Web Security

Bug Bounty Journal

Year 2 semester 2

BSc (Hons) In Information Technology

Specializing In Cyber Security

Sri Lanka Institute Of Information Technology

Malabe

Sri Lanka

Student name	Registration number
W.M.K.R.Wanasinghe	IT22298508

Table of Contents

Acknowledgement

Foremost, as a second year second semester Cyber Security student I would like to convey my sincere gratitude to Ms. Chethana Liyanapathirana, the Lecture in Charge of the Web Security module. Her support, guidance and advice helped us to complete this web audit/ bug bounty task successfully. Her constant guidance also helped us during the time of writing this report.

And also, I would like to express my sincere gratitude to our Lab Instructor who guided us and helped us in the Labs that made it easier to complete this bug bounty/ web audit. I appreciate the advice and assistance you provided so that I could complete the task successfully.

Abstract

Cybercrimes, data breaches, and misrepresentation are all dangerous risks that any Company or an organization might face. A large amount of information has been lost, and now organizations need to find out the steps to take to stop the danger from getting worse and to avoid more bad things happening. An investigation was conducted to investigate the processes related to IT security web audits. How they can assist businesses in improving their IT security. This study looks at the impact of the understanding of the threats posed by cybercrime. This investigation shows the web application vulnerabilities that are possessed by a company by evaluating the remediations to be taken to get rid of the security risks. This study has a detailed description about the vulnerabilities that the specific domains of the website possess. This study was done to learn more about cybercrime and gather more data on it. The aim is to put together more extensive information about the vulnerabilities of the website and help the company to mitigate them in order to get rid of the security issues. The investigation clearly showed that IT security audits are more critical for growth of every organization which uses Information Technology.

Introduction

Cyber attacks

Cyber-attacks refer to any offensive maneuvers aimed at computer systems, networks, or infrastructure with the intent to steal, alter, destroy, or exploit data. These attacks are executed by cybercriminals using various techniques and tools to breach security measures. Key types of cyber-attacks include:

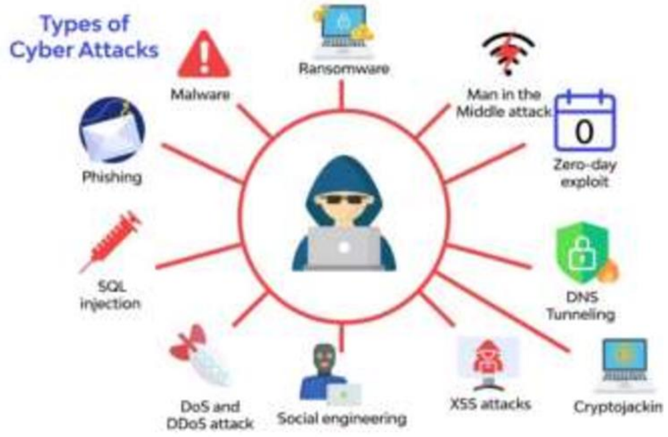
1 - **Phishing**: Deceptive emails or messages designed to trick recipients into revealing sensitive information.

2 - **Malware**: Malicious software like viruses, worms, and ransomware that infects systems to cause damage or gain unauthorized access.

3 - **DDoS (Distributed Denial of Service) Attacks**: Overwhelming a network or service with excessive traffic to make it unavailable to users.

4 - **Man-in-the-Middle (MitM) Attacks**: Intercepting and manipulating communication between two parties without their knowledge.

5 - **SQL Injection**: Inserting malicious SQL code into a database query to manipulate and access data.



Cyber crimes

Cyber-crimes encompass illegal activities carried out using computers and the internet. They can target individuals, corporations, or governments and often involve financial gain or data theft. Major categories of cyber-crimes include:

1 - **Identity Theft**: Stealing personal information to commit fraud, such as opening bank accounts or making purchases in the victim's name.

2 - **Financial Fraud**: Illegally accessing financial accounts or conducting unauthorized transactions.

3 - **Cyberstalking and Harassment**: Using digital communication tools to harass, intimidate, or stalk individuals.

4 - **Intellectual Property Theft**: Stealing or copying proprietary information, software, or digital content.

5 - **Cyber Espionage**: Unauthorized access to confidential information, often for political or competitive advantage.

Daily journal – (20.04.2024)

Objectives:

- Learn about what is OWASP TOP 10 vulnerabilities.
- Leard the vulnerabilities.
- What are the impact and the solutions for those vulnerabilities.

Sources used:

- Google
- OWASP website
- Portswigger.

OWASP TOP 10 Security risk and vulnerabilities.

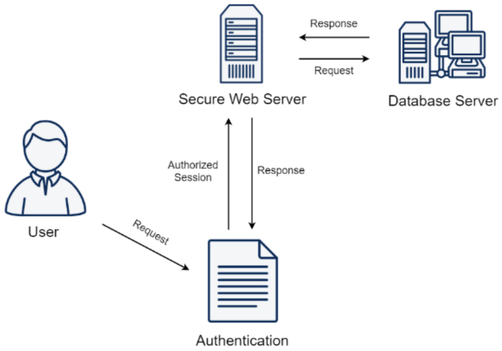
The OWASP Top 10 is a widely recognized list of the most critical web application security vulnerabilities, compiled by the Open Web Application Security Project (OWASP). The list highlights the top threats that developers and security professionals should be aware of, including Injection (e.g., SQL injection), Broken Authentication, Sensitive Data Exposure, XML External Entities (XXE), Broken Access Control, Security Misconfigurations, Cross-Site Scripting (XSS), Insecure Deserialization, Using Components with Known Vulnerabilities, and Insufficient Logging & Monitoring. These vulnerabilities are identified and ranked based on their prevalence, exploitability, and potential impact, serving as a foundational resource for improving web application security.

Why OWASP TOP 10 is important

- Helps in Mitigation of risks.
- Guidance to Prioritize Security concerns.
- Awareness and gain knowledge about Web Application vulnerabilities.
- Improvement of the security of organization’s web applications.
- To improve the security Design of the web applications.

1. Broken Authentication.

Broken authentication is a security vulnerability that occurs when applications improperly implement mechanisms to handle user credentials, session tokens, or other authentication-related processes. This flaw can allow attackers to gain unauthorized access to user accounts and sensitive data. Common causes include weak password policies, session fixation, insufficient credential storage, and failure to properly invalidate sessions. Mitigating this risk involves enforcing strong password requirements, implementing multi-factor authentication, securely managing session tokens, and regularly reviewing authentication mechanisms for potential weaknesses.



Countermeasures

- Use of strong passwords.
- Session management.
- Brute force detection and monitoring.
- API security.
- Regular security audits and penetration test.

2. Injection.

Injections in cybersecurity refer to a type of attack where an attacker sends malicious code into a system or application to manipulate its behavior. This can occur through various means, such as SQL injection, where malicious SQL queries are sent to a database, or command injection, where operating system commands are executed without proper validation. These vulnerabilities arise when user inputs are not correctly sanitized or validated, allowing attackers to execute unintended commands, access sensitive data, or disrupt the normal functioning of applications. Preventing injection attacks involves implementing robust input validation, using parameterized queries, and employing secure coding practices.

There are several types of injection attacks such as:

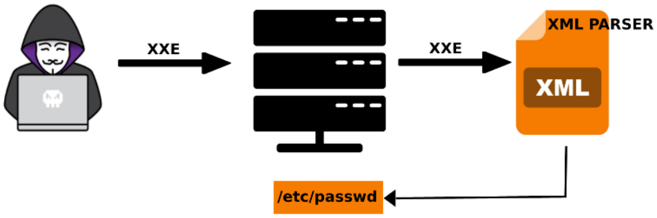
1. SQL Injection. (the most common injection attack)
2. Command Injection.
3. XML Injection.

Countermeasures

- Use stored procedures.
- Least privilege.
- Use object relation mapping.
- Use prepared statement.
- Input validation.
- Character escaping.

3. XML External Entities. (XXE).

XML External Entities (XXE) is a weakness in the way web applications handle XML files. This happens when a hacker can change or insert malicious things into a computer program that reads XML. These outside sources can point to files on your computer or other computers, which can cause problems with keeping information private, stopping services from working, and even letting someone run code on your computer from a distance. XXE attacks are really worrying because they can take advantage of XML data transfer, which is commonly used in web services and communication between API.



Countermeasures

- Input Validation and Sanitization.
- Use of Firewalls.
- Update and Patch Software.
- Access Controls.
- Use Modern and secure parsers.
- Disable external entity processing.

4. Sensitive data exposure.

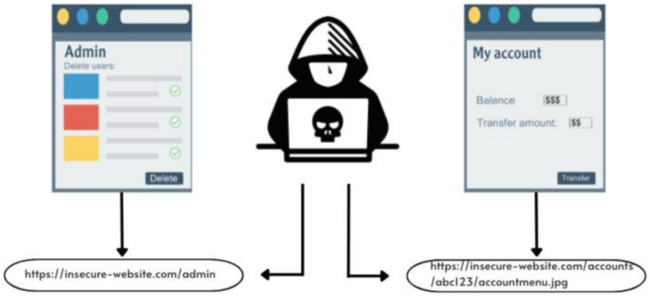
Sensitive Data Exposure occurs when an app or system accidentally gives out private information to people who shouldn't have it. This weakness can appear in different ways, like not storing important information properly, using weak encryption methods, or having insufficient access controls. Sensitive data usually refers to personal details, financial documents, login information, and other important information. When someone sees this information, attackers can use it to steal identities, commit fraud, or do other bad things.

Countermeasures

- Access Controls.
- Security Testing.
- Secure Data Storage.
- Data Encryption.
- Data Masking and Redaction.

5. Broken access control.

Broken Access Control is a big problem in web applications. It happens when the system doesn't properly check who can access certain things. It lets people who shouldn't be able to access certain places or do certain things in a computer program, do those things anyway. This weakness usually happens when there are mistakes in how a person is identified and given permission to access something. It can also be caused by not managing a person's session properly. When access control is broken, attackers can use it to look at private information, change data, or even get control over the system.

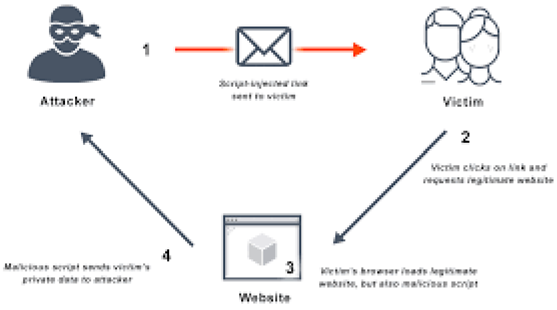


Countermeasures.

- Explicit Authorization Checks.
- Session Management.
- Access Control Testing.
- Authentication Tokens.
- Error- Handling

6. Cross-site scripting. (XSS)

Cross-Site Scripting (XSS) is a common and risky security problem on websites. It happens when a person puts harmful scripts or code into web pages that other people look at. These codes are run on the victim's web browser and can cause various harmful actions like stealing sensitive information, taking over user sessions, changing the appearance of websites, or spreading harmful software. XSS vulnerabilities happen when web applications don't properly check and clean up the information that users enter or display on the website. This makes it possible for attackers to add harmful code into fields where users enter information, website addresses, or other content that users create.

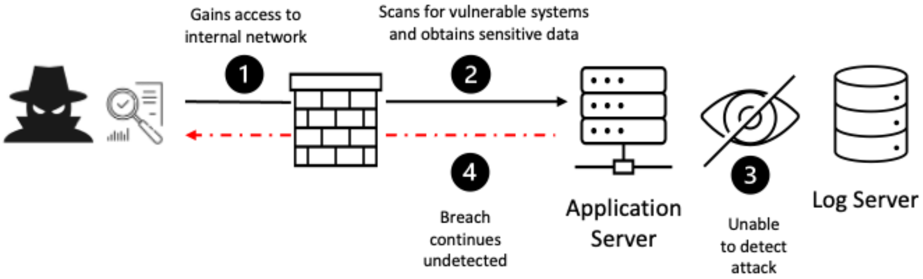

Attacker
1
Script-injected link sent to victim
Victim
2
Victim clicks on link and requests legitimate website
3
Victim's browser loads legitimate website, but also malicious script
Website
4
Malicious script sends victim's private data to attacker

Countermeasures.

- Content Security Policy.
- Regular Security Testing.
- Browser Security Features.
- Use trusted Frameworks and Libraries.
- Output Encoding.
- Security Headers.

7. Insufficient logging and monitoring.

Insufficient Logging and Monitoring means that applications and systems do not keep enough records of security events and incidents. This is a big security problem. This weakness can prevent organizations from knowing about unauthorized access, suspicious activities, and possible breaches. Not having enough defined event logs, proper security information and event management tools, or proper alerting mechanisms can lead to inadequate logging and monitoring. The results of this problem are that security incidents are not found quickly, it takes a long time to respond to them, and threats cannot be investigated or stopped effectively


Gains access to internal network
1
Scans for vulnerable systems and obtains sensitive data
2
4
Breach continues undetected
Application Server
3
Unable to detect attack
Log Server

Countermeasures.

- Log all critical events.
- Incident Response plan.
- Secure Log Storage.
- Real-Time Monitoring.
- Regular review and analysis of logs.
- Regular Testing and Simulation.

8. Using components with known vulnerabilities.

Using software components, libraries, frameworks, or modules that have known security weaknesses is a very risky thing for organizations to do. This happens when they add these components to their applications or systems without checking if they are secure. If there are weaknesses in these parts, it can make applications and systems vulnerable to different security problems like unauthorized access, stealing of data, and running harmful code from a remote location. This risk happens when you don't update old or weak parts, don't check for security warnings, or don't evaluate the security of things you depend on.

Using a publicly available exploit, they are able to successfully attack the site and access an administrative account.

Daily journal-day(27.04.2024)

Objectives:

- Find the tools used for bug bounty program.
- Learn how to use them.
- Install the tools.

Sources used:

- OWASP
- Portswigger
- You tube
- Google

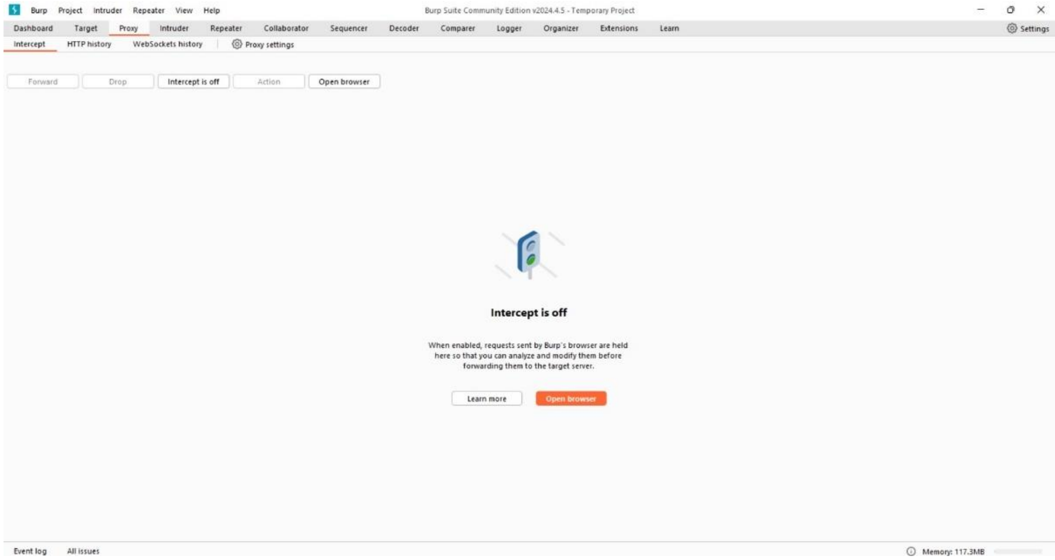
Key tools used for bug bounty hunting.

1. OWASP Zap

A well-known open-source security testing tool for identifying and fixing vulnerabilities in online applications is called OWASP ZAP (Zed Attack Proxy). Users may intercept, examine, and change online traffic for security research with this tool, which serves as both a scanner and a proxy. ZAP aids programmers and security experts in locating typical online application flaws including Cross-Site Scripting (XSS), SQL Injection, and Cross-Site Request Forgery (CSRF). It is used by both novice and expert security testers because to its user-friendly interface and automatic scanning features. ZAP is continuously updated as part of the OWASP project to stay on top of new threats, ensuring that it continues to be an effective tool in the battle against web application vulnerabilities.

2. Burp suite

Burp Suite is a popular and effective web application security testing tool made for locating and repairing vulnerabilities in online applications. The proxy, scanner, repeater, intruder, and sequencer are just a few of the crucial security testing components included in this integrated platform. Burp Suite is used by security experts, ethical hackers, and developers to find common online application flaws including Cross Site Scripting (XSS), SQL Injection, and security configuration errors. People who are interested in penetration testing and security assessments appreciate it because of its interactive and straightforward user interface. The tool's rich feature set and frequent upgrades guarantee its leadership in web application security testing and enable the defense of digital assets against potential attacks.


Burp Suite Community Edition v2024.4.5 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Settings
Intercept HTTP history WebSockets history Proxy settings
Forward Drop Intercept is off Action Open browser
Learn more
Open browser
Event log All issues
Memory: 117.3MB

Intercept is off

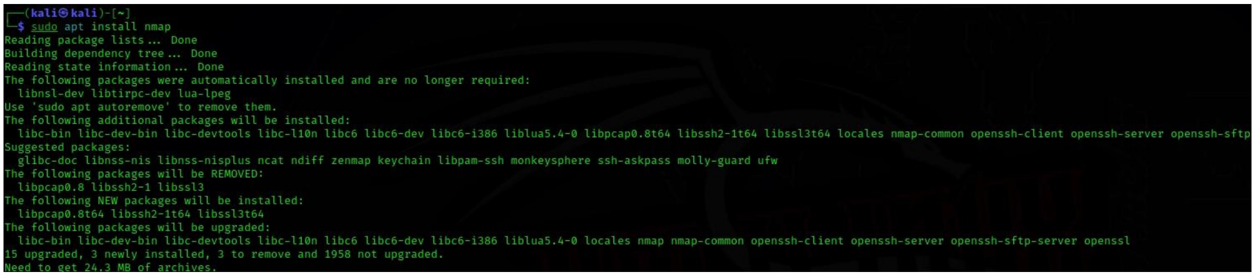
When enabled, requests sent by Burp's browser are held here so that you can analyze and modify them before forwarding them to the target server.

3. Metasploit

A well-known and flexible penetration testing tool, Metasploit helps security experts and ethical hackers evaluate and improve the security of systems and networks. It was created by Rapid7 and offers a full range of tools, exploits, and payloads for finding, using, and fixing vulnerabilities in a controlled setting. With its modular design, Metasploit's extensive library of known vulnerabilities allows for customization and automation of security assessments. Its effectiveness in finding weak spots, presenting actual assault scenarios, and assisting in the creation of strong defense methods is the reason for its appeal. Those that are devoted to protecting digital assets continue to have Metasploit as a key tool in their toolbox.

4. Nmap

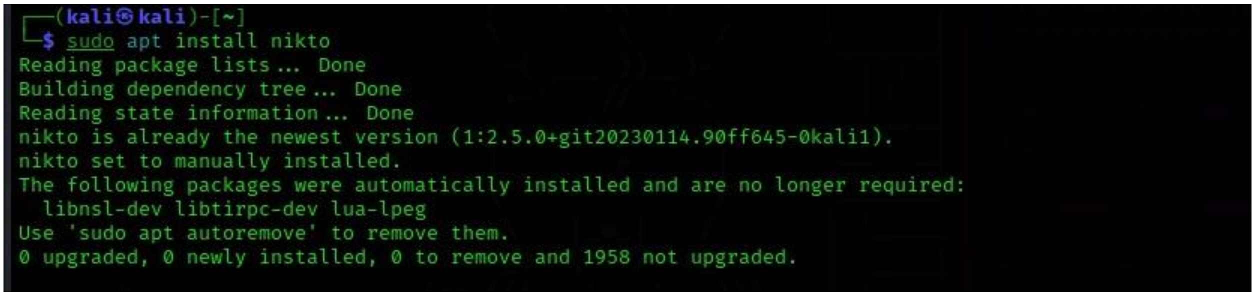
Network administrators and security experts use the well-known open-source network scanning application Nmap, also known as Network Mapper, to find and examine hosts and services on networks. Users may use this flexible tool to map network topologies, find available ports, and acquire crucial data about distant systems. Nmap includes a wide range of scanning methods, such as OS identification and version enumeration, and is very flexible. It helps identify possible vulnerabilities and enables businesses to guarantee the integrity of their network infrastructure, making it a crucial tool for network security evaluations. Nmap is a crucial tool for preserving and strengthening network security because of its ongoing development and community support.



```
┌──(kali㉿kali)-[~]
└─$ sudo apt install nmap
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libnsl-dev libtirpc-dev lua-lpeg
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libc-bin libc-dev-bin libc-devtools libc-l10n libc6 libc6-dev libc6-i386 liblua5.4-0 libpcap0.8t64 libssh2-1t64 libssl3t64 locales nmap-common openssh-client openssh-server openssh-sftp
Suggested packages:
  glibc-doc libnss-nis libnss-nisplus ncat ndiff zenmap keychain libpam-ssh monkeysphere ssh-askpass molly-guard ufw
The following packages will be REMOVED:
  libpcap0.8 libssh2-1 libssl3
The following NEW packages will be installed:
  libpcap0.8t64 libssh2-1t64 libssl3t64
The following packages will be upgraded:
  libc-bin libc-dev-bin libc-devtools libc-l10n libc6 libc6-dev libc6-i386 liblua5.4-0 locales nmap nmap-common openssh-client openssh-server openssh-sftp-server openssl
15 upgraded, 3 newly installed, 3 to remove and 1958 not upgraded.
Need to get 24.3 MB of archives.
```

5. Nikto

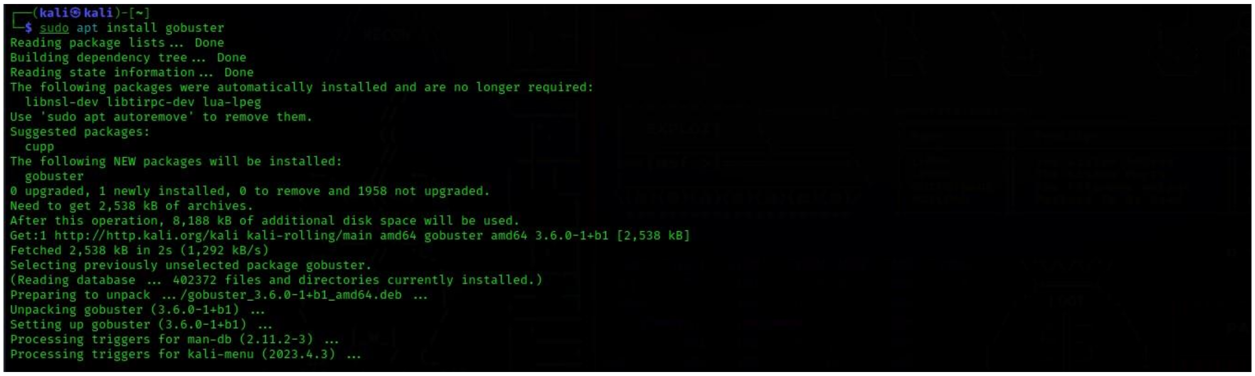
Nikto is a well-known open-source web server vulnerability scanner that helps security experts find possible security problems in web servers and applications. Nikto, created by CIRT (Center for Internet Security), does thorough scans, evaluating a variety of typical online vulnerabilities and misconfigurations, including out-of-date software, known vulnerabilities, and server-specific problems. It is a command-line program that generates thorough results, making it appropriate for both human testing and automated security evaluations. Nikto is a significant addition to any security professional's toolset since it strengthens web server defenses by identifying and fixing vulnerabilities before bad actors can take advantage of them.



```
┌──(kali㉿kali)-[~]
└─$ sudo apt install nikto
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nikto is already the newest version (1:2.5.0+git20230114.90ff645-0kali1).
nikto set to manually installed.
The following packages were automatically installed and are no longer required:
  libnsl-dev libtirpc-dev lua-lpeg
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 1958 not upgraded.
```

6. Go buster

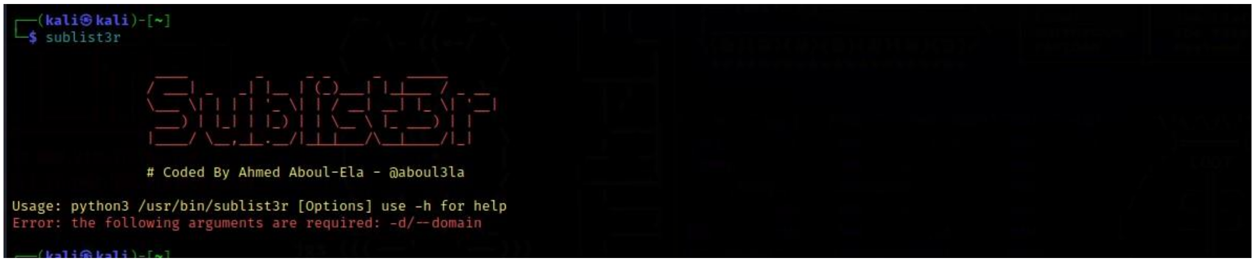
On web servers and web applications, Go buster is a command-line program used for directory and file brute-forcing. By methodically and thoroughly verifying for the existence of folders, files, and subdomains on a target web server, this open-source tool aids security experts and penetration testers in finding hidden or sensitive material. Go buster is a flexible tool for detecting possible security flaws since users may define wordlists, HTTP methods, and other criteria to personalize their scans. Go buster helps to find unlinked resources, incorrect setups, and secret entry points by methodically examining the directory structure of a web server. This results in a more stable and secure environment for web applications.



```
┌──(kali㉿kali)-[~]
└─$ sudo apt install gobuster
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libnsl-dev libtirpc-dev lua-lpeg
Use 'sudo apt autoremove' to remove them.
Suggested packages:
  cupp
The following NEW packages will be installed:
  gobuster
0 upgraded, 1 newly installed, 0 to remove and 1958 not upgraded.
Need to get 2,538 kB of archives.
After this operation, 8,188 kB of additional disk space will be used.
Get:1 http://http.kali.org/kali kali-rolling/main amd64 gobuster amd64 3.6.0-1+b1 [2,538 kB]
Fetched 2,538 kB in 2s (1,292 kB/s)
Selecting previously unselected package gobuster.
(Reading database ... 402372 files and directories currently installed.)
Preparing to unpack .../gobuster_3.6.0-1+b1_amd64.deb ...
Unpacking gobuster (3.6.0-1+b1) ...
Setting up gobuster (3.6.0-1+b1) ...
Processing triggers for man-db (2.11.2-3) ...
Processing triggers for kali-menu (2023.4.3) ...
```

7. Sublist3r

A Python-based open-source program called Sublist3r is made specifically for subdomain enumeration and reconnaissance. It is used by penetration testers and security experts to find the subdomains connected to a target domain, assisting in the discovery of possible security issues and attack vectors. Sublist3r gathers a list of subdomains connected to the target domain using search engines like Google, Bing, and Yahoo as well as numerous DNS databases. This knowledge can be essential for calculating the assault surface and finding hidden assets that could otherwise go unnoticed. For experts working in online application security and penetration testing, Sublist3r is a useful tool because of its simplicity, speed, and interaction with other tools.



```
┌──(kali㉿kali)-[~]
└─$ sublist3r

                 # Coded By Ahmed Aboul-Ela - @aboul3la

Usage: python3 /usr/bin/sublist3r [Options] use -h for help
Error: the following arguments are required: -d/--domain

┌──(kali㉿kali)-[~]
```

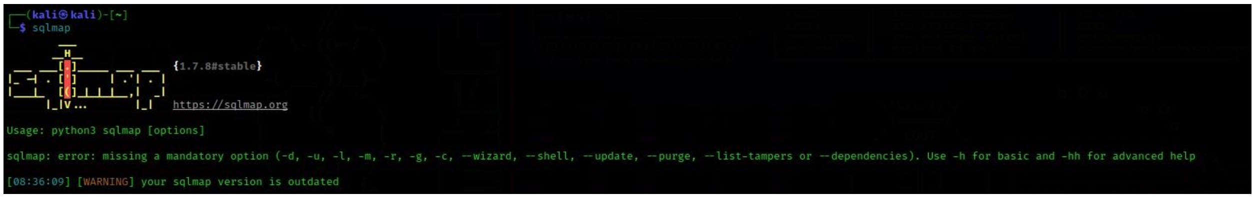
8. Netsparker

Netsparker is a well-known online application security scanner and vulnerability assessment tool that businesses rely on to find and fix web-based security vulnerabilities. This automatic scanner, created by Netsparker Ltd., specializes in identifying a variety of vulnerabilities, such as SQL injection, Cross-Site Scripting (XSS), and more. Because itnot only identifies vulnerabilities but also verifies their presence, Proof-Based Scanning technology from Netsparker avoids false positives. It appeals to developers and security experts alike because to its automation features, user-friendly interface, and connection with development environments. In order for enterprises to properly safeguard their digital assets, Netsparker plays a crucial role in increasing web application security.

netsparker®
web application security scanner

9. SQL Map

SQL Map is a popular open-source penetration testing tool specifically designed for detecting and exploiting SQL injection vulnerabilities in web applications and databases. Developed in Python, it streamlines the process of identifying and assessing SQL injection vulnerabilities, which can be a severe security risk. SQL Map automates the process of fingerprinting the database, enumerating tables, and extracting data, making it a valuable asset for ethical hackers and security professionals. Its extensive feature set and user-friendly command-line interface enable users to uncover and address these vulnerabilities, helping organizations secure their web applications against potentially devastating data breaches and manipulation attacks.



