



Web Security – IE2062

10.Airbnb Bug Bounty

A D A Ihansa

IT22899606

Web Audit

airbnb.com

Contents

Introduction to Bug Bounty program and audit scope.....	4
Information gathering phase.....	6
Finding active subdomains and their states	6
Sublist3r	6
HTTPProbe	9
Netcraft.....	10
Spiderfoot.....	11
Google Dorks	13
Directory and services enumeration.....	16
Dirbuster.....	16
Nmap	18
Automated Testing	19
OWASP ZAP	19
Manual Testing	33
SQL injection.....	33
Checking for Insecure HTTP methods	35
Conclusion.....	36
References	36

Introduction to Bug Bounty program and audit scope

Airbnb is an online marketplace that connects people looking to rent out their homes with those looking for accommodations. It allows hosts to list their properties and provides a platform for guests to find and book unique places to stay. Airbnb's offerings range from single rooms to entire houses, catering to various budgets and preferences. The service is known for its diverse and unique listings, user-friendly interface, and focus on providing a personalized travel experience. It operates globally, giving travelers the opportunity to enjoy local experiences in destinations around the world. Airbnb also emphasizes community and trust, with features like reviews and verified profiles to help ensure safe and reliable transactions.

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

www.airbnb.com

www.hoteltonight.com

open.airbnb.com

support-api.airbnb.com

assets.airbnb.com

The bug bounty program specifies the eligible subdomains within its scope, stating that any subdomain falling under airbnb.com is included.

Asset name ↑	Type ↑	Coverage ↑	Max. severity ↓	Bounty ↑	Last update ↑
m.airbnb.com Higher Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
omgpro.airbnb.com Higher Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
one.airbnb.com Higher Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
open.airbnb.com Lower Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
callbacks.airbnb.com Higher Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
*.atairbnb.com Lower Impact Scope	Other	In scope	Critical	Eligible	Aug 17, 2023
*.byairbnb.com Lower Impact Scope	Other	In scope	Critical	Eligible	Aug 17, 2023
*.luxuryretreats.com Lower Impact Scope	Other	In scope	Critical	Eligible	Aug 17, 2023
com.luxuryretreats.ios Lower Impact Scope	iOS: App Store	In scope	Critical	Eligible	Aug 17, 2023
*.airnbccitizen.com Lower Impact Scope	Other	In scope	Critical	Eligible	Aug 17, 2023
support-api.airbnb.com Higher Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
next.airbnb.com Higher Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
api.airbnb.com Higher Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
*.muscache.com Lower Impact Scope	Other	In scope	Critical	Eligible	Aug 17, 2023
com.airbnb.app Higher Impact Scope	iOS: App Store	In scope	Critical	Eligible	Aug 17, 2023
Localized airbnb sites listed at the link below: https://www.airbnb.com/sitemaps/localized Higher Impact Scope	Other	In scope	Critical	Eligible	Aug 17, 2023
www.hoteltonight.com Lower Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
*.hoteltonight-test.com Lower Impact Scope	Other	In scope	Critical	Eligible	Aug 17, 2023
*.withairbnb.com Lower Impact Scope	Other	In scope	Critical	Eligible	Aug 17, 2023
www.airbnb.com Higher Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
*.airbnb.com Higher Impact Scope	Other	In scope	Critical	Eligible	Aug 17, 2023
*.airbnb-aws.com Lower Impact Scope	Other	In scope	Critical	Eligible	Aug 17, 2023
assets.airbnb.com Higher Impact Scope	Domain	In scope	Critical	Eligible	Aug 17, 2023
com.airbnb.android Higher Impact Scope	Android: Play Store	In scope	Critical	Eligible	Aug 17, 2023

Information gathering phase.

The initial phase of information gathering, commonly known as reconnaissance or recon, is crucial for obtaining insights into the nature and behavior of the target. This phase holds significant importance during audits or attacks as it facilitates the identification of potential vulnerabilities by gaining a deeper understanding of the target.

There are two main methods for conducting information gathering scans:

1. Active Scanning: This method involves generating substantial activity on the target system, often resulting in the retrieval of extensive information.
2. Passive Scanning: In contrast to active scanning, this approach minimizes disruption to the target system, albeit typically providing 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 are essential for gathering insights into their services, devices, and exposed information. This enables testers to develop 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

