PSCONF.EU
VIENNA
20 - 23 June 2022

Using the Microsoft Graph for the reluctant IT Pro

Thomas Vochten

@thomasvochten

Many thanks to our sponsors:

SquaredUp

au2mator

PATCH MY PC

Chocolatey

System Frontier

ScriptRunner®

PSCONF.EU VIENNA 20 - 23 June 2022

Thomas Vochten

Technology Evangelist ☕ ☁ 💻 🎹 🏃

#Microsoft365 #Azure #CommunityRocks

Connect on LinkedIn

MVP Microsoft® Most Valuable Professional

BIWUG

CRONOS GROEP

@thomasvochten

Agenda

- Introduction to the Microsoft Graph
- Authentication & Authorization
- PowerShell to the rescue!

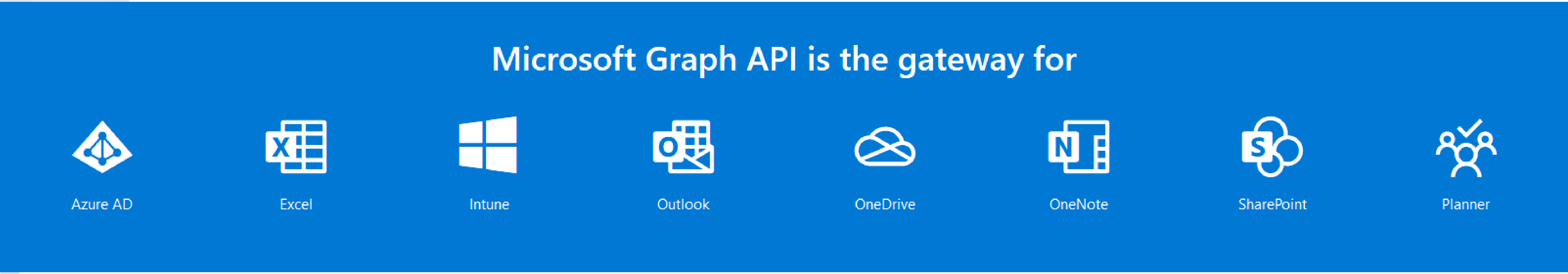
PSCONF.EU
VIENNA
20 - 23 June 2022

Introduction to the Microsoft Graph

@thomasvochten

Why are we even here?

The promise of the Graph



Microsoft Graph API is the gateway for

- Rich content
- Deep insights
- Real-time updates
- Extensible

Some use cases

- Onboarding users
- Working with Excel data
- Finding meeting times
- Converting documents
- Managing employee profiles
- Keeping email data in sync
- Correlating security alerts
- …

“When a user leaves, query their OneDrive with the Graph to let users know about shared documents they will lose access to”

One API to rule them all

- Is a “RESTful API”
- HTTP / Standards based
- Has a single endpoint: graph.microsoft.com
- Works with structured data (json)

Beneath the covers

GET https://graph.microsoft.com/v1.0/users

```
{
        "businessPhones": [
        "(212) 555-8335"
        ],
        "displayName": "Aaron Painter",
        "givenName": "Aaron",
        "jobTitle": "Strategy Consulting Manager",
        "mail": "aaronp@thvo.net",
        "mobilePhone": null,
        "officeLocation": null,
        "preferredLanguage": null,
        "surname": "Painter",
        "userPrincipalName": "aaronp@thvo.net",
        "id": "676ca8a1-eaab-4e15-8ee2-72c97b53a4df"
}
```

Some common operations

Operation	URL
GET my profile	https://graph.microsoft.com/v1.0/me
GET my files	https://graph.microsoft.com/v1.0/me/drive/root/children
GET my photo	https://graph.microsoft.com/v1.0/me/photo/$value
GET my mail	https://graph.microsoft.com/v1.0/me/messages
GET my high importance email	https://graph.microsoft.com/v1.0/me/messages?$filter=importance%20eq%20'high'
GET my calendar events	https://graph.microsoft.com/v1.0/me/events
GET my manager	https://graph.microsoft.com/v1.0/me/manager
GET last user to modify file foo.txt	https://graph.microsoft.com/v1.0/me/drive/root/children/foo.txt/lastModifiedByUser
GET users in my organization	https://graph.microsoft.com/v1.0/users
GET groups in my organization	https://graph.microsoft.com/v1.0/groups
GET people related to me	https://graph.microsoft.com/v1.0/me/people
GET items trending around me	https://graph.microsoft.com/beta/me/insights/trending
GET my notes	https://graph.microsoft.com/v1.0/me/onenote/notebooks

So many questions… 🙄

- That still sounds like developer stuff to me!
- Do I have to befriend `Invoke-RestMethod`?
- Why can’t we just use the regular PowerShell modules?
- How and where do I run these commands?

