**CYBERSECURITY PROJECT**

**WEB APPLICATION ANALYSIS TOOLS**

**SUBMITTED BY -**

**SREENIDHI GANACHARI**

**19BCE7230**

**SUBMITTED TO –**

**DR. KUMAR DEBASIS**

**1 Introduction**

Web Application Analysis refers to all applications that are accessed through a browser. Security testing for your application is very important if data leaks or modifications are unacceptable and intolerable. For example, if a e-commerce applications, which sometimes involve banking transactions, security testing is critical. It should also ensure that sufficient authentication and authorization mechanisms are in place.

Web security testing aims to find security vulnerabilities in Web applications and their configuration. The primary target is the application layer (i.e., what is running on the HTTP protocol). Testing the security of a Web application often involves sending different types of input to provoke errors and make the system behave in unexpected ways. These so called “negative tests” examine whether the system is doing something it isn’t designed to do.

It is also important to understand that Web security testing is not only about testing the security features (e.g., authentication and authorization) that may be implemented in the application. It is equally important to test that other features are implemented in a secure way (e.g., business logic and the use of proper input validation and output encoding). The goal is to ensure that the functions exposed in the Web application are secure.

Security testing can be static or dynamic-

Static testing involves doing a static code analysis to check for any vulnerabilities. The goal is to understand the code flow and check for security threats by walking through the code.

Dynamic testing entails running the application to see if the response is as expected for the associated request. It is very similar to black box testing.In this project various web applications tools have been discussed and demonstrated .