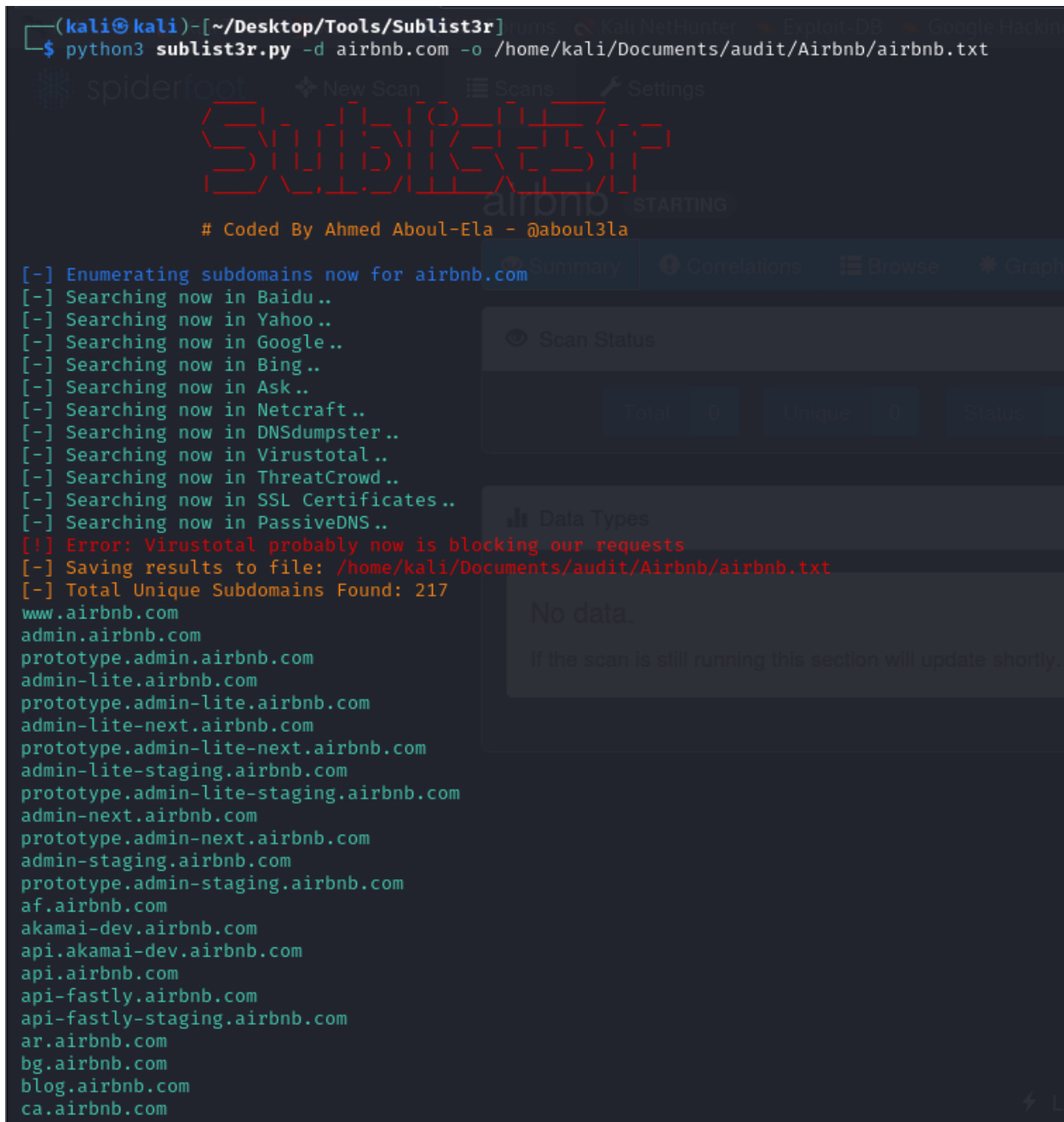
`python3 sublist3r.py -d <domain_name>`

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.

```
(kali㉿kali)-[~/Desktop/Tools/Sublist3r]
$ python3 sublist3r.py -d airbnb.com -o /home/kali/Documents/audit/Airbnb/airbnb.txt
```



```
[...] Enumerating subdomains now for airbnb.com
[-] Searching now in Baidu..
[-] Searching now in Yahoo..
[-] Searching now in Google..
[-] Searching now in Bing..
[-] Searching now in Ask..
[-] Searching now in Metcraft..
[-] Searching now in DNSdumpster..
[-] Searching now in Virustotal..
[-] Searching now in ThreatCrowd..
[-] Searching now in SSL Certificates..
[-] Searching now in PassiveDNS..
[!] Error: Virustotal probably now is blocking our requests
[-] Saving results to file: /home/kali/Documents/audit/Airbnb/airbnb.txt
[-] Total Unique Subdomains Found: 217
www.airbnb.com
admin.airbnb.com
prototype.admin.airbnb.com
admin-lite.airbnb.com
prototype.admin-lite.airbnb.com
admin-lite-next.airbnb.com
prototype.admin-lite-next.airbnb.com
admin-lite-staging.airbnb.com
prototype.admin-lite-staging.airbnb.com
admin-next.airbnb.com
prototype.admin-next.airbnb.com
admin-staging.airbnb.com
prototype.admin-staging.airbnb.com
af.airbnb.com
akamai-dev.airbnb.com
api.akamai-dev.airbnb.com
api.airbnb.com
api-fastly.airbnb.com
api-fastly-staging.airbnb.com
ar.airbnb.com
bg.airbnb.com
blog.airbnb.com
ca.airbnb.com
```

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 'httpprobe'.

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/Airbnb/airbnb.txt > /home/kali/Documents/audit/Airbnb/active_airbnb.txt
```


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

```
(kali@kali)-[~/Desktop/Tools/Sublist3r]
$ cat /home/kali/Documents/audit/Airbnb/active_airbnb.txt
https://www.airbnb.com
https://prototype.admin-lite.airbnb.com
https://admin-lite.airbnb.com
http://www.airbnb.com
https://admin.airbnb.com
http://admin-lite.airbnb.com
https://prototype.admin-next.airbnb.com
https://prototype.admin.airbnb.com
https://admin-staging.airbnb.com
http://prototype.admin-lite.airbnb.com
https://prototype.admin-lite-next.airbnb.com
http://prototype.admin-next.airbnb.com
https://admin-lite-staging.airbnb.com
https://admin-lite-next.airbnb.com
http://prototype.admin.airbnb.com
http://admin.airbnb.com
https://api.airbnb.com
http://prototype.admin-lite-next.airbnb.com
http://admin-lite-staging.airbnb.com
http://admin-lite-next.airbnb.com
https://prototype.admin-lite-staging.airbnb.com
https://admin-next.airbnb.com
http://api.airbnb.com
https://akamai-dev.airbnb.com
https://ar.airbnb.com
http://prototype.admin-lite-staging.airbnb.com
http://admin-next.airbnb.com
https://prototype.admin-staging.airbnb.com
http://ar.airbnb.com
https://ca.airbnb.com
https://blog.airbnb.com
https://bg.airbnb.com
http://akamai-dev.airbnb.com
https://api.akamai-dev.airbnb.com
http://ca.airbnb.com
https://careers.airbnb.com
http://bg.airbnb.com
http://prototype.admin-staging.airbnb.com
http://blog.airbnb.com
http://careers.airbnb.com
https://developer.airbnb.com
https://cs.airbnb.com
https://es-l.airbnb.com
https://es.airbnb.com
https://de.airbnb.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.airbnb.com>


► 🔍 Look up another site?

Share:     

Background

Site title	Airbnb Holiday rentals, cabins, beach houses & more	Date first seen	November 2008
Site rank	664	Primary language	English
Description	Get an Airbnb for every kind of trip → 7 million holiday rentals → 2 million Guest Favourites → 220+ countries and regions worldwide		

Network

Site	http://www.airbnb.com	Domain	airbnb.com
Netblock Owner	Akamai Technologies	Nameserver	ns-1977.awsdns-55.co.uk
Hosting company	Akamai Technologies	Domain registrar	markmonitor.com
Hosting country	 EU	Nameserver organisation	whois.nic.uk
IPv4 address	2.19.176.33 (VirusTotal)	Organisation	Airbnb, Inc., United States

For full site report: [Site report for http://www.airbnb.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.

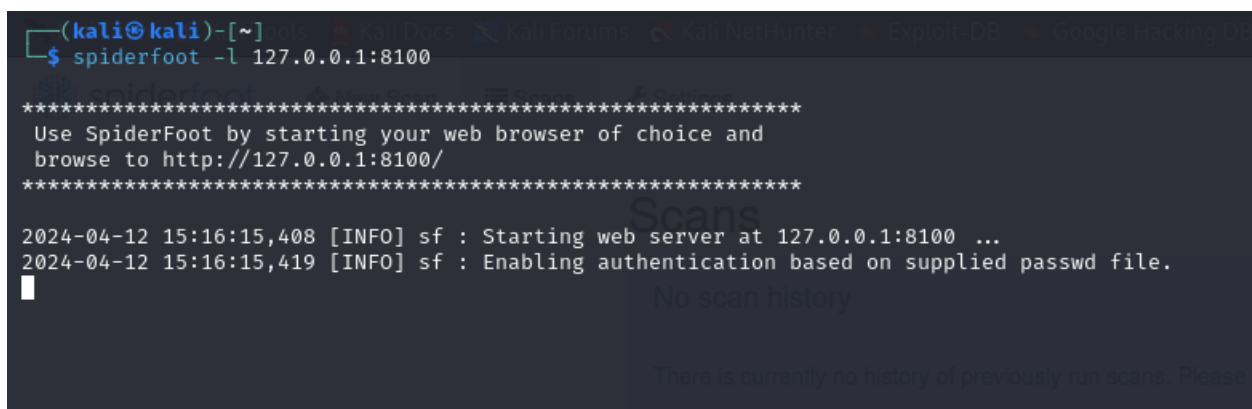
Spiderfoot

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 messages are displayed: '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 message. 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

The screenshot shows the 'New Scan' configuration page in the Spiderfoot web interface. The interface is dark-themed with a top navigation bar containing 'New Scan', 'Scans', and 'Settings' buttons. On the right, there are 'Dark Mode' and 'About' links. The main content area is titled 'New Scan' and includes a 'Scan Name' field with the value 'airbnb' and a 'Scan Target' field with the value 'www.airbnb.com'. To the right of these fields is a box containing a list of supported target formats: Domain Name, IPv4 Address, IPv6 Address, Hostname/Sub-domain, Subnet, Bitcoin Address, E-mail address, Phone Number, Human Name, Username, and Network ASN, each with an example. Below this, there are three tabs: 'By Use Case', 'By Required Data', and 'By Module'. Under 'By Use Case', there are three radio buttons: 'All', 'Footprint', and 'Investigate'. The 'Footprint' option is selected. Below the radio buttons are three descriptive boxes for each use case. At the bottom, there is a link to 'Join the SpiderFoot community Discord!'.

