RLDatix

Setting up Azure to Send Email Using OAuth

Table of Contents

Overview

This guide details how to configure an Azure Enterprise Application to send emails using OAuth authentication with Microsoft Graph API or Office 365 SMTP. It includes all necessary PowerShell commands to register a Service Principal and grant Full Access permissions to a mailbox. To complete these steps, the user must have Global Admin rights within the tenant.

Where possible it is recommended to use Microsoft Graph (Azure) for OAuth authentication as it is easier to configure both on the mail server, and within RiskMan

Setup on the Mail Server

Step 1: Register an Azure AD Application

1. Create a New App Registration
    a. Log in to Azure Portal (portal.azure.com).
    b. Navigate to Azure Active Directory > App registrations.
    c. Click + New registration.
    d. Enter a name (e.g., RiskManSMTP-Mail-Application).
    e. Choose Accounts in this organizational directory only.
    f. Click Register.
2. Record Key Information
    a. After registration, record the following values:
        - Application (client) ID
        - Directory (tenant) ID

Step 2a: Configure API Permissions for use with Microsoft Graph (Azure) Option in RiskMan

Add Microsoft Graph API Permissions (ie for ***Microsoft Graph (Azure)*** sending type within RiskMan)

1. Navigate to Azure Active Directory > App registrations.
2. Select the newly created app.

3. Go to API permissions > + Add a permission.
4. Choose Microsoft Graph.
5. Select Application permissions.
6. Search and add the following permissions:
    - Mail.Send
7. Click Add permissions.
8. Click Grant admin consent for [Your Tenant].

Note – Application Permissions (not delegated) when you register an Azure AD App and grant it application-level permissions like:

Mail.Send

… and give admin consent, the app can:

- Send emails as any mailbox (including shared mailboxes)
- Do this without user login (no delegated access)
- Work even if the shared mailbox itself has no license

OR Step 2b: Configure API Permissions for use with MailKit Option in RiskMan

Add Exchange Online SMTP Permissions (ie for ***MailKit*** sending type within RiskMan)

1. Navigate to API permissions > + Add a permission.
2. Select APIs my organization uses.
3. Search for Office 365 Exchange Online.
4. Select Application permissions.
5. Add:
    - SMTP.SendAsApp
6. Click Add permissions.
7. Grant admin consent. for [Your Tenant].

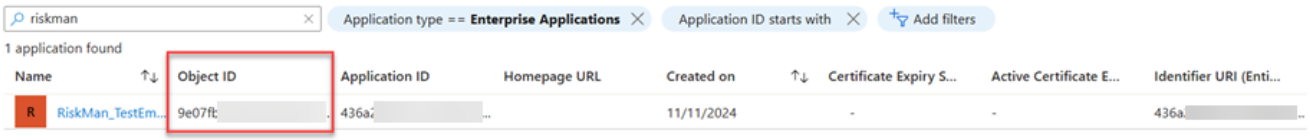
Step 3: Generate and Store Client Secret

1. Create Client Secret
    a. Navigate to Certificates & secrets.
    b. Click + New client secret.
    c. Enter a description (e.g., SMTP-Secret).
    d. Set expiry (e.g., 1 or 2 years).
    e. Click Add.
    f. Copy and save the Value (Client Secret). Save this value because this cannot be viewed after exiting

Step 4: Assign the App Service Principal in Exchange Online (Only required for *MailKit* sending type within RiskMan)

This part required only for the above Step 2b to complete the configuration

1. Connect to Exchange Online
    a. Run the following PowerShell command to connect to Exchange Online ensuring to update for your domain
    b. Connect-ExchangeOnline -UserPrincipalName admin@yourdomain.com
2. Create a Service Principal
    a. Replace <APPLICATION_ID> from Step 1. To get the <OBJECT_ID>, Go to Enterprise applications – Select your application - copy the value in Object ID



Note: The Object ID on the App Registration page differs from that on the Enterprise Applications. If the incorrect one is copied, authentication will fail. Make sure to get it as highlighted

    b. `New-ServicePrincipal -AppId <APPLICATION_ID> -ObjectId <OBJECT_ID> -DisplayName "RiskManSMTP-Mail-App"`

3. Get Service Principal Identity
    a. `$EXOServicePrincipal = Get-ServicePrincipal -Identity "RiskManSMTP-Mail-App"`

4. Assign FullAccess Permissions to the Mailbox
    a. Replace user@domain.com with the mailbox you want to use:
    b. `Add-MailboxPermission -Identity "user@domain.com" -User $EXOServicePrincipal.Identity -AccessRights FullAccess`
5. Verify Permissions
    a. Get-MailboxPermission -Identity user@domain.com
    b. If all is successful, the message should look like this

Setup within RiskMan

Step 5a: Configure OAuth on RiskMan for use with Microsoft Graph (Azure)

When the Global Setting for Mail | 10) Email Sending Method is set to *Microsoft Graph (Azure)*, the following settings will need to be configured under Mail | Mail Configuration:

Note: Some of these may already have been configured for the use of MailKit, however if using Microsoft Graph (Azure) will need to be updated as appropriate, and the settings highlighted are new and specific to OAuth

Global Setting	Description
21) Mail Server Client ID for OAuth	This setting is the Client ID from the Mail server when using the OAuth authentication method. This is a requirement for using Azure mail servers without using BASIC authentication. Basic authentication is due to be deprecated on Azure mail servers so this will be a mandatory. To work, all 3 OAuth values need to be supplied, ClientID, TenantID and ClientSecret, and SSL/TLS must be enabled.
22) Mail Server Tenant ID for OAuth	This setting is the Client ID from the Mail server when using the OAuth authentication method. This is a requirement for using Azure mail servers without using BASIC authentication. Basic authentication is due to be deprecated on Azure mail servers so this will be mandatory.

