IE2062 [2023/JUL]- Web Security

Web Security BB Assignment

Report 02 – Officient.io

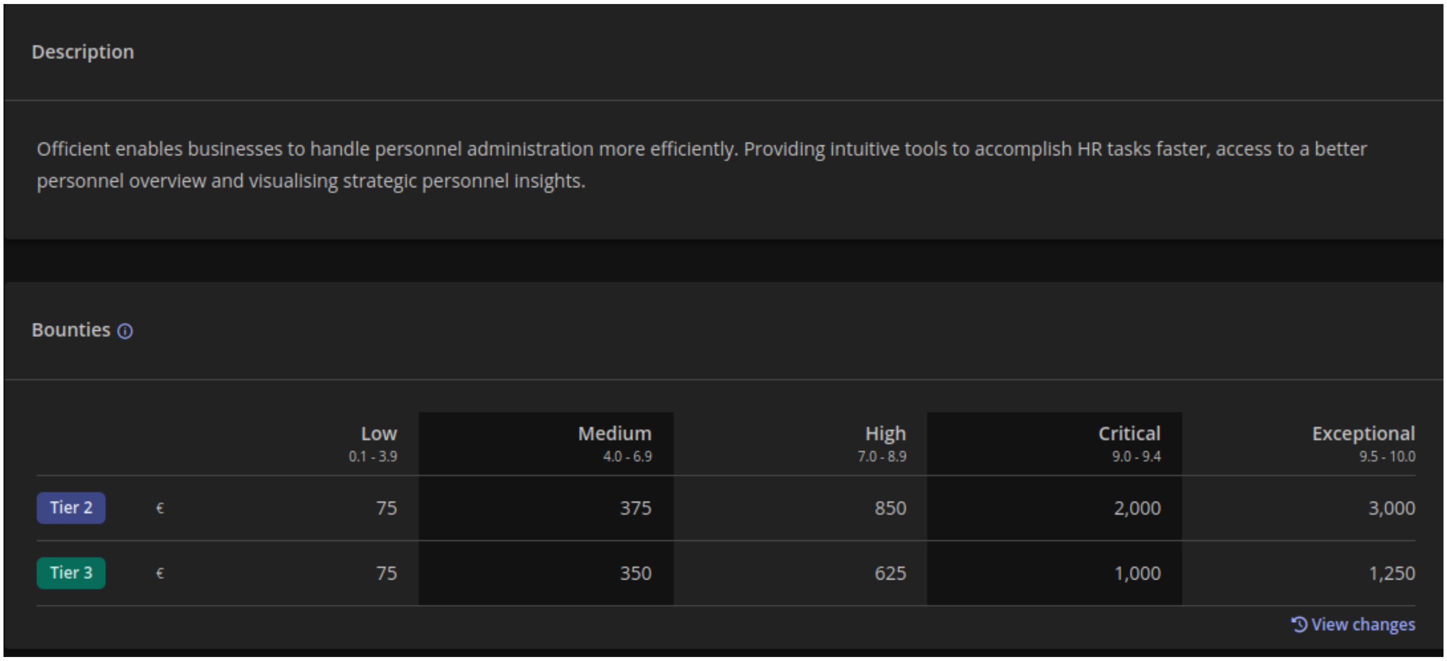
Lecturer in charge – Ms. Chethana Liyanapathirana

IT21826368 – Nanayakkara Y.D.T.D

Contents

1. Officient.io/Officient.io/Detail

1.1. Overview



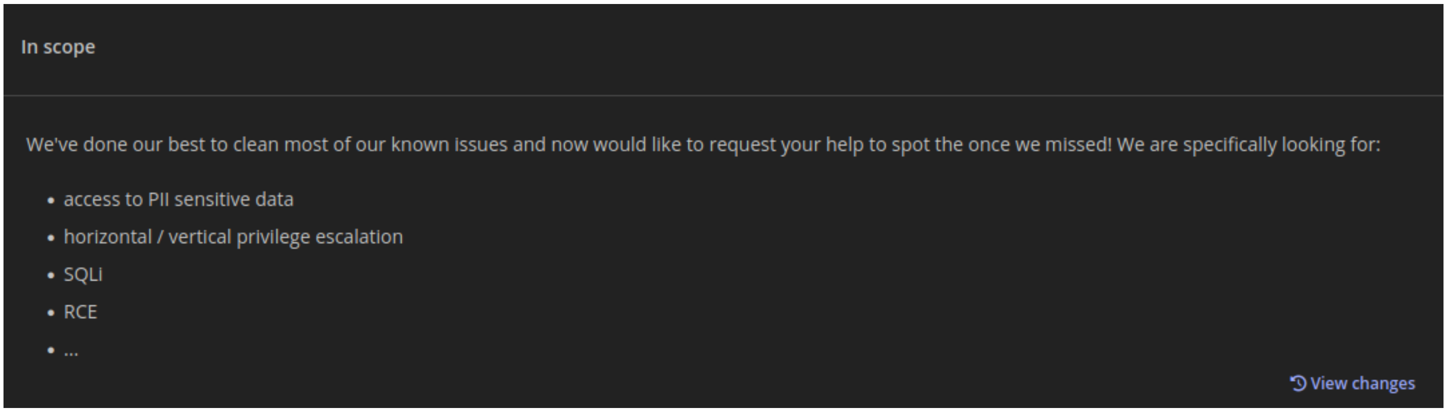
Description

Officient enables businesses to handle personnel administration more efficiently. Providing intuitive tools to accomplish HR tasks faster, access to a better personnel overview and visualising strategic personnel insights.

