Sri Lanka Institute of Information Technology

IE2062-Web Security

IT Number	Name
IT22581402	C.D.Aluthge

Information gathering and reconnaissance phase

a. Subdomain enumeration
    i. Recon-ng

b. Getting alive subdomains
    i. Nslookup

c. DNS enumeration
    i. Dnsrecon
    ii. Dnsdumper
    iii. Nikto
    iv. uniscan

d. Public devices enumeration
    i. Censys
    ii. Whatweb
    iii. Shodan

e. Find WAF (web application firewall) protection.
    i. Wafwoof

f. Find open ports.
    i. Nmap

g. Exploitation
    i. sqlmap

vulnerability analysis phase

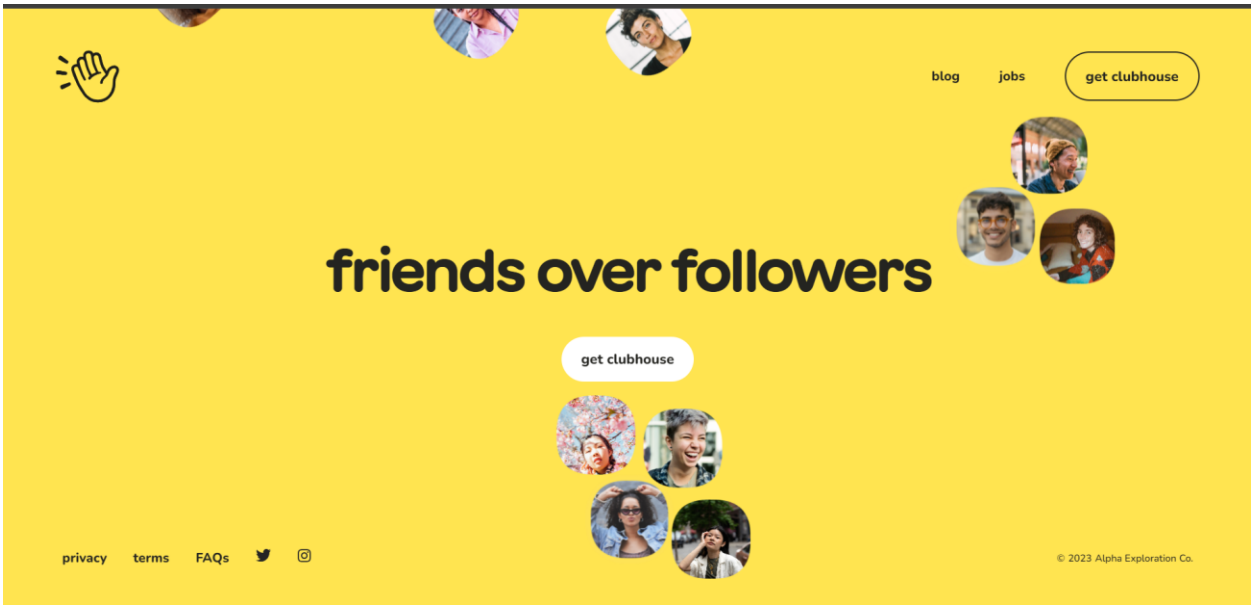
1. Target domain: http://www.clubhouse.com

    a. Content Security Policy (CSP) Header Not Set
    b. Hidden File Found (1)
    c. Strict-Transport-Security Header Not Set
    d. Cookie with SameSite Attribute None
    e. Cross-Domain JavaScript Source File Inclusion
    f. Timestamp Disclosure - Unix

Conclusion

Scope:

Clubhouse is an app for social networking that emphasizes audio communication. It enables users to construct and enter virtual rooms where they can talk with other users in real time about a variety of topics using audio. In its early days, the app was known for being exclusive and required an invitation to use. The clubhouse has been utilized for a variety of talks, from informal get-togethers to business networking and talks on certain sectors.



In Scope:

Asset name	Type	Coverage	Max. severity	Bounty	Last update
*.joinclubhouse.com	Wildcard	In scope	Critical	Eligible	May 15, 2023
com.clubhouse.android Android Application	Android: Play Store	In scope	Critical	Eligible	Nov 9, 2021
*.clubhouse.com	Wildcard	In scope	Critical	Eligible	May 15, 2023
*.clubhouseapi.com	Wildcard	In scope	Critical	Eligible	May 15, 2023
Clubhouse Production and Corporate Infrastructure	Other	In scope	Critical	Eligible	Nov 9, 2021
1503133294 iOS application	iOS: App Store	In scope	Critical	Eligible	Nov 9, 2021

OWASP top 10 vulnerabilities

- Broken Access Control
- Cryptographic Failures
- Injection
- Insecure Design
- Security Misconfiguration
- Vulnerable and Outdated Components
- Identification and Authentication Failures
- Software and Data Integrity Failures
- Security Logging and Monitoring Failures
- Server-Side Request Forgery

Broken Access Control

When users are unable to adequately impose limitations on what they can access or do within a system, this is known as broken access control. This vulnerability makes it possible for unauthorized individuals to access private data or carry out tasks that shouldn't be permitted. Insecure references to objects, improperly configured permissions, and a lack of authentication checks are the causes of it. Because it can result in illegal data exposure, manipulation, or system penetration, it's a major problem.

Cryptographic Failures

Cryptographic collapses are the result of poor algorithms, incorrect implementations, or improper management of encryption keys, which compromise the security of

information encrypted via encryption techniques. Undermining security safeguards and perhaps leading in data breaches, this can lead to unauthorized access to or exposure of sensitive data.

Injection

Injection: it is a type of security vulnerability in which untrusted data is sent to an interpreter as part of a command or query. Most often, attackers use this to inject malicious code into input fields, for example, login forms or search boxes. The interpreter then executes this code, which can give the attacker unauthorized access to data, alter the application's behavior, or gain full control over the system. There are many types of injections, such as SQL injection, where the attackers manipulate database queries, or cross-site scripting, where the malicious script is injected into a web page that other users view.

Insecure Design

Insecure design means the security flaws within the fundamental system or application design born out of insufficient attention to security at the design time. Consequently, systems have vulnerabilities that enable unauthorized system access or breaches. Examples are poor user authentication or no network segmentation. To fix insecure design, significant changes to the architecture are needed to develop the designs securely.

Security Misconfiguration

Imagine you accidentally leave your front door closed but not locked. That's what a security misconfiguration is like. Essentially, it's not properly securing your software systems, servers, or web applications. This can include default settings, shipped insecurely, or unnecessary features. Unauthorized users might target such

vulnerabilities to access your info or disrupt the operation. Therefore, one should monitor and update security measures continuously to avoid misconfiguration.

Vulnerable and Outdated Components

Vulnerable components are software elements that are known to contain security flaws that could be exploited by hackers. However, software components that have not received an upgrade in a long time are known as outdated components, and they may not include important security updates and bug fixes. In order to maintain a secure system environment, both sorts of components present security concerns and should be routinely updated and monitored.

Identification and Authentication Failures

When systems are unable to accurately confirm user identities or authenticate their access credentials, identification and authentication failures take place. Whereas authentication failures are caused by an inability to verify user identities, identification failures are the consequence of incorrect user recognition. In order to identify and fix system vulnerabilities, strong authentication procedures and frequent security assessments are essential. These failures might result in unauthorized access and security breaches.

Software and Data Integrity Failures

Software code or data accuracy, consistency, and reliability are jeopardized in software and data integrity issues. Data integrity problems are caused by mistakes, unauthorized access, or cyberattacks, whereas programming errors, malicious code, or unauthorized alterations are the causes of software integrity problems. To reduce risks and guarantee system security and dependability, preventive measures include backups, data encryption, access controls, and regular updates.

Security Logging and Monitoring Failures

Failed security logging and monitoring systems result in a delayed detection of security incidents and higher risks since they are unable to reliably capture and evaluate security-related events. Monitoring failures are caused by insufficient tools or response protocols, whereas logging failures are caused by misconfigurations or deactivated logging systems. Organizations should establish strong response protocols, use comprehensive logging practices, deploy efficient monitoring tools like SIEM tools, train security teams, and continuously improve logging and monitoring configurations in light of best practices and emerging threats in order to address these failures. By taking these steps, cybersecurity defenses are strengthened and the effects of security events are lessened.

Server-Side Request Forgery

A security flaw known as server-side request forgery (SSRF) allows attackers to take control of a susceptible server and send unwanted requests, usually to external or internal systems. Unauthorized access, data loss, and more exploitation may result from this. Input validation, whitelisting authorized resources, network segmentation, and access controls are all part of prevention. Frequent security testing improves overall system security by assisting in the identification and mitigation of SSRF threats.

Information gathering and reconnaissance phase.

The objective is to gather as much pertinent data as you can about the intended system or organization. Understanding the target's infrastructure, seeing potential vulnerabilities, and organizing additional security testing operations are all made easier with the aid of this information.

Domain: http://www.clubhouse.com

Subdomain enumeration

- Recon-ng

Recon-ng is an open-source reconnaissance tool with Python foundation that is used for penetration testing and security evaluations. It collects data from domains, IPs, emails, and social media platforms using a modular framework with different modules. The advantages of recon-ng are its flexibility for scripting and automation, integration with many data sources, and data enrichment capabilities. It is used by security experts to

gather and evaluate intelligence, spot possible weaknesses, and improve overall security posture during the first reconnaissance phase. Its vibrant community guarantees continuous upgrades and support, which makes it an invaluable addition to cybersecurity toolkits.

