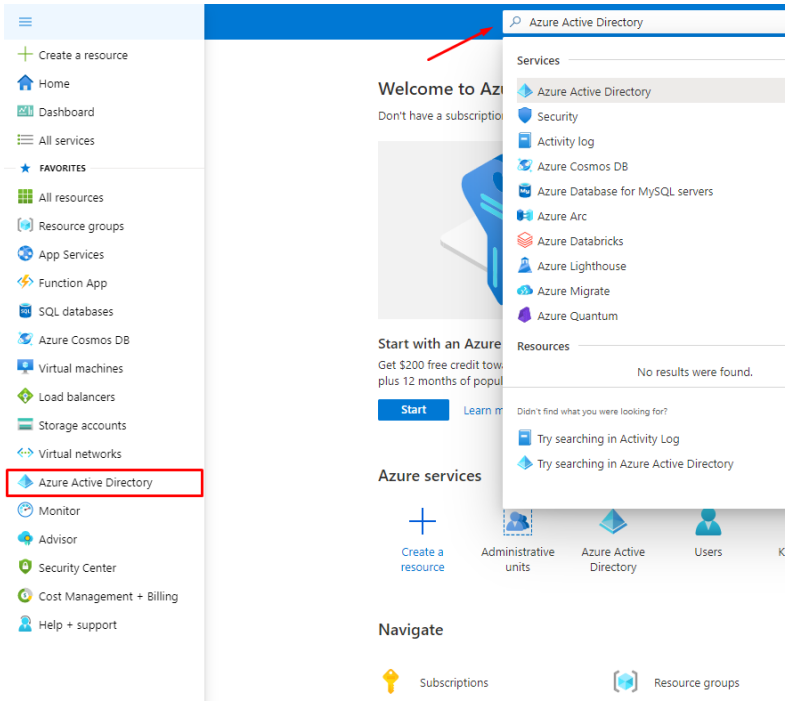
## Creating an MS Azure Application

If you already have an MS Azure Application, you can skip this step. Just remember to acquire the Application client-ID and Directory tenant-ID.



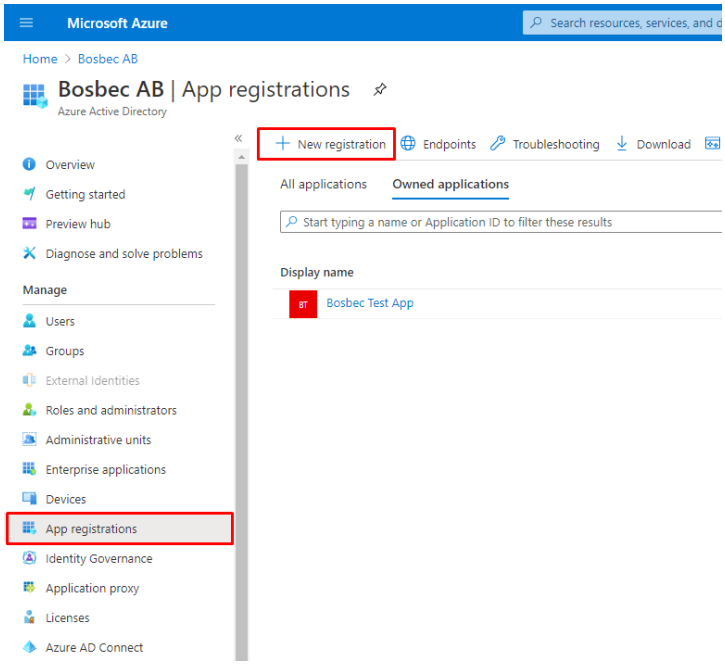

*Copy the client-ID and the tenant-ID of your application.*

Go to your Azure Active Directory to create an application. Either search for “Azure Active Directory” in the top search bar or locate the directory in the navigation bar to the left of Azure Portal.



