Sri Lanka Institute of Information Technology

BUG BOUNTY REPORT - 9

Web Security – IE2062

IT22362780

Jayaweera N.S

Report Details

Report # - 09

Domain - https://pixabay.com

Platform -bugcrowd.com

Scans performed -

Recon-ng scan

Nmap scan

Wafw00f scan

Dotdotpwn scan

Nikto scan

Sqlmap scan

Manual scanning using Wapplyzer

Text injection

File upload vulnerability testing

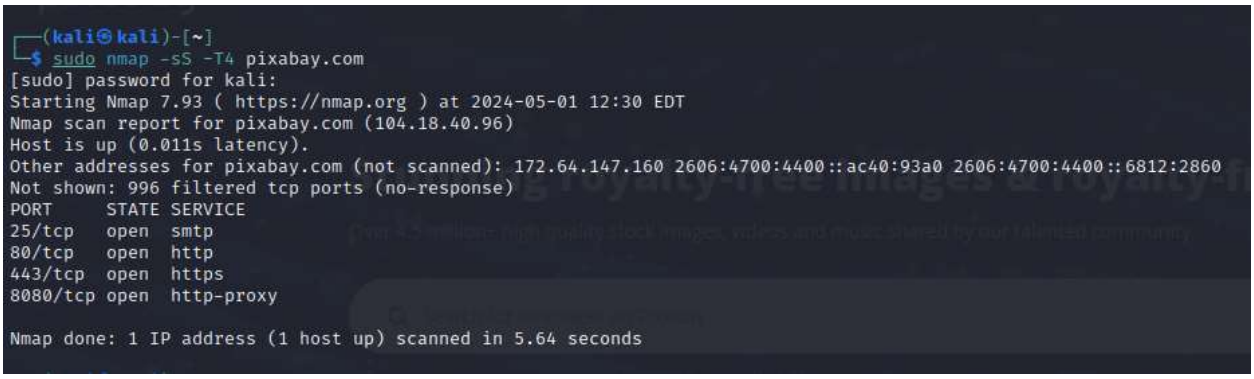
Command injection

XSS injection

Nslookup scan

Nmap scan

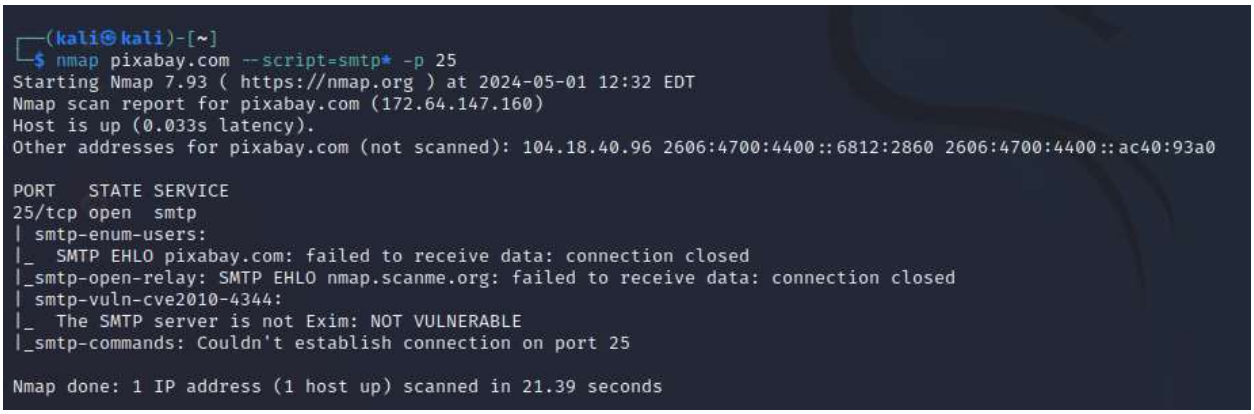
Using nmap scan all the open ports in the target can be identified.



No unusual ports are open.