Proof of concept:

- To get google website give this command.

```
[recon-ng][default] > marketplace search google
[*] Searching module index for 'google' ...
  +-----------------------------------------------------------------------------+
  |              Path                | Version |   Status  |  Updated   | D | K |
  +-----------------------------------------------------------------------------+
  | recon/domains-hosts/google_site_web | 1.0     | installed | 2019-06-24 |   |   |
  +-----------------------------------------------------------------------------+

  D = Has dependencies. See info for details.
  K = Requires keys. See info for details.

[recon-ng][default] > marketplace install recon/domains-hosts/google_site_web
[*] Module installed: recon/domains-hosts/google_site_web
[*] Reloading modules ...
[recon-ng][default] > show info
Shows various framework items

Usage: show <companies|contacts|credentials|domains|hosts|leaks|locations|netblocks|ports|profiles|pushpins|re
positories|vulnerabilities>
```

- You can see it’s not installed yet. We must download installation path.

```
[recon-ng][default] > marketplace search google
[*] Searching module index for 'google' ...
  +-----------------------------------------------------------------------------+
  |              Path                | Version |   Status  |  Updated   | D | K |
  +-----------------------------------------------------------------------------+
  | recon/domains-hosts/google_site_web | 1.0     | installed | 2019-06-24 |   |   |
  +-----------------------------------------------------------------------------+

  D = Has dependencies. See info for details.
  K = Requires keys. See info for details.

[recon-ng][default] > modules load recon/domains-hosts/google_site_web
```

- After installing path using show info to see its download or not
- Load the installed module path and use info see options.
- Go to options and set source to our targeted domain indrive.com and run it.

```
[recon-ng][default][google_site_web] > info

      Name: Google Hostname Enumerator
    Author: Tim Tomes (@lanmaster53)
   Version: 1.0

Description:
  Harvests hosts from Google.com by using the 'site' search operator. Updates the 'hosts' table with
  the results.

Options:
  Name    Current Value  Required  Description
  ------  -------------  --------  -----------
  SOURCE  smtp2go.com    yes       source of input (see 'info' for details)

Source Options:
  default         SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
  <string>        string representing a single input
  <path>          path to a file containing a list of inputs
  query <sql>     database query returning one column of inputs

[recon-ng][default][google_site_web] > options set source clubhouse.com
SOURCE => clubhouse.com
[recon-ng][default][google_site_web] > run
```

Proof of concept:

```
[*] Host: support.clubhouse.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] ----------------------------------------------------------
[*] Country: None
[*] Host: blog.clubhouse.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] ----------------------------------------------------------
[*] Country: None
[*] Host: newsletter.clubhouse.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] ----------------------------------------------------------
```

```
[*] Host: www.clubhouse.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] ------------------------------------------------
[*] Searching Google for: site:clubhouse.com -site:support.clubhouse.com -site:blog.clubhouse.com -site:newsletter.clubhouse.com -site:www.clubhouse.com
[*] No New Subdomains Found on the Current Page. Jumping to Result 201.
[*] Searching Google for: site:clubhouse.com -site:support.clubhouse.com -site:blog.clubhouse.com -site:newsletter.clubhouse.com -site:www.clubhouse.com
[*] No New Subdomains Found on the Current Page. Jumping to Result 301.
[*] Searching Google for: site:clubhouse.com -site:support.clubhouse.com -site:blog.clubhouse.com -site:newsletter.clubhouse.com -site:www.clubhouse.com
[*] No New Subdomains Found on the Current Page. Jumping to Result 401.
[*] Searching Google for: site:clubhouse.com -site:support.clubhouse.com -site:blog.clubhouse.com -site:newsletter.clubhouse.com -site:www.clubhouse.com
[*] Country: None
[*] Host: ios.clubhouse.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
```

Proof of concept:

```
SUMMARY
-------
[*] 5 total (5 new) hosts found.
[recon-ng][default][google_site_web] >
```

Getting alive subdomains

- Nslookup

A command-line utility called `nslookup` is used to query DNS (Domain Name System) servers and get DNS-related data. In addition to performing reverse DNS lookups and retrieving other DNS records like A, AAAA, MX, NS, PTR, and TXT records, it resolves domain names to IP addresses. It is flexible for diagnosing DNS problems, validating DNS configurations, and testing DNS servers because it may be used in both interactive and non-interactive modes. Network administrators and cybersecurity experts frequently utilize `nslookup`, which runs on several platforms, for DNS-related activities and diagnostics.

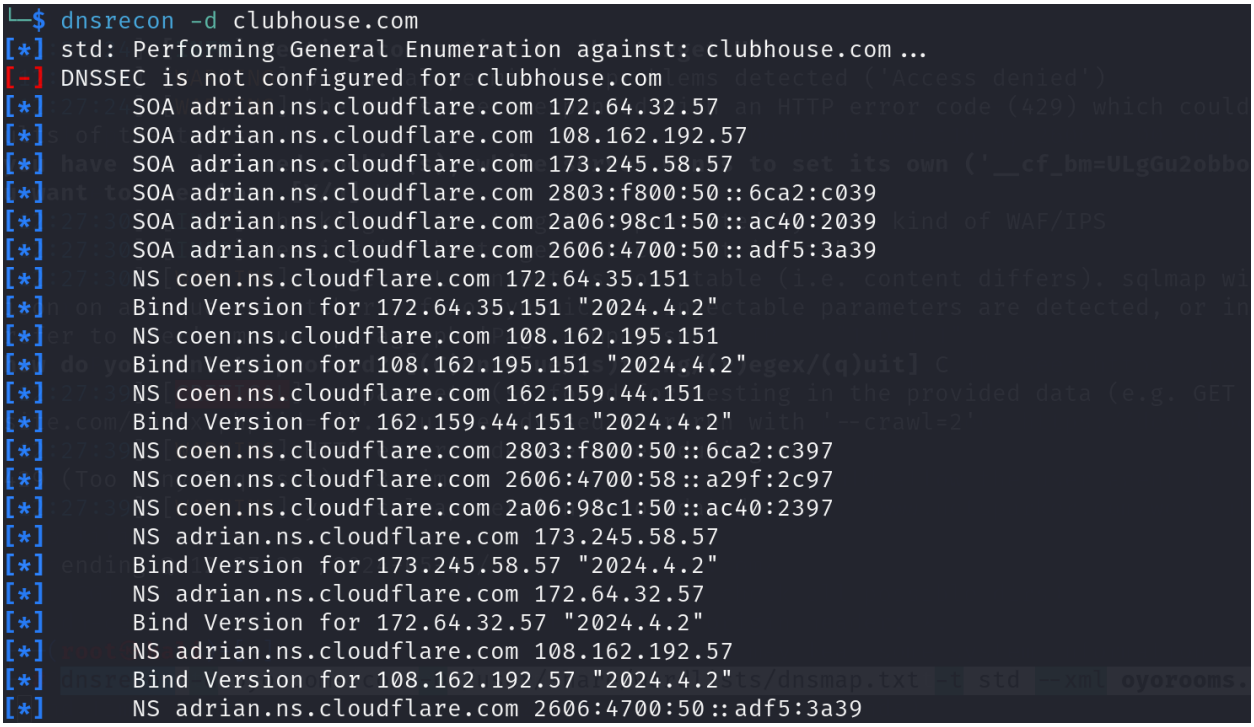
Proof of concept:

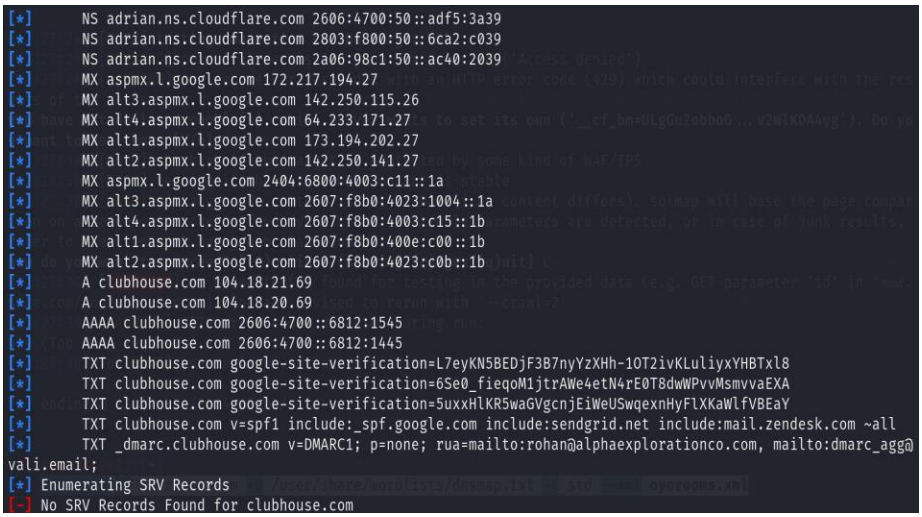
DNS enumeration

- Dnsrecon

For security evaluations and penetration tests, DNSRecon is a powerful open-source tool for DNS reconnaissance that finds and counts DNS information. It is particularly good at finding subdomains, enumerating DNS records (A, AAAA, MX, NS, SOA, TXT), transferring DNS zones, and it can handle wordlist-based and brute-force attacks. With its automation, customization, and integration features, it's a useful tool for figuring out possible attack points, comprehending target infrastructure, and raising security awareness in general. DNSRecon is still a dependable option for DNS reconnaissance operations because of its community support and frequent updates.

Proof of concept:

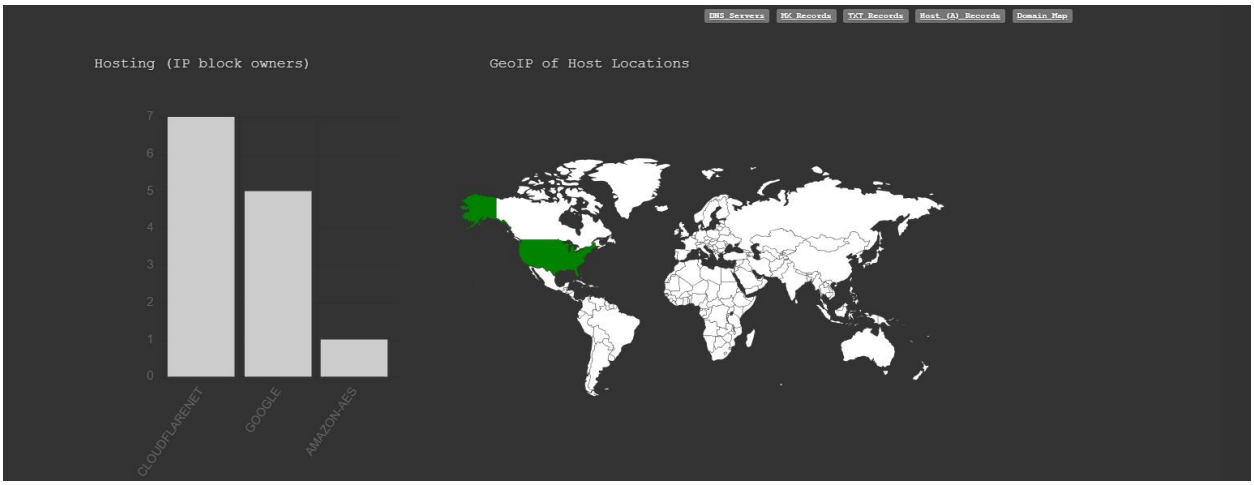




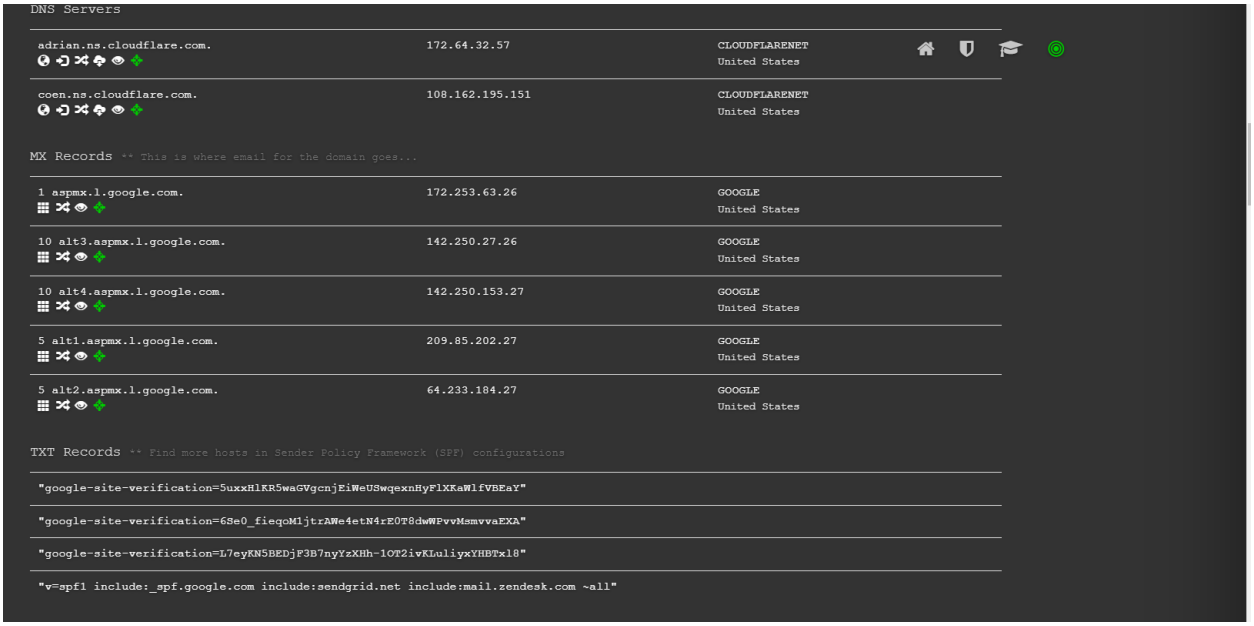
- Dnsdumper

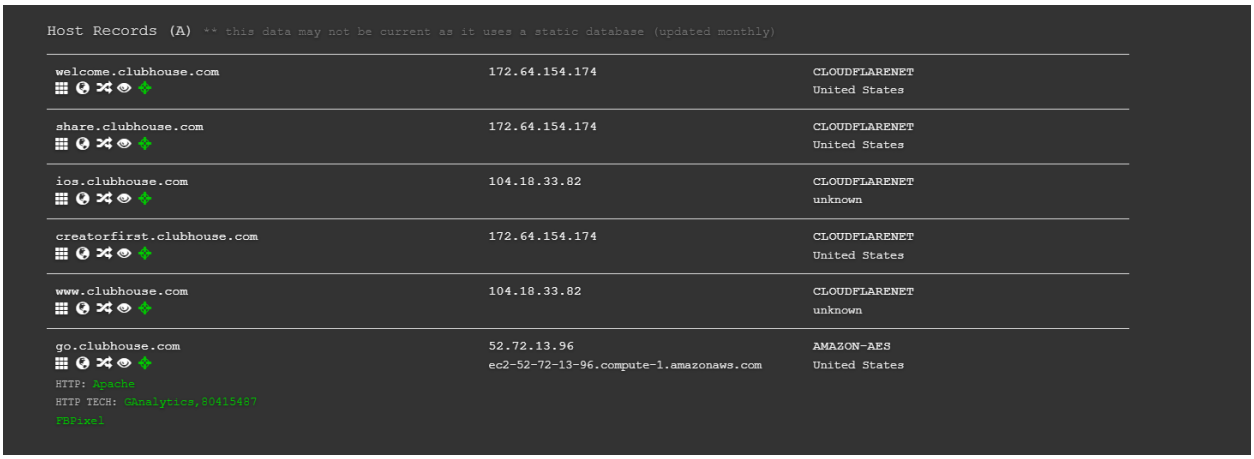
DNSDumpster is an online tool focused on DNS information gathering for a target domain. It assists in discovering subdomains, viewing DNS records (A, AAAA, MX, NS, SOA, TXT), mapping domain names to IP addresses, and providing visual representations of DNS-related data. Security professionals and penetration testers use DNSDumpster during reconnaissance to understand a domain's infrastructure, identify potential vulnerabilities, and aid in security assessments. While it offers valuable insights, users should supplement its findings with other tools for a comprehensive analysis, considering that its data may not always be the most current or complete.

- **Proof of concept:**



- **Proof of concept:**

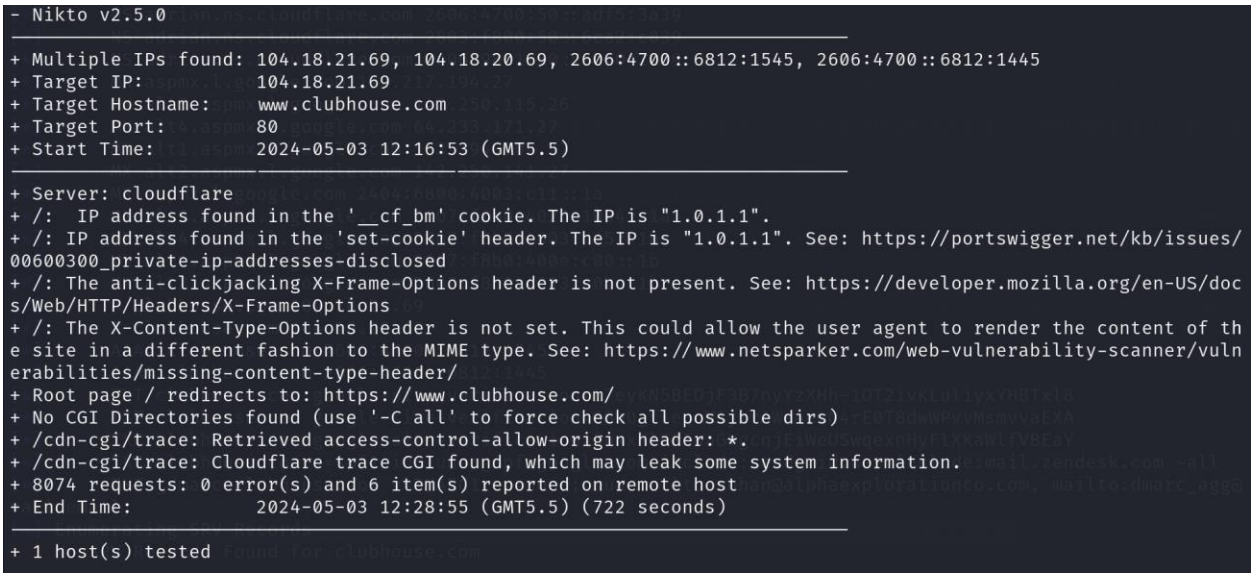




- Nikto

Nikto is an open-source web server scanner made to find possible security holes in applications and web servers. With its extensive scanning capabilities, it may identify problems including out-of-date software, unsafe configurations, weak scripts, and incorrect SSL/TLS settings. In addition to producing comprehensive results after scanning, Nikto may be customized and scripted, connects with automated workflows, and receives community assistance and frequent database upgrades. For security experts performing web security assessments, it's a useful tool that helps with vulnerability discovery and mitigation to improve overall security posture.

- **Proof of concept:**



- **Uniscan**

A free penetration testing tool is Uniscan. This program is used to check for vulnerabilities in web applications. With the help of this scan, we can check the target online application for SQL injection, cross-site scripting (XSS), PHP injection, Remote file inclusion (LFI), remote command execution, web shell vulnerabilities, and backup files.

- **Proof of concept:**

```
└─# sudo uniscan -u http://www.clubhouse.com -qweds
########################################
# Uniscan project                      #
# http://uniscan.sourceforge.net/      #
########################################
V. 6.3

Scan date: 3-5-2024 14:25:15
============================================================================================
| Domain: http://www.clubhouse.com/
| Server: cloudflare
| IP: 104.18.20.69
============================================================================================
|
| Directory check:
| Skipped because http://www.clubhouse.com/uniscan518/ did not return the code 404
============================================================================================
|
| File check:
| Skipped because http://www.clubhouse.com/uniscan876/ did not return the code 404
============================================================================================
```

- **Proof of concept:**

```
| Check robots.txt:
| Check sitemap.xml:
============================================================================================
|
| Crawler Started:
| Plugin name: Timthumb ⩽ 1.32 vulnerability v.1 Loaded.
| Plugin name: Web Backdoor Disclosure v.1.1 Loaded.
| Plugin name: External Host Detect v.1.2 Loaded.
| Plugin name: Code Disclosure v.1.1 Loaded.
| Plugin name: E-mail Detection v.1.1 Loaded.
| Plugin name: FCKeditor upload test v.1 Loaded.
| Plugin name: Upload Form Detect v.1.1 Loaded.
| Plugin name: phpinfo() Disclosure v.1 Loaded.
| [+] Crawling finished, 1 URL's found!
```

```
| Timthumb:
|
| Web Backdoors:
|
| External hosts:
|
| Source Code Disclosure:
|
| E-mails:
|
| FCKeditor File Upload:
|
| File Upload Forms:
|
| PHPinfo() Disclosure:
|
| Ignored Files:
```

- **Proof of concept:**

```
| Backup Files:
| Skipped because http://www.clubhouse.com/testing123 did not return the code 404
|
|
| Blind SQL Injection:
|
|
| Local File Include:
|
| PHP CGI Argument Injection:
|
|
| Remote Command Execution:
|
| Remote File Include:
|
|
| SQL Injection:
|
| Cross-Site Scripting (XSS):
|
|
```

Public devices enumeration

Censys

Censys is a potent internet scanning tool that scans and monitors devices, networks, and services online all the time. It performs thorough scans, keeps track of SSL/TLS certificates, keeps an eye on attack surfaces, finds vulnerabilities, and offers threat information. Large volumes of data may be searched and analyzed, processes can be automated with the help of the API, and Censys can be integrated into security workflows. In the connected digital world of today, Censys is useful for asset discovery, vulnerability management, compliance monitoring, and proactive risk reduction.

