Sri Lanka Institute of Information Technology

BUG BOUNTY REPORT - 1

Web Security – IE2062

IT22362780

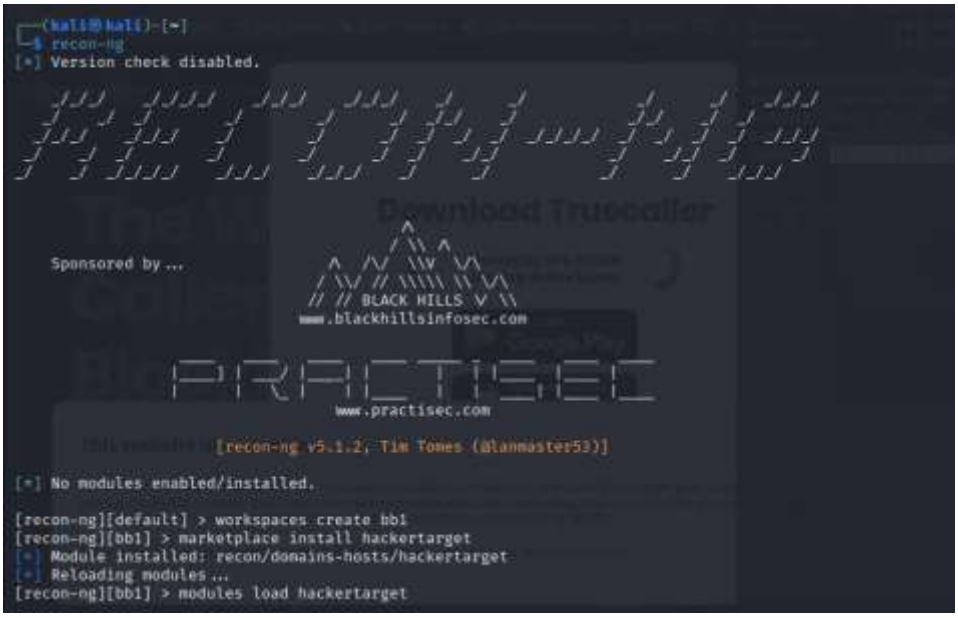
Jayaweera N.S

Report Details

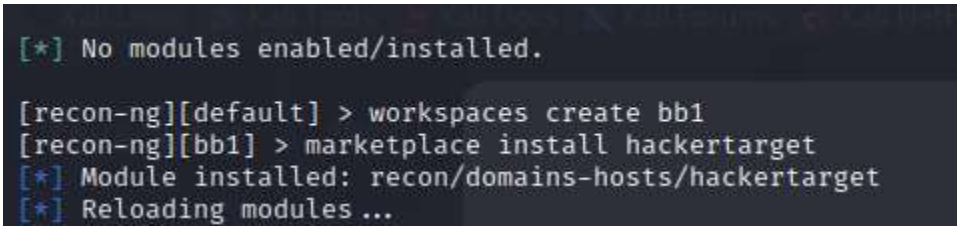
Report #	- 01
Domain	- https://truecaller.com
Platform	-hackerone.com
Scans performed -	Recon-ng scan Nmap scan Metasploit testing Nslookup Wafw00f scan Gobuster scan Wapplyzer scan Zap scan Dotdotpwn scan Sqlmap scan Text injection testing XSS injection testing Command injection testing

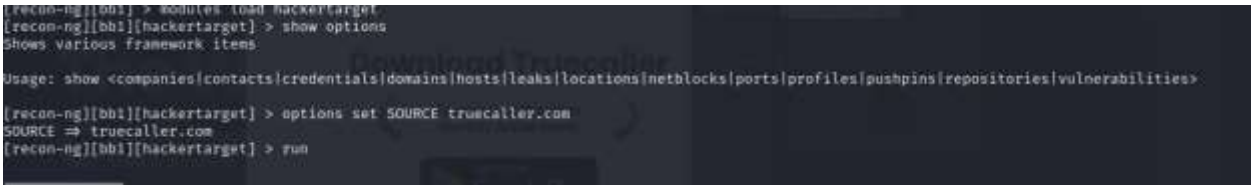
Recon scan

The recon-ng will be used to find all the sub domains in the target.



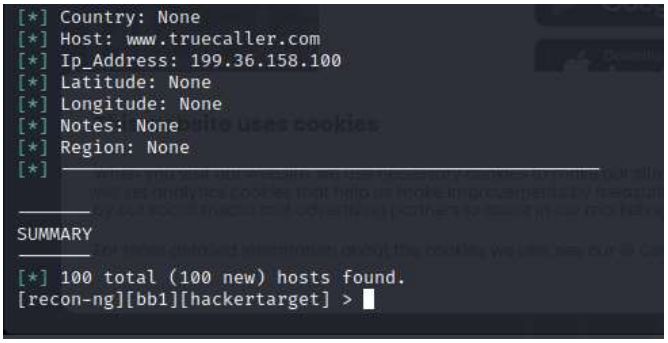
Installing the module" hackertarget".





