Sri Lanka Institute of Information Technology

VULNERABILITY ASSESSMENT - WEB AUDIT

https://www.takeaway.com

Individual Assignment

IE2062 – Web Security

Submitted by:

Student Registration Number	Student Name
IT19957258	**A.D Edirisinghe**

Date of submission: 31-05-2021

Content

Introduction

This is a security assessment of the web domain www.takeaway.com done for the Web Security module of second year second semester. This security assessment will contain vulnerability assessments performed to the sub-domains of the www.takeaway.com website. This assessment is done to assess whether the relevant website contains vulnerabilities regarding to the OWASP Top 10. All the tools that were used to perform this vulnerability assessment will be mentioned and explained briefly about how they operate.

The information gathering phase will exhibit about how the relevant website was selected, how the sub-domains were found, the scope and out of scopes of the assessment. What tools were used and how the process took place will be explained under the section “Discovering the Sub-Domains”. Scope and the out-of-scope section will include what kind of vulnerabilities are accepted and not accepted by the bug bounty program and the relevant domains which domains should be used and should not be used.

The vulnerability assessment phase will describe the information related to what the vulnerability is and the solution for them. This will include the detailed information of what type of the vulnerability that was found and what tools were used to discover this vulnerability and what are the steps that should be taken to resolve the issues found in the vulnerability.

OWASP Top 10

Open Web Application Security Project (OWASP) is an online international nonprofit foundation community that is dedicated in identifying and improving security aspects of software and web applications. It provides with tools and resources, community and networking and education and training.

OWASP community will publish a report containing the most identified vulnerabilities in web applications called as OWASP Top 10. This will include the top ten vulnerabilities that are existing and will be updated regularly. The top 10 vulnerabilities with the latest updates are given below.

1. Injection
2. Broken authentication
3. Sensitive data exposure
4. XML External Entities (XXE)
5. Broken access control
6. Security misconfigurations
7. Cross-site scripting (XSS)
8. Insecure deserialization
9. Using components with known vulnerabilities
10. Insufficient logging and monitoring

Injection	Injecting different kinds of malicious codes through any input in the given website to manipulate the website to obtain access to sensitive functions or sensitive data. This kind of vulnerabilities will occur due to simple reasons like not validating the user input and not having parameterized queries. **Few types of injections are** – SQL, NoSQL, OS, and LDAP injection
Broken Authentication	Functions like authentication and session management will often be implemented in a wrong way which would pave the path to hackers to gain advantage to exploit a website by compromising session tokens, passwords and keys. This will lead to stealing of another user's identity.
Sensitive Data Exposure	Often sensitive information like credit card information, health care information will be weakly encrypted which will allow an attacker to obtain and decrypt them easily. This could happen to data stored or data in transit.

XML External Entities (XXE)	XML documents can contain external entity references which can be processed by the XML processor. If these XML processors are poorly configured or older ones are used, this would lead to Remote Code Execution (RCE), internal port scanning, internal file shares or denial of service attacks.
Broken access control	What content is allowed for authenticated users are normally specified. But when they are not properly implemented, attackers can access unauthorized and sensitive data or functions by exploiting the wrongly implemented rules. Which will lead to broken access control.
Security misconfigurations	Misconfiguring resources or leaving the resources with the default configurations would cause security issues which attacker could easily guessed and exploited. Some misconfigurations are misconfigured HTTP headers, verbose error messages containing sensitive information and open cloud storage.
Cross-site scripting (XSS)	Malicious scripts can be executed in the user's browser by an attacker to hijack user sessions or redirect the user to a malicious website. This happens when untrusted data is included in an application.

Insecure deserialization	This vulnerability will allow an attacker to perform Remote Code Execution (RCE). Replay attacks, privilege escalation attacks and injection attacks could also be performed.
Using components with known vulnerabilities	When an application is built and run that contains known vulnerabilities. This could lead to a server takeover or a serious data loss.
Insufficient logging and monitoring	Due to insufficient logging and monitoring of a website, an attacker can keep on trying to exploit functions like login function by brute forcing passwords, an attacker could maintain access to exploited vulnerabilities and features for longer periods.

Tools used for the assessment

1. Sublist3r
2. Httprobe
3. WAFW00F
4. NMAP
5. OWASP ZAP
6. Netsparker

Sublist3r

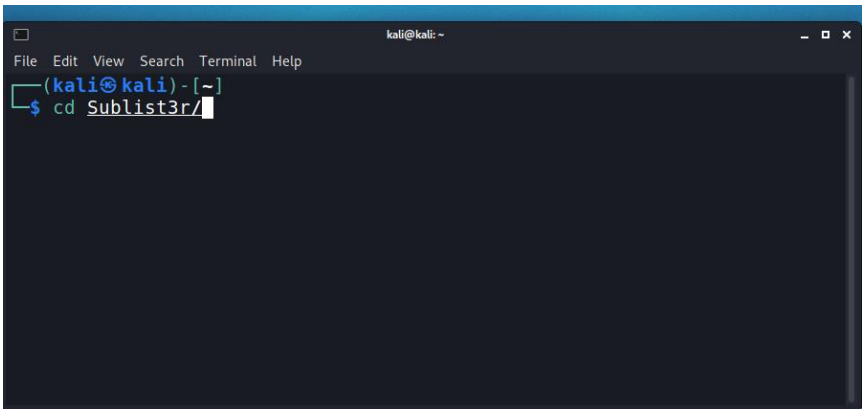
Sublist3r is a sub-domain discovery tool which uses search engines like Yahoo, Baidu, Bing, Google and use programs like Netcraft, Virustotal, DNSdumpster and ThreatCrowd for the enumeration process of sub-domains. Sublist3r can be downloaded from GitHub.

You can use the below mentioned command to download sublist3r from GitHub.

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

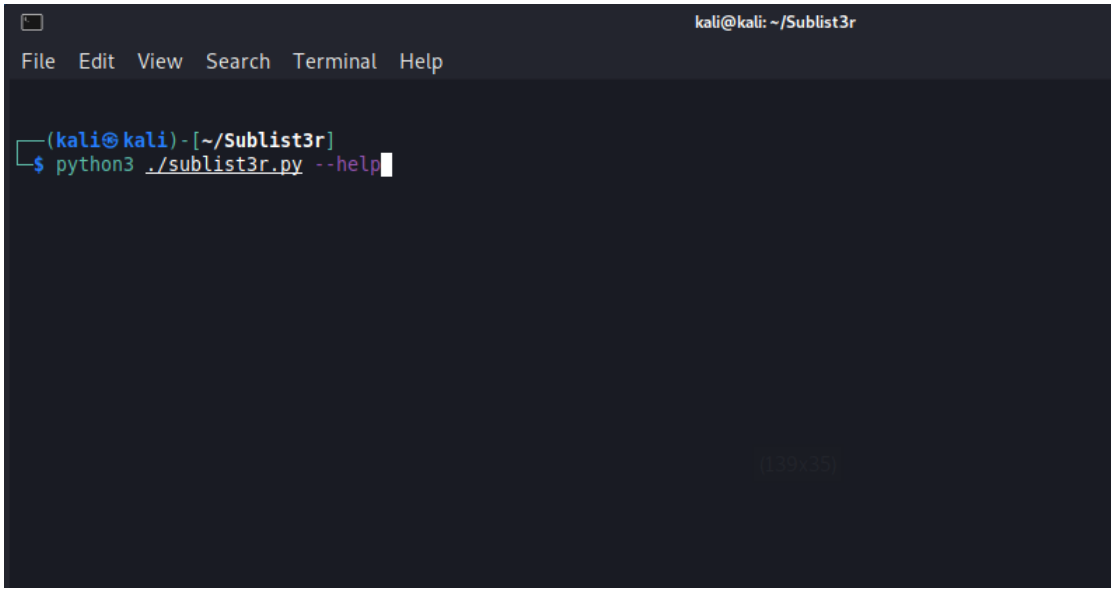
It will be downloaded to the current directory you are in. After the download completes, you can input the following command to proceed to the sublist3r directory.

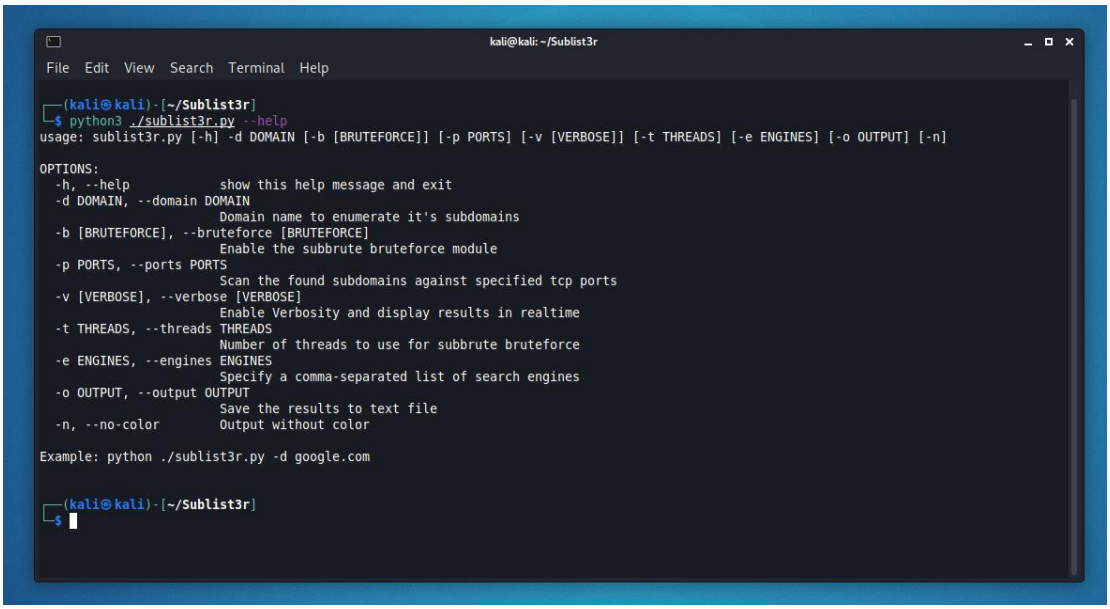
```
cd Sublist3r
```



Then, you can input the following command on the terminal to start and view the instruction on how to use.

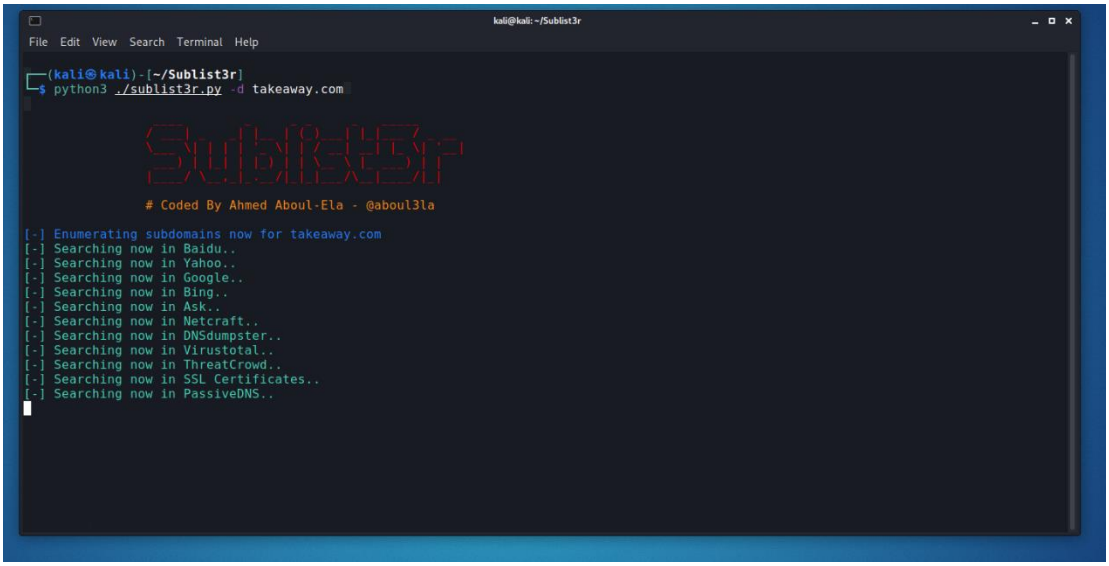
```
python3./sublist3r.py –help
```





Below is an example of how to run this tool with the command.

```
python3./sublist3r.py -d takeaway.com
```



The tool called subbrute is also embedded to this which makes this more useful that we could use a wordlist to brute force and search for sub-domains.

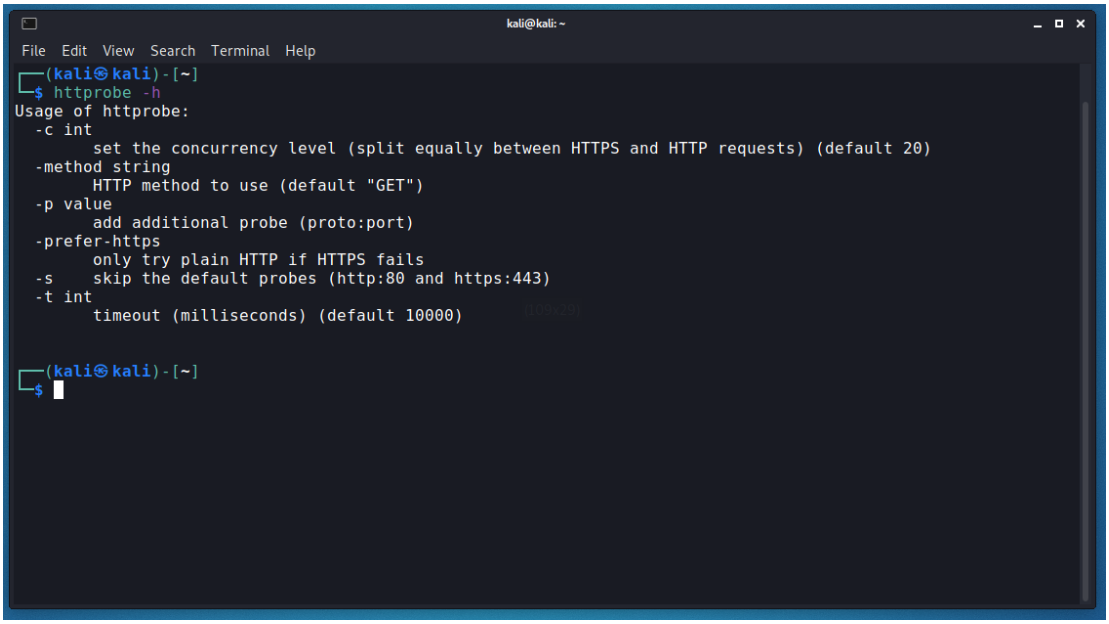
Httprobe

This tool will scan and verify the alive http or https servers. This tool comes as a pre-installed tool on kali linux and parrot linux. If you need to download this tool, you can download it by simply entering the following command. You need to install Golang in order to use the following command.

```
go get -u github.com/tomnomnom/httprobe
```

To get the instruction on how to use this tool, we can input the following command on the terminal.

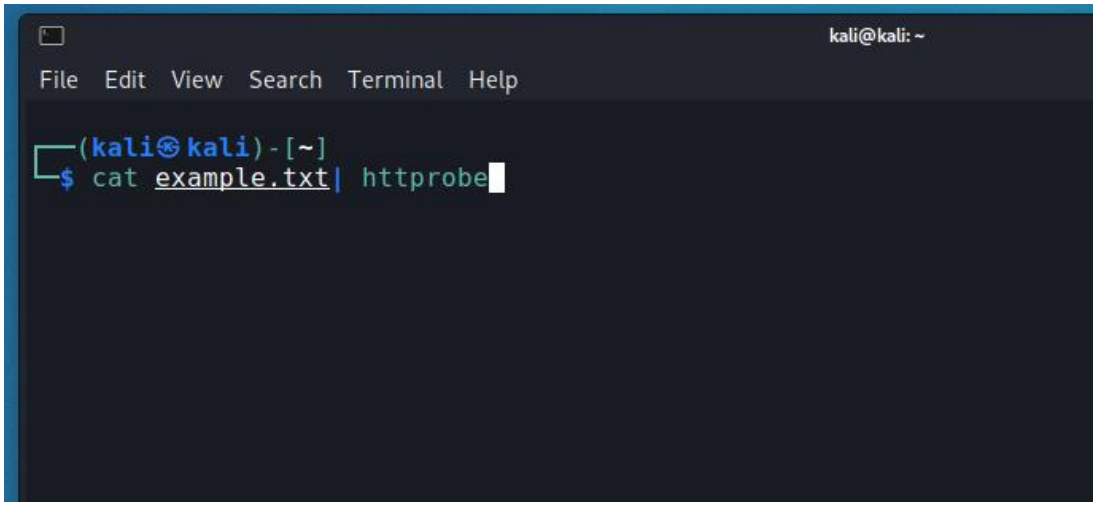
```
httprobe -h
```



To do the sub-domain enumeration, we can input the following command.

```
Cat example.txt | httprobe
```

```
example.txt = This is the file containing the sub-domain URL's
```

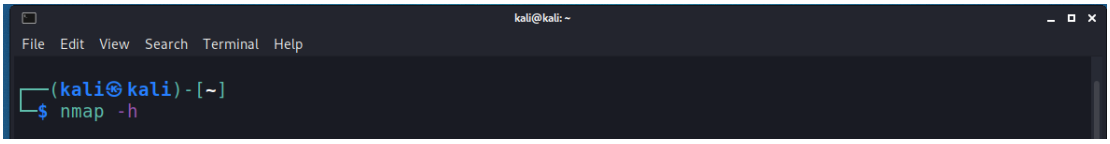


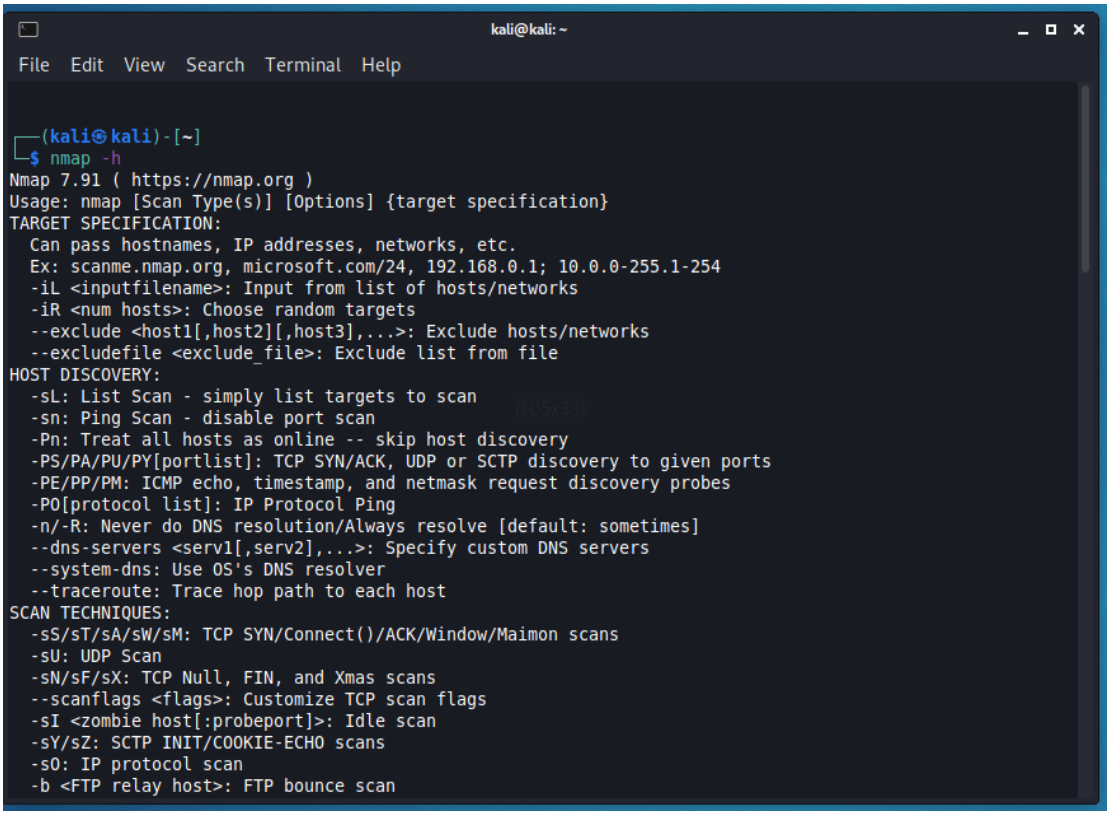
Network Mapper (NMAP)

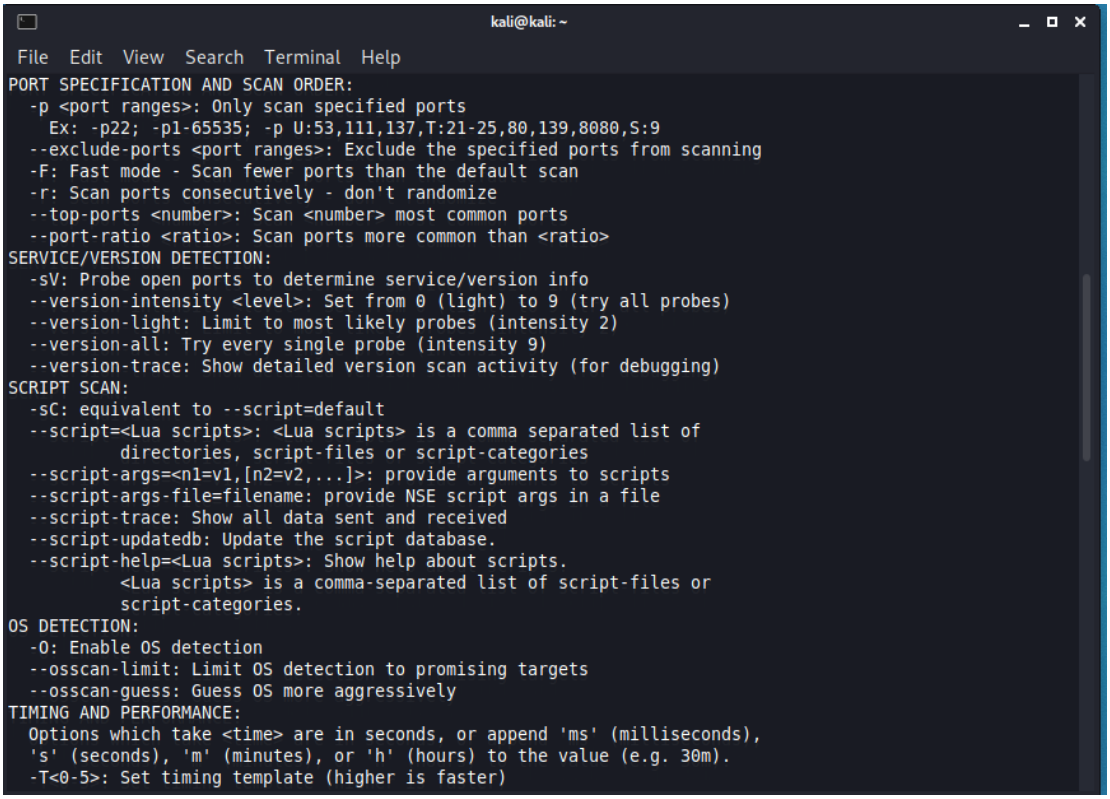
This is a tool which is used to perform scans to discover IP addresses, installed applications, find the devices that are currently running on the network, ports on the network and detect vulnerabilities. NMAP comes as a pre-installed package on Kali linux and Parrot linux.

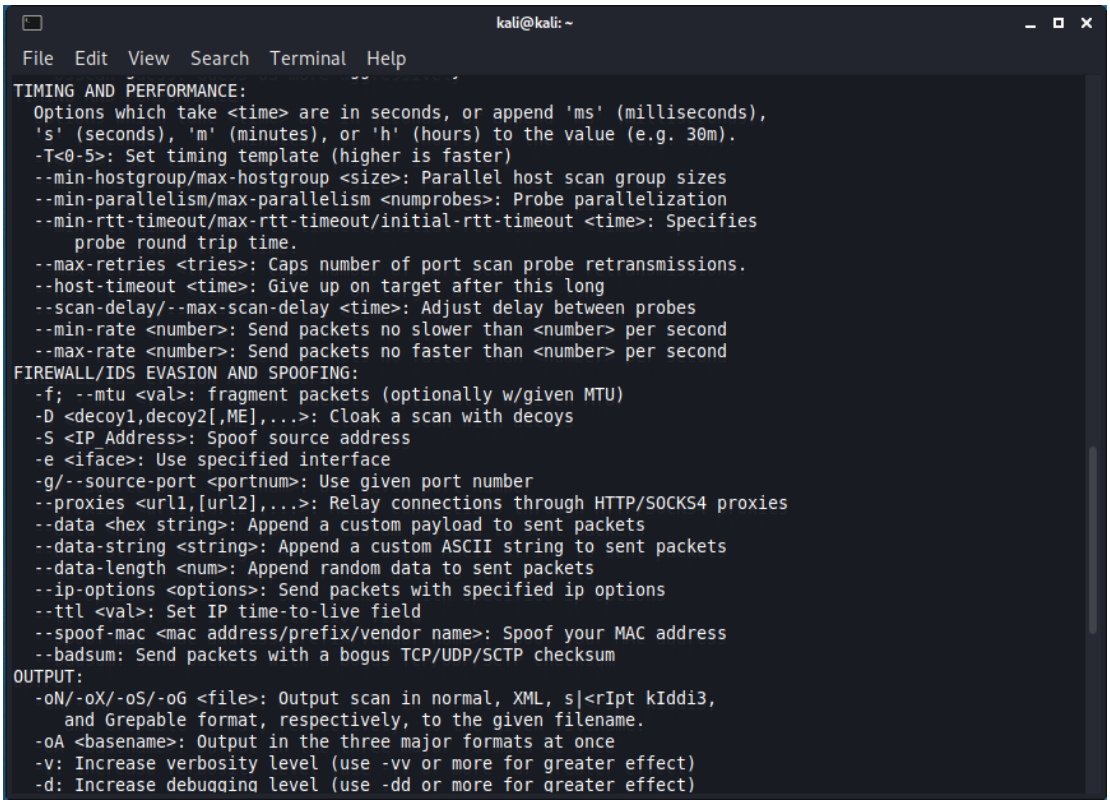
We can input the following command in the terminal to get the instruction on how to use it.