Proof of concept:

Basic Information

Forward DNS	email.n.danscomp.com, expertpackaging.au, bsl-pim-production.merchantturnkey.com, www.danscomp.com, helpcenter.arlo.com.cdn.cloudflare.net, ...
Routing	104.18.16.0/20 via CLOUDFLARENET, US (AS13335)
Services (13)	80/HTTP, 443/HTTP, 2052/HTTP, 2053/HTTP, 2082/HTTP, 2083/HTTP, 2086/HTTP, 2087/HTTP, 2095/HTTP, 2096/HTTP, 8080/HTTP, 8443/HTTP, 8880/HTTP

HTTP 80/TCP

05/02/2024 17:57 UTC

VIEW ALL DATA | GO

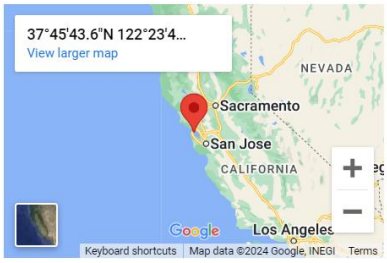
Software

CloudFlare Load Balancer

Details

http://104.18.20.69/

Status	403 Forbidden
Body Hash	sha1:78449f7152a2281169fcc9ae889c4b9631d7dbdd
HTML Title	Direct IP access not allowed | Cloudflare
Response Body	EXPAND



Geographic Location

City	San Francisco
State	California
Country	United States (US)
Coordinates	37.7621, -122.3971
Timezone	America/Los_Angeles

Proof of concept:

HTTP 443/TCP

05/02/2024 17:35 UTC

VIEW ALL DATA | GO

Software

CloudFlare Load Balancer

Details

http://104.18.20.69:443/

Status	400 Bad Request
Body Hash	sha1:108b6115dc6ebfde76aef4336126f605252d957f
HTML Title	400 The plain HTTP request was sent to HTTPS port
Response Body	EXPAND

HTTP 2052/TCP

05/03/2024 11:01 UTC

VIEW ALL DATA | GO

Software

CloudFlare Load Balancer

Details

http://104.18.20.69:2052/

Status	403 Forbidden
Body Hash	sha1:b814d51b490dd6276cd83fc9510d72089171ae96
HTML Title	Direct IP access not allowed | Cloudflare
Response Body	EXPAND

HTTP 2053/TCP

05/03/2024 10:08 UTC

VIEW ALL DATA | GO

Software

CloudFlare Load Balancer

Details

http://104.18.20.69:2053/

Status	400 Bad Request
Body Hash	sha1:108b6115dc6ebfde76aef4336126f605252d957f
HTML Title	400 The plain HTTP request was sent to HTTPS port
Response Body	EXPAND

HTTP 2082/TCP

05/02/2024 14:09 UTC

VIEW ALL DATA | GO

Software

CloudFlare Load Balancer

Details

http://104.18.20.69:2082/

Status	403 Forbidden
Body Hash	sha1:3fa428514793e2147269660cb07575b4876909dc
HTML Title	Direct IP access not allowed | Cloudflare
Response Body	EXPAND

Proof of concept:

HTTP 2083/TCP

05/03/2024 16:52 UTC

VIEW ALL DATA | GO

Software

CloudFlare Load Balancer

Details

http://104.18.20.69:2083/

Status	400 Bad Request
Body Hash	sha1:108b6115dc6ebfde76aef4336126f605252d957f
HTML Title	400 The plain HTTP request was sent to HTTPS port
Response Body	EXPAND

HTTP 2086/TCP

05/03/2024 05:04 UTC

VIEW ALL DATA | GO

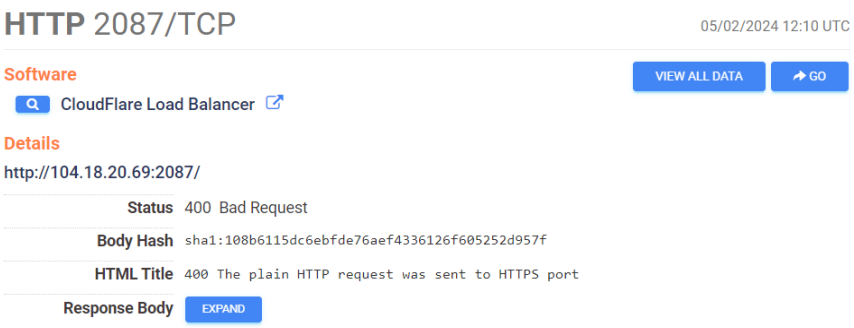
Software

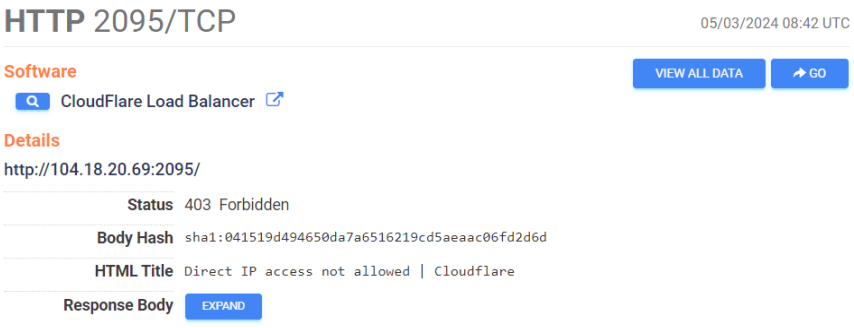
CloudFlare Load Balancer

Details

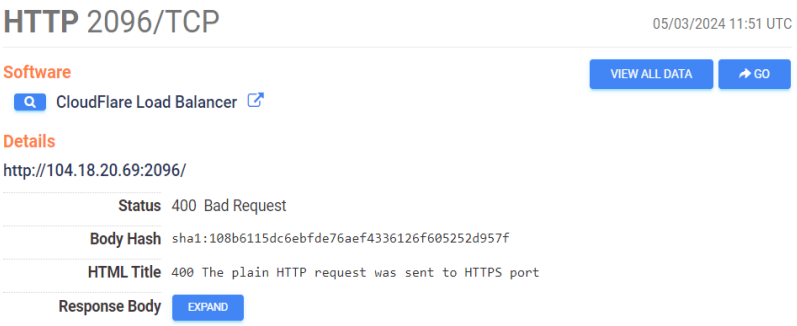
http://104.18.20.69:2086/

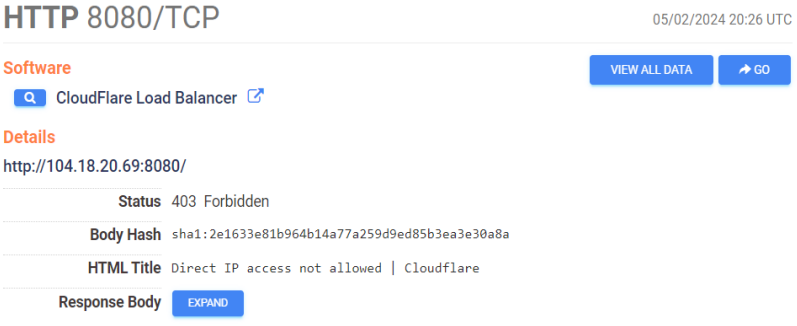
Status	403 Forbidden
Body Hash	sha1:40ff5f859f4b2be2c48a842eebfa3ed54786d207
HTML Title	Direct IP access not allowed | Cloudflare
Response Body	EXPAND

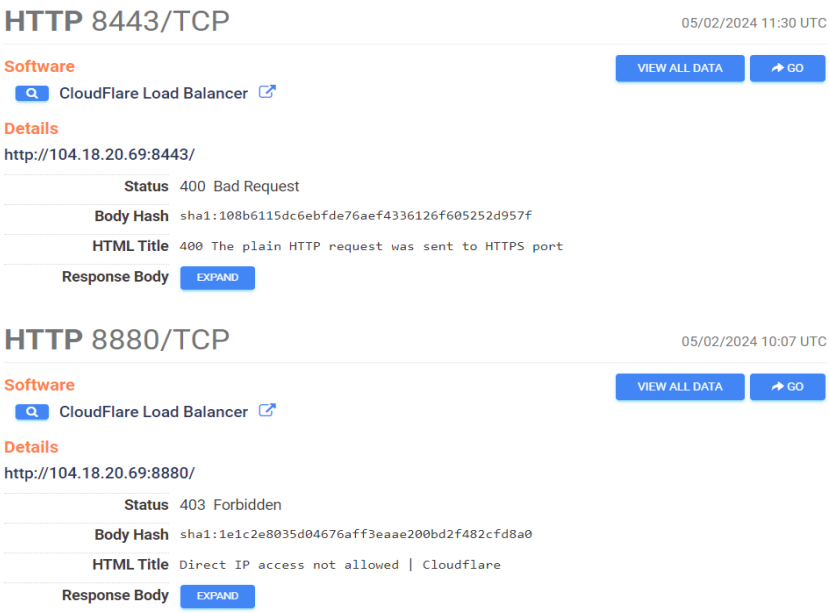




Proof of concept:







- **Whatweb**

WhatWeb is a powerful open-source website scanner that specializes in identifying the technologies utilized by websites. It excels in detecting web servers, frameworks, content management systems (CMS), programming languages, and more through its fingerprinting and HTTP header analysis capabilities. Security professionals leverage WhatWeb during reconnaissance and vulnerability assessments to gather insights into target websites' technology stacks, assess security risks associated with detected technologies, and identify potential vulnerabilities or outdated components. Its scriptable and customizable nature, along with integration options, makes it a valuable tool for automating scanning workflows and enhancing security assessments.