**The tools demonstrated in this project are –**

1) Skipfish

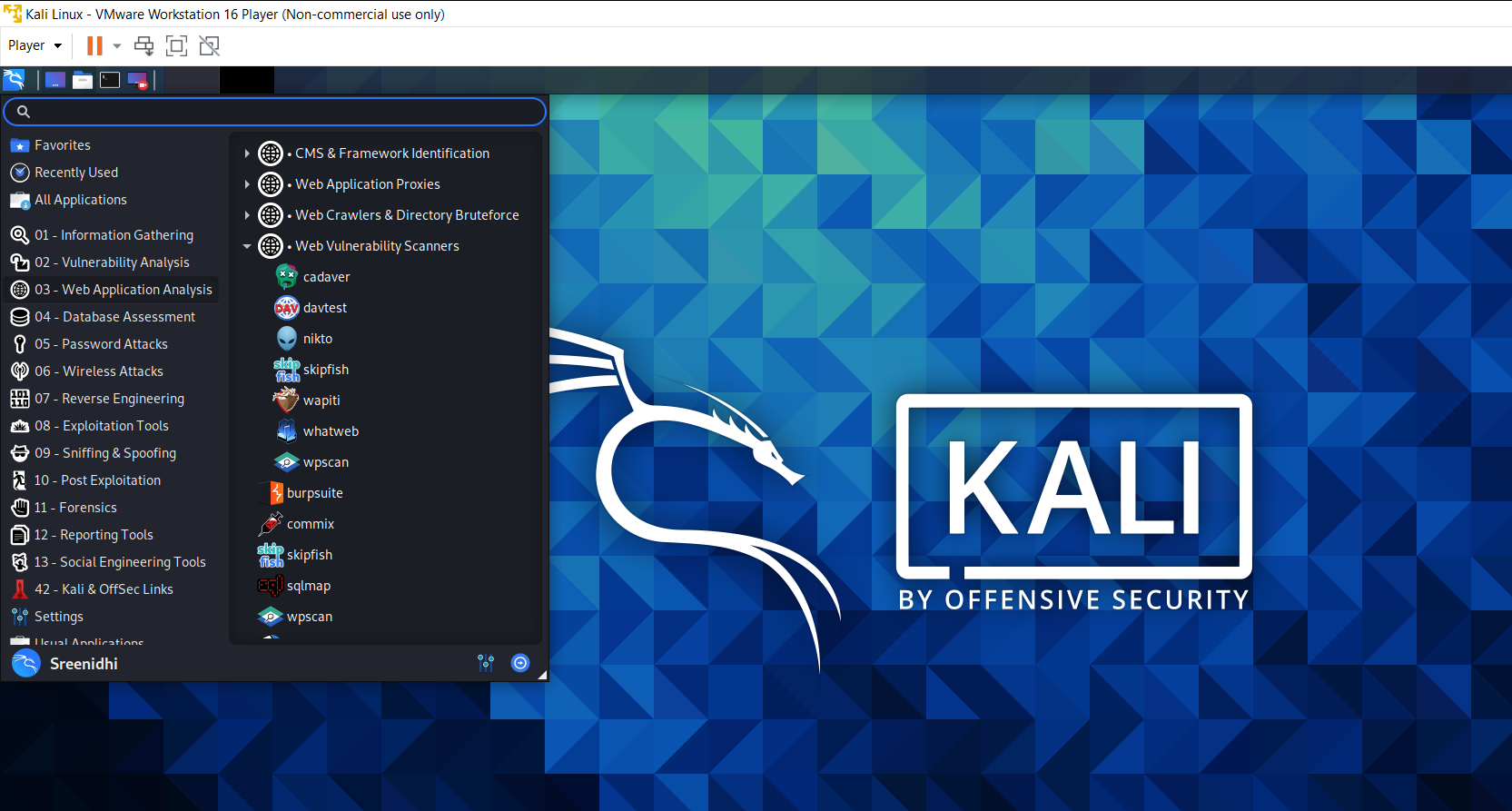
2) Whatweb

3) WPScan

4) Nikto



The tools highlighted in red are used in this project



The tools highlighted in red are used in this project

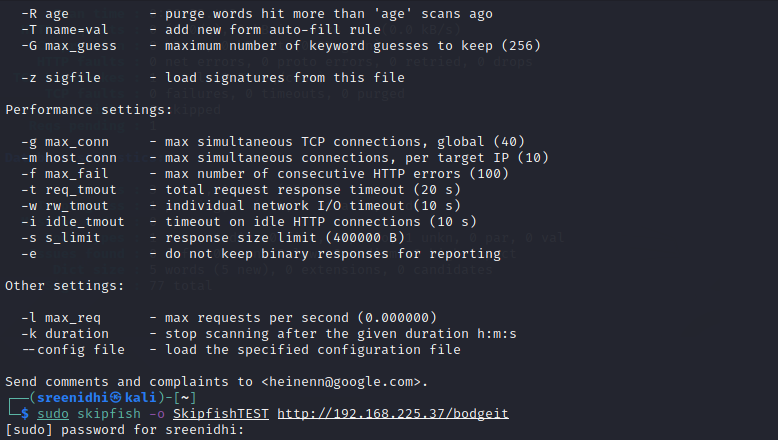
**1.1 Skipfish**

In Kali Linux, Skipfish is an active web application security reconnaissance tool. It uses a recursive crawl and dictionary-based probes to create an interactive sitemap for the chosen site. The resulting map is then annotated with the output of several active (but hopefully non-disruptive) security checks. The tool's final report is intended to be used as a starting point for professional web application security evaluations.

Skipfish is a free and open-source Automated Penetration Testing (APT) tool for security researchers that can be found on GitHub. Skipfish is used for information gathering and testing the security of websites and web servers. Skipfish is one of the most user-friendly and effective penetration testing tools available. It comes with several integrated tools for penetrating testing the target system. This tool is also called an active web application security reconnaissance tool. This tool works and maps the target site's console using recursive crawls and dictionary-based probes. This tool displays all of the active security checks in the domain. Finally, this tool creates a report that can be utilized for security assessments.