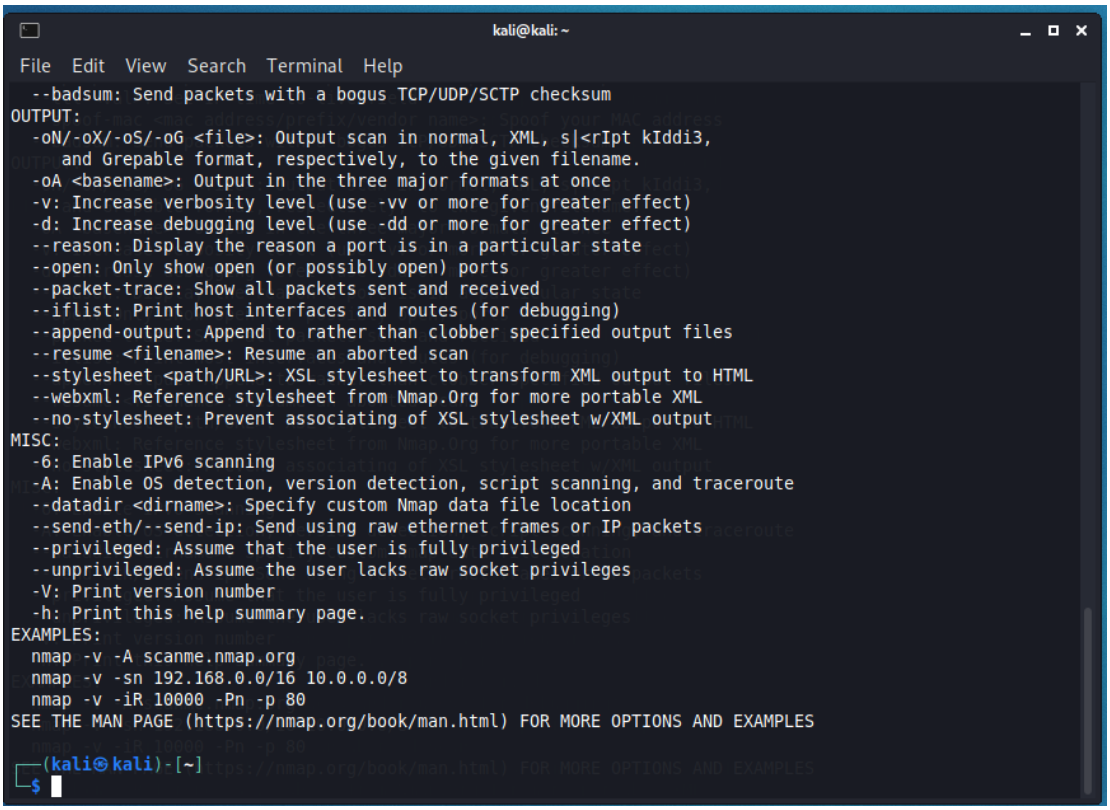
```
nmap -h
```






kali@kali: ~
File Edit View Search Terminal Help
PORT SPECIFICATION AND SCAN ORDER:
-p <port ranges>: Only scan specified ports
Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9
--exclude-ports <port ranges>: Exclude the specified ports from scanning
-F: Fast mode - Scan fewer ports than the default scan
-r: Scan ports consecutively - don't randomize
--top-ports <number>: Scan <number> most common ports
--port-ratio <ratio>: Scan ports more common than <ratio>
SERVICE/VERSION DETECTION:
-sV: Probe open ports to determine service/version info
--version-intensity <level>: Set from 0 (light) to 9 (try all probes)
--version-light: Limit to most likely probes (intensity 2)
--version-all: Try every single probe (intensity 9)
--version-trace: Show detailed version scan activity (for debugging)
SCRIPT SCAN:
-sC: equivalent to --script=default
--script=<Lua scripts>: <Lua scripts> is a comma separated list of
directories, script-files or script-categories
--script-args=<n1=v1,[n2=v2,...]>: provide arguments to scripts
--script-args-file=filename: provide NSE script args in a file
--script-trace: Show all data sent and received
--script-updatedb: Update the script database.
--script-help=<Lua scripts>: Show help about scripts.
<Lua scripts> is a comma-separated list of script-files or
script-categories.
OS DETECTION:
-O: Enable OS detection
--osscan-limit: Limit OS detection to promising targets
--osscan-guess: Guess OS more aggressively
TIMING AND PERFORMANCE:
Options which take <time> are in seconds, or append 'ms' (milliseconds),
's' (seconds), 'm' (minutes), or 'h' (hours) to the value (e.g. 30m).
-T<0-5>: Set timing template (higher is faster)


kali@kali: ~
File Edit View Search Terminal Help
TIMING AND PERFORMANCE:
Options which take <time> are in seconds, or append 'ms' (milliseconds),
's' (seconds), 'm' (minutes), or 'h' (hours) to the value (e.g. 30m).
-T<0-5>: Set timing template (higher is faster)
--min-hostgroup/max-hostgroup <size>: Parallel host scan group sizes
--min-parallelism/max-parallelism <numprobes>: Probe parallelization
--min-rtt-timeout/max-rtt-timeout/initial-rtt-timeout <time>: Specifies
probe round trip time.
--max-retries <tries>: Caps number of port scan probe retransmissions.
--host-timeout <time>: Give up on target after this long
--scan-delay/--max-scan-delay <time>: Adjust delay between probes
--min-rate <number>: Send packets no slower than <number> per second
--max-rate <number>: Send packets no faster than <number> per second
FIREWALL/IDS EVASION AND SPOOFING:
-f; --mtu <val>: fragment packets (optionally w/given MTU)
-D <decoy1,decoy2[,ME],...>: Cloak a scan with decoys
-S <IP_Address>: Spoof source address
-e <iface>: Use specified interface
-g/--source-port <portnum>: Use given port number
--proxies <url1,[url2],...>: Relay connections through HTTP/SOCKS4 proxies
--data <hex string>: Append a custom payload to sent packets
--data-string <string>: Append a custom ASCII string to sent packets
--data-length <num>: Append random data to sent packets
--ip-options <options>: Send packets with specified ip options
--ttl <val>: Set IP time-to-live field
--spoof-mac <mac address/prefix/vendor name>: Spoof your MAC address
--badsum: Send packets with a bogus TCP/UDP/SCTP checksum
OUTPUT:
-oN/-oX/-oS/-oG <file>: Output scan in normal, XML, s|<rIpt kIddi3,
and Grepable format, respectively, to the given filename.
-oA <basename>: Output in the three major formats at once
-v: Increase verbosity level (use -vv or more for greater effect)
-d: Increase debugging level (use -dd or more for greater effect)



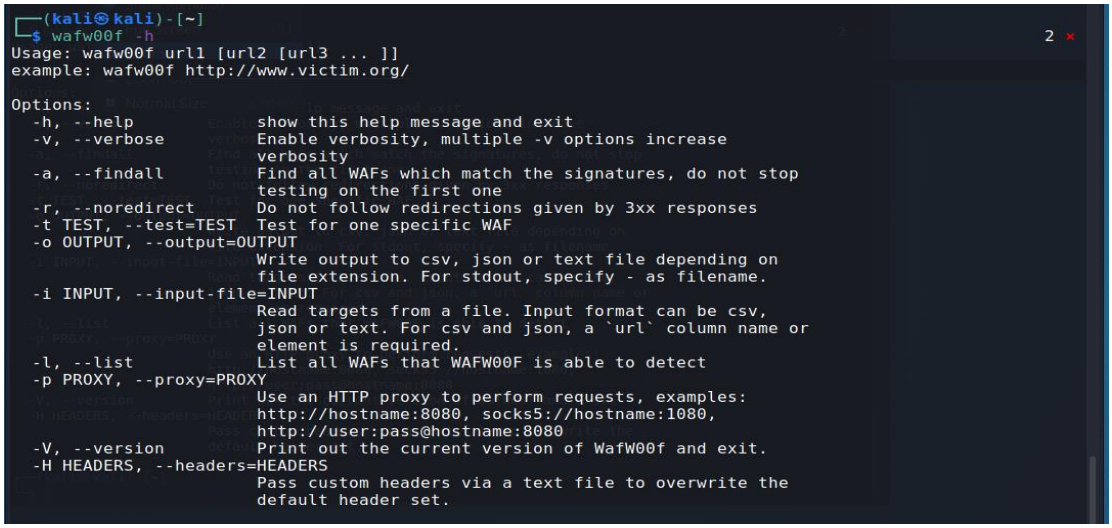
The way to type the command is shown in the instruction page in the end under examples.

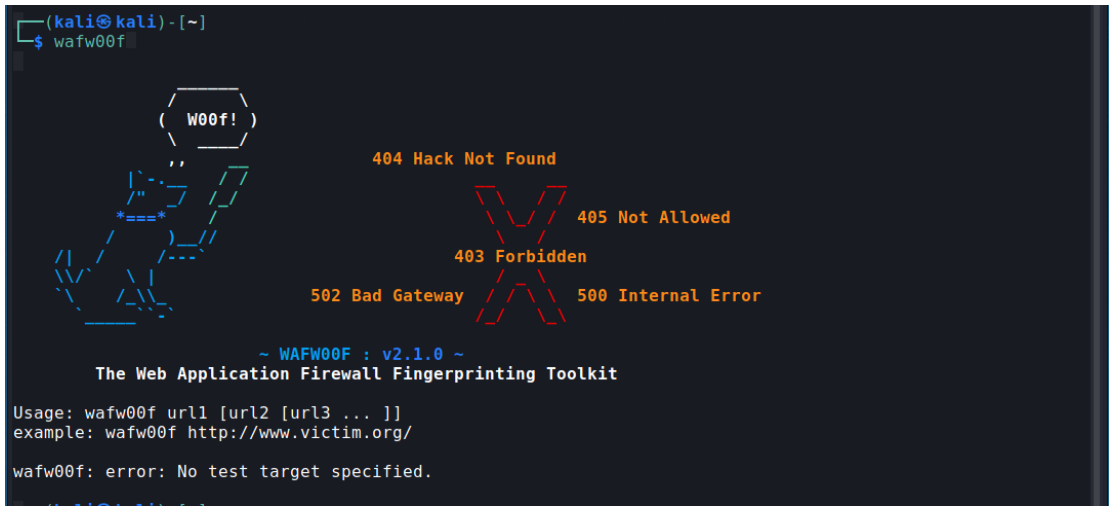
WAFW00F

This tool is a web application firewall detection tool which accurately discover whether the relevant web application is hosted behind a firewall or not and provides us with the name of the firewall provider. WAFW00F is also a tool that comes pre-installed in Kali and Parrot Linux.

We can proceed to the instructions page by simply inputting the following command on the terminal.

```
wafw00f -h
```



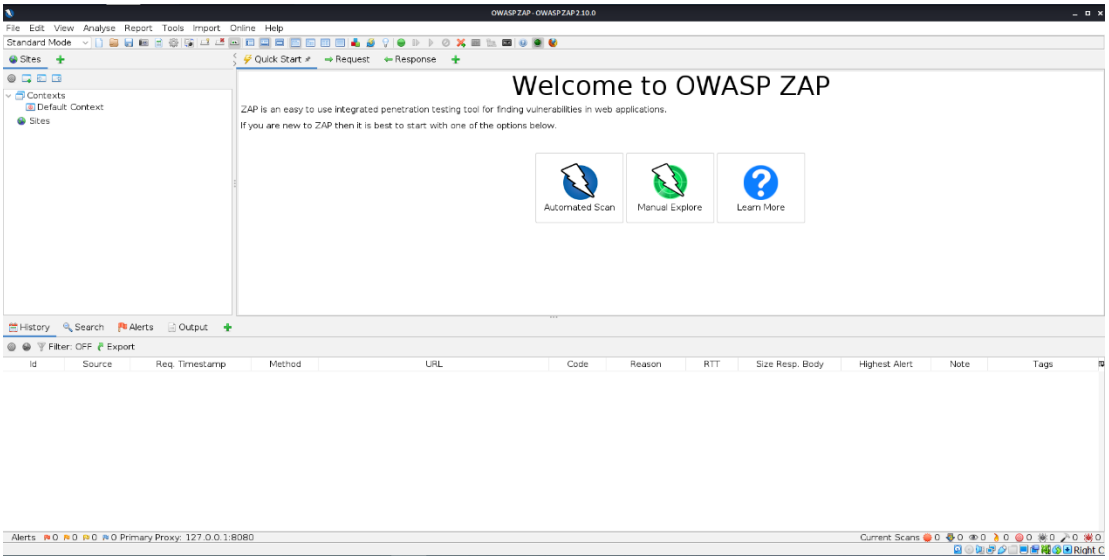


OWASP Zed Attack Proxy (ZAP)

This is a tool developed by Open Web Application Security Project (OWASP) to automatically discover and identify various web application vulnerabilities. This also acts as a proxy server to intercept web requests. This is a free and open-source software, and this is like the tool called Burp suite. This tool comes pre-installed on Kali and Parrot linux.

For other Operating system versions, you can download the software from the below mentioned website.

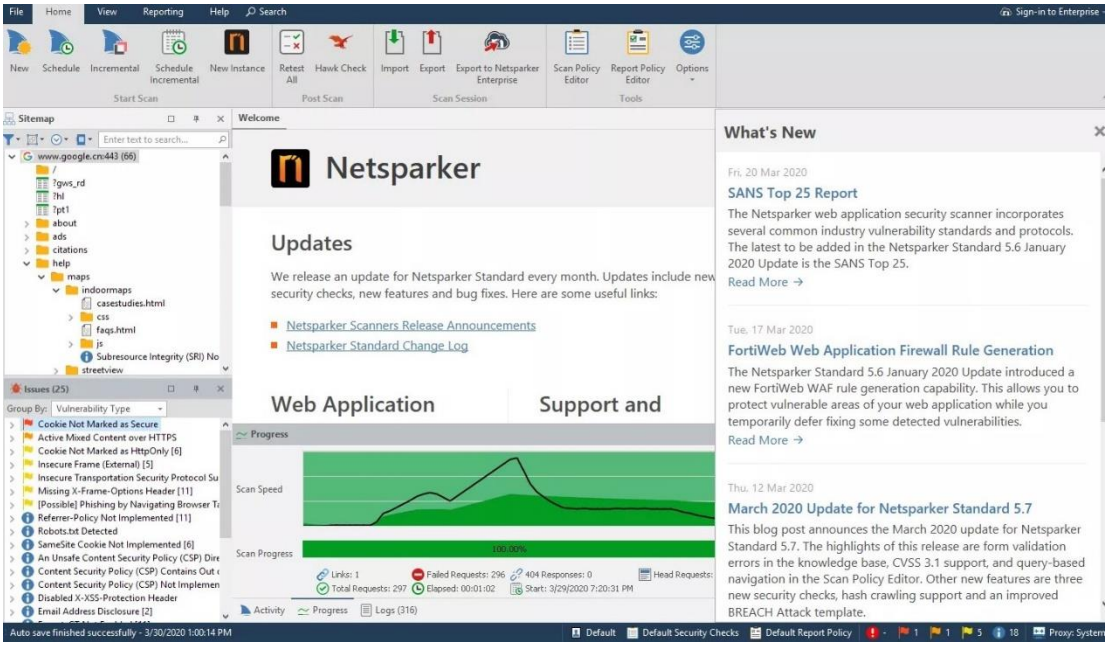
`https://www.zaproxy.org/`



One of the most important things in this tool is that we use the built-in web crawler(spider) to scan and create a site map of the relevant website. So, we can get a good understanding about how the site resources are arranged and how the site is operating.

Netsparker Professional

This is a commercial tool which will scan for different types of vulnerabilities automatically and provides a detailed report on what vulnerabilities are present in the scanned web domain. If a vulnerability is detected while scanning, this tool will automatically try to exploit that vulnerability in a safe method to confirm whether it is exploitable or not. This tool will support almost all types of websites, web servers and web applications.



Information Gathering Phase

This is the phase where all the information relevant to the relevant website is being gathered in order to do the vulnerability assessment. This phase can be called the starting phase of the vulnerability assessment. Information gathering is very crucial because this is the phase where we can identify and discover what type of server the website is using, is the website hosted behind a web application firewall, what ports are open, what operating systems are used, what IP addresses are being used, check for the sitemap etc.

The Target Domain

- https://www.takeaway.com
- This bug bounty program is hosted on the bug crowd platform.

Scope and the Out of Scope of the Assessment

Scope

Important notes for targets:

- **All main domains (takeaway.com, lieferando.de, thuisbezorgd.nl etc.; you can check if it redirects to a page with same functionality) have the same codebase behind them and thus, identical vulnerabilities on different main domains and on their subdomains will be treated as duplicates**

Some of the domains allowed for testing is given below.

- ***.takeaway.com**
- ***.thuisbezorgd.nl**
- ***.lieferando.de**
- ***.citymeal.com**
- ***.just-eat.no**

Focus Areas

We encourage researchers to focus their efforts in the following areas:

- **Business Logic Flaws**
- **Exfiltration of Sensitive or Personal Data**
- **Remote Code Execution**
- **SQL and Command Injection**
- **Authentication Bypass**
- **Cross-Site Request Forgery (CSRF) in sensitive functions**

(Mentioned above are the focus areas as mentioned in the program page given by the website)

Out of Scope

Any domain/property of Takeaway not listed in the targets section is out of scope. This includes any/all subdomains not listed above.

The specifically mentioned out of scope domains are listed below.

- **http://logistikjobs.lieferando.de/**
- **https://*.pyszne.pl**
- **https://www.lieferando.at/en/vouchercode/new-customer**

- **https://www.lieferando.at/gutschein/neukunde**
- ***.vietnammm.com**

For any other details about this bug bounty program, you can visit the following link.

https://bugcrowd.com/takeaway

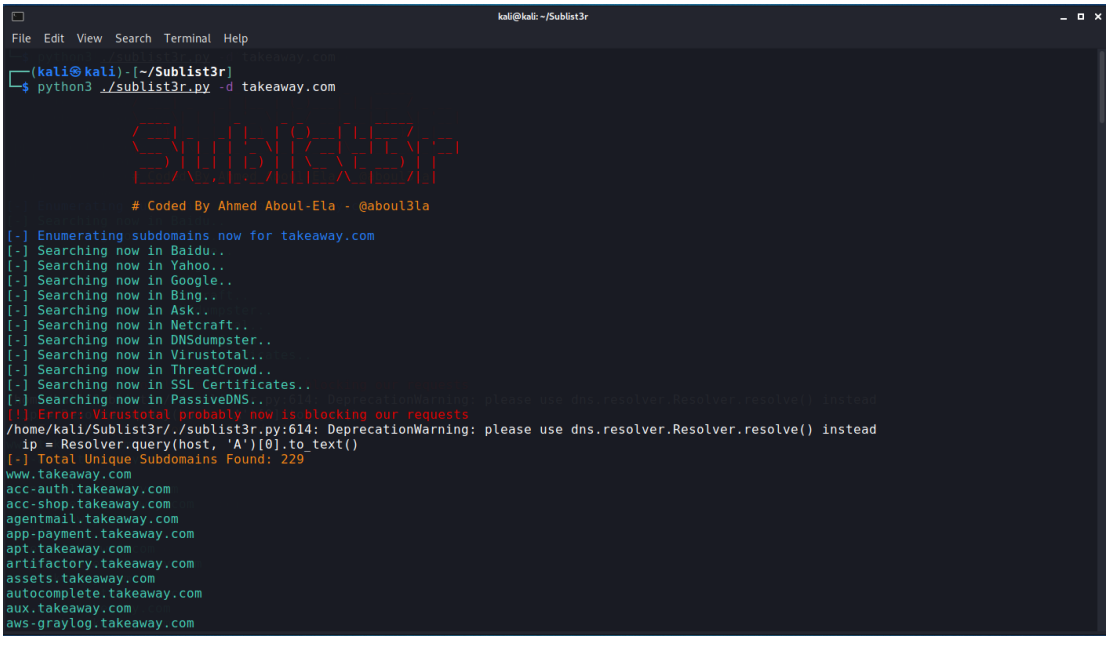
As we selected the bounty program and we have chosen the relevant domain, now we can check for the subdomains the website has.

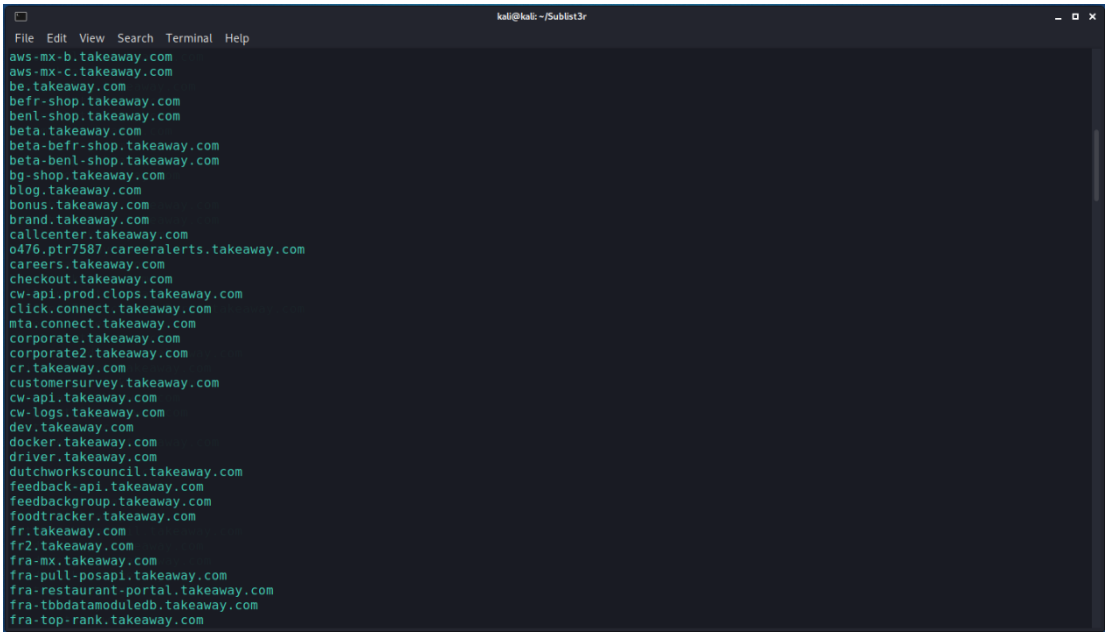
Discovering Sub-Domains

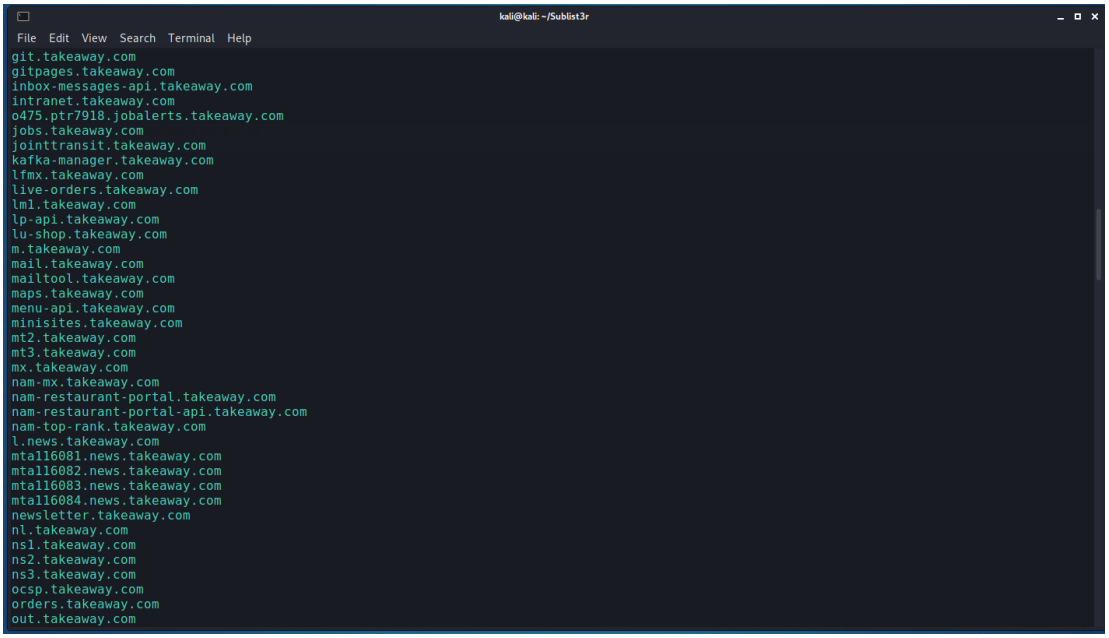
I used the tool called Sublist3r to discover the sub-domains the chosen website has. I used the following command to get the sub-domains through the Sublist3r tool.