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

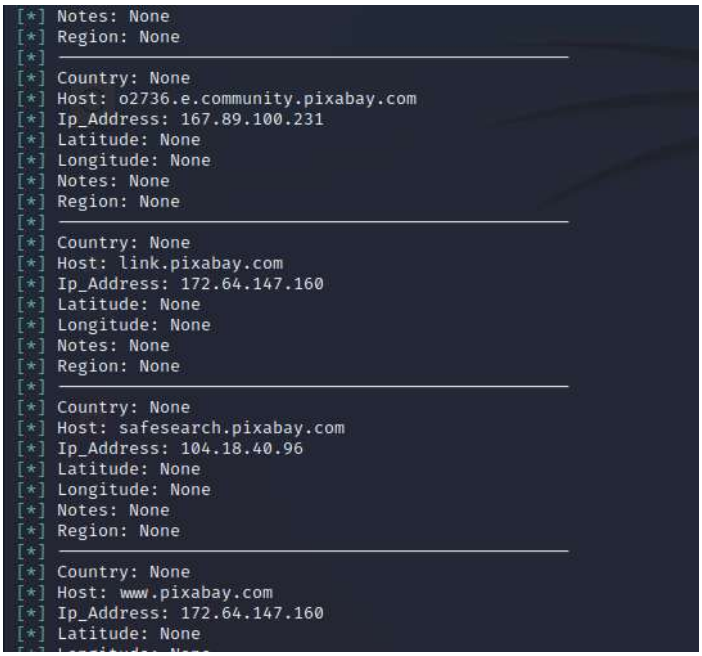
Let's see if we can establish a connection on port 25.

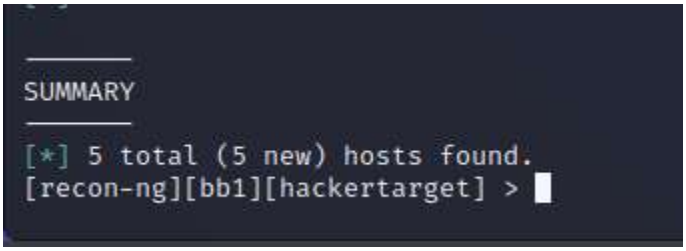


Connection cannot be established therefore a vulnerability cannot be identified.

Recon-ng

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

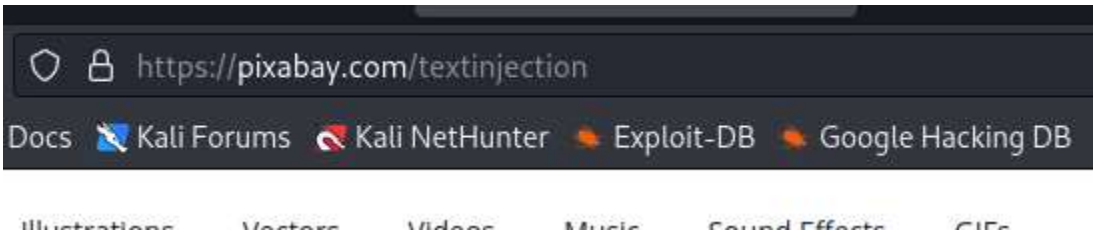




5 total subdomains found.