Bounties ⓘ

		Low 0.1 - 3.9	Medium 4.0 - 6.9	High 7.0 - 8.9	Critical 9.0 - 9.4	Exceptional 9.5 - 10.0
Tier 2	€	75	375	850	2,000	3,000
Tier 3	€	75	350	625	1,000	1,250

1.2. Scope

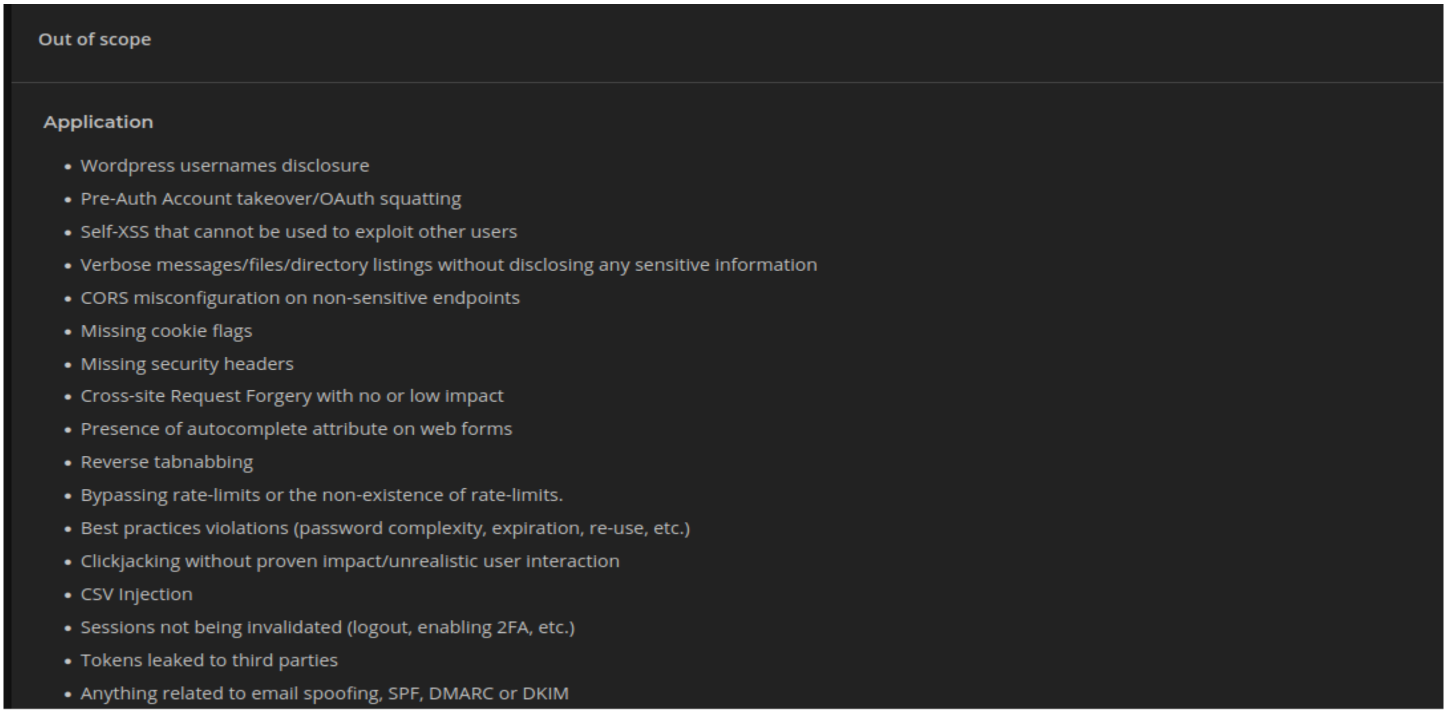


In scope

We've done our best to clean most of our known issues and now would like to request your help to spot the once we missed! We are specifically looking for:

- access to PII sensitive data
- horizontal / vertical privilege escalation
- SQLi
- RCE
- ...