**Features of Skipfish tool**

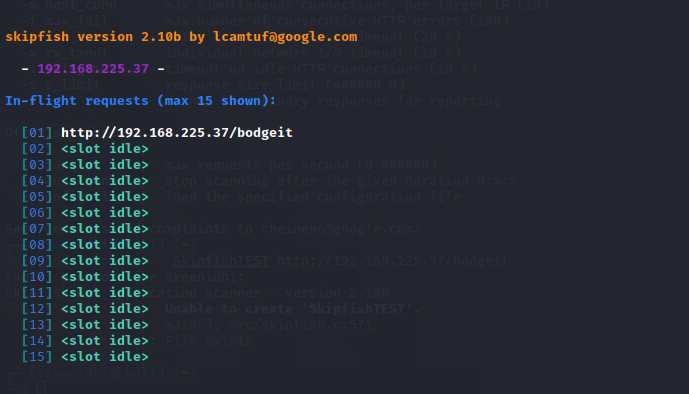
* Skipfish can track enumeration.
* Skipfish is used to scanning websites and web apps.
* Skipfish is an Open-source intelligence tool.
* Skipfish contains various modules such as wananga, metagoofil, etc.
* Skipfish may detect vulnerabilities (CMS), such as WordPress, Joomla, and others.
* There are more than 15 modules present in Skipfish which can be used for penetration testing.
* We used Skipfish to scan content management systems (CMS).
* Cutting-edge security logic: low false positive, capable of detecting a wide range of subtle defects, high quality, differential security checks, including blind injection vectors.
* Ease of Use: Heuristics to support a wide range of odd web frameworks and mixed-technology sites, including automatic learning capabilities, on-the-fly wordlist creation, and form auto-completion.
* High speed: pure C code, highly optimized HTTP handling, minimal CPU footprint- easily achieving 2000 requests per second with responsive targets.



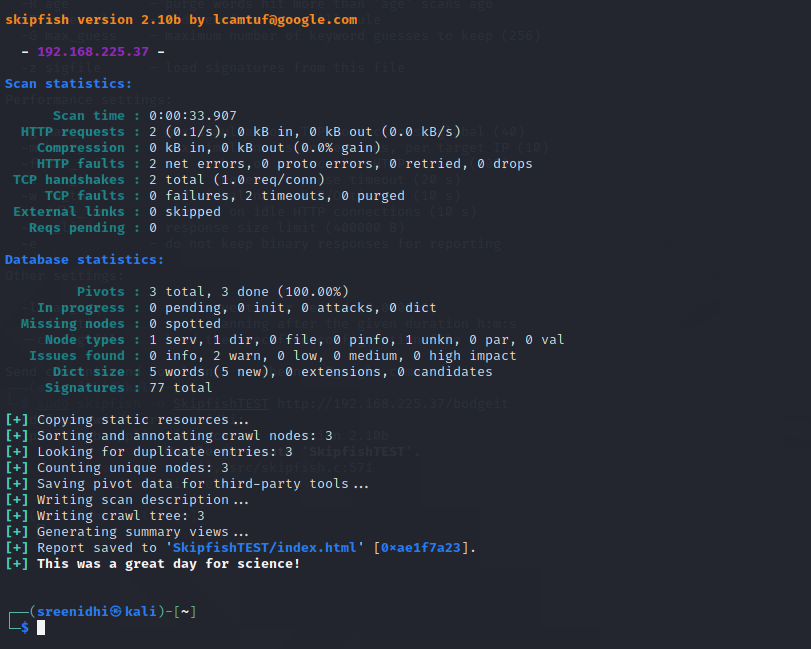
1) Using the command skipfish we have tried to find out the possible vulnerabilities