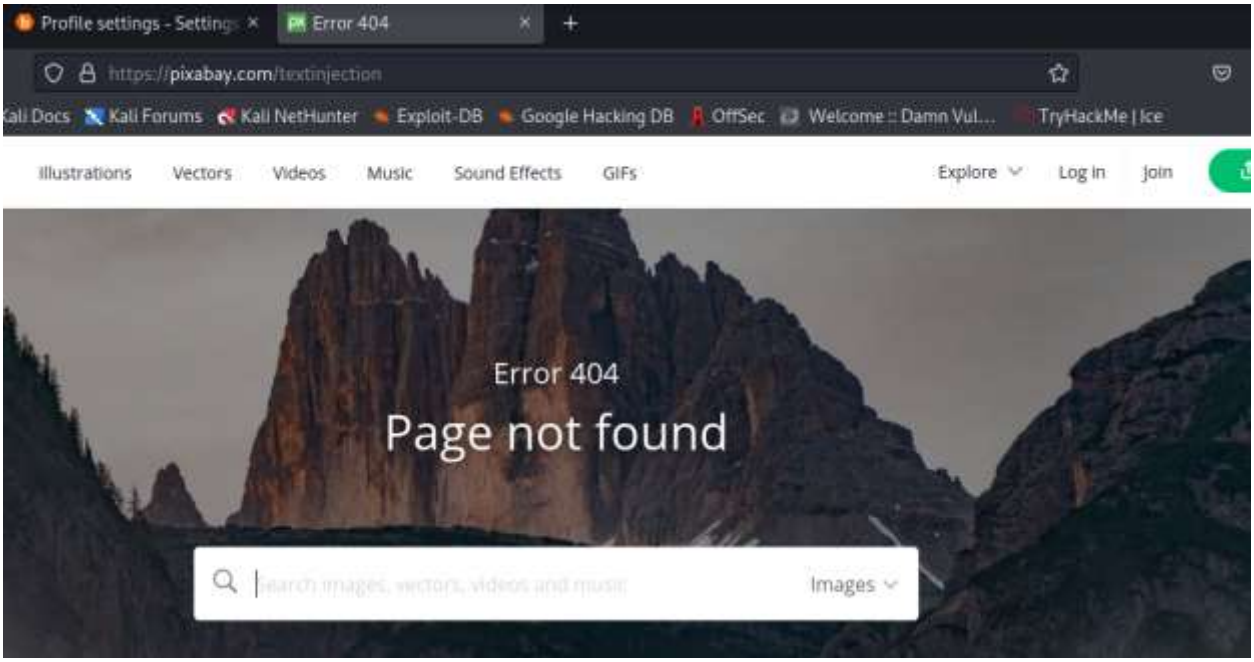
Text injection

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.

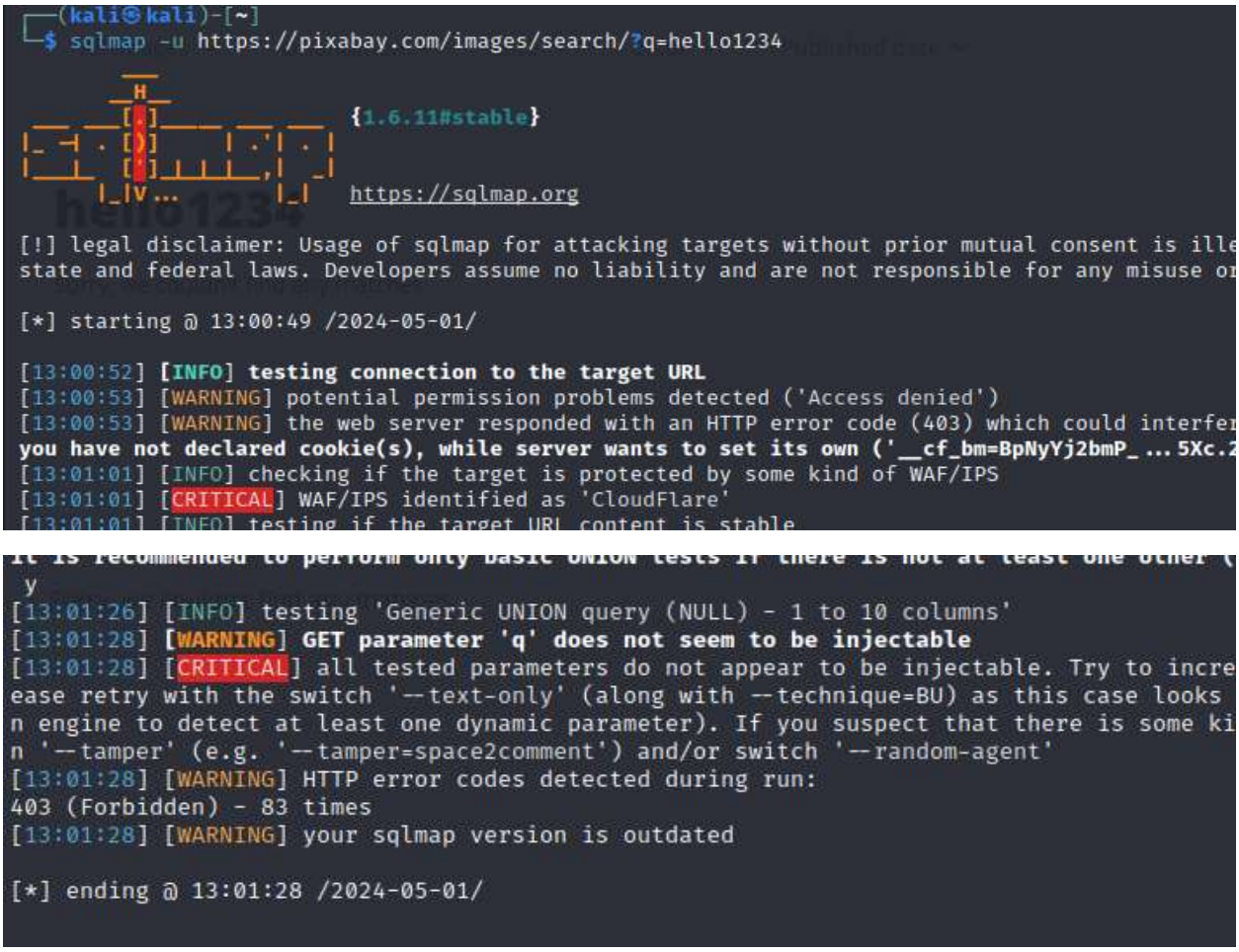
If not, the web application is safe.