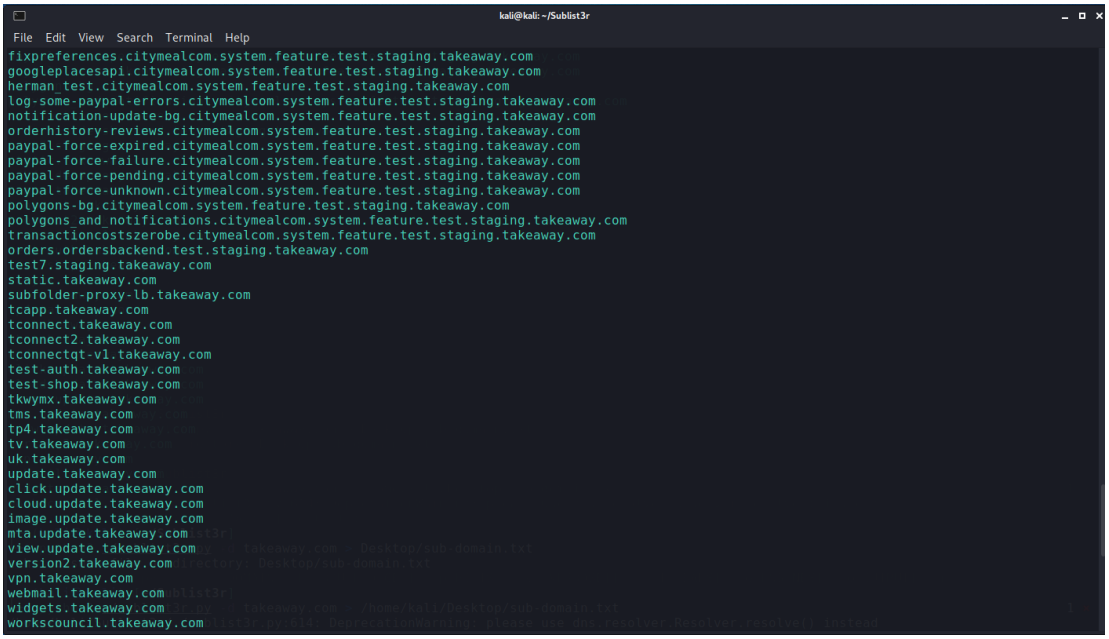
```
python3 ./sublist3r.py -d takeaway.com
```

("-d" = setting the domain name)




kali@kali: ~/Sublist3r
File Edit View Search Terminal Help
aws-mx-b.takeaway.com
aws-mx-c.takeaway.com
be.takeaway.com
befr-shop.takeaway.com
benl-shop.takeaway.com
beta.takeaway.com
beta-befr-shop.takeaway.com
beta-benl-shop.takeaway.com
bg-shop.takeaway.com
blog.takeaway.com
bonus.takeaway.com
brand.takeaway.com
callcenter.takeaway.com
o476.ptr7587.careeralerts.takeaway.com
careers.takeaway.com
checkout.takeaway.com
cw-api.prod.clops.takeaway.com
click.connect.takeaway.com
mta.connect.takeaway.com
corporate.takeaway.com
corporate2.takeaway.com
cr.takeaway.com
customersurvey.takeaway.com
cw-api.takeaway.com
cw-logs.takeaway.com
dev.takeaway.com
docker.takeaway.com
driver.takeaway.com
dutchworkscouncil.takeaway.com
feedback-api.takeaway.com
feedbackgroup.takeaway.com
foodtracker.takeaway.com
fr.takeaway.com
fr2.takeaway.com
fra-mx.takeaway.com
fra-pull-posapi.takeaway.com
fra-restaurant-portal.takeaway.com
fra-tbbdatamoduledb.takeaway.com
fra-top-rank.takeaway.com

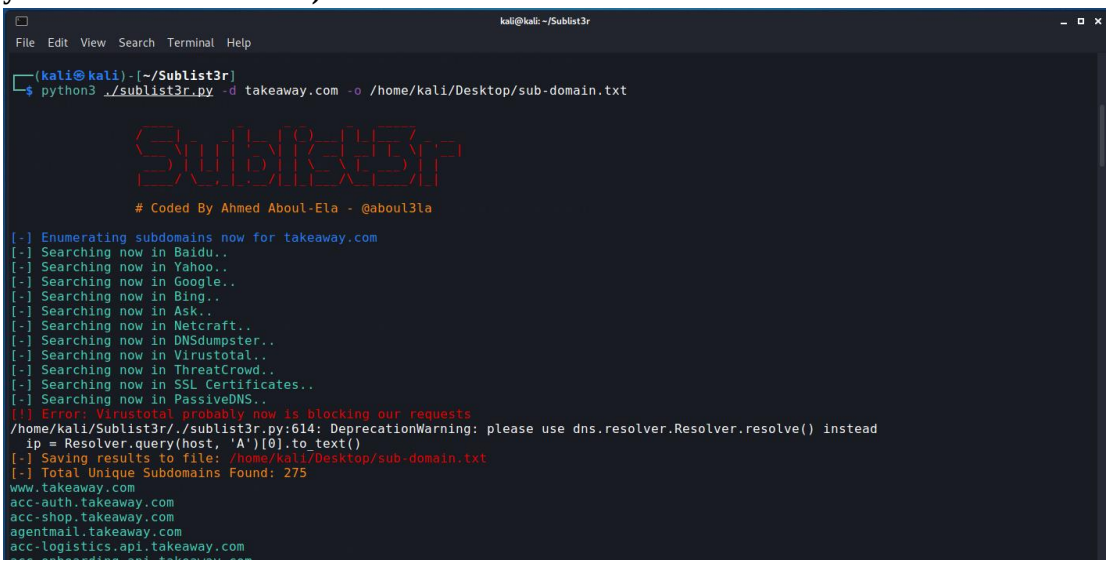

kali@kali: ~/Sublist3r
File Edit View Search Terminal Help
git.takeaway.com
gitpages.takeaway.com
inbox-messages-api.takeaway.com
intranet.takeaway.com
o475.ptr7918.jobalerts.takeaway.com
jobs.takeaway.com
jointtransit.takeaway.com
kafka-manager.takeaway.com
lfmx.takeaway.com
live-orders.takeaway.com
lm1.takeaway.com
lp-api.takeaway.com
lu-shop.takeaway.com
m.takeaway.com
mail.takeaway.com
mailtool.takeaway.com
maps.takeaway.com
menu-api.takeaway.com
minisites.takeaway.com
mt2.takeaway.com
mt3.takeaway.com
mx.takeaway.com
nam-mx.takeaway.com
nam-restaurant-portal.takeaway.com
nam-restaurant-portal-api.takeaway.com
nam-top-rank.takeaway.com
l.news.takeaway.com
mta116081.news.takeaway.com
mta116082.news.takeaway.com
mta116083.news.takeaway.com
mta116084.news.takeaway.com
newsletter.takeaway.com
nl.takeaway.com
ns1.takeaway.com
ns2.takeaway.com
ns3.takeaway.com
ocsp.takeaway.com
orders.takeaway.com
out.takeaway.com



As all the sub-domain names have been displayed on the terminal, if we want to save it directly to a text file, you can use the below mentioned command. This command will store them onto a text file.

```
python3 ./sublist3r.py -d takeaway.com -o /home/kali/Desktop/sub-domain.txt
```

(The path to new file can be an absolute path or a relative path. All depends on where you want to store the file)



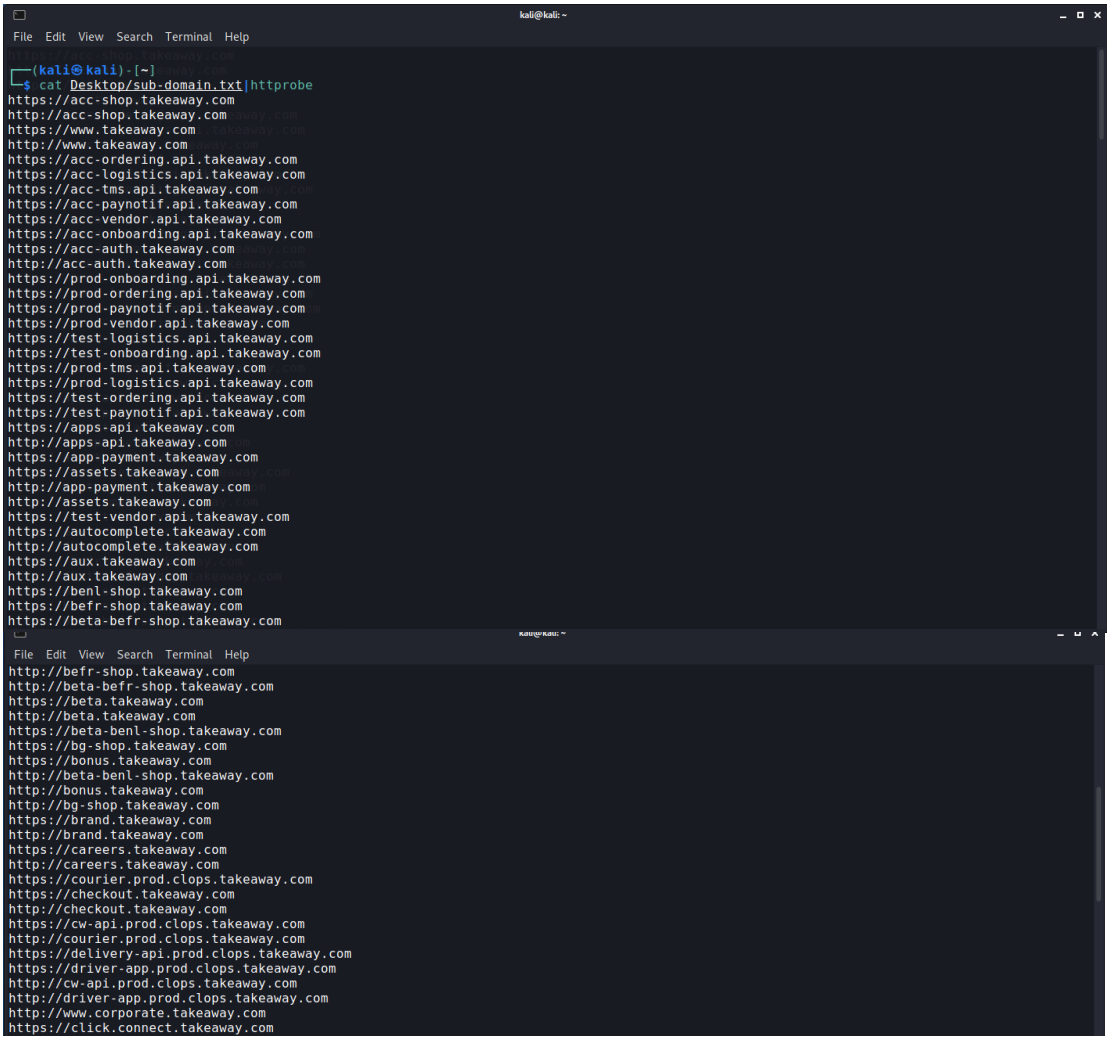
Although we found the sub-domains through the sublist3r tool, not all the found sub-domains are alive/active. So, we need to use another tool to find what sub-domains are alive from the list we have found.

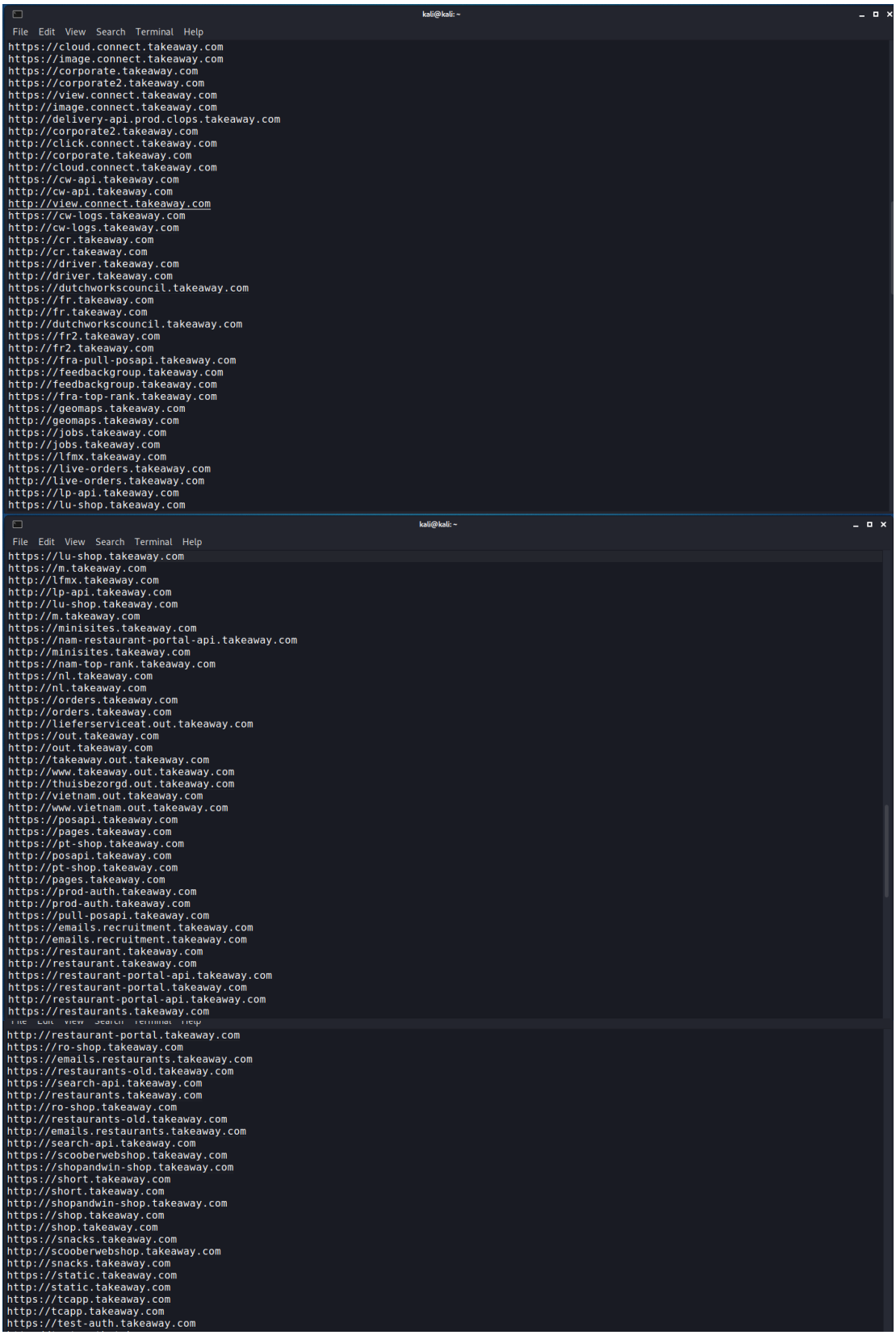
Discovery of alive sub-domains

To perform the discovery, we are going to use the tool called Httprobe. You can use the following command to start and perform the discovery.

```
cat Desktop/sub-domain.txt | httprobe
```

```
(cat <path to the file which contains the sub-domains> | httprobe)
```

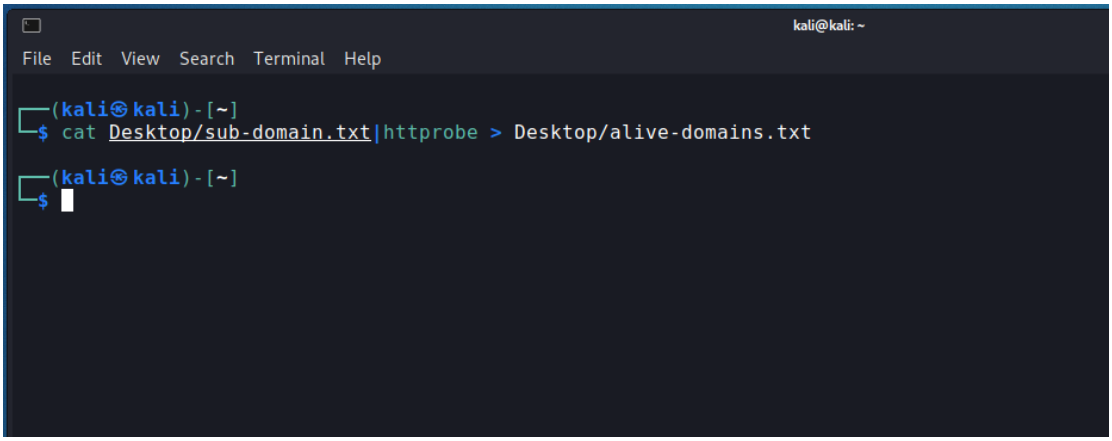



kali@kali: ~
File Edit View Search Terminal Help
https://cloud.connect.takeaway.com
https://image.connect.takeaway.com
https://corporate.takeaway.com
https://corporate2.takeaway.com
https://view.connect.takeaway.com
http://image.connect.takeaway.com
http://delivery-api.prod.clops.takeaway.com
http://corporate2.takeaway.com
http://click.connect.takeaway.com
http://corporate.takeaway.com
http://cloud.connect.takeaway.com
https://cw-api.takeaway.com
http://cw-api.takeaway.com
http://view.connect.takeaway.com
https://cw-logs.takeaway.com
http://cw-logs.takeaway.com
https://cr.takeaway.com
http://cr.takeaway.com
https://driver.takeaway.com
http://driver.takeaway.com
https://dutchworkscouncil.takeaway.com
https://fr.takeaway.com
http://fr.takeaway.com
http://dutchworkscouncil.takeaway.com
https://fr2.takeaway.com
http://fr2.takeaway.com
https://fra-pull-posapi.takeaway.com
https://feedbackgroup.takeaway.com
http://feedbackgroup.takeaway.com
https://fra-top-rank.takeaway.com
https://geomaps.takeaway.com
http://geomaps.takeaway.com
https://jobs.takeaway.com
http://jobs.takeaway.com
https://lfmx.takeaway.com
https://live-orders.takeaway.com
http://live-orders.takeaway.com
https://lp-api.takeaway.com
https://lu-shop.takeaway.com
kali@kali: ~
File Edit View Search Terminal Help
https://lu-shop.takeaway.com
https://m.takeaway.com
http://lfmx.takeaway.com
http://lp-api.takeaway.com
http://lu-shop.takeaway.com
http://m.takeaway.com
https://minisites.takeaway.com
https://nam-restaurant-portal-api.takeaway.com
http://minisites.takeaway.com
https://nam-top-rank.takeaway.com
https://nl.takeaway.com
http://nl.takeaway.com
https://orders.takeaway.com
http://orders.takeaway.com
http://lieferserviceat.out.takeaway.com
https://out.takeaway.com
http://out.takeaway.com
http://takeaway.out.takeaway.com
http://www.takeaway.out.takeaway.com
http://thuisbezorgd.out.takeaway.com
http://vietnam.out.takeaway.com
http://www.vietnam.out.takeaway.com
https://posapi.takeaway.com
https://pages.takeaway.com
https://pt-shop.takeaway.com
http://posapi.takeaway.com
http://pt-shop.takeaway.com
http://pages.takeaway.com
https://prod-auth.takeaway.com
http://prod-auth.takeaway.com
https://pull-posapi.takeaway.com
https://emails.recruitment.takeaway.com
http://emails.recruitment.takeaway.com
https://restaurant.takeaway.com
http://restaurant.takeaway.com
https://restaurant-portal-api.takeaway.com
https://restaurant-portal.takeaway.com
http://restaurant-portal-api.takeaway.com
https://restaurants.takeaway.com
http://restaurant-portal.takeaway.com
https://ro-shop.takeaway.com
https://emails.restaurants.takeaway.com
https://restaurants-old.takeaway.com
https://search-api.takeaway.com
http://restaurants.takeaway.com
http://ro-shop.takeaway.com
http://restaurants-old.takeaway.com
http://emails.restaurants.takeaway.com
http://search-api.takeaway.com
https://scooberwebshop.takeaway.com
https://shopandwin-shop.takeaway.com
https://short.takeaway.com
http://short.takeaway.com
http://shopandwin-shop.takeaway.com
https://shop.takeaway.com
http://shop.takeaway.com
https://snacks.takeaway.com
http://scooberwebshop.takeaway.com
http://snacks.takeaway.com
https://static.takeaway.com
http://static.takeaway.com
https://tcapp.takeaway.com
http://tcapp.takeaway.com
https://test-auth.takeaway.com

As there are many found links displayed, it is better to store them to a text file. So, you can access it later for clarifications. To store the results to a text file, you can use the following command on the terminal.

```
cat Desktop/sub-domain.txt | httprobe > Desktop/alive-domains.txt
```

(cat <path to the file which contains the sub-domains> | httprobe > <path the new file to be created with alive sub-domains>)



This tool will display the URLs against both http and https protocols. So, the same URL might be repeated and printed to the display or the text file. This might double the count of the URLs and will waste time to go through it from top to bottom.

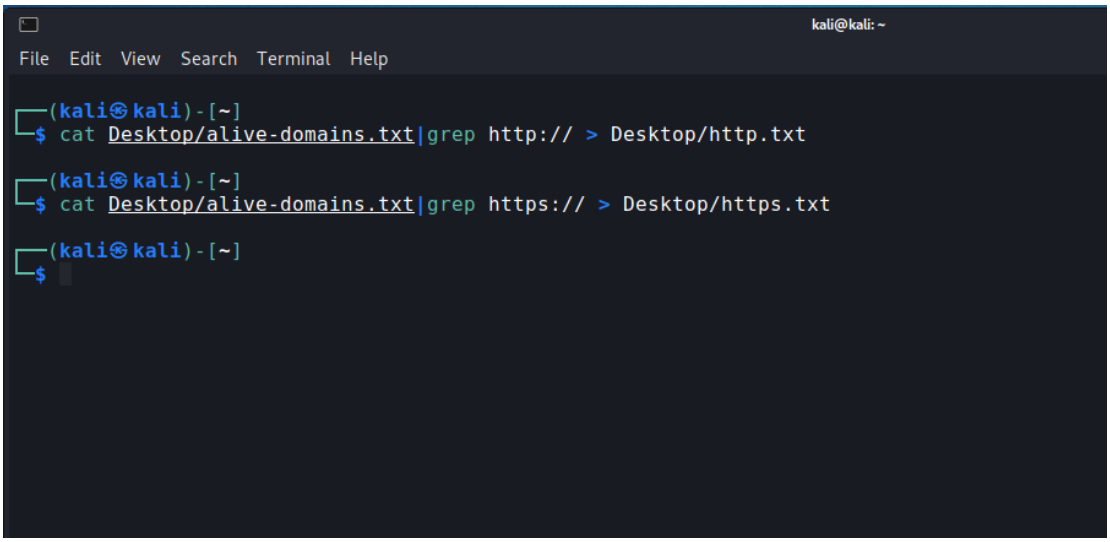
You can use the below mentioned commands to store http and https URLs in two different files.

To get the http URL list

```
cat Desktop/alive-subdomains.txt | grep http:// > Desktop/http.txt
```

To get the https URL list

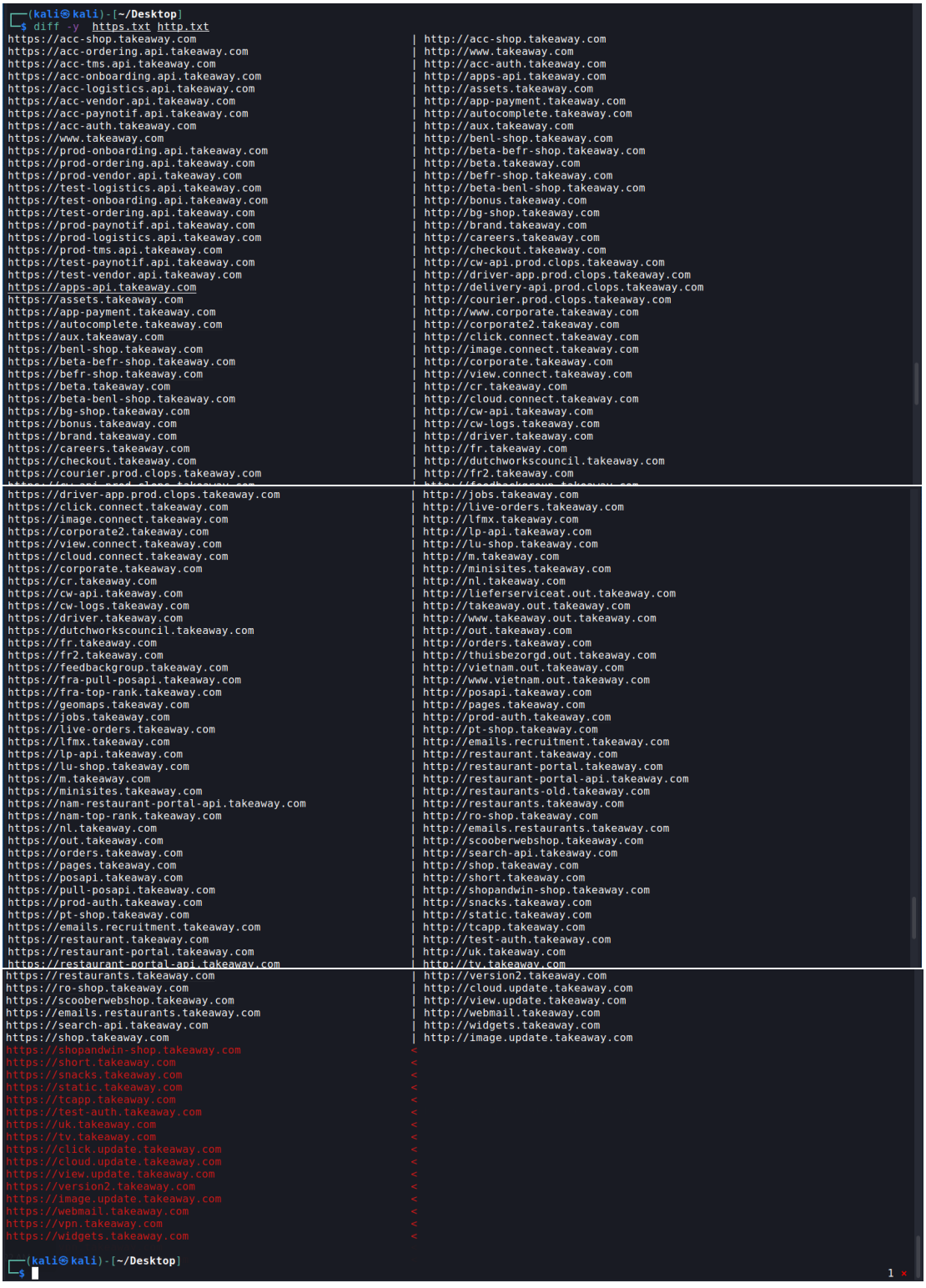
```
cat Desktop/alive-subdomains.txt | grep https:// > Desktop/https.txt
```



To look for the different values in both files created (http.txt and https.txt), we can use simple too called **diff.** Following command will be used to differentiate both files and identify which sites use both https and which site does not. Then we can extract and discard the duplicated values with http/https.

```
diff -y https.txt http.txt
```

(-y = used to display the content of two files side by side for a better view)