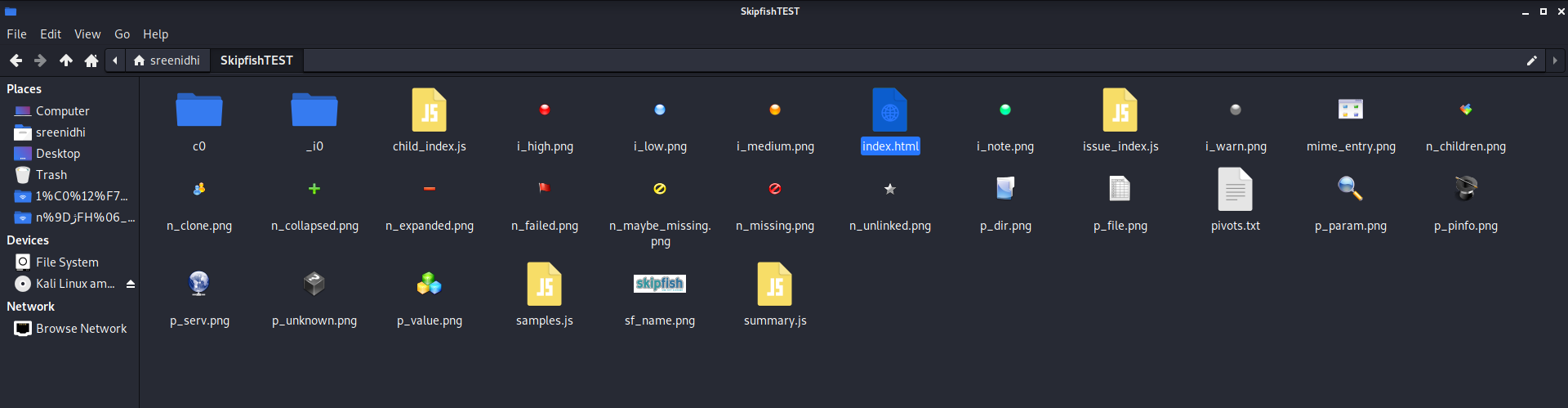
2) The command is executed and it displays the requests that have been sent