No text injection vulnerability can be found.

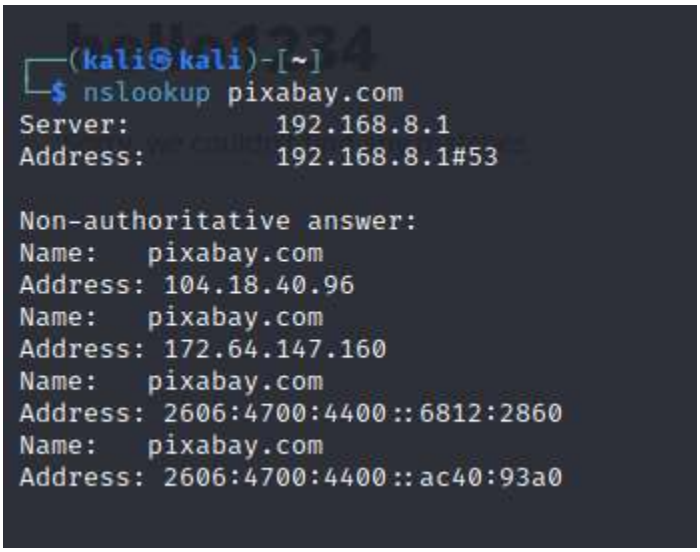
Sqlmap

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



There is no injection vulnerability in the above web application.

