Create application access token & authorization header

This article explains how to authenticate to Microsoft Graph using **application permissions**. Use this method for unattended automation where no signed-in user is involved.

When to use application authentication

Use app-only authentication when:

- the script runs unattended
- the process runs in Azure Automation, Functions, pipelines, or services
- no user interaction is possible
- the automation should act as the application itself

Preferred options in enterprise environments:

1. **Managed Identity** for Azure-hosted workloads
2. **Certificate-based authentication**
3. **Client secret** only when the first two are not possible

> Note The OAuth 2.0 client credentials flow does **not** return a refresh token. To get a new token, request a new access token again.

Prerequisites

- Microsoft Entra ID app registration
- Microsoft Graph **application permissions**
- admin consent granted
- tenant ID
- client ID
- client credential (certificate or secret)

Example: client secret flow

```
$TenantId     = "<tenant-id>"
$ClientId     = "<app-client-id>"
$ClientSecret = "<client-secret>"

$TokenBody = @{
    grant_type    = "client_credentials"
    scope         = "https://graph.microsoft.com/.default"
    client_id     = $ClientId
    client_secret = $ClientSecret
}

$TokenResponse = Invoke-RestMethod `
    -Method POST `
    -Uri "https://login.microsoftonline.com/$TenantId/oauth2/v2.0/token" `
    -Body $TokenBody `
    -ContentType "application/x-www-form-urlencoded"

$Header = @{
    Authorization = "Bearer $($TokenResponse.access_token)"
    "Content-Type" = "application/json"
}
```

Example: Managed Identity in Azure

```
Connect-MgGraph -Identity
Get-MgContext
```

Best practices

- use **least privilege**
- prefer **Managed Identity** over secrets
- prefer **certificates** over client secrets when Managed Identity is not possible
- store secrets in **Key Vault**
- review Graph app permissions regularly

Common mistakes

- expecting a **refresh token** in client credentials flow
- using app-only auth when a user context is actually required
- hardcoding secrets in scripts
- forgetting admin consent for Graph application permissions

Summary

For unattended Microsoft Graph automation, use **application permissions** and prefer **Managed Identity** whenever possible.

Revision #21
Created 2022-12-07 22:07:13 UTC
Updated 2026-04-15 21:31:03 UTC by Caprez-OpenClaw02