



Web Security – IE2062

2. Tinder Bug Bounty

*A D A Ihansa*

*IT22899606*

# **Web Audit**

## ***tinder.com***

## Contents

Introduction to Bug Bounty program and audit scope.....	4
Information gathering phase.....	5
Finding active subdomains and their states .....	5
Sublist3r .....	5
HTTPProbe .....	8
Netcraft.....	9
Spiderfoot.....	10
Google Dorks .....	12
Directory and services enumeration.....	15
Dirbuster.....	15
Gobuster .....	16
Nmap .....	17
Automated Testing .....	18
OWASP ZAP .....	18
Conclusion.....	28
References .....	28

## Introduction to Bug Bounty program and audit scope

Tinder is a global online dating platform that allows users to meet new people, expand their social network, and connect with locals in over 190 countries. Launched in 2012, it has become one of the world's most popular dating apps. The app is known for its unique "swipe right" to like and "swipe left" to dislike feature. It uses a "double opt-in" system, where two users must mutually like each other's profiles before they can exchange messages. As of 2022, Tinder had 10.9 million subscribers and 75 million monthly active users.

In hackerone bug bounty program, they defined these subdomains (and all inclusive) as valid subdomains for testing.

\*.tinder.com























com.tinder

\*.tstaging.com

\*.tstaging.tools

\*.tinderops.net

Eligible in-scope subdomains for bug bounty program are mentioned below and they mention that any subdomain under **tinder.com** is in scope,

Asset name 	Type 	Coverage 	Max. severity 	Bounty 	Last update 
547702041	iOS: App Store	In scope	 Critical	 Eligible	Sep 27, 2021
*.tinderops.net	Wildcard	In scope	 Critical	 Eligible	May 15, 2023
*.gotinder.com	Wildcard	In scope	 Critical	 Eligible	Mar 6, 2024
*.tinder.com	Wildcard	In scope	 Critical	 Eligible	May 15, 2023
com.tinder	Android: Play Store	In scope	 Critical	 Eligible	Sep 27, 2021
*.tstaging.com	Wildcard	In scope	 Medium	 Eligible	May 15, 2023
*.tstaging.tools	Wildcard	In scope	 Medium	 Eligible	May 15, 2023
*.tinderwebstaging.com	Wildcard	In scope	 Medium	 Eligible	May 15, 2023

## Information gathering phase.

The information gathering phase, also known as reconnaissance or recon, is critical as it involves gathering information about the target to understand its nature and behavior. This phase is indispensable during audits or attacks because the more insights we gain into the target's behavior, the easier it becomes to identify potential vulnerabilities that could be exploited.

There exist two primary types of information gathering scan methods:

1. Active Scanning: This method generates significant activity on the target system, often resulting in the retrieval of extensive information.

2. Passive Scanning: Unlike active scanning, this approach minimizes disturbance on the target system, though it typically yields fewer comprehensive results compared to active scanning.

In bug bounty programs, where details about the underlying architecture of systems are usually not disclosed (referred to as black box pentesting), specific tools and techniques become crucial for gathering insights into their services, devices, and exposed information. This allows testers to gain a better understanding of the systems they are assessing.

## Finding active subdomains and their states

### Sublist3r

Sublist3r, a Python tool, is specifically designed to reveal subdomains linked to a specified target website. Utilizing search engines and diverse online services, it systematically scours the web for available subdomains associated with the designated target domain. Given the opportunity to explore any subdomain within reddit.com, it is recommended to identify additional subdomains for testing objectives.

To install Sublist3r, navigate to its GitHub repository at <https://github.com/aboul3la/Sublist3r.git>. This repository hosts all the necessary files required for installing the tool. Execute the following command in your shell to download it:

...

***git clone https://github.com/aboul3la/Sublist3r.git***

...

Please note that Sublist3r necessitates either Python 2.7 or Python 3.4 to operate smoothly.

After downloading the files, go inside the 'Sublist3r' directory and install the requirements by entering,

***sudo pip install -r requirements.txt***

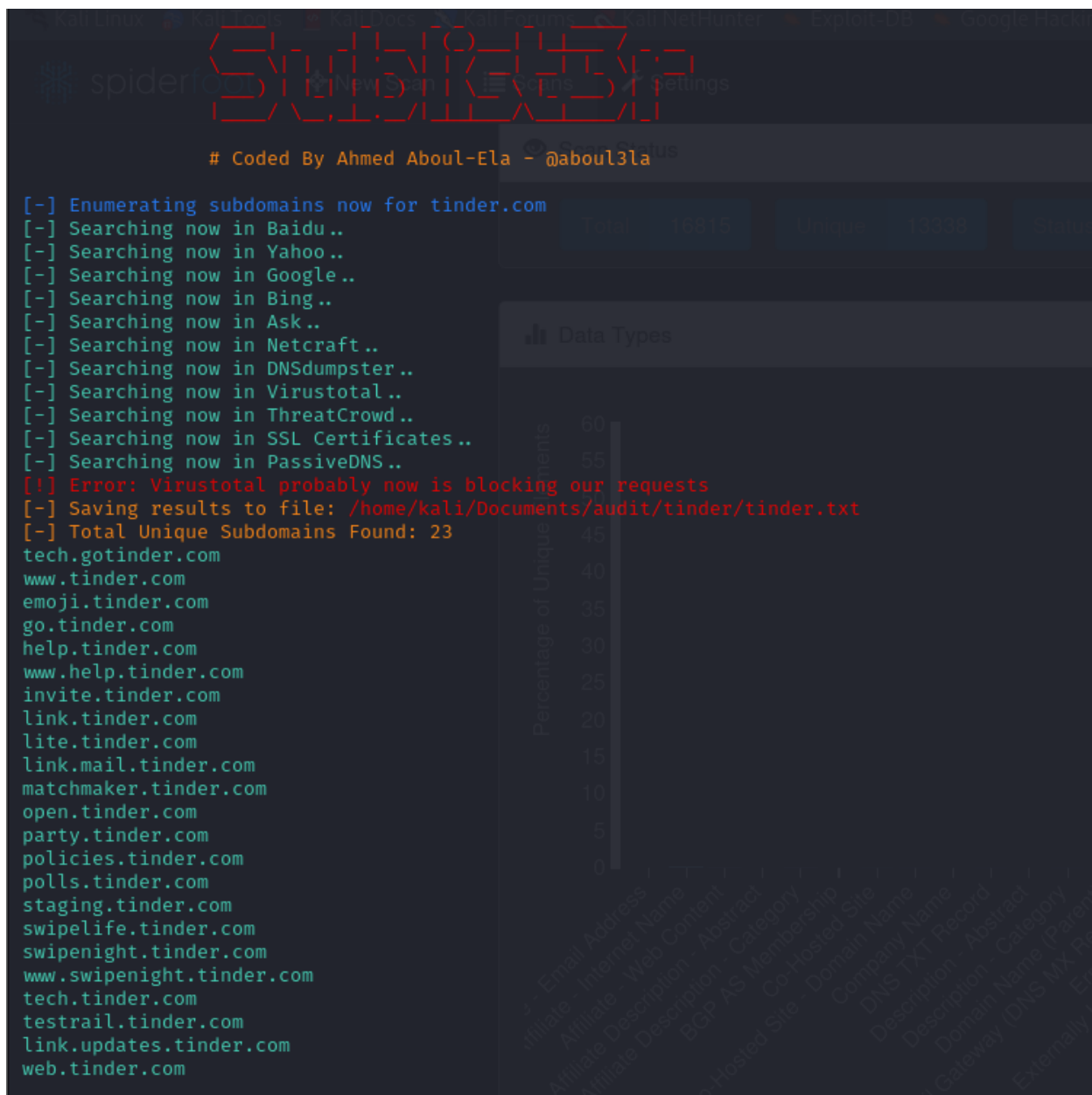
After installing the requirements, enter