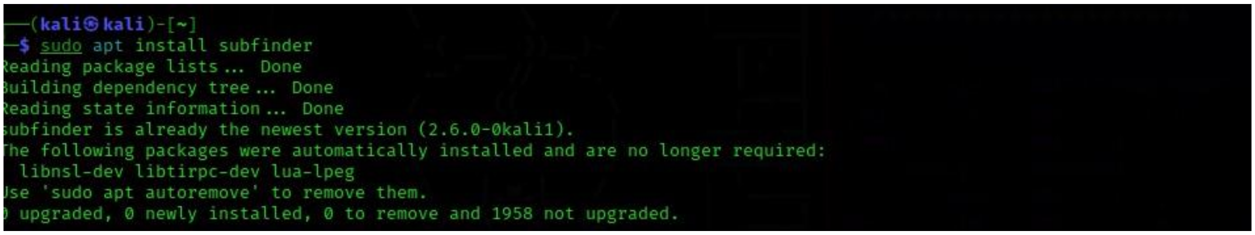
```
┌──(kali㉿kali)-[~]
└─$ sqlmap
                              {1.7.8#stable}

                              https://sqlmap.org

Usage: python3 sqlmap [options]

sqlmap: error: missing a mandatory option (-d, -u, -l, -m, -r, -g, -c, --wizard, --shell, --update, --purge, --list-tampers or --dependencies). Use -h for basic and -hh for advanced help

[08:36:09] [WARNING] your sqlmap version is outdated
```

10. Recon-ng

In the area of cybersecurity and penetration testing, Recon-ng is a potent opensource reconnaissance framework created for data collection and foot printing. It was created in Python and offers security experts and ethical hackers a flexible and expandable platform to gather information about their targets. Using Recon-ng, users may carry out activities like DNS enumeration, web scraping, and open-source intelligence gathering. It provides a wide range of modules and features. It is a useful tool for expediting the reconnaissance part of security assessments due to its versatility and integration capabilities. Recon-ng gives users the ability to better understand the target environment, facilitating more knowledgeable and efficient security testing.

11. subfinder

A well-known and extremely efficient open-source tool for cybersecurity reconnaissance is called Subfinder. For security experts and ethical hackers, it is a vital tool since it specializes in locating subdomains connected to a certain domain name. Subfinder helps to identify possible holes and weak spots in a network's perimeter by methodically searching the internet for subdomains, hence improving overall security. Its adaptability and powerful powers make it a priceless tool in the toolbox of those responsible for safeguarding digital assets and infrastructure. Subfinder is still a crucial tool in the continuing fight against cyber threats since it is always changing and adapting.



```
—(kali㉿kali)-[~]
—$ sudo apt install subfinder
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
subfinder is already the newest version (2.6.0-0kali1).
The following packages were automatically installed and are no longer required:
  libnsl-dev libtirpc-dev lua-lpeg
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 1958 not upgraded.
```

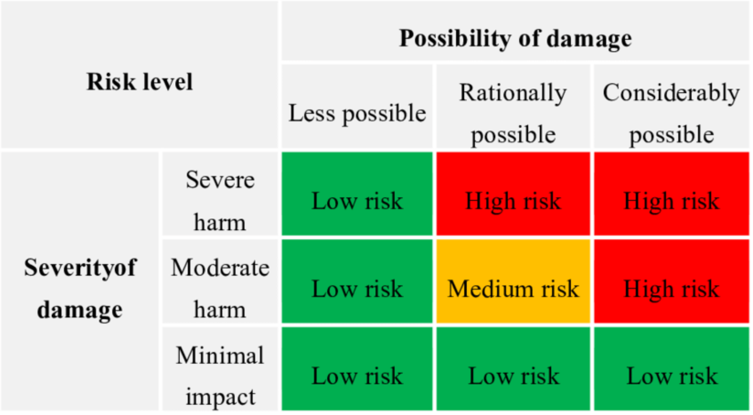
12. Masscan

Masscan is a fast and flexible network scanning tool used for extensive host finding and port scanning. This open-source tool is renowned for its astounding speed and can scan enormous IP address ranges in a fraction of the time it takes other scanners. Security experts and penetration testers particularly like Masscan because of how well it can find open ports, which is essential for determining network vulnerabilities. Asynchronous Transmission Control Protocol (TCP) connections are used throughout operation, allowing for quick scans with less resource use. For network reconnaissance and security evaluations in both offensive and defensive cybersecurity situations, Masscan's speed and scalability make it a priceless tool.

13. Whois

A key source of information for domain names and the internet is the WHOIS tool. It enables users to get vital information about domain names, IP addresses, and autonomous system numbers (ASNs). By using WHOIS, it is possible to discover details about a domain's ownership, registration dates, name servers, and registrar. In domain management, network troubleshooting, and cybersecurity, it is essential. Inquiries into concerns may be investigated, domain availability can be checked, and assistance with legal or security-related questions can be provided by people, companies, and cybersecurity specialists by searching this vast database to learn more about the organizations behind web addresses and IP allocations. In order to go about the digital world, WHOIS is still a valuable resource.

Risk level information



Risk level		Possibility of damage		
		Less possible	Rationally possible	Considerably possible
Severityof damage	Severe harm	Low risk	High risk	High risk
	Moderate harm	Low risk	Medium risk	High risk
	Minimal impact	Low risk	Low risk	Low risk

Low: The low-risk level denotes the danger that is least likely to be connected to a certain vulnerability. Gaining knowledge about the web application that was not meant to be known otherwise may result from this.

Medium: The medium risk level denotes a significant risk in conjunction with a particular vulnerability. An attacker can obtain low-level information about the program by taking advantage of a medium vulnerability. Medium risk vulnerabilities should be addressed after high-risk vulnerabilities.

High: The high-risk rating displays the greatest danger connected to a particular vulnerability. The target application can be effectively exploited by an attacker, and the application data may be compromised partially or whole. An attacker may cause the web application's data to be modified or deleted.

Daily Journal-Day (02.05.2024)

This phase is the vast area of the bug bounty program. I had to spend several hours learning how to find new bugs and how to use the tools to find the bugs. With the limited time had it was so much harder to allocate time for my other work.

Objectives :

- How to use the tools property.
- How to use automated tools.
- How to identify vulnerabilities.
- Find open ports.
- Find subdomains.
- Find vulnerabilities.
- Create reports.

Sources used:

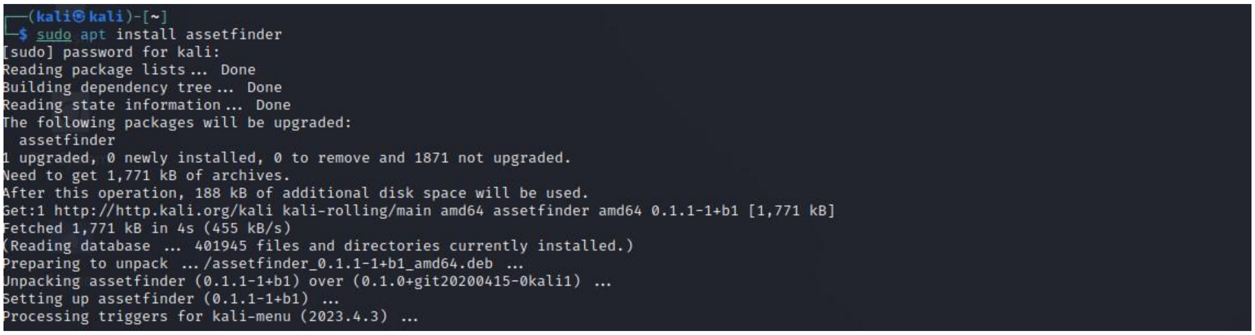
- Youtube.
- Portswigger.
- Hackerone.
- Linkedin.
- Facebook (bug bounty groups)
- Google.

1) Report 1(Expedia Group)

Company name	Expedia group
Website	https://www.expediagroup.com/
Ip Address	162.159.129.11
Platform	Hackerone

Reconnaissance

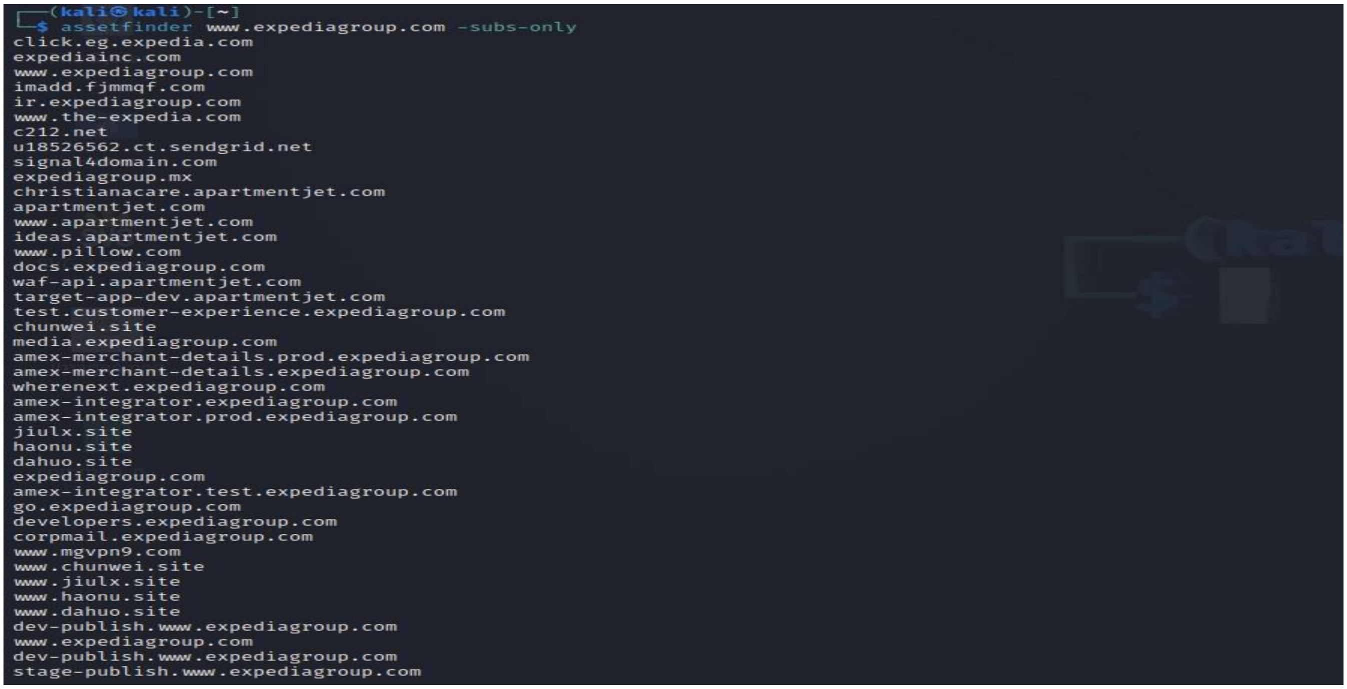
Firstly, I used the Assetfinder tool to find the subdomains of the Expedia Group website. Assetfinder tool is used to find the subdomains of a website. To install Assetfinder we must use the command.



```
┌──(kali㉿kali)-[~]
└─$ sudo apt install assetfinder
[sudo] password for kali:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following packages will be upgraded:
  assetfinder
1 upgraded, 0 newly installed, 0 to remove and 1871 not upgraded.
Need to get 1,771 kB of archives.
After this operation, 188 kB of additional disk space will be used.
Get:1 http://http.kali.org/kali kali-rolling/main amd64 assetfinder amd64 0.1.1-1+b1 [1,771 kB]
Fetched 1,771 kB in 4s (455 kB/s)
(Reading database ... 401945 files and directories currently installed.)
Preparing to unpack .../assetfinder_0.1.1-1+b1_amd64.deb ...
Unpacking assetfinder (0.1.1-1+b1) over (0.1.0+git20200415-0kali1) ...
Setting up assetfinder (0.1.1-1+b1) ...
Processing triggers for kali-menu (2023.4.3) ...
```

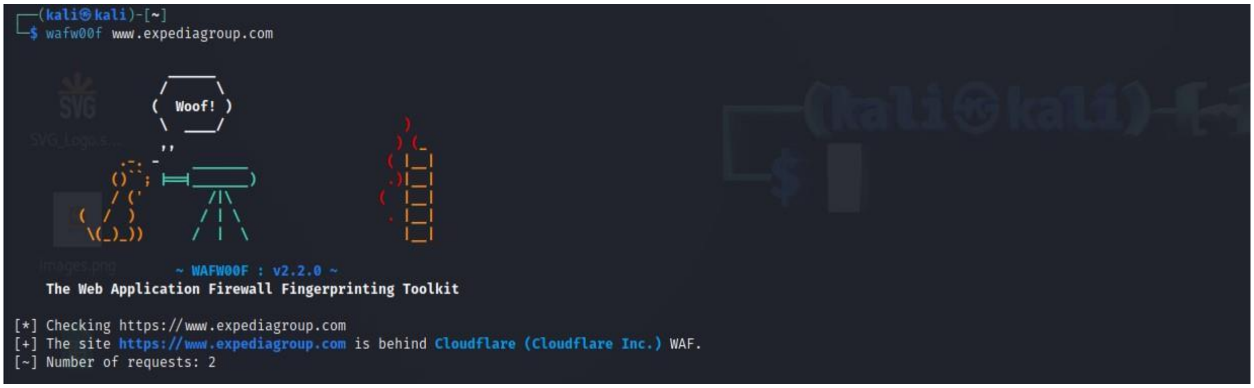
To find the subdomains, you must do as below;

Use the code -sub-only when given the command to find the subdomains. That code is used to search the subdomains only.



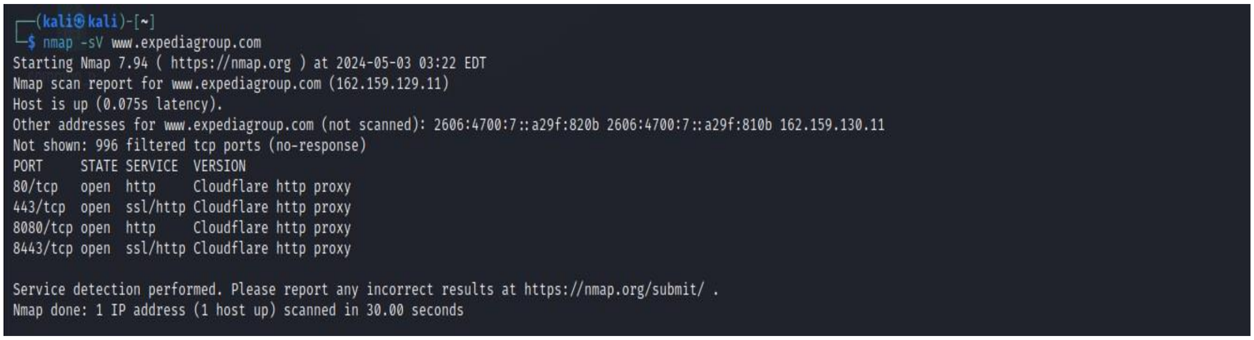
```
┌──(kali㉿kali)-[~]
└─$ assetfinder www.expediagroup.com -subs-only
click.eg.expedia.com
expediainc.com
www.expediagroup.com
imadd.fjmmqf.com
ir.expediagroup.com
www.the-expedia.com
c212.net
u18526562.ct.sendgrid.net
signal4domain.com
expediagroup.mx
christianacare.apartmentjet.com
apartmentjet.com
www.apartmentjet.com
ideas.apartmentjet.com
www.pillow.com
docs.expediagroup.com
waf-api.apartmentjet.com
target-app-dev.apartmentjet.com
test.customer-experience.expediagroup.com
chunwei.site
media.expediagroup.com
amex-merchant-details.prod.expediagroup.com
amex-merchant-details.expediagroup.com
wherenext.expediagroup.com
amex-integrator.expediagroup.com
amex-integrator.prod.expediagroup.com
jiulx.site
haonu.site
dahuo.site
expediagroup.com
amex-integrator.test.expediagroup.com
go.expediagroup.com
developers.expediagroup.com
corpmail.expediagroup.com
www.mgvpn9.com
www.chunwei.site
www.jiulx.site
www.haonu.site
www.dahuo.site
dev-publish.www.expediagroup.com
www.expediagroup.com
dev-publish.www.expediagroup.com
stage-publish.www.expediagroup.com
```

I also found that Expedia Group website is running behind a firewall **“Cloudflare (Cloudflare Inc.)”**. To find the firewall I used the **wafw00f** tool. The commands and the results are show from the below.



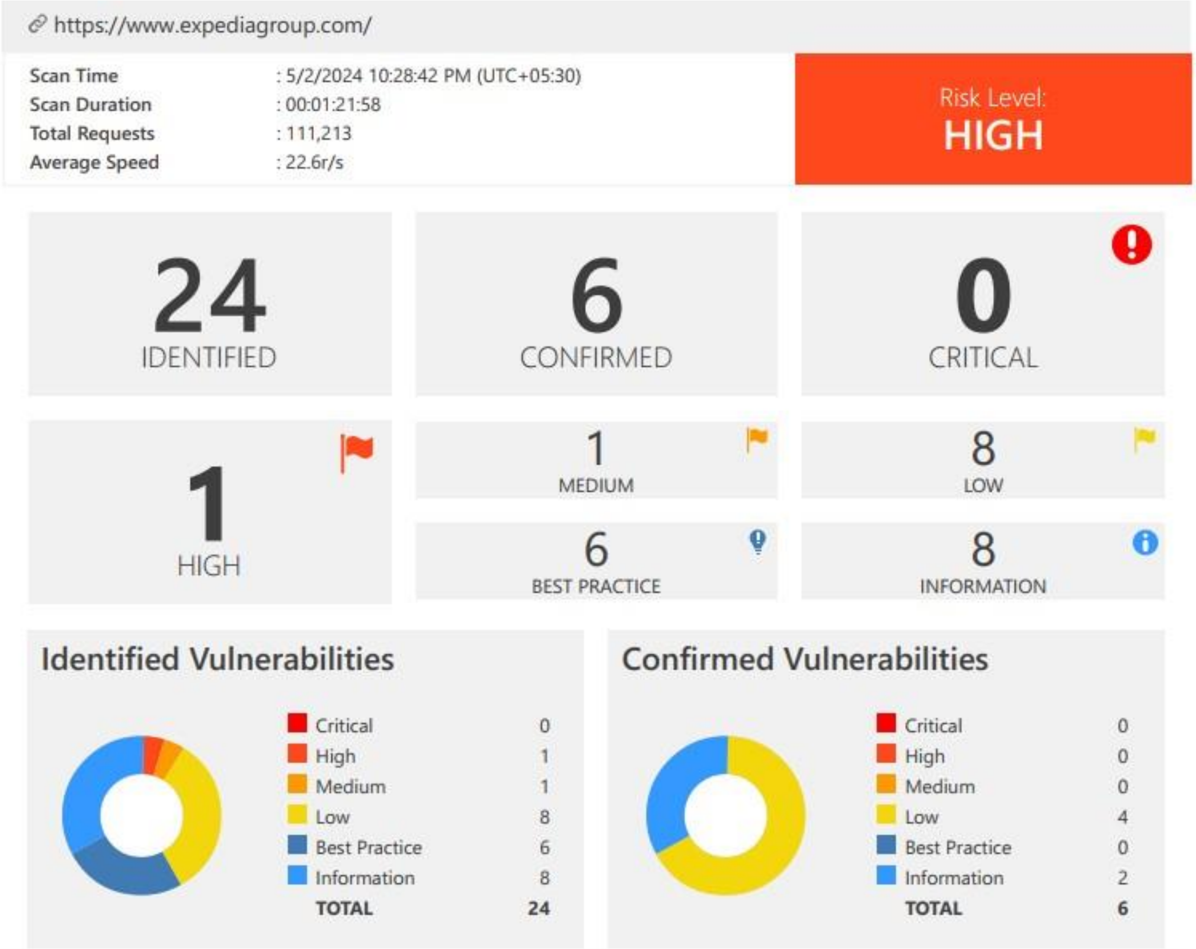
```
┌──(kali㉿kali)-[~]
└─$ wafw00f www.expediagroup.com

                ______
               /      \
              (  Woof! )
               \  ____/                      )
               ,,                           ) (_
           .-. -    _______                ( |__|
          ()``; |==|_______)                .)|__|
          / ('        /|\                  (  |__|
      (  /  )        / | \                  . |__|
       \(_)_))      /  |  \                   |__|

                    ~ WAFW00F : v2.2.0 ~
    The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://www.expediagroup.com
[+] The site https://www.expediagroup.com is behind Cloudflare (Cloudflare Inc.) WAF.
[~] Number of requests: 2
```

I used the **Nmap tool** find whether there are any unusual ports are opened but there weren’t any unusual ports opened.



```
┌──(kali㉿kali)-[~]
└─$ nmap -sV www.expediagroup.com
Starting Nmap 7.94 ( https://nmap.org ) at 2024-05-03 03:22 EDT
Nmap scan report for www.expediagroup.com (162.159.129.11)
Host is up (0.075s latency).
Other addresses for www.expediagroup.com (not scanned): 2606:4700:7::a29f:820b 2606:4700:7::a29f:810b 162.159.130.11
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE SERVICE  VERSION
80/tcp   open  http     Cloudflare http proxy
443/tcp  open  ssl/http Cloudflare http proxy
8080/tcp open  http     Cloudflare http proxy
8443/tcp open  ssl/http Cloudflare http proxy

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 30.00 seconds
```

Then I used the netsparker tool to scan the website to find the vulnerabilities. Through the website I found out several vulnerabilities as below.



https://www.expediagroup.com/

Scan Time	: 5/2/2024 10:28:42 PM (UTC+05:30)
Scan Duration	: 00:01:21:58
Total Requests	: 111,213
Average Speed	: 22.6r/s

Risk Level:
HIGH

24 IDENTIFIED

6 CONFIRMED

0 CRITICAL

1 HIGH

1 MEDIUM

8 LOW

6 BEST PRACTICE

8 INFORMATION

Identified Vulnerabilities

🟥 Critical	0
🟧 High	1
🟧 Medium	1
🟨 Low	8
🟦 Best Practice	6
🟦 Information	8
TOTAL	**24**

Confirmed Vulnerabilities

🟥 Critical	0
🟧 High	0
🟧 Medium	0
🟨 Low	4
🟦 Best Practice	0
🟦 Information	2
TOTAL	**6**

The Risk level of the website was **high**.

Vulnerabilities found.

Vulnerability title: Out of data version (Highcharts)(A9)

Risk Level – High

Method – GET

Description

The "Out-of-date Version (Highcharts)" vulnerability refers to a security issue arising from the use of an outdated version of the Highcharts library in a web application. Highcharts is a popular JavaScript library used for creating interactive charts and graphs on websites and web applications.