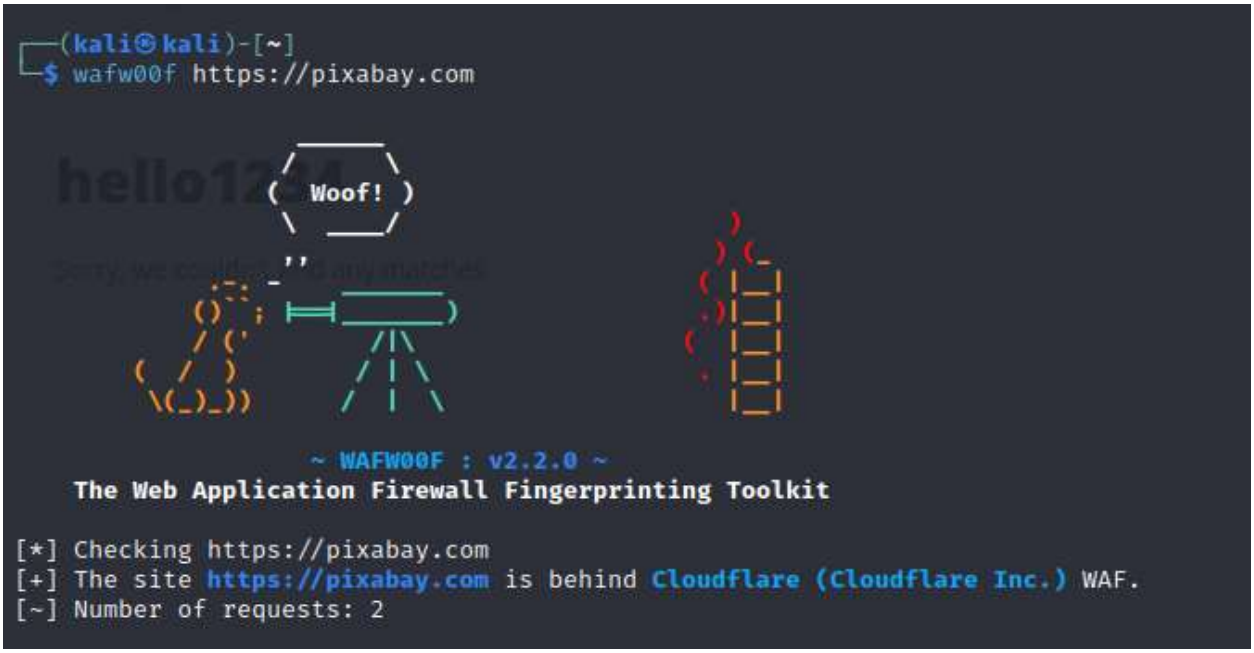
Nslookup



The ip address of the domain is found.

Wafw00f

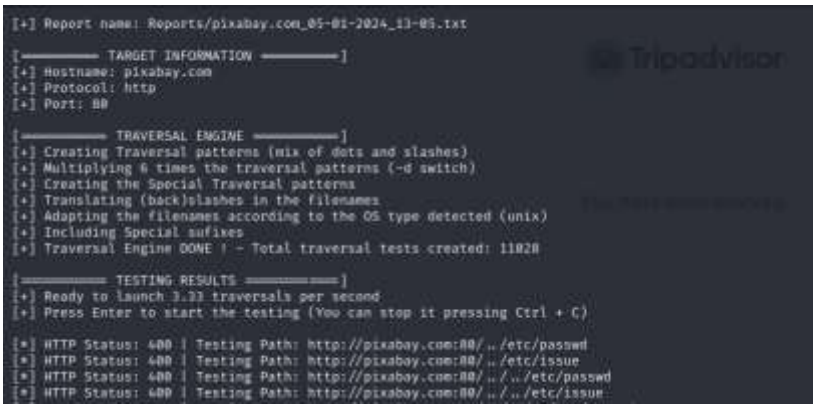
Used to identify the type of WAF that is used to protect the web application.



The WAF used : Cloudflare

Dotdotpwn

Dotdotpwn is a directory traversal checker.

