

Managing Microsoft 365 through the command-line

@thomasvochten

About me



Thomas Vochten



Technology evangelist Microsoft 365 & Security

@thomasvochten

<https://thomasvochten.com>

mail@thomasvochten.com



Microsoft®
Most Valuable
Professional

BIVVUG



Agenda

1. Managing Microsoft 365 in a nutshell
2. Getting to know the Microsoft Graph
3. PowerShell to the rescue?
4. The wonderful world of the CLI
5. Automating all the things



Managing Microsoft 365 in a nutshell

@thomasvochten

Anyone need a portal?

- `admin.microsoft.com`
- `security.microsoft.com`
- `compliance.microsoft.com`
- `entra.microsoft.com`
- `intune.microsoft.com`
- ...

Advanced deployment guides

<https://setup.cloud.microsoft>

● Overview

○ Windows Hello for Business

○ Finish

About Passwordless authentication

Passwordless authentication is an alternative sign-in approach that allows users to access their devices securely with one of the following methods:

- Windows Hello for Business
- Microsoft Authenticator app
- Security keys

To learn more about the benefits of passwordless authentication and planning your deployment, see [Plan a passwordless authentication deployment in Microsoft Entra ID](#).

① Many customers don't enable passwordless authentication because they think pins are less secure, but a username and password remains a weak form of authentication that can be exposed or brute-force attacked.

Status

Microsoft Authenticator app	FIDO2
● On	● On
Temporary Access Pass	
● On	



[cmd.ms]

Home

Microsoft Portals

Azure

Azure Active Directory

Defender

Intune

Microsoft 365

My Pages

Docs

Power User Tips

Contributing

Related

Home

Home

Home

[cmd.ms] the Microsoft Cloud command line!

Are you tired of clicking around in Microsoft portals to get to a blade?

Use the power of your browser's address bar to quickly get to your favorite blade in Azure, Microsoft 365, Azure AD, Intune...

Try it out. Open a new tab and type **{command}.cmd.ms** using any of the commands or alias from the list below.

Remember to [check out the browser extensions](#) that add support for auto-complete in the address bar.

All Microsoft Portals

Search commands...

	NAME	COMMAND	ALIAS
	Azure Active Directory (Entra)	aad.cmd.ms	aad,entra
	Azure Active Directory	azad.cmd.ms	
	Microsoft 365 Admin	admin.cmd.ms	
	Azure	az.cmd.ms	azure
	Azure Preview Portal	azp.cmd.ms	azurepreview

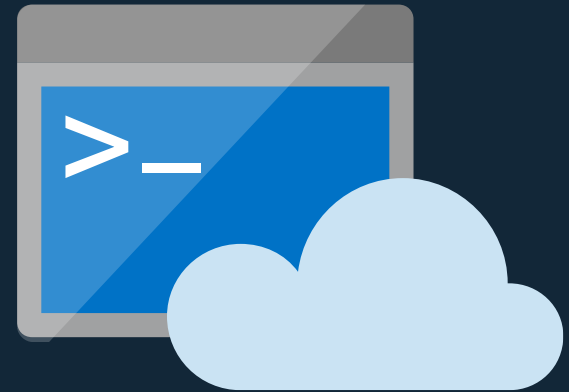
Tools & Terminals



Windows Terminal



Visual Studio Code



Azure Cloud Shell

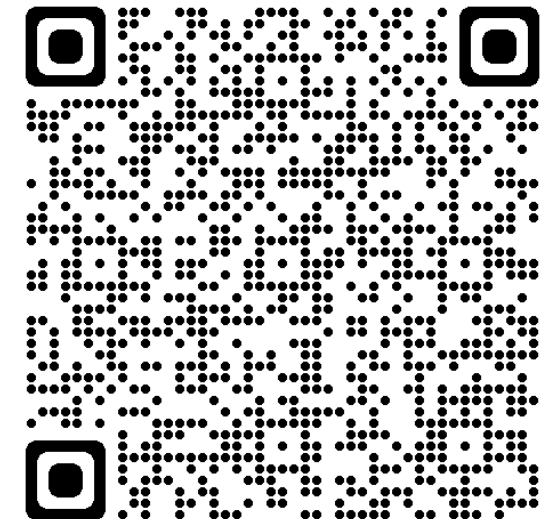

```
PowerShell
~\Dev\cli-microsoft365 main dir

Directory: C:\Users\ThomasVochten\Dev\cli-microsoft365

Mode                LastWriteTime         Length Name
----                -
d----            20/05/2024    08:38             .devcontainer
d----            20/05/2024    08:38             .github
d----            20/05/2024    08:38             .vscode
d----            20/05/2024    08:38             docs
d----            20/05/2024    08:38             eslint-rules
d----            20/05/2024    08:38             scripts
d----            20/05/2024    08:38             src
-a---            20/05/2024    08:38          320 {} .c8rc.json
-a---            20/05/2024    08:38         4887 {} .eslinttrc.cjs
-a---            20/05/2024    08:38         1300 {} .gitignore
-a---            20/05/2024    08:38          222 {} .mocharc.json
-a---            20/05/2024    08:38          302 {} .npmignore
-a---            20/05/2024    08:38          451 {} CODE_OF_CONDUCT.md
-a---            20/05/2024    08:38         3524 {} CONTRIBUTING.md
-a---            20/05/2024    08:38          492 {} tsconfig.json

~\Dev\cli-microsoft365 main _
```