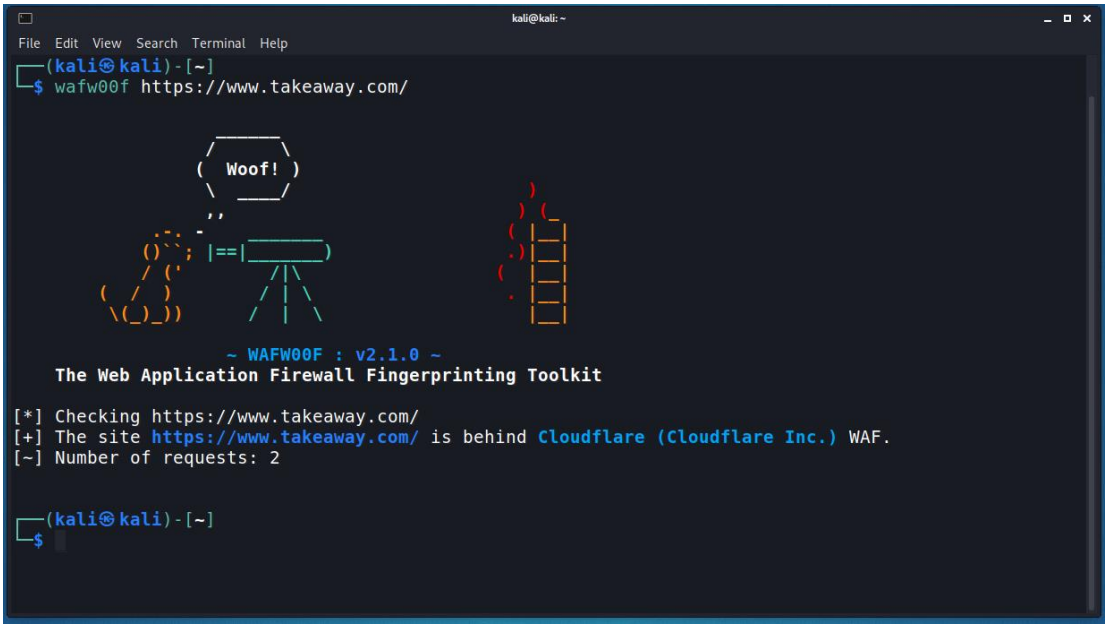

┌──(kali㉿kali)-[~/Desktop]
└─$ diff -y https.txt http.txt
https://acc-shop.takeaway.com | http://acc-shop.takeaway.com
https://acc-ordering.api.takeaway.com | http://www.takeaway.com
https://acc-tms.api.takeaway.com | http://acc-auth.takeaway.com
https://acc-onboarding.api.takeaway.com | http://apps-api.takeaway.com
https://acc-logistics.api.takeaway.com | http://assets.takeaway.com
https://acc-vendor.api.takeaway.com | http://app-payment.takeaway.com
https://acc-paynotif.api.takeaway.com | http://autocomplete.takeaway.com
https://acc-auth.takeaway.com | http://aux.takeaway.com
https://www.takeaway.com | http://benl-shop.takeaway.com
https://prod-onboarding.api.takeaway.com | http://beta-befr-shop.takeaway.com
https://prod-ordering.api.takeaway.com | http://beta.takeaway.com
https://prod-vendor.api.takeaway.com | http://befr-shop.takeaway.com
https://test-logistics.api.takeaway.com | http://beta-benl-shop.takeaway.com
https://test-onboarding.api.takeaway.com | http://bonus.takeaway.com
https://test-ordering.api.takeaway.com | http://bg-shop.takeaway.com
https://prod-paynotif.api.takeaway.com | http://brand.takeaway.com
https://prod-logistics.api.takeaway.com | http://careers.takeaway.com
https://prod-tms.api.takeaway.com | http://checkout.takeaway.com
https://test-paynotif.api.takeaway.com | http://cw-api.prod.clops.takeaway.com
https://test-vendor.api.takeaway.com | http://driver-app.prod.clops.takeaway.com
https://apps-api.takeaway.com | http://delivery-api.prod.clops.takeaway.com
https://assets.takeaway.com | http://courier.prod.clops.takeaway.com
https://app-payment.takeaway.com | http://www.corporate.takeaway.com
https://autocomplete.takeaway.com | http://corporate2.takeaway.com
https://aux.takeaway.com | http://click.connect.takeaway.com
https://benl-shop.takeaway.com | http://image.connect.takeaway.com
https://beta-befr-shop.takeaway.com | http://corporate.takeaway.com
https://befr-shop.takeaway.com | http://view.connect.takeaway.com
https://beta.takeaway.com | http://cr.takeaway.com
https://beta-benl-shop.takeaway.com | http://cloud.connect.takeaway.com
https://bg-shop.takeaway.com | http://cw-api.takeaway.com
https://bonus.takeaway.com | http://cw-logs.takeaway.com
https://brand.takeaway.com | http://driver.takeaway.com
https://careers.takeaway.com | http://fr.takeaway.com
https://checkout.takeaway.com | http://dutchworkscouncil.takeaway.com
https://courier.prod.clops.takeaway.com | http://fr2.takeaway.com
https://driver-app.prod.clops.takeaway.com | http://jobs.takeaway.com
https://click.connect.takeaway.com | http://live-orders.takeaway.com
https://image.connect.takeaway.com | http://lfmx.takeaway.com
https://corporate2.takeaway.com | http://lp-api.takeaway.com
https://view.connect.takeaway.com | http://lu-shop.takeaway.com
https://cloud.connect.takeaway.com | http://m.takeaway.com
https://corporate.takeaway.com | http://minisites.takeaway.com
https://cr.takeaway.com | http://nl.takeaway.com
https://cw-api.takeaway.com | http://lieferserviceat.out.takeaway.com
https://cw-logs.takeaway.com | http://takeaway.out.takeaway.com
https://driver.takeaway.com | http://www.takeaway.out.takeaway.com
https://dutchworkscouncil.takeaway.com | http://out.takeaway.com
https://fr.takeaway.com | http://orders.takeaway.com
https://fr2.takeaway.com | http://thuisbezorgd.out.takeaway.com
https://feedbackgroup.takeaway.com | http://vietnam.out.takeaway.com
https://fra-pull-posapi.takeaway.com | http://www.vietnam.out.takeaway.com
https://fra-top-rank.takeaway.com | http://posapi.takeaway.com
https://geomaps.takeaway.com | http://pages.takeaway.com
https://jobs.takeaway.com | http://prod-auth.takeaway.com
https://live-orders.takeaway.com | http://pt-shop.takeaway.com
https://lfmx.takeaway.com | http://emails.recruitment.takeaway.com
https://lp-api.takeaway.com | http://restaurant.takeaway.com
https://lu-shop.takeaway.com | http://restaurant-portal.takeaway.com
https://m.takeaway.com | http://restaurant-portal-api.takeaway.com
https://minisites.takeaway.com | http://restaurants-old.takeaway.com
https://nam-restaurant-portal-api.takeaway.com | http://restaurants.takeaway.com
https://nam-top-rank.takeaway.com | http://ro-shop.takeaway.com
https://nl.takeaway.com | http://emails.restaurants.takeaway.com
https://out.takeaway.com | http://scooberwebshop.takeaway.com
https://orders.takeaway.com | http://search-api.takeaway.com
https://pages.takeaway.com | http://shop.takeaway.com
https://posapi.takeaway.com | http://short.takeaway.com
https://pull-posapi.takeaway.com | http://shopandwin-shop.takeaway.com
https://prod-auth.takeaway.com | http://snacks.takeaway.com
https://pt-shop.takeaway.com | http://static.takeaway.com
https://emails.recruitment.takeaway.com | http://tcapp.takeaway.com
https://restaurant.takeaway.com | http://test-auth.takeaway.com
https://restaurant-portal.takeaway.com | http://uk.takeaway.com
https://restaurant-portal-api.takeaway.com | http://tv.takeaway.com
https://restaurants.takeaway.com | http://version2.takeaway.com
https://ro-shop.takeaway.com | http://cloud.update.takeaway.com
https://scooberwebshop.takeaway.com | http://view.update.takeaway.com
https://emails.restaurants.takeaway.com | http://webmail.takeaway.com
https://search-api.takeaway.com | http://widgets.takeaway.com
https://shop.takeaway.com | http://image.update.takeaway.com
https://shopandwin-shop.takeaway.com <
https://short.takeaway.com <
https://snacks.takeaway.com <
https://static.takeaway.com <
https://tcapp.takeaway.com <
https://test-auth.takeaway.com <
https://uk.takeaway.com <
https://tv.takeaway.com <
https://click.update.takeaway.com <
https://cloud.update.takeaway.com <
https://view.update.takeaway.com <
https://version2.takeaway.com <
https://image.update.takeaway.com <
https://webmail.takeaway.com <
https://vpn.takeaway.com <
https://widgets.takeaway.com <
┌──(kali㉿kali)-[~/Desktop]
└─$
1 ×

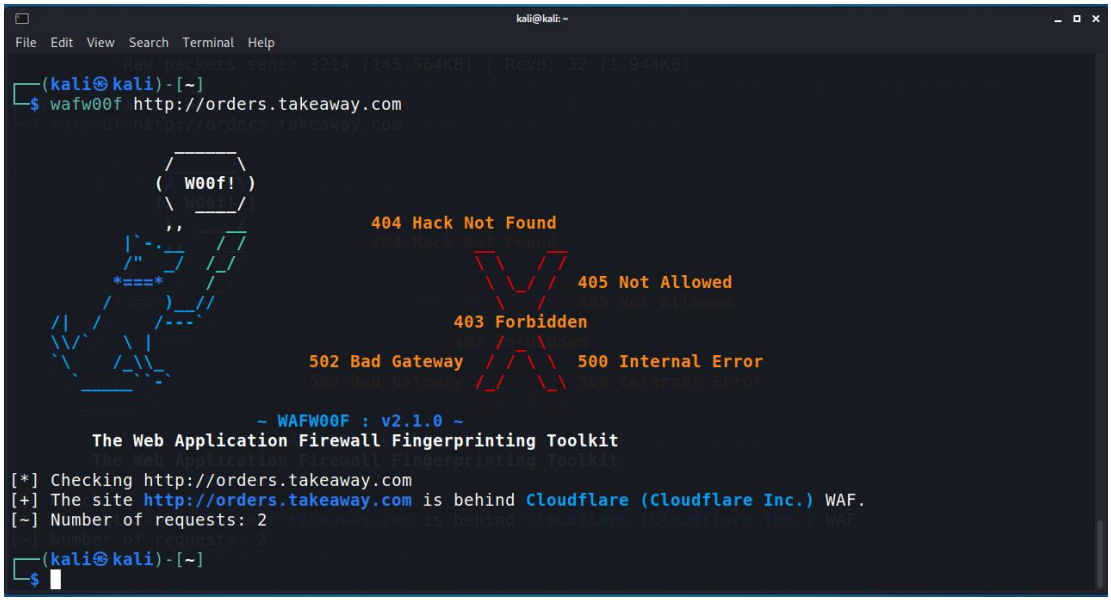
Searching for Web Application Firewalls (WAF)

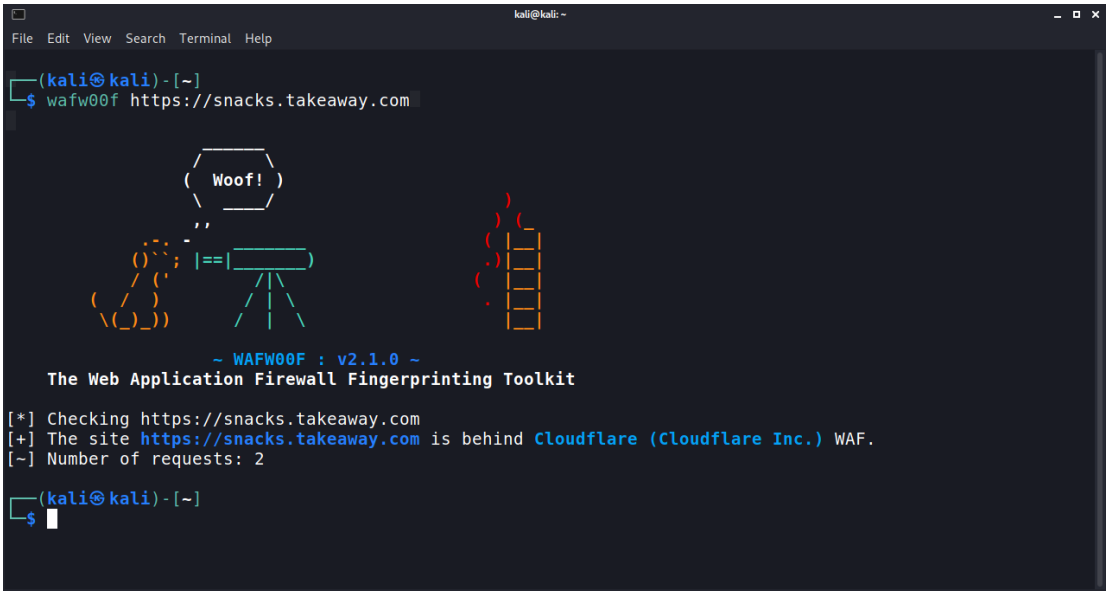
To check whether the website is hosted behind a web application firewall, we can use the tool called WAFW00F. A WAF will block cross site scripting, SQL injections and malicious attacks that is performed towards the website. The following command can be used to start WAFW00f and check for any WAF is present in front of the website.

```
wafw00f https://www.takeaway.com/
```

(when the word wafw00f is typed, make sure to use two ZEROs in between the fourth letter 'w' and last letter 'f')







We can observe the results of the scan from the above image. The website https://www.takeaway.com/ is hosted behind a web application firewall. The firewall provider is Cloudflare Inc.

Scanning for software versions, open ports and running devices

Network Mapper (NMAP)

Through this tool we are going to find out what ports are open on the webserver, find what application versions are currently running and to find out the Operating System and version. The below mentioned command can be used to start and proceed with NMAP.

```
sudo nmap -sS -sV -A -v -oN Desktop/nmap-output.txt
takeaway.com
```

-sS = Stealth scanning (SYN/ACK)

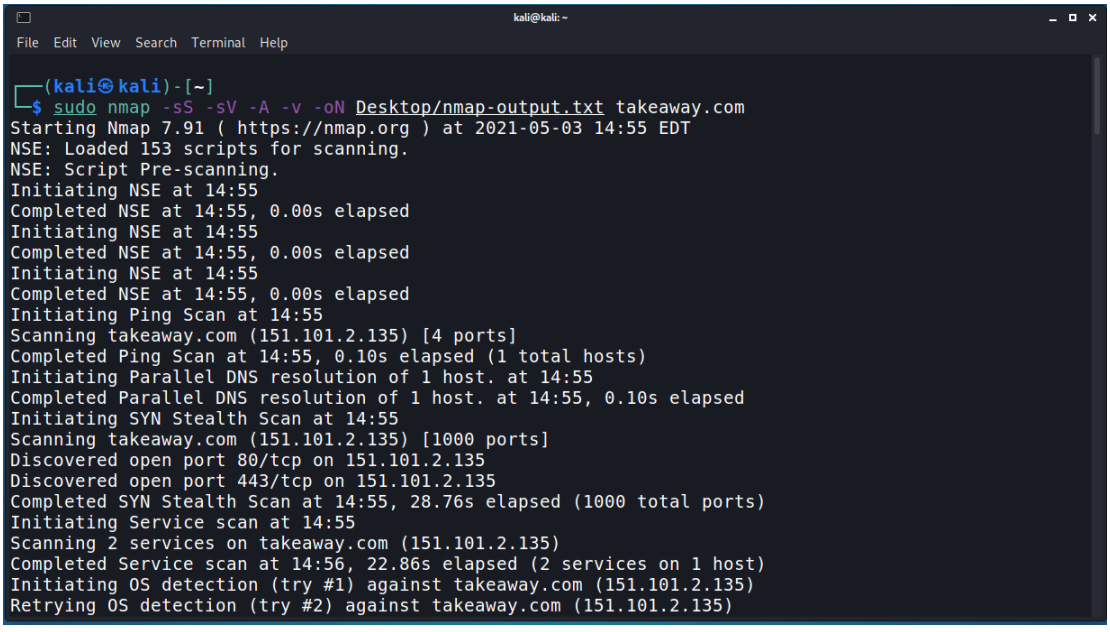
-sV = Software and OS version scanning

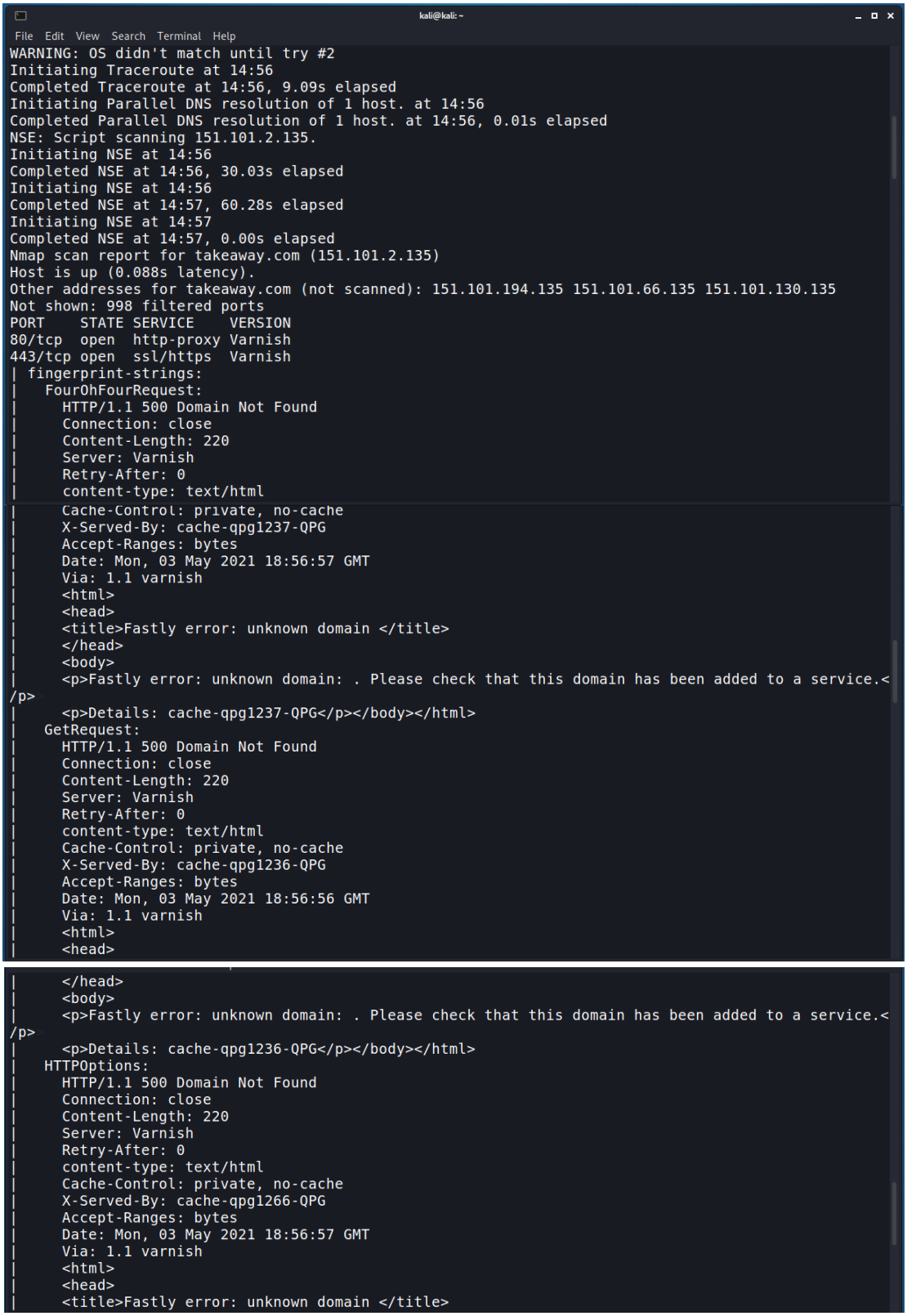
-A = Scan in aggressive mode

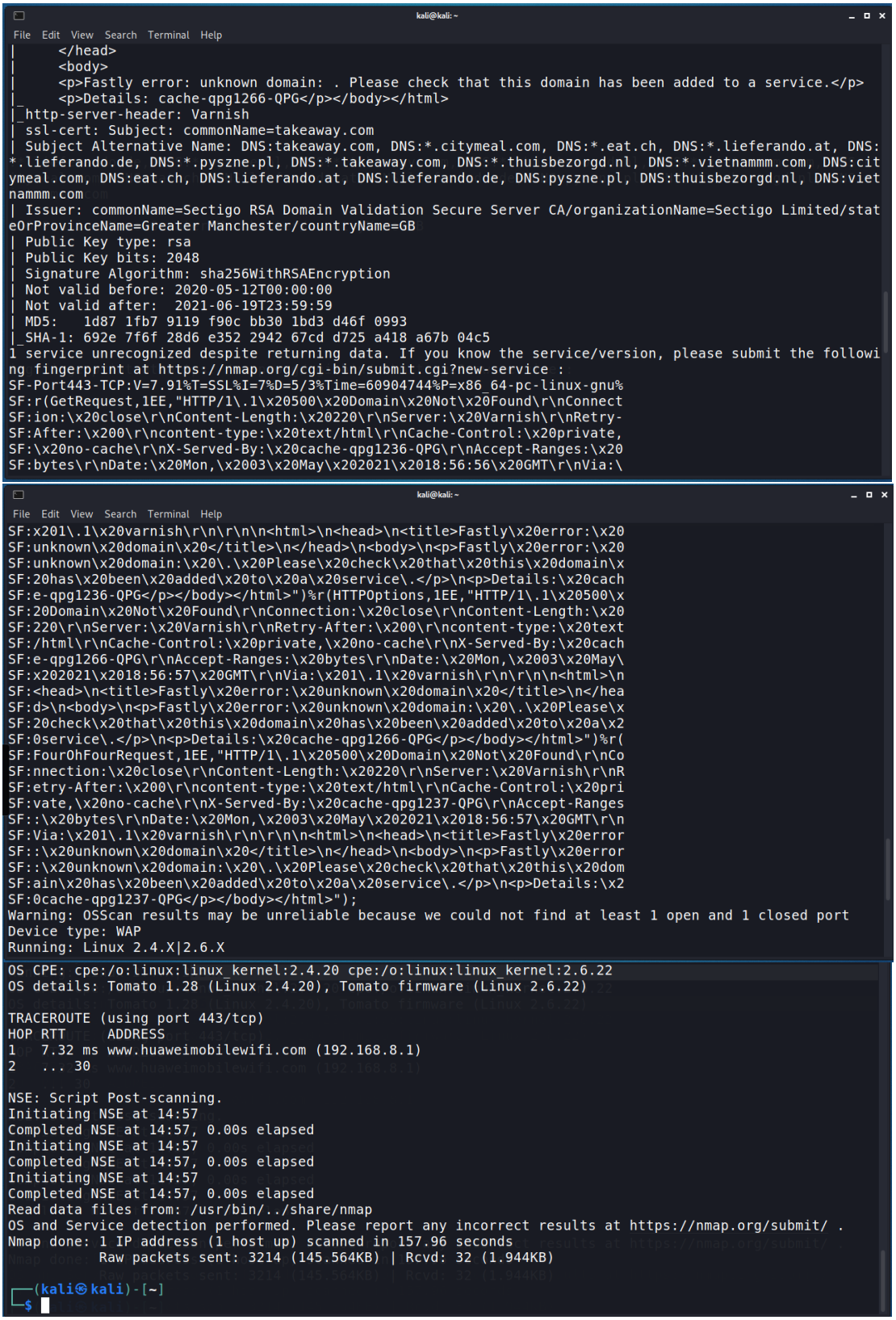
-v = To display what is happening in real time

-oN = Saving the results to an output text file

The results from the scan of **takeaway.com** is presented below with the screen snaps.




kali@kali: ~
File Edit View Search Terminal Help
WARNING: OS didn't match until try #2
Initiating Traceroute at 14:56
Completed Traceroute at 14:56, 9.09s elapsed
Initiating Parallel DNS resolution of 1 host. at 14:56
Completed Parallel DNS resolution of 1 host. at 14:56, 0.01s elapsed
NSE: Script scanning 151.101.2.135.
Initiating NSE at 14:56
Completed NSE at 14:56, 30.03s elapsed
Initiating NSE at 14:56
Completed NSE at 14:57, 60.28s elapsed
Initiating NSE at 14:57
Completed NSE at 14:57, 0.00s elapsed
Nmap scan report for takeaway.com (151.101.2.135)
Host is up (0.088s latency).
Other addresses for takeaway.com (not scanned): 151.101.194.135 151.101.66.135 151.101.130.135
Not shown: 998 filtered ports
PORT STATE SERVICE VERSION
80/tcp open http-proxy Varnish
443/tcp open ssl/https Varnish
| fingerprint-strings:
| FourOhFourRequest:
| HTTP/1.1 500 Domain Not Found
| Connection: close
| Content-Length: 220
| Server: Varnish
| Retry-After: 0
| content-type: text/html
| Cache-Control: private, no-cache
| X-Served-By: cache-qpg1237-QPG
| Accept-Ranges: bytes
| Date: Mon, 03 May 2021 18:56:57 GMT
| Via: 1.1 varnish
| <html>
| <head>
| <title>Fastly error: unknown domain </title>
| </head>
| <body>
| Fastly error: unknown domain: . Please check that this domain has been added to a service.<
/p>
| Details: cache-qpg1237-QPG </body></html>
| GetRequest:
| HTTP/1.1 500 Domain Not Found
| Connection: close
| Content-Length: 220
| Server: Varnish
| Retry-After: 0
| content-type: text/html
| Cache-Control: private, no-cache
| X-Served-By: cache-qpg1236-QPG
| Accept-Ranges: bytes
| Date: Mon, 03 May 2021 18:56:56 GMT
| Via: 1.1 varnish
| <html>
| <head>
| </head>
| <body>
| Fastly error: unknown domain: . Please check that this domain has been added to a service.<
/p>
| Details: cache-qpg1236-QPG </body></html>
| HTTPOptions:
| HTTP/1.1 500 Domain Not Found
| Connection: close
| Content-Length: 220
| Server: Varnish
| Retry-After: 0
| content-type: text/html
| Cache-Control: private, no-cache
| X-Served-By: cache-qpg1266-QPG
| Accept-Ranges: bytes
| Date: Mon, 03 May 2021 18:56:57 GMT
| Via: 1.1 varnish
| <html>
| <head>
| <title>Fastly error: unknown domain </title>


kali@kali: ~
File Edit View Search Terminal Help
| </head>
| <body>
| Fastly error: unknown domain: . Please check that this domain has been added to a service.
|_ Details: cache-qpg1266-QPG </body></html>
|_http-server-header: Varnish
| ssl-cert: Subject: commonName=takeaway.com
| Subject Alternative Name: DNS:takeaway.com, DNS:*.citymeal.com, DNS:*.eat.ch, DNS:*.lieferando.at, DNS:*.lieferando.de, DNS:*.pyszne.pl, DNS:*.takeaway.com, DNS:*.thuisbezorgd.nl, DNS:*.vietnammm.com, DNS:citymeal.com, DNS:eat.ch, DNS:lieferando.at, DNS:lieferando.de, DNS:pyszne.pl, DNS:thuisbezorgd.nl, DNS:vietnammm.com
| Issuer: commonName=Sectigo RSA Domain Validation Secure Server CA/organizationName=Sectigo Limited/stateOrProvinceName=Greater Manchester/countryName=GB
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2020-05-12T00:00:00
| Not valid after: 2021-06-19T23:59:59
| MD5: 1d87 1fb7 9119 f90c bb30 1bd3 d46f 0993
|_SHA-1: 692e 7f6f 28d6 e352 2942 67cd d725 a418 a67b 04c5
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-Port443-TCP:V=7.91%T=SSL%I=7%D=5/3%Time=60904744%P=x86_64-pc-linux-gnu%
SF:r(GetRequest,1EE,"HTTP/1\.1\x20500\x20Domain\x20Not\x20Found\r\nConnect
SF:ion:\x20close\r\nContent-Length:\x20220\r\nServer:\x20Varnish\r\nRetry-
SF:After:\x200\r\ncontent-type:\x20text/html\r\nCache-Control:\x20private,
SF:\x20no-cache\r\nX-Served-By:\x20cache-qpg1236-QPG\r\nAccept-Ranges:\x20
SF:bytes\r\nDate:\x20Mon,\x2003\x20May\x202021\x2018:56:56\x20GMT\r\nVia:\
kali@kali: ~
File Edit View Search Terminal Help
SF:x201\.1\x20varnish\r\n\r\n\n<html>\n<head>\n<title>Fastly\x20error:\x20
SF:unknown\x20domain\x20</title>\n</head>\n<body>\n Fastly\x20error:\x20
SF:unknown\x20domain:\x20\.\x20Please\x20check\x20that\x20this\x20domain\x
SF:20has\x20been\x20added\x20to\x20a\x20service\. \n Details:\x20cach
SF:e-qpg1236-QPG </body></html>")%r(HTTPOptions,1EE,"HTTP/1\.1\x20500\x
SF:20Domain\x20Not\x20Found\r\nConnection:\x20close\r\nContent-Length:\x20
SF:220\r\nServer:\x20Varnish\r\nRetry-After:\x200\r\ncontent-type:\x20text
SF:/html\r\nCache-Control:\x20private,\x20no-cache\r\nX-Served-By:\x20cach
SF:e-qpg1266-QPG\r\nAccept-Ranges:\x20bytes\r\nDate:\x20Mon,\x2003\x20May\
SF:x202021\x2018:56:57\x20GMT\r\nVia:\x201\.1\x20varnish\r\n\r\n\n<html>\n
SF:<head>\n<title>Fastly\x20error:\x20unknown\x20domain\x20</title>\n</hea
SF:d>\n<body>\n Fastly\x20error:\x20unknown\x20domain:\x20\.\x20Please\x
SF:20check\x20that\x20this\x20domain\x20has\x20been\x20added\x20to\x20a\x2
SF:0service\. \n Details:\x20cache-qpg1266-QPG </body></html>")%r(
SF:FourOhFourRequest,1EE,"HTTP/1\.1\x20500\x20Domain\x20Not\x20Found\r\nCo
SF:nnection:\x20close\r\nContent-Length:\x20220\r\nServer:\x20Varnish\r\nR
SF:etry-After:\x200\r\ncontent-type:\x20text/html\r\nCache-Control:\x20pri
SF:vate,\x20no-cache\r\nX-Served-By:\x20cache-qpg1237-QPG\r\nAccept-Ranges
SF::\x20bytes\r\nDate:\x20Mon,\x2003\x20May\x202021\x2018:56:57\x20GMT\r\n
SF:Via:\x201\.1\x20varnish\r\n\r\n\n<html>\n<head>\n<title>Fastly\x20error
SF::\x20unknown\x20domain\x20</title>\n</head>\n<body>\n Fastly\x20error
SF::\x20unknown\x20domain:\x20\.\x20Please\x20check\x20that\x20this\x20dom
SF:ain\x20has\x20been\x20added\x20to\x20a\x20service\. \n Details:\x2
SF:0cache-qpg1237-QPG </body></html>");
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: WAP
Running: Linux 2.4.X|2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.4.20 cpe:/o:linux:linux_kernel:2.6.22
OS details: Tomato 1.28 (Linux 2.4.20), Tomato firmware (Linux 2.6.22)
TRACEROUTE (using port 443/tcp)
HOP RTT ADDRESS
1 7.32 ms www.huaweimobilewifi.com (192.168.8.1)
2 ... 30
NSE: Script Post-scanning.
Initiating NSE at 14:57
Completed NSE at 14:57, 0.00s elapsed
Initiating NSE at 14:57
Completed NSE at 14:57, 0.00s elapsed
Initiating NSE at 14:57
Completed NSE at 14:57, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
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 157.96 seconds
Raw packets sent: 3214 (145.564KB) | Rcvd: 32 (1.944KB)
(kali㉿kali)-[~]
$