Scan Name
airbnb

Scan Target
www.airbnb.com

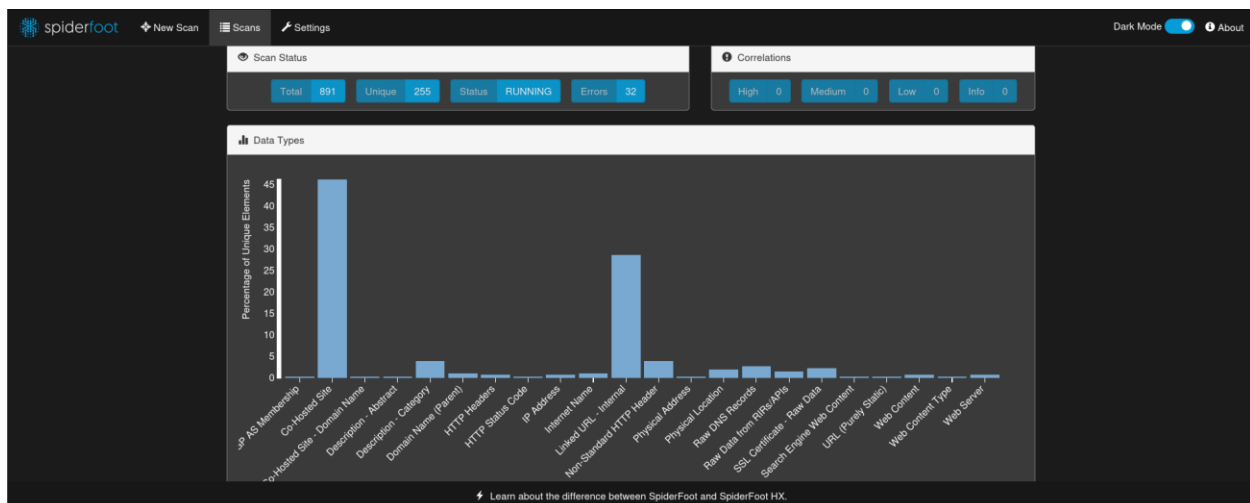
Your scan target may be one of the following. SpiderFoot will automatically detect the target type based on the format of your input:

- Domain Name: e.g. example.com
- IPv4 Address: e.g. 1.2.3.4
- IPv6 Address: e.g. 2608:4700:4700:1111
- Hostname/Sub-domain: e.g. abc.example.com
- Subnet: e.g. 1.2.3.0/24
- Bitcoin Address: e.g. 1HesYJSP1QqyPEjRQvzBL1wjuNGe7R
- 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. "jrmh2020" (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.
- ☐ Passive: When you don't want the target to even suspect that you are being investigated.

Join the SpiderFoot community Discord!

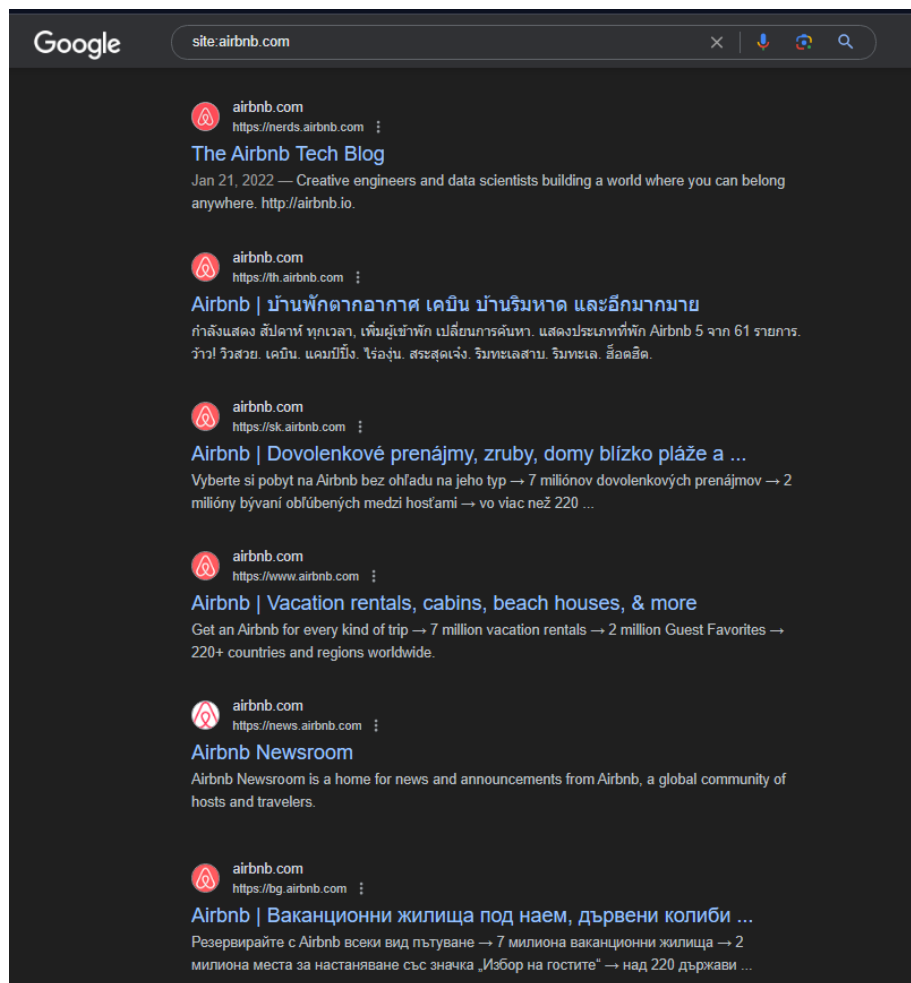


The scan has produced noteworthy findings, such as usernames, SSL certificates, and physical addresses. A significant portion of this data seems to be publicly accessible information and links leading to external websites. However, it's essential to highlight those usernames, especially when coupled 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

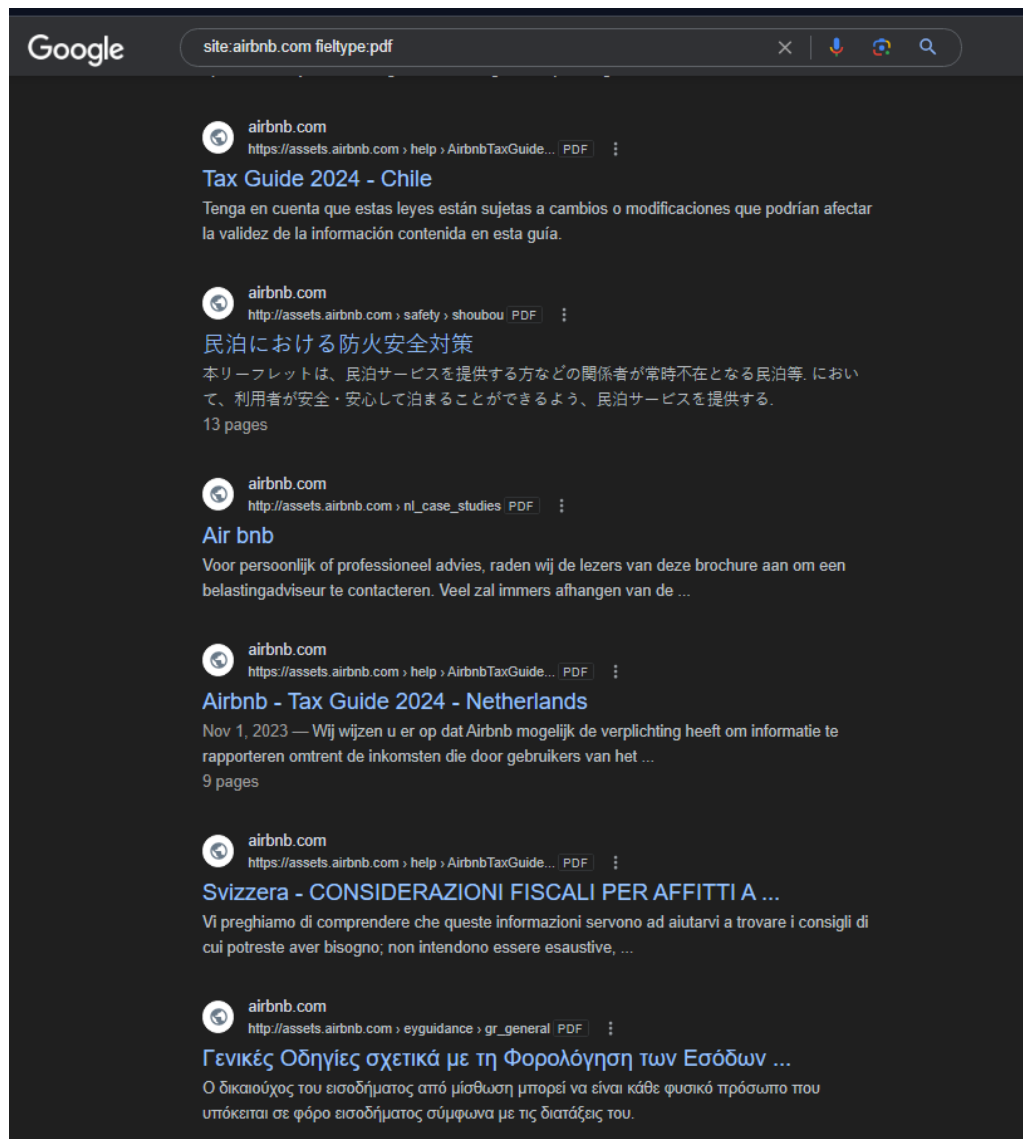
Google Dorks, also known as Google Hacking or Google Dorking, are specialized search queries that leverage Google's powerful search engine to unearth specific information and vulnerabilities that might not be accessible through standard searches. These are advanced search queries that use special operators 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:airbnb.com operator searches for websites that has “**airbnb.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