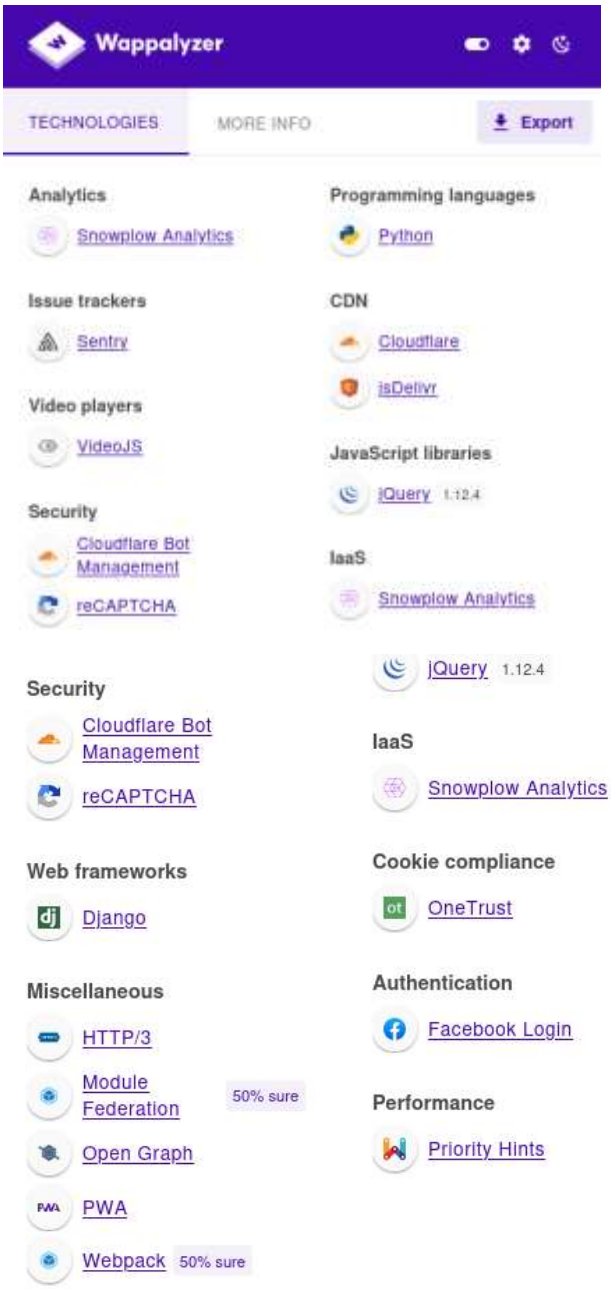


The scan results returned status codes within the range 400 (400-499). It shows a client error.

Vulnerable paths were found.

Wapplyzer

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



These are the technologies used.

The jquery 1.12.4 used is vulnerable.

M

Cross-site Scripting (XSS)

<1.12.0

>=1.12.3 <3.0.0-beta1

jquery is a package that makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers.

Affected versions of this package are vulnerable to Cross-site Scripting (XSS) attacks when a cross-domain ajax request is performed without the `dataType` option causing `text/javascript` responses to be executed.

Note: After being implemented in version 1.12.0, the fix of this vulnerability was reverted in 1.12.3, and then was only reintroduced in version 3.0.0-beta1. The fix was never released in any tag of the 2.x.x branch, as it was reverted out of the branch before being released.

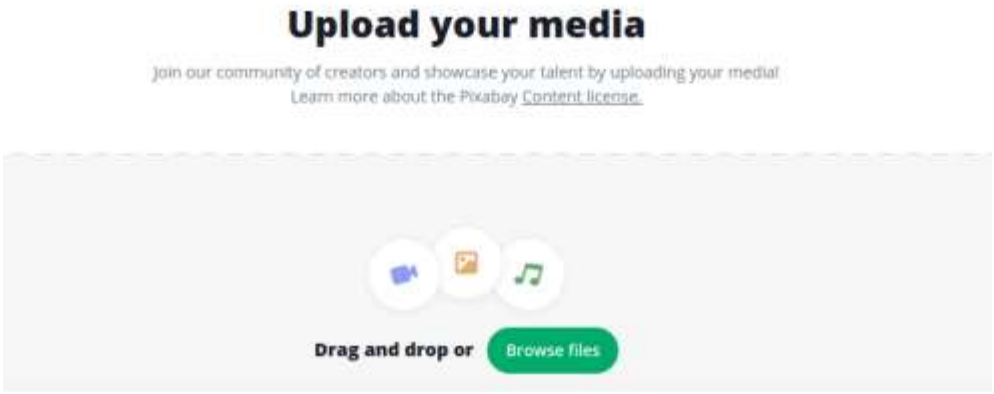
Note: CVE-2017-16012 is a duplicate of CVE-2015-9251

How to fix Cross-site Scripting (XSS)?
Upgrade `jquery` to version 1.12.0, 3.0.0-beta1 or higher.

In order to mitigate the risk, update the jquery scripts to 1.12.0, 3.0.0-beta1 or higher.

File upload vulnerability

If a .php file can be uploaded from the file uploading facility, there is a possibility to upload and execute a reverse shell php code.