Solutions

To mitigate the "Out-of-date Version (Highcharts)" vulnerability, web developers and administrators should consider the following best practices:

I. Update Highcharts Frequently: Make sure the web application is running the most recent stable version of Highcharts. Check the Highcharts development team's website frequently for updates and security fixes, and apply them to the application as necessary.

II. Use Content Delivery Networks (CDNs): Consider serving Highcharts from a reputable CDN (Content Delivery Network) such as Highcharts' own CDN or other popular CDN providers. CDNs often provide automatic updates and caching benefits, reducing the maintenance overhead of managing Highcharts versions.

III. Static Analysis Tools: Use static code analysis tools to scan the application codebase for outdated Highcharts versions and other vulnerable dependencies. These tools can help identify security vulnerabilities and outdated libraries that need to be updated.

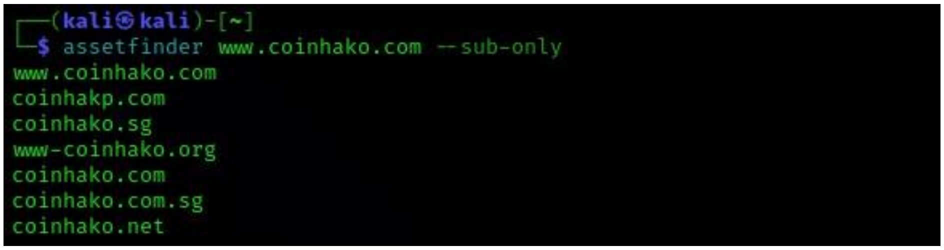
IV. Security Audits and Testing: Perform frequent security audits and penetration testing on the online application to identify and address vulnerabilities, especially those related to outdated versions of Highcharts. Examine the program for common security holes like data injection, cross-site scripting, and other client-side weaknesses.

2) Report 2(Coinhako)

Company name	coinhako
website	https://coinhako.com/
Ip address	104.18.2.84
Platform	hackerone

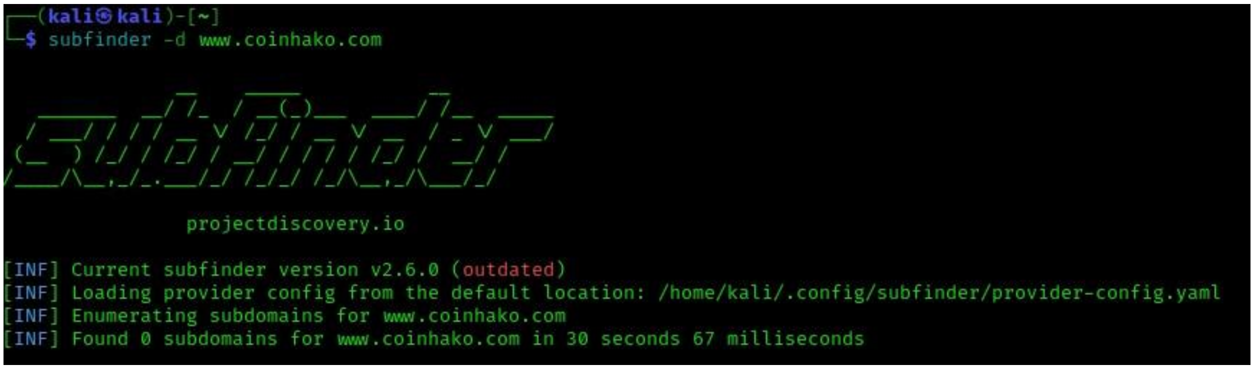
Reconnaissance

As the first step I used the assetfinder tool for acanning the sub domains of the website.



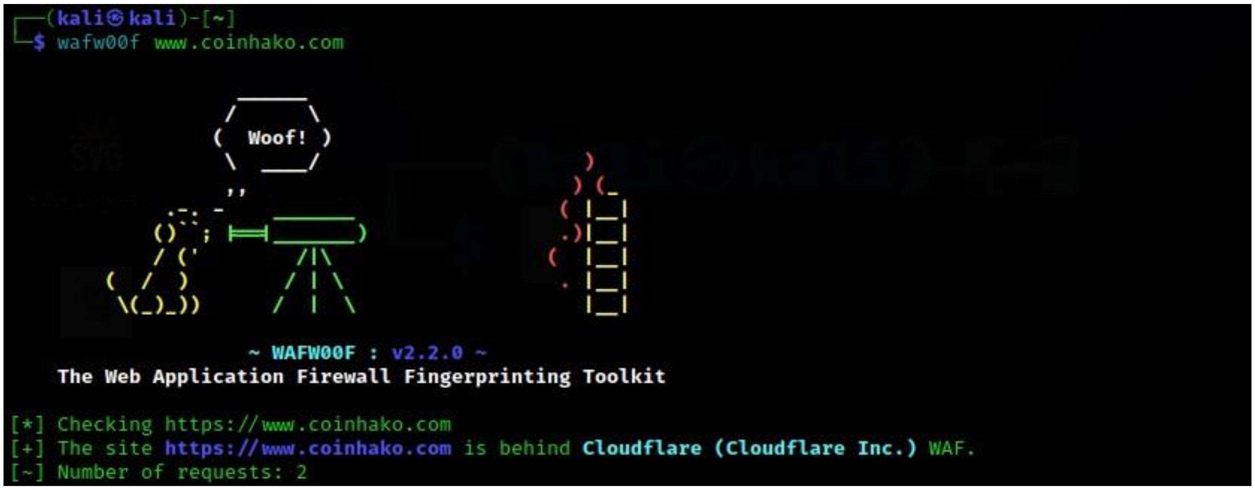
```
┌──(kali㉿kali)-[~]
└─$ assetfinder www.coinhako.com --sub-only
www.coinhako.com
coinhakp.com
coinhako.sg
www-coinhako.org
coinhako.com
coinhako.com.sg
coinhako.net
```

I was able to find a smaller number of subdomains using the assetfinder tool. I also used the subfinder tool to find the subdomains, but I was not able to find any subdomain from it



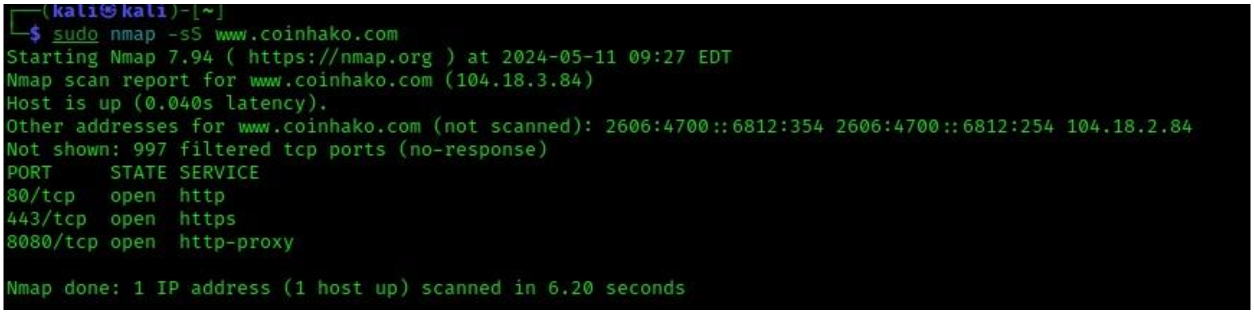
```
┌──(kali㉿kali)-[~]
└─$ subfinder -d www.coinhako.com

               __    _____           __
   _______  __/ /_  / __(_)___  ____/ /__  _____
  / ___/ / / / __ \/ /_/ / __ \/ __  / _ \/ ___/
 (__  ) /_/ / /_/ / __/ / / / / /_/ /  __/ /
/____/\__,_/_.___/_/ /_/_/ /_/\__,_/\___/_/

                projectdiscovery.io

[INF] Current subfinder version v2.6.0 (outdated)
[INF] Loading provider config from the default location: /home/kali/.config/subfinder/provider-config.yaml
[INF] Enumerating subdomains for www.coinhako.com
[INF] Found 0 subdomains for www.coinhako.com in 30 seconds 67 milliseconds
```

I checked whether the website is using a Firewall using the wafw00f tool. It provided that the website is behind the Cloudfare (Cloudfare Inc.) Web Application Firewall (WAF).



```
┌──(kali㉿kali)-[~]
└─$ wafw00f www.coinhako.com

                ~ WAFW00F : v2.2.0 ~
    The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://www.coinhako.com
[+] The site https://www.coinhako.com is behind Cloudflare (Cloudflare Inc.) WAF.
[~] Number of requests: 2
```

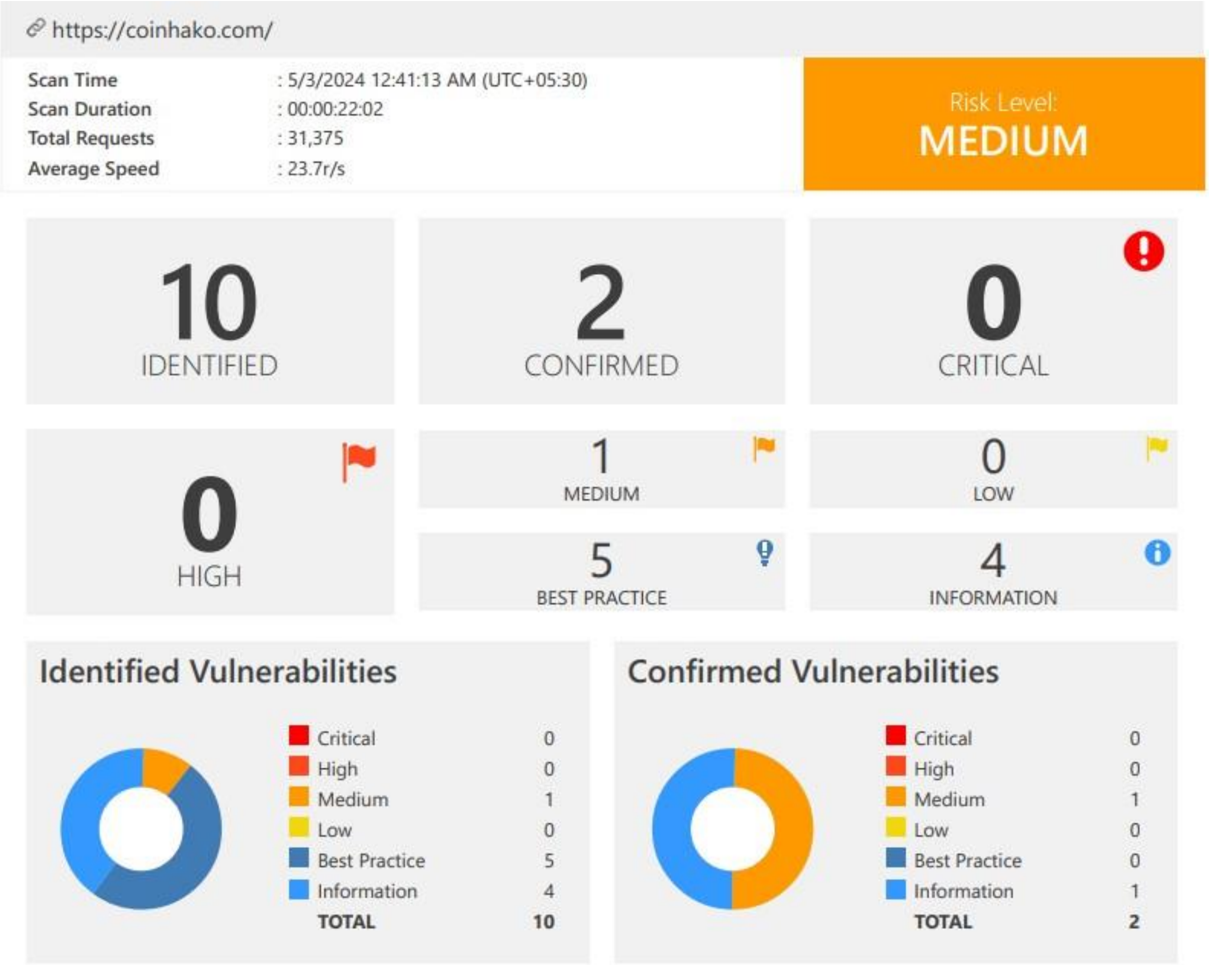
I used the nmap tool to find the open ports of the website www.coinhako.com .



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS www.coinhako.com
Starting Nmap 7.94 ( https://nmap.org ) at 2024-05-11 09:27 EDT
Nmap scan report for www.coinhako.com (104.18.3.84)
Host is up (0.040s latency).
Other addresses for www.coinhako.com (not scanned): 2606:4700::6812:354 2606:4700::6812:254 104.18.2.84
Not shown: 997 filtered tcp ports (no-response)
PORT     STATE SERVICE
80/tcp   open  http
443/tcp  open  https
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 6.20 seconds
```

Vulnerabilities found.

Using the Invicti tool I scanned the website to find the vulnerabilities of it. I found several vulnerabilities that belong to the owasp top 10 catergory. The overall risk level of the website was Medium.



https://coinhako.com/

Scan Time : 5/3/2024 12:41:13 AM (UTC+05:30)
Scan Duration : 00:00:22:02
Total Requests : 31,375
Average Speed : 23.7r/s

Risk Level:
MEDIUM

10 IDENTIFIED

2 CONFIRMED

0 CRITICAL

0 HIGH

1 MEDIUM

0 LOW

5 BEST PRACTICE

4 INFORMATION

Identified Vulnerabilities

🟥 Critical	0
🟧 High	0
🟧 Medium	1
🟨 Low	0
🟦 Best Practice	5
🟦 Information	4
TOTAL	**10**

Confirmed Vulnerabilities

🟥 Critical	0
🟧 High	0
🟧 Medium	1
🟨 Low	0
🟦 Best Practice	0
🟦 Information	1
TOTAL	**2**

Vulnerability title: Weak ciphers enabled(A3)

Risk level – medium

Method – GET

OWASP TOP 10 – (A3)-2017

Description

Cryptographic algorithms deemed insecure because of flaws or deficiencies in their design are referred to as weak ciphers. When weak ciphers are enabled in a system's cryptographic setup, there might be serious security issues since attackers may use them to intercept or alter encrypted data.

Impact assessment

The Weak Cipher Enabled vulnerability under the OWASP category on https://coinhako.com has led to the discovery of a number of potential security vulnerabilities because of the use of insufficient encryption protocols or ciphers. The following possible outcomes are all included in the significant impact of ignoring this vulnerability:

I. **Loss of User Trust**: Consumers put their trust in e-commerce sites like Traffic Factory to safeguard their private information. This confidence is undermined by weak ciphers that expose users' information to dangers. Echo Box's credibility and reputation may suffer greatly as a result, and consumer loyalty and trust may drop.

II. **Data Breaches**: Data breaches are more likely when ciphers are weak. Attackers may obtain sensitive user data if they are able to take advantage of these vulnerabilities, which could result in data breaches and the related financial and legal repercussions.

III. **Enhanced Attack Surface**: Cybercriminals can more easily exploit vulnerabilities and obtain illegal access to a website, its servers, and underlying systems when weak ciphers provide them a larger attack surface

IV. **Man-in-the-Middle Attacks**: Vulnerabilities in ciphers allow attackers to intercept and modify data that users exchange with websites. This poses serious security issues since it might result in data alteration and unauthorized access.

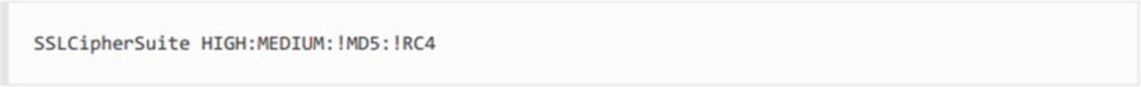
V. **Data Breaches**: Weak ciphers can break encryption, exposing private customer information and perhaps resulting in legal and financial repercussions for Echo Box.

When weak ciphers are enabled in a system's cryptographic setup, the system becomes vulnerable to a number of security risks, such as sensitive data interception, unauthorized access, and data manipulation. Disabling weak ciphers and sticking to strong cryptographic algorithms that have been security-verified by cryptography specialists is advised.

You should check and change the cryptographic configuration settings to make sure that only strong and secure ciphers are enabled in order to address the problem of weak ciphers being enabled in a system. This might entail switching to more secure current cryptographic algorithms and disabling insecure ciphers via configuring web servers, SSL/TLS implementations, and other network services. Frequent vulnerability scans and security audits can assist in locating and fixing problems with weak ciphers.

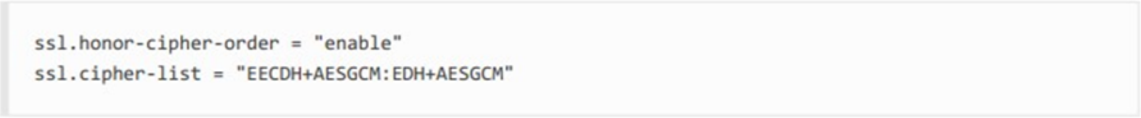
Steps to reproduce

1. For Apache, you should modify the SSLCipherSuite directive in the httpd.conf.



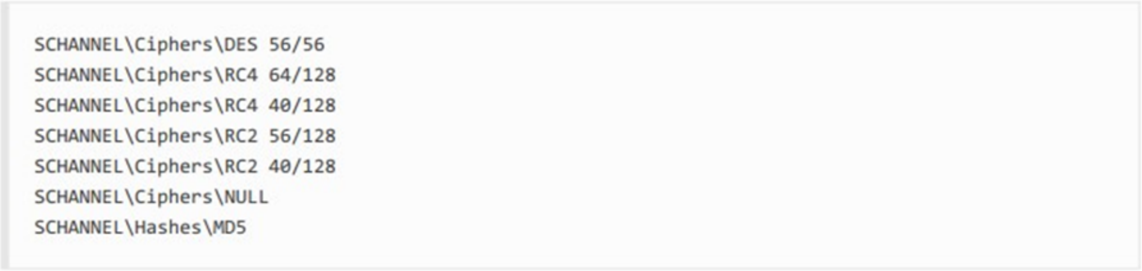
```
SSLCipherSuite HIGH:MEDIUM:!MD5:!RC4
```

2. Lighttpd:



```
ssl.honor-cipher-order = "enable"
ssl.cipher-list = "EECDH+AESGCM:EDH+AESGCM"
```

3. For Microsoft IIS, you should make some changes to the system registry. Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.
    a) Click Start, click Run, type regedt32or type regedit, and then click OK
    b) In Registry Editor, locate the following registry key: HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders
    c) Set "Enabled" DWORD to "0x0" for the following registry keys:



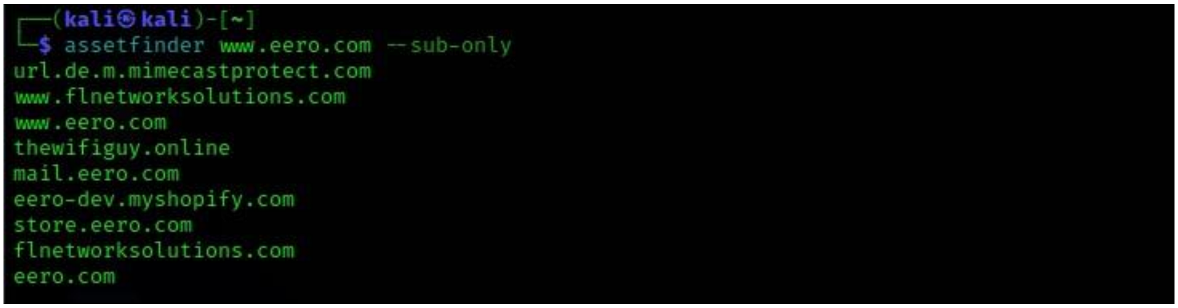
```
SCHANNEL\Ciphers\DES 56/56
SCHANNEL\Ciphers\RC4 64/128
SCHANNEL\Ciphers\RC4 40/128
SCHANNEL\Ciphers\RC2 56/128
SCHANNEL\Ciphers\RC2 40/128
SCHANNEL\Ciphers\NULL
SCHANNEL\Hashes\MD5
```

3) Report 3(eero)

Company name	eero
Website	www.eero.com
Ip address	44.240.38.203
platform	Hackerone

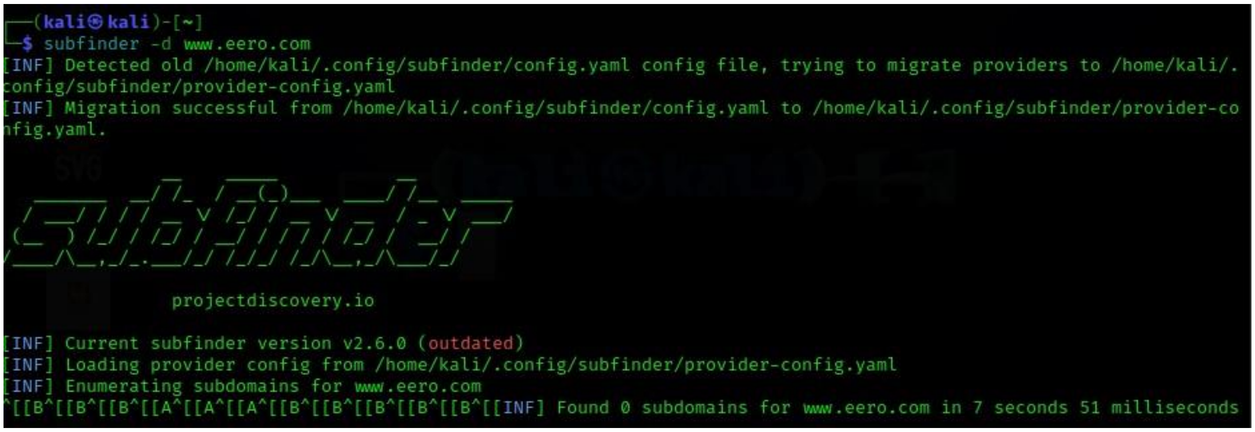
Reconnaissance

As the first step I used the assetfinder tool for acanning the sub domains of the website.



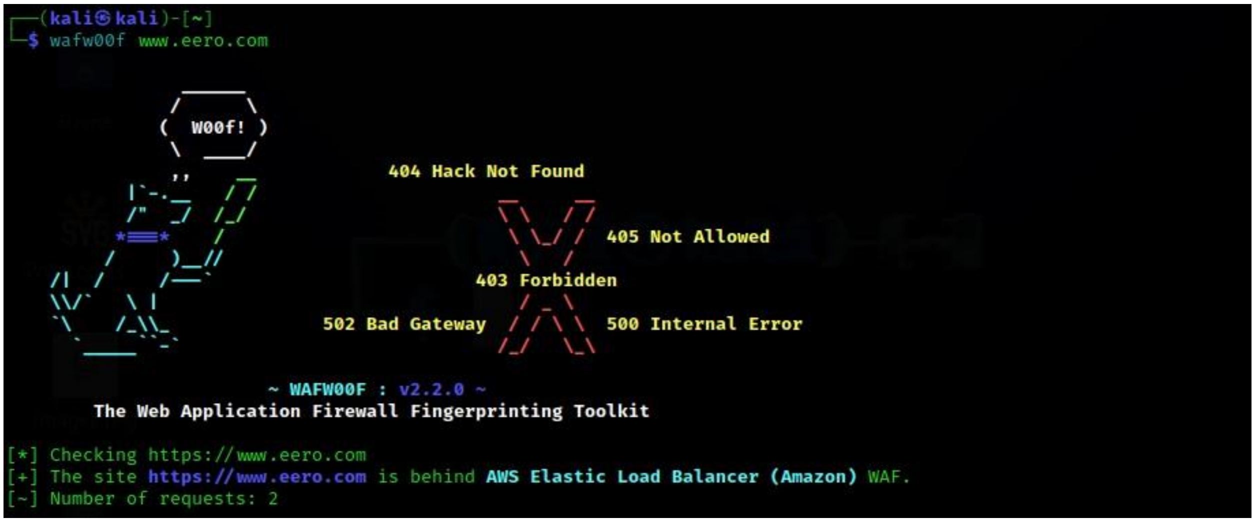
```
┌──(kali㉿kali)-[~]
└─$ assetfinder www.eero.com --sub-only
url.de.m.mimecastprotect.com
www.flnetworksolutions.com
www.eero.com
thewifiguy.online
mail.eero.com
eero-dev.myshopify.com
store.eero.com
flnetworksolutions.com
eero.com
```

I was able to find a smaller number of subdomains using the assetfinder tool. I also used the subfinder tool to find the subdomains, but I was not able to find any subdomain from it



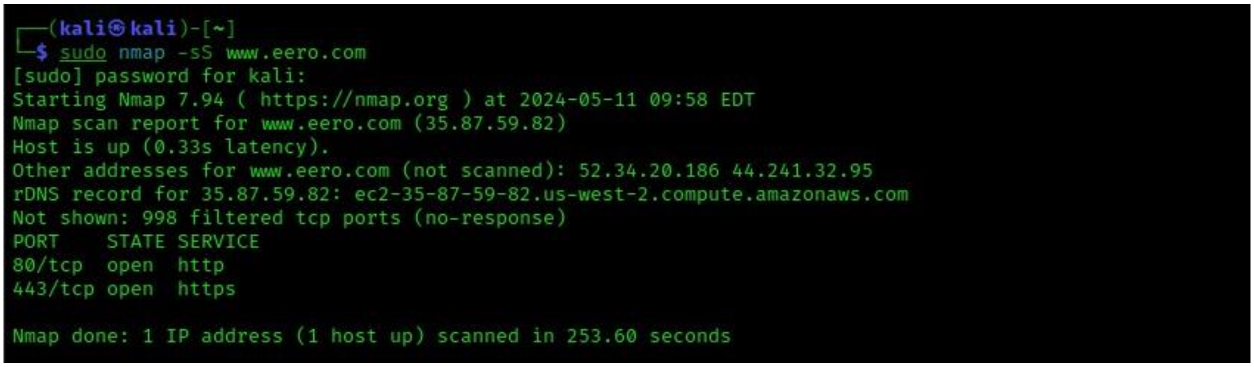
```
┌──(kali㉿kali)-[~]
└─$ subfinder -d www.eero.com
[INF] Detected old /home/kali/.config/subfinder/config.yaml config file, trying to migrate providers to /home/kali/.
config/subfinder/provider-config.yaml
[INF] Migration successful from /home/kali/.config/subfinder/config.yaml to /home/kali/.config/subfinder/provider-co
nfig.yaml.