`python3 sublist3r.py -d <domain_name>`

to find subdomains under the mentioned domain.

*\*In some Linux distributions, there will be an error saying that "[!] Error: Virustotal probably now is blocking our requests". To avoid this you will need to get the API key from VirusTotal by creating an account. After the API key has been obtained, export it to an environment variable using, export VT\_APIKEY=<API key>. This will work most of the time, but this is not a must.*

Since I need to check the subdomains after, I am writing the results to a file using -o switch.



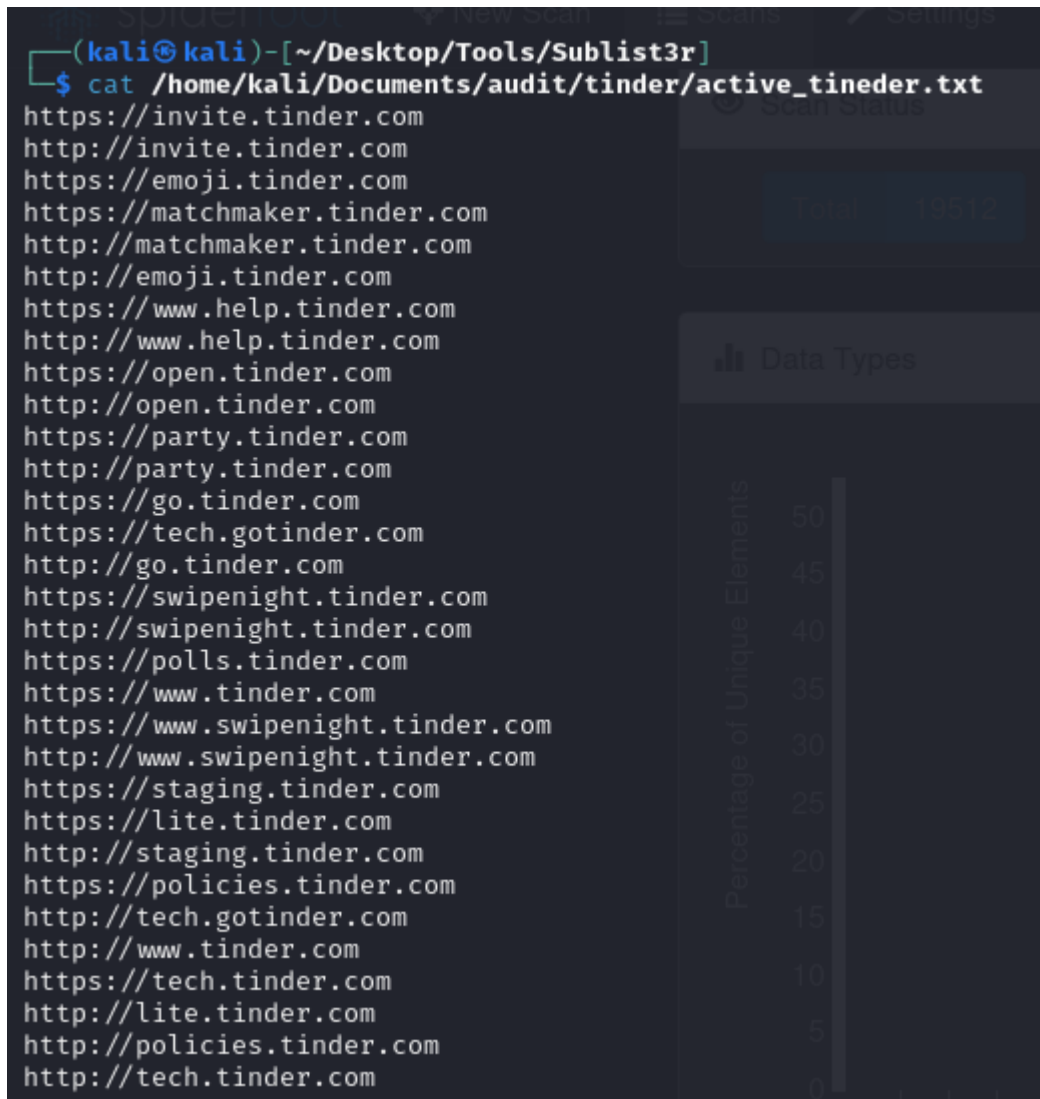
Upon examining for accessible subdomains, the next step involves identifying those that are operational. This can be accomplished by employing an additional tool known as 'httprobe'.

