

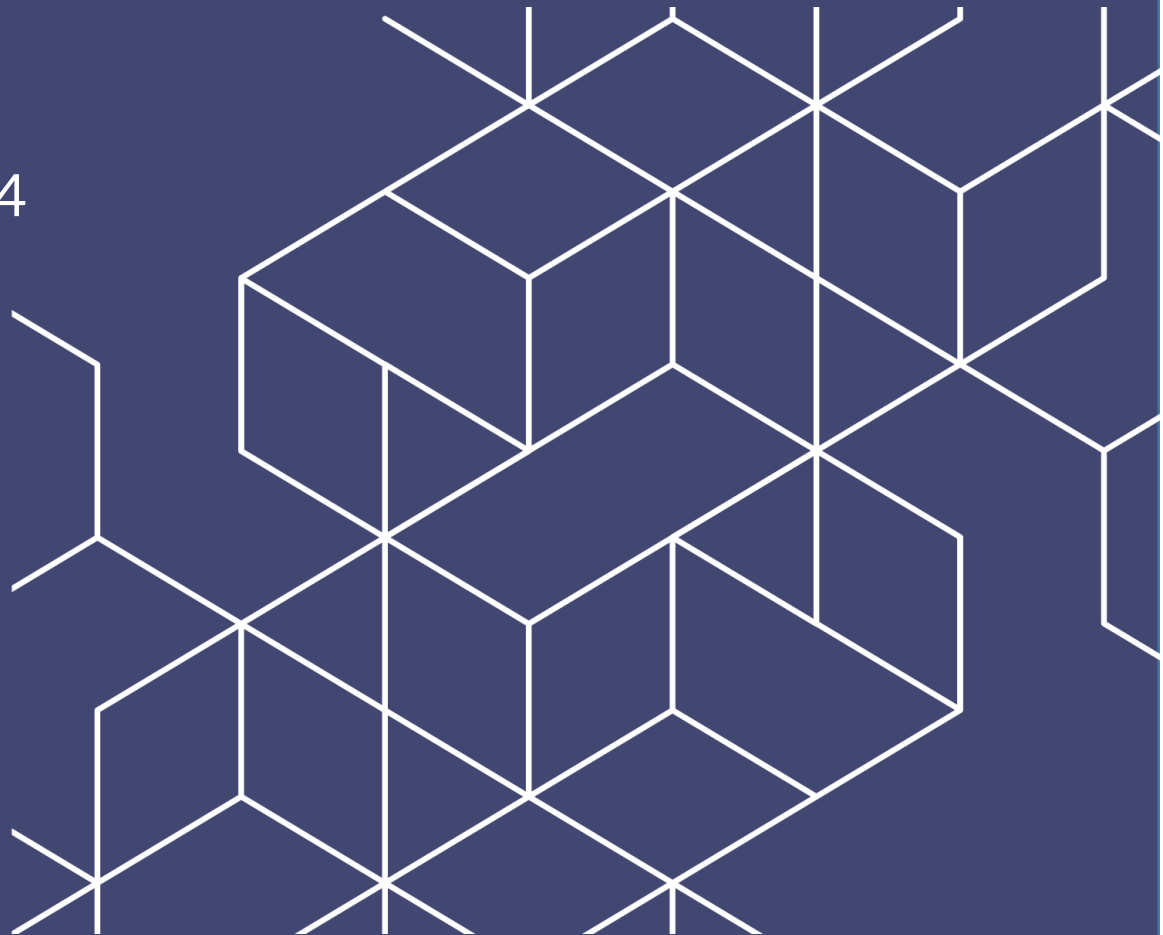
eC CASE



Office 365 OAuth Configuration

v11.3.0

May 2024



eCASE 11.3.0 Office 365 OAuth Configuration

Notice of Rights

Copyright © 2024, AINS, LLC d/b/a OPEXUS. All rights reserved. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of the publisher: AINS, LLC. For information on obtaining permission for reprints and excerpts, contact info@opexustech.com.

Additionally, all copyrights, confidential information, patents, design rights and all other intellectual property rights of whatsoever nature contained herein are, and shall remain, the sole and exclusive property of the publisher.