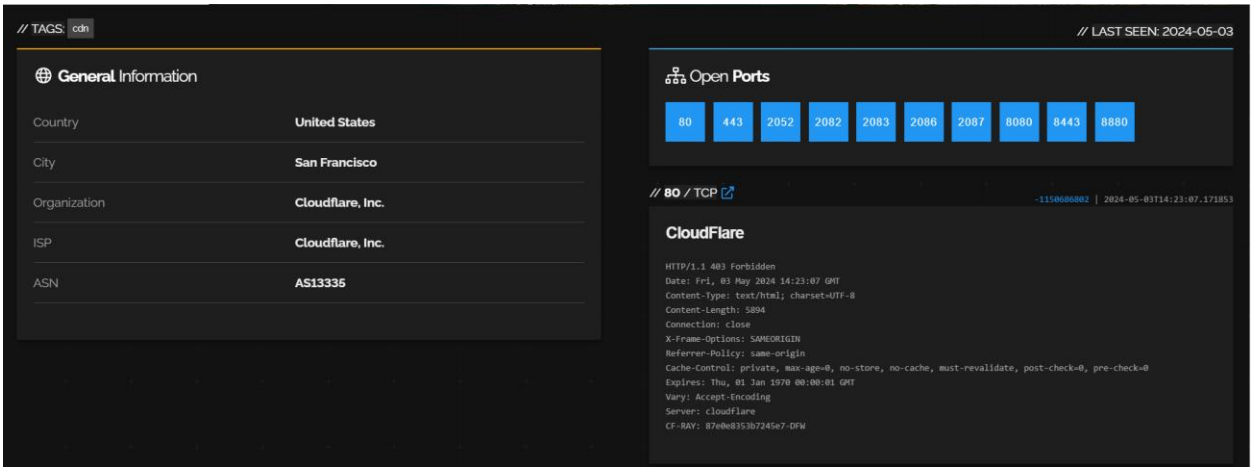
Proof of concept:

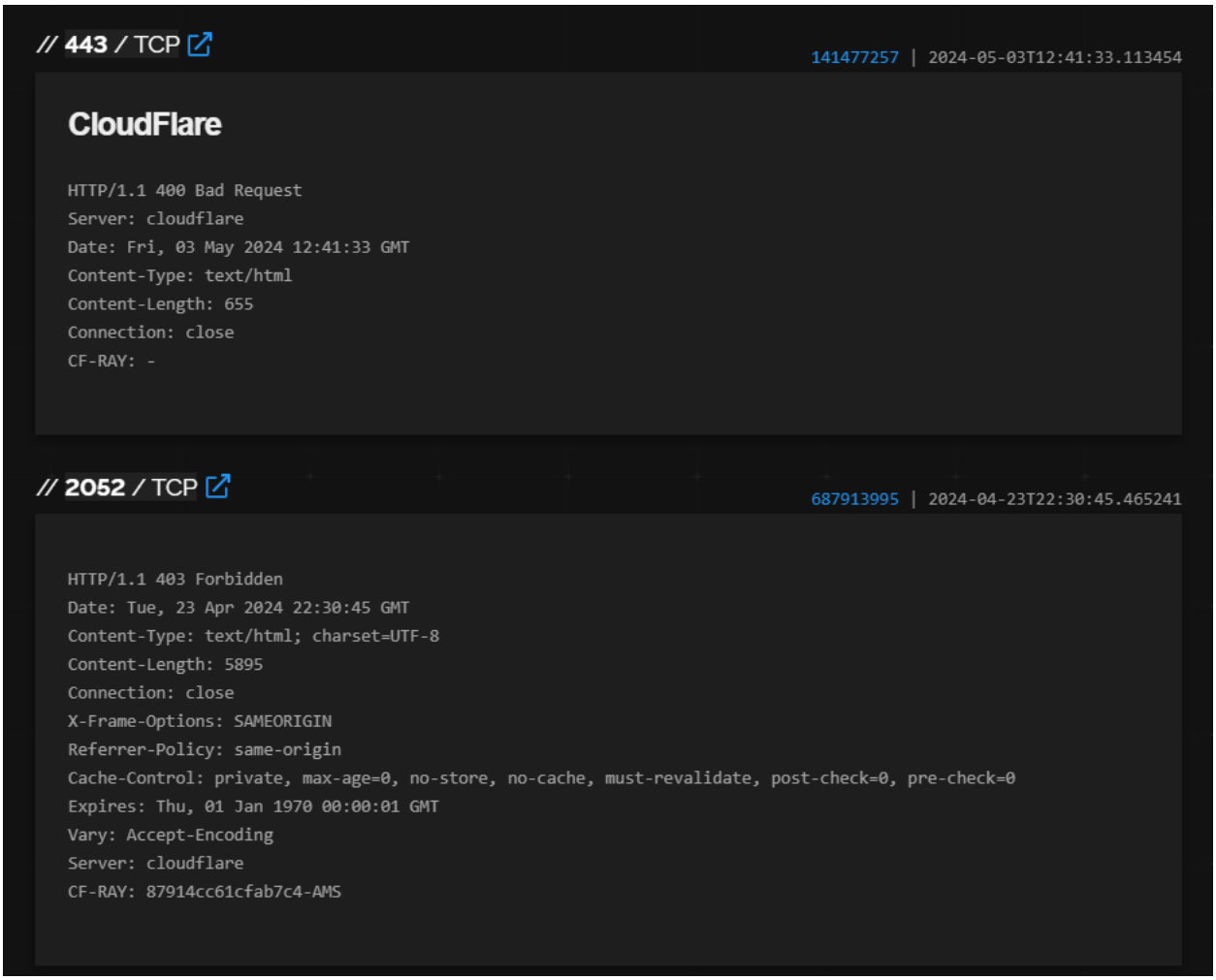
```
┌──(root㉿kali)-[~]
└─# whatweb clubhouse.com
http://clubhouse.com [301 Moved Permanently] Cookies[__cf_bm], Country[UNITED STATES][US], HTTPServer[cloudfla
re], HttpOnly[__cf_bm], IP[104.18.21.69], RedirectLocation[https://www.clubhouse.com], Title[301 Moved Permane
ntly], UncommonHeaders[cf-ray]
https://www.clubhouse.com [200 OK] Cookies[__cf_bm], Country[UNITED STATES][US], Email[6ce348d3f97e4220a9db5e6
9183fcd17@o325556.ingest.sentry.io], Frame, HTTPServer[cloudflare], HttpOnly[__cf_bm], IP[104.18.21.69], Open-
Graph-Protocol[website], Script[application/json,text/javascript], Title[Clubhouse][Title element contains new
line(s)!], UncommonHeaders[x-content-type-options,referrer-policy,cross-origin-opener-policy,cf-cache-status,c
f-ray], X-Frame-Options[DENY], X-UA-Compatible[ie=edge]
```

- Shodan

Shodan is a specialized search engine that indexes and provides information about internet-connected devices and systems, such as webcams, routers, servers, and IoT devices. It offers details like open ports, running services, device types, locations, and vulnerabilities. While it's a valuable tool for security professionals and researchers to analyze networks and devices, it's important to use it ethically and legally, as unauthorized access to devices is against the law.

Proof of concept:





Proof of concept:

// 2082 / TCP

280086675 | 2024-05-03T07:46:26.162558

```
HTTP/1.1 403 Forbidden
Date: Fri, 03 May 2024 07:46:26 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 5892
Connection: close
X-Frame-Options: SAMEORIGIN
Referrer-Policy: same-origin
Cache-Control: private, max-age=0, no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Expires: Thu, 01 Jan 1970 00:00:01 GMT
Vary: Accept-Encoding
Server: cloudflare
CF-RAY: 87dea3211d429274-FRA
```

// 2083 / TCP

141477257 | 2024-05-03T15:35:32.161649

```
HTTP/1.1 400 Bad Request
Server: cloudflare
Date: Fri, 03 May 2024 15:35:32 GMT
Content-Type: text/html
Content-Length: 655
Connection: close
CF-RAY: -
```

// 2086 / TCP

891399479 | 2024-05-03T15:58:25.744368

```
HTTP/1.1 403 Forbidden
Date: Fri, 03 May 2024 15:58:25 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 5896
Connection: close
X-Frame-Options: SAMEORIGIN
Referrer-Policy: same-origin
Cache-Control: private, max-age=0, no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Expires: Thu, 01 Jan 1970 00:00:01 GMT
Vary: Accept-Encoding
Server: cloudflare
CF-RAY: 87e173d27dfe18e4-FRA
```