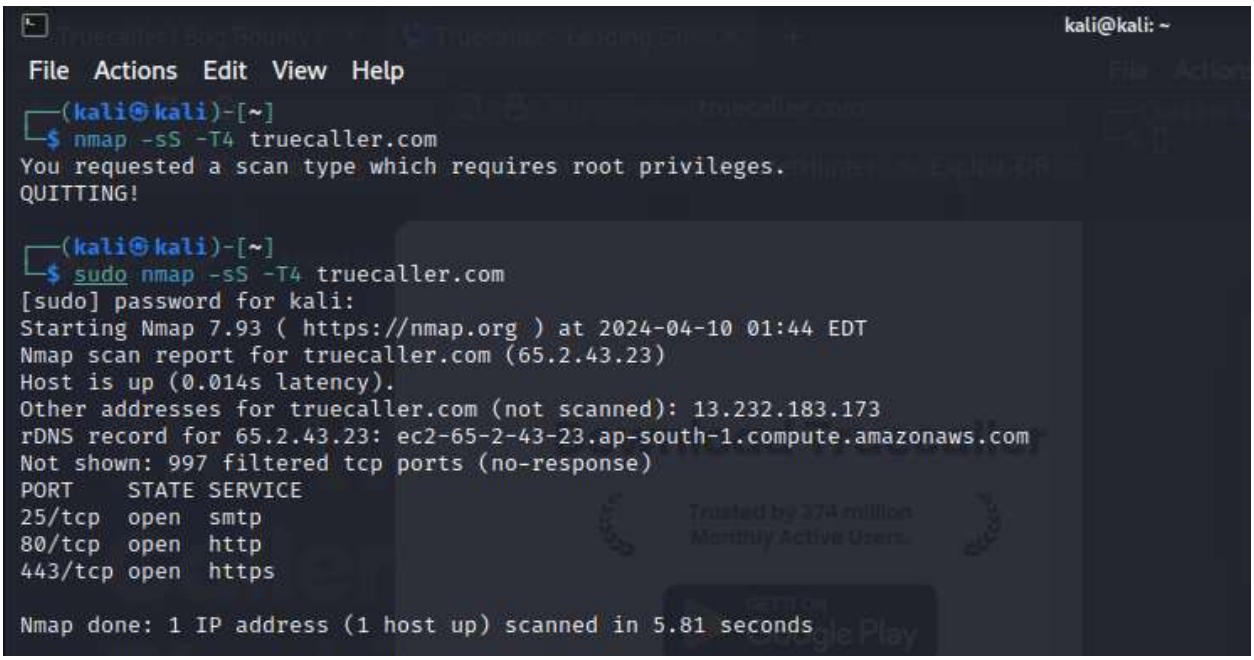
Set the source to truecaller.com

100 sub domains are found.



Nmap scan

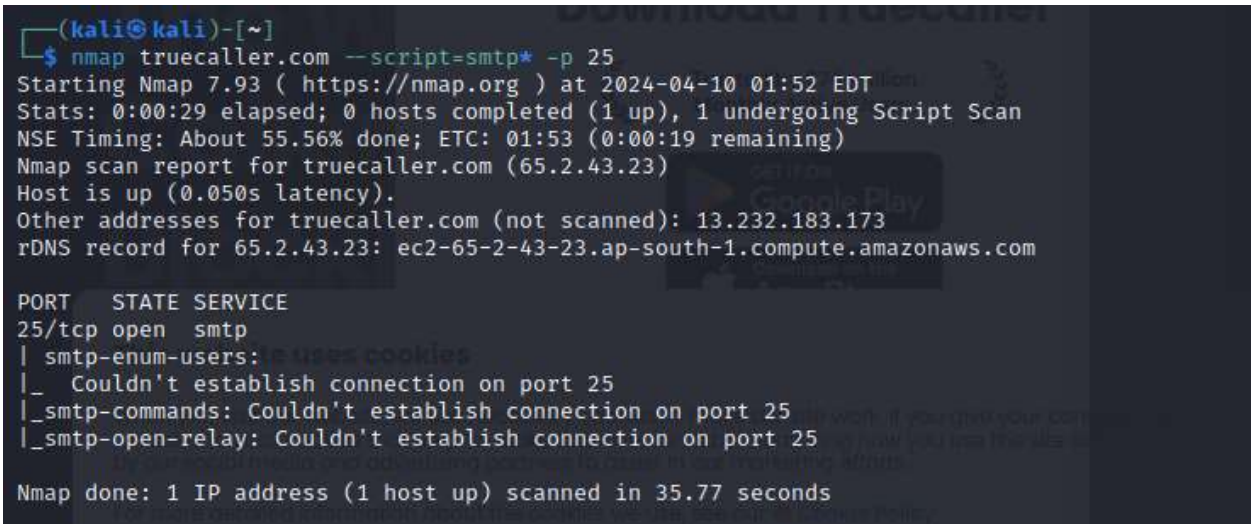
Using nmap scan all the open ports in the target can be identified.



No unusual ports are open.

But Smtp port 25 is vulnerable when it's opened, because it lacks authentication and encryption.