Notice of Liability

The information in this publication is believed to be accurate and reliable. However, the information is distributed by the publisher (AINS, LLC.) on an “As Is” basis without warranty for its use, or for any infringements of patents or other rights of third parties resulting from its use.

While every precaution has been taken in the preparation of this publication, neither the author (or authors) nor the publisher will have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused, directly or indirectly, by the information contained in this publication or by the computer software and hardware products described in it.

Notice of Trademarks

The publisher’s company name, company logo, company patents, and company proprietary products are trademarks or registered trademarks of the publisher: AINS, LLC. All other trademarks or registered trademarks are the property of their respective owners.

Non-Disclosure Statement

This document’s contents are confidential and proprietary to AINS, LLC. This document cannot be released publicly or outside the purchasing agency without prior written permission from AINS, LLC.

Images in this manual are used as examples and may contain data and versioning that may not be consistent with your version of the application or information in your environment.

Additional Notice

Information in this documentation is subject to change without notice and does not represent a commitment on the part of AINS, LLC.

Notwithstanding any of the foregoing, if this document was produced as a Deliverable or other work for hire under a contract on behalf of a U.S. Government end user, the terms and conditions of that contract shall apply in the event of a conflict.



Contents

- 1 About Office 365 OAuth Configuration 4
 - 1.1 In This Manual 4
 - 1.2 Prerequisites 4
- 2 Register Application 5
- 3 Create Client Secret 7
- 4 API Permissions..... 9



1 About Office 365 OAuth Configuration

1.1 In This Manual

This manual contains steps to configure eCASE to integrate with OAuth for sending emails from a system account. The steps to complete this configuration take place in three parts:

- Register Application: Register your application through the Azure Portal.
- Create Client Secret: Using the Azure Portal, create a Client Secret you will use for OAuth configuration.
- API Permissions: Add appropriate API permissions to permit sending emails.

1.2 Prerequisites

The following are prerequisites for completing OAuth configuration:

- OAuth configuration should be completed by a system administrator with the appropriate knowledge and access to complete all required steps.
- Before beginning the configuration, you must create an Exchange mailbox in Office 365 (ex. noreply@opexustech.com). This email appears as the sender for all system messages from eCASE and is used to complete the configuration.



2 Register Application

The first step in OAuth configuration is to register your app. Follow the steps below to register the app in Azure:

1. Log in to portal.azure.com using the Exchange mailbox created as the eCASE system account.
2. Click **App Registrations > New Registration**. The *Register an application* screen appears:

Microsoft Azure Search resources, services, and docs (G+)

Home > App registrations > Register an application

* Name **A**

The user-facing display name for this application (this can be changed later).

Supported account types **B**

Who can use this application or access this API?

☒ Accounts in this organizational directory only (Ains, Inc only - Single tenant)

☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant)

☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

☐ Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional) **C**

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Select a platform e.g. https://example.com/auth

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from [Enterprise applications](#).

By proceeding, you agree to the [Microsoft Platform Policies](#)

Register

3. Enter a **(A) Name** for the app in the field provided.
4. Under **(B) Supported account types**, the top option is selected by default. You may need to make a different selection depending on your organization's needs.
5. Under **(C) Redirect URL**, enter "<Application Admin URL>/connectors/SMTP.aspx" where <Application URL> is replaced with your application's admin URL.
6. Click **Register** to register the app.
7. The app is registered. The screen displays the **(A) Application (client) ID** and **(B) Directory (tenant) ID**. Copy both to your clipboard or otherwise save for later reference:



Register Application

Home > App registrations > OAUTH Application

Search (Ctrl+/) Delete Endpoints Preview features

Overview

Quickstart

Integration assistant

Manage

Branding & properties

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

Essentials

Display name
OAUTH Application

Application (client) ID
94206c5b-35dc-48d4-980e-a14db1eafb45

Object ID
edf2ca10-7fff-4fde-a520-21fb070e7100

Directory (tenant) ID
2e6c73c6-1977-4164-8037-33ca7fa29664

Supported account types
My organization only

Client credentials

Add a certificate or secret

Redirect URIs

1 web, 0 spa, 0 public client

Application ID URI

Add an Application ID URI

Managed application in local directory

OAUTH Application

8. Access eCASE Administration (eCASE > Settings > Connectors > eMail (SMTP)). The configuration screen appears as shown below:

Save Close Note: * fields are mandatory when enabled

Enable/Disable: ☒

Authentication Mode: * OAuth

User Name: *

OAuth Client ID: * A

Public Key (secret):

Tenant ID: B

Use SSL: No

Recipient Limit: 0

9. Under (A) OAuth Client ID, enter the Application (client) ID from step 7.
10. Under (B) Tenant ID, enter the Directory (tenant) ID from step 7.
11. In the User Name field, enter the email address being used as the system account for this configuration (ex. noreply@ains.com).
12. Click Save to save the changes.



3 Create Client Secret

Next, you'll follow the steps below to create a new Client Secret:

1. Within the Azure Portal, access your application, then access the *Certificates & Secrets* screen.
2. Click **New Client Secret**:

Home > App registrations > OAUTH Application

OAUTH Application | Certificates & secrets

cer Got feedback?

Manage

Certificates & secrets

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) **Client secrets (0)** Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value ⓘ	Secret ID
No client secrets have been created for this application.			

3. The *Add a client secret* screen appears. First enter a *Description* in the field provided. This is an internal description that is visible only to Admin users:

Add a client secret

Description

Expires

Add **Cancel**

4. Use the *Expires* field to determine an expiration date based on your organization's preference (with a 24-month maximum).

(!!) Note: Take a note of this expiration date, as this Client Secret will need to be renewed prior to the expiration for continuous operation.

5. Click **Add** to generate the Client Secret.
6. The Client Secret is successfully generated, and the secret appears as shown in the example below:



Create Client Secret

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) **Client secrets (1)** Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value ⓘ	Secret ID
New client secret	2/18/2023	hoZ8Q~tKQUtNjHsFMsUp4GX0.uK~Odz... ⓘ	f627ab0e-4bb8-4894-92df-92f09f168577 ⓘ

(!!) Note: Save the “Value” as this cannot be retrieved. You will need this to complete the configuration.

- Copy the *Value* field to your clipboard.
- Access eCASE Administration (**eCASE > Settings > Connectors > eMail (SMTP)**).
- Copy the value obtained in step 7 into the Public Key (secret) field:

Save | Close Note: * fields are mandatory when enabled

Enable/Disable:	<input checked="" type="checkbox"/>
Authentication Mode: *	OAuth ▾
User Name: *	<input type="text"/>
OAuth Client ID: *	<input type="text"/>
Public Key (secret):	<input type="text"/>
Tenant ID:	<input type="text"/>
Use SSL:	No ▾
Recipient Limit:	<input type="text" value="0"/>

- Click **Save** to save the changes.



4 API Permissions

The final step to enable OAuth is configuring API permissions:

1. Open the Azure Portal and access your application page, then click **API permissions**:

All services > App registrations > OAUTH Application

OAUTH Application | API permissions

Search (Ctrl+/) Refresh Got feedback?

Overview
Quickstart
Integration assistant
Manage
Branding & properties
Authentication
Certificates & secrets
Token configuration
API permissions
Expose an API
App roles
Owners
Roles and administrators
Manifest
Support + Troubleshooting

The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect the value in your organization, or in organizations where this app will be used. [Learn more](#)

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

+ Add a permission Grant admin consent for Ains, Inc

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (1)				
User.Read	Delegated	Sign in and read user profile	No	...

To view and manage permissions and user consent, try [Enterprise applications](#).


2. From the API Permissions screen click **Add a Permission**.
3. The *Request API Permissions* screen appears. Click **Microsoft Graph**:

Request API permissions


Select an API

Microsoft APIs APIs my organization uses My APIs


Commonly used Microsoft APIs

 **Microsoft Graph**


Take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, and Windows 10. Access Azure AD, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner, and more through a single endpoint.