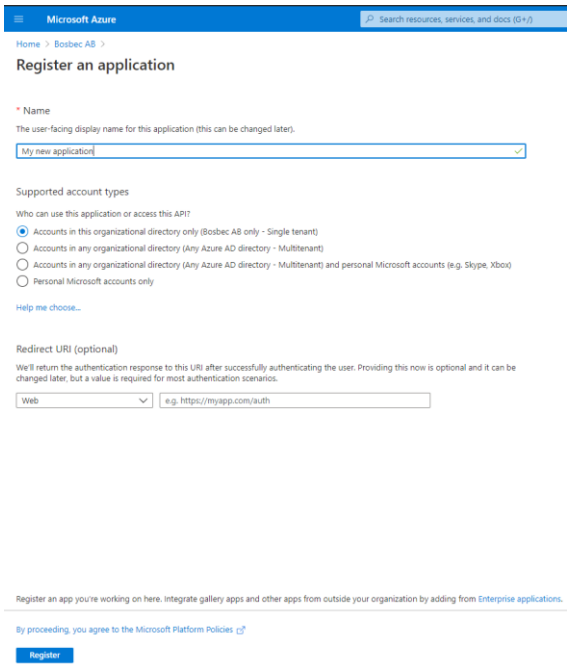

*Navigate to your Azure Active Directory in the AzureAD interface.*

Go to your “App registrations”, located in the navigation bar to the left. Here, click on “New Registration” to create a new application.




*Create a new app registration.*

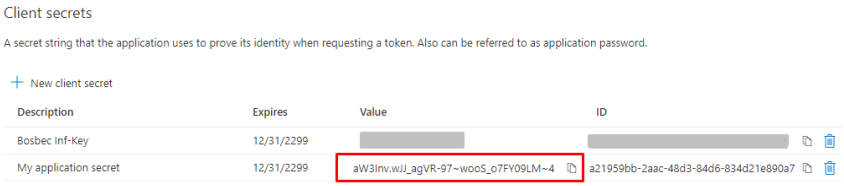
Give your application a name and select the level of permission.  Redirect URI can be left out.




*Create an application in AzureAD.*

To make your API-requests unique and secure, create a secret for your application. Navigate to “Certificates & secrets” located in the navigation bar to the left. Select “New client secret”. Select the expiration time for the secret and click “Add”.

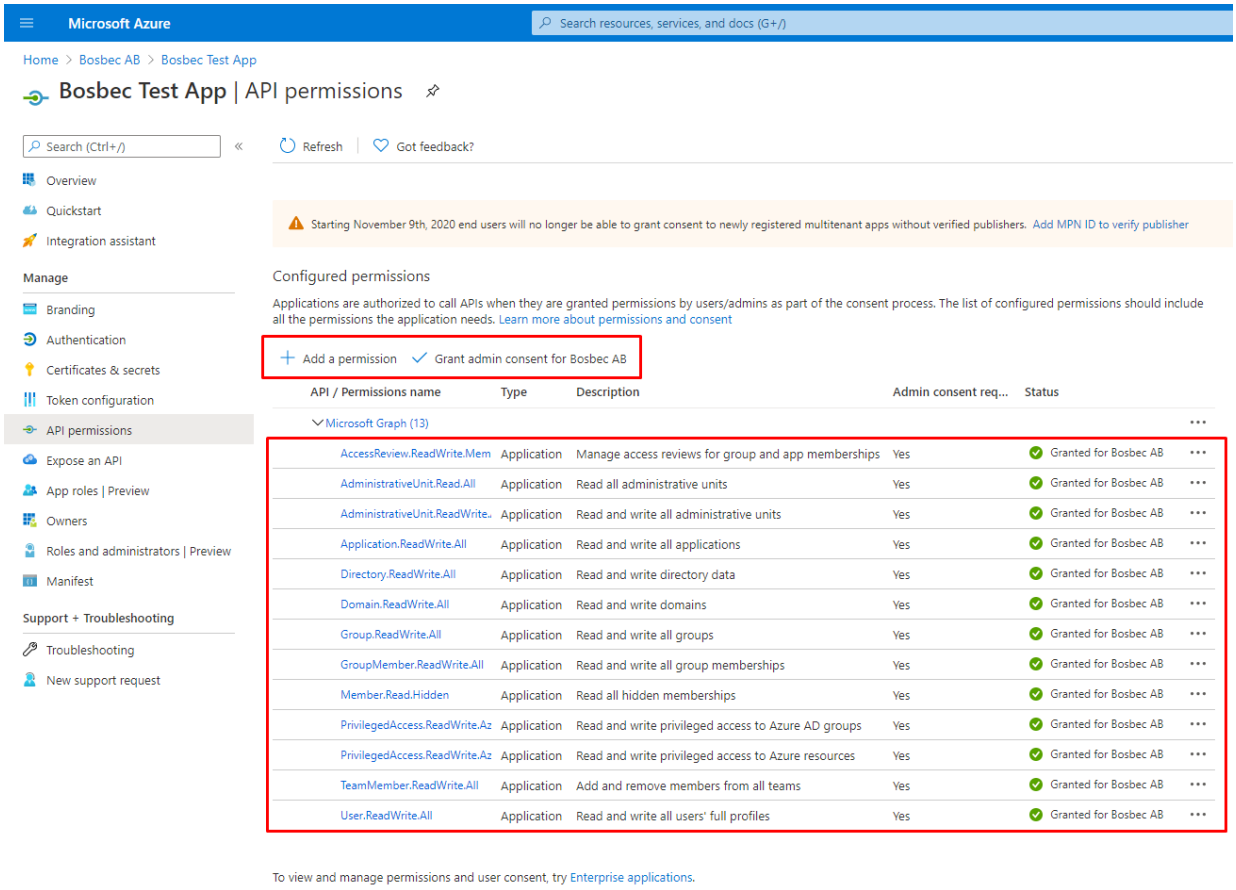
**Observe!** Once you have created the secret you can only access its full value **once**! Copy your secret and store it in a safe place, otherwise, you will have to create a new token.