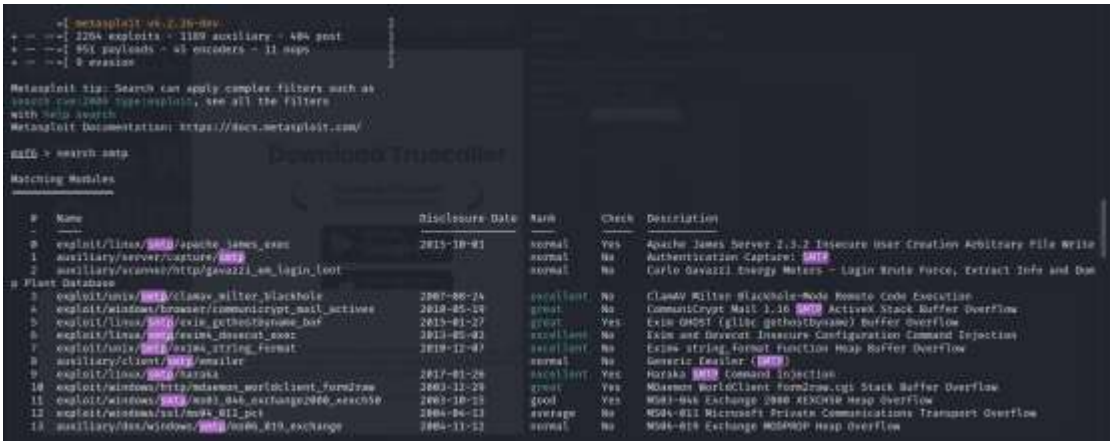
Let's see if we can establish a connection on port 25.



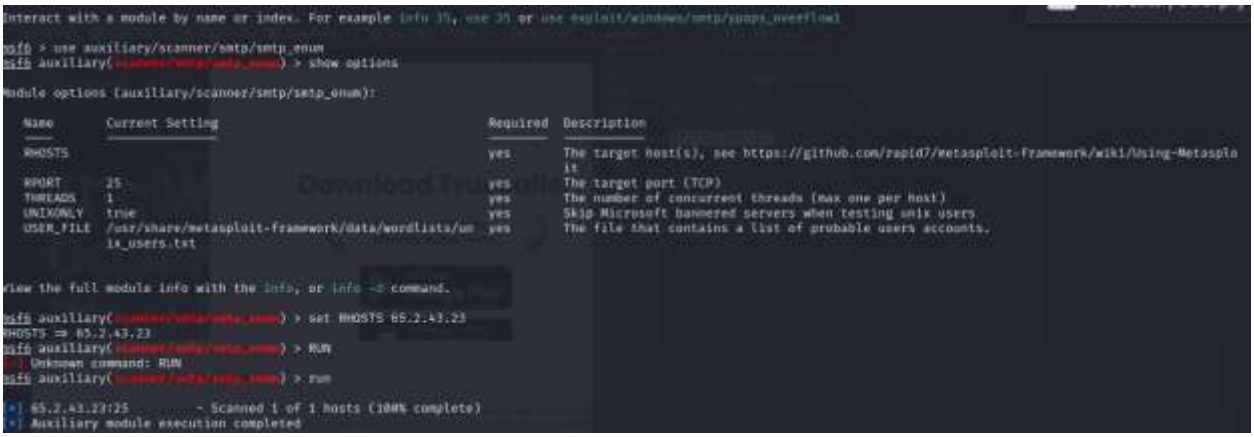
Connection cannot be established therefore a vulnerability cannot be identified.