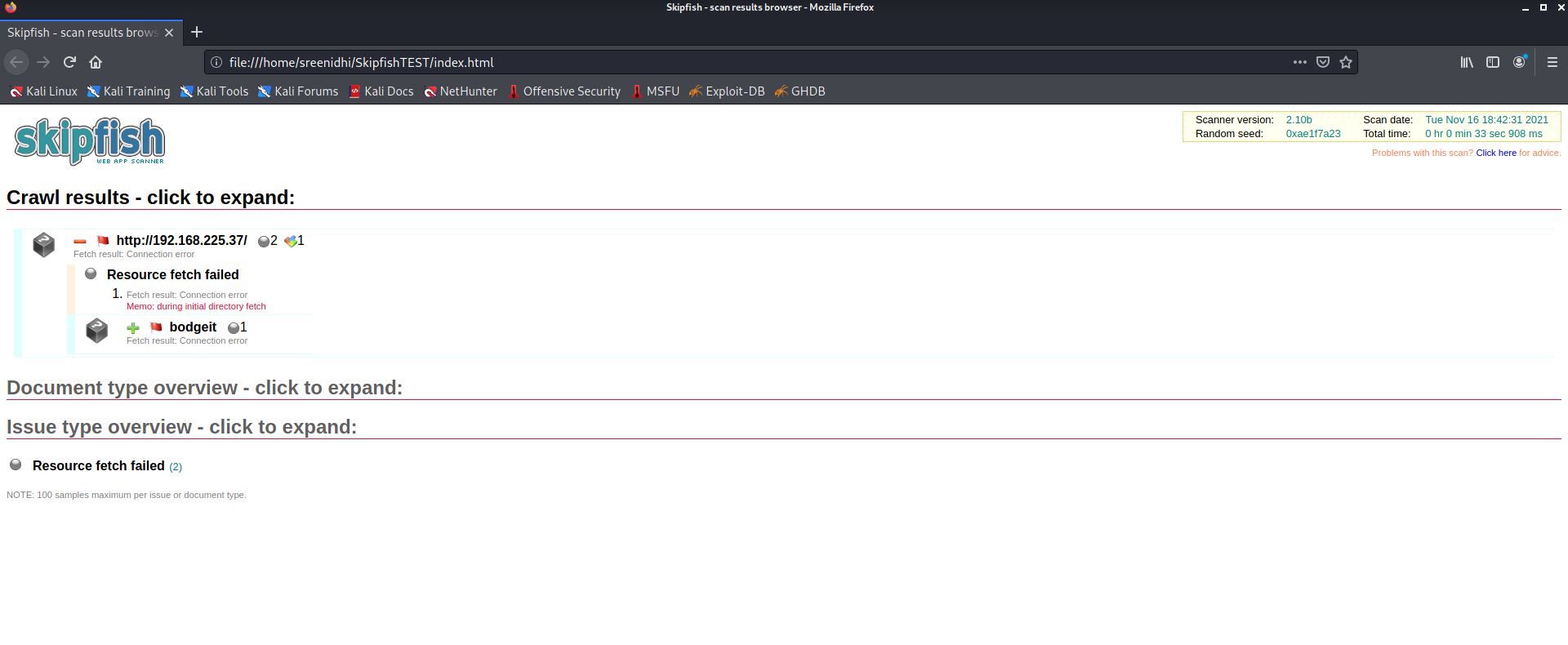


3) After sending requests , it gets some information such as the TCP Handshakes , HTTP requests , HTTP faults , missing nodes etc .

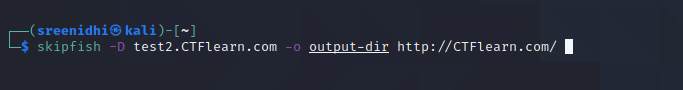


4) In the file created “SkipfishTEST” we view the index.html file to check the vulnerabilities identified .