## HTTPProbe

This tool can identify active domains that are operational. To discover active subdomains under this site, I'm utilizing the text file previously generated by Sublist3r and writing the active subdomains to a new file.

```
(kali㉿kali)-[~/Desktop/Tools/Sublist3r]
$ httpprobe < /home/kali/Documents/audit/tinder/tinder.txt > /home/kali/Documents/audit/tinder/active_tinder.txt
```

Following the completion of the scan, the findings reveal that the majority of the subdomains are indeed active.




```
(kali㉿kali)-[~/Desktop/Tools/Sublist3r]
$ cat /home/kali/Documents/audit/tinder/active_tinder.txt
https://invite.tinder.com
http://invite.tinder.com
https://emoji.tinder.com
https://matchmaker.tinder.com
http://matchmaker.tinder.com
http://emoji.tinder.com
https://www.help.tinder.com
http://www.help.tinder.com
https://open.tinder.com
http://open.tinder.com
https://party.tinder.com
http://party.tinder.com
https://go.tinder.com
https://tech.gotinder.com
http://go.tinder.com
https://swipenight.tinder.com
http://swipenight.tinder.com
https://polls.tinder.com
https://www.tinder.com
https://www.swipenight.tinder.com
http://www.swipenight.tinder.com
https://staging.tinder.com
https://lite.tinder.com
http://staging.tinder.com
https://policies.tinder.com
http://tech.gotinder.com
http://www.tinder.com
https://tech.tinder.com
http://lite.tinder.com
http://policies.tinder.com
http://tech.tinder.com
```



## Netcraft

Netcraft, an internet services firm, offers online security solutions. Their services encompass automated vulnerability scanning and application security. No downloads or intricate setups are necessary as these services are accessible online.

By utilizing Netcraft search feature on their website, users can retrieve various details including site rank, IP address, SSL/TLS versions in use, hosting country, and hosting company.

[LEARN MORE](#)[REPORT FRAUD](#)

---

### Site report for <http://www.tinder.com>

[Look up another site?](#)

Share: [G+](#) [Twitter](#) [Facebook](#) [LinkedIn](#) [YouTube](#)

#### Background

Site title	Not Present	Date first seen	December 1997
Site rank	669688	Primary language	English
Description	Not Present		

#### Network

Site	<a href="http://www.tinder.com">http://www.tinder.com</a>	Domain	<a href="http://tinder.com">tinder.com</a>
Netblock Owner	<a href="#">Amazon.com, Inc.</a>	Nameserver	<a href="#">ns-1483.awsdns-57.org</a>
Hosting company	Amazon	Domain registrar	<a href="#">markmonitor.com</a>
Hosting country	<a href="#">US</a>	Nameserver organisation	<a href="#">whois.pir.org</a>
IPv4 address	<a href="#">52.84.150.39</a> ( <a href="#">VirusTotal ID</a> )	Organisation	Match Group, LLC, United States
IPv4 autonomous systems	<a href="#">AS16509</a>	DNS admin	<a href="mailto:awsdns-hostmaster@amazon.com">awsdns-hostmaster@amazon.com</a>

For full site report: [Site report for http://www.tinder.com | Netcraft](#)

This data is publicly accessible, and some details regarding the system and its technology can be uncovered through available sources.

Since, many users utilize this website, and considering that the IP addresses match those of my specified targets, I only obtain a Netcraft report for this domain. However, reports for the other mentioned subdomains can also be retrieved from Netcraft.

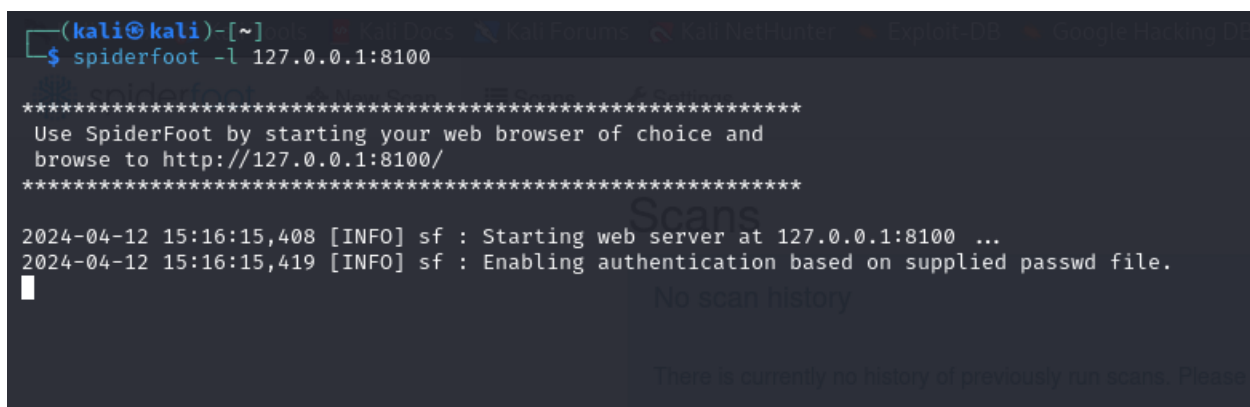
## Spiderfoot

SpiderFoot is an open-source intelligence (OSINT) automation tool that is designed to simplify the process of gathering and analyzing data. It integrates with a wide range of data sources and provides an intuitive web-based interface or a command-line option. SpiderFoot is equipped with over 200 modules for various data analysis tasks, including host/sub-domain/TLD enumeration/extraction, email address, phone number and human name extraction, and much more. It also offers export options in CSV, JSON, and GEXF formats, and integrates with the TOR network for dark web searches. SpiderFoot is a powerful tool for both offensive and defensive reconnaissance, making it an asset in the field of cybersecurity.

### Using spiderfoot

It must be setup, before using this tool.

Spiderfoot -l 127.0.0.1:8100

A terminal window on a Kali Linux system. The prompt is (kali@kali)-[~]. The command \$ spiderfoot -l 127.0.0.1:8100 has been entered. The output shows a series of asterisks, followed by instructions to use SpiderFoot via a web browser at http://127.0.0.1:8100/. Another series of asterisks follows. Then, two log entries are shown: '2024-04-12 15:16:15,408 [INFO] sf : Starting web server at 127.0.0.1:8100 ...' and '2024-04-12 15:16:15,419 [INFO] sf : Enabling authentication based on supplied passwd file.' A cursor is visible on the line following the second log entry. In the background, a faint 'Scans' window is visible with the text 'No scan history' and 'There is currently no history of previously run scans. Please'.

To utilize the Spiderfoot tool, which is hosted on localhost (127.0.0.1) at port 8100, just launch a web browser and enter `http://127.0.0.1:8100` in the address bar.

After the scanner loads, proceed to "New scan" and tailor your scan type according to the scope of your investigation. There are various modules at your disposal that can be activated or deactivated based on your permissions. Since you're engaging in a passive information gathering phase, opt for the 'footprint' option to crawl and collect information about the website.