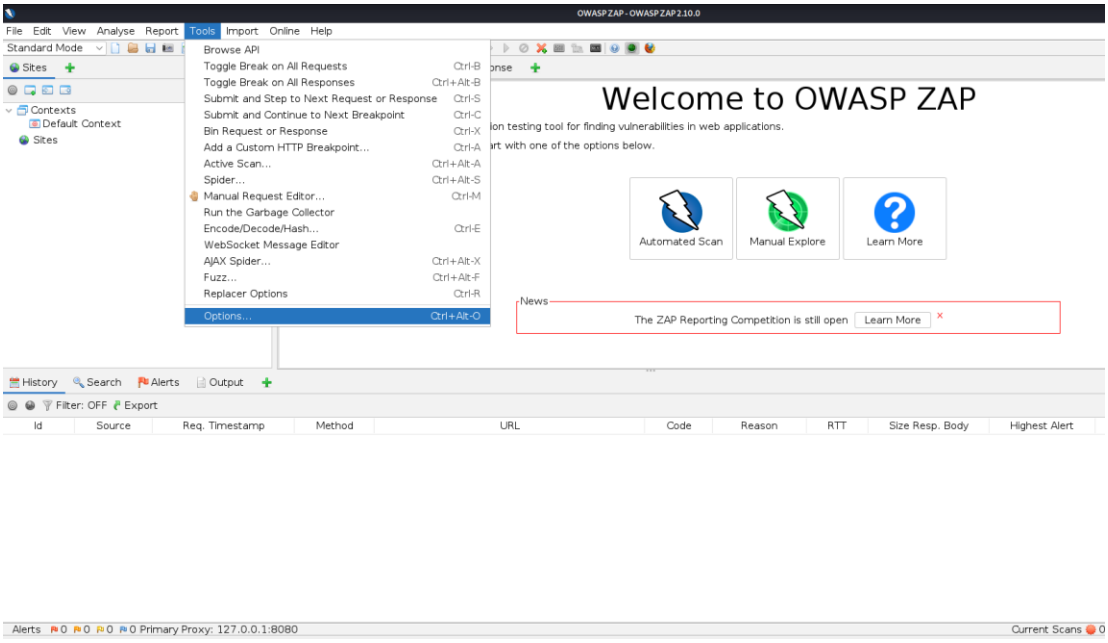
Finding the structure of the website (Sitemap)

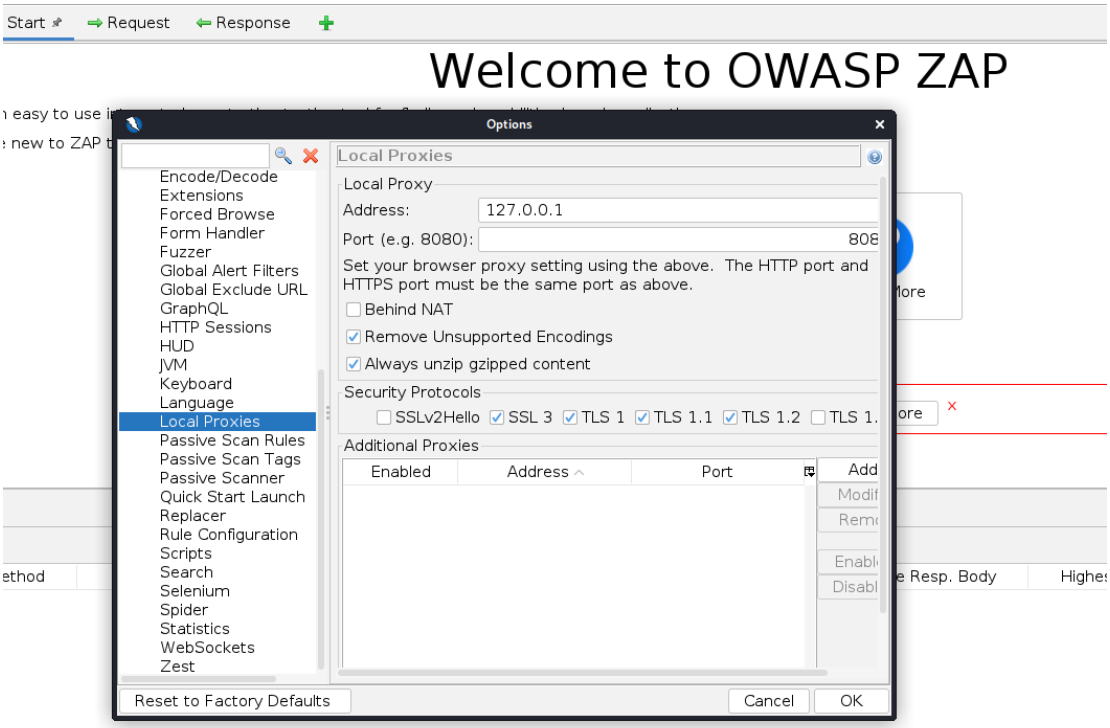
To find the structure of the website, we will use the tool called OWASP Zed Attack Proxy (ZAP). For this to proceed, we must enable a manual proxy setting on the web browser and the ZAP itself. I used Firefox browser to perform this. The same manual proxy configuration that I set up in the browser should be configured in ZAP also.

To configure the ZAP local proxy settings, use the following steps.

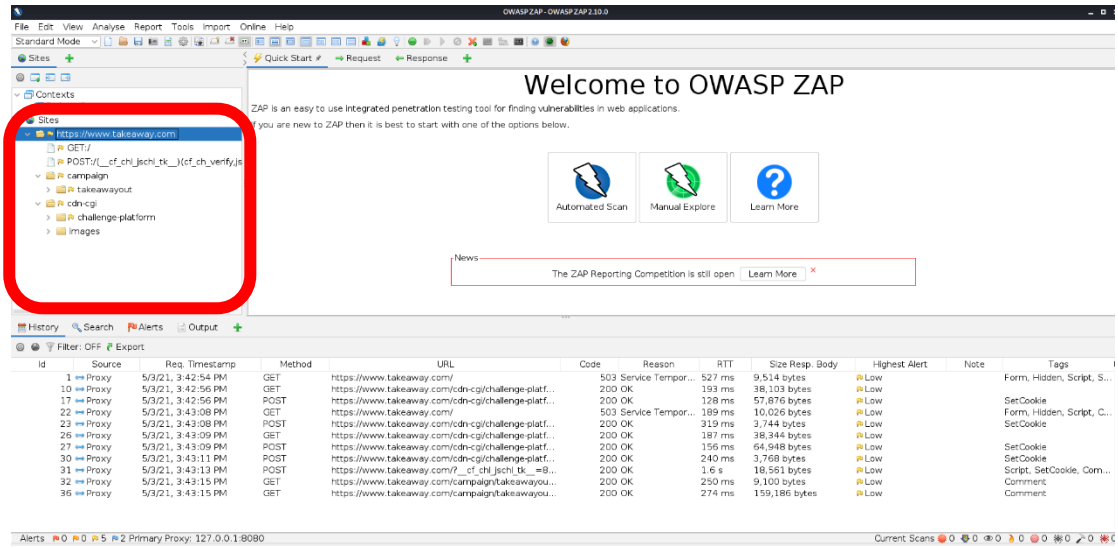
Tools > Options > Local Proxies > Add the address (127.0.0.1) > Add the port (8080) > press OK button

(Here, I used the loopback address(127.0.0.1). We can use any address, but we must use the same address and the port in both the web browser and ZAP.)



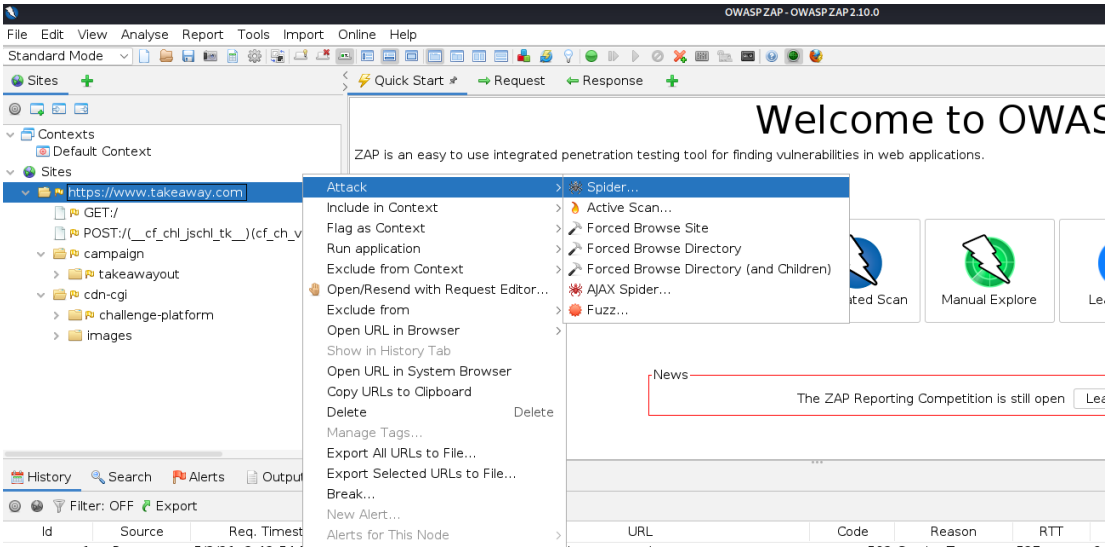


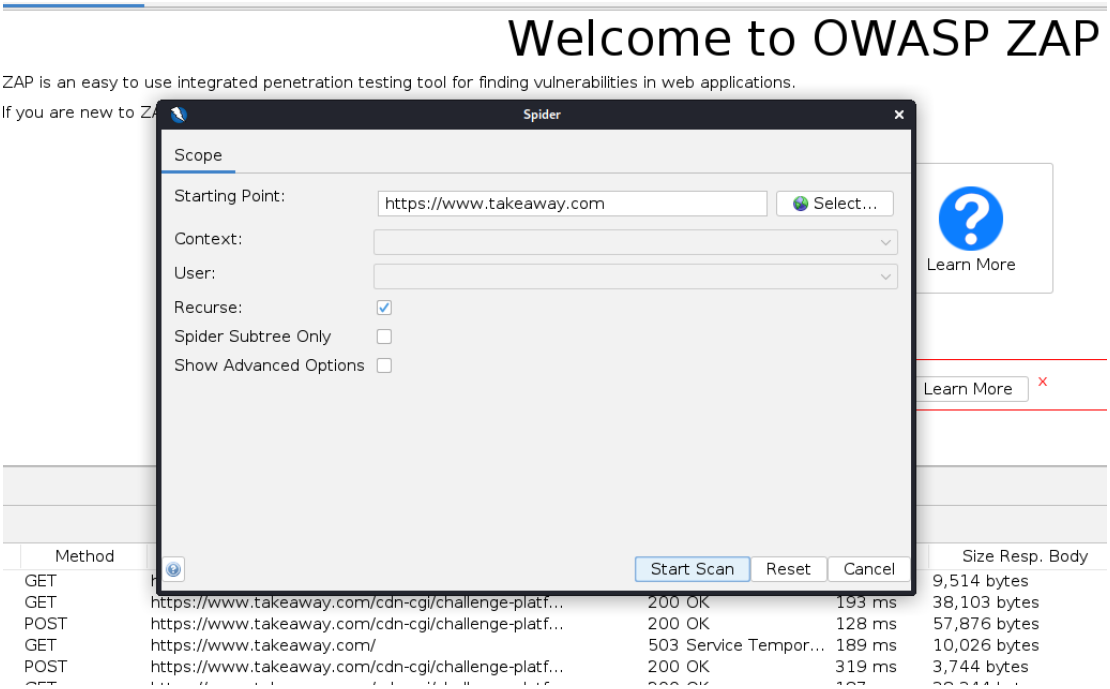
Now, ZAP will log and generate a simple sitemap of the URL we enter on the browser.



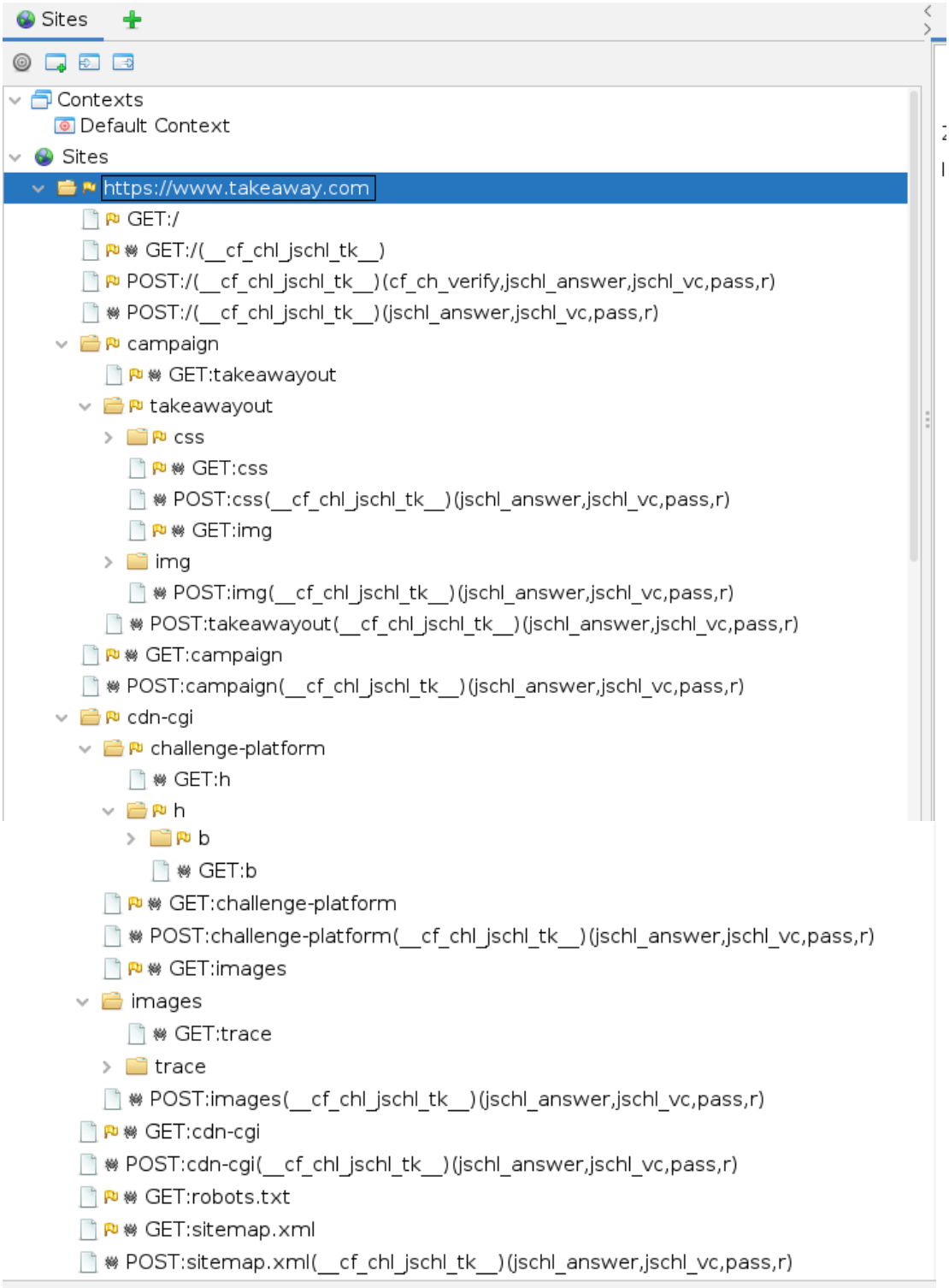
To obtain more depth version of the site map, we can use a web crawler (spider). This spider will crawl through all the links that it comes in a recursive method. ZAP has a built-in spider. The use the spider, we can use the following steps.

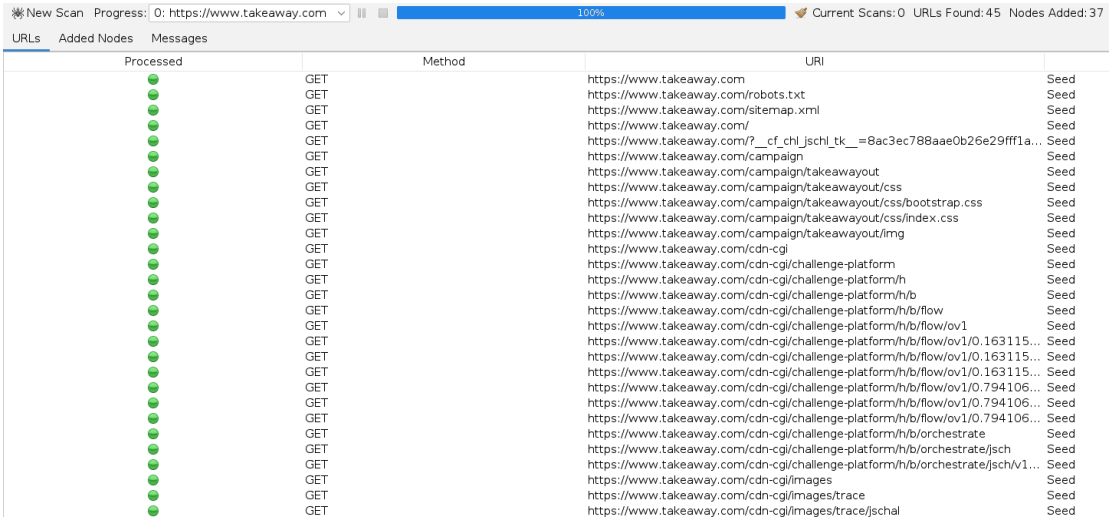
Right click the domain name > press Attack > press spider > press Start scan



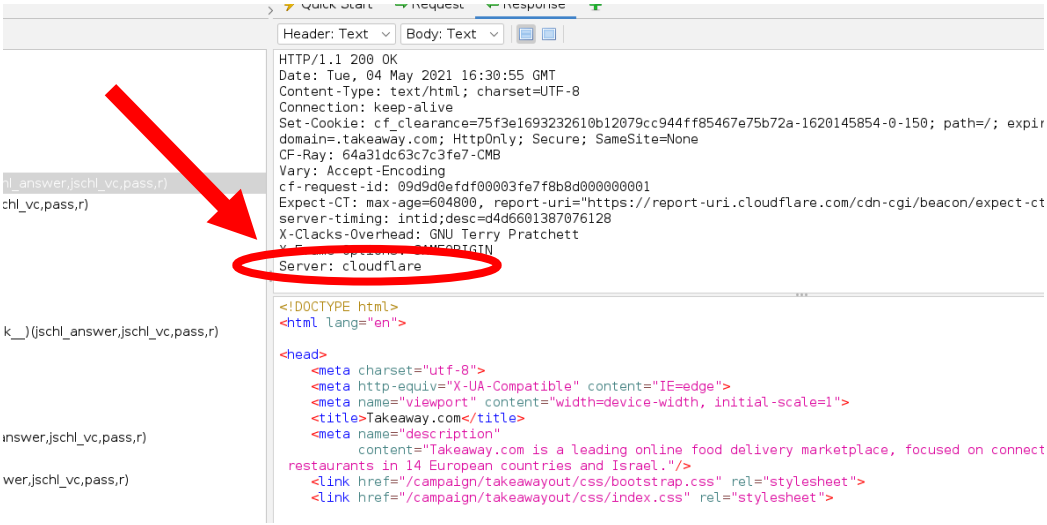


Now, it is visible that more parts have been added to the sitemap after the spider finished running. We can observe it from the images provided below.



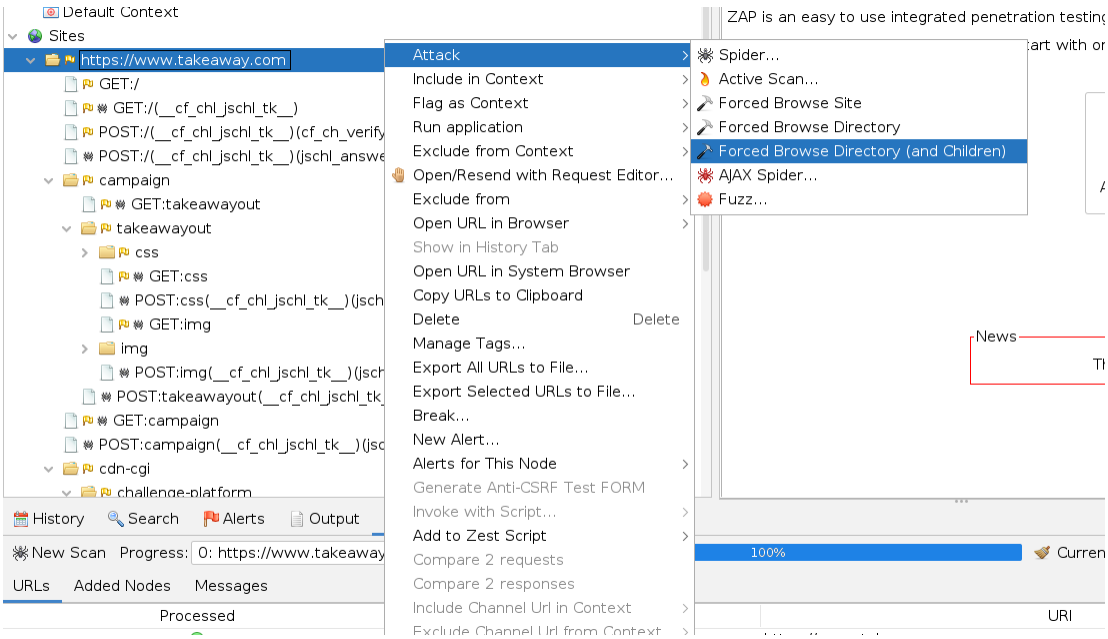


We can identify that the server name this website is using is Cloudflare. We can observe it from below shown image from a response we have received from the website.

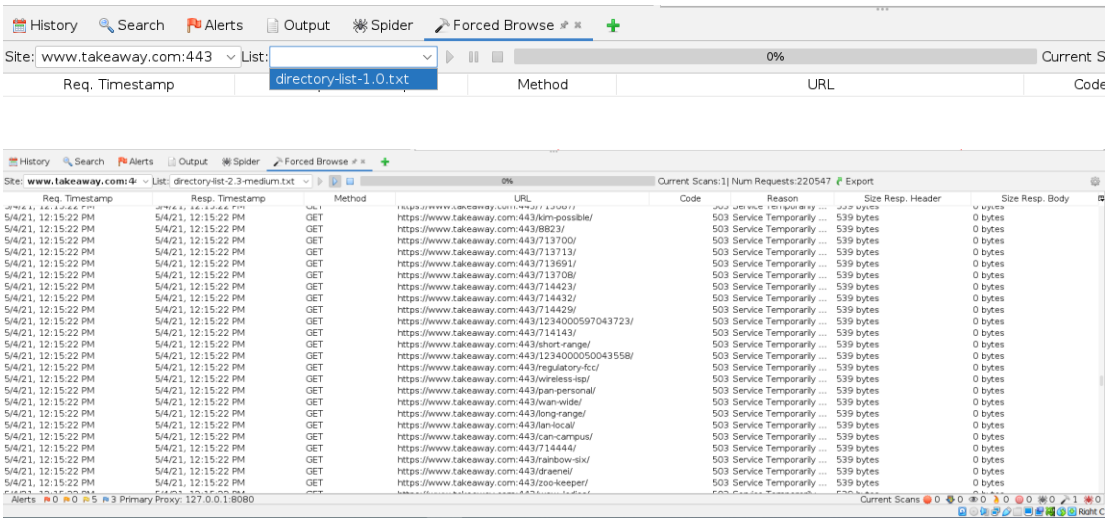


Now, to find hidden directories and files, we can use Forced Browse Directory Attack(and Children). This will use a directory list file to brute force and find hidden items. You can follow the following path to use this option.

Right click the domain name > press Attack > press Forced Browse Directory(and Children)



Choosing the directory list text file to use in the brute force.