               __    _____           __
   _______  __/ /_  / __(_)___  ____/ /__  _____
  / ___/ / / / __ \/ /_/ / __ \/ __  / _ \/ ___/
 (__  ) /_/ / /_/ / __/ / / / / /_/ /  __/ /
/____/\__,_/_.___/_/ /_/_/ /_/\__,_/\___/_/

                projectdiscovery.io

[INF] Current subfinder version v2.6.0 (outdated)
[INF] Loading provider config from /home/kali/.config/subfinder/provider-config.yaml
[INF] Enumerating subdomains for www.eero.com
^[[B^[[B^[[B^[[A^[[A^[[A^[[B^[[B^[[B^[[B^[[B^[[INF] Found 0 subdomains for www.eero.com in 7 seconds 51 milliseconds
```

I checked whether the website is using a Firewall using the wafw00f tool. It provided that the website is behind the AWS Elastic load balancer (Amazon) Web Application Firewall (WAF).



```
┌──(kali㉿kali)-[~]
└─$ wafw00f www.eero.com

                ______
               /      \
              (  W00f! )
               \  ____/
               ,,    __            404 Hack Not Found
           |`-.__   / /                      __     __
           /"  _/  /_/                       \ \   / /
          *===*    /                          \ \_/ /  405 Not Allowed
         /     )__//                           \   /
    /|  /     /---`                        403 Forbidden
    \\/`   \ |                                 / \
    `\    /_\\_              502 Bad Gateway  / / \ \  500 Internal Error
      `_____``-`                             /_/   \_\

                        ~ WAFW00F : v2.2.0 ~
        The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://www.eero.com
[+] The site https://www.eero.com is behind AWS Elastic Load Balancer (Amazon) WAF.
[~] Number of requests: 2
```

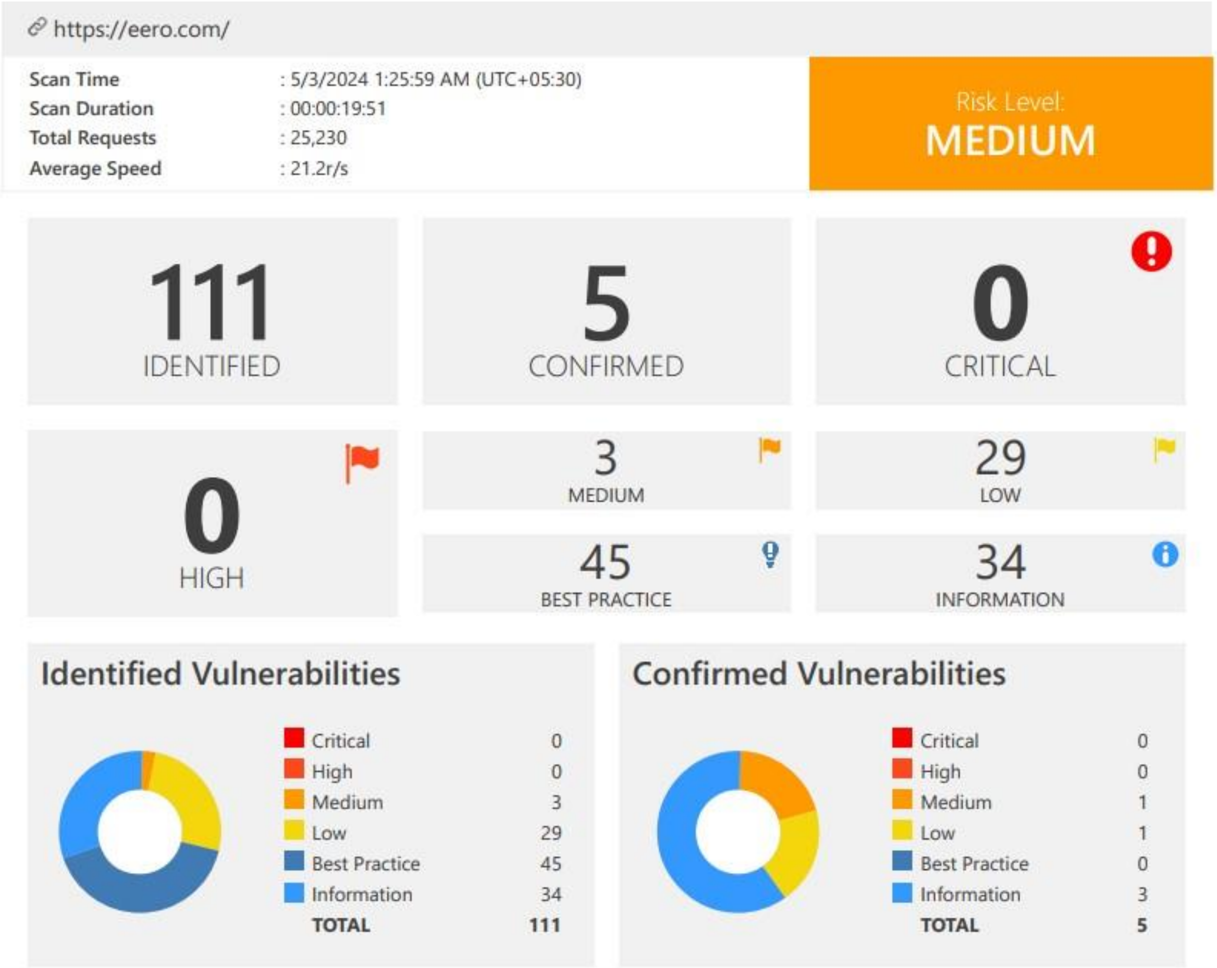
I used the nmap tool to find the open ports of the website www.eero.com .



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS www.eero.com
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2024-05-11 09:58 EDT
Nmap scan report for www.eero.com (35.87.59.82)
Host is up (0.33s latency).
Other addresses for www.eero.com (not scanned): 52.34.20.186 44.241.32.95
rDNS record for 35.87.59.82: ec2-35-87-59-82.us-west-2.compute.amazonaws.com
Not shown: 998 filtered tcp ports (no-response)
PORT    STATE SERVICE
80/tcp  open  http
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 253.60 seconds
```

Vulnerabilities found.

Using the Invicti tool I scanned the website to find the vulnerabilities of it. I found several vulnerabilities that belong to the owasp top 10 catergory. The overall risk level of the website was Medium.



https://eero.com/

Scan Time	: 5/3/2024 1:25:59 AM (UTC+05:30)
Scan Duration	: 00:00:19:51
Total Requests	: 25,230
Average Speed	: 21.2r/s

Risk Level:
MEDIUM

111 IDENTIFIED

5 CONFIRMED

0 CRITICAL

0 HIGH

3 MEDIUM

29 LOW

45 BEST PRACTICE

34 INFORMATION

Identified Vulnerabilities

🟥 Critical	0
🟥 High	0
🟧 Medium	3
🟨 Low	29
🟦 Best Practice	45
🟦 Information	34
TOTAL	**111**

Confirmed Vulnerabilities

🟥 Critical	0
🟥 High	0
🟧 Medium	1
🟨 Low	1
🟦 Best Practice	0
🟦 Information	3
TOTAL	**5**

Vulnerability title: [POSSIBLE] BREACH attack detected

Risk level – medium

Method – GET

OWASP TOP 10 – (A9)-2017

Description

The BREACH attack is a security vulnerability that targets web applications and leverages the HTTP compression feature to perform plaintext recovery of sensitive information, such as authentication tokens, CSRF tokens, and other secrets transmitted over HTTPS. BREACH stands for "Browser Reconnaissance and Exfiltration via Adaptive Compression of Hypertext

Impact assessment

Here's how the BREACH attack works:

I. **HTTP compression**: Many web servers reduce the size of HTTP replies using compression methods like Gzip or Deflate, which enhances performance and uses less bandwidth.

II. **Adaptive compression**: BREACH takes advantage of the fact that some parts of an HTTP response may change dynamically based on user input, such as CSRF tokens, session identifiers, or other secrets. By observing changes in the compressed response size, an attacker can infer information about the plaintext contents.

III. **Repetitive patterns**: The attacker injects specially crafted payloads into the victim's browser, causing the web application to generate predictable patterns in the HTTP response. By observing changes in the compressed response size caused by these injected payloads, the attacker can deduce information about the plaintext secrets.

IV. **Plaintext recovery**: By repeatedly injecting payloads and monitoring changes in the compressed response size, the attacker can gradually recover sensitive information transmitted over HTTPS, such as authentication tokens or other secrets.

The BREACH attack is a variant of the CRIME attack and requires the attacker to have a man-in-the-middle position or another way to observe the victim's HTTPS traffic. It can be mitigated by disabling HTTP compression, randomizing secrets to prevent predictable patterns, or implementing countermeasures such as padding or rate limiting.

Solutions

Netsparker reported a Possible BREACH Attack issue because the target web page meets the following conditions that facilitate it.

- If possible, disable HTTP level compression.
- Reflects user-input in the HTTP response bodies.
- Contains sensitive information (such as a CSRF token) in HTTP response bodies.

To mitigate the issue, we recommend the following solutions:

I. If possible, disable HTTP level compression.

II. Separate sensitive information from user input.

III. Protect vulnerable pages with CSRF token. The SameSite Cookie attribute will mitigate this issue, because to exploit this issue an attacker forces the victim to visit a target website using invisible frames.With the SameSite cookie attribute added, cookies that belong to the target won't be sent with a request that does not include top level navigation.

IV. Hide the length of the traffic by adding a random number of bytes to the responses.

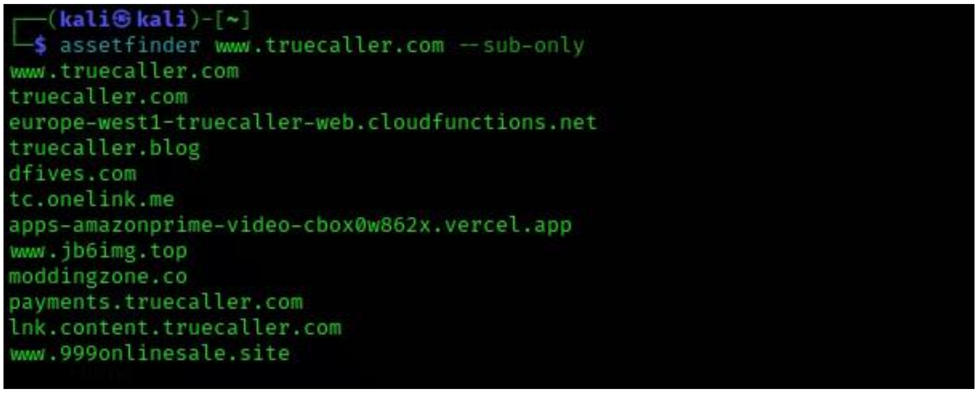
V. Add in a rate limit, so that the page maximum is reached five times per minute.

4) Report 4(truecaller)

Company name	Truecaller
website	www.truecaller.com
Ip address	194.9.94.86
platform	hackerone

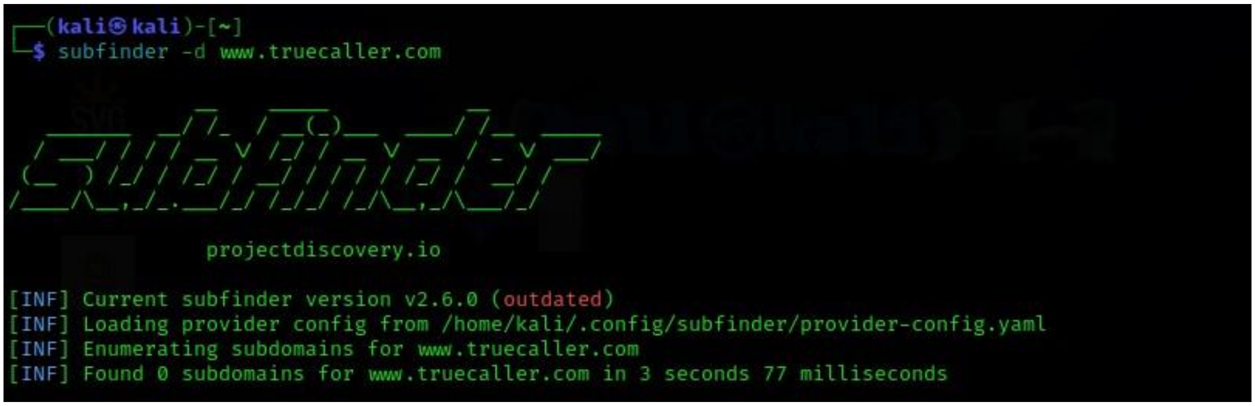
Reconnaissance

As the first step I used the assetfinder tool for acanning the sub domains of the website.



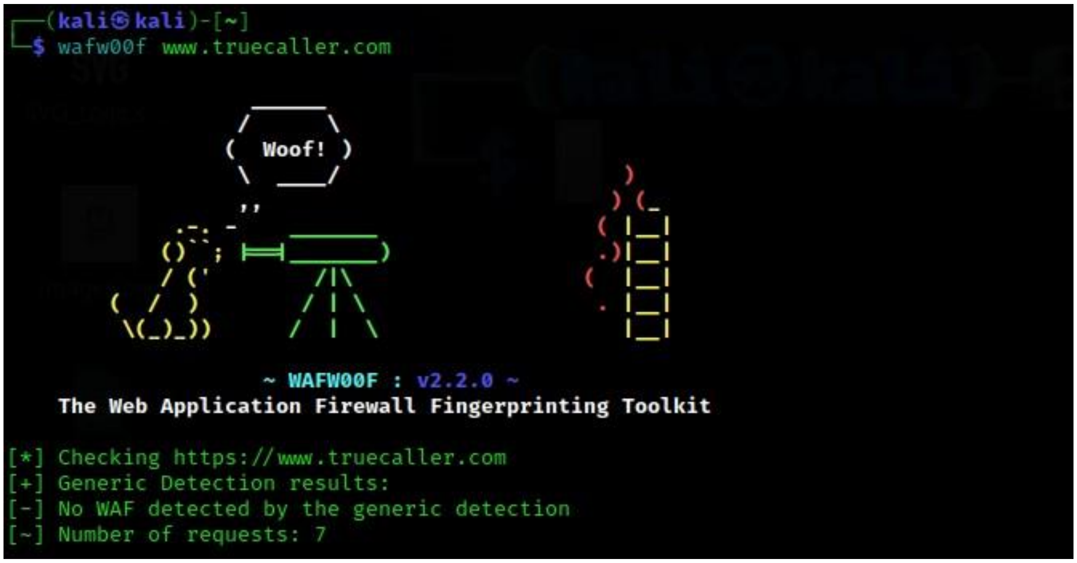
```
┌──(kali㉿kali)-[~]
└─$ assetfinder www.truecaller.com --sub-only
www.truecaller.com
truecaller.com
europe-west1-truecaller-web.cloudfunctions.net
truecaller.blog
dfives.com
tc.onelink.me
apps-amazonprime-video-cbox0w862x.vercel.app
www.jb6img.top
moddingzone.co
payments.truecaller.com
lnk.content.truecaller.com
www.999onlinesale.site
```

I was able to find a smaller number of subdomains using the assetfinder tool. I also used the subfinder tool to find the subdomains, but I was not able to find any subdomain from it



```
┌──(kali㉿kali)-[~]
└─$ subfinder -d www.truecaller.com

                projectdiscovery.io

[INF] Current subfinder version v2.6.0 (outdated)
[INF] Loading provider config from /home/kali/.config/subfinder/provider-config.yaml
[INF] Enumerating subdomains for www.truecaller.com
[INF] Found 0 subdomains for www.truecaller.com in 3 seconds 77 milliseconds
```

I checked whether the website is using a Firewall using the wafw00f tool. It provided that the website is behind the AWS Elastic load balancer (Amazon) Web Application Firewall (WAF).

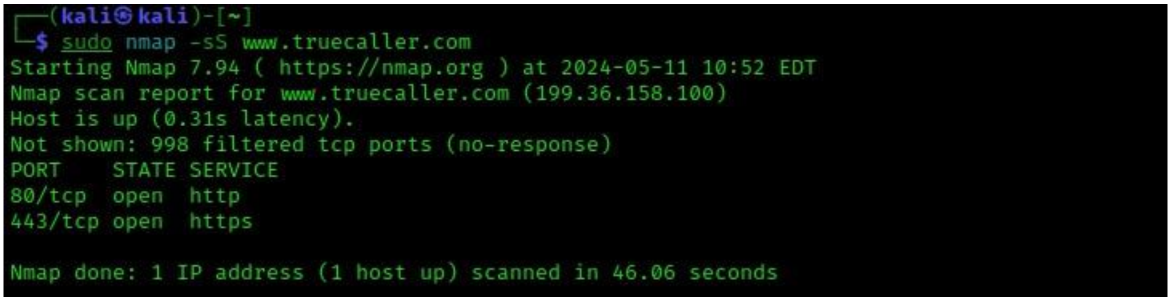


```
┌──(kali㉿kali)-[~]
└─$ wafw00f www.truecaller.com

                ______
               /      \
              (  Woof! )
               \  ____/                      )
               ,,                           ) (_
          .-. -    _______                 ( |__|
         ()``; |==|_______)                .)|__|
         / ('        /|\                  (  |__|
     (  /  )        / | \                  . |__|
      \(_)_))      /  |  \                   |__|

                    ~ WAFW00F : v2.2.0 ~
    The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://www.truecaller.com
[+] Generic Detection results:
[-] No WAF detected by the generic detection
[~] Number of requests: 7
```

There was not any firewall behind the website. A website may be more exposed to several kinds of assaults if there is no firewall operating or if the firewall is not set up correctly. These situations might have serious security ramifications. The following are some possible consequences and dangers linked to either not having a firewall installed at all or having a firewall that is not configured correctly for a website:

- Vulnerability attacks may increase.
    1. DOS attacks
    2. SQL Injections
    3. Cross Site Scripting
    4. Cross Site Request Forgery
    5. Brute Force Attacks
- Unauthorized access to sensitive data or restricted parts of the website.
- Data loss and manipulation which will lead to loss of reputation of the company or deletion of much needed data.

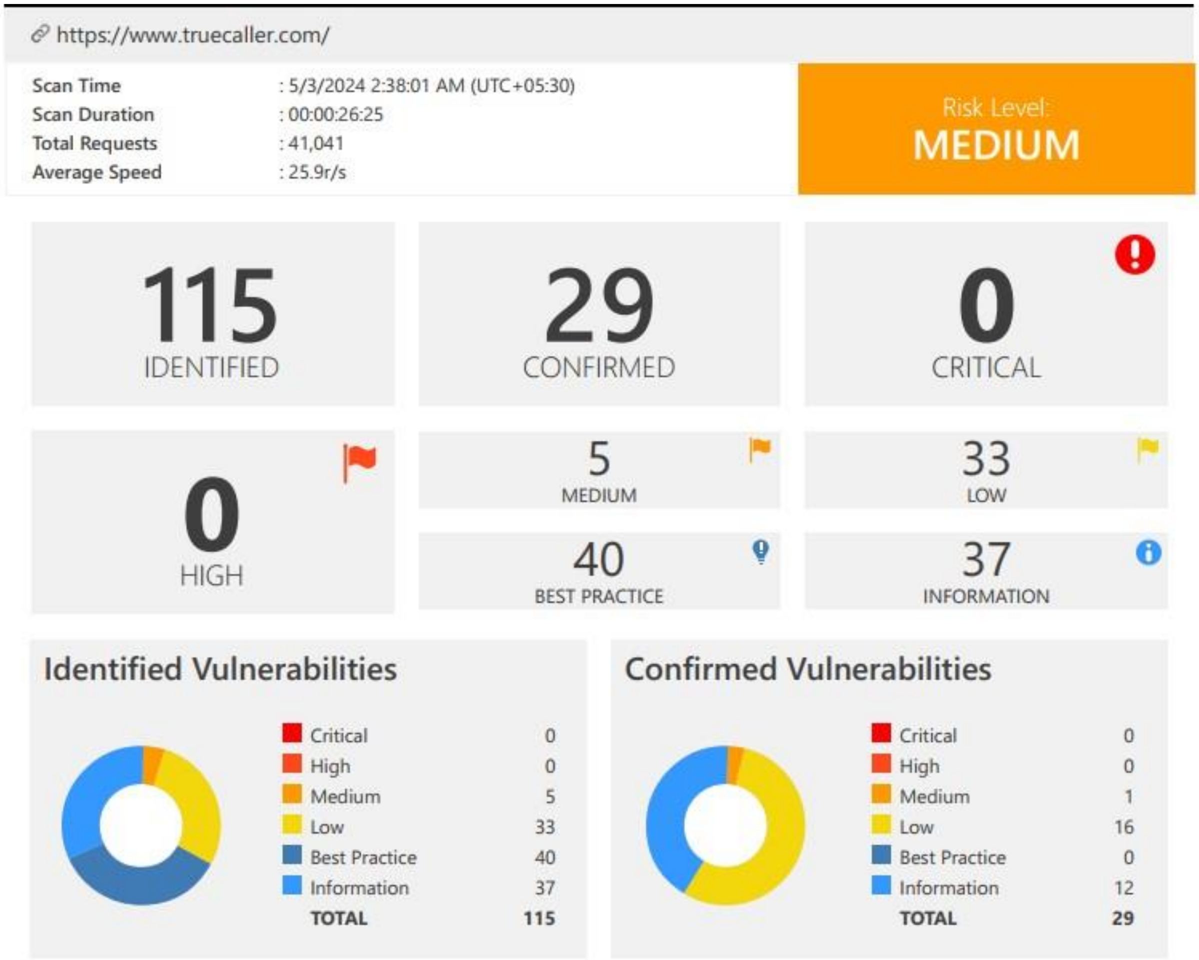
I used the nmap tool to find the open ports of the website