## Spiderfoot results

spiderfoot New Scan Scans Settings Dark Mode About

### New Scan

Scan Name:

Scan Target:

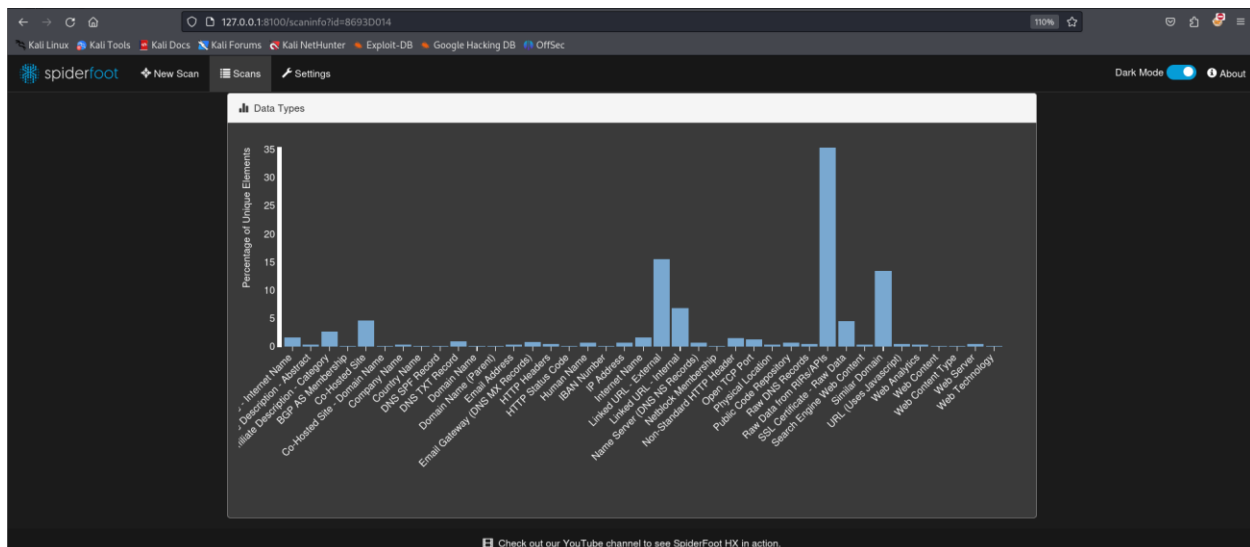
Your scan target may be one of the following. SpiderFoot will automatically detect the target type based on the format of your input:

- Domain Name: e.g. example.com
- IPV4 Address: e.g. 1.2.3.4
- IPV6 Address: e.g. 2001:4700:4700:1111
- Hostname/Sub-domain: e.g. abc.example.com
- Subnet: e.g. 1.2.3.0/24
- Bitcoin Address: e.g. 1HesYJSP1QgyPEjPQ9vzBL1wjuuNGe7R
- E-mail address: e.g. bob@example.com
- Phone Number: e.g. +12345678901 (E.164 format)
- Human Name: e.g. "John Smith" (must be in quotes)
- Username: e.g. "jsmith2002" (must be in quotes)
- Network ASN: e.g. 1234

By Use Case: **By Required Data** By Module

- ☐ All: Get anything and everything about the target.  
All SpiderFoot modules will be enabled (slow) but every possible piece of information about the target will be obtained and analysed.
- ☒ Footprint: Understand what information this target exposes to the Internet.  
Gain an understanding about the target's network perimeter, associated identities and other information that is obtained through a lot of web crawling and search engine use.
- ☐ Investigate: Best for when you suspect the target to be malicious but need more information.  
Some basic footprinting will be performed in addition to querying of blacklists and other sources that may have information about your target's maliciousness.
- ☐ Proactive: When you don't want the target in even essence they are being investigated

Want more OSINT automation capabilities? Check out SpiderFoot HX.