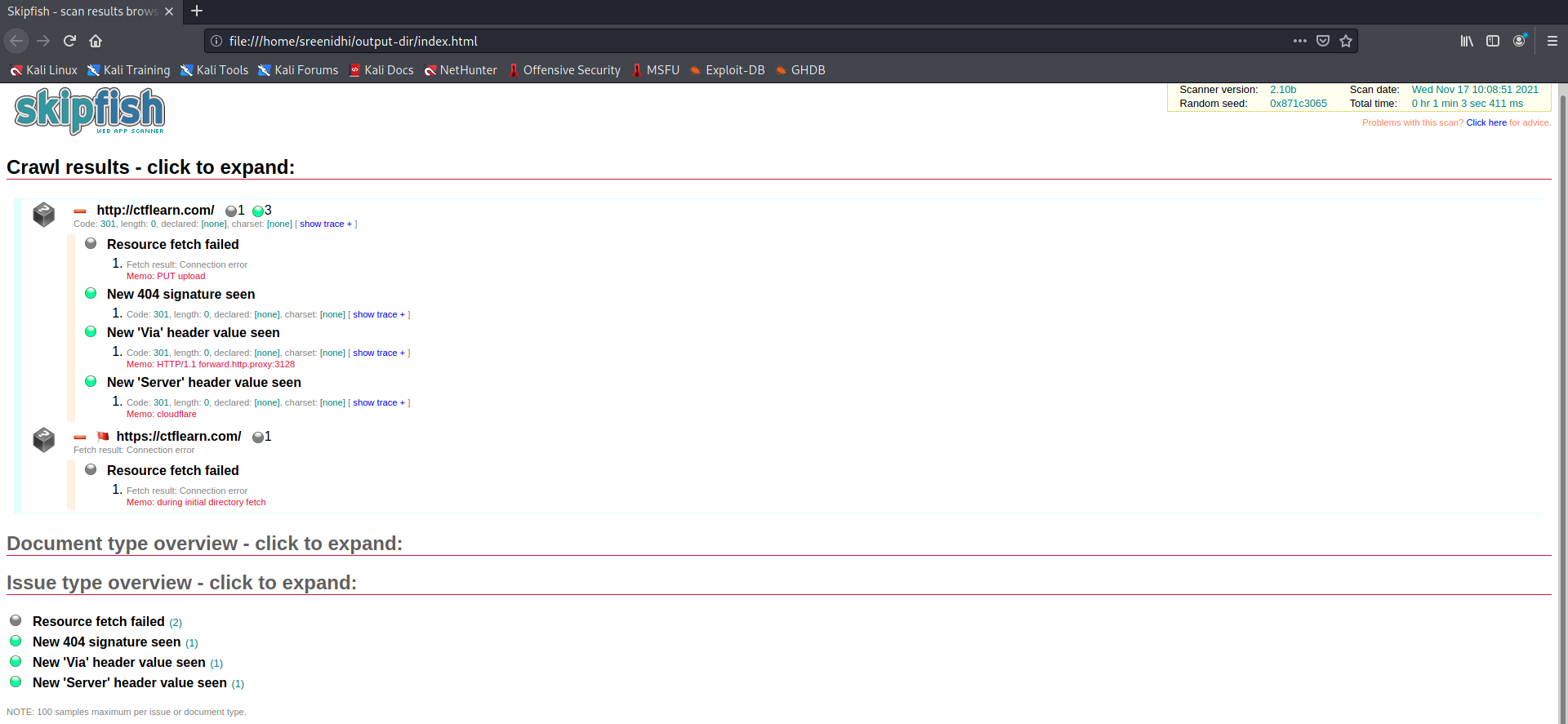




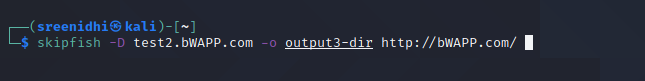
5) The same process is being with a URL to find the vulnerabilities



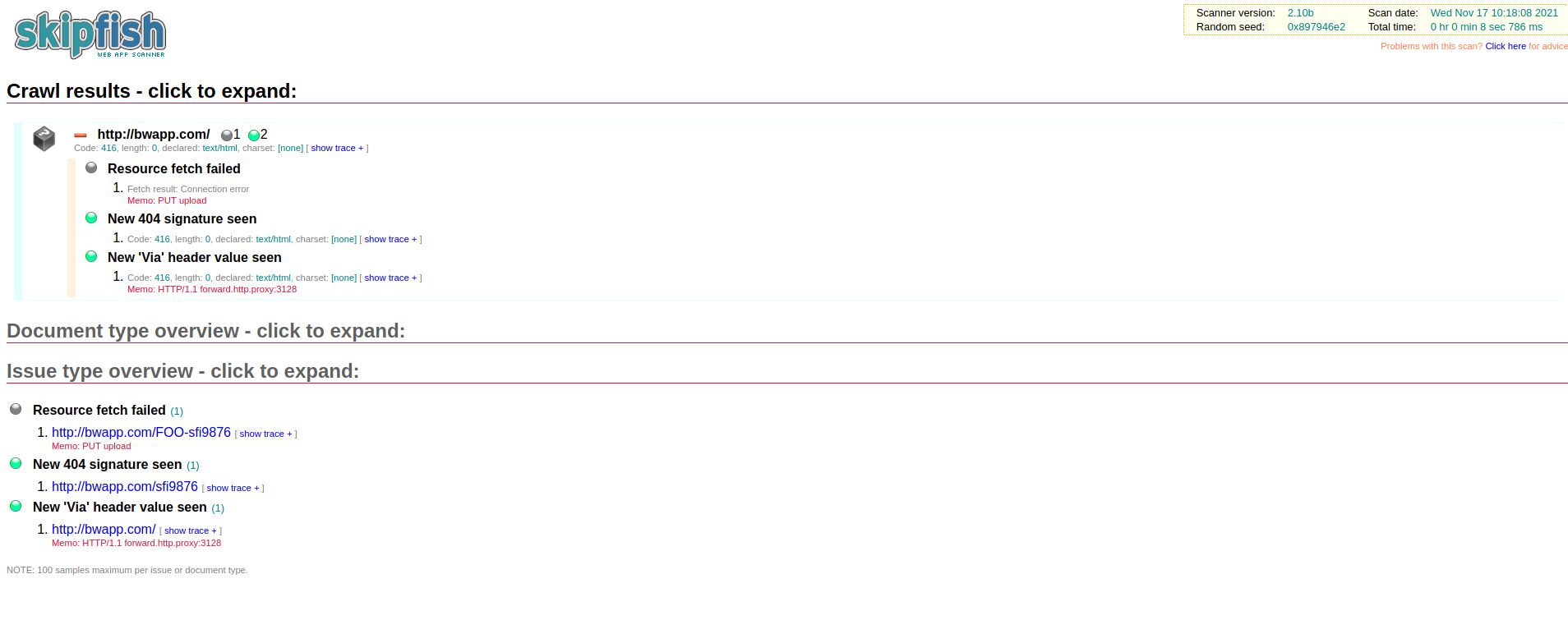




3) Considering some vulnerable website , we have further tested for other vulnerabilities using skipfish tool .