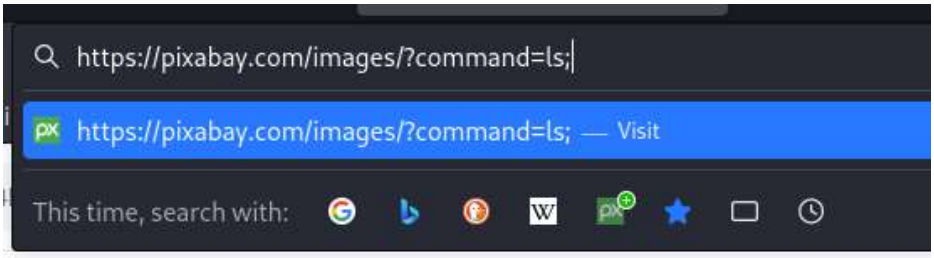


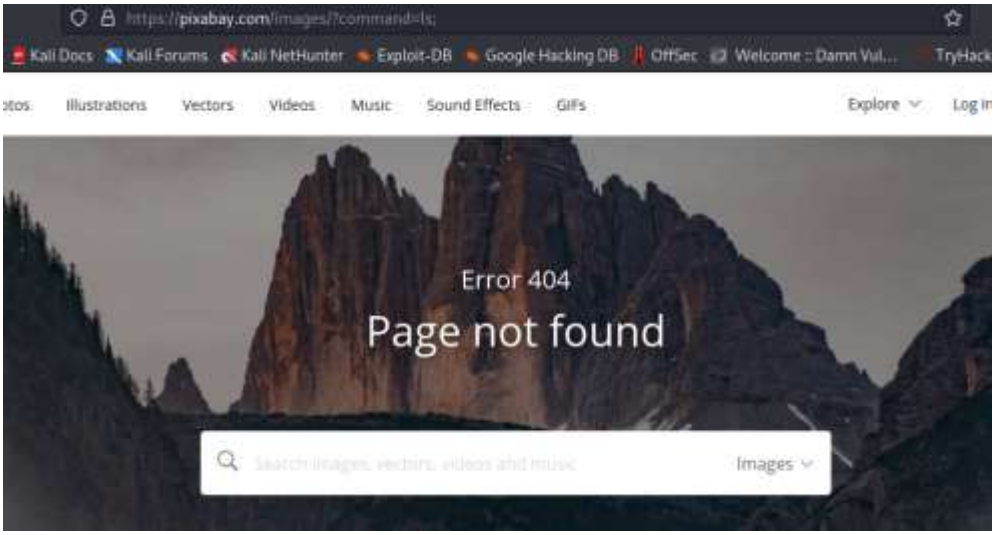
No vulnerability found.

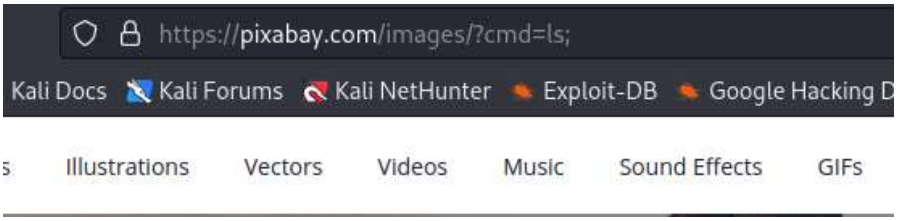
Command injection

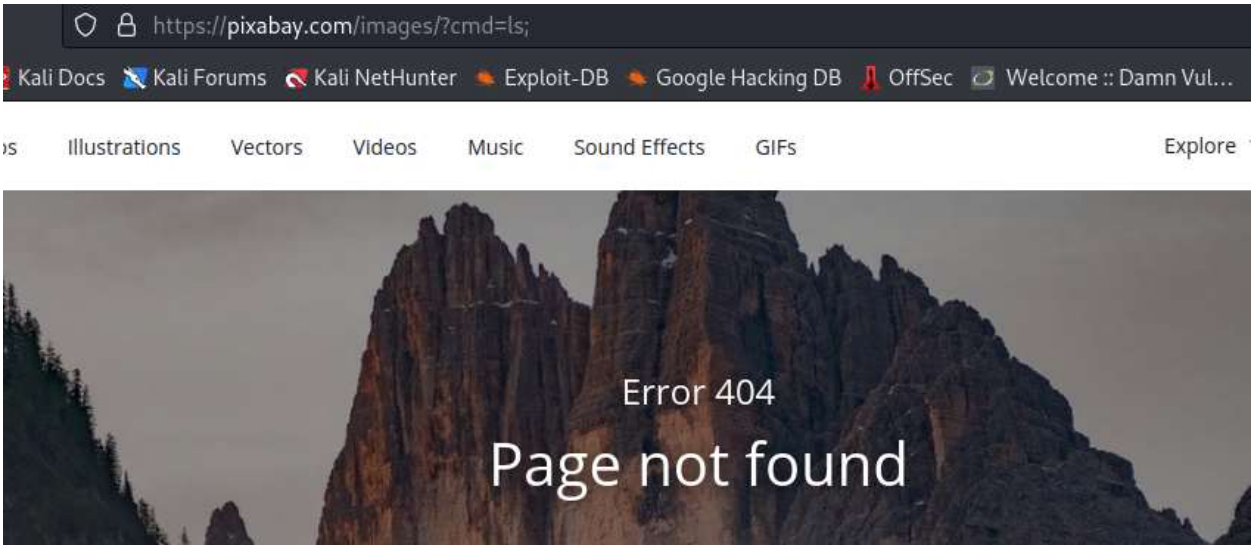
The query that is used for searching is used against this vulnerability.