As we can observe from the above image, even though we tried to do a directory and hidden file brute force attack, all the responses return the code as 503 (Service temporarily unavailable). So, I had to terminate the attack manually. This means that the WAF is blocking the requests made by ZED.

Scanning for Vulnerabilities

To scan the vulnerabilities of the website, I use the tool called Netsparker Professional.

1. **https://www.takeaway.com/**

Using components with known vulnerabilities

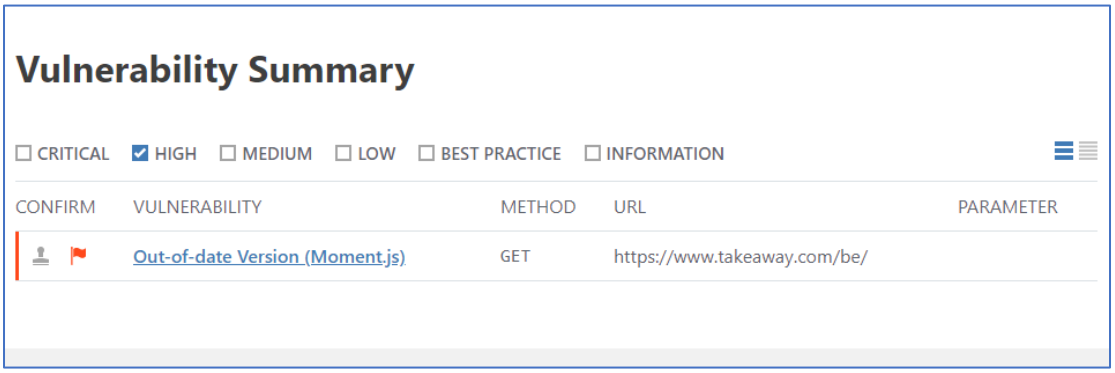
a. Using outdated version of Moment.js

- Severity :- High

This vulnerability will allow an attacker to perform a Denial of Service (DoS) attack. Successfully exploiting this vulnerability will cause a problem to the availability of the services provided by the website to its legitimate users.

- Out-dated version :- 2.18.1
- Latest version available :- 2.29.1

Proof of Vulnerability



Impacts due to this vulnerability

- It may be vulnerable to attacks due to being an older version of this software.
- Prone to a regular expression denial of service via a crafted date string.

Solution to the vulnerability

- Upgrade Moment.js installation to the latest stable version

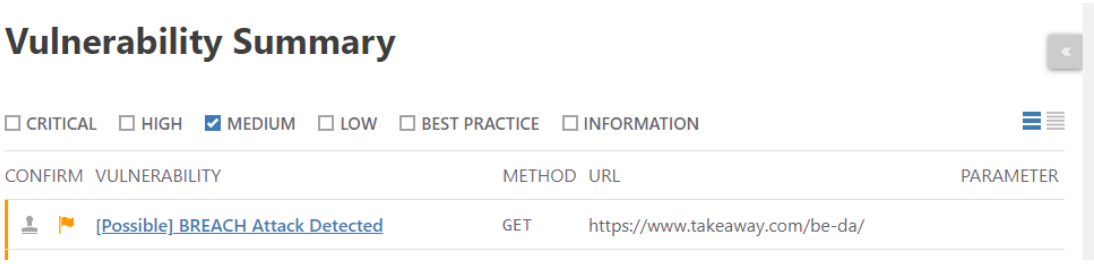
b. [Possible] BREACH attack

- **Severity :-** Medium

This is an attack which is used to obtain information regarding the secrets that are used in encrypted and compressed responses by means of an attack called oracle attack. What this simply does is that it will send few numbers of requests to the web server that is vulnerable to this attack and analyze the responses received to those requests and will try and obtain to identify secrets which are undisclosed.

BREACH (Browser Reconnaissance & Exfiltration via Adaptive Compression of Hypertext)

Proof of Vulnerability



Impact

Even if you use an SSL/TLS protected connection, an attacker can still view the victim's encrypted traffic and cause the victim to send HTTP requests to the vulnerable web server (by using invisible frames). Following these steps, an attacker could steal information from the website and do the following:

- Inject partial plaintext they have uncovered into a victim's requests
- Measure the size of encrypted traffic

Solution for the vulnerability

Remedy

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

- Served from a server that uses HTTP-level compression (ie. gzip)
- Reflects user-input in the HTTP response bodies
- Contains sensitive information (such as a CSRF token) in HTTP response bodies

To mitigate the issue, we recommend the following solutions:

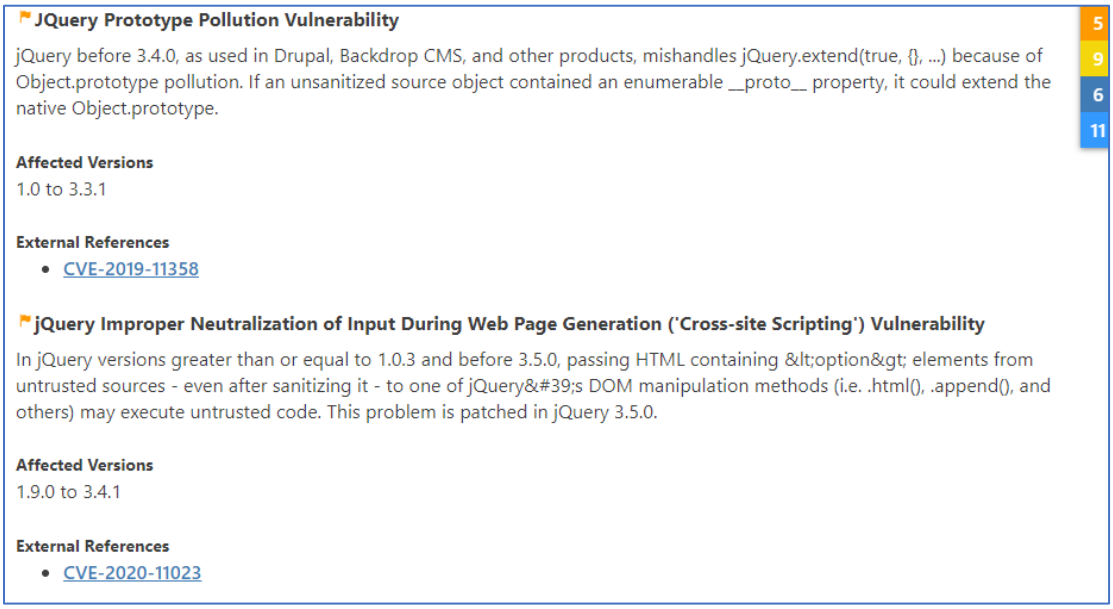
1. If possible, disable HTTP level compression
2. Separate sensitive information from user input
3. Protect vulnerable pages with CSRF token. The SameSite Cookie attribute will mitigate this issue, because to exploit this issue an attacker forces the victim to visit a target website using invisible frames. With the SameSite cookie attribute added, cookies that belong to the target won't be sent with a request that does not include top level navigation.
4. Hide the length of the traffic by adding a random number of bytes to the responses.
5. Add in a rate limit, so that the page maximum is reached five times per minute.

Above information about impact and solution was directly extracted from the vulnerability report of Netsparker

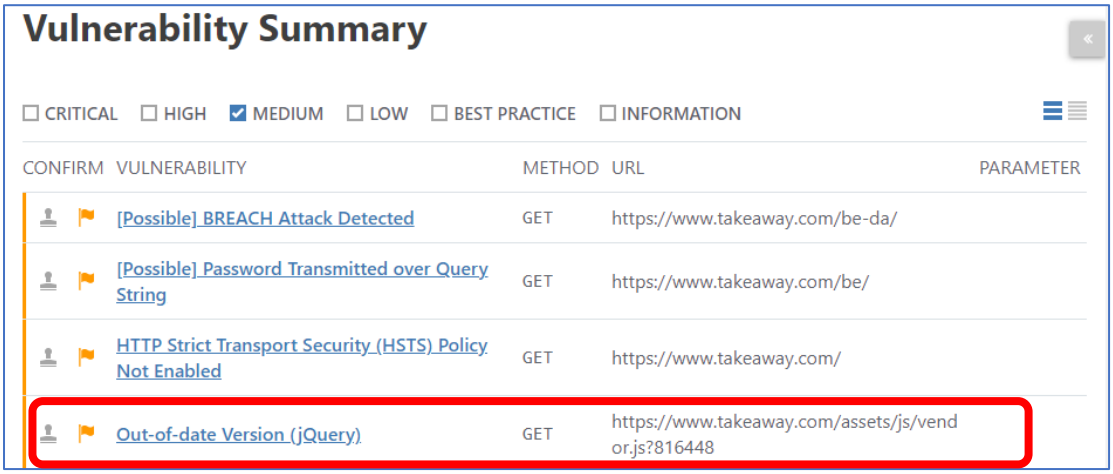
c. Out-of-date version of jQuery

- Severity :- Medium

Due to this out-of-date version of jQuery, an attacker will be able to exploit two type of vulnerabilities in them. The version of jQuery that is identified in this website is version 3.3.1. This version is vulnerable for both vulnerabilities that exist in the older versions. They are mentioned in the image provided below.



Proof of Vulnerability



Impact

- Due to this being an older version of software, it maybe vulnerable to know and unknown attacks.

Solution for the vulnerability

- Upgrading to the latest stable version of jQuery installation will solve this vulnerability

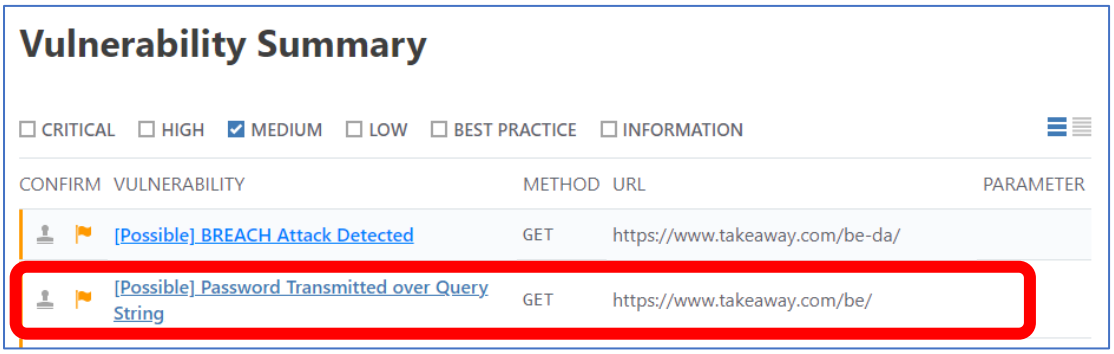
Sensitive Data Exposure

a. [Possible] Passwords transmitted over Query String

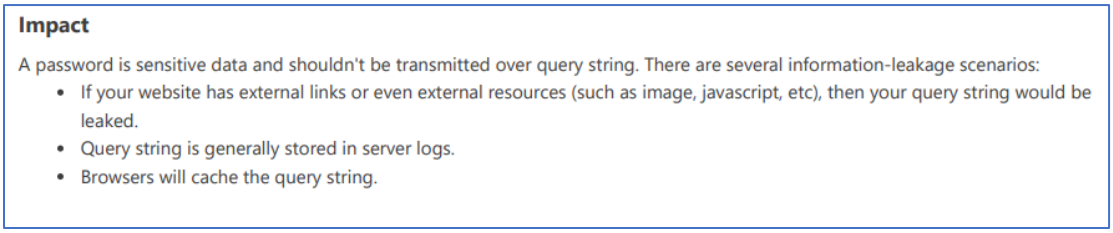
- Severity :- Medium

A password is a sensitive data that should not be transmitted over a query string. If a Man in the middle attack was performed, the attacker can easily obtain the password from the query string. This will have a serious impact to the relevant users account and their sensitive data stored in the account such as location details, credit card numbers, phone number etc.

Proof of Vulnerability



The scanner also says that although the checked form uses a get method to transfer the data, it will not directly submit that data to the server. It will use background submission method using AJAX with POST method.



Solution for this vulnerability

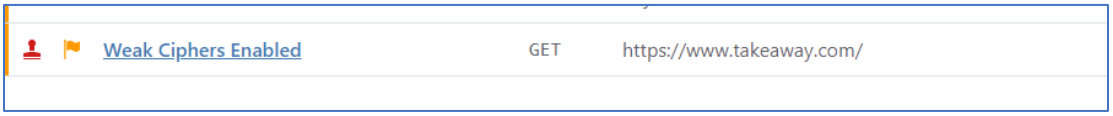
- Sensitive data should not be sent through a query string.

b. Weak Ciphers Enabled

- Severity – Medium

Using weak ciphers to encrypt the data in transit would be a threat to the confidentiality of the data.

Proof of vulnerability



Solution for the vulnerability

Actions to Take

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

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

2. Lighttpd:

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

3. For Microsoft IIS, you should make some changes to the system registry. **Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.**

   a.Click Start, click Run, type `regedt32`or type `regedit`, and then click OK.
   b.In Registry Editor, locate the following registry key: `HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders`
   c.Set "Enabled" DWORD to "0x0" for the following registry keys:

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

Remedy

Configure your web server to disallow using weak ciphers.

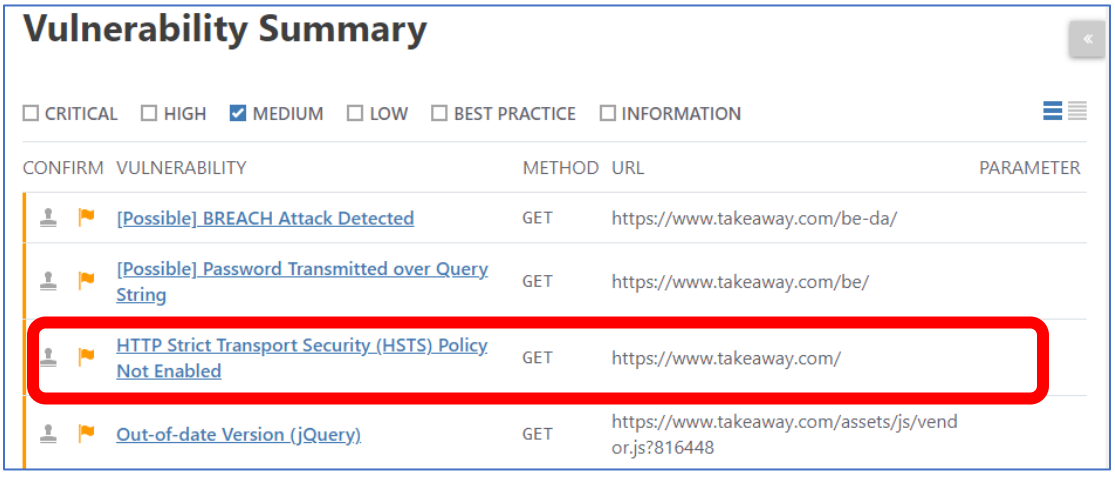
Security Misconfigurations

a. HTTP Strict Transport Security(HSTS) Policy Not Enabled

- Severity :- Medium

This security header will force the browser to use the HTTPS protocol instead of HTTP protocol when communicating with the server. This header will stop any attacker from downgrading any https protocol to http protocol. Without using this header, an attacker could downgrade the protocol to http and extract any sensitive information that is in transit by performing a Man in the Middle attack.

Proof of Vulnerability



Solution for the vulnerability

Remedy

Configure your webserver to redirect HTTP requests to HTTPS.

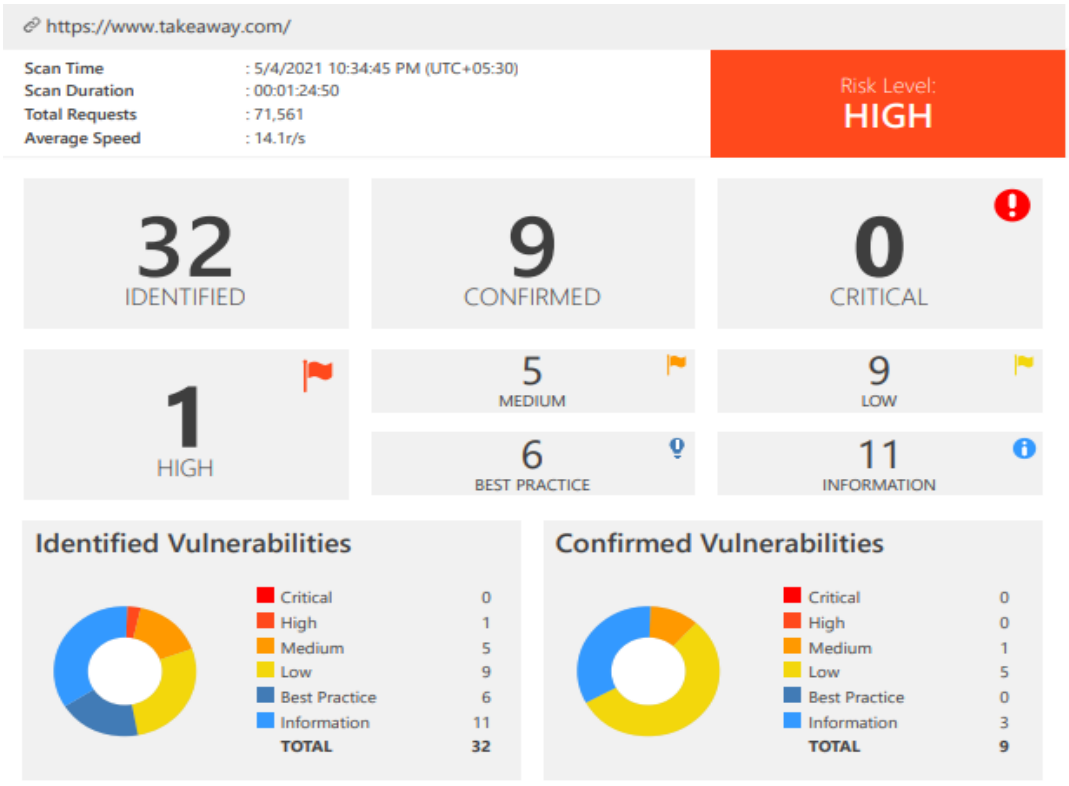
i.e. for Apache, you should have modification in the httpd.conf. For more configurations, please refer to External References section.

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

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

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

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

» 0 1 5 9 6 11

(The HSTS header)

Summary



2. https://careers.takeaway.com/

Security Misconfigurations

a. Misconfigured Access-Control-Allow-Origin Header

- Severity :- Low

Impact

This cookie will be transmitted over a HTTP connection, therefore if this cookie is important (*such as a session cookie*), an attacker might intercept it and hijack a victim's session. If the attacker can carry out a man-in-the-middle attack, he/she can force the victim to make an HTTP request to steal the cookie.

Response Time (ms) :	Total Bytes Received :	Body Length :	Is Compressed :
196.0951	569	0	No

```
HTTP/1.1 303 See Other
Set-Cookie: JSESSIONID=21c97288-ad35-4af7-a4e1-0f391c337f1b; Path=/; Secure
X-Permitted-Cross-Domain-Policies: master-only
X-Content-Type-Options: nosniff
Strict-Transport-Security: max-age=31536000; preload
Connection: keep-alive
X-XSS-Protection: 1; mode=block
Content-Length: 0
Access-Control-Allow-Credentials: true
X-Frame-Options: SAMEORIGIN
https: //careers.takeaway.comâ
Access-Control-Allow-Origin: http://r87.com
Location: /global/en
Date: Tue, 04 May 2021 18:45:03 GMT
Vary: Origin
Cache-Control: no-cache,no-store
```

Solution for the vulnerability

Remedy

If this page is intended to be accessible to everyone, you don't need to take any action. Otherwise please follow the guidelines for different architectures below in order to set this header and permit outside domain.

Apache

- Add the following line inside either the <directory>, <location>, <files> or <virtualhost> sections of your server config (usually located in `httpd.conf` or `apache.conf`), or within a `.htaccess` file.

```
Header set Access-Control-Allow-Origin "domain"
```

IIS6

1. Open Internet Information Service (IIS) Manager
2. Right click the site you want to enable CORS for and go to Properties
3. Change to the HTTP Headers tab
4. In the Custom HTTP headers section, click Add
5. Enter Access-Control-Allow-Origin as the header name
6. Enter `domain` as the header value