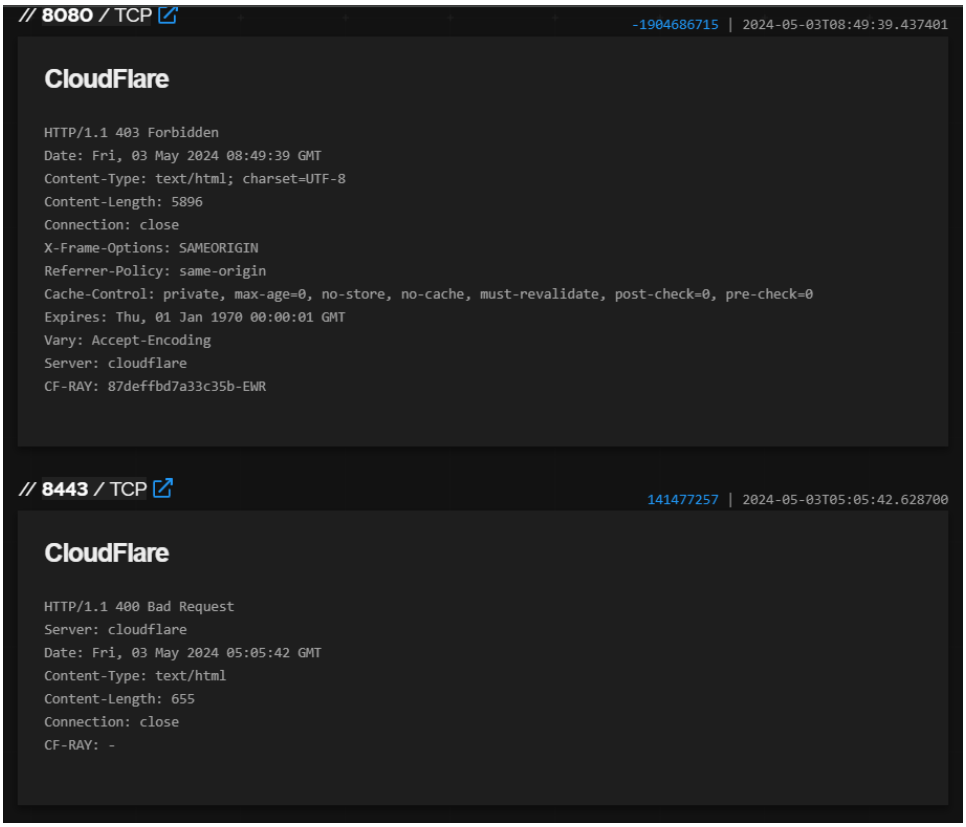
// 2087 / TCP

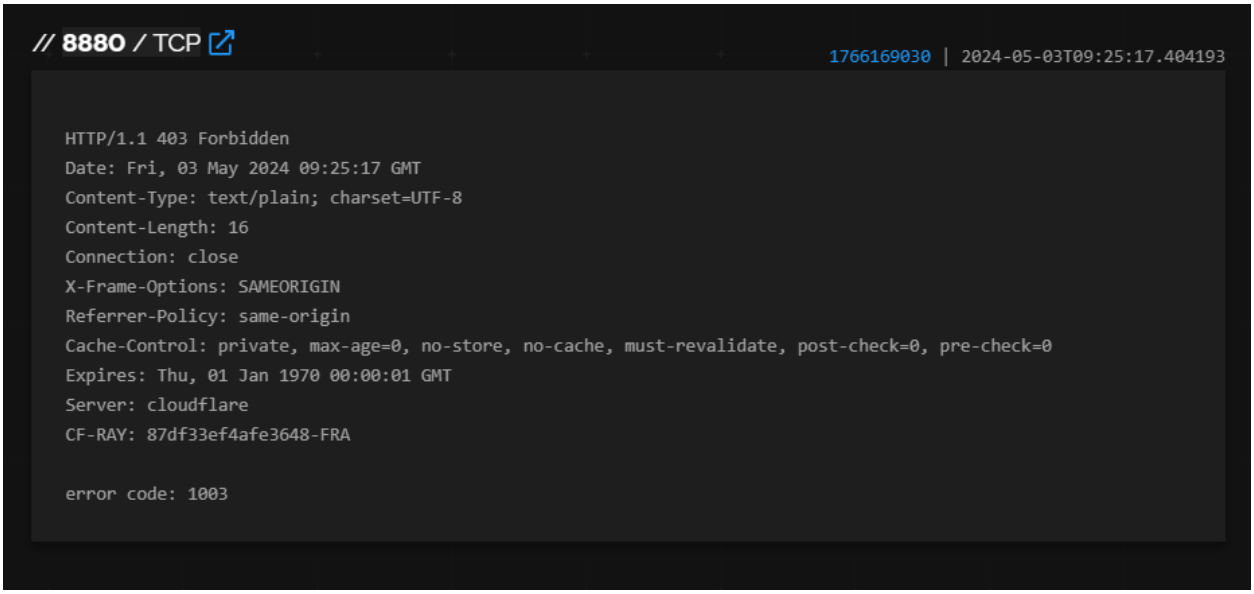
-741334147 | 2024-05-03T17:30:18.695628

```
HTTP/1.1 400 Bad Request
Server: cloudflare
Date: Fri, 03 May 2024 17:30:18 GMT
Content-Type: text/html
Content-Length: 155
Connection: close
CF-RAY: -

<html>
<head><title>400 Bad Request</title></head>
<body>
<center><h1>400 Bad Request</h1></center>
<hr><center>cloudflare</center>
</body>
</html>
```

Proof of concept:



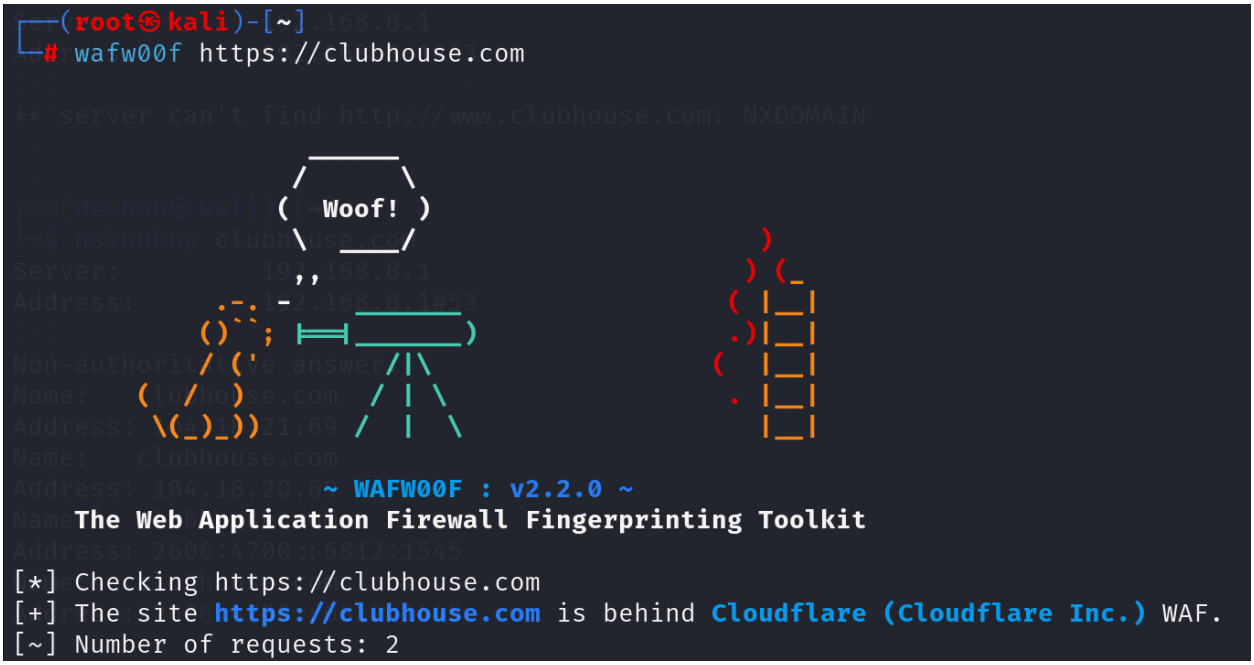


Find WAF (web application firewall) protection.

- Wafwoof

Wafw00f is an open-source program that uses content patterns, HTTP responses, and header analysis to detect and fingerprint web application firewalls (WAFs). It gives testers and security experts information about the particular WAF technology or vendor being used as well as assists them in determining whether a web application is protected by a WAF. In addition to being user-friendly and seamlessly integrating with automated testing workflows, Wafw00f also keeps an updated database of WAF signatures and offers community support. It is a useful tool for security assessment and reconnaissance jobs, supporting the creation of efficient testing plans and workarounds.

Proof of concept:



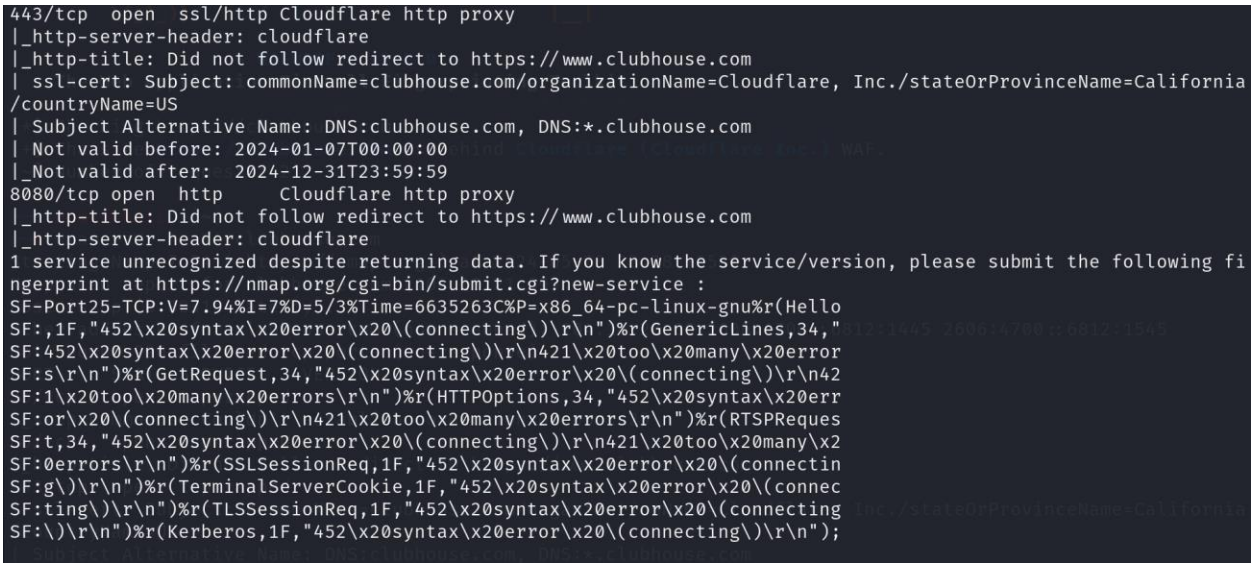
Find open ports.

- Nmap

Nmap, sometimes known as "Network Mapper," is a potent open-source network scanning program used for vulnerability analysis, security audits, and network discovery. It is particularly good at operating system detection, host discovery, port scanning, and service version detection. Because it supports custom scripting and automation, Nmap's scripting engine (NSE) is adaptable to a wide range of security activities and integration requirements. Because of its dependability, adaptability, and large feature set, network administrators, security experts, and penetration testers frequently use it for network reconnaissance, security audits, and network monitoring.