The “ls” command is appended to the url





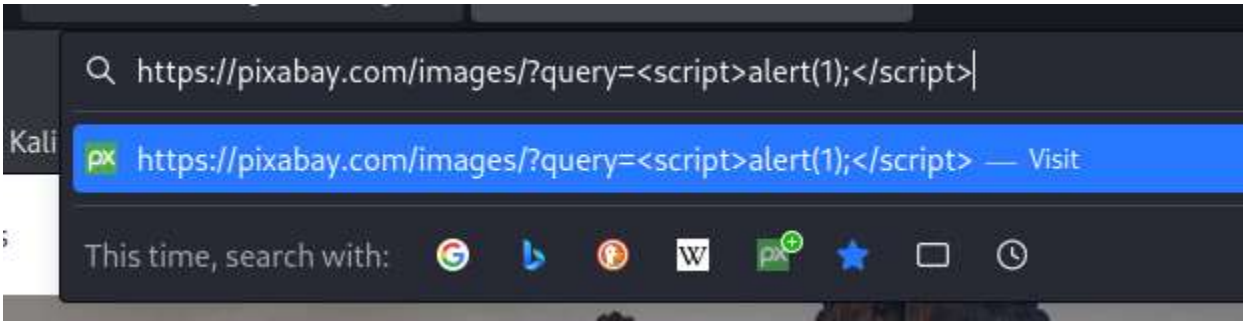


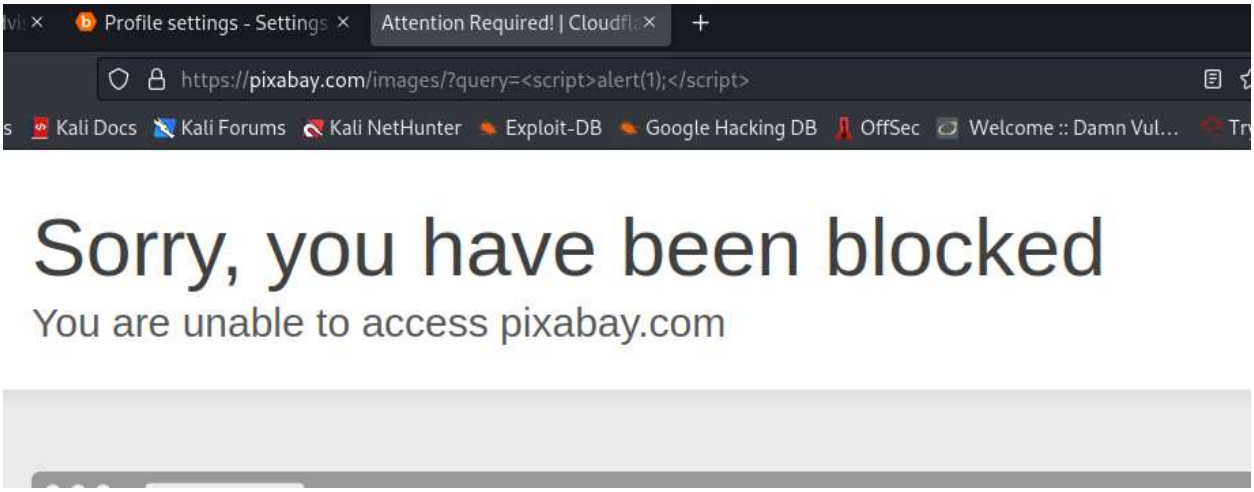


No command injection vulnerability can be found.

XSS injection

A payload is apended to the url to test against xss injection.





It’s not vulnerable to XSS injection.