***Important**: Copy your application secret value and store it in a safe place!*

Next, you need to add API permissions to your application to acquire the **MS AzureAD Access Token**. This token is generated through your application and is required to make API-requests to your AzureAD account in order to manage your Administrative unit.

Listed below are the necessary permissions for your application. Select “Add a permission” on the “API permissions” directory, located in the navigation bar to the left. Search for the permissions below in the permission list (located under “Microsoft Graph”) and add them to your application. Remember to “Grant admin consent” for the requested permissions.




*Add the following permissions to your application. Remember to grant admin consent!*

This list is also available at the end of this documentation file.

Well done! Your AzureAD is now set up and ready to be managed externally by your Bosbec account. To summarize all the steps made from the AzureAD Portal, please confirm that all steps in the checklist below has been conducted.

- Create and/or acquire your Administrative Unit Object-ID
- Create a AzureAD Application
- Acquire your Application client-ID and Directory tenant-ID from your application.
- Create a secret for your application. Remember to store it in a safe place!
- Provide your application with all necessary permissions.

# Configure your Workflow

## Import the Workflow Template

In your Workflow Builder, select “Edit” and “Workflow Library”. Expand the “Templates”-folder and select “Manage AzureAD Administrative Units”.

## Replace the placeholders in the template

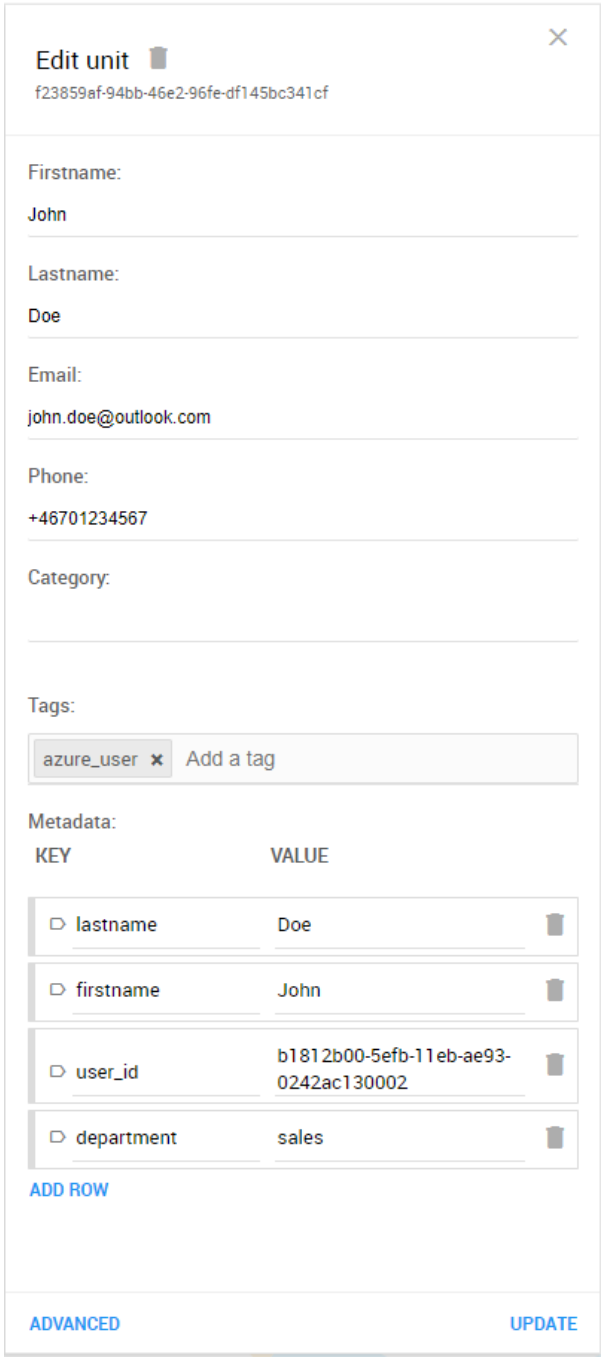
Listed below are the following workflows jobs which contains placeholders for all IDs you have acquired in your previous steps.