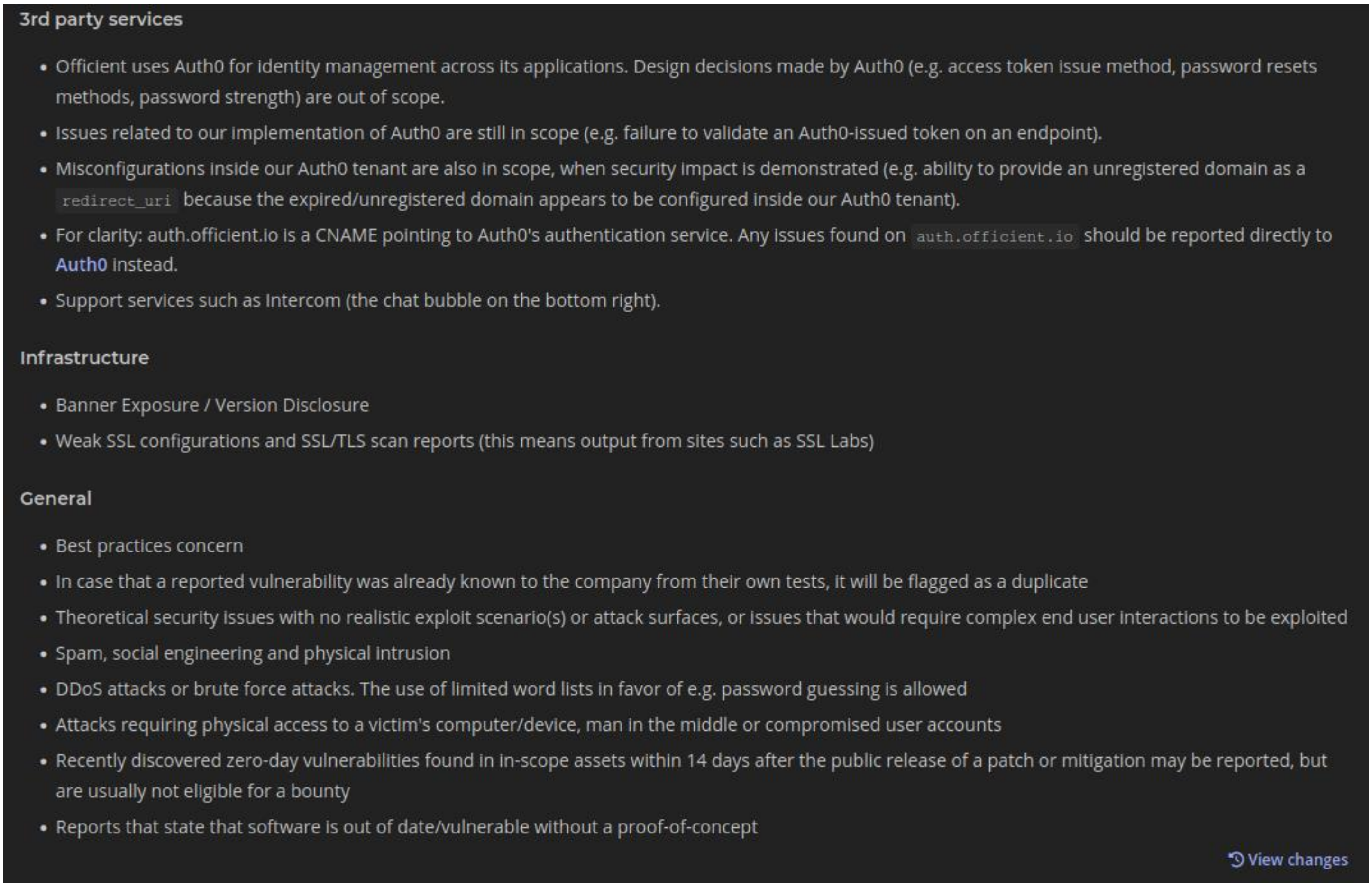
1.3.Out of Scope



Out of scope

Application

- Wordpress usernames disclosure
- Pre-Auth Account takeover/OAuth squatting
- Self-XSS that cannot be used to exploit other users
- Verbose messages/files/directory listings without disclosing any sensitive information
- CORS misconfiguration on non-sensitive endpoints
- Missing cookie flags
- Missing security headers
- Cross-site Request Forgery with no or low impact
- Presence of autocomplete attribute on web forms
- Reverse tabnabbing
- Bypassing rate-limits or the non-existence of rate-limits.
- Best practices violations (password complexity, expiration, re-use, etc.)
- Clickjacking without proven impact/unrealistic user interaction
- CSV Injection
- Sessions not being invalidated (logout, enabling 2FA, etc.)
- Tokens leaked to third parties
- Anything related to email spoofing, SPF, DMARC or DKIM



3rd party services

- Officient uses Auth0 for identity management across its applications. Design decisions made by Auth0 (e.g. access token issue method, password resets methods, password strength) are out of scope.
- Issues related to our implementation of Auth0 are still in scope (e.g. failure to validate an Auth0-issued token on an endpoint).
- Misconfigurations inside our Auth0 tenant are also in scope, when security impact is demonstrated (e.g. ability to provide an unregistered domain as a `redirect_uri` because the expired/unregistered domain appears to be configured inside our Auth0 tenant).
- For clarity: auth.officient.io is a CNAME pointing to Auth0's authentication service. Any issues found on `auth.officient.io` should be reported directly to **Auth0** instead.
- Support services such as Intercom (the chat bubble on the bottom right).

Infrastructure

- Banner Exposure / Version Disclosure
- Weak SSL configurations and SSL/TLS scan reports (this means output from sites such as SSL Labs)

General

- Best practices concern
- In case that a reported vulnerability was already known to the company from their own tests, it will be flagged as a duplicate
- Theoretical security issues with no realistic exploit scenario(s) or attack surfaces, or issues that would require complex end user interactions to be exploited
- Spam, social engineering and physical intrusion
- DDoS attacks or brute force attacks. The use of limited word lists in favor of e.g. password guessing is allowed
- Attacks requiring physical access to a victim's computer/device, man in the middle or compromised user accounts
- Recently discovered zero-day vulnerabilities found in in-scope assets within 14 days after the public release of a patch or mitigation may be reported, but are usually not eligible for a bounty
- Reports that state that software is out of date/vulnerable without a proof-of-concept