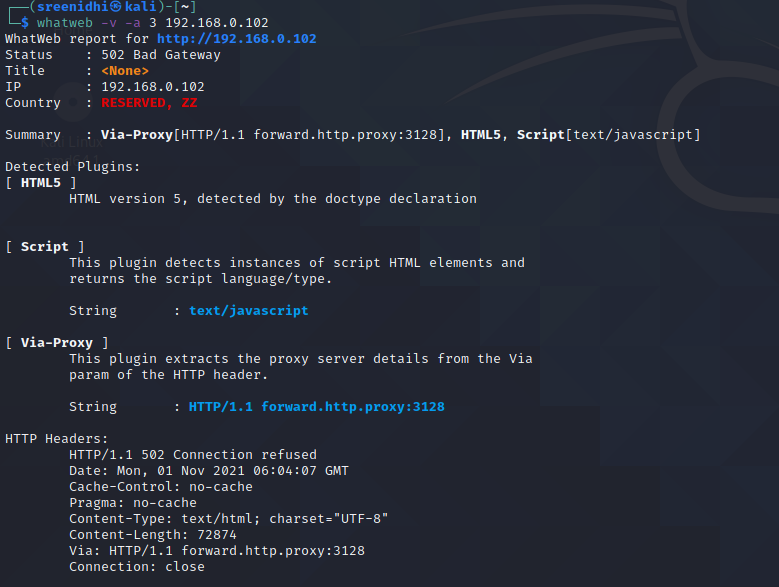




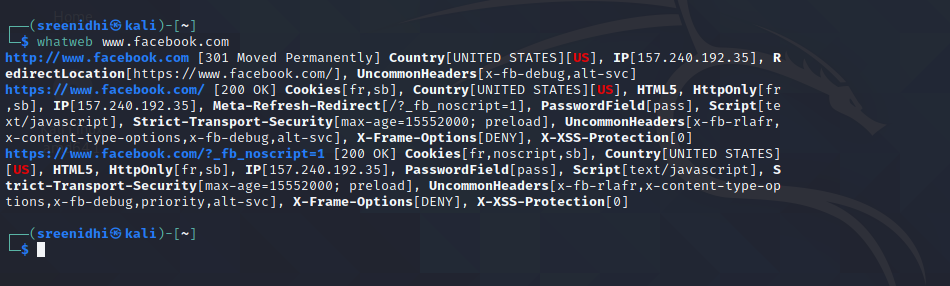
**1.2 WhatWeb**

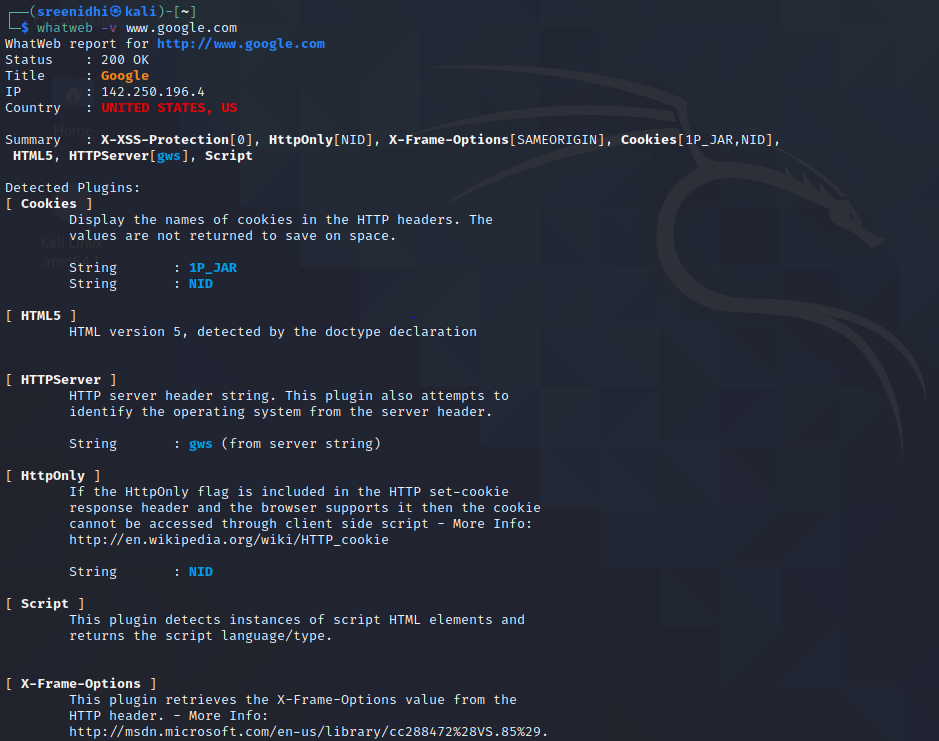
WhatWeb identifies websites. Its goal is to answer the question, "What is that Website?". WhatWeb recognises web technologies including content management systems (CMS), blogging platforms, statistic/analytics