Proof of concept:

Proof of concept:

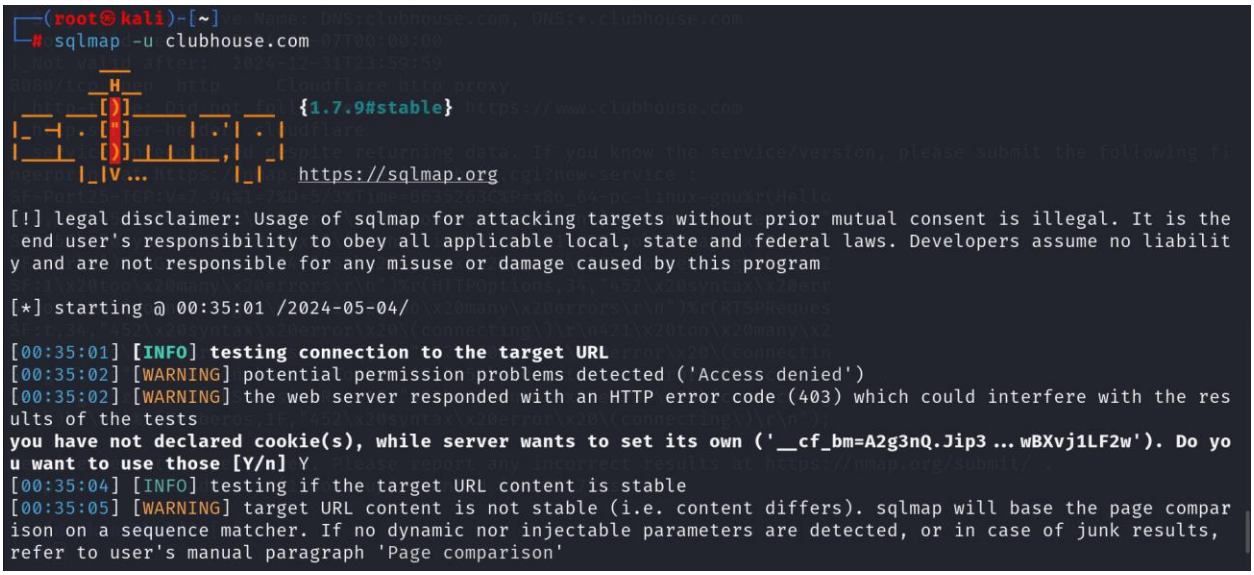


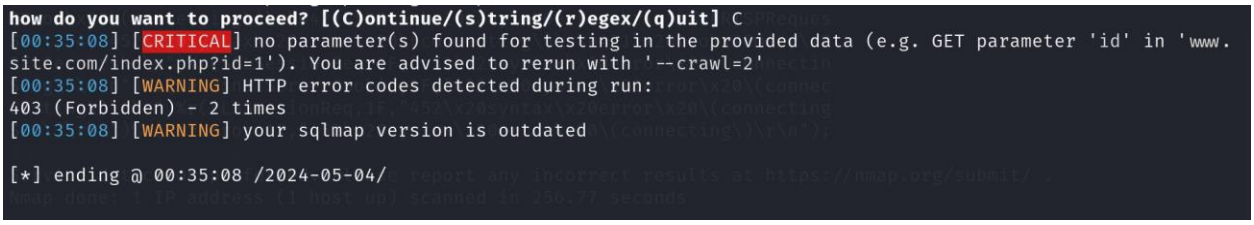
Exploitation

- Sqlmap

Web applications with SQL injection vulnerabilities can be automatically found and exploited with SQLMap, an open-source penetration testing tool. It employs a number of detecting methods, supports a number of database management systems, and provides customization choices. Security experts and penetration testers find SQLMap useful as it automates testing, enumeration, data extraction, and reporting procedures during security assessments. It should, therefore, be utilized sensibly, morally, and with the appropriate authorization to test programs and systems.

Proof of concept:





Vulnerability analysis phase

I used tool like ZAP (zed attack proxy) to process and catch bugs and vulnerabilities are based on OWASP top 10.

Targeted Domain: - http://www.clubhouse.com

- ZAP (zed attack proxy)

OWASP created the open-source Zed Attack Proxy (ZAP) tool for online application security testing. Users can examine and alter HTTP/HTTPS communication between web browsers and programs by using it as an intercepting proxy. ZAP provides spidering to map application structures in addition to active and passive scanning for vulnerabilities like SQL injection and XSS. It facilitates input validation testing with fuzzing, controls sessions for authentication verification, and offers automation via scripting and APIs. ZAP helps with risk management and safe application development by providing thorough reports on vulnerabilities that have been found.

Vulnerability title

- Content Security Policy (CSP) Header Not Set

Vulnerability description

A web server may include different headers in its response to a browser request in order to offer more guidance or security precautions. Among these headers is the Content Security Policy (CSP) header, which specifies the locations from which scripts, stylesheets, and pictures can be loaded, hence aiding in the prevention of specific attacks like Cross-Site Scripting (XSS).

A web server's response omitting the CSP header indicates that the resource or webpage in question does not have the Content Security

Policy enabled. This lack of protection can be dangerous since it makes the website more susceptible to XSS attacks and other security flaws by enabling potentially harmful scripts or material to be executed.

In order to identify the reliable sources from which content can be loaded, web developers and administrators should set up and include CSP headers in their web server responses. This improves the website's overall security posture and lessens the chance of harmful or unauthorized code execution.

Impact assessment

The absence of a Content Security Policy (CSP) header poses significant risks across various domains. Security vulnerabilities, especially Cross-Site Scripting (XSS) attacks, are heightened, potentially leading to data breaches and unauthorized access. This not only damages user trust and tarnishes the website's reputation but also exposes organizations to legal and compliance issues, with possible fines and operational disruptions.

Affected components

The absence of a Content Security Policy (CSP) header impacts several components within a web environment. It leaves web pages vulnerable to attacks like XSS, affecting client-side scripts, third-party resources,

user input handling, and authentication mechanisms. These vulnerabilities can compromise sensitive data and lead to unauthorized access. Implementing CSP headers and enforcing proper security policies are crucial steps to protect these components, reduce risks, and bolster overall web security.

How to mitigate?

Mitigating the risks associated with a "Content Security Policy (CSP) Header Not Set" issue involves implementing CSP headers and defining appropriate security policies. Here are steps to mitigate the risks effectively:

1. Implement CSP Headers - Configure web servers to include CSP headers in HTTP responses for all web pages. This header should specify directives that control the behavior of resources such as scripts, stylesheets, fonts, and images.

2. Define CSP Directives - Specify trusted sources for loading resources using CSP directives such as `default-src`, `script-src`, `style-src`, `img-src`, `font-src`, and others. Limit the sources to only those necessary for the website's functionality.

3. Use Nonces and Hashes - For inline scripts and styles that must be included, use nonces (random values) or hashes in CSP directives

(`script-src 'nonce-12345'`) to allow only specific scripts identified by the nonce or hash.

4. Report Violations - Configure CSP to report policy violations (`report-uri` directive) to a designated endpoint. Monitoring these reports helps identify issues during the development phase and allows fine-tuning of CSP policies.

5. Test and Validate - Use CSP testing tools and browser developer tools to validate CSP policies and ensure they do not inadvertently block legitimate resources or functionalities.

6. Educate Developers - Train developers on best practices for secure coding, emphasizing the importance of CSP and how to implement and maintain CSP headers effectively.

7. Regularly Update Policies - Periodically review and update CSP policies as the website's content and functionality evolve. Consider adding or modifying directives based on changes in resource loading requirements.

8. Implement Other Security Measures - Complement CSP with other security measures such as input validation, secure coding practices, server-side security configurations, and regular security audits.

By following these mitigation steps, organizations can significantly reduce the risks associated with the absence of CSP headers, enhance

web application security, and protect against common attack vectors such as XSS.

Proof of concept:

Other vulnerabilities were identified during the scan

- Hidden File Found

Proof of concept:

Risk=Medium, Confidence=Low (1)

http://www.clubhouse.com (1)

Hidden File Found (1)

▾ GET http://www.clubhouse.com/.hg

Alert tags	▪ OWASP_2021_A05 ▪ WSTG-v42-CONF-05 ▪ OWASP_2017_A06
Alert description	A sensitive file was identified as accessible or available. This may leak administrative, configuration, or credential information which can be leveraged by a malicious individual to further attack the system or conduct social engineering efforts.
Request	▾ Request line and header section (238 bytes) GET http://www.clubhouse.com/.hg HTTP/1.1 host: www.clubhouse.com user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36 pragma: no-cache cache-control: no-cache ▾ Request body (0 bytes)

Proof of concept:

Response	▾ Status line and header section (565 bytes) HTTP/1.1 301 Moved Permanently Date: Fri, 03 May 2024 20:10:26 GMT Content-Type: text/html Content-Length: 167 Connection: keep-alive Cache-Control: max-age=3600 Expires: Fri, 03 May 2024 21:10:26 GMT Location: https://www.clubhouse.com/.hg Set-Cookie: __cf_bm=xA2Sk5Kaz5pVwJqkFgZ0Y6o_IaEg5tczNTV6DN5itqI- 1714767026-1.0.1.1-04dOCEuY9SlKbcUyInXOTK2tH8wDT.G0eF RD0LMVD8tuqum_DarEqE7zRM8jV8Gh9LwPwoZ8lg84yoMJ72zlow; path=/; expires=Fri, 03-May-24 20:40:26 GMT; domain=.clubhouse.com; HttpOnly; SameSite=None Server: cloudflare CF-RAY: 87e2e4fc984cb2ff-CMB ▾ Response body (167 bytes) <html> <head><title>301 Moved Permanently</title></head> <body> 301 Moved Permanently <hr> cloudflare </body> </html>
Evidence	HTTP/1.1 301 Moved Permanently
Solution	Consider whether or not the component is actually required in production, if it isn't then disable it. If it is then ensure access to it requires appropriate authentication and authorization, or limit exposure to internal systems or specific source IPs, etc.

- Strict-Transport-Security Header Not Set

Proof of concept:

Risk=Low, Confidence=High (1)

https://www.clubhouse.com (1)

Strict-Transport-Security Header Not Set (1)

▾ GET https://www.clubhouse.com/sitemap.xml

Alert tags

- OWASP_2021_A05
- OWASP_2017_A06

Alert description

HTTP Strict Transport Security (HSTS) is a web security policy mechanism whereby a web server declares that complying user agents (such as a web browser) are to interact with it using only secure HTTPS connections (i.e. HTTP layered over TLS/SSL). HSTS is an IETF standards track protocol and is specified in RFC 6797.