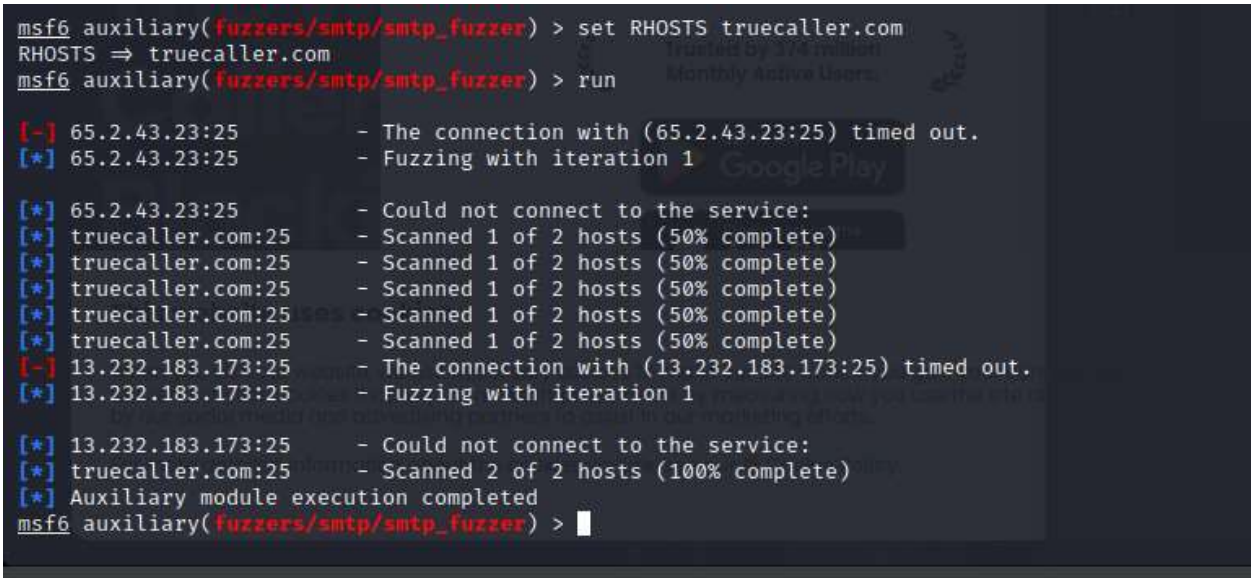
Metasploit scan

Search for the smtp.



Use the module “fuzzer” to fuzz the smtp service. And for username enumeration use the “smtp_enum”

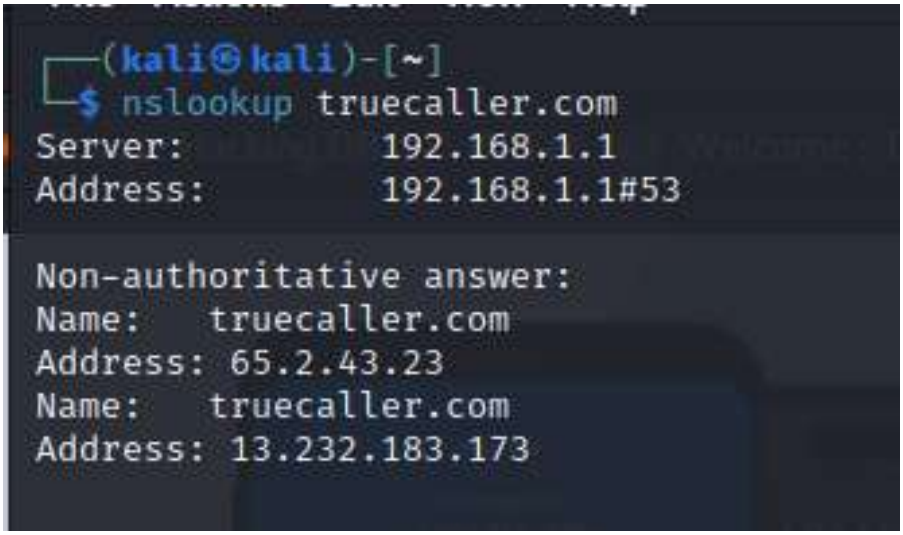




Fuzzing failed due to connection time out indicating inability to enumerate the service.