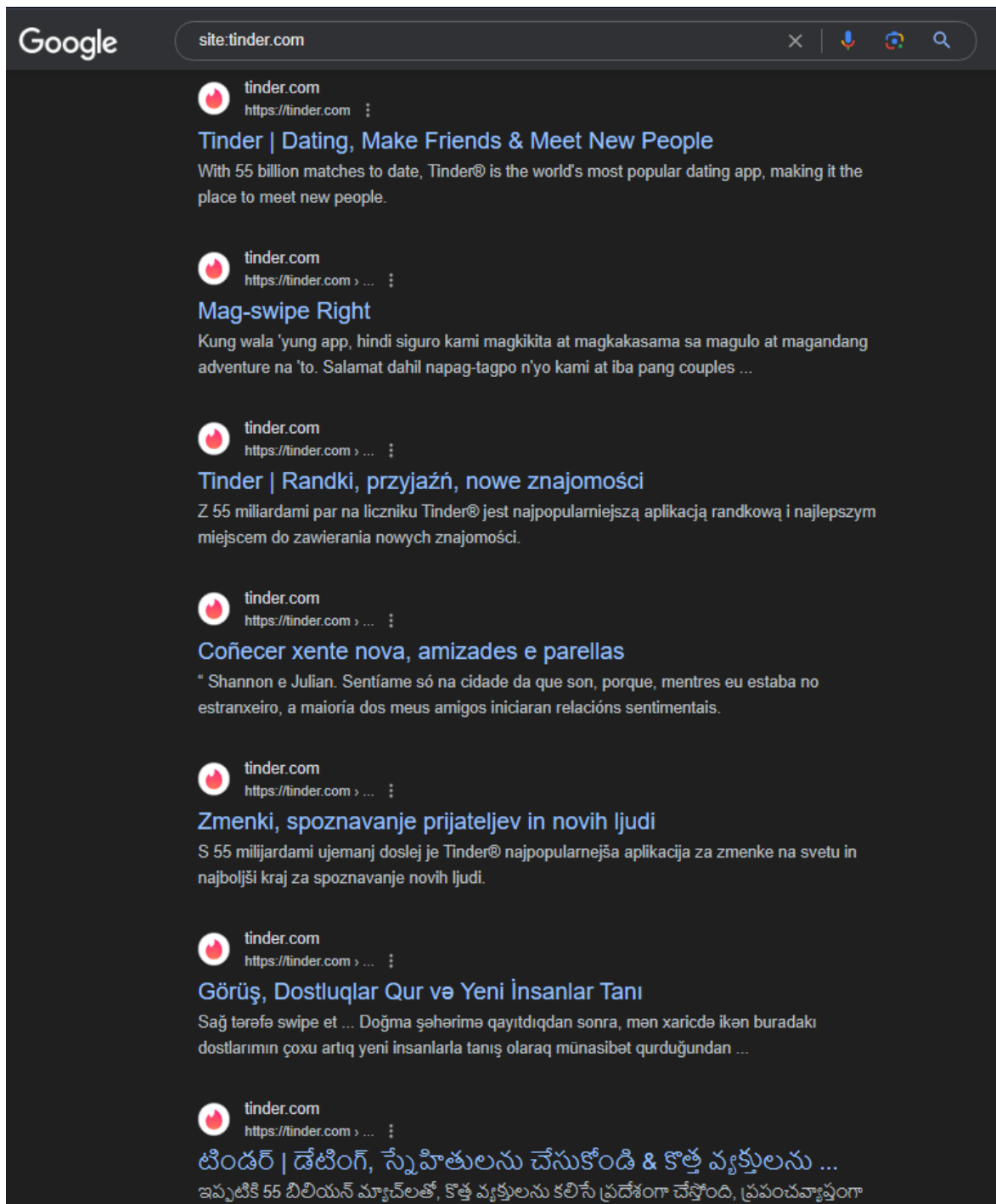


The scan has produced noteworthy findings, such as usernames, SSL certificates, and physical addresses. A significant portion of this data seems to be publicly accessible information and links leading to external websites. However, it's essential to highlight those usernames, especially when coupled with their corresponding email addresses, could potentially become avenues for social engineering or spear phishing attacks. Nevertheless, it's important to acknowledge that addressing such concerns lies beyond the boundaries of this assessment.

## Google Dorks

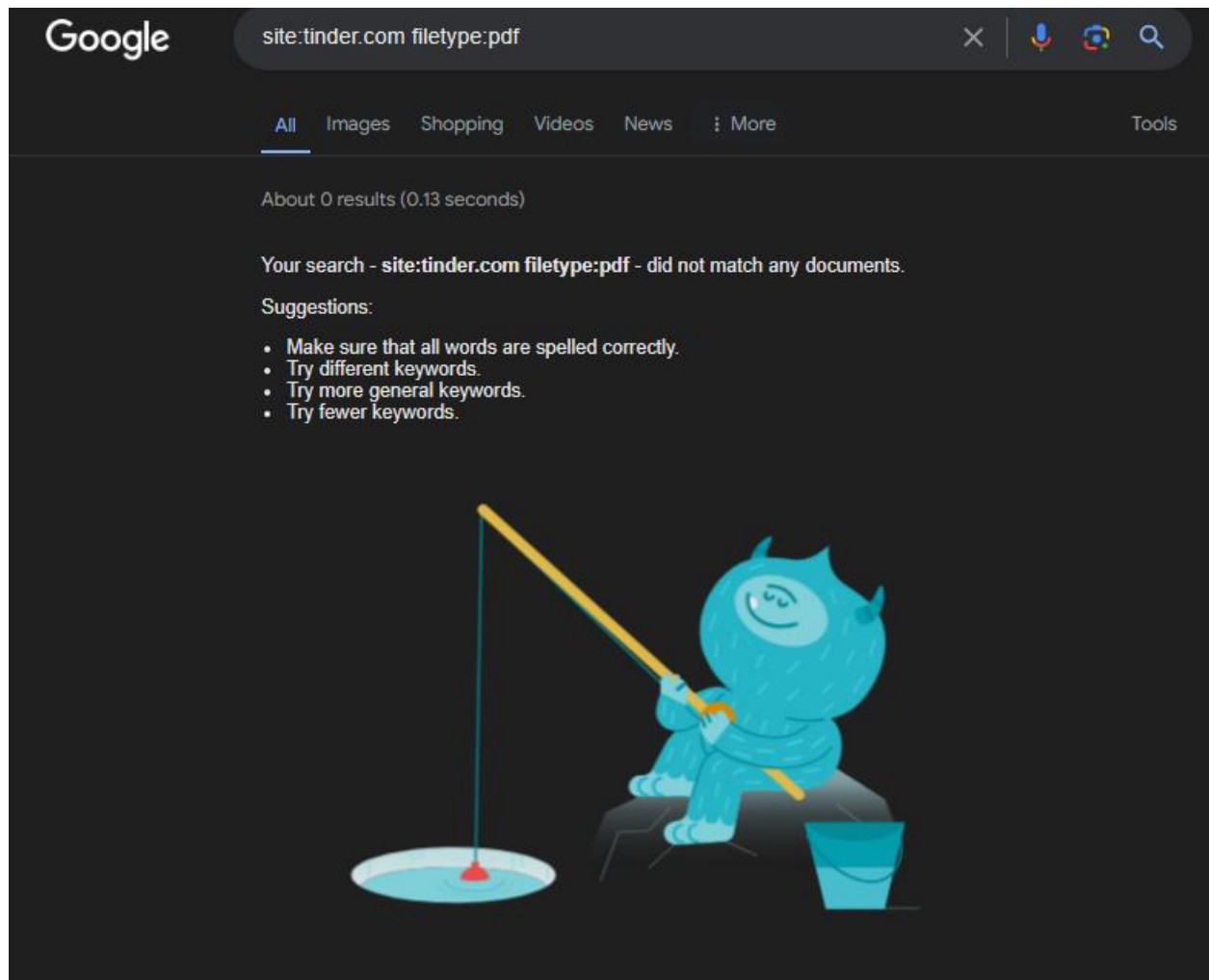
Google Dorks, also known as Google Hacking or Google Dorking, are specialized search queries that leverage Google's powerful search engine to unearth specific information and vulnerabilities that might not be accessible through standard searches. These are advanced search queries that use special operators to find specific information in Google's databases. They can be used to uncover hidden data or vulnerabilities on websites. By employing these dorks, you can focus on specific search results, unveiling hidden gems that ordinary searches might miss. They are valuable for security research but also pose risks if used maliciously. For example, they can reveal sensitive or private information about websites and the companies, organizations, and individuals that own and operate them. Google Dorks are a powerful tool for information gathering and vulnerability scanning, but they should be used responsibly to avoid unintended consequences.

**site:tinder.com** operator searches for websites that has "**tinder.com**" in their domains.



Certainly, the appearance of subdomains in search results illustrates a common utilization of Google Dorking. This method facilitates the discovery of different file types and pages that are exposed by a specific website on the internet, whether it's deliberate or unintentional. Through the refinement of search queries, users can unearth information and potentially pinpoint vulnerabilities or confidential data that might be accessible online.

During a search for various file types exposed on the internet, I came across some intriguing and possibly concerning PDFs within this subdomain.



The management of information and files within this subdomain appears to be efficient, as no additional significant files were uncovered apart from the previously mentioned PDFs.