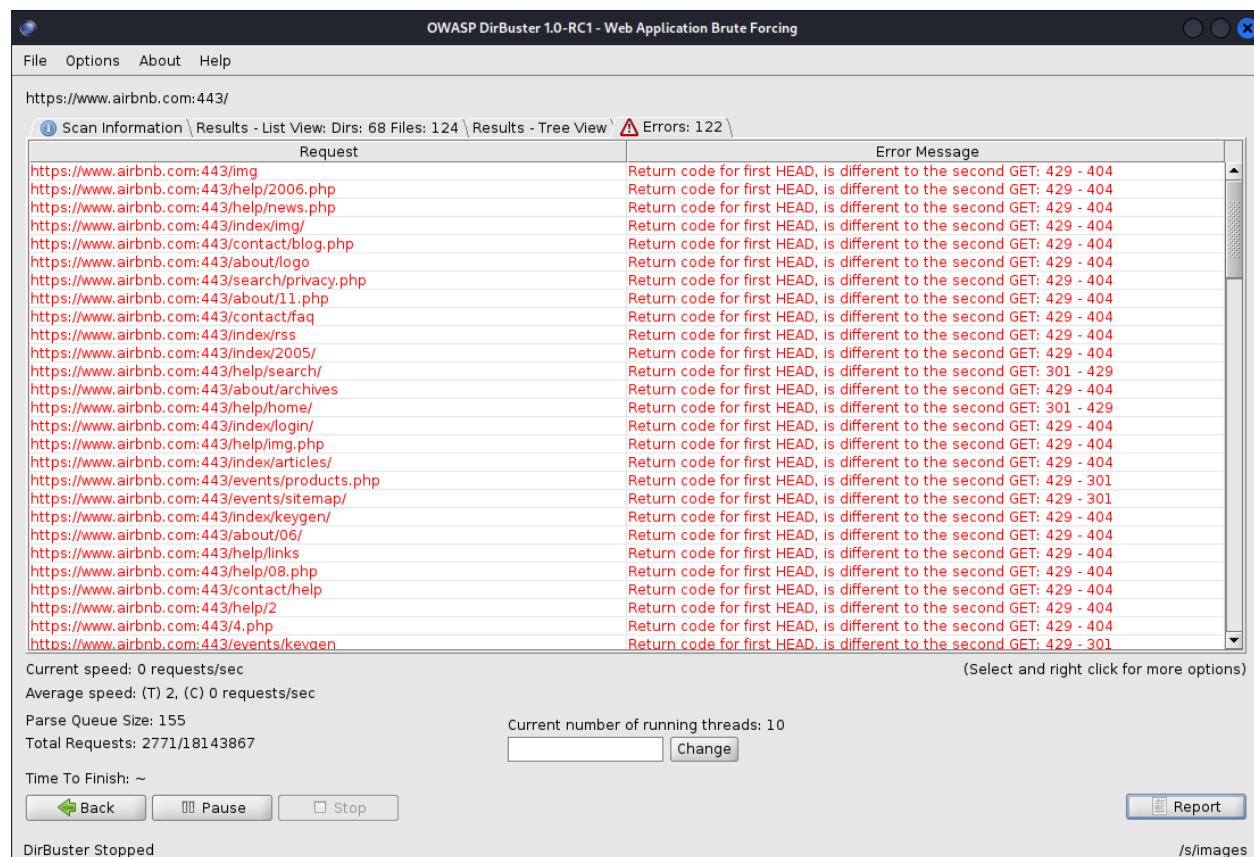
Dirbuster

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.

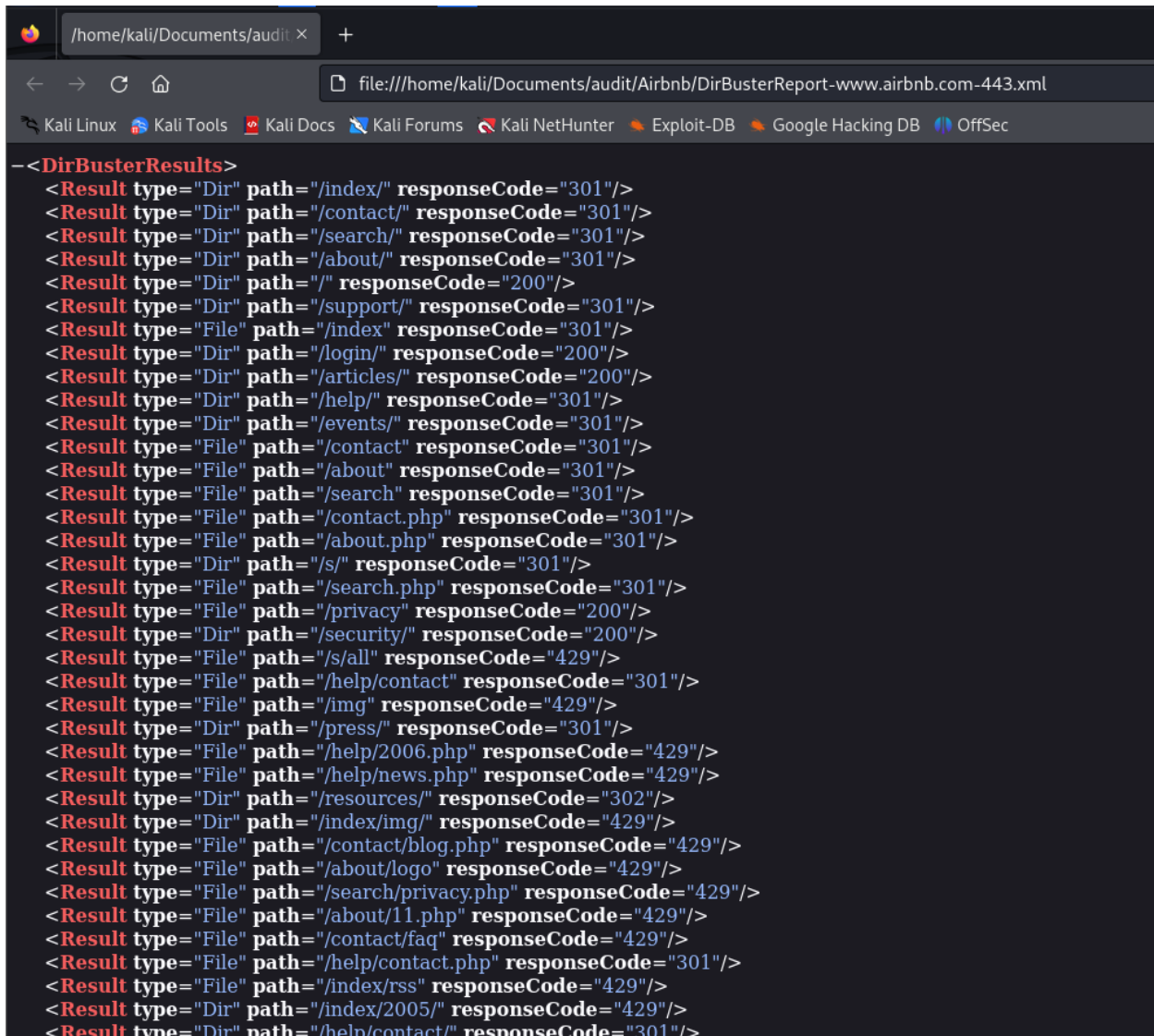
Employing this tool facilitates the identification of directories or files that might be accessible yet not overtly exposed. Furthermore, it offers a glimpse into the server's file and folder arrangement, assisting in comprehending its structure and potential vulnerabilities.