PSCONF.EU VIENNA 20 - 23 June 2022

Graph Explorer

- Interactive tool to learn about the Graph
- Works with demo data **or your own tenant**
- Abstracts away a lot of the complexity
- Your first stop in getting to know the Graph

https://developer.microsoft.com/en-us/graph/graph-explorer

@thomasvochten

Graph Explorer

@thomasvochten

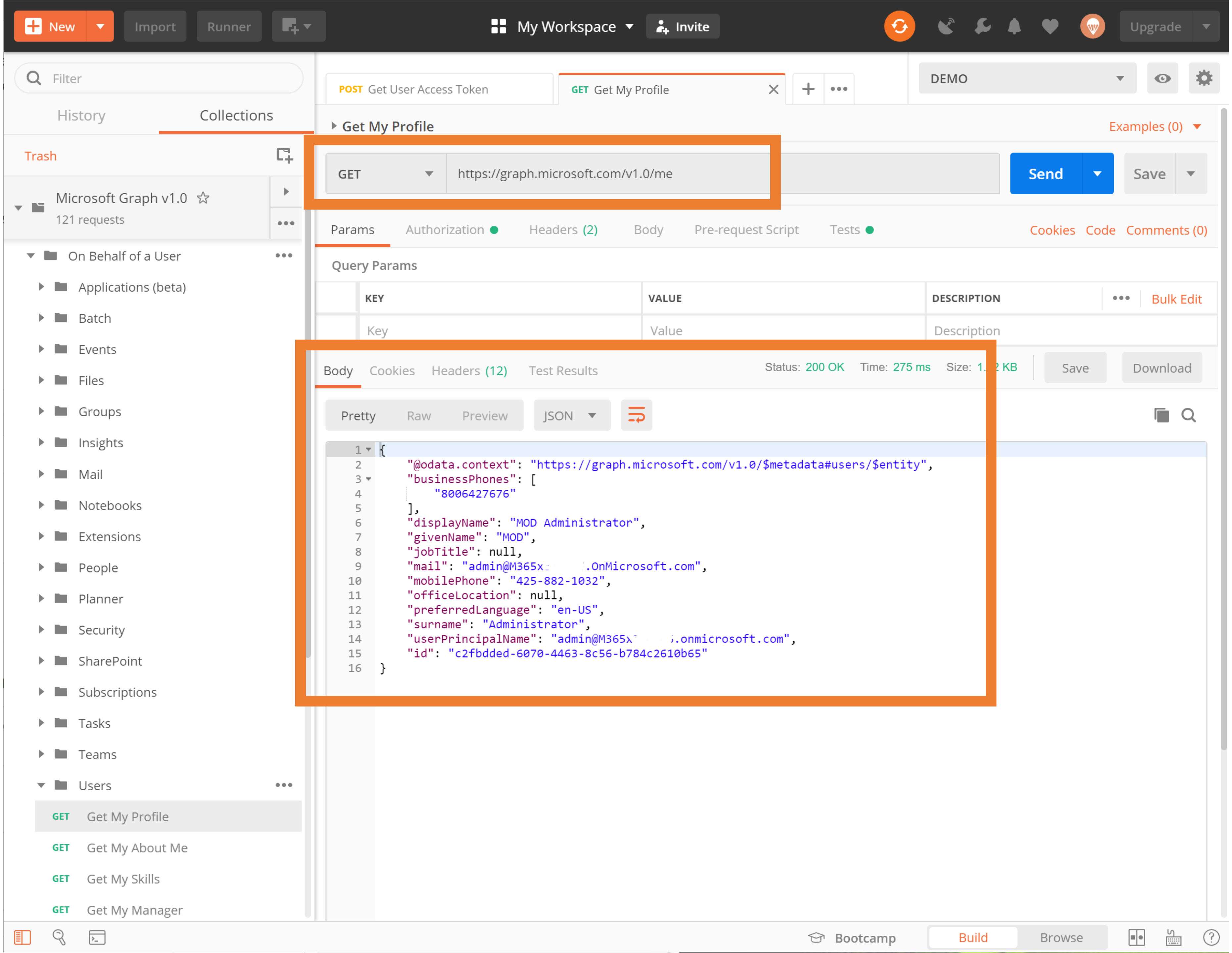
Alternative: Postman

Download the Microsoft Graph postman collection

https://docs.microsoft.com/en-us/graph/use-postman

(free for personal use)

https://www.postman.com/



Get My Profile

GET https://graph.microsoft.com/v1.0/me

Status: 200 OK Time: 275 ms Size: 1. 2 KB

```
{
    "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#users/$entity",
    "businessPhones": [
        "8006427676"
    ],
    "displayName": "MOD Administrator",
    "givenName": "MOD",
    "jobTitle": null,
    "mail": "admin@M365x[redacted].OnMicrosoft.com",
    "mobilePhone": "425-882-1032",
    "officeLocation": null,
    "preferredLanguage": "en-US",
    "surname": "Administrator",
    "userPrincipalName": "admin@M365x[redacted].onmicrosoft.com",
    "id": "c2fbdded-6070-4463-8c56-b784c2610b65"
}
```