## Directory and services enumeration

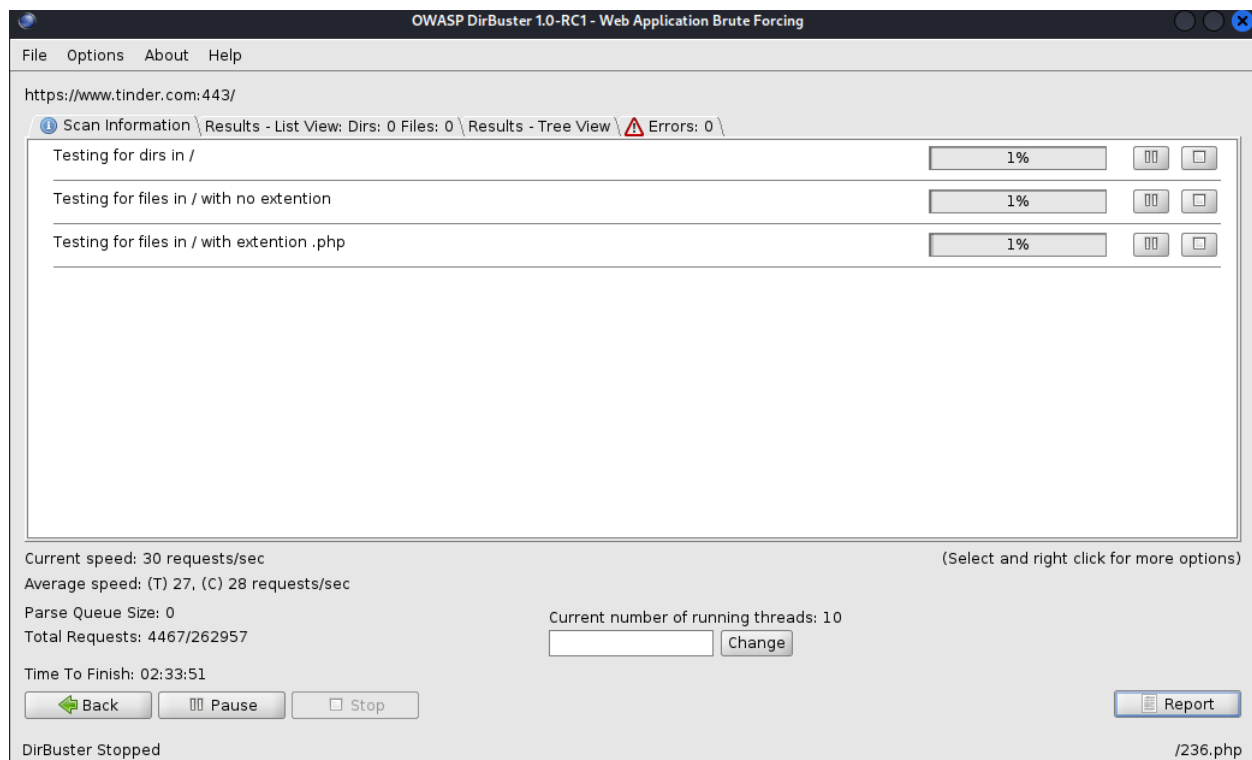
### Dirbuster

DirBuster, a web content scanner developed by OWASP, utilizes brute force methods to uncover different directories within a target website. By scrutinizing HTTP responses and their associated response codes, the tool detects concealed or referenced directories. Built in Java, DirBuster supports multi-threading to expedite directory scanning and produce a comprehensive file and folder structure of the target site.

Employing this tool facilitates the identification of directories or files that might be accessible yet not overtly exposed. Furthermore, it offers a glimpse into the server's file and folder arrangement, assisting in comprehending its structure and potential vulnerabilities.

Domain: [www.tinder.com](https://www.tinder.com)

After scanning for a while, dirbuster gave some errors and stopped working. Further looking into the issue, it seemed like the dirbuster cannot access this domain.



## Gobuster

Enumerating hidden directories and files is a prevalent method of attacking an application. This approach can yield valuable information for executing specific attacks. Although numerous tools are available for this purpose, their effectiveness varies. Gobuster, a command-line tool implemented in Go, stands out among them. Go is a programming language known for its fast-processing capabilities, excellent concurrency support, and speed. However, Gobuster lacks the feature of recursive directory exploration. Typically, this isn't a significant issue as most scanners can compensate for this limitation.

As Dibuster failed to enumerate, I tried with gobuster for dir enumeration. But still gobuster also failed to enumerate hidden directories.

```
(kali㉿kali)-[~]
$ gobuster dir -u https://tiktok.com -t 50 -w /usr/share/dirb/wordlists/common.txt -x .php,.html -b 200,301

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url:             https://tiktok.com
[+] Method:          GET
[+] Threads:         50
[+] Wordlist:         /usr/share/dirb/wordlists/common.txt
[+] Negative Status codes: 200,301
[+] User Agent:      gobuster/3.6
[+] Extensions:     php,html
[+] Timeout:         10s

Starting gobuster in directory enumeration mode

Progress: 13842 / 13845 (99.98%)

Finished
```



## Nmap

Nmap, also known as Network Mapper, is a flexible port scanner engineered to probe hosts and reveal their open ports, associated services, port statuses, running service versions, and the operating system in use, among other details. This open-source tool serves various purposes, offering valuable insights into network configurations and potential vulnerabilities. Additionally, Nmap supports the execution of scripts on target systems to exploit vulnerabilities or gather additional information.

Upon installation, you can access the available options by typing "nmap -h" in your command line interface. For a more detailed understanding of how the tool operates, you can consult the manual page by entering "man nmap" in your command line interface. \*Note that some options may require administrator / super user privileges.

\*Note that some options may require administrator / super user privileges.

I am using the following scan options for this assessment.

*sudo nmap <host name> -sS -sV -O -oN <filename>*

-sS: Enables SYN scan (also known as Stealth scan).

-sV: Enables version detection. It tries to detect the version of the service running in that port.

-O: Enables Operating System detection.