Upgrade your command prompt



Cloud Shell

- Authenticated shell when you need it
- Needs to be deployed to an Azure subscription
- Comes preloaded with the most common tools

```
PowerShell ? [?] {} [?]
core 2.53.0 *
telemetry 1.1.0

Extensions:
ai-examples 0.2.5
ml 2.21.1
ssh 2.0.2

Dependencies:
msal 1.24.0b2
azure-mgmt-resource 23.1.0b2

Python location '/usr/bin/python3.9'
Extensions directory '/home/thomas/.azure/cliextensions'
Extensions system directory '/usr/lib/python3.9/site-packages/azure-cli-extensions'

Python (Linux) 3.9.14 (main, Oct 12 2023, 19:48:32)
[GCC 11.2.0]
```



Getting to know the Microsoft Graph

@thomasvochten

Microsoft 365 Platform



Extend Microsoft 365 experiences



Documents



Conversations



Portals



Timeline



Search



Build your experience



Web
apps



Bots &
agents



Device
& native



Daemon
apps



Workflow
automation



Analytics
apps



Microsoft Graph
REST APIs and webhooks

External data



Connectors



Microsoft Graph

Microsoft 365

Windows 10

Enterprise Mobility + Security

Azure AI platform



Microsoft Graph
Data Connect

Microsoft Identity

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"
}
```

@thomasvochten

Some common operations

Operation

GET my profile

GET my files

GET my photo

GET my mail

GET my high importance email

GET my calendar events

GET my manager

GET last user to modify file foo.txt

GET users in my organization

GET groups in my organization

GET people related to me

GET items trending around me

GET my notes

URL

<https://graph.microsoft.com/v1.0/me>

<https://graph.microsoft.com/v1.0/me/drive/root/children>

[https://graph.microsoft.com/v1.0/me/photo/\\$value](https://graph.microsoft.com/v1.0/me/photo/$value)