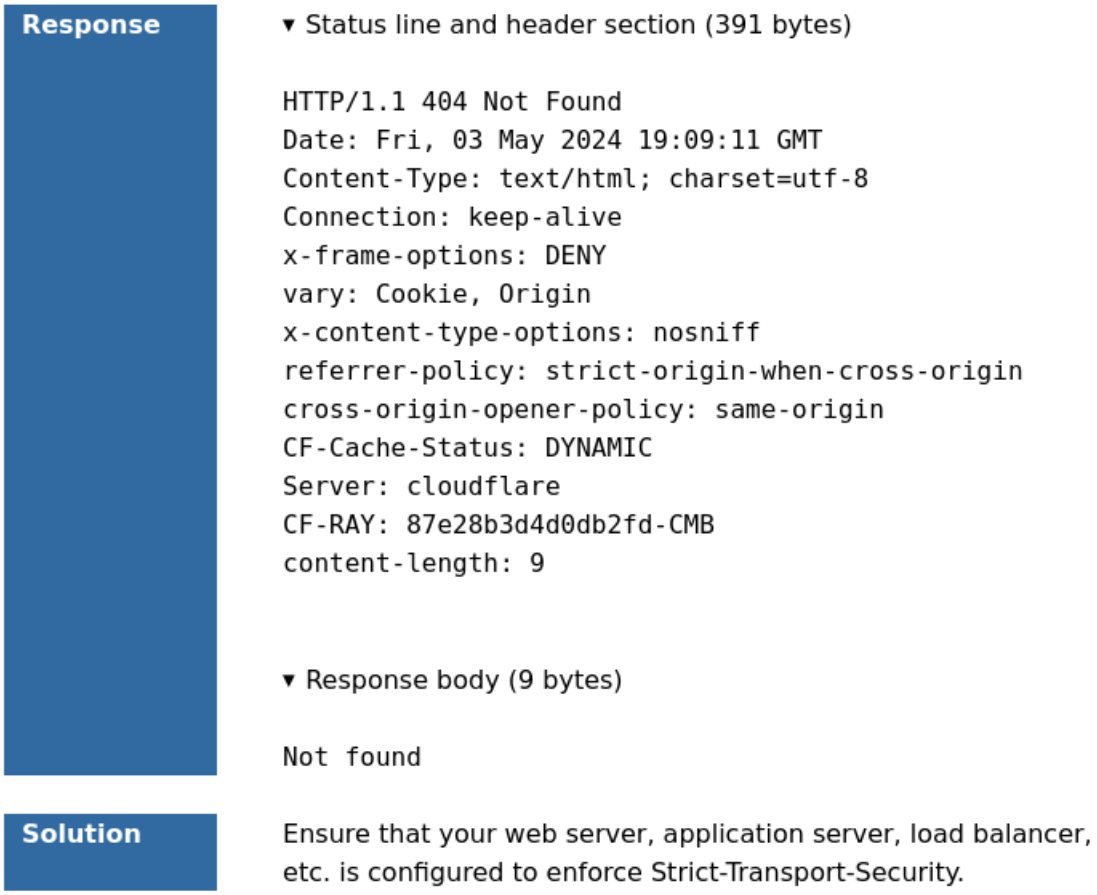
Request

▾ Request line and header section (461 bytes)

```
GET https://www.clubhouse.com/sitemap.xml HTTP/1.1
host: www.clubhouse.com
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64;
x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/116.0.0.0 Safari/537.36
pragma: no-cache
cache-control: no-cache
referer: http://www.clubhouse.com/sitemap.xml
Cookie:
__cf_bm=IVhd4MsWdwrTmtwdYaVSSkj_7f5cdxlYtS9X5TPo4Rg-
1714763350-1.0.1.1-
cWlXYF5FFfw8GVcMOvoYabKBaKXOp1rRaxv.8RLIVLd0rPo5iftM5
Z6EojDBp7zK1ag1nXNZm9lxr0c4WU9N.g
```

▾ Request body (0 bytes)

Proof of concept:



- Cookie with SameSite Attribute None

Proof of concept:

Risk=Low, Confidence=Medium (2)

http://www.clubhouse.com (2)

Cookie with SameSite Attribute None (1)

▾ GET http://www.clubhouse.com/sitemap.xml

Alert tags

- OWASP_2021_A01
- WSTG-v42-SESS-02
- OWASP_2017_A05

Alert description

A cookie has been set with its SameSite attribute set to "none", which means that the cookie can be sent as a result of a 'cross-site' request. The SameSite attribute is an effective counter measure to cross-site request forgery, cross-site script inclusion, and timing attacks.

Request

▾ Request line and header section (246 bytes)

```
GET http://www.clubhouse.com/sitemap.xml HTTP/1.1
host: www.clubhouse.com
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64;
x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/116.0.0.0 Safari/537.36
pragma: no-cache
cache-control: no-cache
```

▾ Request body (0 bytes)

Proof of concept:

Response	▾ Status line and header section (573 bytes) HTTP/1.1 301 Moved Permanently Date: Fri, 03 May 2024 19:09:10 GMT Content-Type: text/html Content-Length: 167 Connection: keep-alive Cache-Control: max-age=3600 Expires: Fri, 03 May 2024 20:09:10 GMT Location: https://www.clubhouse.com/sitemap.xml Set-Cookie: __cf_bm=IVhd4MsWdwrTmtwdYaVSSkj_7f5cdxlYtS9X5TPo4Rg-1714763350-1.0.1.1-cWlXYF5FFfw8GVcMOvoYabKBaKXOp1rRaxv.8RLIVLd0rPo5iftM5Z6EojDBp7zK1ag1nXNZm9lxr0c4WU9N.g; path=/; expires=Fri, 03-May-24 19:39:10 GMT; domain=.clubhouse.com; HttpOnly; SameSite=None Server: cloudflare CF-RAY: 87e28b3c4c65b300-CMB ▾ Response body (167 bytes) <html> <head><title>301 Moved Permanently</title></head> <body> 301 Moved Permanently <hr> cloudflare </body> </html>
Parameter	__cf_bm
Evidence	Set-Cookie: __cf_bm
Solution	Ensure that the SameSite attribute is set to either 'lax' or ideally 'strict' for all cookies.

- **Cross-Domain JavaScript Source File Inclusion**

Proof of concept:

▾ GET http://www.clubhouse.com

Alert tags	▪ OWASP_2021_A08
Alert description	The page includes one or more script files from a third-party domain.
Request	▾ Request line and header section (234 bytes) GET http://www.clubhouse.com HTTP/1.1 host: www.clubhouse.com user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36 pragma: no-cache cache-control: no-cache ▾ Request body (0 bytes)

Proof of concept:

Response	▾ Status line and header section (529 bytes) HTTP/1.1 200 OK Date: Fri, 03 May 2024 19:09:10 GMT Content-Type: text/html; charset=utf-8 Connection: keep-alive Cache-Control: public, max-age=14400 x-frame-options: DENY vary: Cookie, Accept-Encoding, Origin x-content-type-options: nosniff referrer-policy: strict-origin-when-cross-origin cross-origin-opener-policy: same-origin Last-Modified: Fri, 03 May 2024 19:02:35 GMT CF-Cache-Status: EXPIRED Expires: Fri, 03 May 2024 23:09:10 GMT Server: cloudflare CF-RAY: 87e28b342ca2b2f9-CMB content-length: 23071 ▸ Response body (23071 bytes)
Parameter	https://static-assets.clubhouseapi.com/static/dotcom/bundle.9d53a80a5011.js
Evidence	<script src="https://static-assets.clubhouseapi.com/static/dotcom/bundle.9d53a80a5011.js"></script>
Solution	Ensure JavaScript source files are loaded from only trusted sources, and the sources can't be controlled by end users of the application.

Timestamp Disclosure – Unix

Proof of concept:

Risk=Low, Confidence=Low (1)

http://www.clubhouse.com (1)

Timestamp Disclosure - Unix (1)

▾ GET http://www.clubhouse.com

Alert tags	▪ OWASP_2021_A01 ▪ OWASP_2017_A03
Alert description	A timestamp was disclosed by the application/web server - Unix
Other info	1714763350, which evaluates to: 2024-05-04 00:39:10
Request	▾ Request line and header section (234 bytes) GET http://www.clubhouse.com HTTP/1.1 host: www.clubhouse.com user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36 pragma: no-cache cache-control: no-cache ▾ Request body (0 bytes)

Proof of concept:

Response

▾ Status line and header section (562 bytes)

```
HTTP/1.1 301 Moved Permanently
Date: Fri, 03 May 2024 19:09:10 GMT
Content-Type: text/html
Content-Length: 167
Connection: keep-alive
Cache-Control: max-age=3600
Expires: Fri, 03 May 2024 20:09:10 GMT
Location: https://www.clubhouse.com/
Set-Cookie:
__cf_bm=Nv3aRZvnjuITroHjsXcYtp6iGySiTodwYd_N2QQsbRY-
1714763350-1.0.1.1-
trUmoiTVhvdUqAJpXALKmo_wnfxTZFqVcPON5STmdQeh6wTkPO7Pn
5P0FuMNBiCsmTbUXQpBx_hQlkR_GuZ.yw; path=/;
expires=Fri, 03-May-24 19:39:10 GMT;
domain=.clubhouse.com; HttpOnly; SameSite=None
Server: cloudflare
CF-RAY: 87e28b3c1e13a13e-CMB
```

▾ Response body (167 bytes)

```
<html>
<head><title>301 Moved Permanently</title></head>
<body>
<center><h1>301 Moved Permanently</h1></center>
<hr><center>cloudflare</center>
</body>
</html>
```

Evidence

1714763350

Solution

Manually confirm that the timestamp data is not sensitive, and that the data cannot be aggregated to disclose exploitable patterns.

Conclusion

Important security headers like X-XSS-Protection are missing, weak ciphers and probable phishing risks were among the key

vulnerabilities found in the security assessment of "http://www.clubhouse.com" that was completed throughout multiple phases. These flaws highlight the necessity for strong security measures by putting the platform at risk for data breaches, illegal access, and manipulation.

The evaluation also identified security holes that fell under the OWASP Top 10 categories, including Injection bugs, Cryptographic Failures, and Broken Access Control. It is advised to employ mitigation techniques such as enforcing access controls, upgrading cryptographic setups, implementing Content Security Policy (CSP) headers, and utilizing secure coding techniques.

In summary, protecting sensitive data, upholding user confidence, and reducing security risks for the Clubhouse platform depend on resolving these vulnerabilities through proactive security measures, ongoing monitoring, and adherence to best practices.