 **Azure Batch**

Schedule large-scale parallel and HPC applications in the cloud

 **Azure Communication Services**

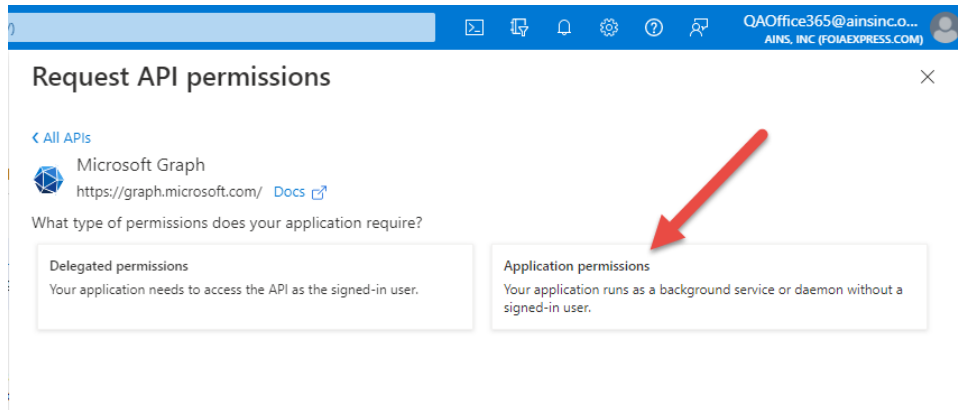
Rich communication experiences with the same secure CPaaS platform used by Microsoft Teams

 **Azure Cosmos DB**

Fast NoSQL database with open APIs for any scale.

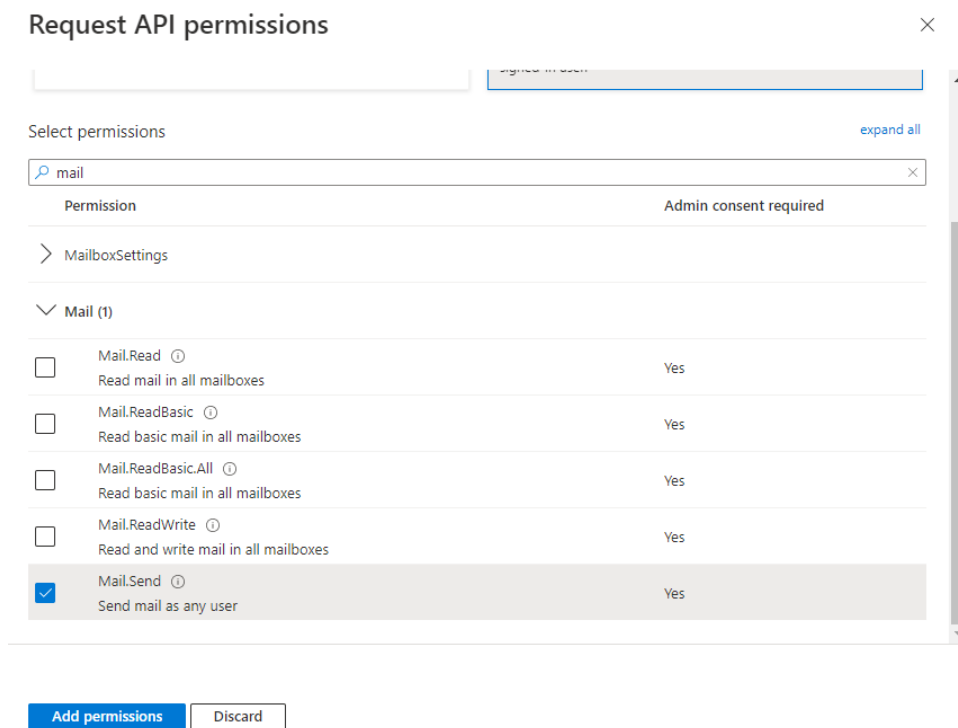


4. Next click **Application Permissions**.



5. In the Select Permissions field, type "mail".

6. Locate and expand the Mail permissions, then select **Mail.Send**:



7. Click **Add Permissions** to apply the selected permission.

8. The Administrator must grant these permissions. The Admin receives a notification to grant the requested permission and, once this permission is granted, the mailbox can send mail from the system account.