<https://graph.microsoft.com/v1.0/me/messages>

[https://graph.microsoft.com/v1.0/me/messages?\\$filter=importance%20eq%20'high'](https://graph.microsoft.com/v1.0/me/messages?$filter=importance%20eq%20'high')

<https://graph.microsoft.com/v1.0/me/events>

<https://graph.microsoft.com/v1.0/me/manager>

<https://graph.microsoft.com/v1.0/me/drive/root/children/foo.txt/lastModifiedByUser>

<https://graph.microsoft.com/v1.0/users>

<https://graph.microsoft.com/v1.0/groups>

<https://graph.microsoft.com/v1.0/me/people>

<https://graph.microsoft.com/beta/me/insights/trending>

<https://graph.microsoft.com/v1.0/me/onenote/notebooks>

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



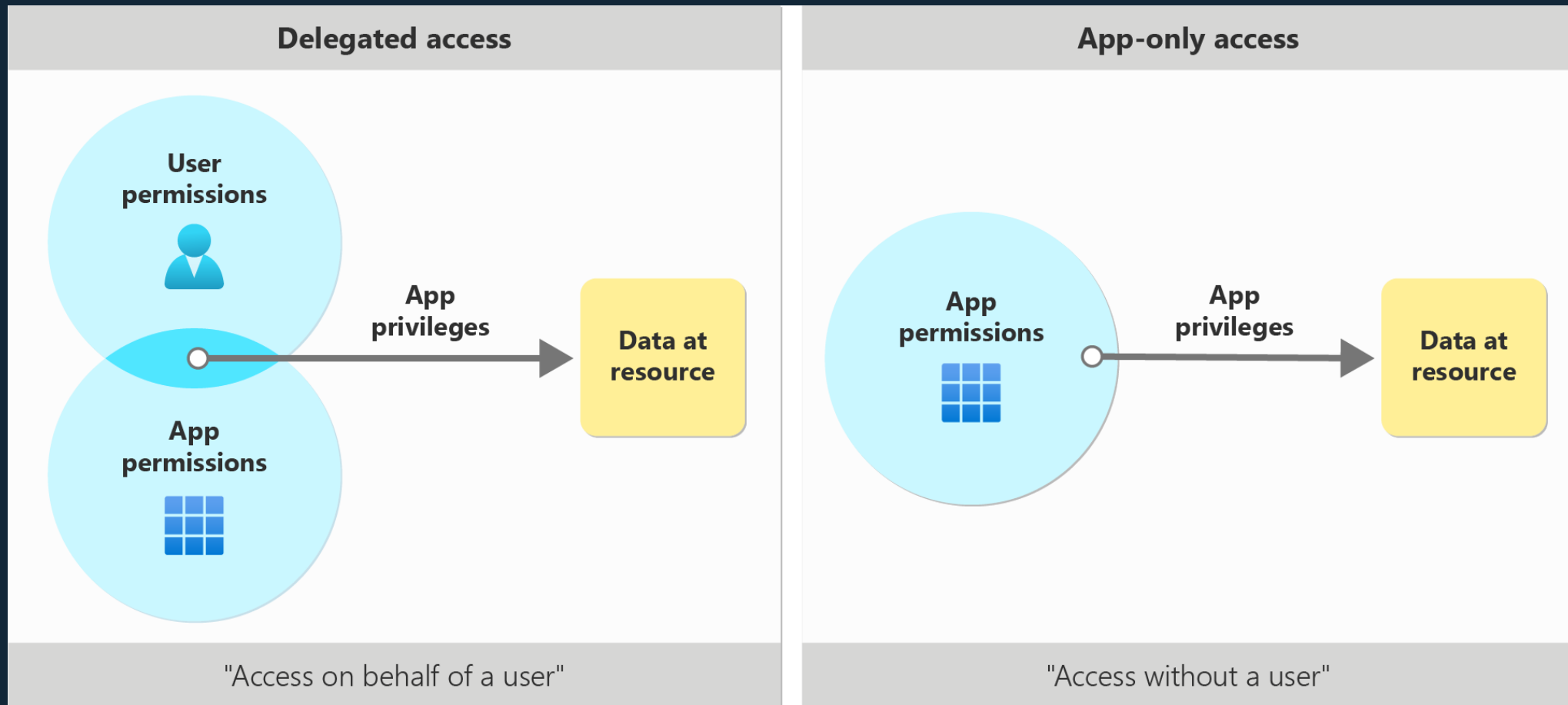
<https://developer.microsoft.com/en-us/graph/graph-explorer>

Demo

A quick look at the Graph Explorer

@thomasvochten

Graph permissions model



Typical AuthN & AuthZ workflow



Register an application in Entra ID



Configure authentication



Grant the necessary permissions to the app

Authentication options



User prompt / device code (interactive)



AppId & Secret



Certificate



Managed Identity

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



PowerShell to the rescue!

@thomasvochten

So many PowerShell modules

- Teams
- SharePoint Online
- Exchange
- AZ
- PnP
- ...

Microsoft Graph SDK for PowerShell

One module to rule them all

Cross-platform

Modern Authentication

Fine-grained privilege control



The possibilities are endless...

- 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”

PowerShell SDK

- Abstracts away most of the complexity
- `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`

Navigating the SDK