Domain: www.airbnb.com

After running the scan for some time, DirBuster encountered errors and ceased functioning. Upon further investigation, it appeared that DirBuster was unable to access the domain.



Despite the constraints, I was able to develop a broad understanding of the folder structure within the web server through exploration with DirBuster.



```
-<DirBusterResults>
<Result type="Dir" path="/index/" responseCode="301"/>
<Result type="Dir" path="/contact/" responseCode="301"/>
<Result type="Dir" path="/search/" responseCode="301"/>
<Result type="Dir" path="/about/" responseCode="301"/>
<Result type="Dir" path="/" responseCode="200"/>
<Result type="Dir" path="/support/" responseCode="301"/>
<Result type="File" path="/index" responseCode="301"/>
<Result type="Dir" path="/login/" responseCode="200"/>
<Result type="Dir" path="/articles/" responseCode="200"/>
<Result type="Dir" path="/help/" responseCode="301"/>
<Result type="Dir" path="/events/" responseCode="301"/>
<Result type="File" path="/contact" responseCode="301"/>
<Result type="File" path="/about" responseCode="301"/>
<Result type="File" path="/search" responseCode="301"/>
<Result type="File" path="/contact.php" responseCode="301"/>
<Result type="File" path="/about.php" responseCode="301"/>
<Result type="Dir" path="/s/" responseCode="301"/>
<Result type="File" path="/search.php" responseCode="301"/>
<Result type="File" path="/privacy" responseCode="200"/>
<Result type="Dir" path="/security/" responseCode="200"/>
<Result type="File" path="/s/all" responseCode="429"/>
<Result type="File" path="/help/contact" responseCode="301"/>
<Result type="File" path="/img" responseCode="429"/>
<Result type="Dir" path="/press/" responseCode="301"/>
<Result type="File" path="/help/2006.php" responseCode="429"/>
<Result type="File" path="/help/news.php" responseCode="429"/>
<Result type="Dir" path="/resources/" responseCode="302"/>
<Result type="Dir" path="/index/img/" responseCode="429"/>
<Result type="File" path="/contact/blog.php" responseCode="429"/>
<Result type="File" path="/about/logo" responseCode="429"/>
<Result type="File" path="/search/privacy.php" responseCode="429"/>
<Result type="File" path="/about/11.php" responseCode="429"/>
<Result type="File" path="/contact/faq" responseCode="429"/>
<Result type="File" path="/help/contact.php" responseCode="301"/>
<Result type="File" path="/index/rss" responseCode="429"/>
<Result type="Dir" path="/index/2005/" responseCode="429"/>
<Result type="Dir" path="/help/contact/" responseCode="301"/>
```

I manually inspected each result and did not discover any suspicious or flawed findings.

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.

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.airbnb.com/>

```
(kali@kali)~$ sudo nmap airbnb.com -sS -sV -O -oN nmap_airbnb.txt
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-21 04:55 EDT
Nmap scan report for airbnb.com (52.71.18.217)
Host is up (0.022s latency).
Other addresses for airbnb.com (not scanned): 54.165.84.136 54.208.193.172
RDNS record for 52.71.18.217: ec2-52-71-18-217.compute-1.amazonaws.com
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
25/tcp    open  smtp?
80/tcp    open  http   nginx
443/tcp   open  ssl/http nginx
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|RD=4/21|Time=66240486|P=x86_64-pc-linux-gnu|H
SF:ello,2A,"552|x20Invalid|x20domain|x20name|x20in|x20EHLO|x20command|.p\
SF:n")|R(GenericLines,28,"500|x20Syntax|x20error,\x20command|x20unrecogniz
SF:ed\r\n")|R(GetRequest,28,"500|x20Syntax|x20error,\x20command|x20unrecog
SF:nized\r\n")|R(HTTPOptions,28,"500|x20Syntax|x20error,\x20command|x20unr
SF:ecognized\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 46.74 seconds
```

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

```
(kali@kali)-[~]
$ sudo nmap hoteltonight.com -sS -sV -O -oN nmap_hoteltonight.txt
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-21 04:57 EDT
Nmap scan report for hoteltonight.com (13.35.18.65)
Host is up (0.045s latency).
Other addresses for hoteltonight.com (not scanned): 13.35.18.62 13.35.18.13 13.35.18.33
rDNS record for 13.35.18.65: server-13-35-18-65.sin5.r.cloudfront.net
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
80/tcp    open  http      Amazon CloudFront httpd
443/tcp   open  ssl/http  Amazon CloudFront httpd
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 30.40 seconds
```

Automated Testing

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

OWASP ZAP

The Open Web Application Security Project Zed Attack Proxy (OWASP ZAP) is a well-known open-source vulnerability scanner recognized for its ability to operate as a Man-in-the-Middle (MITM) proxy. It evaluates various vulnerabilities by examining responses from the web application or server. OWASP ZAP is notably user-friendly and offers customization options through the installation of modules, allowing for efficient management of results.

Within this proxy, there are primarily two types of scans available:

1. Automated Scan: Users input the target URL and initiate the attack. The behavior can be customized by selecting the ZAP mode, triggering all scripts against the target to detect vulnerabilities and generate reports accordingly.
2. Manual Explore: Users can navigate to the target web application and begin 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 in automated mode.

Alert counts by site and risk

This table shows, for each site for which one or more alerts were raised, the number of alerts raised at each risk level.

Alerts with a confidence level of "False Positive" have been excluded from these counts.

(The numbers in brackets are the number of alerts raised for the site at or above that risk level.)

		Risk			
		High (= High)	Medium (>= Medium)	Low (>= Low)	Informational (>= Informational)
Site	https://4620401.fls.doubleclick.net	0 (0)	2 (2)	1 (3)	0 (3)
	https://www.google.com	0 (0)	0 (0)	1 (1)	0 (1)
	https://www.googletagmanager.com	0 (0)	0 (0)	1 (1)	0 (1)
	https://a0.muscache.com	0 (0)	1 (1)	2 (3)	1 (4)
	https://www.airbnb.com	1 (1)	6 (7)	6 (13)	7 (20)

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

Below 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://www.airbnb.com> (1)

PII Disclosure (1)

▼ GET <https://www.airbnb.com/>

Alert tags

- [OWASP_2021_A04](#)
- [OWASP_2017_A03](#)

Alert description

The response contains Personally Identifiable Information, such as CC number, SSN and similar sensitive data.

Other Info

Credit Card Type detected: Maestro

Bank Identification Number: 572672

Brand: MAESTRO

Category:

Issuer:

Request

- Request line and header section (422 bytes)
- ▼ Request body (0 bytes)

Response

- Status line and header section (7103 bytes)
- Response body (654207 bytes)

Evidence

5726720380242754067

Solution

Check the response for the potential presence of personally identifiable information (PII), ensure nothing sensitive is leaked by the application.