1.4. Selected Domains

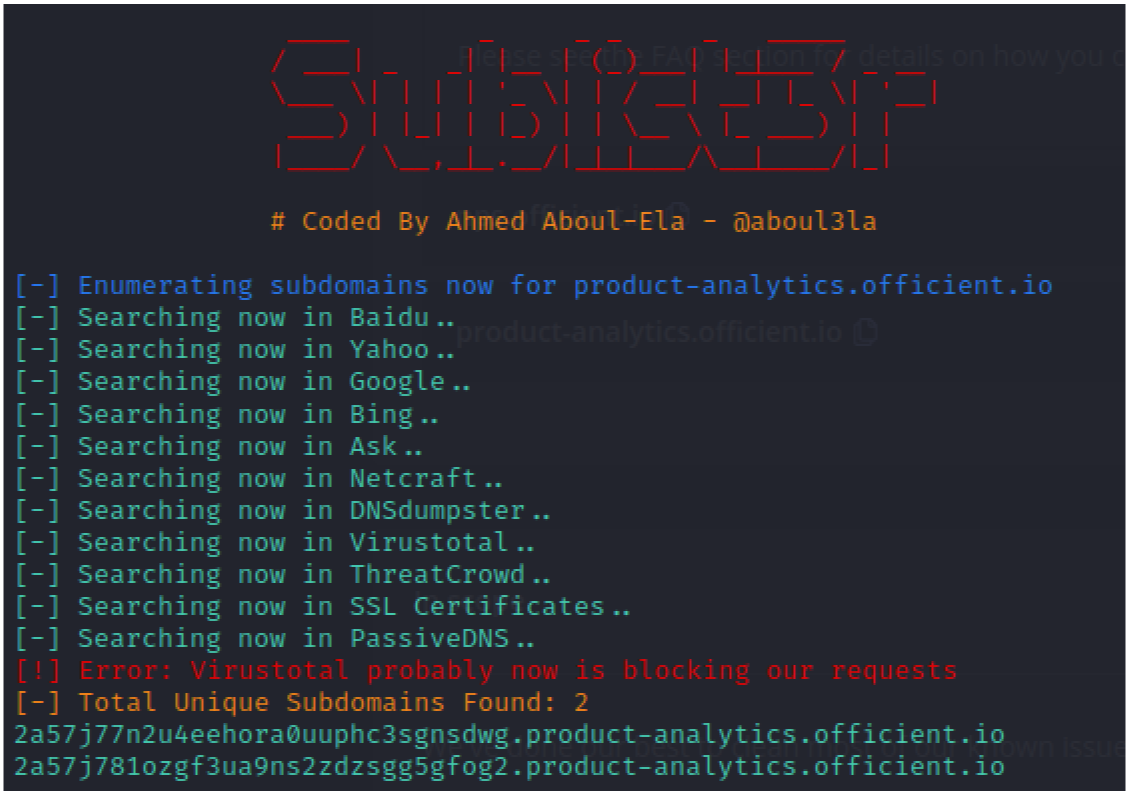
product-analytics.officient.io Tier 3 URL

product-analytics.officient.io is the URL using in this report.

2. Information Gathering

2.1. Sublist3r

Using Sublist3r for the subdomain scan.

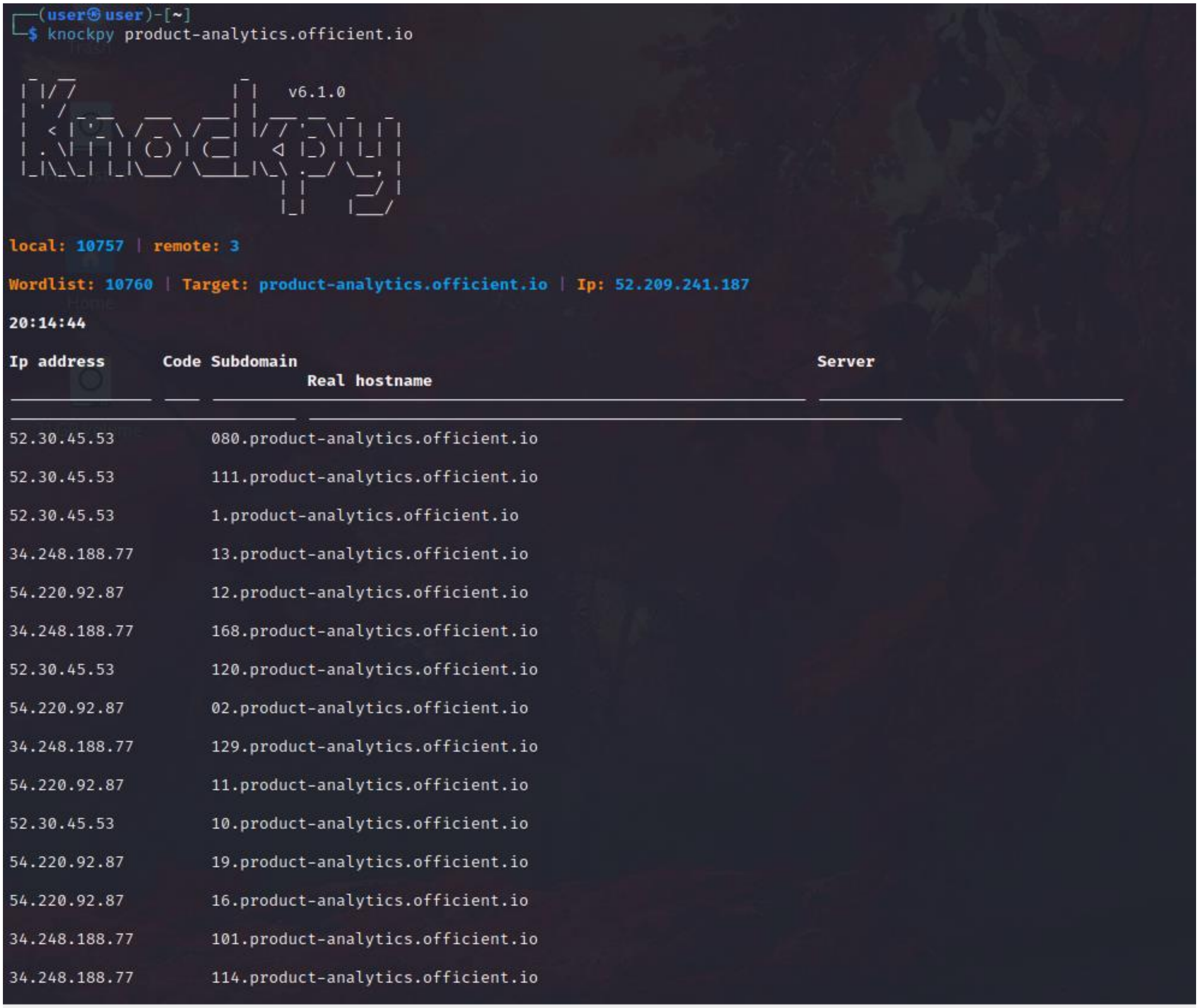


```
                 ____        _     _ _     _   _____
                / ___| _   _| |__ | (_)___| |_|___ / _ __
                \___ \| | | | '_ \| | / __| __| |_ \| '__|
                 ___) | |_| | |_) | | \__ \ |_ ___) | |
                |____/ \__,_|_.__/|_|_|___/\__|____/|_|

                # Coded By Ahmed Aboul-Ela - @aboul3la

[-] Enumerating subdomains now for product-analytics.officient.io
[-] Searching now in Baidu..
[-] Searching now in Yahoo..
[-] Searching now in Google..
[-] Searching now in Bing..
[-] Searching now in Ask..
[-] Searching now in Netcraft..
[-] Searching now in DNSdumpster..
[-] Searching now in Virustotal..
[-] Searching now in ThreatCrowd..
[-] Searching now in SSL Certificates..
[-] Searching now in PassiveDNS..
[!] Error: Virustotal probably now is blocking our requests
[-] Total Unique Subdomains Found: 2
2a57j77n2u4eehora0uuphc3sgnsdwg.product-analytics.officient.io
2a57j781ozgf3ua9ns2zdzsgg5gfog2.product-analytics.officient.io
```