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS www.truecaller.com
Starting Nmap 7.94 ( https://nmap.org ) at 2024-05-11 10:52 EDT
Nmap scan report for www.truecaller.com (199.36.158.100)
Host is up (0.31s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT    STATE SERVICE
80/tcp  open  http
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 46.06 seconds
```

Vulnerabilities found.

Using the Invicti tool I scanned the website to find the vulnerabilities of it. I found several vulnerabilities that belong to the owasp top 10 catergory. The overall risk level of the website was Medium.



https://www.truecaller.com/

Scan Time	: 5/3/2024 2:38:01 AM (UTC+05:30)
Scan Duration	: 00:00:26:25
Total Requests	: 41,041
Average Speed	: 25.9r/s

Risk Level:
MEDIUM

115 IDENTIFIED

29 CONFIRMED

0 CRITICAL

0 HIGH

5 MEDIUM

33 LOW

40 BEST PRACTICE

37 INFORMATION

Identified Vulnerabilities

🟥 Critical	0
🟧 High	0
🟧 Medium	5
🟨 Low	33
🟦 Best Practice	40
🟦 Information	37
TOTAL	**115**

Confirmed Vulnerabilities

🟥 Critical	0
🟧 High	0
🟧 Medium	1
🟨 Low	16
🟦 Best Practice	0
🟦 Information	12
TOTAL	**29**

[possible] source code disclosure (PHP)

Risk level – medium

Method – GET

OWASP TOP 10 – (A3)-2017

Description

A vulnerability known as Source Code Disclosure (PHP) arises when a web server is set incorrectly or when an attacker takes advantage of a weakness in a web application to obtain unauthorized access to PHP files' source code. One popular server-side scripting language, particularly for web development, is PHP (Hypertext Preprocessor).

Impact assessment

Here's how source code disclosure vulnerabilities in PHP typically occur:

I. Misconfigured Web Server: In some cases, web servers may be misconfigured to serve PHP files as plain text instead of executing them as scripts. This misconfiguration allows attackers to access the source code of PHP files directly by requesting them through the web server.

II. Path Traversal Vulnerabilities: Web applications may contain vulnerabilities such as path traversal, where attackers can manipulate input parameters to access files outside of the intended directory structure. If PHP files containing sensitive code are stored in publicly accessible directories or if the application relies on predictable file paths, attackers can exploit path traversal vulnerabilities to disclose the source code.

III. Backup Files: Developers sometimes create backup copies of PHP files with extensions like ".bak" or ".old" and store them in the web server's document root or other accessible directories. If these backup files are not properly secured or removed, attackers can access them and disclose the source code.

IV. Error Messages: Error messages generated by the PHP interpreter or web server may inadvertently disclose the source code of PHP files if they contain sensitive information or if error reporting is not properly configured.

V. Debugging Information: In development environments, PHP files may contain debugging information or comments that reveal sensitive details about the application's logic, database structure, or authentication mechanisms. If these details are exposed in production environments, they can aid attackers in understanding the application's architecture and vulnerabilities.

Solutions

I. Confirm exactly what aspects of the source code are actually disclosed; due to the limitations of this type of vulnerability, it might not be possible to confirm this in all instances. Confirm this is not an intended functionality.

II. If it is a file required by the application, change its permissions to prevent public users from accessing it. If it is not, then.

III. Ensure that the server has all the current security patches applied.

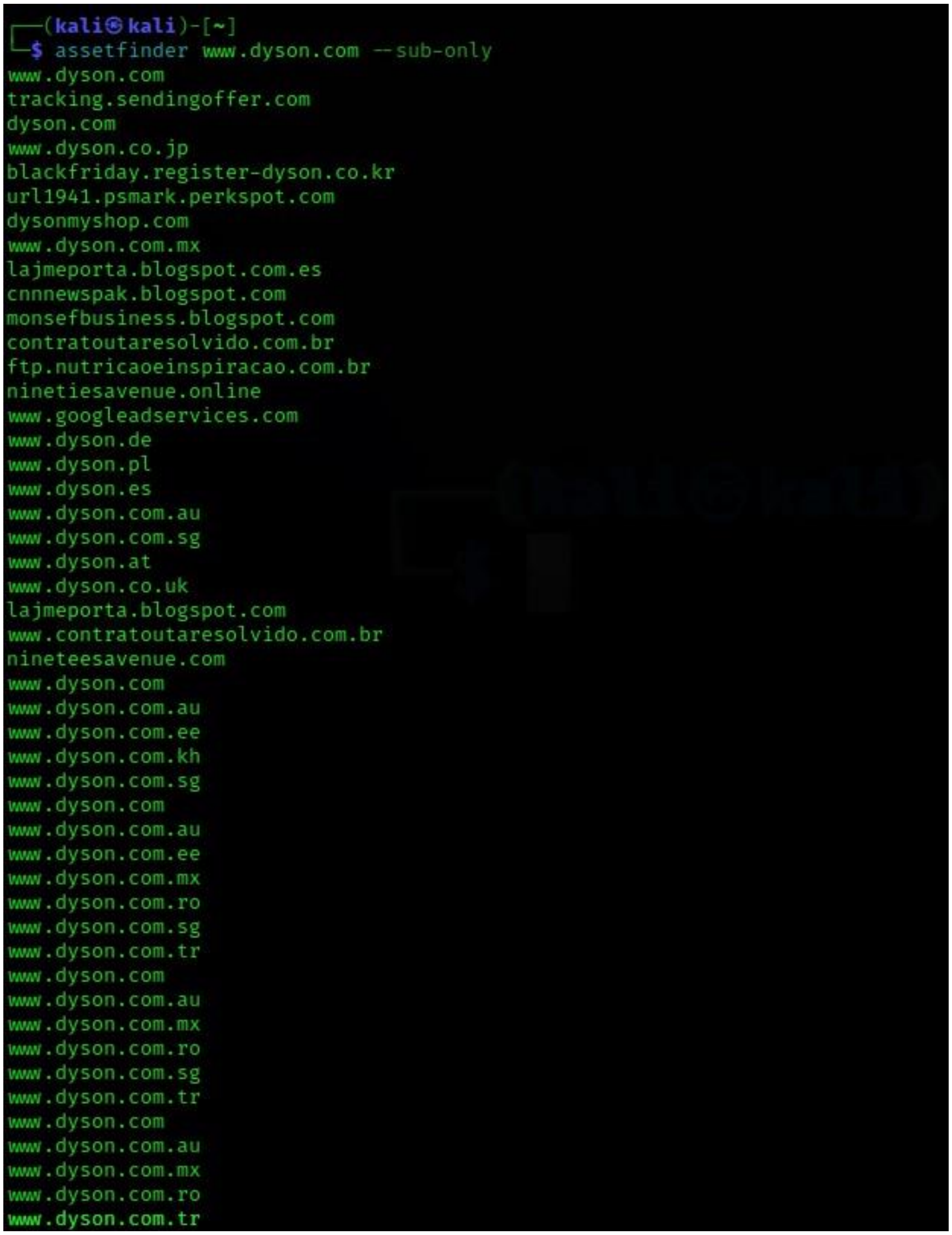
IV. Remove all temporary and backup files from the web server.

5) Report 5(dyson)

company	Dyson
Website	www.dyson.com
Ip	
platform	hackerone

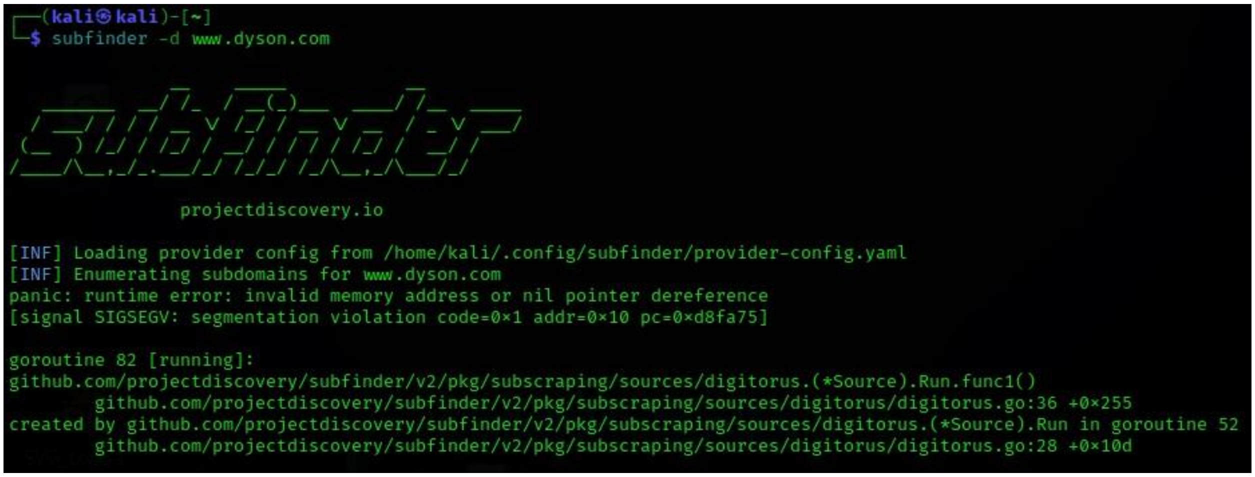
Reconnaissance

As the first step I used the assetfinder tool for scanning the sub domains of the website.



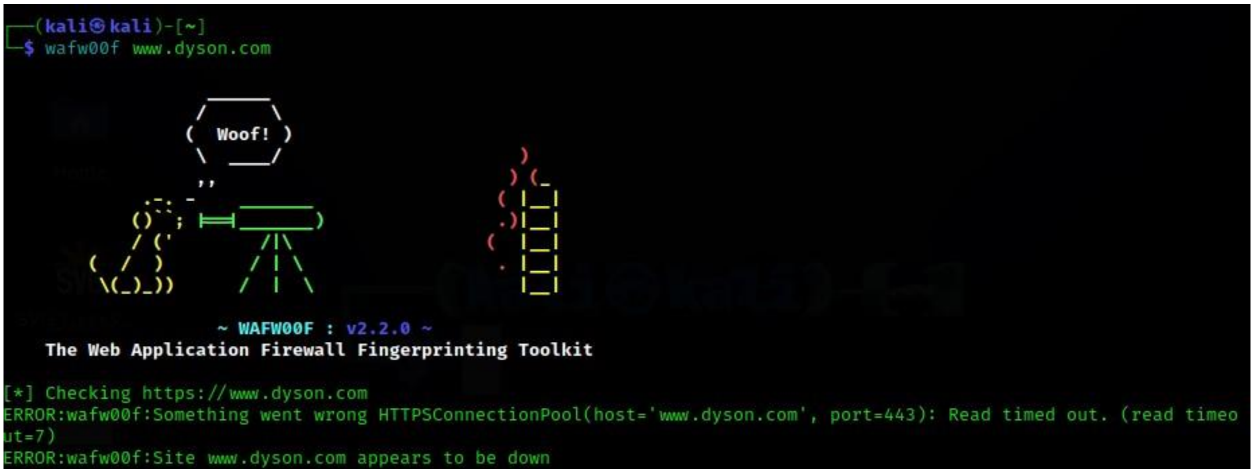
```
┌──(kali㉿kali)-[~]
└─$ assetfinder www.dyson.com --sub-only
www.dyson.com
tracking.sendingoffer.com
dyson.com
www.dyson.co.jp
blackfriday.register-dyson.co.kr
url1941.psmark.perkspot.com
dysonmyshop.com
www.dyson.com.mx
lajmeporta.blogspot.com.es
cnnnewspak.blogspot.com
monsefbusiness.blogspot.com
contratoutaresolvido.com.br
ftp.nutricaoeinspiracao.com.br
ninetiesavenue.online
www.googleadservices.com
www.dyson.de
www.dyson.pl
www.dyson.es
www.dyson.com.au
www.dyson.com.sg
www.dyson.at
www.dyson.co.uk
lajmeporta.blogspot.com
www.contratoutaresolvido.com.br
nineteesavenue.com
www.dyson.com
www.dyson.com.au
www.dyson.com.ee
www.dyson.com.kh
www.dyson.com.sg
www.dyson.com
www.dyson.com.au
www.dyson.com.ee
www.dyson.com.mx
www.dyson.com.ro
www.dyson.com.sg
www.dyson.com.tr
www.dyson.com
www.dyson.com.au
www.dyson.com.mx
www.dyson.com.ro
www.dyson.com.sg
www.dyson.com.tr
www.dyson.com
www.dyson.com.au
www.dyson.com.mx
www.dyson.com.ro
www.dyson.com.tr
```

I was able to find a smaller number of subdomains using the assetfinder tool. I also used the subfinder tool to find the subdomains, but I was not able to find any subdomain from it



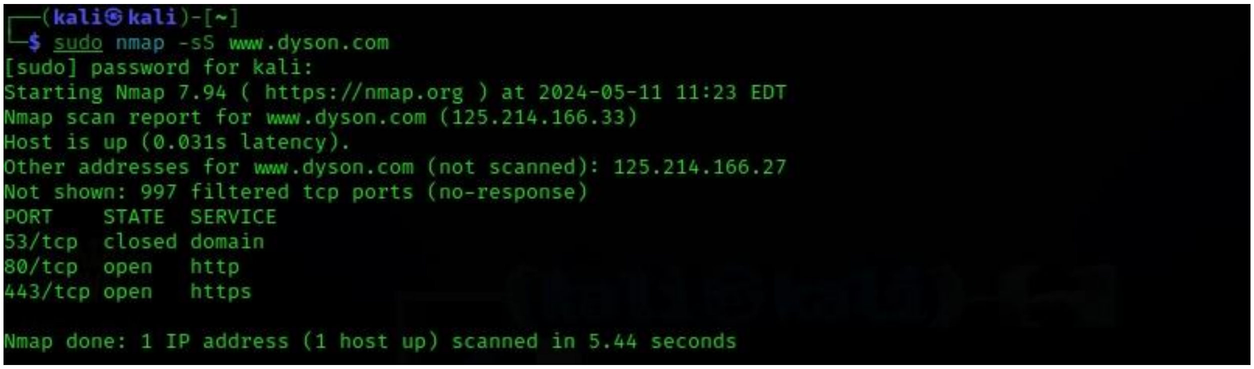
```
┌──(kali㉿kali)-[~]
└─$ subfinder -d www.dyson.com

               projectdiscovery.io

[INF] Loading provider config from /home/kali/.config/subfinder/provider-config.yaml
[INF] Enumerating subdomains for www.dyson.com
panic: runtime error: invalid memory address or nil pointer dereference
[signal SIGSEGV: segmentation violation code=0×1 addr=0×10 pc=0×d8fa75]

goroutine 82 [running]:
github.com/projectdiscovery/subfinder/v2/pkg/subscraping/sources/digitorus.(*Source).Run.func1()
        github.com/projectdiscovery/subfinder/v2/pkg/subscraping/sources/digitorus/digitorus.go:36 +0×255
created by github.com/projectdiscovery/subfinder/v2/pkg/subscraping/sources/digitorus.(*Source).Run in goroutine 52
        github.com/projectdiscovery/subfinder/v2/pkg/subscraping/sources/digitorus/digitorus.go:28 +0×10d
```

I checked whether the website is using a Firewall using the wafw00f tool.



```
┌──(kali㉿kali)-[~]
└─$ wafw00f www.dyson.com

                ~ WAFW00F : v2.2.0 ~
    The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://www.dyson.com
ERROR:wafw00f:Something went wrong HTTPSConnectionPool(host='www.dyson.com', port=443): Read timed out. (read timeo
ut=7)
ERROR:wafw00f:Site www.dyson.com appears to be down
```

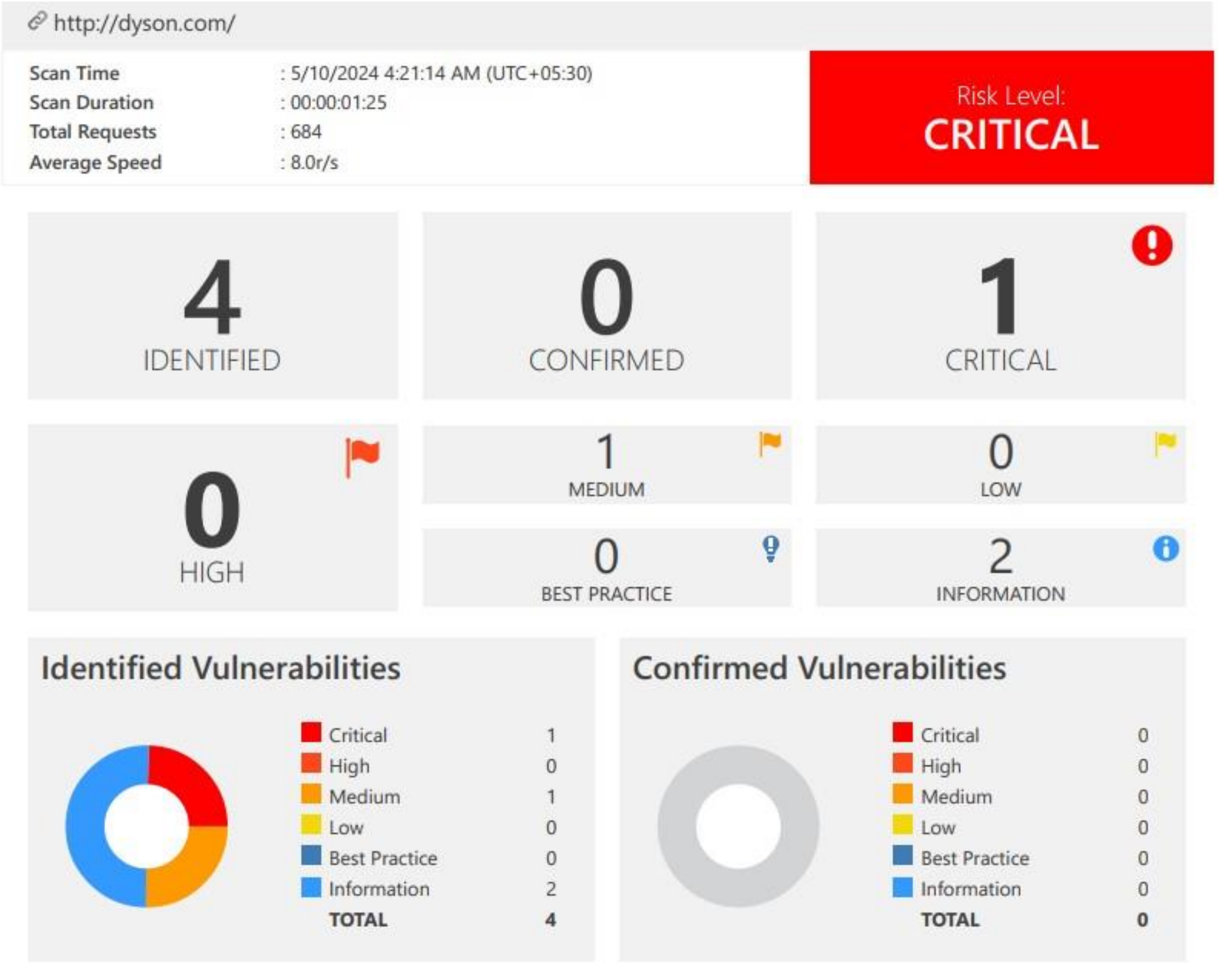
I used the nmap tool to find the open ports of the website



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS www.dyson.com
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2024-05-11 11:23 EDT
Nmap scan report for www.dyson.com (125.214.166.33)
Host is up (0.031s latency).
Other addresses for www.dyson.com (not scanned): 125.214.166.27
Not shown: 997 filtered tcp ports (no-response)
PORT    STATE  SERVICE
53/tcp  closed domain
80/tcp  open   http
443/tcp open   https

Nmap done: 1 IP address (1 host up) scanned in 5.44 seconds
```

Vulnerabilities found.

Using the Invicti tool I scanned the website to find the vulnerabilities of it. I found several vulnerabilities that belong to the owasp top 10 catergory. The overall risk level of the website was Medium.



http://dyson.com/

Scan Time	: 5/10/2024 4:21:14 AM (UTC+05:30)
Scan Duration	: 00:00:01:25
Total Requests	: 684
Average Speed	: 8.0r/s

Risk Level:
CRITICAL

4 IDENTIFIED

0 CONFIRMED

1 CRITICAL

0 HIGH

1 MEDIUM

0 LOW

0 BEST PRACTICE

2 INFORMATION

Identified Vulnerabilities

🟥 Critical	1
🟧 High	0
🟧 Medium	1
🟨 Low	0
🟦 Best Practice	0
🟦 Information	2
TOTAL	**4**

Confirmed Vulnerabilities

🟥 Critical	0
🟧 High	0
🟧 Medium	0
🟨 Low	0
🟦 Best Practice	0
🟦 Information	0
TOTAL	**0**

Vulnerability title: Out-of-date version (IIS)

Risk level – CRITICAL

Method – GET

OWASP TOP 10 – (A9)-2017

DESCRIPTION

A security risk linked to using out-of-date or unpatched versions of Microsoft's Internet Information Services (IIS) web server software is known as the "Out-of-date Version (IIS)" vulnerability. On Windows servers, webpages and online applications are hosted via the well-known web server software IIS.

Impact assessment

Running an outdated version of IIS can pose several security risks, including:

I. **Unpatched Security Vulnerabilities**: It's possible that security flaws in older versions of IIS have been fixed in more recent iterations. These vulnerabilities can be used by attackers to take control of the server, jeopardize the security of websites that are hosted, or steal confidential information.

II. **Lack of Security Updates**: Running an outdated version of IIS means missing out on security updates and patches released by Microsoft to address newly discovered vulnerabilities and security issues. Without these updates, the server remains vulnerable to known attack vectors and exploitation techniques.

III. **Compatibility issues**: It's possible that outdated versions of IIS don't support encryption techniques, contemporary security protocols, and other security features that are necessary for safeguarding sensitive data while it's in transit and securing web applications. This may put the server at risk for security breaches and cause compatibility problems with more recent web technologies.

IV. **Increased Risk of Compromise**: Attackers actively scan the internet for servers running outdated software versions, including IIS, as they are more likely to be vulnerable to known exploits and attacks. By targeting servers with outdated IIS versions, attackers can compromise the server and use it for malicious purposes, such as hosting phishing sites, distributing malware, or launching distributed denial-of-service (DDoS) attacks.

Solutions

To reproduce the Out-of-Date Version vulnerability, follow these steps:

- Identify the version of Internet Information Services (IIS) currently running on the web server.

- Research and obtain information on the latest version of IIS available from the official vendor's website.

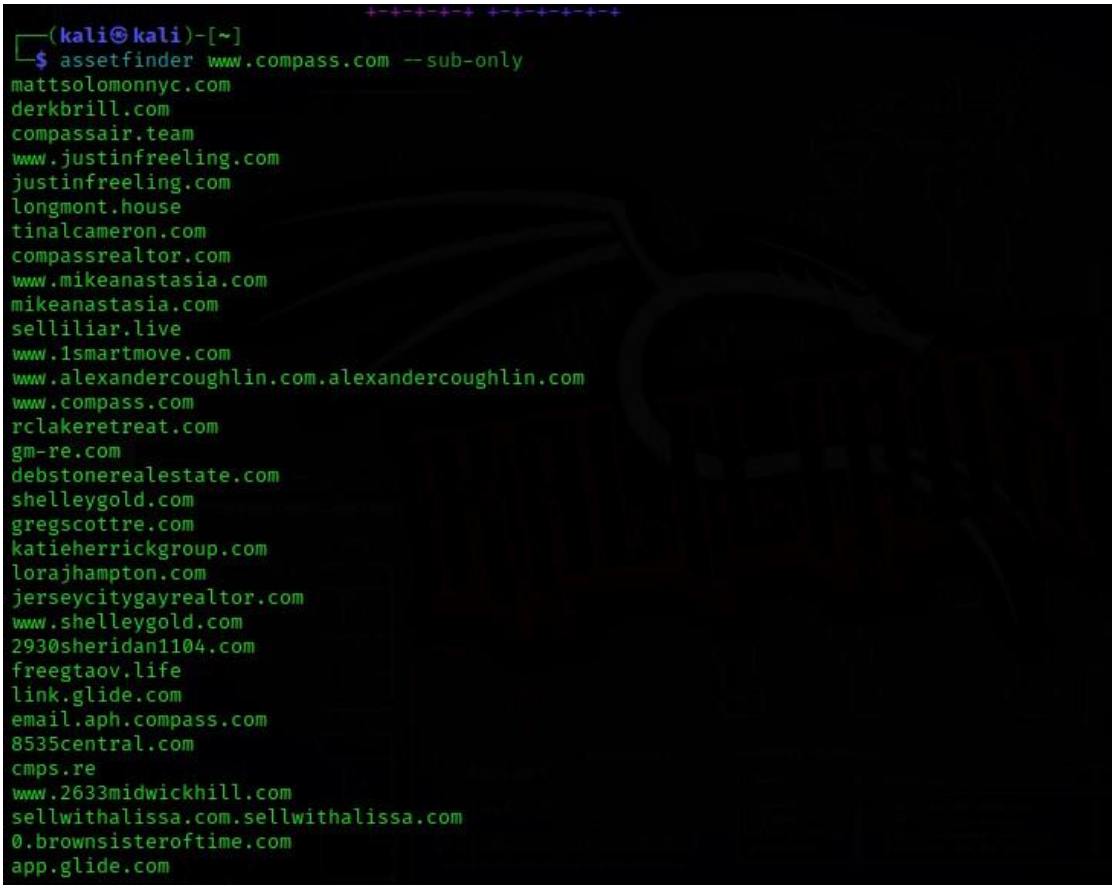
- Compare the version running on the server with the latest version to determine if it is out of date.

6) Report 6(compass)

Company name	Compass
Website	www.Compass.com
Ip	216.146.204.192
platform	Hackerone

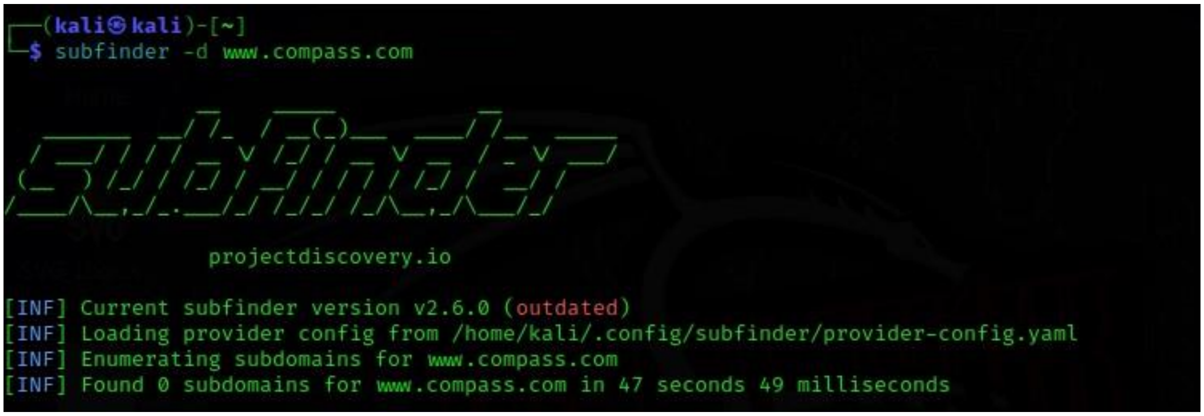
Reconnaissance

As the first step I used the assetfinder tool for scanning the sub domains of the website.