<https://www.airbnb.com> (1)

Application Error Disclosure (1)

▼ GET https://www.airbnb.com/api/v3/GetConsentForUserQuery/58439ee95ca9bce5f2e7d92a2b7f74c53a861277cb37ff9d2f4b7b08a1a1352e?operationName=GetConsentForUserQuery&locale=en¤cy=LKR&variables=%7B%22includeConfigView%22%3Atrue%2C%22deviceId%22%3A%221714197044_MTFmMzg50GNiMGZi%22%7D&extensions=%7B%22persistedQuery%22%3A%7B%22version%22%3A1%2C%22sha256Hash%22%3A%2258439ee95ca9bce5f2e7d92a2b7f74c53a861277cb37ff9d2f4b7b08a1a1352e%22%7D%7D

Alert tags

- [WSTG-v42-ERRH-02](#)
- [WSTG-v42-ERRH-01](#)
- [OWASP_2021_A05](#)
- [OWASP_2017_A06](#)

Alert description

This page contains an error/warning message that may disclose sensitive information like the location of the file that produced the unhandled exception. This information can be used to launch further attacks against the web application. The alert could be a false positive if the error message is found inside a documentation page.

Request

► Request line and header section (1811 bytes)

▼ Request body (0 bytes)

Response

► Status line and header section (2096 bytes)

► Response body (118831 bytes)

Evidence

ASP.NET_SessionId

Solution

Review the source code of this page. Implement custom error pages. Consider implementing a mechanism to provide a unique error reference/identifier to the client (browser) while logging the details on the server side and not exposing them to the user.

<https://www.airbnb.com> (1)

Absence of Anti-CSRF Tokens (1)

▼ GET <https://www.airbnb.com/>

Alert tags

- [OWASP_2021_A01](#)
- [WSTG-v42-SESS-05](#)
- [OWASP_2017_A05](#)

Alert description

No Anti-CSRF tokens were found in a HTML submission form.

A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an action as the victim. The underlying cause is application functionality using predictable URL/form actions in a repeatable way. The nature of the attack is that CSRF exploits the trust that a web site has for a user. By contrast, cross-site scripting (XSS) exploits the trust that a user has for a web site. Like XSS, CSRF attacks are not necessarily cross-site, but they can be. Cross-site request forgery is also known as CSRF, XSRF, one-click attack, session riding, confused deputy, and sea surf.

CSRF attacks are effective in a number of situations, including:

- * The victim has an active session on the target site.
- * The victim is authenticated via HTTP auth on the target site.
- * The victim is on the same local network as the target site.

CSRF has primarily been used to perform an action against a target site using the victim's privileges, but recent techniques have been discovered to disclose information by gaining access to the response. The risk of information disclosure is dramatically increased when the target site is vulnerable to XSS, because XSS can be used as a platform for CSRF, allowing the attack to operate within the bounds of the same-origin policy.

Other info	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "bigsearch-query-location-input" "refinement_paths[]"].
Request	<ul style="list-style-type: none"> ► Request line and header section (422 bytes) ▼ Request body (0 bytes)
Response	<ul style="list-style-type: none"> ► Status line and header section (7104 bytes) ► Response body (654208 bytes)
Evidence	<code><form class="f114qjlg atm_gi_xjk4d9 atm_j3_1an8f3t dir dir-ltr" action="/s/homes" method="get" role="search"></code>
Solution	<p>Phase: Architecture and Design</p> <p>Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid.</p> <p>For example, use anti-CSRF packages such as the OWASP CSRFGuard.</p> <p>Phase: Implementation</p> <p>Ensure that your application is free of cross-site scripting issues, because most CSRF defenses can be bypassed using attacker-controlled script.</p> <p>Phase: Architecture and Design</p> <p>Generate a unique nonce for each form, place the nonce into the form, and verify the nonce upon receipt of the form. Be sure that the nonce is not predictable (CWE-330).</p> <p>Note that this can be bypassed using XSS.</p> <p>Identify especially dangerous operations. When the user performs a dangerous operation, send a separate confirmation request to ensure that the user intended to perform that operation.</p> <p>Note that this can be bypassed using XSS.</p> <p>Use the ESAPI Session Management control.</p> <p>This control includes a component for CSRF.</p> <p>Do not use the GET method for any request that triggers a state change.</p> <p>Phase: Implementation</p> <p>Check the HTTP Referer header to see if the request originated from an expected page. This could break legitimate functionality, because users or proxies may have disabled sending the Referer for privacy reasons.</p>

<https://www.airbnb.com> (5)

Cookie No HttpOnly Flag (1)

▼ GET <https://www.airbnb.com/>

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 (422 bytes)

```
GET https://www.airbnb.com/ HTTP/1.1
host: www.airbnb.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
```

▼ Request body (0 bytes)

Response

► Status line and header section (7104 bytes)

► Response body (654208 bytes)

Parameter

bev

Evidence

Set-Cookie: bev

Solution

Ensure that the HttpOnly flag is set for all cookies.

<https://www.airbnb.com> (5)

Cookie No HttpOnly Flag (1)

► GET <https://www.airbnb.com/>

Cookie Without Secure Flag (1)

▼ GET <https://www.airbnb.com/update-your-browser>

Alert tags

- [OWASP_2021_A05](#)
- [WSTG-v42-SESS-02](#)
- [OWASP_2017_A06](#)

Alert description

A cookie has been set without the secure flag, which means that the cookie can be accessed via unencrypted connections.

Request

► Request line and header section (1017 bytes)

▼ Request body (0 bytes)

Response

► Status line and header section (1033 bytes)

▼ Response body (393 bytes)

```
<HTML><HEAD>
<TITLE>Access Denied</TITLE>
</HEAD><BODY>
<H1>Access Denied</H1>
```

```
You don't have permission to access
"http&#58;&#47;&#47;www&#46;airbnb&#46;com&#47;update&#45;your&#45;browser" on this server.<P>
Reference&#32;&#35;18&#46;41d2017&#46;1714197079&#46;4b32ee4f
<P>https&#58;&#47;&#47;errors&#46;edgesuite&#46;net&#47;18&#46;41d2017&#46;1714197079&#46;4b32ee4f</P>
</BODY>
</HTML>
```

Parameter

cdn_exp_407dea183d85926f8

Evidence

Set-Cookie: cdn_exp_407dea183d85926f8

Solution

Whenever a cookie contains sensitive information or is a session token, then it should always be passed using an encrypted channel. Ensure that the secure flag is set for cookies containing such sensitive information.

Cross-Domain JavaScript Source File Inclusion (1)

▼ GET https://www.airbnb.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 (422 bytes)
- ▼ Request body (0 bytes)

Response

- Status line and header section (7104 bytes)
- Response body (654208 bytes)

Parameter

https://a0.muscache.com/airbnb/static/packages/web/common/frontend/hyperloop-browser/metroRequire.ae703c8f7e.js

Evidence

```
<script src="https://a0.muscache.com/airbnb/static/packages/web/common/frontend/hyperloop-browser/metroRequire.ae703c8f7e.js" crossorigin="anonymous"></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.

<https://www.airbnb.com> (1)

Session Management Response Identified (1)

▼ POST <https://www.airbnb.com/tracking/airdog>

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: _user_attributes

Request

► Request line and header section (1122 bytes)

▼ Request body (372 bytes)

```
[{"type":"count","metric":"facebook.js.get_login_status.started","value":1,"tags":["loop:core-guest-loop","app:core-guest-spa","host:www.airbnb.com","js_env:browser","protocol:http"]}, {"type":"count","metric":"facebook_sdk.no_cookie_check","value":1,"tags":["country:US","loop:core-guest-loop","app:core-guest-spa","host:www.airbnb.com","js_env:browser","protocol:http"]}]]
```

Response

► Status line and header section (974 bytes)

▼ Response body (0 bytes)

Parameter

_user_attributes

Evidence