As you can see, we found 2 unique subdomains.

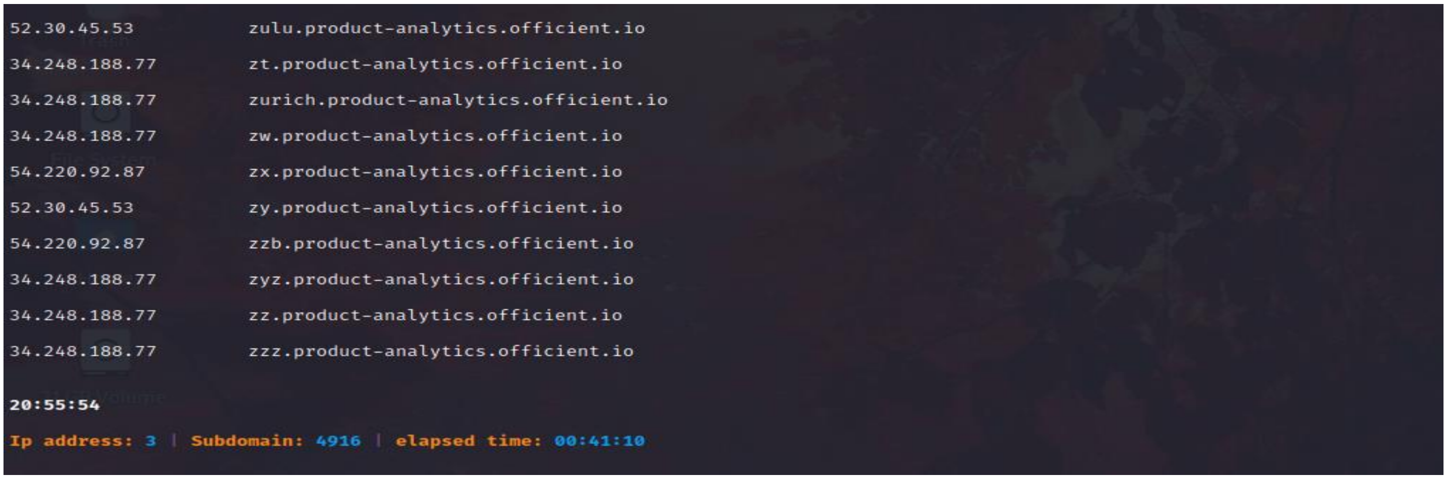
2.2. Knockpy

Using Knockpy tool to scan Ip addresses and subdomains.



```
┌──(user㉿user)-[~]
└─$ knockpy product-analytics.officient.io

  _  __                 _
 | |/ /                | |   v6.1.0
 | ' / _ __   ___   ___| | ___ __  _   _
 |  < | '_ \ / _ \ / __| |/ / '_ \| | | |
 | . \| | | | (_) | (__|   <| |_) | |_| |
 |_|\_\_| |_|\___/ \___|_|\_\ .__/ \__, |
                            | |     __/ |
                            |_|    |___/

local: 10757 | remote: 3

Wordlist: 10760 | Target: product-analytics.officient.io | Ip: 52.209.241.187

20:14:44

Ip address      Code Subdomain                                                          Server
                          Real hostname
_______________ ____ ______________________________________________________________ ________________________________
_____________________________ ______________________________________________________________

52.30.45.53          080.product-analytics.officient.io

52.30.45.53          111.product-analytics.officient.io

52.30.45.53          1.product-analytics.officient.io

34.248.188.77        13.product-analytics.officient.io

54.220.92.87         12.product-analytics.officient.io

34.248.188.77        168.product-analytics.officient.io

52.30.45.53          120.product-analytics.officient.io

54.220.92.87         02.product-analytics.officient.io

34.248.188.77        129.product-analytics.officient.io

54.220.92.87         11.product-analytics.officient.io

52.30.45.53          10.product-analytics.officient.io

54.220.92.87         19.product-analytics.officient.io

54.220.92.87         16.product-analytics.officient.io

34.248.188.77        101.product-analytics.officient.io

34.248.188.77        114.product-analytics.officient.io
```