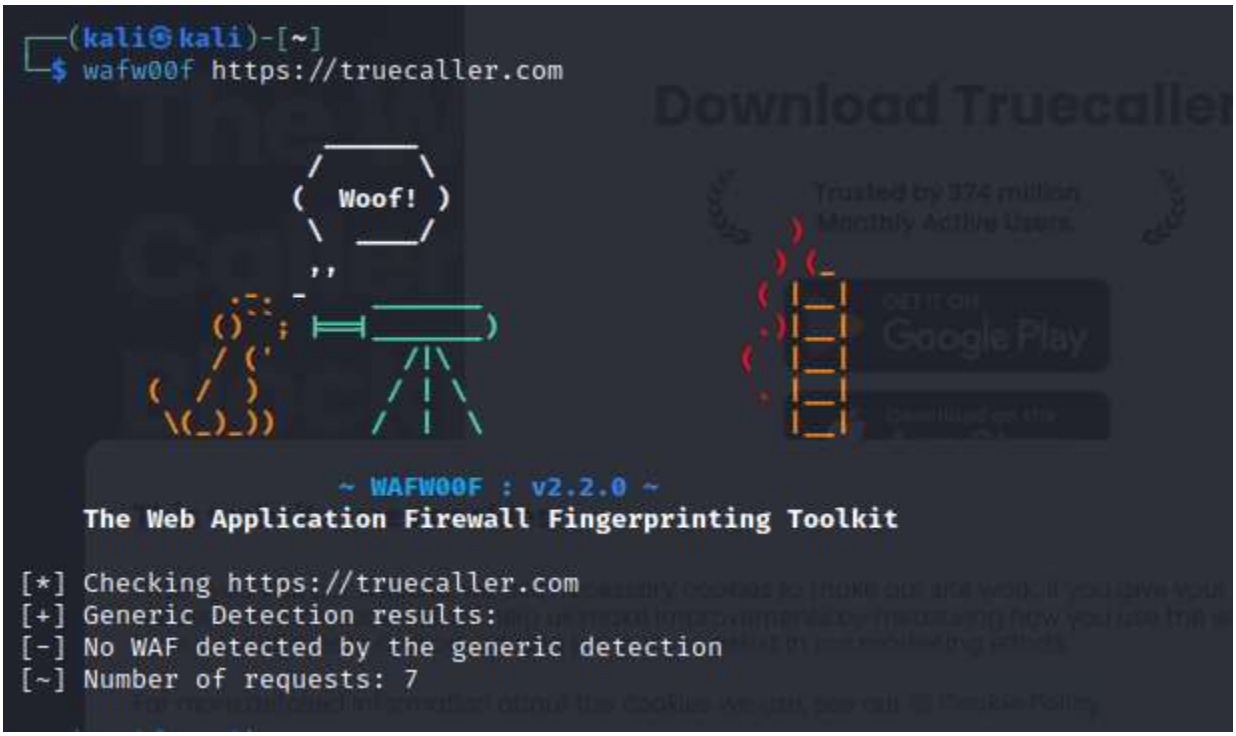
Nslookup

With the use of the command below, the ip address of the web applications can be found.



Wafw00f scan

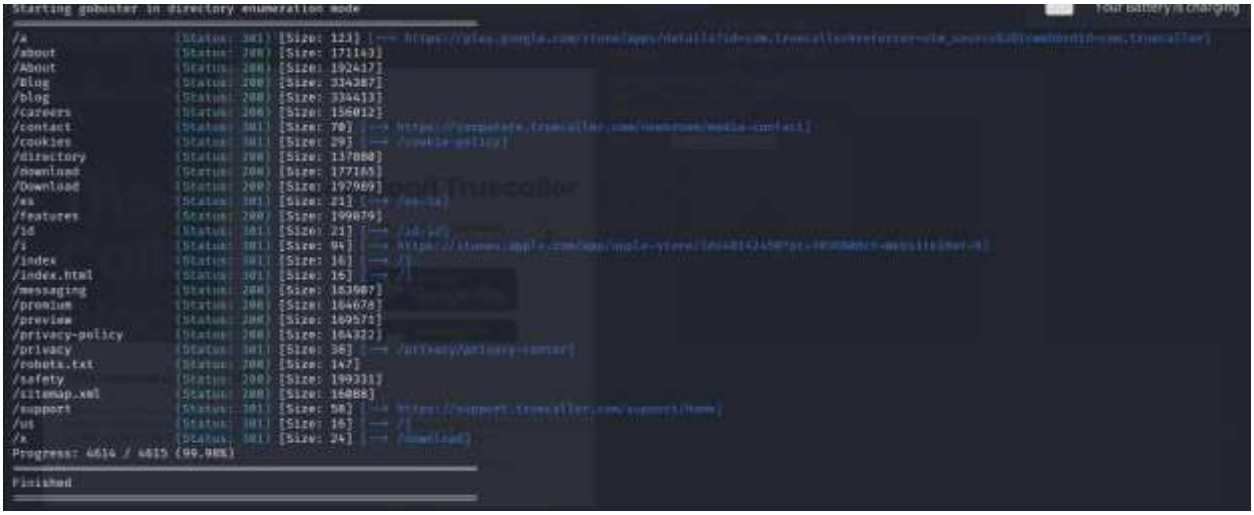
Used to identify the type of WAF that is used to protect the web application.



No WAF detected from the above scan.

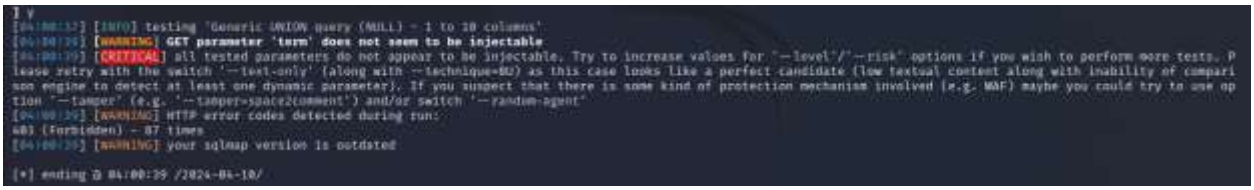
Gobuster scan

The gobuster scan helps to find hidden Directories, URLs, Sub-Domains, and S3 Buckets seamlessly.