```
%7B%22device_profiling_session_id%22%3A%221714197044--blaf0af9a1b207619160d7d3%22%2C%22giftcard_profiling_session_id%22%3A%221714197044--93a5d7dbc6368bcca9e0a9fc%22%2C%22reservation_profiling_session_id%22%3A%221714197044--70214508a2ab77fb6d9b07bd%22%2C%22curr%22%3A%22LKR%22%7D
```

Solution

This is an informational alert rather than a vulnerability and so there is nothing to fix.

User Controllable HTML Element Attribute (Potential XSS) (1)

▼ GET <https://www.airbnb.com/help/article/2273?locale=en>

Alert tags

- [OWASP_2021_A03](#)
- [OWASP_2017_A01](#)

Alert description

This check looks at user-supplied input in query string parameters and POST data to identify where certain HTML attribute values might be controlled. This provides hot-spot detection for XSS (cross-site scripting) that will require further review by a security analyst to determine exploitability.

Other info

User-controlled HTML attribute values were found. Try injecting special characters to see if XSS might be possible. The page at the following URL:

<https://www.airbnb.com/help/article/2273?locale=en>

appears to include user input in:

a(n) [html] tag [lang] attribute

The user input found was:

locale=en

The user-controlled value was:

en

Request

► Request line and header section (1227 bytes)

▼ Request body (0 bytes)

Response

► Status line and header section (5359 bytes)

► Response body (1154588 bytes)

Parameter

locale

Solution

Validate all input and sanitize output it before writing to any HTML attributes.

<https://www.airbnb.com> (4)

Cookie Poisoning (1)

▼ GET [#### Alert tags](https://www.airbnb.com/sgtm/g/collect?v=2&tid=G-2P6Q8PGG16>m=45je44o0v874143154z8551874za200&_p=1714197045091&gcs=G100&gcd=13m3m3m3m5&npa=1&dma_cps=-&dma=0&cid=1340446784.1714197053&ecid=366381040&ul=en-us&sr=1918x920&_fplc=0&ur=LK-1&pscdl=denied&ec_mode=c&sst.rnd=1046050566.1714197052&sst.gse=1&sst.etld=google.lk&sst.gcd=13m3m3m3m5&sst.tft=1714197045091&sst.ude=0&_s=1&dp=%2F&sid=1714197053&sct=1&seg=0&dl=https%3A%2F%2Fwww.airbnb.com%2F&dt=Airbnb%20%7C%20Vacation%20rentals%2C%20cabins%2C%20beach%20houses%2C%20%26%20more&en=page_view&_fv=1&_nsi=1&_ss=2&ep.content_group=Home%20Page&ep.fb_matching_url=https%3A%2F%2Fwww.airbnb.com%2F&ep.fb_matching_city=&ep.fb_matching_country=&ep.fb_matching_state=&ep.fb_matching_email=&ep.fb_matching_phone=&ep.fb_matching_first_name=&ep.fb_matching_last_name=&ep.fb_matching_gender=&ep.fb_matching_dob=&ep.fb_matching_external_id=1714197044_MTFmMzg50GNiMGZi&ep.fb_matching_eid=36926dc9-b14f-47f6-968b-d504b7ad96e4&ep.has_account=false&ep.audience_type=visitor&epn.au=0&ep.fb_matching_eid_alternative=_&ep.tag_source=cGTM&ep.OnetrustActiveGroups=&ep.snap_uuid=&ep.event_id=ed6d9eb3-81b8-4dd4-a72f-b946dde6350c&ep.page_hostname=www.airbnb.com&ep.user_data.sha256_email_address=&ep.user_data.sha256_phone_number=&ep.user_data._tag_mode=CODE&up.user_country=LK&up.OnetrustActiveGroups=&up.content_group=Home%20Page&tfd=11297&richsstsse</p></div><div data-bbox=)

- [OWASP 2021_A03](#)
- [OWASP 2017_A01](#)

Alert description

This check looks at user-supplied input in query string parameters and POST data to identify where cookie parameters might be controlled. This is called a cookie poisoning attack, and becomes exploitable when an attacker can manipulate the cookie in various ways. In some cases this will not be exploitable, however, allowing URL parameters to set cookie values is generally considered a bug.

Other info

An attacker may be able to poison cookie values through URL parameters. Try injecting a semicolon to see if you can add cookie values (e.g. name=controlledValue; name=anotherValue;).

This was identified at:

```
https://www.airbnb.com/sgtm/g/collect?v=2&tid=G-2P6Q8PGG16&gtm=45je44o0v874143154z8551874za200&_p=1714197045091&gcs=G100&gcd=13m3m3m3m5&npa=1&dma_cps=-&dma=0&cid=1340446784.1714197053&ecid=366381040&ul=en-us&sr=1918x920&_fplc=0&ur=LK-1&pscdl=denied&ec_mode=c&sst.md=1046050566.1714197052&sst.gse=1&sst.etId=google.lk&sst.gcd=13m3m3m3m5&sst.tft=1714197045091&sst.ude=0&_s=1&dp=%2F&sid=1714197053&sct=1&seg=0&dl=https%3A%2F%2Fwww.airbnb.com%2F&dt=Airbnb%20%7C%20Vacation%20rentals%2C%20cabins%2C%20beach%20houses%2C%20%26%20more&en=page_view&_fv=1&_nsi=1&_ss=2&ep.content_group=Home%20Page&ep.fb_matching_url=https%3A%2F%2Fwww.airbnb.com%2F&ep.fb_matching_city=&ep.fb_matching_country=&ep.fb_matching_state=&ep.fb_matching_email=&ep.fb_matching_phone=&ep.fb_matching_first_name=&ep.fb_matching_last_name=&ep.fb_matching_gender=&ep.fb_matching_dob=&ep.fb_matching_external_id=1714197044_MTFmMzg5OGNiMGZi&ep.fb_matching_eid=36926dc9-b14f-47f6-968b-d504b7ad96e4&ep.has_account=false&ep.audience_type=visitor&epn.au=0&ep.fb_matching_eid_alternative=_&ep.tag_source=cGTM&ep.OnetrustActiveGroups=&ep.snap_uid=&ep.event_id=ed6d9eb3-81b8-4dd4-a72f-b946dde6350c&ep.page_hostname=www.airbnb.com&ep.user_data.sha256_email_address=&ep.user_data.sha256_phone_number=&ep.user_data.tag_mode=CODE&up.user_country=LK&up.OnetrustActiveGroups=&up.content_group=Home%20Page&tfd=11297&richsstsse
```

User-input was found in the following cookie:

country=LK; path=/; domain=.airbnb.com; secure

The user input was:

up.user_country=LK

Request

- Request line and header section (3588 bytes)
- Request body (0 bytes)

Response

- Status line and header section (1369 bytes)
- ▼ Response body (65 bytes)