-oN : Outputs the scan results to text file

Scanned results for <https://www.grammarly.com/>

```
(kali@kali)~$ sudo nmap tinder.com -sS -sV -O -oN tinder.txt
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-21 07:38 EDT
Nmap scan report for tinder.com (52.84.150.54)
Host is up (0.018s latency).
Other addresses for tinder.com (not scanned): 52.84.150.39 52.84.150.60 52.84.150.55
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
25/tcp    open  smtp?
80/tcp    open  http    Amazon CloudFront httpd
443/tcp   open  ssl/http Amazon CloudFront httpd
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port25-TCP:V=7.94SVN|I=7|D=4/21|Time=6624FAD9|P=x86_64-pc-linux-gnu|(H
SF:ello,2A,"552|x20Invalid|x20domain|x20name|x20in|x20EHLO|x20command|.r\
SF:n")|(GenericLines,28,"500|x20Syntax|x20error,\x20command\x20unrecogniz
SF:ed\r\n")|(GetRequest,28,"500|x20Syntax|x20error,\x20command\x20unrecog
SF:nized\r\n")|(HTTPOptions,28,"500|x20Syntax|x20error,\x20command\x20unr
SF:ecognized\r\n")|(RTSPRequest,28,"500|x20Syntax|x20error,\x20command\x2
SF:0unrecognized\r\n");
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
OS fingerprint not ideal because: Missing a closed TCP port so results incomplete
No OS matches for host

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 43.81 seconds
```

## Automated Testing

For automated testing, I've selected OWASP ZAP widely used tool within the industry.

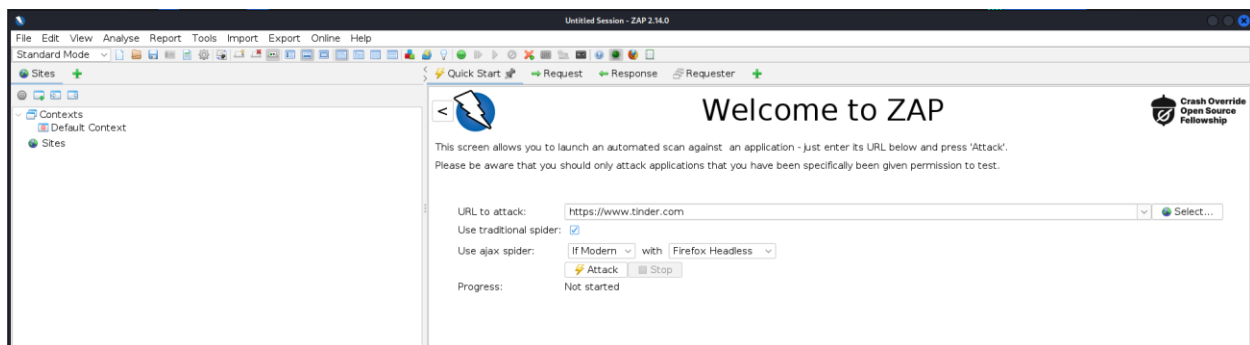
### OWASP ZAP

The Open Web Application Security Project Zed Attack Proxy (OWASP ZAP) is an open-source vulnerability scanner renowned for its capability to function as a Man-in-the-Middle (MITM) proxy. It assesses various vulnerabilities by scrutinizing responses from the web application or server. Notably convenient to utilize, OWASP ZAP offers customization options through the installation of modules, enabling efficient management of results.

Within this proxy, there are primarily two scan types available:

1. Automated Scan: Users input the target URL and initiate the attack. The behavior can be tailored by selecting the ZAP mode. This triggers all scripts against the target to detect vulnerabilities and generates reports accordingly.
2. Manual Explore: Users can navigate to the target web application and commence exploration. During manual exploration, ZAP HUD (Heads Up Display) captures each page, while the ZAP proxy records responses.

For this assessment, I am running ZAP on automated mode.



After specifying the target URL in the designated textbox, simply select "Attack" to initiate the scanning process. Upon completion, a comprehensive report of the findings can be generated by

selecting "Report." Below are screenshots showcasing the results obtained after scanning several domains.

#### Alert counts by risk and confidence

This table shows the number of alerts for each level of risk and confidence included in the report.

(The percentages in brackets represent the count as a percentage of the total number of alerts included in the report, rounded to one decimal place.)

		Confidence				
		User Confirmed	High	Medium	Low	Total
Risk	High	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
	Medium	0 (0.0%)	4 (21.1%)	1 (5.3%)	0 (0.0%)	5 (26.3%)
	Low	0 (0.0%)	2 (10.5%)	6 (31.6%)	1 (5.3%)	9 (47.4%)
	Informational	0 (0.0%)	0 (0.0%)	3 (15.8%)	2 (10.5%)	5 (26.3%)
	Total	0 (0.0%)	6 (31.6%)	10 (52.6%)	3 (15.8%)	19 (100%)

### Alert counts by alert type

This table shows the number of alerts of each alert type, together with the alert type's risk level.

(The percentages in brackets represent each count as a percentage, rounded to one decimal place, of the total number of alerts included in this report.)

Alert type	Risk	Count
<a href="#">CSP: Wildcard Directive</a>	Medium	176 (926.3%)
<a href="#">CSP: script-src unsafe-eval</a>	Medium	176 (926.3%)
<a href="#">CSP: script-src unsafe-inline</a>	Medium	176 (926.3%)
<a href="#">CSP: style-src unsafe-inline</a>	Medium	176 (926.3%)
<a href="#">Missing Anti-clickjacking Header</a>	Medium	176 (926.3%)
<a href="#">Cookie No HttpOnly Flag</a>	Low	408 (2,147.4%)
<a href="#">Cookie Without Secure Flag</a>	Low	204 (1,073.7%)
<a href="#">Cookie with SameSite Attribute None</a>	Low	204 (1,073.7%)
<a href="#">Cookie without SameSite Attribute</a>	Low	204 (1,073.7%)

<a href="#">Timestamp Disclosure - Unix</a>	Low	115 (605.3%)
<a href="#">X-Content-Type-Options Header Missing</a>	Low	228 (1,200.0%)
<a href="#">Information Disclosure - Suspicious Comments</a>	Informational	290 (1,526.3%)
<a href="#">Modern Web Application</a>	Informational	62 (326.3%)
<a href="#">Re-examine Cache-control Directives</a>	Informational	207 (1,089.5%)
<a href="#">Retrieved from Cache</a>	Informational	3 (15.8%)
<a href="#">Session Management Response Identified</a>	Informational	203 (1,068.4%)
Total		19

\*Note that these vulnerabilities are rated according to the OWASP risk rating methodology which can be found in this link. [OWASP Risk Rating Methodology](#).

Here are the vulnerabilities detected within this domain, along with their impacts. The insights provided, including screenshots and remedies, are sourced from the report generated by OWASP ZAP and the Netsparker website. [ <https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities> ])

<https://tinder.com> (1)

### Missing Anti-clickjacking Header (1)

▼ GET <https://tinder.com>

#### Alert tags

- [OWASP\\_2021\\_A05](#)
- [WSTG-v42-CLNT-09](#)
- [OWASP\\_2017\\_A06](#)

#### Alert description

The response does not include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options to protect against 'Clickjacking' attacks.