IIS7

- Merge the following xml into the web.config file at the root of your application or site:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <system.webserver>
    <httpprotocol>
      <customheaders>
        <add name="Access-Control-Allow-Origin" value="domain" />
      </customheaders>
    </httpprotocol>
  </system.webserver>
</configuration>
```

ASP.NET

- If you don't have access to configure IIS, you can still add the header through ASP.NET by adding the following line to your source pages:

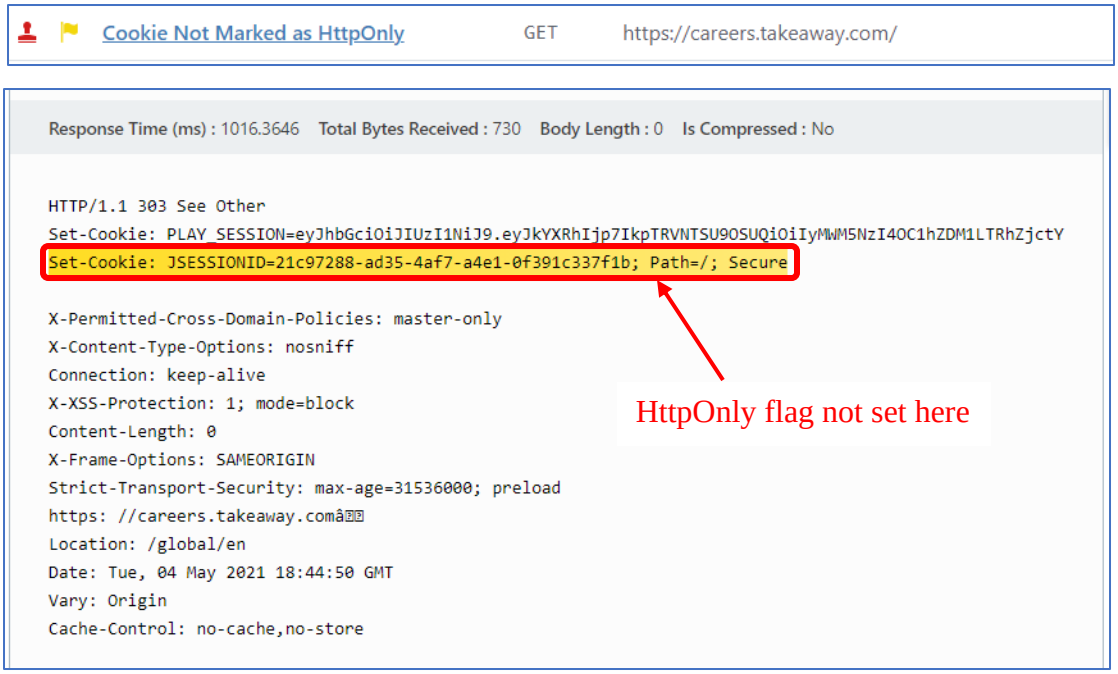
```
Response.AppendHeader("Access-Control-Allow-Origin", "domain");
```

b. Cookie Not Marked as HttpOnly

- Severity : Low

HttpOnly is a flag set in the Set-Cookie HTTP response header. This will help to reduce the risk of accessing the protected cookie by client-side scripts. Browser must support this flag in order to execute the rules. If the browser does not support this, the browser will not break any webpage, it will simply ignore it and process the rest.

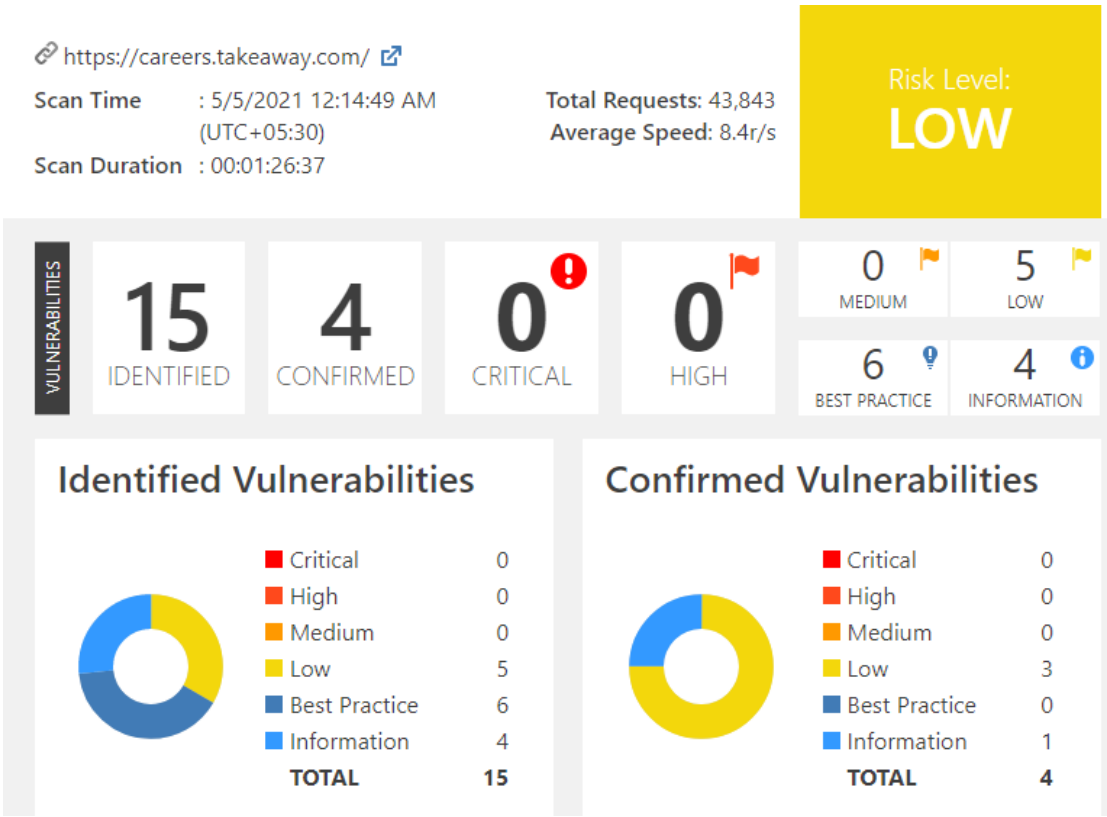
Proof of vulnerability



Solution for the vulnerability

- Setting HttpOnly flag in almost all the cookies used by the website. This will add an extra later of security for Cross Site Scripting attacks but will not complete stop the attacks against XSS.

Summary



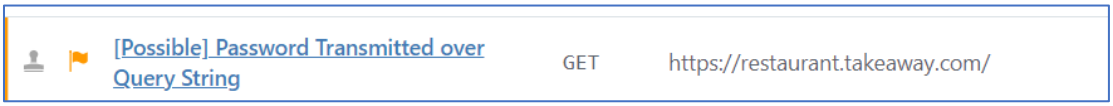
3. https://restaurant.takeaway.com/

Sensitive Data Exposure

a. [Possible] Password Transmitted over Query String

- Severity :- Medium

Proof of vulnerability



Impact

A password is sensitive data and shouldn't be transmitted over query string. There are several information-leakage scenarios:

- If your website has external links or even external resources (such as image, javascript, etc), then your query string would be leaked.
- Query string is generally stored in server logs.
- Browsers will cache the query string.

Solution to the vulnerability

- Do not send any sensitive data through query string.

b. Weak Ciphers Enabled

- Severity :- Medium

Proof of vulnerability

Impact

Attackers might decrypt SSL traffic between your server and your visitors.

Solution for the vulnerability

Remedy

Configure your web server to disallow using weak ciphers.

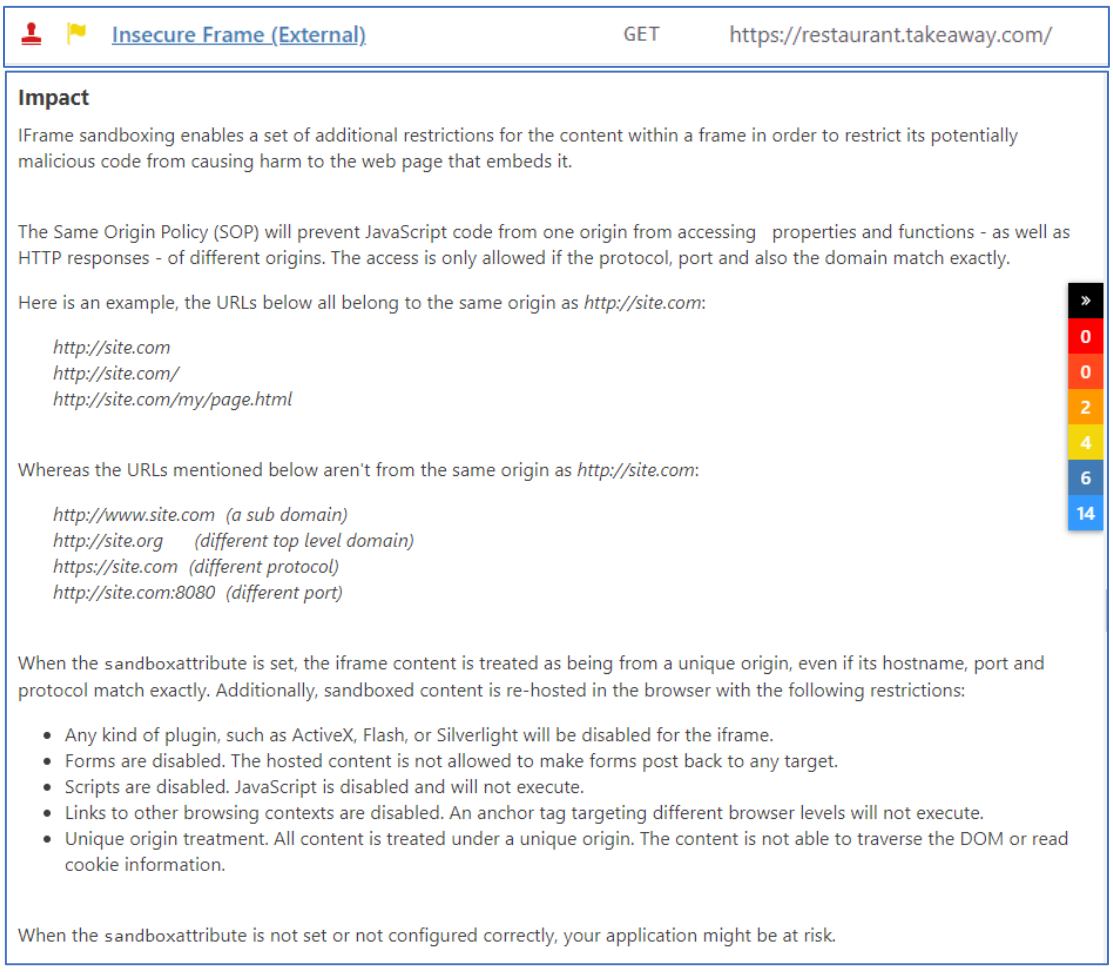
Security Misconfigurations

a. Insecure Frame (External)

- Severity :- Low

Iframe is a frame that is withing a frame and it allow us to embed interactive media and even other web pages withing a webpage. If proper security attributes and flags are not implemented correctly, this could even lead to a XSS attack if other variables adhere to the situation. Security header attributes like sandbox should be used in order to protect the website from attacks.

Proof of Vulnerability



Solution for the vulnerability

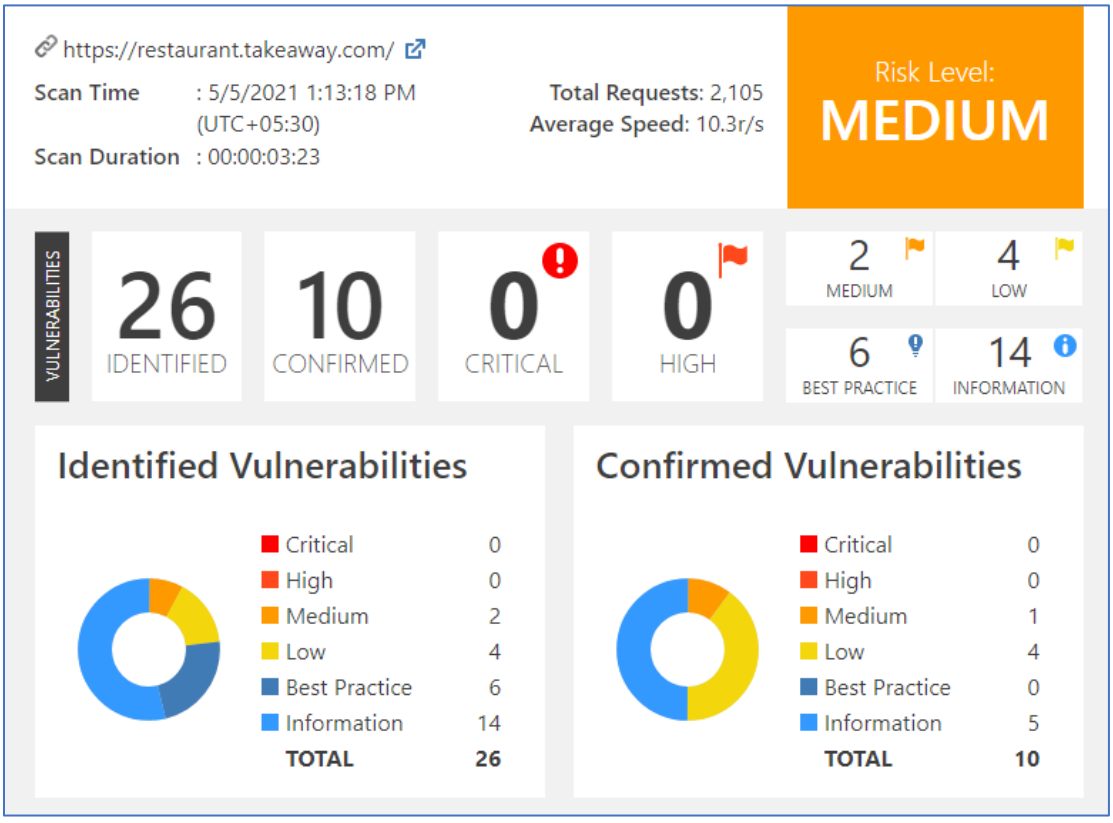
Remedy

- Apply sandboxing in inline frame

```
<iframe sandbox src="framed-page-url"></iframe>
```

- For untrusted content, avoid the usage of `seamless`attribute and `allow-top-navigation`, `allow-popups`and `allow-scripts`in sandbox attribute.

Summary



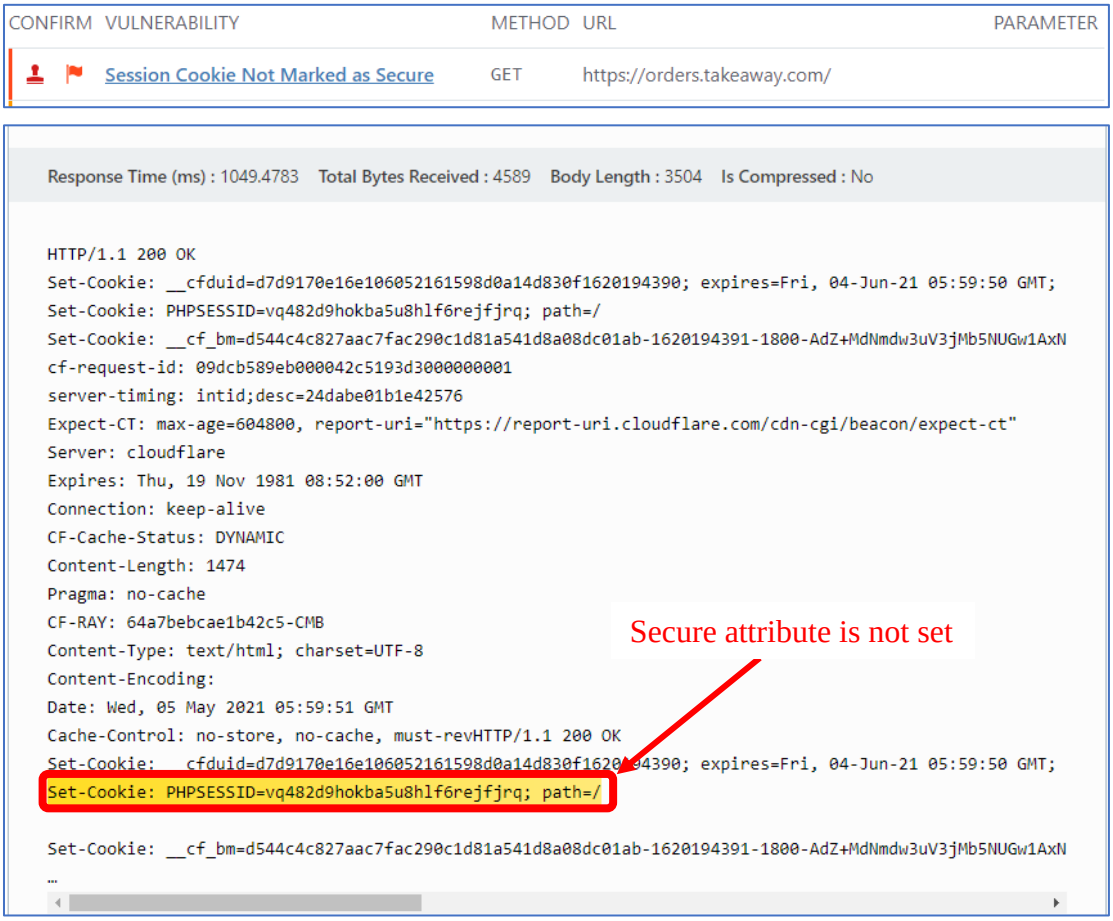
4. https://orders.takeaway.com/

a. Session Cookie not Marked as Secure

- Severity :- High

The reason to set the Secure flag to a cookie is that when the secure flag is set, the cookie will be sent/transmitted only through a secure channel. When this flag is not set, the cookie can be transmitted even through http. So, here the secure flag is not set, so an attacker can steal the cookie by doing a man in the middle attack and highjack the relevant users session.

Proof of vulnerability



Impact

This cookie will be transmitted over a HTTP connection, therefore an attacker might intercept it and hijack a victim's session. If the attacker can carry out a man-in-the-middle attack, he/she can force the victim to make an HTTP request to your website in order to steal the cookie.

Solution for the vulnerability

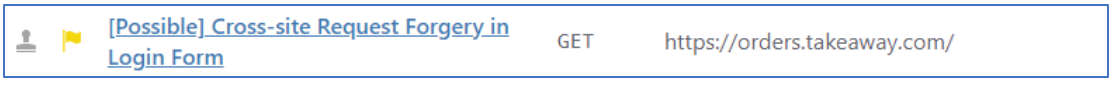
- Use the Secure attribute on all the cookies used within the website.

b. [Possible] Cross-site Request Forgery in Login Form

- Severity :- Low

Cross-Site Request Forgery (CSRF) is a type of attack that induces an authenticated user to perform unauthorized actions on a web application. An attacker may use social engineering to trick users of a web application into performing acts that the attacker desires. If the victim is a regular user, a successful CSRF attack will compel them to make state-changing demands, such as transferring funds or changing their email address. CSRF will compromise the entire web application if the victim is an administrative account.

Proof of Vulnerability



Impact

In this particular case CSRF affects the login form in which the impact of this vulnerability is decreased significantly. Unlike normal CSRF vulnerabilities this will only allow an attacker to exploit some complex XSS vulnerabilities otherwise it can't be exploited.

For example;

If there is a page that's different for every user (such as "edit my profile") and vulnerable to XSS (Cross-site Scripting) then normally it cannot be exploited. However if the login form is vulnerable, an attacker can prepare a special profile, force victim to login as that user which will trigger the XSS exploit. Again attacker is still quite limited with this XSS as there is no active session. However the attacker can leverage this XSS in many ways such as showing the same login form again but this time capturing and sending the entered username/password to the attacker.

Solution to the vulnerability

Remedy

- Send additional information in each HTTP request that can be used to determine whether the request came from an authorized source. This "validation token" should be hard to guess for attacker who does not already have access to the user's account. If a request is missing a validation token or the token does not match the expected value, the server should reject the request.
- If you are posting form in ajax request, custom HTTP headers can be used to prevent CSRF because the browser prevents sites from sending custom HTTP headers to another site but allows sites to send custom HTTP headers to themselves using XMLHttpRequest.
    - For native XMLHttpRequest (XHR) object in JavaScript;

```
xhr = new XMLHttpRequest();
xhr.setRequestHeader('custom-header', 'valueNULL');
```

For JQuery, if you want to add a custom header (or set of headers) to
a. **individual request**

```
$.ajax({
    url: 'foo/bar',
    headers: { 'x-my-custom-header': 'some value' }
});
```

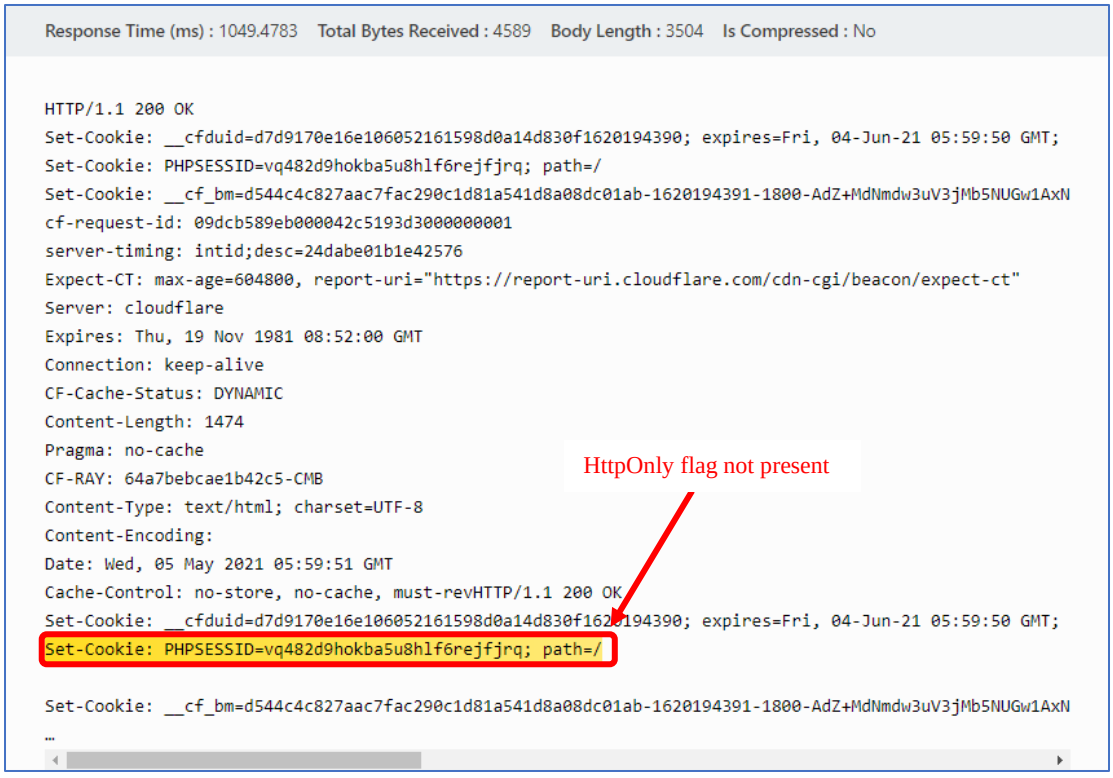
b. **every request**

```
$.ajaxSetup({
    headers: { 'x-my-custom-header': 'some value' }
});
OR
$.ajaxSetup({
    beforeSend: function(xhr) {
        xhr.setRequestHeader('x-my-custom-header', 'some value');
    }
});
```

c. Cookies Not Marked as HttpOnly

- Severity :- Low

Proof of vulnerability



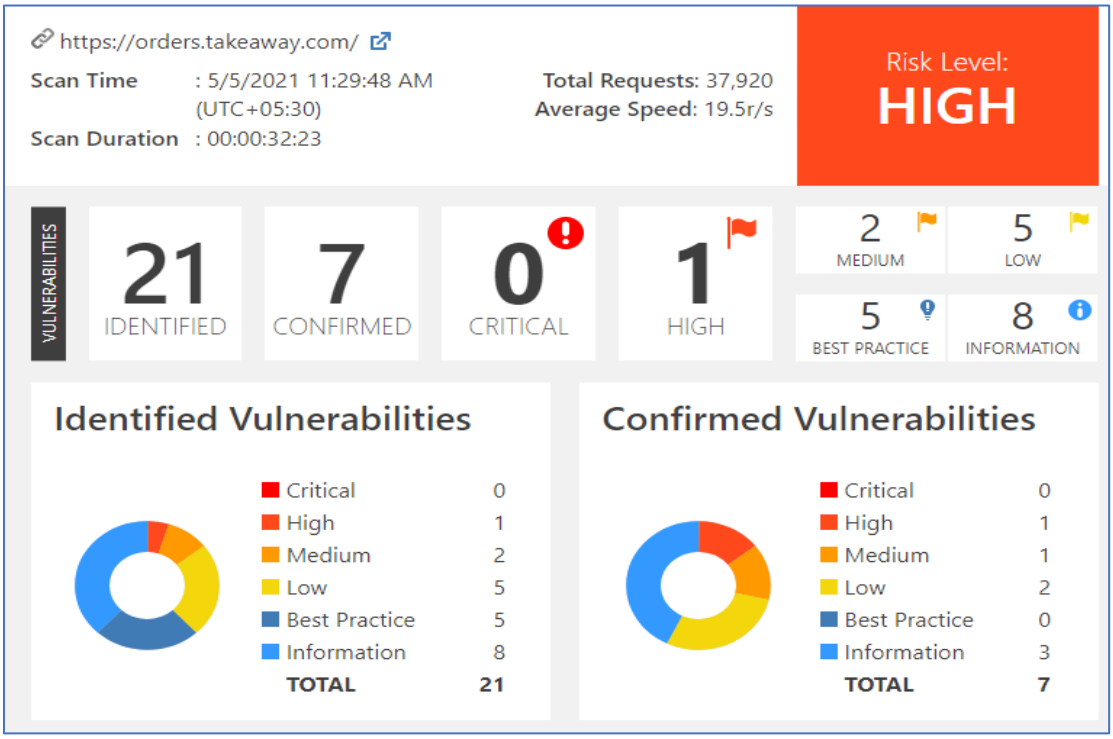
Impact

During a cross-site scripting attack, an attacker might easily access cookies and hijack the victim's session.

Solution to the vulnerability

> **Remedy**
>
> Mark the cookie as HTTPOnly. This will be an extra layer of defense against XSS. However this is not a silver bullet and will not protect the system against cross-site scripting attacks. An attacker can use a tool such as XSS Tunnelto bypass HTTPOnly protection.

Summary



5. https://shop.takeaway.com/

a. Out-of-date version of Nginx

- Severity :- High

NGINX is a multipurpose application. You can use the same tool for your reverse proxy, load balancer, web server and content cache with NGINX, reducing the amount of tooling and setup your company must keep up to date. web server

Nginx Allocation of Resources Without Limits or Throttling Vulnerability

Some HTTP/2 implementations are vulnerable to a header leak, potentially leading to a denial of service. The attacker sends a stream of headers with a 0-length header name and 0-length header value, optionally Huffman encoded into 1-byte or greater headers. Some implementations allocate memory for these headers and keep the allocation alive until the session dies. This can consume excess memory.

Affected Versions
1.9.5 to 1.16.0

External References
- CVE-2019-9516

Nginx Other Vulnerability

Some HTTP/2 implementations are vulnerable to resource loops, potentially leading to a denial of service. The attacker creates multiple request streams and continually shuffles the priority of the streams in a way that causes substantial churn to the priority tree. This can consume excess CPU.

Affected Versions
1.9.5 to 1.16.0

External References
- CVE-2019-9513

Nginx Allocation of Resources Without Limits or Throttling Vulnerability

Some HTTP/2 implementations are vulnerable to window size manipulation and stream prioritization manipulation, potentially leading to a denial of service. The attacker requests a large amount of data from a specified resource over multiple streams. They manipulate window size and stream priority to force the server to queue the data in 1-byte chunks. Depending on how efficiently this data is queued, this can consume excess CPU, memory, or both.

Affected Versions
1.9.5 to 1.16.0

External References
- CVE-2019-9511

Nginx Inconsistent Interpretation of HTTP Requests ('HTTP Request Smuggling') Vulnerability

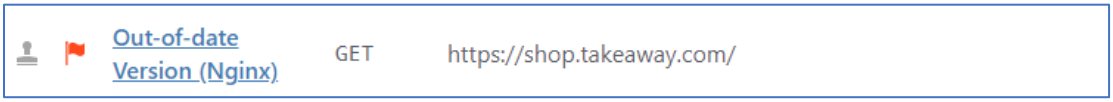
NGINX before 1.17.7, with certain error_page configurations, allows HTTP request smuggling, as demonstrated by the ability of an attacker to read unauthorized web pages in environments where NGINX is being fronted by a load balancer.

Affected Versions
1.7.5 to 1.16.0

External References
- CVE-2019-20372

The identified version of Nginx is version 1.14.0. The latest version available is version1.20.0.

Proof of Vulnerability




```
HTTP/1.1 200 OK
Set-Cookie: AWSALB=mqKNi5mM3nq9g9rw2YcLzLFICKiRyv91IcN9HQH9lY001mOQI03nqz8HNAxW2qtnhSZ7yfA/SKnbapif9Vep
Set-Cookie: AWSALBCORS=mqKNi5mM3nq9g9rw2YcLzLFICKiRyv91IcN9HQH9lY001mOQI03nqz8HNAxW2qtnhSZ7yfA/SKnbapif
Set-Cookie: PLAY_SESSION=eyJhbGciOiJIUzI1NiJ9.eyJkYXRhIjp7ImNzcmZUb2tlbiI6ImEyYmQ5NmI3NTQzOTNjZGJkZDBkN
Set-Cookie: visid_incap_2322767=7bSA1LbXRKO52S04DmT0MXF0kmAAAAAAQUIPAAAAAADBJomHXtwO0wwBD/Lbawih; expir
Set-Cookie: nlbi_2322767=B+w1F5WqRQnE79LQngmJ0gAAAACMbsrFVu50iO/0dtjTe7GX; path=/; Domain=.takeaway.com
Set-Cookie: incap_ses_219_2322767=clR1SO6qljL6vZIfpgsKA3F0kmAAAAAA6PZxQ3wpVsnag2HwaO4U/A==; path=/; Dom
Server: nginx/1.14.0 (Ubuntu)
Expires: 0
Connection: keep-alive
X-Iinfo: 1-285342-285347 NNNN CT(250 251 0) RT(1620210800136 89) q(0 0 5 -1) r(10 10) U12
Pragma: no-cache
Strict-Transport-Security: max-age=31536000
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Content-Encoding:
X-CDN: Imperva
Date: W
…
; path=/; Domain=.takeaway.com; Secure; SameSite=None
Set-Cookie: incap_ses_219_2322767=clR1SO6qljL6vZIfpgsKA3F0kmAAAAAA6PZxQ3wpVsnag2HwaO4U/A==; path=/; Dom
Server: nginx/1.14.0(Ubuntu)
Expires: 0
Connection: keep-alive
X-Iinfo: 1-285342-285347 NNNN CT(250 251 0) RT(1620210800136 89) q(0 0 5 -1) r(10 10) U12
Pragma: no-cache
Strict-Transport-Security: max-age=31536000