```
event: message  
data: {"response":{"status_code":200,"body":""}}
```

Parameter

up.user_country

Solution

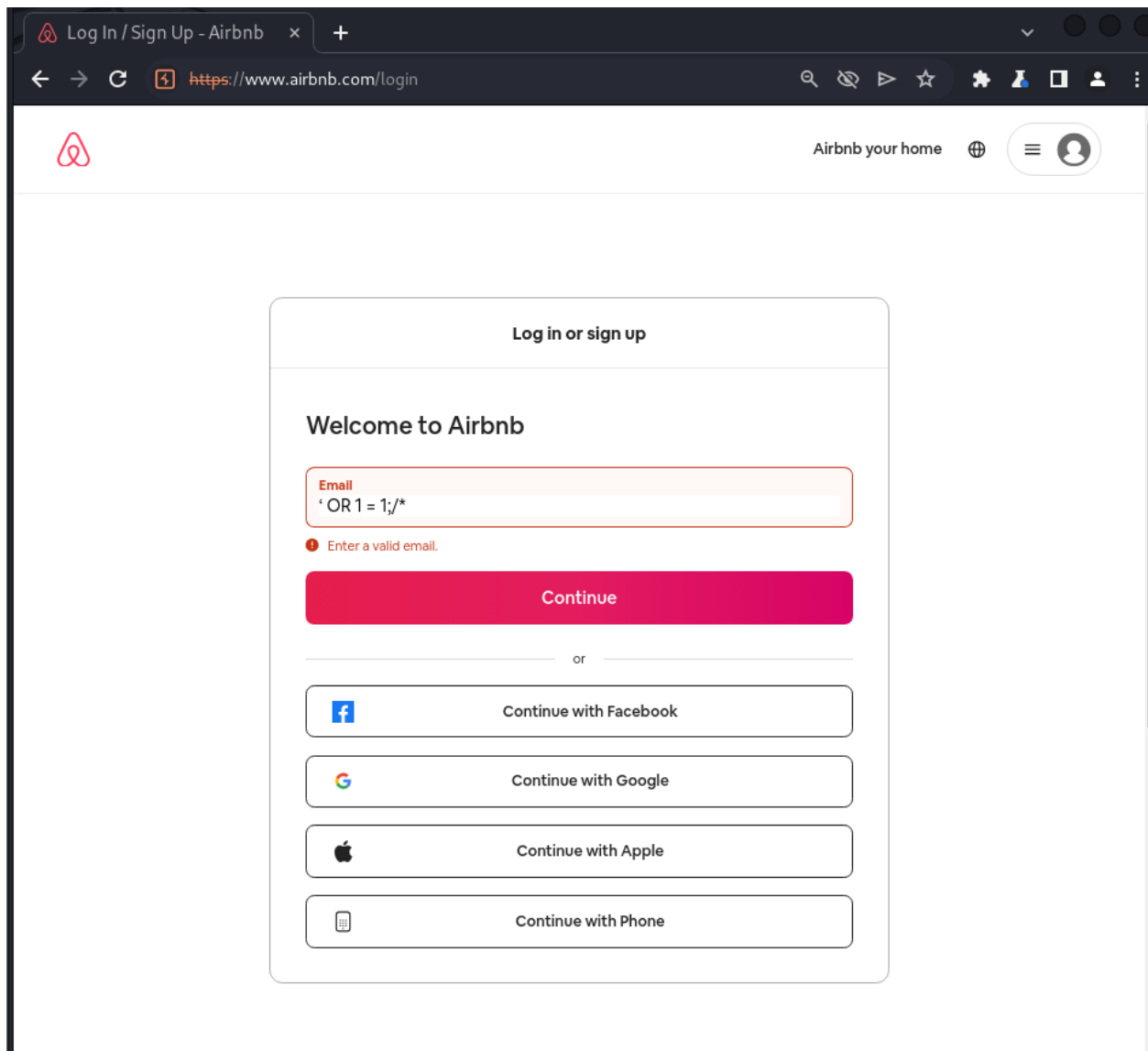
Do not allow user input to control cookie names and values. If some query string parameters must be set in cookie values, be sure to filter out semicolon's that can serve as name/value pair delimiters.

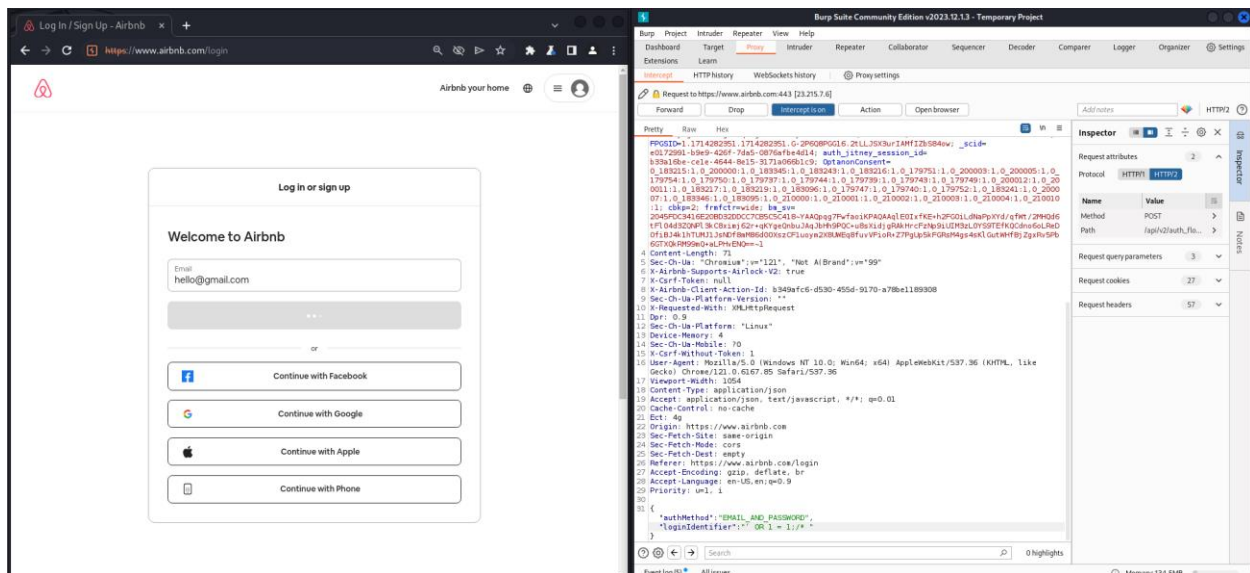
Manual Testing

SQL injection

If this page is vulnerable to SQL injection attack the Query will be executed as:

`SELECT * FROM users WHERE email address = ' OR 1 = 1; /* AND password = */--`



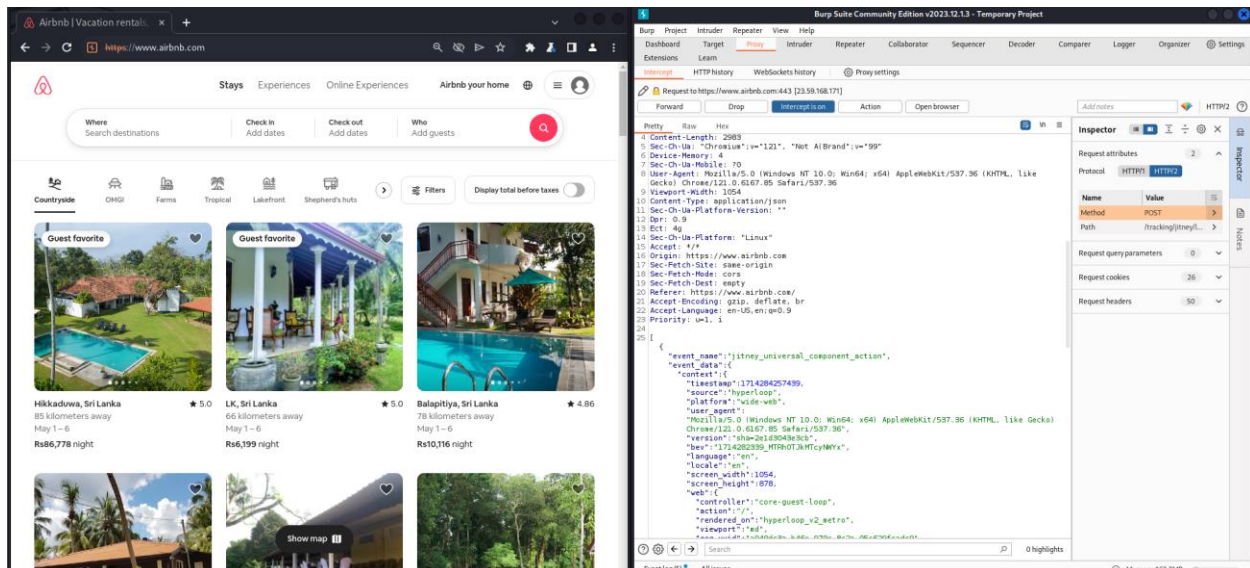


They have implemented password policies, so I attempted to intercept the request and conduct SQL injection, but I was unsuccessful. The system is well secured against SQL injection.

Checking for Insecure HTTP methods

Specific HTTP methods, particularly DELETE and PUT, pose potential security threats to the server. The DELETE method has the capability to eliminate resources from the server, while the PUT method can upload and execute files. However, these methods need to be used with care as they could potentially jeopardize the Confidentiality, Integrity, and Availability of the server and its users.

To determine the HTTP methods that are supported, I utilized the Burp proxy to capture requests and the Repeater to alter the request method. This allowed me to dispatch the altered requests to the server and scrutinize the responses.



The server only supports the POST method. Therefore, no insecure methods were utilized on the server.

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 \(zaproxy.org\)](#)

[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](#)