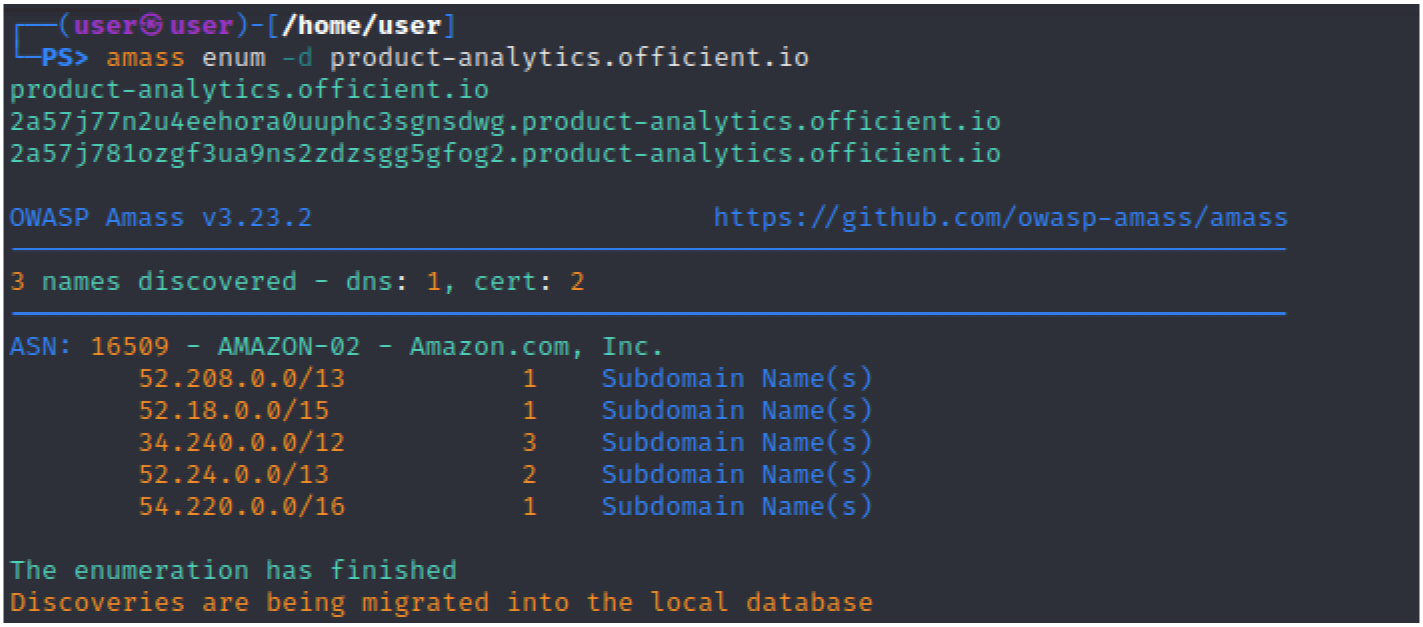
In the below we can see we found 3 IP addresses and 4916 subdomains



```
52.30.45.53          zulu.product-analytics.officient.io
34.248.188.77        zt.product-analytics.officient.io
34.248.188.77        zurich.product-analytics.officient.io
34.248.188.77        zw.product-analytics.officient.io
54.220.92.87         zx.product-analytics.officient.io
52.30.45.53          zy.product-analytics.officient.io
54.220.92.87         zzb.product-analytics.officient.io
34.248.188.77        zyz.product-analytics.officient.io
34.248.188.77        zz.product-analytics.officient.io
34.248.188.77        zzz.product-analytics.officient.io

20:55:54
Ip address: 3 | Subdomain: 4916 | elapsed time: 00:41:10
```

2.3. Amass

We can use the amass to find the IP address names.



```
┌──(user㉿user)-[/home/user]
└─PS> amass enum -d product-analytics.officient.io
product-analytics.officient.io
2a57j77n2u4eehora0uuphc3sgnsdwg.product-analytics.officient.io
2a57j781ozgf3ua9ns2zdzsgg5gfog2.product-analytics.officient.io

OWASP Amass v3.23.2                          https://github.com/owasp-amass/amass
────────────────────────────────────────────────────────────────────────────────
3 names discovered - dns: 1, cert: 2
────────────────────────────────────────────────────────────────────────────────
ASN: 16509 - AMAZON-02 - Amazon.com, Inc.
        52.208.0.0/13           1    Subdomain Name(s)
        52.18.0.0/15            1    Subdomain Name(s)
        34.240.0.0/12           3    Subdomain Name(s)
        52.24.0.0/13            2    Subdomain Name(s)
        54.220.0.0/16           1    Subdomain Name(s)

The enumeration has finished
Discoveries are being migrated into the local database
```