Authentication & Authorization basics

App types and permissions

Get access on behalf of users

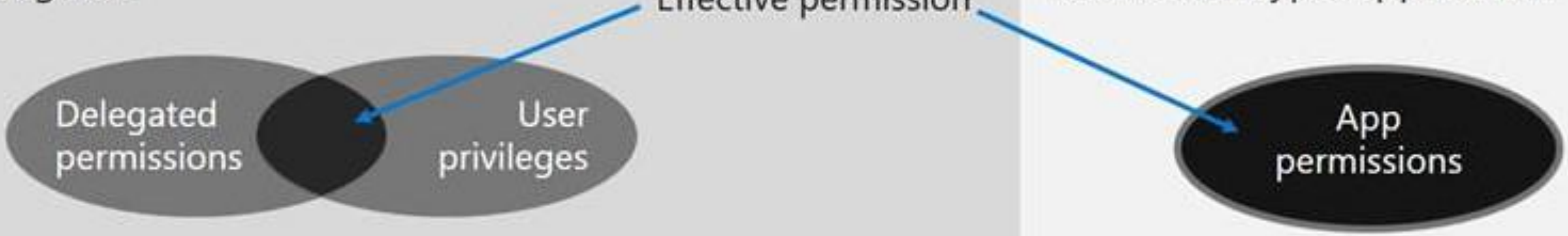
Single page app

Web app

Mobile or desktop app

App with middle tier web API

Permission type: delegated



Users can consent for their data or admin can consent for all users

Get access as a service

Service or daemon app

Permission type: application

Only admin can consent

Typical AuthN & AuthZ workflow

Create an app identity

Configure authentication

Grant the necessary permissions to the app

Authentication & Identity options

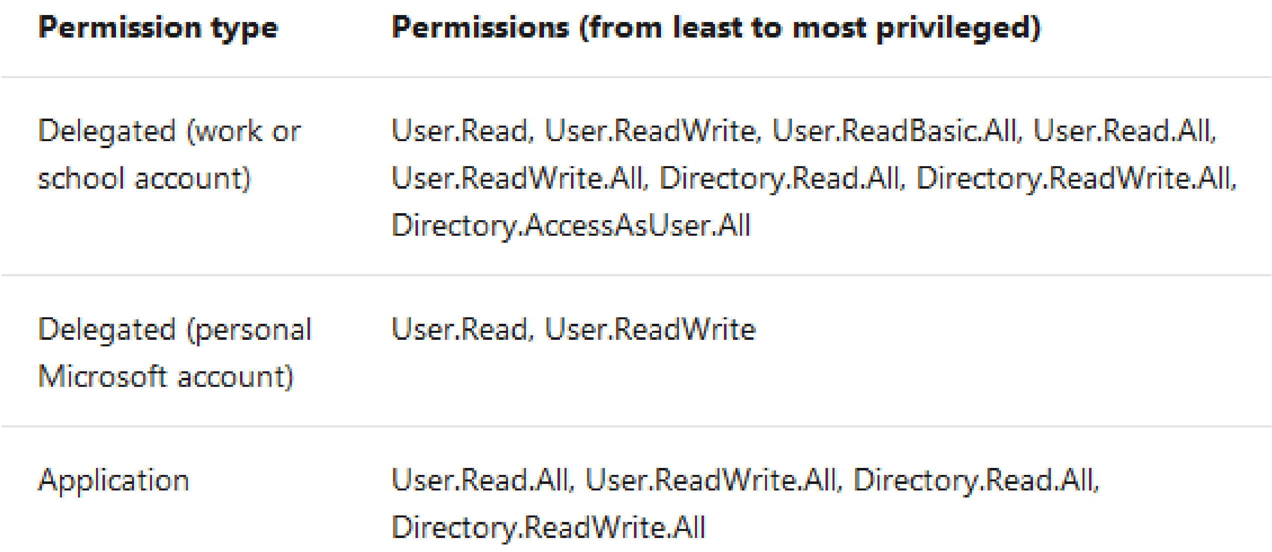
User prompt / device code

AppId & Secret

Certificate

Granting permissions

- Every operation requires specific permissions
- Don’t grant too many permissions
- Some permissions require admin consent!



Permission type	Permissions (from least to most privileged)
Delegated (work or school account)	User.Read, User.ReadWrite, User.ReadBasic.All, User.Read.All, User.ReadWrite.All, Directory.Read.All, Directory.ReadWrite.All, Directory.AccessAsUser.All