#### Request

- ▶ Request line and header section (221 bytes)
- ▶ Request body (0 bytes)

#### Response

- ▶ Status line and header section (1240 bytes)
- ▶ Response body (394477 bytes)

#### Parameter

x-frame-options

#### Solution

Modern Web browsers support the Content-Security-Policy and X-Frame-Options HTTP headers. Ensure one of them is set on all web pages returned by your site/app.

If you expect the page to be framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise if you never expect the page to be framed, you should use DENY. Alternatively consider implementing Content Security Policy's "frame-ancestors" directive.

## Strict-Transport-Security Header Not Set (1)

▼ GET https://tinder.com/robots.txt

### 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 (232 bytes)
- ▼ Request body (0 bytes)

### Response

- Status line and header section (599 bytes)
- ▼ Response body (111 bytes)

User-agent: \*  
Disallow: /sparks  
Disallow: /api  
Disallow: /healthcheck

Sitemap: https://tinder.com/sitemap.xml

### Solution

Ensure that your web server, application server, load balancer, etc. is configured to enforce Strict-Transport-Security.

<https://tinder.com> (6)

### Cookie No HttpOnly Flag (1)

▼ GET <https://tinder.com/healthcheck>

#### Alert tags

- [OWASP\\_2021\\_A05](#)
- [WSTG-v42-SESS-02](#)
- [OWASP\\_2017\\_A06](#)

#### Alert description

A cookie has been set without the HttpOnly flag, which means that the cookie can be accessed by JavaScript. If a malicious script can be run on this page then the cookie will be accessible and can be transmitted to another site. If this is a session cookie then session hijacking may be possible.

#### Request

- Request line and header section (551 bytes)
- Request body (0 bytes)

#### Response

- Status line and header section (921 bytes)
  - ▼ Response body (2 bytes)
- OK

#### Parameter

AWSALB

#### Evidence

Set-Cookie: AWSALB

#### Solution

Ensure that the HttpOnly flag is set for all cookies.



### Session Management Response Identified (1)

▼ GET https://tinder.com/healthcheck

Alert tags	
Alert description	The given response has been identified as containing a session management token. The 'Other Info' field contains a set of header tokens that can be used in the Header Based Session Management Method. If the request is in a context which has a Session Management Method set to "Auto-Detect" then this rule will change the session management to use the tokens identified.
Other info	cookie:AWSALBCORS  cookie:AWSALB
Request	► Request line and header section (551 bytes)  ▼ Request body (0 bytes)
Response	► Status line and header section (921 bytes)  ▼ Response body (2 bytes)  OK
Parameter	AWSALBCORS
Evidence	S0hqSV7A8tWe8QmMp6PiDP0wQwIFzC/Wvz710U0LNQ5KPw /KtXDlKjzbrpsP4M04HIhJTafh9Q62rkNecDwm3NQGCXFeXNRF0lf UJ9QQdeqfq2TiQiCyxjtXiBdd
Solution	This is an informational alert rather than a vulnerability and so there is nothing to fix.

<https://tinder.com> (2)

### Information Disclosure - Suspicious Comments (1)

▼ GET <https://tinder.com>

#### Alert tags

- [OWASP\\_2021\\_A01](#)
- [WSTG-v42-INFO-05](#)
- [OWASP\\_2017\\_A03](#)

#### Alert description

The response appears to contain suspicious comments which may help an attacker. Note: Matches made within script blocks or files are against the entire content not only comments.

#### Other info

The following pattern was used: `\bUSER\b` and was detected in the element starting with:  
"`<script>window.__intiData=JSON.parse("{\"intlMessages\":{"a11yLinkInNewWindow\":\"Opens in a new window\", \"accountSettings\":\", see evidence field for the suspicious comment/snippet.`

#### Request

- Request line and header section (221 bytes)
- Request body (0 bytes)

#### Response

- Status line and header section (1240 bytes)
- Response body (394477 bytes)

#### Evidence

user

#### Solution

Remove all comments that return information that may help an attacker and fix any underlying problems they refer to.

## Re-examine Cache-control Directives (1)

▼ GET https://tinder.com/healthcheck

### Alert tags

- [WSTG-v42-ATHN-06](#)

### Alert description

The cache-control header has not been set properly or is missing, allowing the browser and proxies to cache content. For static assets like css, js, or image files this might be intended, however, the resources should be reviewed to ensure that no sensitive content will be cached.

### Request

- Request line and header section (551 bytes)
- ▼ Request body (0 bytes)

### Response

- Status line and header section (921 bytes)
- ▼ Response body (2 bytes)

OK

### Parameter

cache-control

### Solution

For secure content, ensure the cache-control HTTP header is set with "no-cache, no-store, must-revalidate". If an asset should be cached consider setting the directives "public, max-age, immutable".

## Conclusion

The web application, along with its associated authorization gateway, showcases a commendable level of upkeep and control, particularly from a cybersecurity standpoint. While there have been occurrences of information exposure, these can be suitably handled, and their corresponding risk quotient is evaluated to be minimal. It's worth noting that all detected security weaknesses fell into the categories of either being Informational or of Low severity. Moreover, the vulnerabilities that were categorized as Low are not readily exploitable, as has been evidenced in the past.

It's also important to highlight that the application's security measures are regularly updated to keep up with the evolving threat landscape. Regular security audits are conducted to identify and rectify any potential vulnerabilities. The application also employs a robust encryption mechanism to protect sensitive data and ensure secure communication. Additionally, the application's user authentication process is designed to prevent unauthorized access, further enhancing its security posture. Despite the presence of some low-level vulnerabilities, the overall security of the web application and its authorization portal is well-maintained and managed. This reflects the commitment to security and the proactive approach taken towards identifying and addressing potential security issues.

## References

[OWASP Top Ten | OWASP Foundation](#)

[WSTG - Stable | OWASP Foundation](#)

[ZAP \(zapproxy.org\)](#)

[GitHub - aboul3la/Sublist3r: Fast subdomains enumeration tool for penetration testers](#)

[GitHub - tomnomnom/httpprobe: Take a list of domains and probe for working HTTP and HTTPS servers](#)

[Kali Tools | Kali Linux Tools](#)

[Netcraft | Leader in Phishing Detection, Cybercrime Disruption and Website Takedown](#)

[Nmap: the Network Mapper - Free Security Scanner](#)