Sqlmap scan

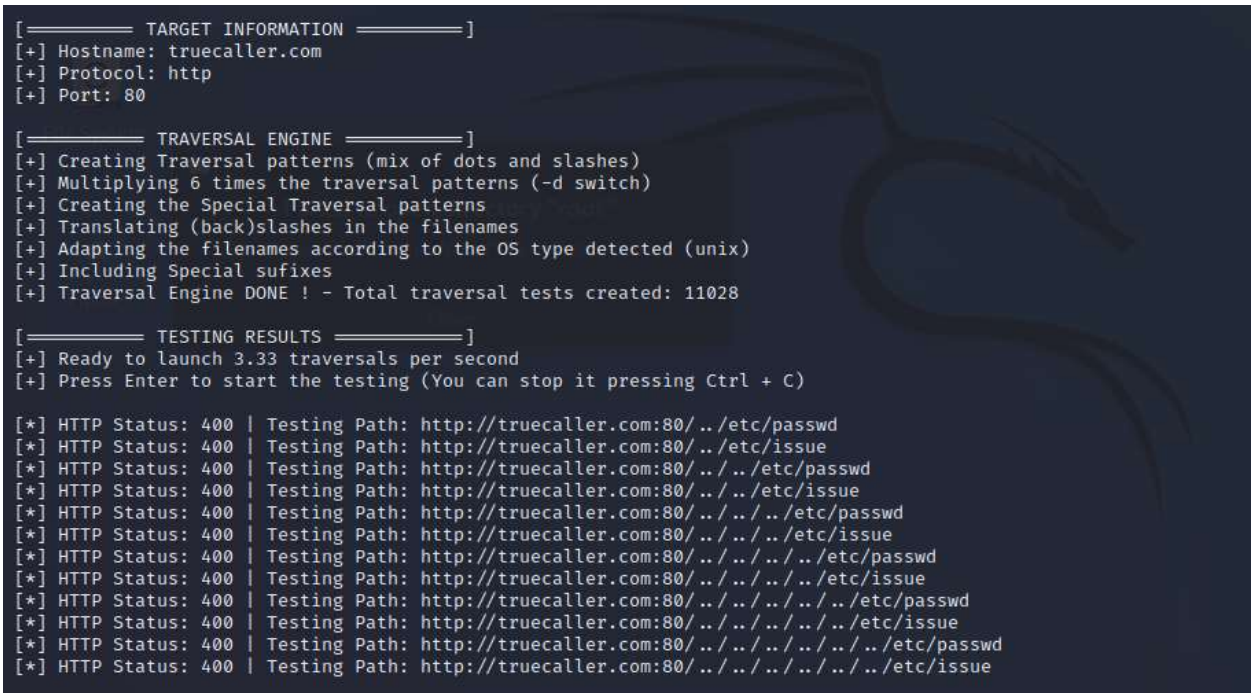
With the use of this scan, we can identify whether a sql injection can be done or not.



The above results prove that there is no injection vulnerability in the above web application.

Dotdotpwn scan

Dotdotpwn is a directory traversal checker.

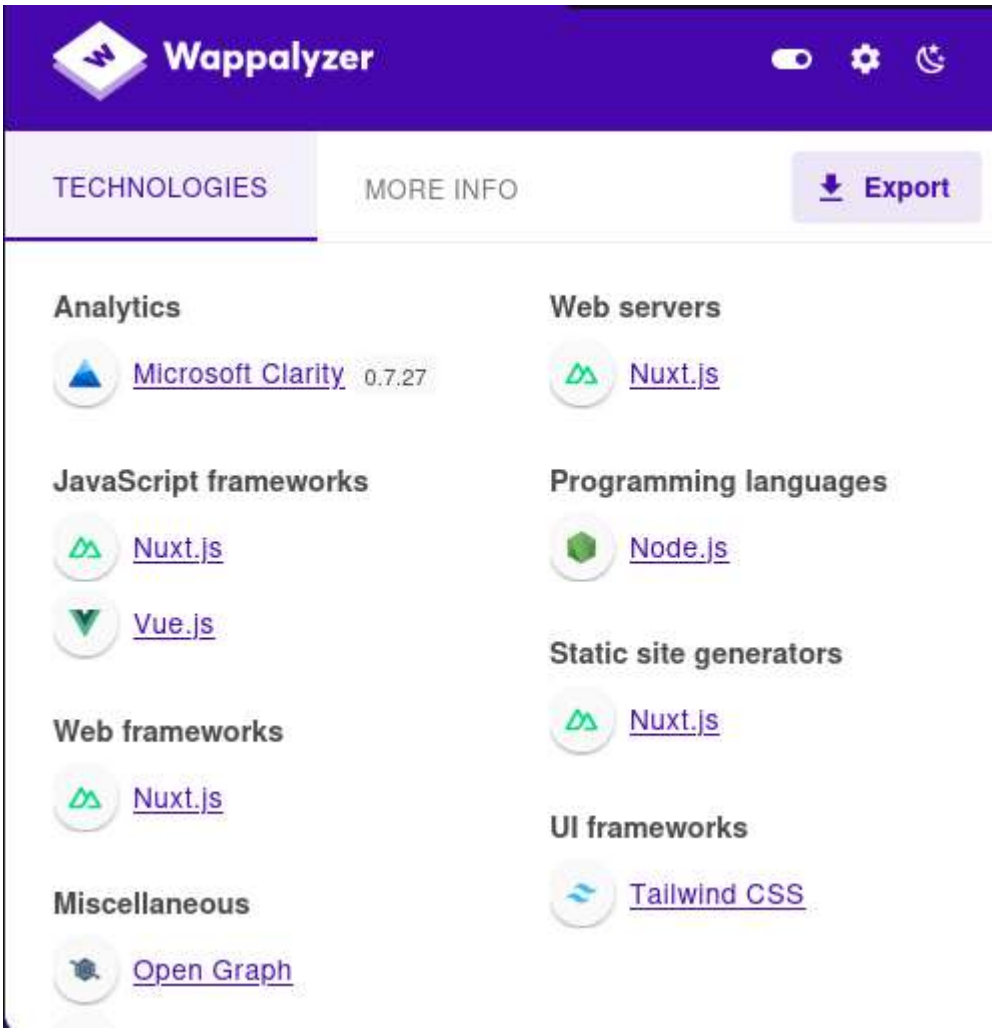


The scan results returned status codes within the range 400 (400-499). It shows a client error.