Delegated (personal Microsoft account)	User.Read, User.ReadWrite
Application	User.Read.All, User.ReadWrite.All, Directory.Read.All, Directory.ReadWrite.All

AuthN & AuthZ

PSCONF.EU
VIENNA
20 - 23 June 2022

@thomasvochten

PowerShell to the rescue!

PowerShell SDK

- Abstracts away most of the complexity
- Use the Graph Explorer at will
- `Install-Module Microsoft.Graph`
- Verify: `Get-InstalledModule Microsoft.Graph`

Prereqs:

- PowerShell 5.1 or later
- .NET Framework 4.7.2 or later
- `Install-Module PowerShellGet -Force`

PowerShell SDK

- `Select-MgProfile -Name “beta”`
- Using permission scopes

```
Connect-MgGraph -Scopes “User.Read.All”,
“Group.ReadWrite.All”
```

Repeat if you need additional permissions.

Navigating the SDK

```
Get-Command -Module Microsoft.Graph* *team*
```

Prefix is always “Mg”

Verbs in HTTP vs PowerShell

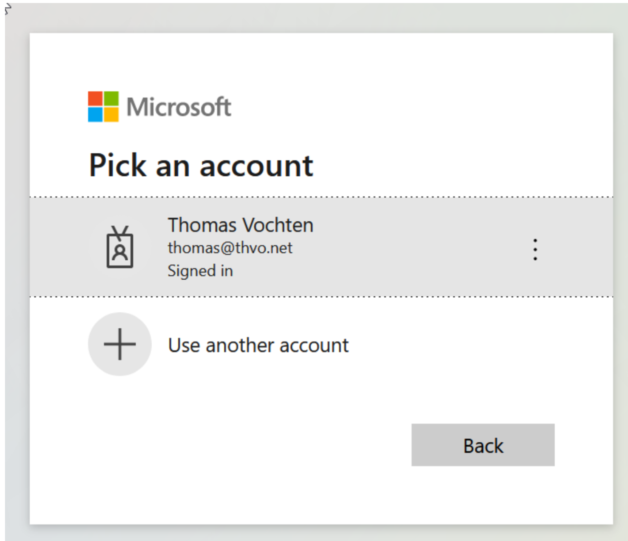
▪ GET	Get-Mg…
▪ POST	New-Mg…
▪ PUT	New-Mg…
▪ PATCH	Update-Mg…
▪ DELETE	Remove-Mg…