```
Get-Command -Module Microsoft.Graph* *team*  
Find-MgGraphCommand -Uri '/users/{id}'
```

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

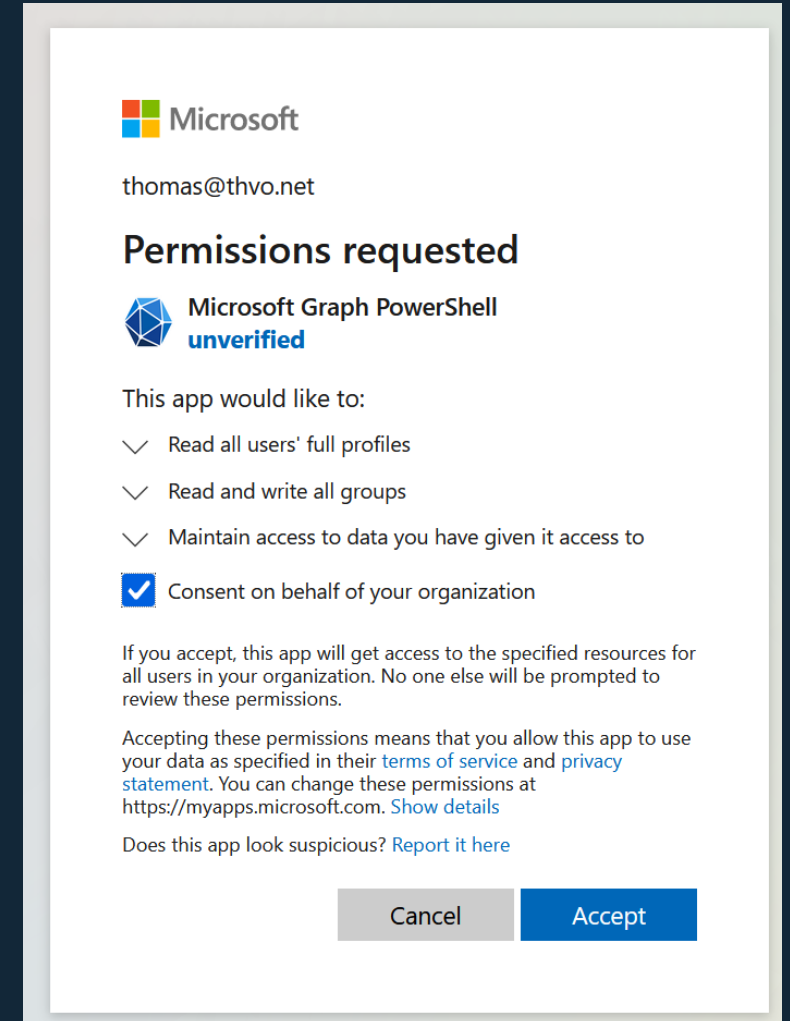
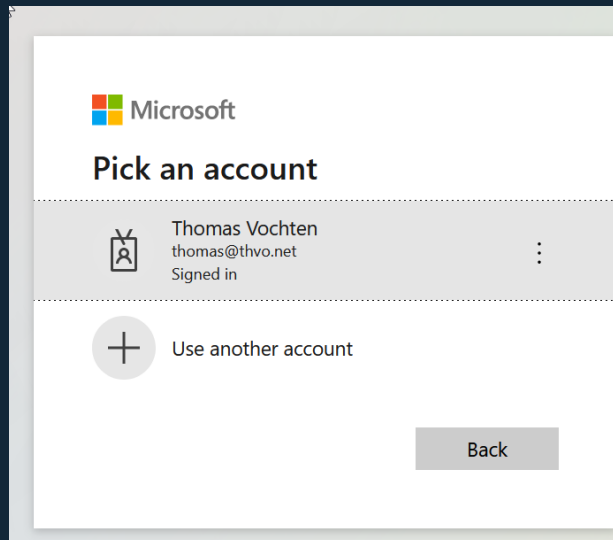
New-MgTeam

Update-MgTeamChannel

Remove-MgGroup

Connecting to the Graph

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



Using permission scopes

```
(Get-MgContext).Scopes
```

```
Connect-MgGraph -Scopes "Device.Read"
```

Repeat if you need additional permissions

A better alternative

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

Connect-MgGraph