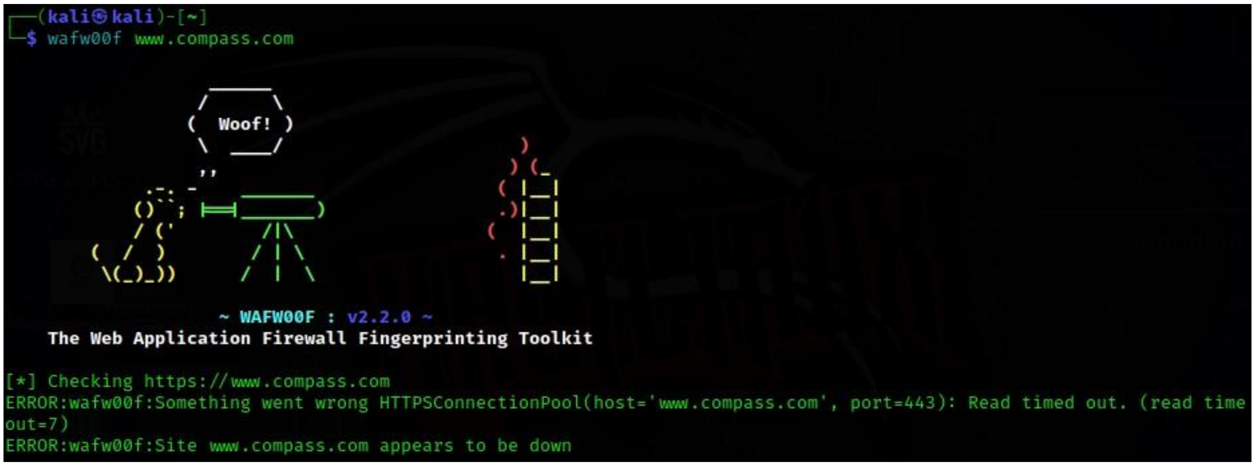
```
┌──(kali㉿kali)-[~]
└─$ assetfinder www.compass.com --sub-only
mattsolomonnyc.com
derkbrill.com
compassair.team
www.justinfreeling.com
justinfreeling.com
longmont.house
tinalcameron.com
compassrealtor.com
www.mikeanastasia.com
mikeanastasia.com
selliliar.live
www.1smartmove.com
www.alexandercoughlin.com.alexandercoughlin.com
www.compass.com
rclakeretreat.com
gm-re.com
debstonerealestate.com
shelleygold.com
gregscottre.com
katieherrickgroup.com
lorajhampton.com
jerseycitygayrealtor.com
www.shelleygold.com
2930sheridan1104.com
freegtaov.life
link.glide.com
email.aph.compass.com
8535central.com
cmps.re
www.2633midwickhill.com
sellwithalissa.com.sellwithalissa.com
0.brownsisteroftime.com
app.glide.com
```

I was able to find a smaller number of subdomains using the assetfinder tool. I also used the subfinder tool to find the subdomains, but I was not able to find any subdomain from it



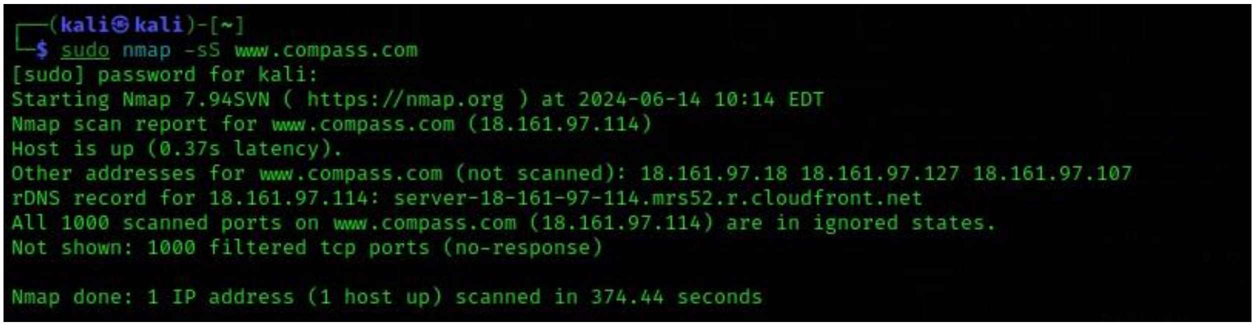
```
┌──(kali㉿kali)-[~]
└─$ subfinder -d www.compass.com

               __    _____           __
   _______  __/ /_  / __(_)___  ____/ /__  _____
  / ___/ / / / __ \/ /_/ / __ \/ __  / _ \/ ___/
 (__  ) /_/ / /_/ / __/ / / / / /_/ /  __/ /
/____/\__,_/_.___/_/ /_/_/ /_/\__,_/\___/_/

                projectdiscovery.io

[INF] Current subfinder version v2.6.0 (outdated)
[INF] Loading provider config from /home/kali/.config/subfinder/provider-config.yaml
[INF] Enumerating subdomains for www.compass.com
[INF] Found 0 subdomains for www.compass.com in 47 seconds 49 milliseconds
```

I checked whether the website is using a Firewall using the wafw00f tool.



```
┌──(kali㉿kali)-[~]
└─$ wafw00f www.compass.com

                   ______
                  /      \
                 (  Woof! )
                  \  ____/                      )
                  ,,                           ) (_
             .-. -    _______                 ( |__|
            ()``; |==|_______)                .)|__|
            / ('        /|\                  (  |__|
        (  /  )        / | \                  . |__|
         \(_)_))      /  |  \                   |__|

                    ~ WAFW00F : v2.2.0 ~
    The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://www.compass.com
ERROR:wafw00f:Something went wrong HTTPSConnectionPool(host='www.compass.com', port=443): Read timed out. (read time
out=7)
ERROR:wafw00f:Site www.compass.com appears to be down
```

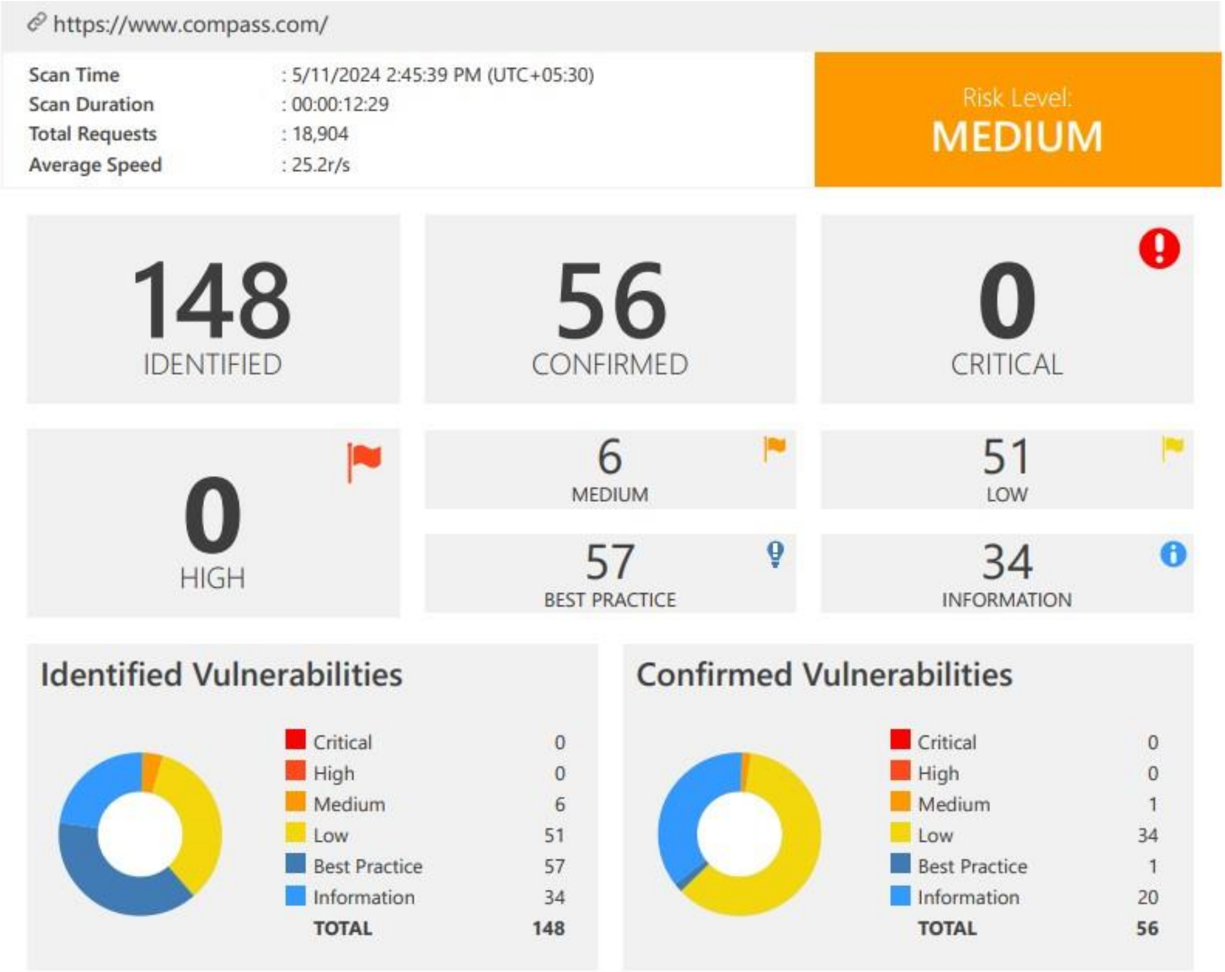
I used the nmap tool to find the open ports of the website



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS www.compass.com
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-14 10:14 EDT
Nmap scan report for www.compass.com (18.161.97.114)
Host is up (0.37s latency).
Other addresses for www.compass.com (not scanned): 18.161.97.18 18.161.97.127 18.161.97.107
rDNS record for 18.161.97.114: server-18-161-97-114.mrs52.r.cloudfront.net
All 1000 scanned ports on www.compass.com (18.161.97.114) are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 374.44 seconds
```

Vulnerabilities found.

Using the Invicti tool I scanned the website to find the vulnerabilities of it. I found several vulnerabilities that belong to the owasp top 10 catergory. The overall risk level of the website was Medium.



https://www.compass.com/

Scan Time	: 5/11/2024 2:45:39 PM (UTC+05:30)
Scan Duration	: 00:00:12:29
Total Requests	: 18,904
Average Speed	: 25.2r/s

Risk Level:
MEDIUM

148
IDENTIFIED

56
CONFIRMED

0
CRITICAL

0
HIGH

6
MEDIUM

51
LOW

57
BEST PRACTICE

34
INFORMATION

Identified Vulnerabilities

🟥 Critical	0
🟥 High	0
🟧 Medium	6
🟨 Low	51
🟦 Best Practice	57
🟦 Information	34
TOTAL	**148**

Confirmed Vulnerabilities

🟥 Critical	0
🟥 High	0
🟧 Medium	1
🟨 Low	34
🟦 Best Practice	1
🟦 Information	20
TOTAL	**56**

Vulnerability title: OUT-of-date version (jquery)

Risk level – medium

Method – GET

OWASP TOP 10 – (A9)-2017

Description

The "Out-of-date Version (jQuery)" vulnerability refers to a security risk associated with using outdated versions of the jQuery JavaScript library in web applications. jQuery is a popular open-source JavaScript library used for simplifying client-side scripting of HTML

Impact assessment

When web applications use outdated versions of jQuery, they may be vulnerable to various security issues, including:

I. Know vulnerabilities: It's possible that jQuery versions that are too old have security flaws that have been fixed in more recent releases. Attackers may use these vulnerabilities to carry out client-side attacks such as cross-site request forgery (CSRF) and cross-site scripting (XSS).

II. Unpatched Bugs: Outdated versions of jQuery may contain bugs or coding errors that could be exploited by attackers to manipulate the behavior of the web application or access sensitive information.

III. Lack of Performance and Compatibility Improvements: Newer versions of jQuery often include performance improvements, bug fixes, and compatibility updates with modern web standards and browsers. Using outdated versions may result in slower performance and compatibility issues with newer browser versions.

IV. Unsupported Features:It's possible that older versions of jQuery don't support more recent features and APIs added in later releases. This might restrict the web application's functionality and capabilities and keep developers from utilizing the most recent web development best practices and approaches.

Solutions

- Identify the functionality that uses the jQuery version that is vulnerable. Concentrate on places that receive user input, including input boxes, form submissions, or AJAX calls that interact with the public jQuery functions. Can use burp suite, chrome tech tools or code editor.

- To build a harmful package including JavaScript code, use Burp Suite, OWASP ZAP, or online XSS

- payload makers. For example, <script>alert('XSS’) </script>

- To view the right page of your web application, use a web browser (such as Google Chrome, Firefox, or Microsoft Edge). Place the carefully built data in input forms, query fields, or any other places that take user input. To change HTML or JavaScript code dynamically, utilize the browser developer tools.

- Once the data has been inserted, watch the browser's movements. Check to see if the payload is performed and if any unusual action, such as an alert box, takes place. To check that the code was properly performed, you can examine network calls and the DOM with the help of browser development tools. To inject can use burp suite or OWASP ZAP or any.

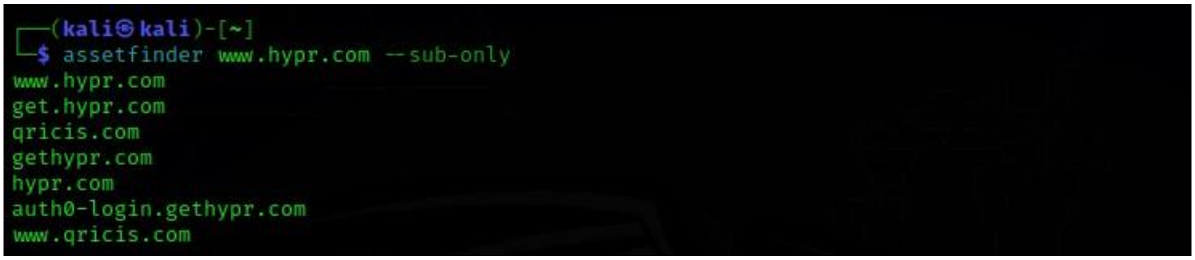
- To make sure that the vulnerability is constant, run the test again with different message changes and more situations. Check to see if the XSS flaw can be continuously recreated and if it affects the security of your service.

7) Report 7(HYPR)

Company name	HYPR
website	www.hypr.com
Ip address	127.0.0.1
platform	Hackerone

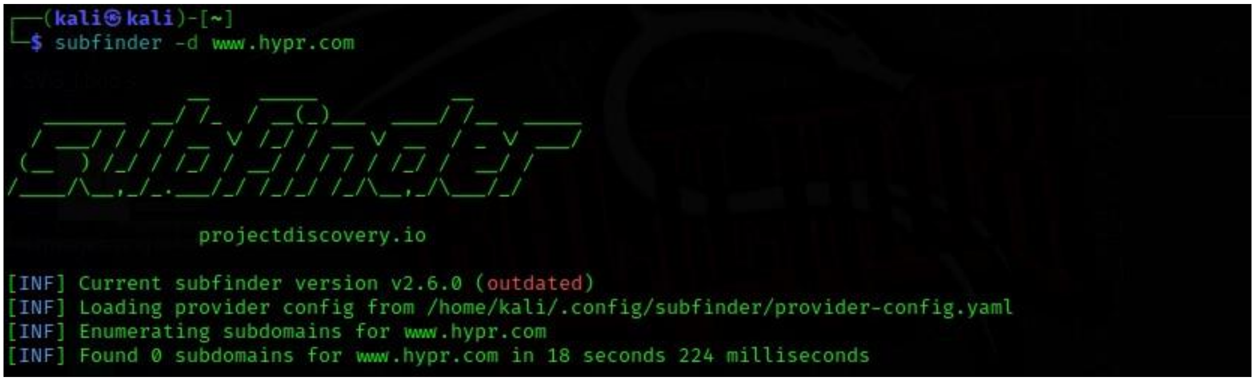
reconnaissance

As the first step I used the assetfinder tool for scanning the sub domains of the website.



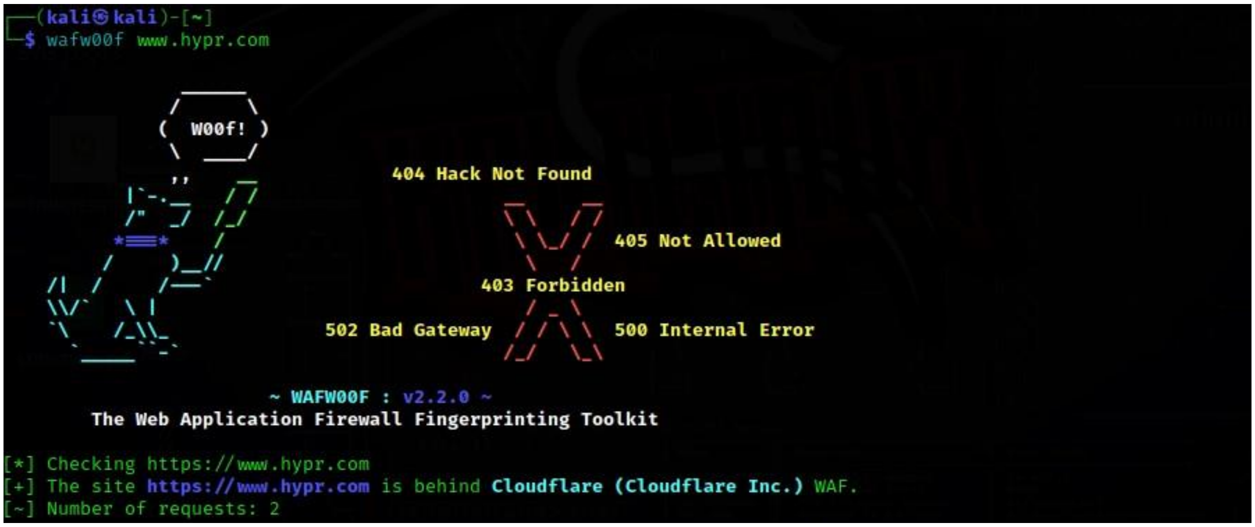
```
┌──(kali㉿kali)-[~]
└─$ assetfinder www.hypr.com --sub-only
www.hypr.com
get.hypr.com
qricis.com
gethypr.com
hypr.com
auth0-login.gethypr.com
www.qricis.com
```

I was able to find a smaller number of subdomains using the assetfinder tool. I also used the subfinder tool to find the subdomains, but I was not able to find any subdomain from it



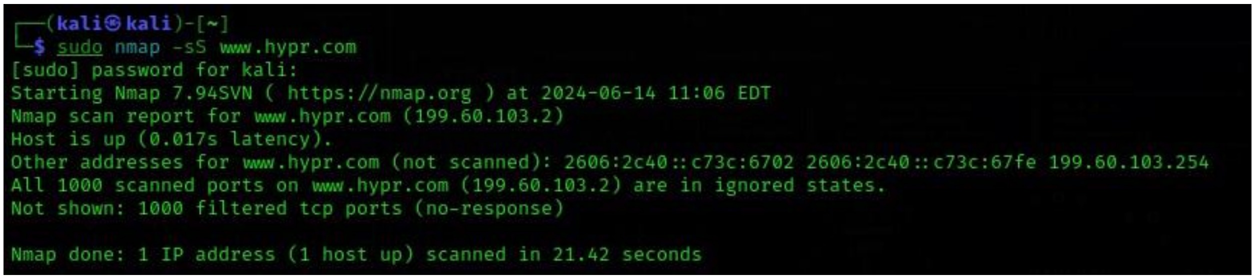
```
┌──(kali㉿kali)-[~]
└─$ subfinder -d www.hypr.com

               __    _____           __
   _______  __/ /_  / __(_)___  ____/ /__  _____
  / ___/ / / / __ \/ /_/ / __ \/ __  / _ \/ ___/
 (__  ) /_/ / /_/ / __/ / / / / /_/ /  __/ /
/____/\__,_/_.___/_/ /_/_/ /_/\__,_/\___/_/

                projectdiscovery.io

[INF] Current subfinder version v2.6.0 (outdated)
[INF] Loading provider config from /home/kali/.config/subfinder/provider-config.yaml
[INF] Enumerating subdomains for www.hypr.com
[INF] Found 0 subdomains for www.hypr.com in 18 seconds 224 milliseconds
```

I checked whether the website is using a Firewall using the wafw00f tool.



```
┌──(kali㉿kali)-[~]
└─$ wafw00f www.hypr.com

                ~ WAFW00F : v2.2.0 ~
    The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://www.hypr.com
[+] The site https://www.hypr.com is behind Cloudflare (Cloudflare Inc.) WAF.
[~] Number of requests: 2
```

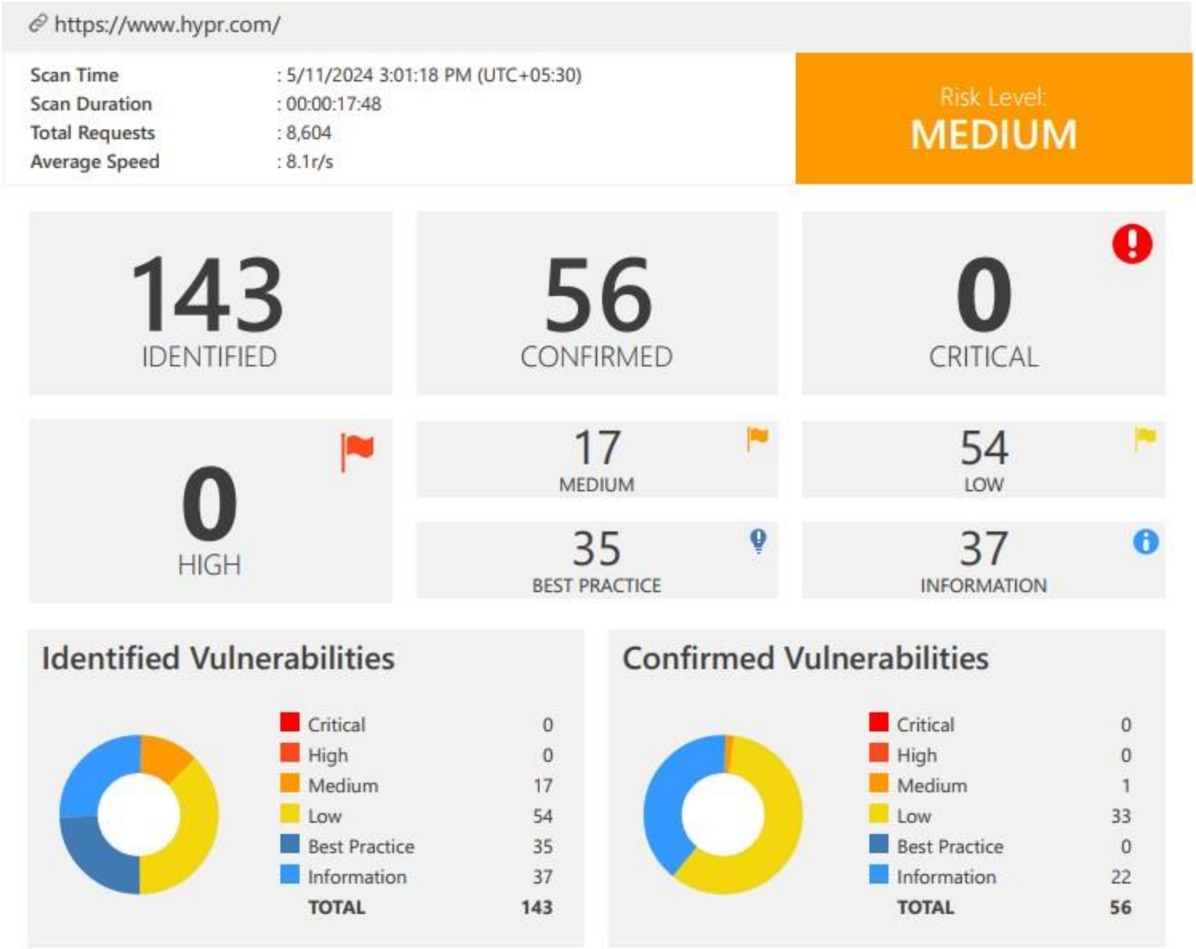
I used the nmap tool to find the open ports of the website



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS www.hypr.com
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-14 11:06 EDT
Nmap scan report for www.hypr.com (199.60.103.2)
Host is up (0.017s latency).
Other addresses for www.hypr.com (not scanned): 2606:2c40::c73c:6702 2606:2c40::c73c:67fe 199.60.103.254
All 1000 scanned ports on www.hypr.com (199.60.103.2) are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 21.42 seconds
```

Vulnerabilities found.

Using the Invicti tool I scanned the website to find the vulnerabilities of it. I found several vulnerabilities that belong to the owasp top 10 catergory. The overall risk level of the website was Medium.



https://www.hypr.com/

Scan Time	: 5/11/2024 3:01:18 PM (UTC+05:30)
Scan Duration	: 00:00:17:48
Total Requests	: 8,604
Average Speed	: 8.1r/s

Risk Level:
MEDIUM

143 IDENTIFIED

56 CONFIRMED

0 CRITICAL

0 HIGH

17 MEDIUM

54 LOW

35 BEST PRACTICE

37 INFORMATION

Identified Vulnerabilities

🟥 Critical	0
🟥 High	0
🟧 Medium	17
🟨 Low	54
🟦 Best Practice	35
🟦 Information	37
TOTAL	**143**

Confirmed Vulnerabilities

🟥 Critical	0
🟥 High	0
🟧 Medium	1
🟨 Low	33
🟦 Best Practice	0
🟦 Information	22
TOTAL	**56**

Vulnerability title: HTTP Strict transport security (HSTS) Errors and warning

Risk level – medium

Method – GET

OWASP TOP 10 – (A6)-2017

Description

HTTP Strict Transport Security (HSTS) is a security feature that helps protect websites from a variety of attacks, such as man-in-the-middle (MITM) and downgrade attacks. It instructs web browsers to exclusively connect to websites using HTTPS (HTTP Secure) and to automatically switch from insecure HTTP requests to HTTPS.

Impact assessment

The following are some possible outcomes and hazard associated with the vulnerability relating to the domain https://www.hypr.com’s lack of an active HTTP strict transport security(HSTS) policy:

I. **Missing HSTS header**: The website does not send the HSTS header in its HTTP responses. Without the HSTS header, web browsers will not enforce HTTPS connections for future visits to the site.

II. **Invalid HSTS header**: The website sends an invalid or malformed HSTS header in its HTTP responses. This could include incorrect syntax, missing directives, or unsupported values.

III. **Short max-age value**: The Max-Age directive in the HSTS header specifies the duration (in seconds) for which the browser should remember to enforce HTTPS. A short Max-Age value (e.g., less than a week) may reduce the effectiveness of HSTS.

IV. **Include sub domains directive issue**: The IncludeSubDomains directive in the HSTS header specifies whether the policy should be applied to all subdomains of the website. If this directive is present but not properly configured, it could lead to issues with subdomain security.

V. **Preloaded HSTS issues**: It is possible to preload websites into the HSTS preload lists of web browsers in order to guarantee that HSTS is applied on the initial visit. Preloaded HSTS errors or warnings might be a sign of problems with the preload setup or status.

VI. **Mixed content warnings**: If the website has resources (such scripts, pictures, or stylesheets) loaded over HTTP rather than HTTPS, HSTS may provide mixed content warnings. The security that HSTS offers may be compromised by mixed material.

VII. **HSTS supercookie risk**: HSTS can be abused to create supercookies that persist even after clearing browser cookies. Errors or warnings related to HSTS supercookie risks may indicate vulnerabilities in HSTS implementations.

Solutions

Ideally, after fixing the errors and warnings, you should consider adding your domain to the HSTS preload list. This will ensure that browsers automatically connect to your website by using HTTPS, actively preventing users from visiting your site using HTTP. Since this list is hardcoded in users' browsers, it will enable HSTS even before they visit your page for the first time, eliminating the need for Trust on First Use (TOFU) with its associated risks and disadvantages. Unless you fix the errors and warnings your website won't meet the conditions required to enter the browser's preload list.