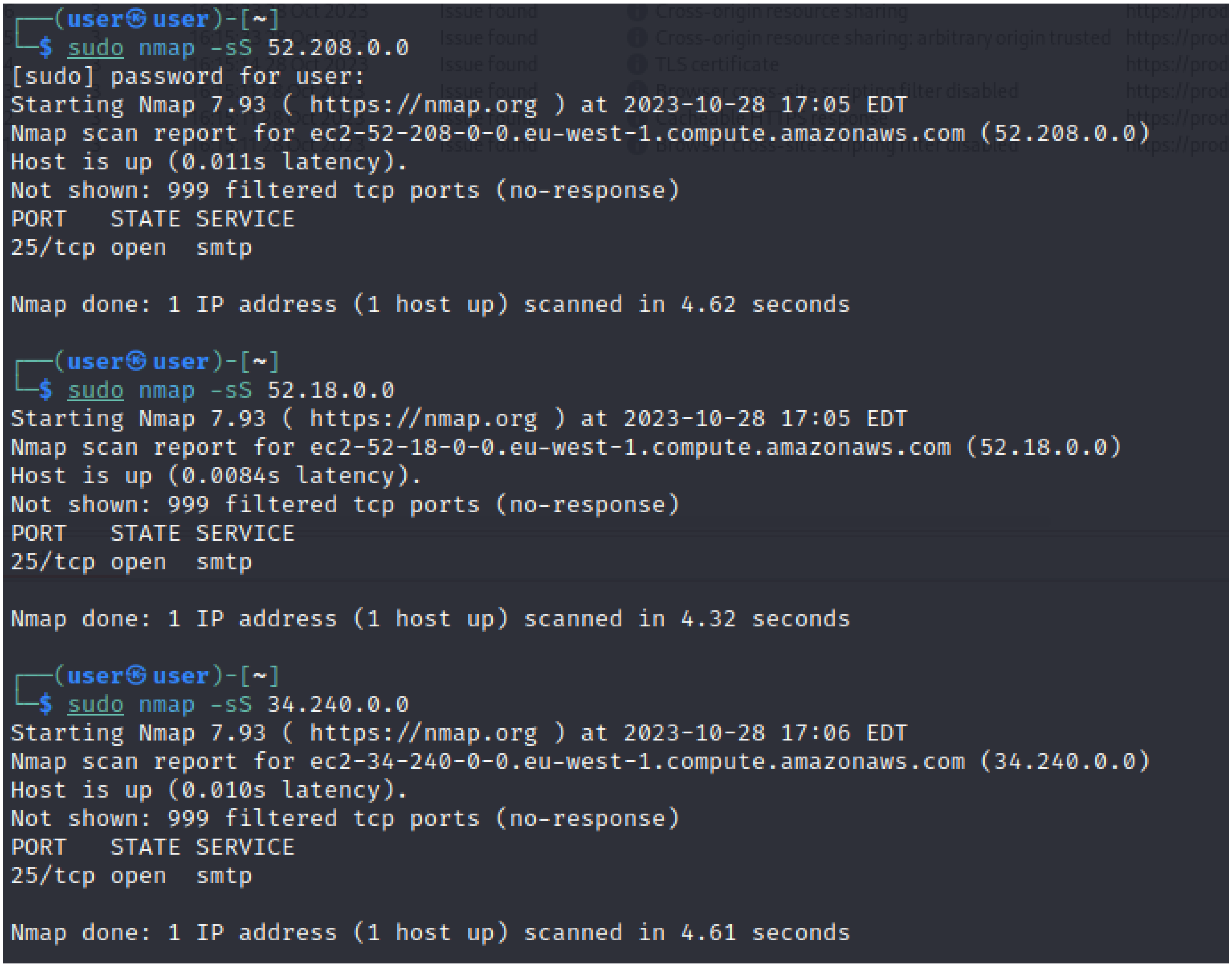
We have found 3 names – 1 DNS and 2 CERT.

It has Amazon.com Domains

3. Scanning Vulnerability

3.1. Nmap

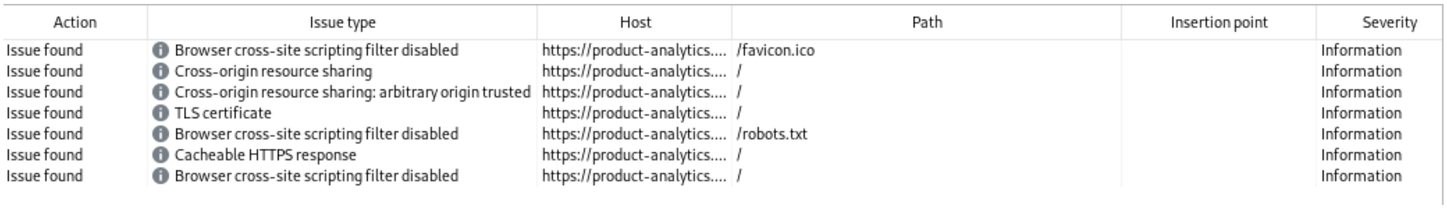
Then we can scan ports that we found using Nmap



```
┌──(user㉿user)-[~]
└─$ sudo nmap -sS 52.208.0.0
[sudo] password for user:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-10-28 17:05 EDT
Nmap scan report for ec2-52-208-0-0.eu-west-1.compute.amazonaws.com (52.208.0.0)
Host is up (0.011s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT   STATE SERVICE
25/tcp open  smtp

Nmap done: 1 IP address (1 host up) scanned in 4.62 seconds

┌──(user㉿user)-[~]
└─$ sudo nmap -sS 52.18.0.0
Starting Nmap 7.93 ( https://nmap.org ) at 2023-10-28 17:05 EDT
Nmap scan report for ec2-52-18-0-0.eu-west-1.compute.amazonaws.com (52.18.0.0)
Host is up (0.0084s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT   STATE SERVICE
25/tcp open  smtp

Nmap done: 1 IP address (1 host up) scanned in 4.32 seconds

┌──(user㉿user)-[~]
└─$ sudo nmap -sS 34.240.0.0
Starting Nmap 7.93 ( https://nmap.org ) at 2023-10-28 17:06 EDT
Nmap scan report for ec2-34-240-0-0.eu-west-1.compute.amazonaws.com (34.240.0.0)
Host is up (0.010s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT   STATE SERVICE
25/tcp open  smtp

Nmap done: 1 IP address (1 host up) scanned in 4.61 seconds
```