```
-ClientId YOUR_APP_ID  
-TenantId YOUR_TENANT_ID  
-CertificateName YOUR_CERT_SUBJECT
```

Get-MgContext to verify

No cmdlet to be found?

Invoke-MGGraphRequest

Invoke-MgGraphRequest -Method GET
<https://graph.microsoft.com/v1.0/me>

Demo

The basics of the Graph PowerShell SDK

@thomasvochten



The wonderful world of the CLI

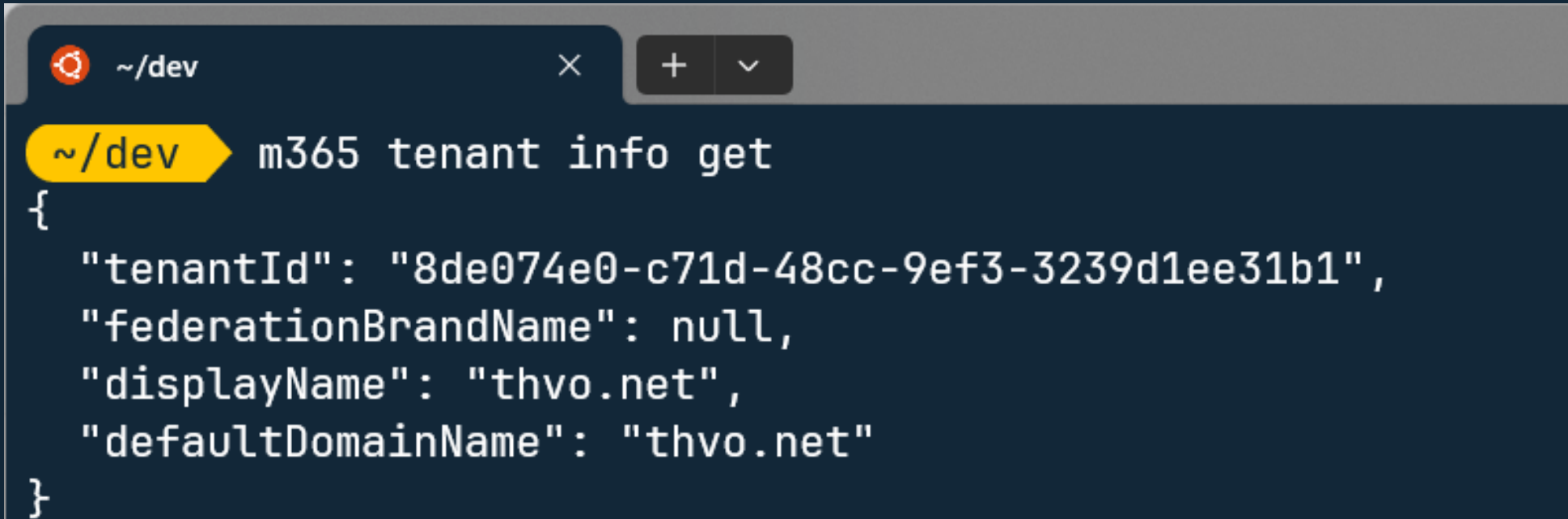
@thomasvochten

The rise of Command-Line Interface (CLI)

- Not PowerShell
- The new normal?
- What's your preference?
- When to use what?

The Microsoft 365 CLI

<https://pnp.github.io/cli-microsoft365>



```
~/dev x + v
~/dev m365 tenant info get
{
  "tenantId": "8de074e0-c71d-48cc-9ef3-3239d1ee31b1",
  "federationBrandName": null,
  "displayName": "thvo.net",
  "defaultDomainName": "thvo.net"
}
```

The Microsoft 365 CLI

- Community-driven
- What can I use it for?
- Prerequisites: node.js (comes as a npm package)

```
npm install -g @pnp/cli-microsoft365
```

- Cross-platform
- Can be run inside a Docker container
- AuthN & AuthZ through app registration

Demo

Using the m365 CLI

@thomasvochten

The MS Graph CLI

<https://learn.microsoft.com/en-us/graph/cli/overview>

```
PowerShell x + v
~\dev mgc me get
{
  "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#users/$entity",
  "businessPhones": [],
  "displayName": "Thomas Vochten",
  "givenName": "Thomas",
  "jobTitle": "Owner",
  "mail": "thomas@thvo.net",
  "mobilePhone": null,
  "officeLocation": null,
  "preferredLanguage": null,
  "surname": "Vochten",
  "userPrincipalName": "thomas@thvo.net",
  "id": "3be288ca-7a34-44fd-856b-ed7286065de5"
}
```

@thomasvochten

The MS Graph CLI

- Official Microsoft CLI
- Currently in preview
- Manual installation for now
- What can I use it for? Why is it there?
- Prerequisites
- AuthN & AuthZ as any Graph application

Demo

Using the Graph CLI

@thomasvochten



Automate all the things

@thomasvochten

Automate what exactly?

- License management
- Reporting
- Auditing
- Monitoring
- Change management
- Operational service management

More advanced business scenarios:

“When a user leaves the company, let other users know which shared documents they will lose access to”



Choices

- Scheduled Tasks
- Azure Webjobs
- Azure Automation
- Azure Functions
- Azure Logic Apps
- Power Automate
- Azure DevOps / GitHub Actions
- Azure batch

When to use what?

flexibility • cost • complexity
• resilience • orchestration •
security • triggers • scaling •
output needs



But I just want to run some PowerShell?

- Azure Automation
- Azure Functions



When to use what

It depends 🤖

- On the capabilities provided
- On the type of automation needed
- On what you're used to

Takeaways

- Get to know the Microsoft Graph, even if you're not a developer.
- PowerShell is still a powerful tool for working with Microsoft 365
- Install and the m365 & graph CLI tools, and use them in combination with PowerShell
- Discover automation possibilities, but use the simplest tool that could possibly work

Thank you

@thomasvochten

<https://thomasvochten.com>

mail@thomasvochten.com

Get the slides