Browser vandors declared:

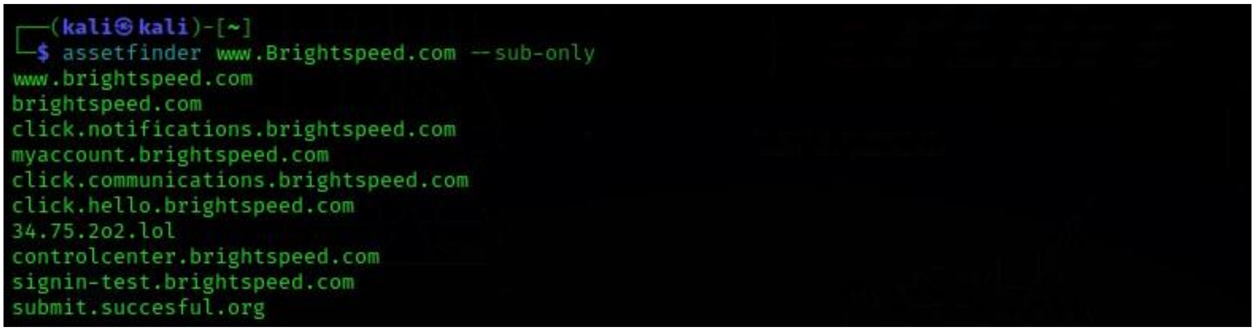
- Serve a valid certificate.
- If you are listening on port 80, redirect all domains from HTTP to HTTPS on the same host. Serve all subdomains over HTTPS:
    - In particular, you must support HTTPS for the www subdomain if a DNS record for that subdomain exists.
- Serve an HSTS header on the base domain for HTTPS requests:
    - The max-agemust be at least 31536000 seconds.
    - The includeSubDomainsdirective must be specified.
    - The preloaddirective must be specified.
    - If you are serving an additional redirect from your HTTPS site, that redirect must have the HSTS header (rather than the page it redirects to)

8) Report 8(brightspeed)

company	Brightspeed
Web site	www.Brightspeed.com
Ip address	216.40.34.41
platform	Hackerone

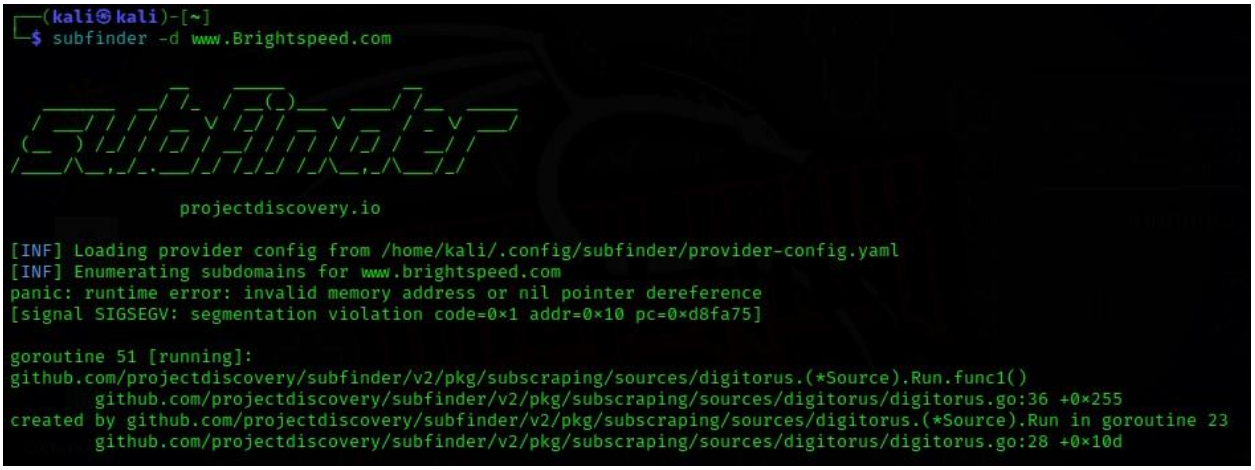
Reconnaissance

As the first step I used the assetfinder tool for scanning the sub domains of the website.



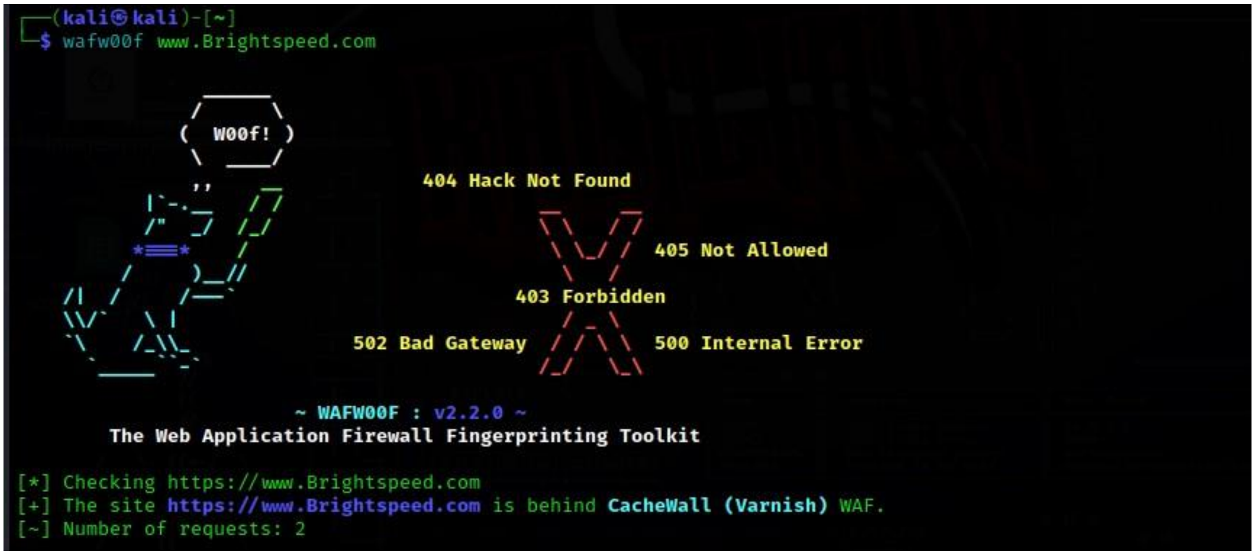
```
┌──(kali㉿kali)-[~]
└─$ assetfinder www.Brightspeed.com --sub-only
www.brightspeed.com
brightspeed.com
click.notifications.brightspeed.com
myaccount.brightspeed.com
click.communications.brightspeed.com
click.hello.brightspeed.com
34.75.2o2.lol
controlcenter.brightspeed.com
signin-test.brightspeed.com
submit.succesful.org
```

I was able to find a smaller number of subdomains using the assetfinder tool. I also used the subfinder tool to find the subdomains, but I was not able to find any subdomain from it



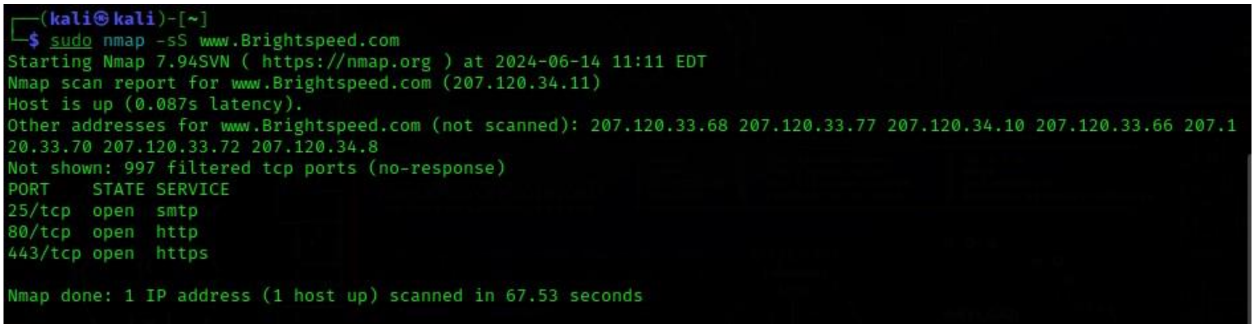
```
┌──(kali㉿kali)-[~]
└─$ subfinder -d www.Brightspeed.com

               __    _____           __
   _______  __/ /_  / __(_)___  ____/ /__  _____
  / ___/ / / / __ \/ /_/ / __ \/ __  / _ \/ ___/
 (__  ) /_/ / /_/ / __/ / / / / /_/ /  __/ /
/____/\__,_/_.___/_/ /_/_/ /_/\__,_/\___/_/

                projectdiscovery.io

[INF] Loading provider config from /home/kali/.config/subfinder/provider-config.yaml
[INF] Enumerating subdomains for www.brightspeed.com
panic: runtime error: invalid memory address or nil pointer dereference
[signal SIGSEGV: segmentation violation code=0x1 addr=0x10 pc=0xd8fa75]

goroutine 51 [running]:
github.com/projectdiscovery/subfinder/v2/pkg/subscraping/sources/digitorus.(*Source).Run.func1()
        github.com/projectdiscovery/subfinder/v2/pkg/subscraping/sources/digitorus/digitorus.go:36 +0x255
created by github.com/projectdiscovery/subfinder/v2/pkg/subscraping/sources/digitorus.(*Source).Run in goroutine 23
        github.com/projectdiscovery/subfinder/v2/pkg/subscraping/sources/digitorus/digitorus.go:28 +0x10d
```

I checked whether the website is using a Firewall using the wafw00f tool.



```
┌──(kali㉿kali)-[~]
└─$ wafw00f www.Brightspeed.com

                ( W00f! )

                                404 Hack Not Found

                                          405 Not Allowed

                                     403 Forbidden

                            502 Bad Gateway       500 Internal Error

                        ~ WAFW00F : v2.2.0 ~
        The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://www.Brightspeed.com
[+] The site https://www.Brightspeed.com is behind CacheWall (Varnish) WAF.
[~] Number of requests: 2
```

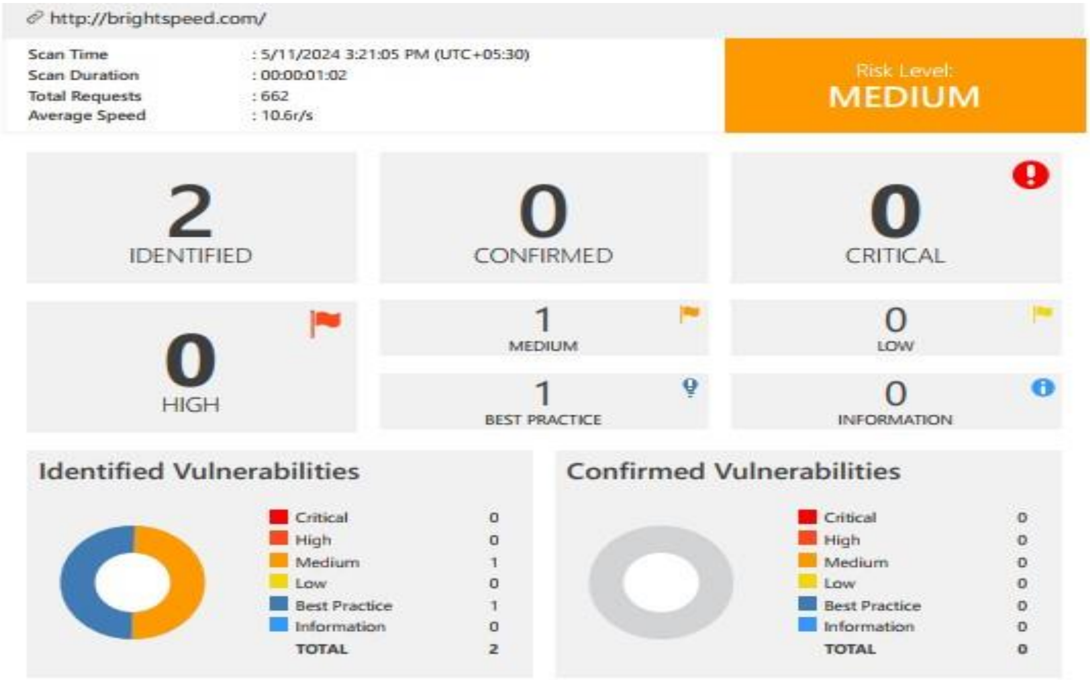
I used the nmap tool to find the open ports of the website



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS www.Brightspeed.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-14 11:11 EDT
Nmap scan report for www.Brightspeed.com (207.120.34.11)
Host is up (0.087s latency).
Other addresses for www.Brightspeed.com (not scanned): 207.120.33.68 207.120.33.77 207.120.34.10 207.120.33.66 207.1
20.33.70 207.120.33.72 207.120.34.8
Not shown: 997 filtered tcp ports (no-response)
PORT    STATE SERVICE
25/tcp  open  smtp
80/tcp  open  http
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 67.53 seconds
```

Vulnerabilities found.

Using the Invicti tool I scanned the website to find the vulnerabilities of it. I found several vulnerabilities that belong to the owasp top 10 catergory. The overall risk level of the website was Medium.



http://brightspeed.com/

Scan Time	: 5/11/2024 3:21:05 PM (UTC+05:30)
Scan Duration	: 00:00:01:02
Total Requests	: 662
Average Speed	: 10.6r/s

Risk Level:
MEDIUM

2 IDENTIFIED

0 CONFIRMED

0 CRITICAL

0 HIGH

1 MEDIUM

0 LOW

1 BEST PRACTICE

0 INFORMATION

Identified Vulnerabilities

🟥 Critical	0
🟥 High	0
🟧 Medium	1
🟨 Low	0
🟦 Best Practice	1
🟦 Information	0
TOTAL	2

Confirmed Vulnerabilities

🟥 Critical	0
🟥 High	0
🟧 Medium	0
🟨 Low	0
🟦 Best Practice	0
🟦 Information	0
TOTAL	0

Vulnerability title: HTTP Strict Transport security (HSTS) erros and warnings.

Risk level – medium

Method – GET

OWASP TOP 10 – (A6)-2017

Description

HTTP Strict Transport Security (HSTS) is a security feature that helps protect websites from a variety of attacks, such as man-in-the-middle (MITM) and downgrade attacks. It instructs web browsers to exclusively connect to websites using HTTPS (HTTP Secure) and to automatically switch from insecure HTTP requests to HTTPS.

Impact assessment

The following are some possible outcomes and hazard associated with the vulnerability relating to the domain http://brightspeed.com’s lack of an active HTTP strict transport security(HSTS) policy:

I. **Missing HSTS header**: The website does not send the HSTS header in its HTTP responses. Without the HSTS header, web browsers will not enforce HTTPS connections for future visits to the site.

II. **Invalid HSTS header**: The website sends an invalid or malformed HSTS header in its HTTP responses. This could include incorrect syntax, missing directives, or unsupported values.

III. **Short max-age value**: The Max-Age directive in the HSTS header specifies the duration (in seconds) for which the browser should remember to enforce HTTPS. A short Max-Age value (e.g., less than a week) may reduce the effectiveness of HSTS.

IV. **Include sub domains directive issue**: The IncludeSubDomains directive in the HSTS header specifies whether the policy should be applied to all subdomains of the website. If this directive is present but not properly configured, it could lead to issues with subdomain security.

V. **Preloaded HSTS issues**: It is possible to preload websites into the HSTS preload lists of web browsers in order to guarantee that HSTS is applied on the initial visit. Preloaded HSTS errors or warnings might be a sign of problems with the preload setup or status.

VI. **Mixed content warnings**: If the website has resources (such scripts, pictures, or stylesheets) loaded over HTTP rather than HTTPS, HSTS may provide mixed content warnings. The security that HSTS offers may be compromised by mixed material.

VII. **HSTS supercookie risk**: HSTS can be abused to create supercookies that persist even after clearing browser cookies. Errors or warnings related to HSTS supercookie risks may indicate vulnerabilities in HSTS implementations.

Solutions

Ideally, after fixing the errors and warnings, you should consider adding your domain to the HSTS preload list. This will ensure that browsers automatically connect to your website by using HTTPS, actively preventing users from visiting your site using HTTP. Since this list is hardcoded in users' browsers, it will enable HSTS even before they visit your page for the first time, eliminating the need for Trust on First Use (TOFU) with its associated risks and disadvantages. Unless you fix the errors and warnings your website won't meet the conditions required to enter the browser's preload list.

Browser vandors declared:

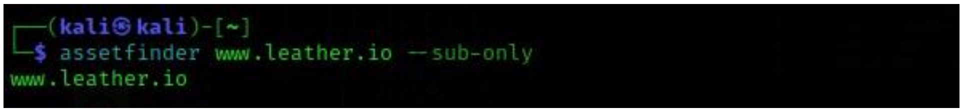
- Serve a valid certificate.
- If you are listening on port 80, redirect all domains from HTTP to HTTPS on the same host. Serve all subdomains over HTTPS:
  - In particular, you must support HTTPS for the www subdomain if a DNS record for that subdomain exists.
- Serve an HSTS header on the base domain for HTTPS requests:
  - The max-agemust be at least 31536000 seconds.
  - The includeSubDomainsdirective must be specified.
  - The preloaddirective must be specified.
  - If you are serving an additional redirect from your HTTPS site, that redirect must have the HSTS header (rather than the page it redirects to)

9) Report 9(leather wallet)

Company name	Leather wallet
Website	http://www.leather.io
Ip address	
platform	hackerone

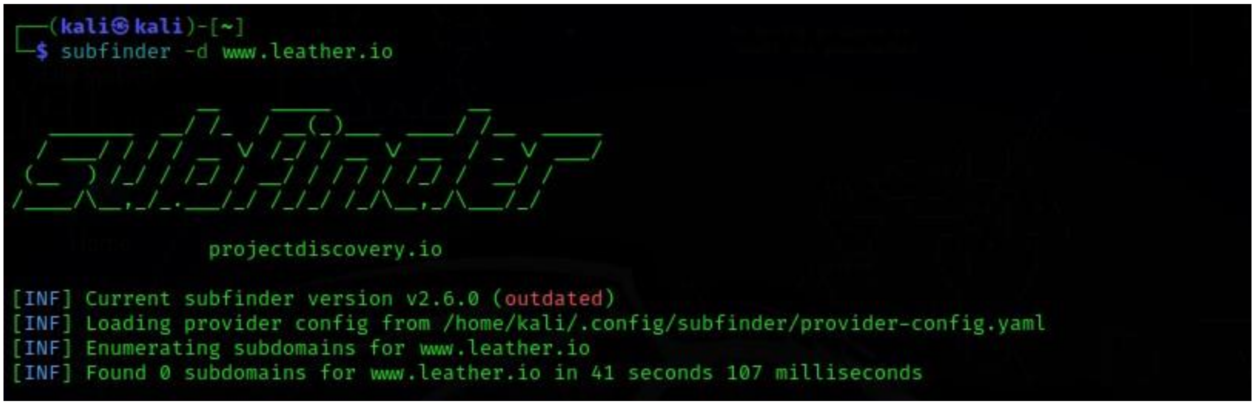
Reconnaissance

As the first step I used the assetfinder tool for scanning the sub domains of the website.



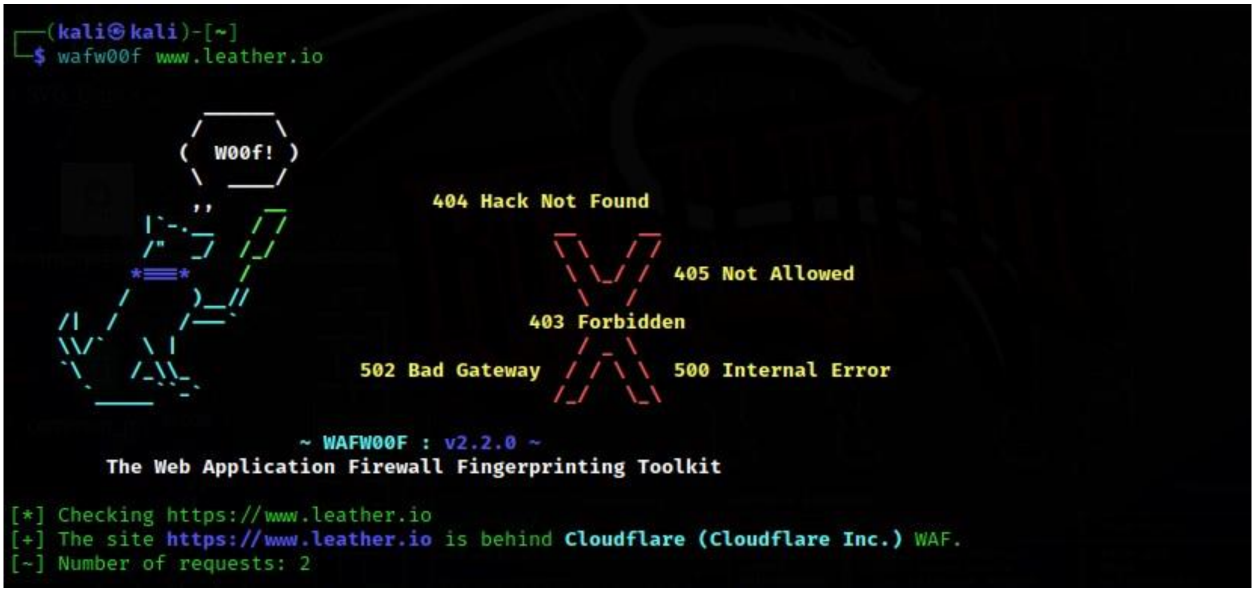
```
┌──(kali㉿kali)-[~]
└─$ assetfinder www.leather.io --sub-only
www.leather.io
```

I was able to find a smaller number of subdomains using the assetfinder tool. I also used the subfinder tool to find the subdomains, but I was not able to find any subdomain from it



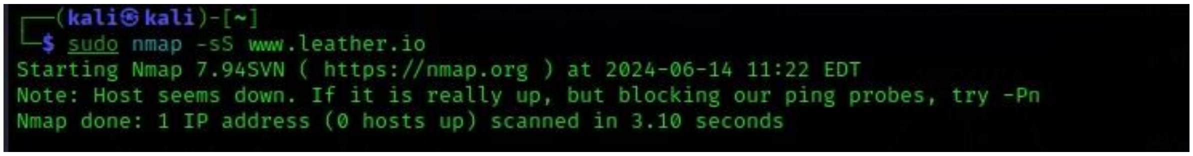
```
┌──(kali㉿kali)-[~]
└─$ subfinder -d www.leather.io

                projectdiscovery.io

[INF] Current subfinder version v2.6.0 (outdated)
[INF] Loading provider config from /home/kali/.config/subfinder/provider-config.yaml
[INF] Enumerating subdomains for www.leather.io
[INF] Found 0 subdomains for www.leather.io in 41 seconds 107 milliseconds
```

I checked whether the website is using a Firewall using the wafw00f tool.



```
┌──(kali㉿kali)-[~]
└─$ wafw00f www.leather.io

                ______
               /      \
              (  W00f! )
               \  ____/
               ,,    __            404 Hack Not Found
           |`-.__   / /                      __     __
           /"  _/  /_/                       \ \   / /
          *===*    /                          \ \_/ /  405 Not Allowed
         /     )__//                           \   /
    /|  /     /---`                        403 Forbidden
    \\/`    \ |                                 / _ \
    `\    /_\\_              502 Bad Gateway  / / \ \  500 Internal Error
      `_____``-`                             /_/   \_\

                        ~ WAFW00F : v2.2.0 ~
        The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://www.leather.io
[+] The site https://www.leather.io is behind Cloudflare (Cloudflare Inc.) WAF.
[~] Number of requests: 2
```

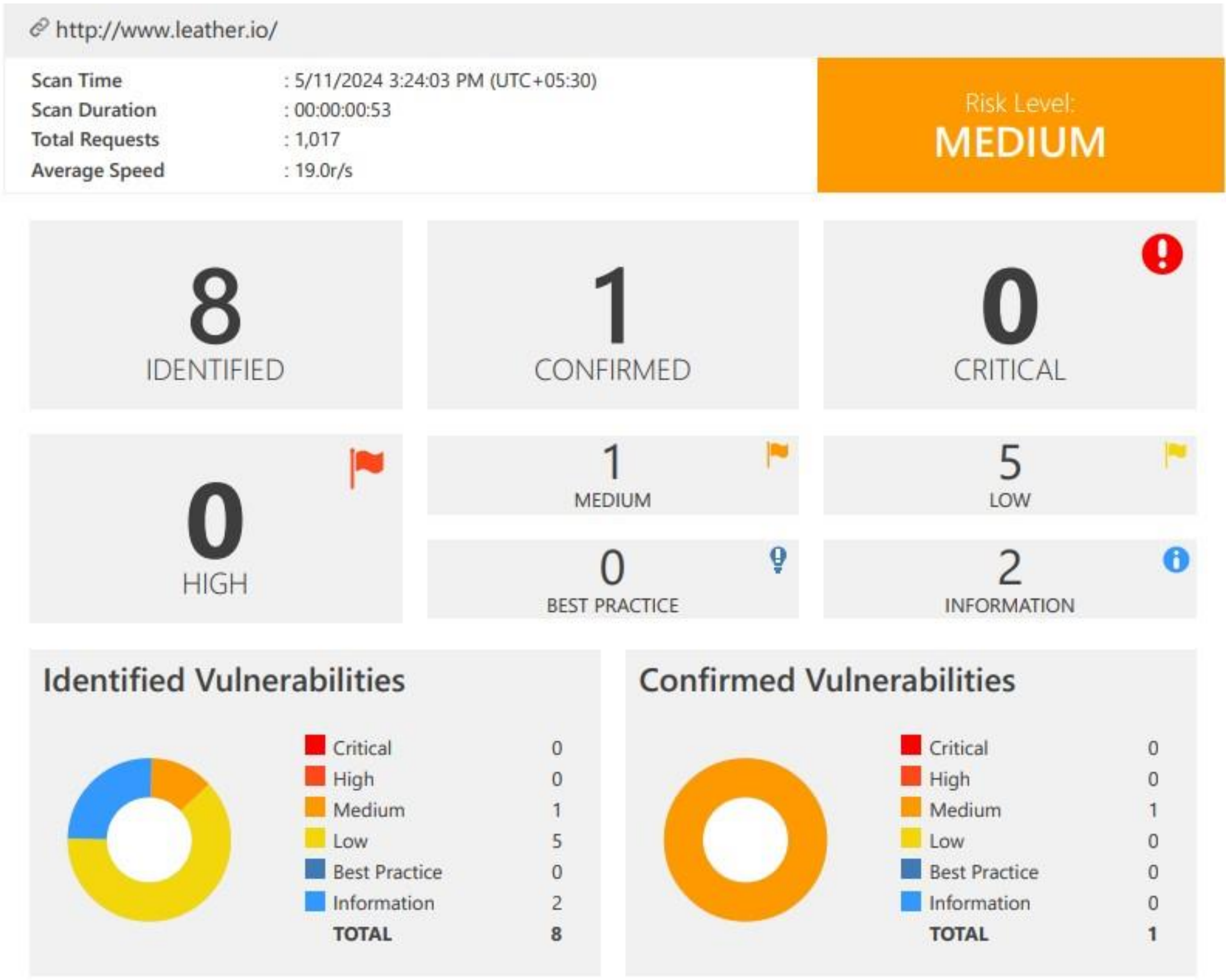
I used the nmap tool to find the open ports of the website



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS www.leather.io
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-14 11:22 EDT
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.10 seconds
```

Vulnerabilities found.