Therefore, we can conclude that the tested destinations are not vulnerable to a directory traversal.

Manual scanning -using Wapplyzer

The Wapplyzer is used to identify the technologies used in the web application.



No unusual/vulnerable versions are found.

Zap scan

With the use of the “active scan”, some potential vulnerabilities can be found.

		Confidence				
		User Confirmed	High	Medium	Low	Total
Risk	High	0 (0.0%)	0 (0.0%)	0 (0.0%)	1 (14.3%)	1 (14.3%)
	Medium	0 (0.0%)	1 (14.3%)	0 (0.0%)	2 (28.6%)	3 (42.9%)
	Low	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
	Informational	0 (0.0%)	0 (0.0%)	2 (28.6%)	1 (14.3%)	3 (42.9%)
	Total	0 (0.0%)	1 (14.3%)	2 (28.6%)	4 (57.1%)	7 (100%)

Risk=Medium, Confidence=Low (2)

http://truecaller.com (2)

Absence of Anti-CSRF Tokens (1)

▸ GET http://truecaller.com

Hidden File Found (1)

▸ GET http://truecaller.com/.hg

Risk=High, Confidence=Low (1)

http://truecaller.com (1)

Cloud Metadata Potentially Exposed (1)

▸ GET http://truecaller.com/latest/meta-data/

Risk=Medium, Confidence=High (1)

http://truecaller.com (1)

Content Security Policy (CSP) Header Not Set (1)

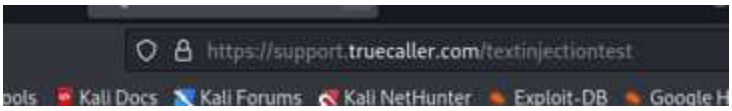
▸ GET http://truecaller.com

According to the above results the following can be found:

- Absence of anti-CSRF tokens.
- One hidden file found.
- Cloud meta data are potentially exposed.
- Content security policy (CSP) header not set.

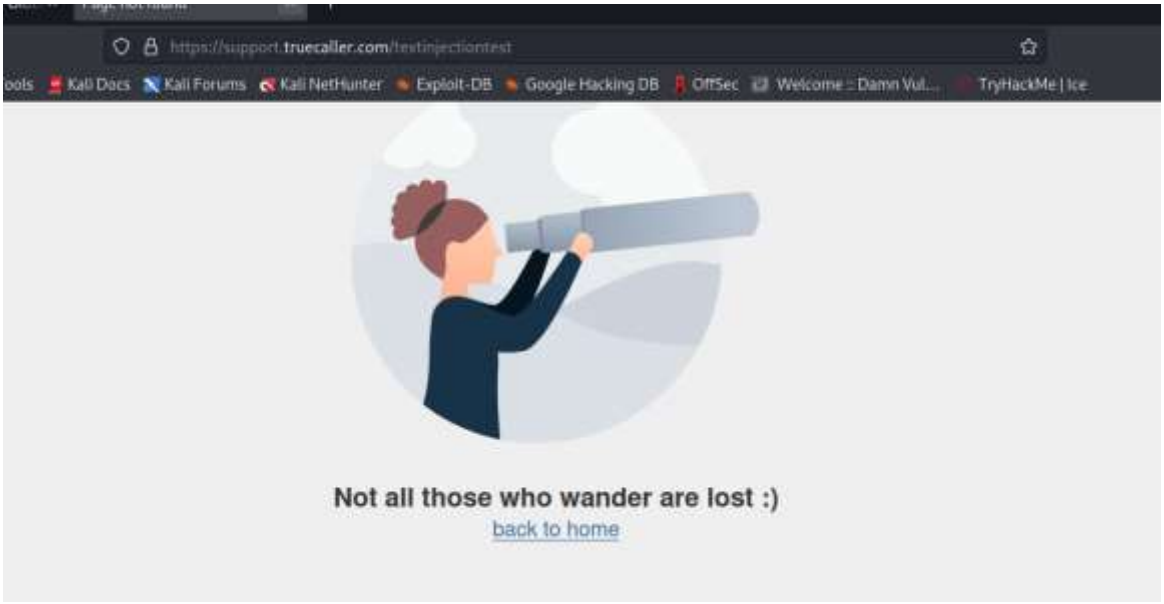
Text injection

An arbitrary string value is appended to the URL to see whether the web application is vulnerable towards a text injection.



If the entered text is reflected on the error response of the web page, there is a possibility to inject malicious content.

If not, the web application is safe.

