Office 365 SMTP OAuth2 Setup Guide

Client Credentials Flow with Certificate Authentication

For use with Limagito File Mover or any application requiring app-only SMTP access

Overview

This guide describes all steps required to configure Office 365 SMTP with OAuth2 client credentials flow (app-only, no user interaction required). The application authenticates using a certificate (.pfx) instead of a client secret. This setup requires configuration in both the Azure AD portal and Exchange Online PowerShell.

Required Information

Item	Value / Description
Tenant ID	Found in Azure AD -> Overview
App (Client) ID	Azure AD -> App Registrations -> your app -> Overview
Object ID (Enterprise App)	Azure AD -> Enterprise Applications -> your app -> Overview
Certificate (.pfx)	Private key file on the application server
Mailbox address	The email address the application will send from

---

Part 1 — Azure AD Portal Configuration

Step 1: Add SMTP.SendAsApp Permission

1. Go to **Azure AD -> App Registrations -> [your app] -> API Permissions**
2. Click **Add a permission**
3. Select the **APIs my organization uses** tab
4. Search for **Office 365 Exchange Online**
5. Select **Application permissions** (NOT Delegated!)
6. Check **SMTP.SendAsApp**
7. Click **Add permissions**

Step 2: Grant Admin Consent

Click **Grant admin consent for [tenant name]** at the top of the permissions page. The status must turn green: **Granted**.

WARNING: Without admin consent, SMTP.SendAsApp will NOT work, even if everything else is configured correctly.

Step 3: Link Certificate to App Registration

Go to **App Registrations -> [your app] -> Certificates & secrets** and upload the public key (.cer file) of the certificate. The application uses the private key (.pfx) to sign the token request. Azure AD uses the public key to

verify it.

Part 2 — Exchange Online PowerShell Configuration

> The following steps CANNOT be performed via the Azure AD portal. Exchange Online maintains its own internal directory of service principals which must be configured separately via PowerShell.

Step 4: Install and Connect Exchange Online Module

Run PowerShell as Administrator for the Install-Module command. Each command must be run separately:

```
Install-Module -Name ExchangeOnlineManagement -Force -AllowClobber -Scope AllUsers
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser
Import-Module ExchangeOnlineManagement
Connect-ExchangeOnline
```

Step 5: Register the App as a Service Principal in Exchange Online

This is a mandatory step — Exchange Online does not automatically recognize the Azure AD app for SMTP client credentials flow. The service principal must be created manually. Replace the GUIDs with your own values.

```
# AppId     = Client ID from the Azure AD App Registration
# ServiceId = Object ID from Enterprise Applications in Azure AD
#             (NOT the Object ID from App Registrations!)

New-ServicePrincipal `
    -AppId "your-app-client-id" `
    -ServiceId "your-enterprise-app-object-id" `
    -DisplayName "YourAppDisplayName"

# Verify
Get-ServicePrincipal | Where-Object {$_.AppId -eq "your-app-client-id"} |
    Format-List DisplayName, AppId, IsValid
```

Step 6: Grant FullAccess and SendAs Permissions on the Mailbox

Both permissions are required: **SendAs** to allow sending from the mailbox, **FullAccess** to allow opening the mailbox store. Omitting either will cause authentication failures.

```
# FullAccess - required to open the mailbox store
Add-MailboxPermission `
    -Identity "mailbox@yourdomain.onmicrosoft.com" `
    -User "your-enterprise-app-object-id" `
    -AccessRights FullAccess `
    -AutoMapping $false `
    -Confirm:$false

# SendAs - required to send as that mailbox address
Add-RecipientPermission `
    -Identity "mailbox@yourdomain.onmicrosoft.com" `
    -Trustee "YourAppDisplayName" `
    -AccessRights SendAs `
    -Confirm:$false

# Verify
Get-MailboxPermission -Identity "mailbox@yourdomain.onmicrosoft.com"
Get-RecipientPermission -Identity "mailbox@yourdomain.onmicrosoft.com"
```

Step 7: Enable SMTP AUTH on the Mailbox

```
# $false means SMTP AUTH is ENABLED (confusing naming by Microsoft!)
```

```
Set-CASMailbox `
    -Identity "mailbox@yourdomain.onmicrosoft.com" `
    -SmtpClientAuthenticationDisabled $false

# Verify - must return False (= SMTP AUTH is enabled)
Get-CASMailbox -Identity "mailbox@yourdomain.onmicrosoft.com" |
    Select SmtpClientAuthenticationDisabled
```

Note: This per-mailbox setting is not visible in the Exchange Admin Center UI. It can only be viewed and changed via PowerShell.

Part 3 — Application Configuration (Limagito File Mover)

OAuth2 Tab Settings

Field	Value
Use OAuth 2.0 authorization	Checked
Send Code Challenge (RFC 7636)	UNCHECKED (client credentials does not use PKCE)
Authorization Endpoint URL	Leave empty (not used for client credentials)
Token Endpoint URL	https://login.microsoftonline.com/{TenantID}/oauth2/v2.0/token
Client ID	App Registration Client ID
Client Secret	Leave empty (certificate is used instead)
Scope	https://outlook.office365.com/.default
Redirect URI	http://localhost

Auth Options Tab Settings

Field	Value
Private Key File (.pfx)	Full path to the .pfx file on the server
Private Key Password	Password of the .pfx file

Email / From Address

WARNING: The **From address** must exactly match the SMTP authentication account. If you authenticate as mailbox@yourdomain.onmicrosoft.com, the From address must also be mailbox@yourdomain.onmicrosoft.com. A mismatch will cause a 554 SendAsDenied error.

Troubleshooting — Common Errors

535 5.7.139 — SmtpClientAuthentication is disabled for the Tenant

-> SMTP AUTH is disabled at tenant or mailbox level. Run Step 7 (Set-CASMailbox). Also check Mail flow settings in Exchange Admin Center: 'Turn off SMTP AUTH protocol' must be unchecked.

535 5.7.3 — Authentication unsuccessful

-> Possible causes: (1) Admin consent not granted for SMTP.SendAsApp — check Azure AD permissions, (2) Exchange Service Principal not created — run Step 5 (New-ServicePrincipal), (3) Propagation delay — wait 5-15 minutes after making changes before retrying.

554 5.2.252 — SendAsDenied

-> The From address in the email does not match the authenticated account. Update the From/Sender address to match the SMTP username exactly.

430 4.2.0 — Mailbox logon failure / Authentication Context has no rights

-> FullAccess permission is missing. Run Add-MailboxPermission (Step 6). Both FullAccess AND SendAs are required.

ExchangeOnlineManagement module errors

-> Run: Install-Module ExchangeOnlineManagement -Force -AllowClobber, then Import-Module ExchangeOnlineManagement -Force, then Connect-ExchangeOnline.

Note: Exchange Online permission changes can take 5-15 minutes to propagate. After running the PowerShell steps, wait a few minutes before testing the SMTP connection. This setup has been tested and confirmed working with Limagito File Mover using Chilkat SMTP library. Use of this information is at your own risk.