Using the Invicti tool I scanned the website to find the vulnerabilities of it. I found several vulnerabilities that belong to the owasp top 10 catergory. The overall risk level of the website was Medium.



http://www.leather.io/

Scan Time	: 5/11/2024 3:24:03 PM (UTC+05:30)
Scan Duration	: 00:00:00:53
Total Requests	: 1,017
Average Speed	: 19.0r/s

Risk Level:
MEDIUM

8 IDENTIFIED

1 CONFIRMED

0 CRITICAL

0 HIGH

1 MEDIUM

5 LOW

0 BEST PRACTICE

2 INFORMATION

Identified Vulnerabilities

🟥	Critical	0
🟥	High	0
🟧	Medium	1
🟨	Low	5
🟦	Best Practice	0
🟦	Information	2
	TOTAL	**8**

Confirmed Vulnerabilities

🟥	Critical	0
🟥	High	0
🟧	Medium	1
🟨	Low	0
🟦	Best Practice	0
🟦	Information	0
	TOTAL	**1**

Vulnerability title: week ciphers enabled(A3)

Risk level – medium

Method – GET

OWASP TOP 10 – (A3)-2017

Description

Cryptographic algorithms deemed insecure because of flaws or deficiencies in their design are referred to as weak ciphers. When weak ciphers are enabled in a system's cryptographic setup, there might be serious security issues since attackers may use them to intercept or alter encrypted data.

Impact assessment

The Weak Cipher Enabled vulnerability under the OWASP category on http://www.leather.io has led to the discovery of a number of potential security vulnerabilities because of the use of insufficient encryption protocols or ciphers. The following possible outcomes are all included in the significant impact of ignoring this vulnerability:

I. **Loss of User Trust**: Consumers put their trust in e-commerce sites like Traffic Factory to safeguard their private information. This confidence is undermined by weak ciphers that expose users' information to dangers. Echo Box's credibility and reputation may suffer greatly as a result, and consumer loyalty and trust may drop.

II. **Data Breaches**: Data breaches are more likely when ciphers are weak. Attackers may obtain sensitive user data if they are able to take advantage of these vulnerabilities, which could result in data breaches and the related financial and legal repercussions.

III. **Enhanced Attack Surface**: Cybercriminals can more easily exploit vulnerabilities and obtain illegal access to a website, its servers, and underlying systems when weak ciphers provide them a larger attack surface

IV. **Man-in-the-Middle Attacks**: Vulnerabilities in ciphers allow attackers to intercept and modify data that users exchange with websites. This poses serious security issues since it might result in data alteration and unauthorized access.

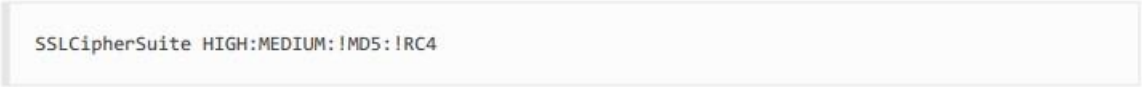
V. **Data Breaches**: Weak ciphers can break encryption, exposing private customer information and perhaps resulting in legal and financial repercussions for Echo Box.

When weak ciphers are enabled in a system's cryptographic setup, the system becomes vulnerable to a number of security risks, such as sensitive data interception, unauthorized access, and data manipulation. Disabling weak ciphers and sticking to strong cryptographic algorithms that have been security-verified by cryptography specialists is advised.

You should check and change the cryptographic configuration settings to make sure that only strong and secure ciphers are enabled in order to address the problem of weak ciphers being enabled in a system. This might entail switching to more secure current cryptographic algorithms and disabling insecure ciphers via configuring web servers, SSL/TLS implementations, and other network services. Frequent vulnerability scans and security audits can assist in locating and fixing problems with weak ciphers.

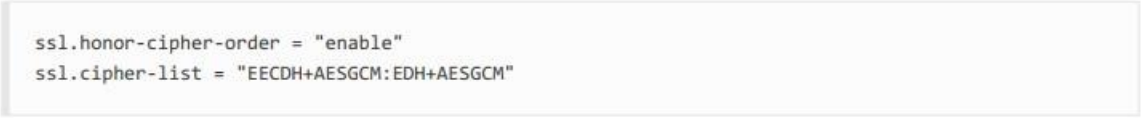
Solutions

1. For Apache, you should modify the SSLCipherSuite directive in the httpd.conf.



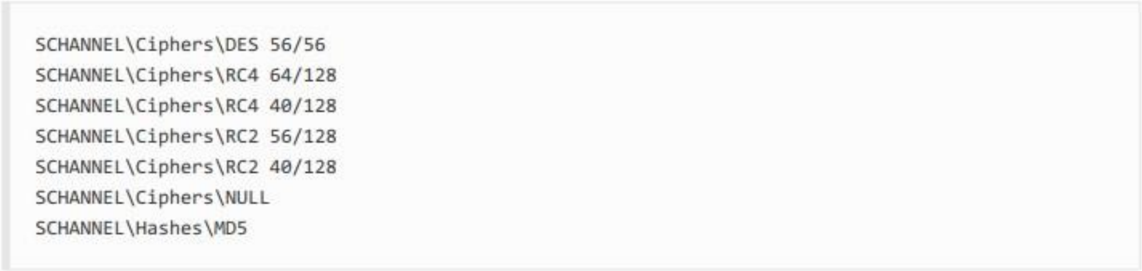
```
SSLCipherSuite HIGH:MEDIUM:!MD5:!RC4
```

2. Lighttpd:



```
ssl.honor-cipher-order = "enable"
ssl.cipher-list = "EECDH+AESGCM:EDH+AESGCM"
```

3. For Microsoft IIS, you should make some changes to the system registry. Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.
   d) Click Start, click Run, type regedt32or type regedit, and then click OK
   e) In Registry Editor, locate the following registry key: HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders
   f) Set "Enabled" DWORD to "0x0" for the following registry keys:



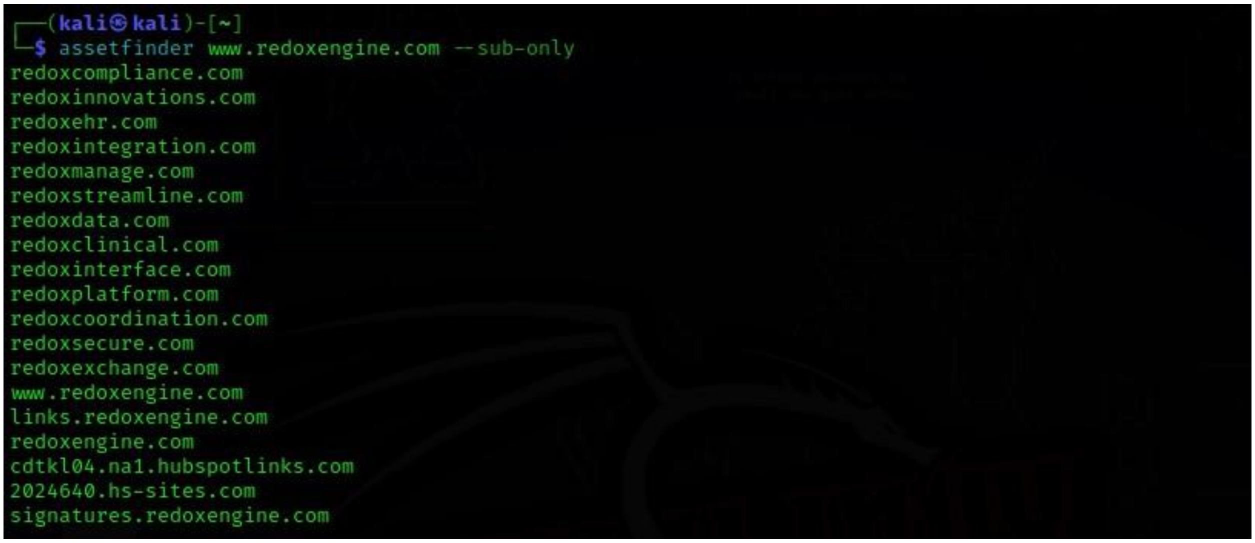
```
SCHANNEL\Ciphers\DES 56/56
SCHANNEL\Ciphers\RC4 64/128
SCHANNEL\Ciphers\RC4 40/128
SCHANNEL\Ciphers\RC2 56/128
SCHANNEL\Ciphers\RC2 40/128
SCHANNEL\Ciphers\NULL
SCHANNEL\Hashes\MD5
```

10) Report 10(redoxengine)

Company name	redoxengine
Website	https://www.redoxengine.com
Ip address	
platform	hackerone

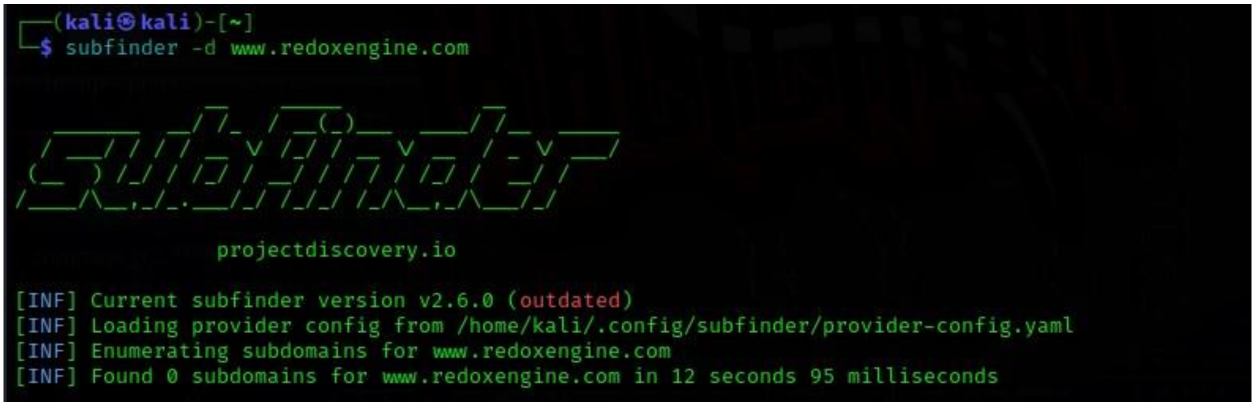
Reconnaissance

As the first step I used the assetfinder tool for scanning the sub domains of the website.



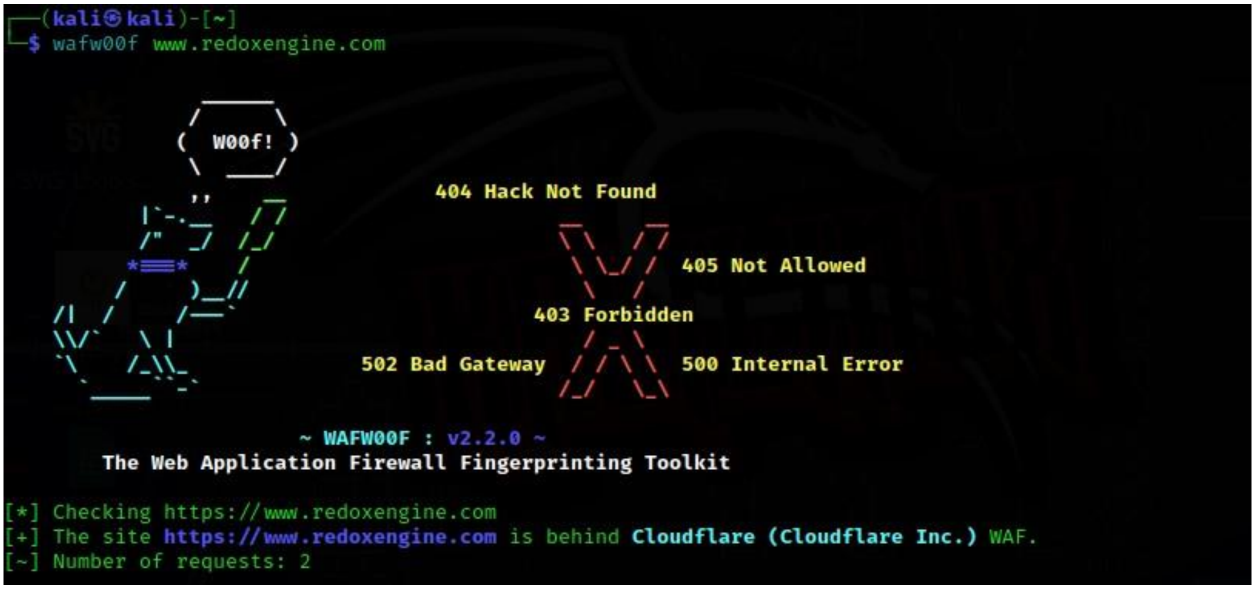
```
┌──(kali㉿kali)-[~]
└─$ assetfinder www.redoxengine.com --sub-only
redoxcompliance.com
redoxinnovations.com
redoxehr.com
redoxintegration.com
redoxmanage.com
redoxstreamline.com
redoxdata.com
redoxclinical.com
redoxinterface.com
redoxplatform.com
redoxcoordination.com
redoxsecure.com
redoxexchange.com
www.redoxengine.com
links.redoxengine.com
redoxengine.com
cdtkl04.na1.hubspotlinks.com
2024640.hs-sites.com
signatures.redoxengine.com
```

I was able to find a smaller number of subdomains using the assetfinder tool. I also used the subfinder tool to find the subdomains, but I was not able to find any subdomain from it



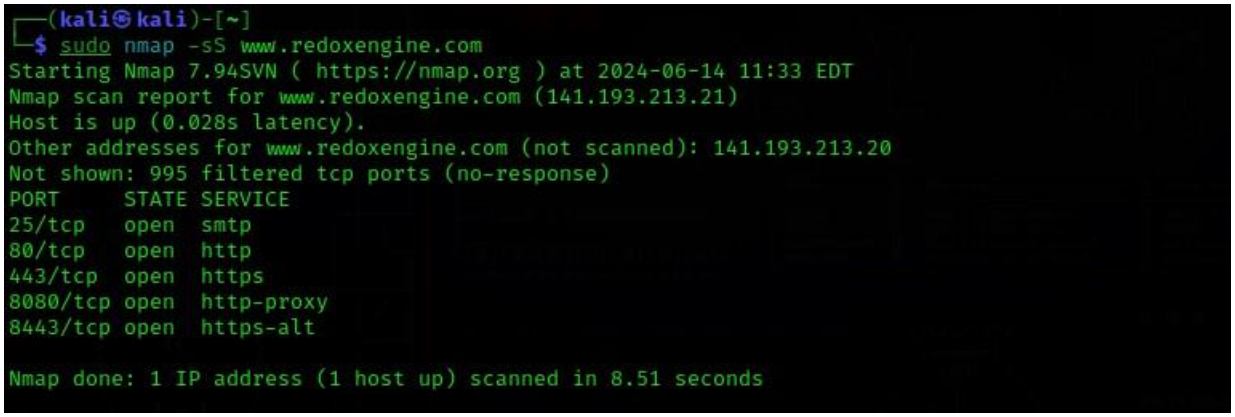
```
┌──(kali㉿kali)-[~]
└─$ subfinder -d www.redoxengine.com

               __    _____           __
   _______  __/ /_  / __(_)___  ____/ /__  _____
  / ___/ / / / __ \/ /_/ / __ \/ __  / _ \/ ___/
 (__  ) /_/ / /_/ / __/ / / / / /_/ /  __/ /
/____/\__,_/_.___/_/ /_/_/ /_/\__,_/\___/_/

                projectdiscovery.io

[INF] Current subfinder version v2.6.0 (outdated)
[INF] Loading provider config from /home/kali/.config/subfinder/provider-config.yaml
[INF] Enumerating subdomains for www.redoxengine.com
[INF] Found 0 subdomains for www.redoxengine.com in 12 seconds 95 milliseconds
```

I checked whether the website is using a Firewall using the wafw00f tool.



```
┌──(kali㉿kali)-[~]
└─$ wafw00f www.redoxengine.com

                ______
               /      \
              (  W00f! )
               \  ____/
               ,,    __            404 Hack Not Found
           |`-.__   / /                      __     __
           /"  _/  /_/                       \ \   / /
          *===*    /                          \ \_/ /  405 Not Allowed
         /     )__//                           \   /
    /|  /     /---`                        403 Forbidden
    \\/`   \ |                                 / _ \
    `\    /_\\_              502 Bad Gateway  / / \ \  500 Internal Error
      `_____``-`                             /_/   \_\

                        ~ WAFW00F : v2.2.0 ~
        The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://www.redoxengine.com
[+] The site https://www.redoxengine.com is behind Cloudflare (Cloudflare Inc.) WAF.
[~] Number of requests: 2
```

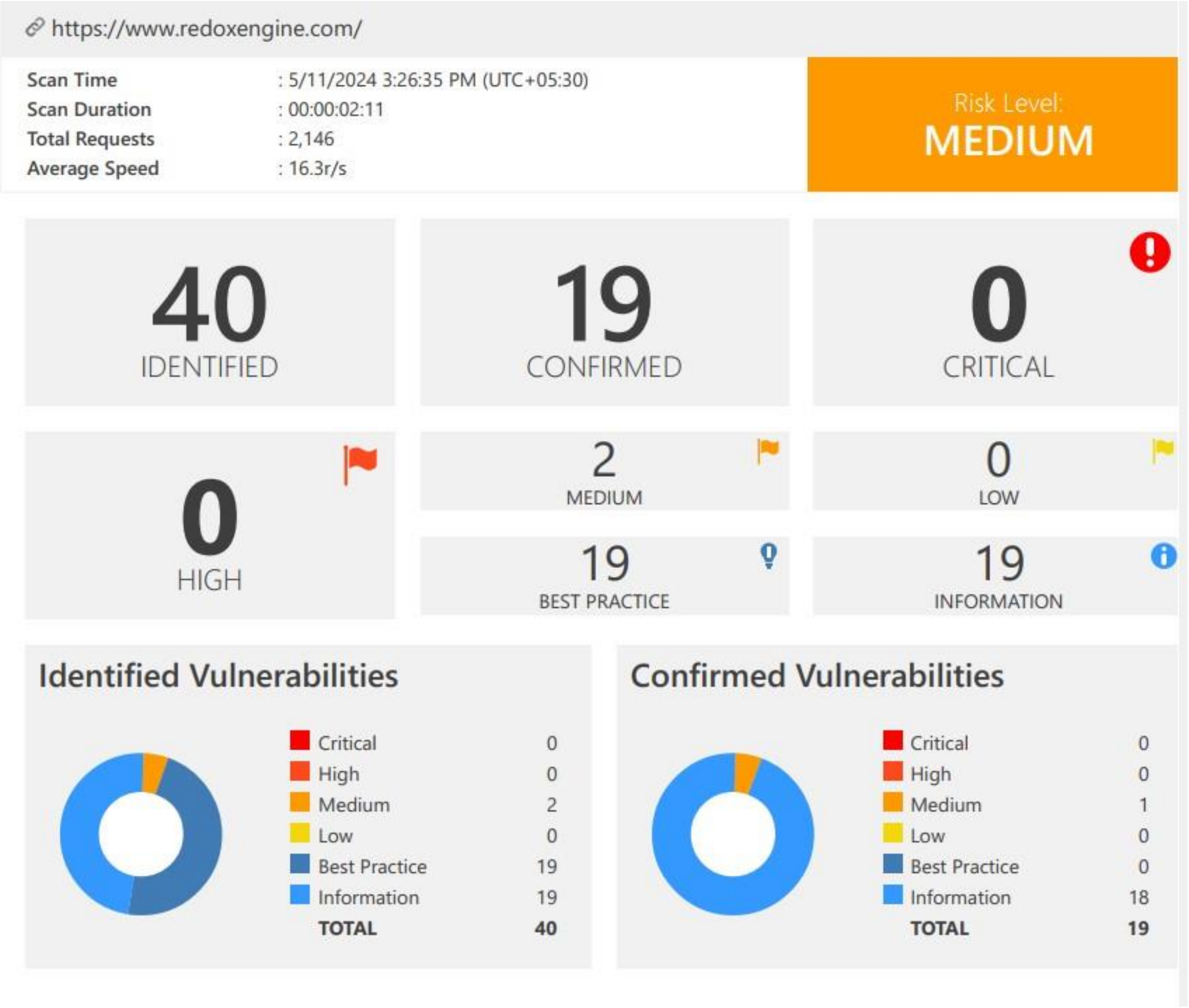
I used the nmap tool to find the open ports of the website



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS www.redoxengine.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-14 11:33 EDT
Nmap scan report for www.redoxengine.com (141.193.213.21)
Host is up (0.028s latency).
Other addresses for www.redoxengine.com (not scanned): 141.193.213.20
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE
25/tcp   open  smtp
80/tcp   open  http
443/tcp  open  https
8080/tcp open  http-proxy
8443/tcp open  https-alt

Nmap done: 1 IP address (1 host up) scanned in 8.51 seconds
```

Vulnerabilities found.

Using the Invicti tool I scanned the website to find the vulnerabilities of it. I found several vulnerabilities that belong to the owasp top 10 catergory. The overall risk level of the website was Medium.



https://www.redoxengine.com/

Scan Time	: 5/11/2024 3:26:35 PM (UTC+05:30)
Scan Duration	: 00:00:02:11
Total Requests	: 2,146
Average Speed	: 16.3r/s

Risk Level:
MEDIUM

40	IDENTIFIED
19	CONFIRMED
0	CRITICAL
0	HIGH
2	MEDIUM
0	LOW
19	BEST PRACTICE
19	INFORMATION

Identified Vulnerabilities

🟥 Critical	0
🟧 High	0
🟧 Medium	2
🟨 Low	0
🟦 Best Practice	19
🟦 Information	19
TOTAL	40

Confirmed Vulnerabilities

🟥 Critical	0
🟧 High	0
🟧 Medium	1
🟨 Low	0
🟦 Best Practice	0
🟦 Information	18
TOTAL	19

Vulnerability title: HTTP Strict Transport Security (HSTS) Policy not Enabled

Risk level – medium

Method – GET

OWASP TOP 10 – (A3)-2017

Description

A web server that implements HTTP Strict Transport Security (HSTS) specifies that compliant user agents—like a web browser—must communicate with it only over secure (HTTPS) connections. The "Strict-Transport-Security" HTTP response header field is how the server notifies the user agent of the HSTS Policy. The HSTS Policy establishes a window of time during which the user agent may only connect to the server securely.

Impact assessment

The following are some possible outcomes and hazards associated with the vulnerability relating to the domain https://www.redoxengine.com's lack of an activated HTTP Strict Transport Security (HSTS) policy:

I. Security weakness: Downgrading attacks are possible in the absence of a HSTS policy, which gives hostile actors the ability to intercept or alter unprotected connections between users and the website. This may lead to security flaws that jeopardize user information.

II. Man-in-the-middle attacts: The website is vulnerable to man-in-the-middle (MitM) attacks in the absence of HSTS, which allow hackers to intercept and perhaps alter user-website communication. Data theft, illegal access, and other nefarious acts may result from this.

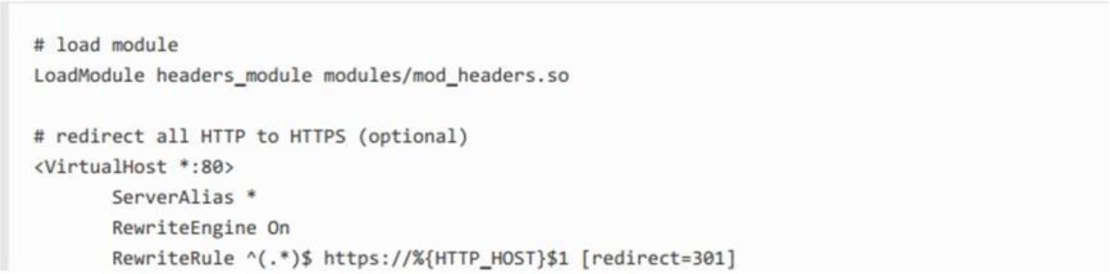
III. Phishing and data theft: By pretending to be www.echobox.com, cybercriminals may leverage this vulnerability to conduct phishing attacks and fool victims into disclosing sensitive information such login passwords, personal information, or bank account information. Financial losses and data theft may come from this.

IV. Loss of user trust: People anticipate private and safe internet communication. Users' confidence in the website's security may drop if HSTS isn't implemented, which may even lead them to stop visiting it altogether.

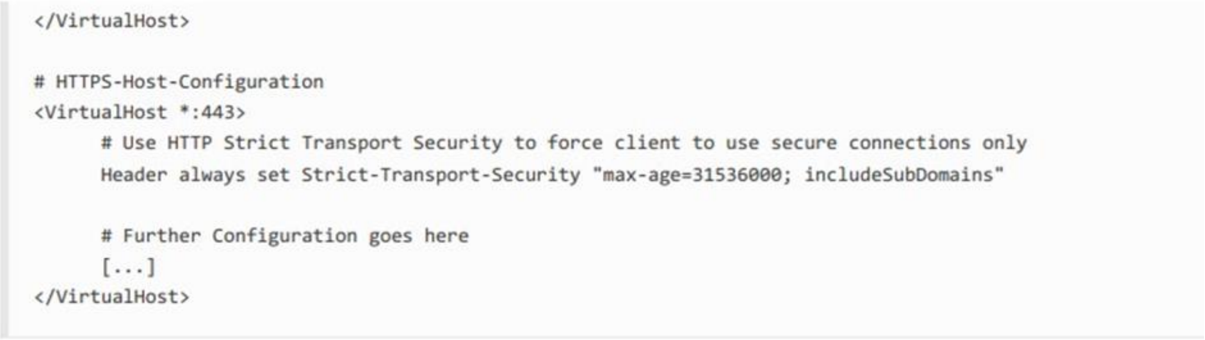
V. Regulatory non-compliance: Failure to implement HSTS may lead to violations of cybersecurity and data protection laws. Penalties and reputational harm are among the financial and legal repercussions that may result from this.

Turning on HSTS is essential to fixing this vulnerability. This security feature prevents downgrading attacks and improves overall security by guaranteeing that all communications with the website are encrypted and safe. Quick HSTS implementation can help the website comply with legal obligations, preserve user confidence, and protect user data. The precise consequences of this vulnerability are contingent upon the activities and potential attacks of malevolent individuals.

Solutions



```
# load module
LoadModule headers_module modules/mod_headers.so

# redirect all HTTP to HTTPS (optional)
<VirtualHost *:80>
       ServerAlias *
       RewriteEngine On
       RewriteRule ^(.*)$ https://%{HTTP_HOST}$1 [redirect=301]
```



```
</VirtualHost>

# HTTPS-Host-Configuration
<VirtualHost *:443>
      # Use HTTP Strict Transport Security to force client to use secure connections only
      Header always set Strict-Transport-Security "max-age=31536000; includeSubDomains"

      # Further Configuration goes here
      [...]
</VirtualHost>
```