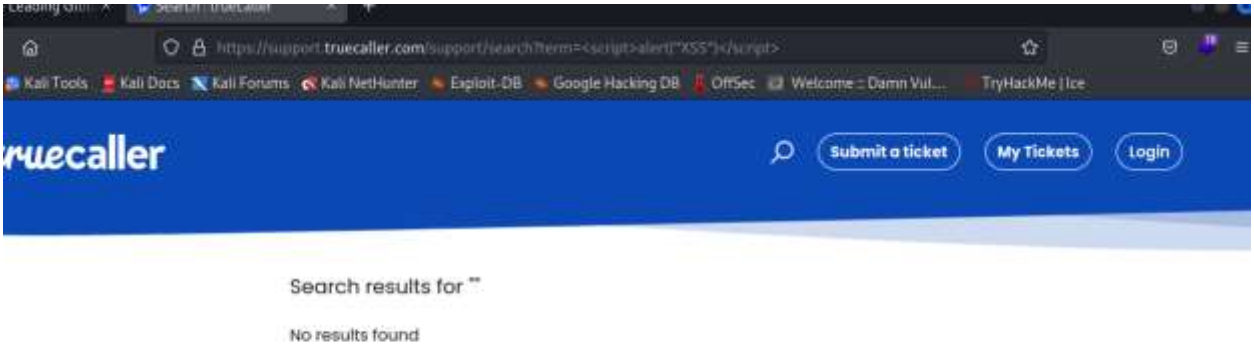


There is no possibility for a text injection.

XSS injection testing

This scan checks whether the web application is vulnerable to XSS injection.

There is a search function, and we can pass the parameters to the url and checked against xss injection.

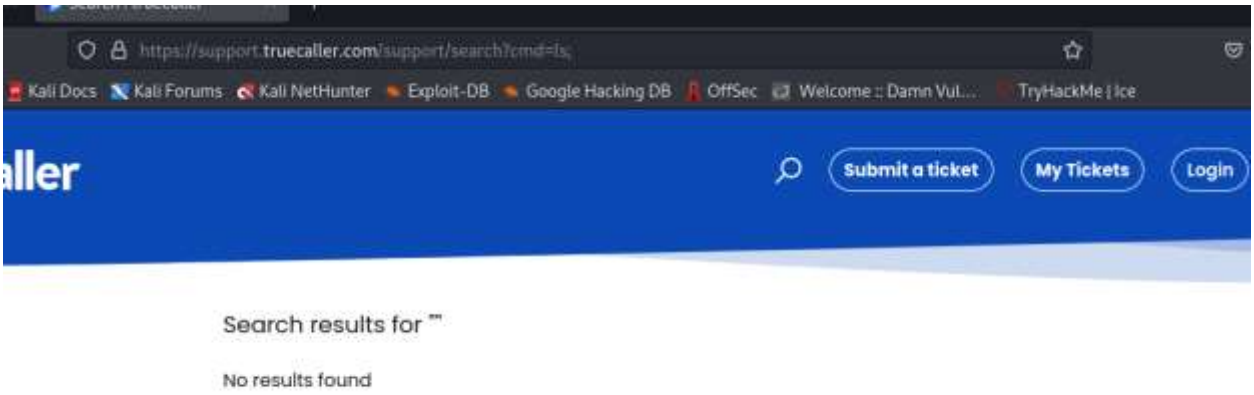


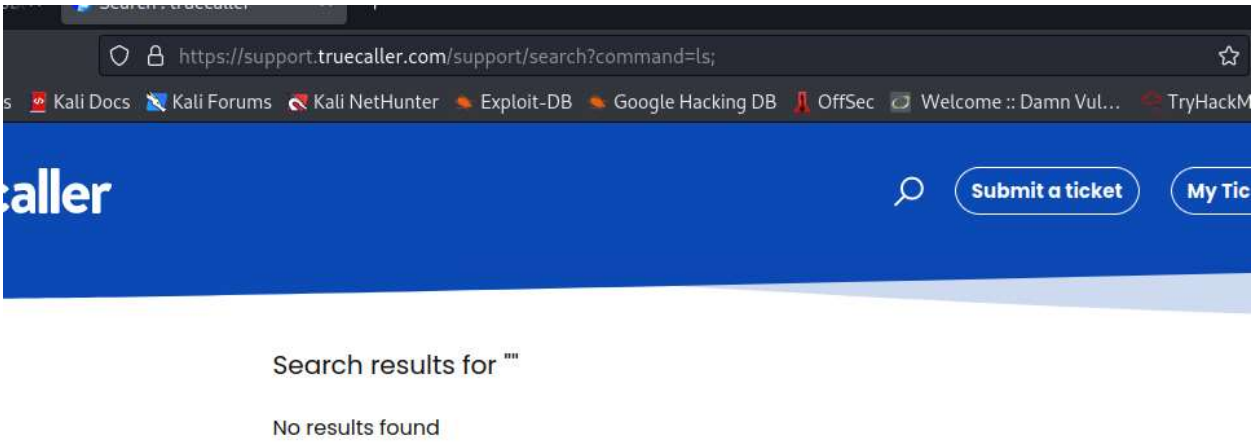
It is safe from reflective xss.

Command injection testing

The query that is used for searching is used against this vulnerability.

The “ls” command is appended to the url.





No command injection vulnerability can be found.