	To work, all 3 OAuth values need to be supplied, ClientID, TenantID and ClientSecret, and SSL/TLS must be enabled.
23) Mail Server Secret for OAuth	This setting is the Secret from the Mail server when using the OAuth authentication method. This is a requirement for using Azure mail servers without using BASIC authentication. Basic authentication is due to be deprecated on Azure mail servers so this will be mandatory. To work, all 3 OAuth values need to be supplied, ClientID, TenantID and ClientSecret, and SSL/TLS must be enabled.
70) SMTP Username.	This setting should only be modified with the assistance of your IT Department and needs to be an email address with the appropriate permission to send emails

OR Step 5b: Configure SMTP Authentication with OAuth on RiskMan for use with MailKit

When the Global Setting for Mail | 10) Email Sending Method is set to *MailKit*, the following settings will need to be configured under Mail | Mail Configuration:

Note: Many of this are likely already configured, the settings highlighted are new and specific to OAuth

Global Setting	Description
20) The address of your Mail Server	This setting nominates the network address of your e-mail server. This may either be a TCP/IP address (e.g. 172.16.1.27), the network name of a server (e.g. MailServer1) or an Internet address (e.g. mail.mailcentral.com). The mail server must be SMTP mail capable. When using MailKit and Office365 the value would be **smtp.office365.com** For Gmail it would be **smtp.gmail.com** This setting is not used for all mail server types.
21) Mail Server Client ID for OAuth	This setting is the Client ID from the Mail server when using the OAuth authentication method. This is a requirement for using Azure mail servers without using BASIC authentication. Basic authentication is due to be deprecated on Azure mail servers so this will be a mandatory. To work, all 3 OAuth values need to be supplied, ClientID, TenantID and ClientSecret, and SSL/TLS must be enabled.

22) Mail Server Tenant ID for OAuth	This setting is the Client ID from the Mail server when using the OAuth authentication method. This is a requirement for using Azure mail servers without using BASIC authentication. Basic authentication is due to be deprecated on Azure mail servers so this will be mandatory. To work, all 3 OAuth values need to be supplied, ClientID, TenantID and ClientSecret, and SSL/TLS must be enabled.
23) Mail Server Secret for OAuth	This setting is the Secret from the Mail server when using the OAuth authentication method. This is a requirement for using Azure mail servers without using BASIC authentication. Basic authentication is due to be deprecated on Azure mail servers so this will be mandatory. To work, all 3 OAuth values need to be supplied, ClientID, TenantID and ClientSecret, and SSL/TLS must be enabled.
24) Mail Server Scope for OAuth	This setting is the Scope from the Mail server when using the OAuth authentication method. The scope describes the type of access token being requested. **Please do not alter this without consultation** This is a requirement for using Azure mail servers without using BASIC authentication. Basic authentication is due to be deprecated on Azure mail servers so this will be mandatory. To work, all 3 OAuth values need to be supplied, ClientID, TenantID and ClientSecret, and SSL/TLS must be enabled.
25) Mail Server Authentication Authority for OAuth	This setting the authority provider details when retreiving a token for the OAuth authentication method when using Mailkit. When left blank, a value suitable for Office365 will be generated. This will be "https://login.microsoftonline.com/{TenantID}" where the TenantID is inserted. For servers other than Microsoft, the entire authority value must be provided. For GMail servers, for example, the value would be expected to be "https://accounts.google.com/o/oauth2/auth"
28) Does the server require SSL enabled?	When set to "Not required" no action will be taken. When set to "SSL required", SSL functionality will be enabled in the mail client. In addition, if a port number is not supplied, the port will be set to 587. This supports "Office 365". When using "MailKit", setting this to "SSL Required" or setting port to 587 explicitly will attempt to enable TLS (if supported). To force SSL instead in MailKit, the port should be 465.
29) What port is required?	If left empty the default port is used. Only put a value here if needed. When the option "SSL required" is set then the default is 587, supporting "Office 365" Otherwise the port will be 25.

70) SMTP Username.	This setting should only be modified with the assistance of your IT Department. When using OAUTH, this will be the email address of a user with permission to send emails. This is expected to be a service account address created for this purpose.
80) SMTP Password.	This setting should only be modified with the assistance of your IT Department.

If any errors occur using OAUTH, verify the following:

- The App ID, Object ID, Tenant ID and SMTP Username are correct.
- The correct permissions are granted and consented.
- The service principal has proper mailbox access.
- The client secret is valid.
- Check the global setting allowing sending emails as the RiskMan user. To utilize this, each user will require a valid email address with send permission in the Azure system.