…
```

Older version of Nginx

Solution for the vulnerability

- Upgrade to the latest stable version of Nginx installation

b. Out-of-date Version of Bootstrap

- Severity :- Medium

Bootstrap is a free front-end platform that makes web development go faster and easier. Bootstrap features interface templates for typography, shapes, buttons, tables, navigation, modals, image carousels, and more, as well as optional JavaScript plugins. Following are the possible impacts due to this vulnerability.

Bootstrap Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability

In Bootstrap before 3.4.0, XSS is possible in the tooltip data-viewport attribute.

Affected Versions
1.0.0 to 3.3.7

External References
- CVE-2018-20676

Bootstrap Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability

In Bootstrap before 3.4.0, XSS is possible in the affix configuration target property.

Affected Versions
1.0.0 to 3.3.7

External References
- CVE-2018-20677

Bootstrap Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability

In Bootstrap before 3.4.1 and 4.3.x before 4.3.1, XSS is possible in the tooltip or popover data-template attribute.

Affected Versions
1.0.0 to 3.4.0

External References
- CVE-2019-8331

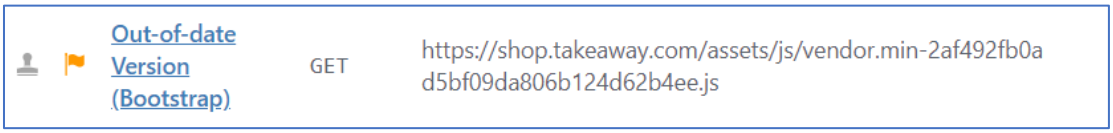
Bootstrap Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability

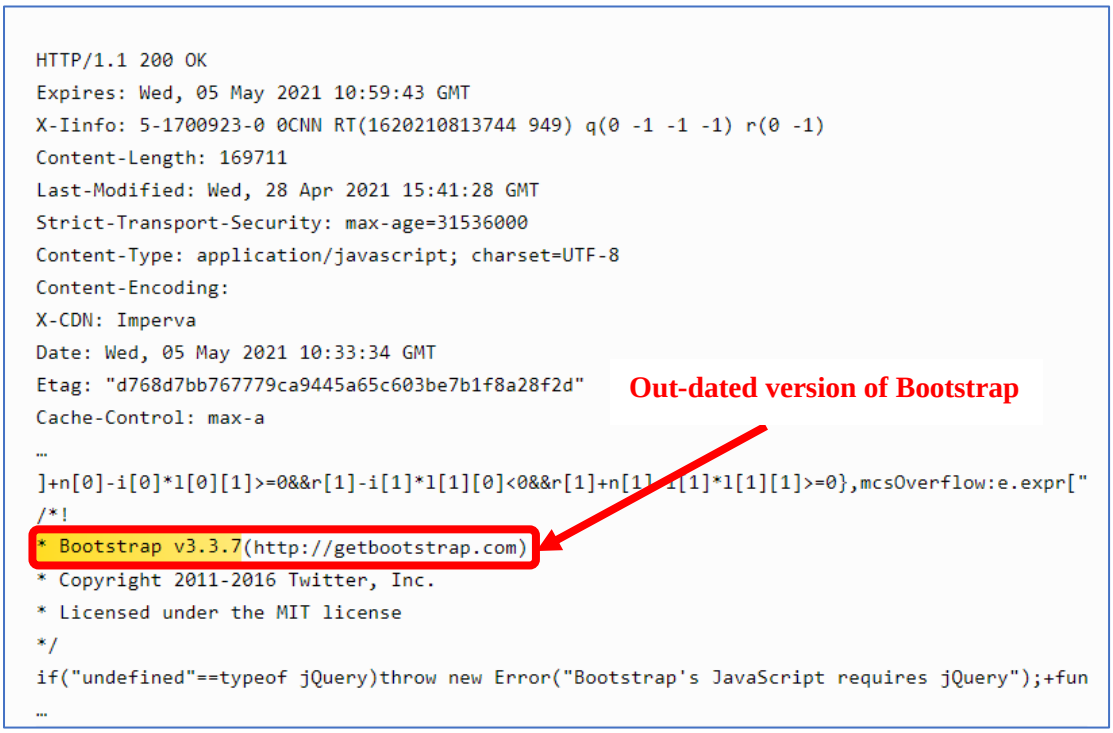
In Bootstrap 3.x before 3.4.0 and 4.x-beta before 4.0.0-beta.2, XSS is possible in the data-target attribute, a different vulnerability than CVE-2018-14041.

Affected Versions
3.0.0 to 3.3.7

External References
- CVE-2016-10735

Proof of Vulnerability





The version used on the website is version 3.3.7 and the latest version is version 3.4.1.

Solution for the vulnerability

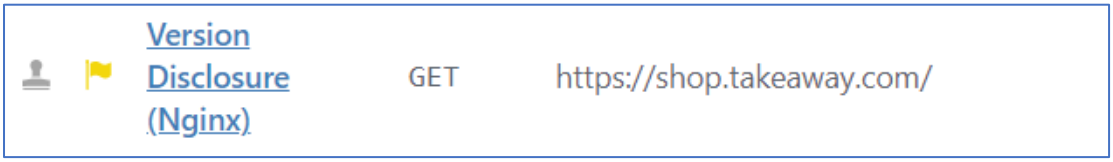
- Upgrade to the latest stable version of Bootstrap installation

c. Version Disclosure (Nginx)

- Severity :- Low

Software version being disclosed is a threat to the system because and attacker could find vulnerabilities known before according to the version of the software.

Proof of Vulnerability




```
HTTP/1.1 200 OK
Set-Cookie: AWSALB=mqKNi5mM3nq9g9rw2YcLzLFICKiRyv91IcN9HQH91Y001mOQI03nqz8HNAxW2qtnhSZ7yfA/S
Set-Cookie: AWSALBCORS=mqKNi5mM3nq9g9rw2YcLzLFICKiRyv91IcN9HQH91Y001mOQI03nqz8HNAxW2qtnhSZ7y
Set-Cookie: PLAY_SESSION=eyJhbGciOiJIUzI1NiJ9.eyJkYXRhIjp7ImNzcmZUb2tlbiI6ImEyYmQ5NmI3NTQzOT
Set-Cookie: visid_incap_2322767=7bSA1LbXRKO52S04DmT0MXF0kmAAAAAAQUIPAAAAAADBJomHXtwO0wwBD/Lb
Set-Cookie: nlbi_2322767=B+w1F5WqRQnE79LQngmJ0gAAAACMbsrFVu50iO/0dtjTe7GX; path=/; Domain=.t
Set-Cookie: incap_ses_219_2322767=clR1SO6qljL6vZIfpgsKA3F0kmAAAAAA6PZxQ3wpVsnag2HwaO4U/A==;
Server: nginx/1.14.0 (Ubuntu)
Expires: 0
Connection: keep-alive
X-Iinfo: 1-285342-285347 NNNN CT(250 251 0) RT(1620210800136 89) q(0 0 5 -1) r(10 10) U12
Pragma: no-cache
Strict-Transport-Security: max-age=31536000
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Content-Encoding:
X-CDN: Imperva
Date: W
…
; path=/; Domain=.takeaway.com; Secure; SameSite=None
Set-Cookie: incap_ses_219_2322767=clR1SO6qljL6vZIfpgsKA3F0kmAAAAAA6PZxQ3wpVsnag2HwaO4U/A==;
Server: nginx/1.14.0(Ubuntu)
Expires: 0
Connection: keep-alive
X-Iinfo: 1-285342-285347 NNNN CT(250 251 0) RT(1620210800136 89) q(0 0 5 -1) r(10 10) U12
Pragma: no-cache
Strict-Transport-Security: max-age=31536000
```

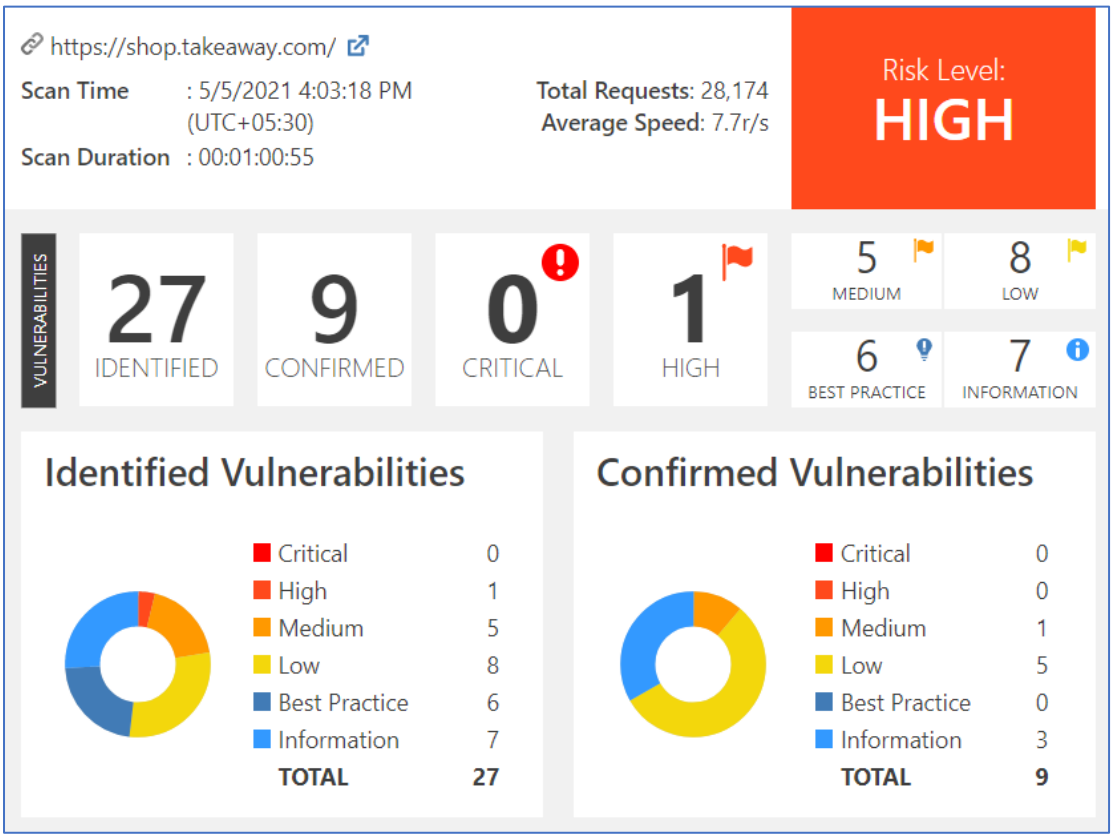
The version being disclosed

Solution to the vulnerability

- Editing the nginx.conf file by inserting the following line mentioned below will prevent the information being leaked from the server

```
server_tokens off
```

Summary



6. https://jobs.takeaway.com/

a. Weak Ciphers Enabled

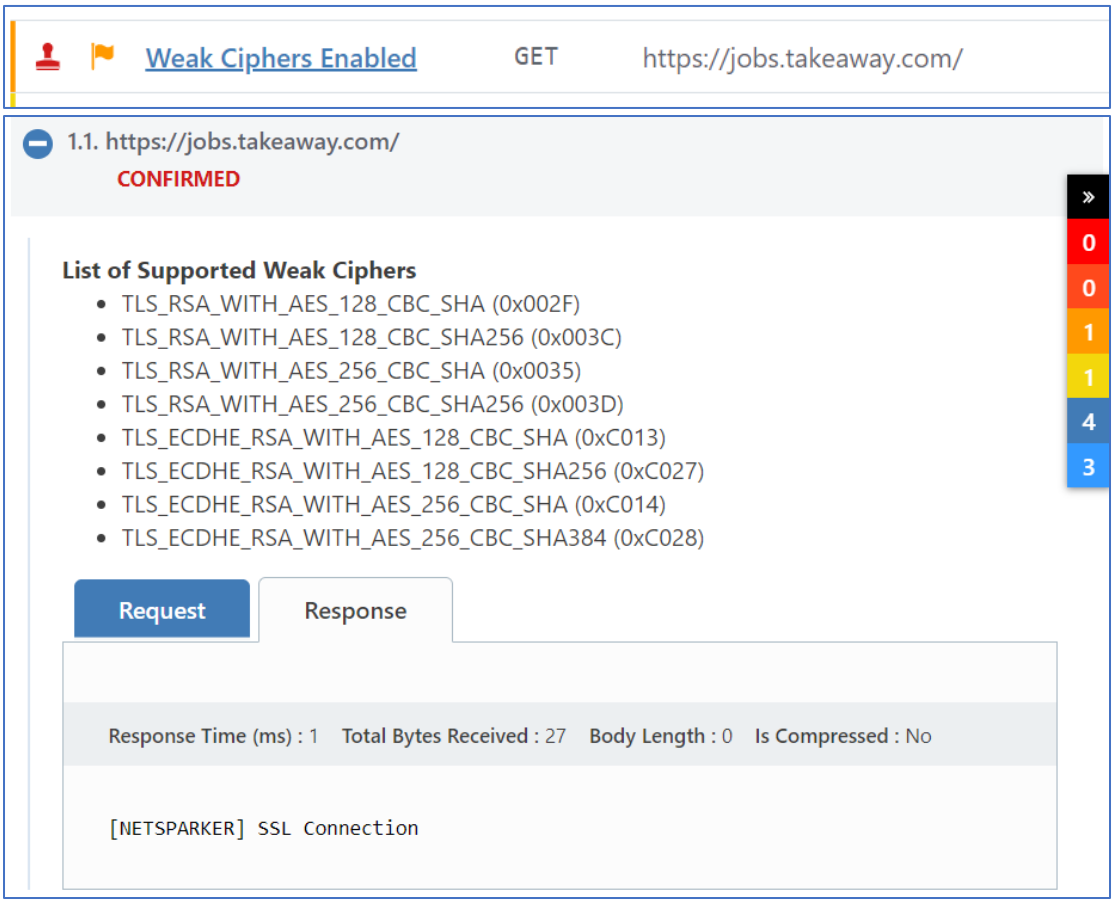
- Severity :- Medium

You should allow only strong ciphers on your web server to protect secure communication with your visitors.

Impact

Attackers might decrypt SSL traffic between your server and your visitors.

Proof of Vulnerability



Solutions for the vulnerability

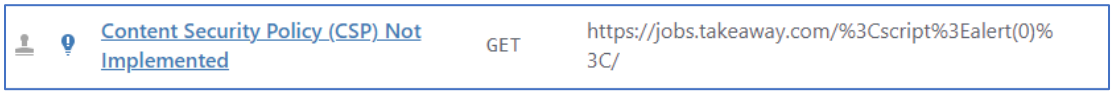
- Usage of weak ciphers on the web server should be disallowed

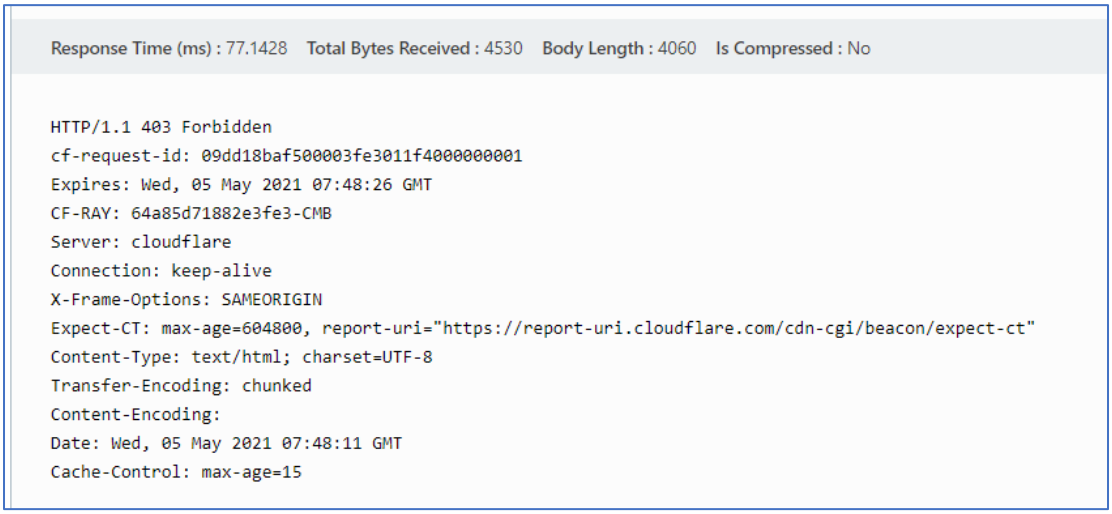
b. Content Security Policy (CSP) not implemented

- Severity :- Best Practice

Content Protection Policy (CSP) is an additional layer of security that aids in the detection and mitigation of such forms of threats, such as Cross-Site Scripting (XSS) and data injection. These attacks are used for a variety of purposes, including data theft, web defacement, and malware distribution.

Proof of vulnerability





Impact

There is no direct effect on your website if you do not use CSP. If your website is vulnerable to a Cross-site Scripting attack, however, CSP may prevent the vulnerability from being exploited successfully. You will be losing out on this extra layer of protection if you do not use CSP.

Solution to the vulnerability

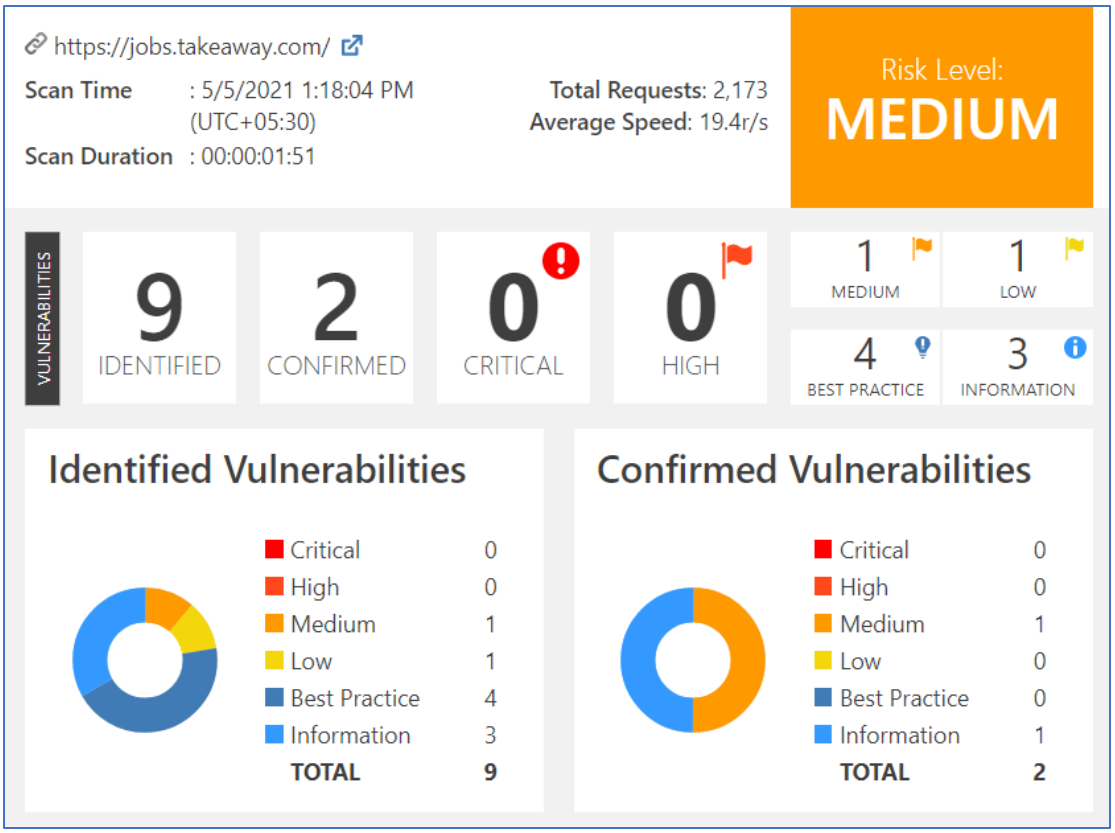
Actions to Take

- Enable CSP on your website by sending the `Content-Security-Policy`in HTTP response headers that instruct the browser to apply the policies you specified.
- Apply the whitelist and policies as strict as possible.
- Rescan your application to see if Netsparker identifies any weaknesses in your policies.

Remedy

Enable CSP on your website by sending the `Content-Security-Policy`in HTTP response headers that instruct the browser to apply the policies you specified.

Summary



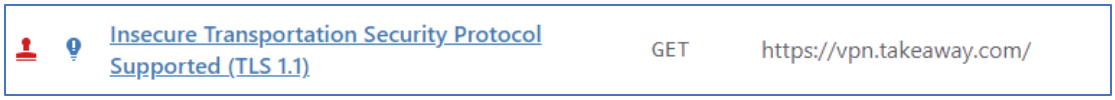
7. https://vpn.takeaway.com/

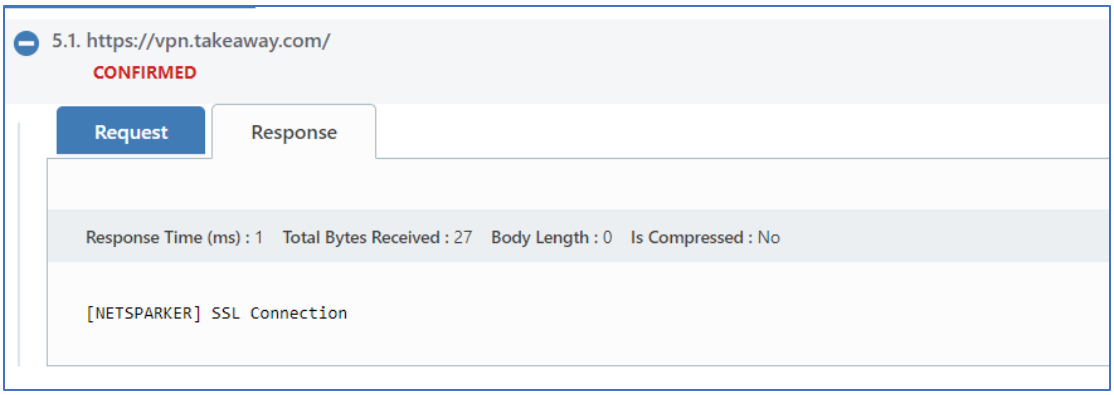
a. Insecure Transportation Security Protocol Supported (TLS 1.1)

- Severity :- Best Practice

Starting from 2020, many major browsers like Chrome, Safari, Edge and Firefox have considered TLS 1.1 to be deprecated because it does not support modern cryptographic functions and algorithms and it contains out of date protocols.

Proof of vulnerability



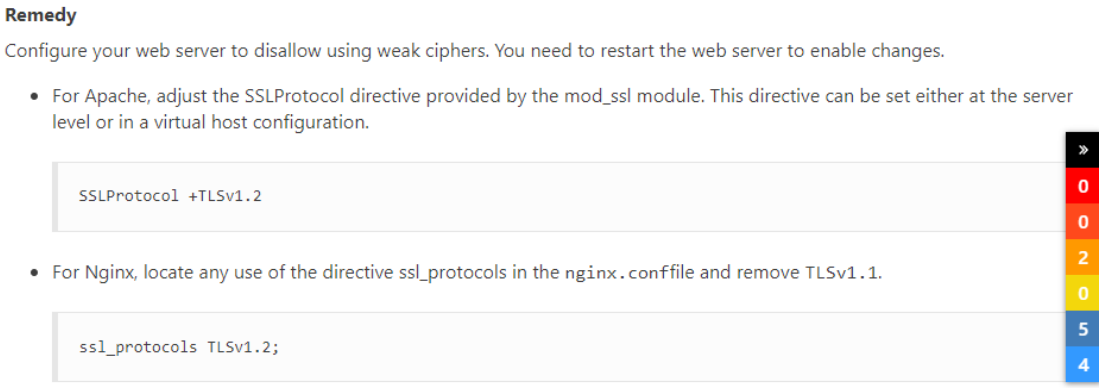


Impact

- Due to the browser deprecation, the website will be inaccessible

Solution to the vulnerability

- **Replace the TSL 1.1 using TSL 1.2 or higher**



8. https://webmail.takeaway.com/

a. SameSite cookie Not Implemented

- Severity :- Best Practice

The Set-Cookie HTTP response header's SameSite attribute lets you specify if your cookie should be restricted to a first-party or same-site context. Same-site cookies allow servers to reduce the possibility of cross-site request forgery (CSRF) and information leakage attacks by specifying that a cookie can only be sent with requests coming from the same registrable domain.

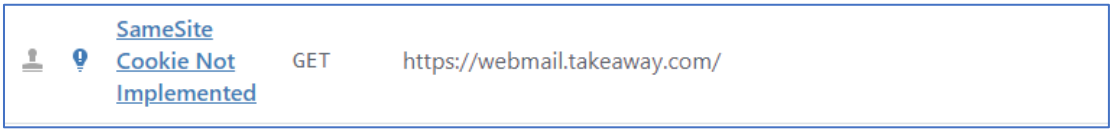
SameSite attribute accepts 3 different values. They are mentioned below.

i. **Lax :-** Cookies are sent when a user navigates to the origin site, not on usual cross-site sub requests.
(If the value for the SameSite attribute is not specifically defined in recent browser versions, this will be set as the default cookie value.)

ii. **Strict :-** Cookies can only be submitted in the form of first-party requests, not requests initiated by third-party websites.

iii. **None :-** Cookies will be submitted in all situations, including first-party and cross-origin requests. The cookie Secure attribute must be set if SameSite=None is set.

Proof of Vulnerability



```
Set-Cookie: cf_use_ob=0; path=/; expires=Wed, 05-May-21 08:02:15 GMT
Expires: Thu, 01 Jan 1970 00:00:01 GMT
CF-RAY: 64a871529fd842c5-CMB
Server: cloudflare
Connection: keep-alive
X-Frame-Options: SAMEORIGIN
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Date: Wed, 05 May 2021 08:01:45 GMT
Cache-Control: private, max-age=0, no-store, no-cache, must-revalidate, post-check=0HTTP/1.1 521
Set-Cookie: cf_use_ob=0; path=/; expires=Wed, 05-May-21 08:02:15 GMT

Expires: Thu, 01 Jan 1970 00:00:01 GMT
CF-RAY: 64a871529fd842c5-CMB
Server: cloudflare
Connection: keep-alive
X-Frame-Options: SAMEORIGIN
Content-Type: text/html; charset=UTF-8
Transfer-Encodin
```

Solution for the vulnerability

Remedy

The server can set a same-site cookie by adding the `SameSite=...`attribute to the `Set-Cookie`header. There are three possible values for the `SameSite`attribute:

- Lax:In this mode, the cookie will only be sent with a top-level get request.

```
Set-Cookie: key=value; SameSite=Lax
```

- Strict: In this mode, the cookie will not be sent with any cross-site usage even if the user follows a link to another website.

```
Set-Cookie: key=value; SameSite=Strict
```

- None: In this mode, the cookie will be sent with the cross-site requests. Cookies with `SameSite=None`must also specify the `Secure`attribute to transfer them via a secure context. Setting a `SameSite=None`cookie without the `Secure`attribute will be rejected by the browsers.

```
Set-Cookie: key=value; SameSite=None; Secure
```

9. https://snacks.takeaway.com/

a. Robots.txt Detected

- Severity :- Information

This is a file which would normally contain details about the pages which the search engine crawlers can or cannot request. This is mostly used to prevent the site from being overburdened with requests and it is not a tool for keeping a web page out of a search engine. Make sure that no site or pages that should be hidden are mentioned here because an attacker would discover it here.

Proof of vulnerability


```
HTTP/1.1 200 OK
cf-request-id: 09ddeed38c000042c5abb7c000000001
CF-Cache-Status: HIT
CF-RAY: 64a9b3ff4a0142c5-CMB
Server: cloudflare
Connection: keep-alive
X-Amzn-Trace-Id: Root=1-60928487-13f1d9cd07cec8803cb01a04;Sampled=0
x-amz-apigw-id: e2mlIEvIjoEFrTA=
Content-Length: 35
Accept-Ranges: bytes
Expect-CT: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
Content-Type: text/plain
Age: 3
Date: Wed, 05 May 2021 11:42:02 GMT
Vary: Accept-Encoding
x-amzn-RequestId: e10323db-9f38-485b-a3d7-0941e3385f93

User-Agent:*
Allow:/$
Disallow:/
```

Solution to the vulnerability

Remedy

Ensure you have nothing sensitive exposed within this file, such as the path of an administration panel. If disallowed paths are sensitive and you want to keep it from unauthorized access, do not write them in the `Robots.txt`, and ensure they are correctly protected by means of authentication.
`Robots.txt`is only used to instruct search robots which resources should be indexed and which ones are not.

Conclusion

This report includes information about the tools that were used to produce the vulnerability scans and to manifest vulnerabilities and potential security threats for which are present in the www.takeaway,com domain. All the demonstrated vulnerabilities are categorized under different severity levels such as informational, low, medium, high and critical.