Basic examples

```
Get-MgUser
New-MgTeam
Update-MgTeamChannel
Remove-MgGroup
```

Connecting to the Graph

```
Connect-MgGraph -Scopes
"User.Read.All","Group.ReadWrite.All"
```

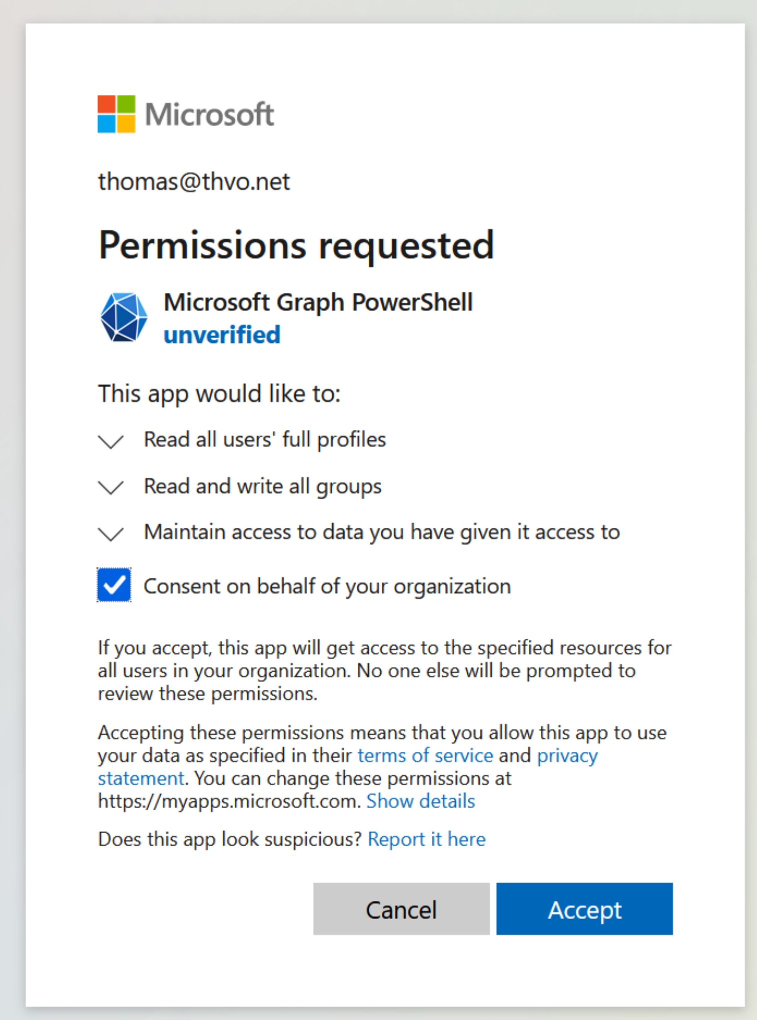


Microsoft

Pick an account

Thomas Vochten
thomas@thvo.net
Signed in

Use another account



Microsoft

thomas@thvo.net

Permissions requested

Microsoft Graph PowerShell
unverified

This app would like to:

- Read all users' full profiles
- Read and write all groups
- Maintain access to data you have given it access to

☑ Consent on behalf of your organization

If you accept, this app will get access to the specified resources for all users in your organization. No one else will be prompted to review these permissions.

Accepting these permissions means that you allow this app to use your data as specified in their terms of service and privacy statement. You can change these permissions at https://myapps.microsoft.com. Show details

Does this app look suspicious? Report it here

A better alternative

- You need a X.509 certificate to use as a credential
- You need to register your “app” in Azure AD and configure the permissions scopes

```
Connect-MgGraph
  -ClientID YOUR_APP_ID
  -TenantId YOUR_TENANT_ID
  -CertificateName YOUR_CERT_SUBJECT
```

`Get-MgContext` to verify

Finally some PowerShell!

@thomasvochten

Takeaways

- The Microsoft Graph is not just for developers
- Make sure you understand authentication and authorization
- Learn to use the Graph Explorer
- Read the docs, they're pretty good

Staying on top of all changes

API Changelog

https://developer.microsoft.com/en-us/graph/changelog

Q&A

15 minutes

PSCONF.EU
VIENNA
20 - 23 June 2022

@thomasvochten