- **get_access_token** (Remote http request): Replace [TENANT] in the “Url”-field with your Directory Tenant-ID.
- **get_access_token** (Remove http request): Replace [CLIENT] in the “Post template”-field with your Application Client-ID.
- **get_access_token** (Remove http request): Replace [SECRET] in the “Post template”-field with your Application Secret.
- **add_user_administrative_unit** (Remote http request): Replace [UNIT] in the “Url”-field with your Administrative Unit Object-ID.
- **add_user_administrative_unit** (Remote http request): Replace [USER] in the “Post template” with a user ID you wish to add.
- **remove_user_from_administrative_unit** (Remote http request): Replace [UNIT] in the “Url”-field with your Administrative Unit Object-ID.
- **Remove_user_from_administrative_unit** (Remote http request): Replace [USER] in the “Url”-field with a user ID you wish to remove.

## Run your workflow

Run your solution by clicking “Run” in the top right corner of the Workflow Builder, select you trigger and click “Run”. Now you have externally altered your AzureAD Administrative unit from the Bosbec platform.

# Store data on a Bosbec Unit




*Configure your data entity (e.g a Microsoft user)*

Bosbec Units are data entities used to store information about staff members, IoT devices or processes. You can store data about your Microsoft users on Bosbec units to easily access them through the workflow.

In the image to the left you can see a Bosbec Unit. This unit represents a Microsoft user, and is doing so by you adding data to the entity. You can tag your unit with “azure_user”, and by searching for that tag in a workflow, collect all Microsoft users and manage them in bulk operations.

Here you can also add specific information about the user, such as “user_id” which is necessary for the workflow to add the user to the Administrative Unit. But also specific information such as “department”. In this case, John Doe works in sales and is represented by this Bosbec unit.

# Further reading

## Further development

You can further develop your solution. Use “Unit Operations” to find all Bosbec units which can be added to your Administrative Unit. Tag all Bosbec Units and access them with the “Unit Operation”-job. Then iterate through all units and conduct multiple “Remote http requests” for each found unit.

Additional documentation and information about Bosbec integration and functionality can be found at https://help.bosbec.io/.

More documentation about Microsoft AzureAD set up can be found in additional documentation below:

- https://docs.microsoft.com/en-us/azure/active-directory/roles/admin-units-add-manage-users
- https://docs.microsoft.com/en-us/graph/api/administrativeunit-delete-members?view=graph-rest-1.0
- https://docs.microsoft.com/sv-se/azure/active-directory/roles/administrative-units
- https://docs.microsoft.com/en-us/azure/active-directory/roles/admin-units-manage
- https://docs.microsoft.com/en-us/azure/data-explorer/provision-azure-ad-app
- https://docs.microsoft.com/en-us/graph/auth-v2-service

## List of API permissions for AzureAD application

The list below contains all required permissions for the AzureAD application, all of which can be found in the “Microsoft Graph” section, when adding new permissions to your app.

- AccessReview.ReadWrite.Membership
- AdministrativeUnit.Read.All
- AdministrativeUnit.ReadWrite.All
- Application.ReadWrite.All
- Directory.ReadWrite.All
- Domain.ReadWrite.All
- Group.ReadWrite.All
- GroupMember.ReadWrite.All
- Member.Read.Hidden
- PrivilegedAccess.ReadWrite.AzureADGroup
- PrivilegedAccess.ReadWrite.AzureResources
- TeamMember.ReadWrite.All
- User.ReadWrite.All