3.2. Burp suite

Scanning Using Burp Suite



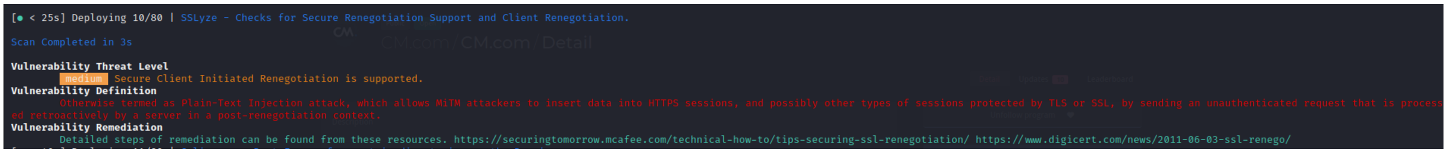
Action	Issue type	Host	Path	Insertion point	Severity
Issue found	Browser cross-site scripting filter disabled	https://product-analytics....	/favicon.ico		Information
Issue found	Cross-origin resource sharing	https://product-analytics....	/		Information
Issue found	Cross-origin resource sharing: arbitrary origin trusted	https://product-analytics....	/		Information
Issue found	TLS certificate	https://product-analytics....	/		Information
Issue found	Browser cross-site scripting filter disabled	https://product-analytics....	/robots.txt		Information
Issue found	Cacheable HTTPS response	https://product-analytics....	/		Information
Issue found	Browser cross-site scripting filter disabled	https://product-analytics....	/		Information

- Browser cross site scripting filter disabled.
- Cross Origin resource sharing
- Cross origin resource sharing arbitrary origin trusted.

Above is the result of the Burp Suite scan craw and audit, it shows those vulnerabilities.

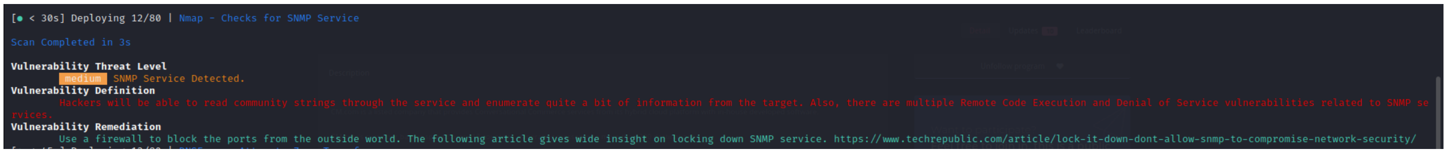
3.3. Scan with rapid scan

- Secure Client Initiated Renegotiation is supported.



```
[● < 25s] Deploying 10/80 | SSLyze - Checks for Secure Renegotiation Support and Client Renegotiation.

Scan Completed in 3s

Vulnerability Threat Level
        medium  Secure Client Initiated Renegotiation is supported.
Vulnerability Definition
        Otherwise termed as Plain-Text Injection attack, which allows MiTM attackers to insert data into HTTPS sessions, and possibly other types of sessions protected by TLS or SSL, by sending an unauthenticated request that is process
ed retroactively by a server in a post-renegotiation context.
Vulnerability Remediation
        Detailed steps of remediation can be found from these resources. https://securingtomorrow.mcafee.com/technical-how-to/tips-securing-ssl-renegotiation/ https://www.digicert.com/news/2011-06-03-ssl-renego/
```

- SNMP Service Detected.



```
[● < 30s] Deploying 12/80 | Nmap - Checks for SNMP Service

Scan Completed in 3s

Vulnerability Threat Level
        medium  SNMP Service Detected.
Vulnerability Definition
        Hackers will be able to read community strings through the service and enumerate quite a bit of information from the target. Also, there are multiple Remote Code Execution and Denial of Service vulnerabilities related to SNMP se
rvices.
Vulnerability Remediation
        Use a firewall to block the ports from the outside world. The following article gives wide insight on locking down SNMP service. https://www.techrepublic.com/article/lock-it-down-dont-allow-snmp-to-compromise-network-security/
```

4. Vulnerability description

- **Browser Cross-Site Scripting Filter Disabled**:
  - Disabling the browser's built-in cross-site scripting (XSS) filter may leave the website vulnerable to XSS attacks, allowing malicious scripts to be injected into web pages.
  -
- **Cross-Origin Resource Sharing (CORS)**:
  - The presence of CORS headers indicates that the website allows cross-origin requests. Misconfigured CORS settings can potentially lead to security issues, such as data exposure to unauthorized domains.
  -
- **Cacheable HTTPS Response**:
  - Cacheable HTTPS responses may introduce security concerns as sensitive data can be stored in caches and exposed to unauthorized users, potentially compromising data confidentiality.
- **Browser Cross-Site Scripting Filter Disabled** (Note: This appears to be a duplicate entry. Please verify if this is intended or if there are different aspects to this vulnerability).
-
- **Secure Client-Initiated Renegotiation Supported**:
  - The support for Secure Client-Initiated Renegotiation may introduce security risks if not configured correctly, potentially facilitating man-in-the-middle attacks.
  -
- **SNMP Service Detected**:
  - The presence of the Simple Network Management Protocol (SNMP) service may expose sensitive information and configuration details, potentially posing a security risk if not adequately secure.

5. Affected components.

- Web application server and client-side code.
- HTTPS response configuration.
- Security settings related to XSS filters and CORS.

6. Impact assessment

- Increased risk of XSS attacks due to disabled XSS filters, potentially leading to data theft or manipulation.
- Risk of data exposure or resource misuse due to CORS misconfiguration.
- Potential data leakage or exposure due to cacheable HTTPS responses.

7. Steps to reproduce.

None

8. Proof of concept (if applicable)

None

9. Proposed mitigation or fix

To address these vulnerabilities, consider the following actions:

- Enable and configure the XSS filter to protect against XSS attacks.
- Review and properly configure CORS policies to restrict cross-origin requests to trusted domains.
- Adjust HTTPS response headers to prevent sensitive data from being cached.
- Regularly update and patch the web application and its components to address security issues.
- Implement security best practices and consider using a Web Application Firewall (WAF) to enhance protection.

10. Summary

The identified vulnerabilities involve XSS filter settings, CORS misconfiguration, and cacheable HTTPS responses. While they are not highly severe, addressing these issues is essential to enhance the security of the web application and mitigate the risk of data exposure, XSS attacks, and other potential threats. The specific actions taken will depend on the unique characteristics of the web application.