packages, JavaScript libraries, web servers, and embedded devices. WhatWeb has over 1800 plugins, each to recognise something different. WhatWeb also identifies version numbers, email addresses, account IDs, web framework modules, SQL errors, and more.

1) Using these commands of whatweb we find out the plugins

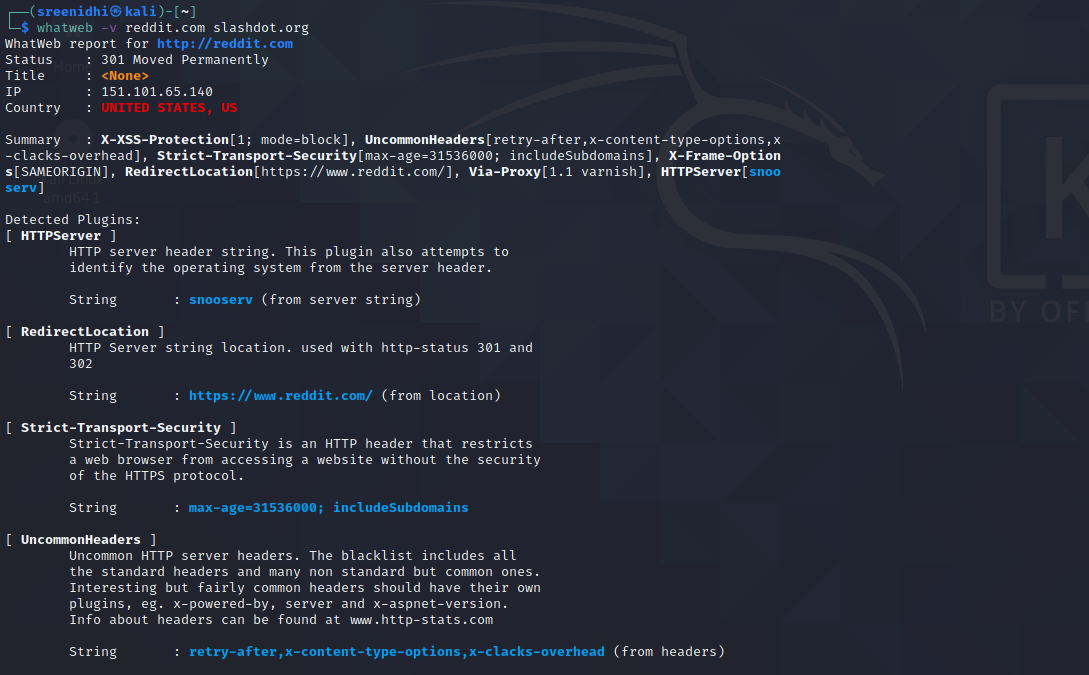


2) If we enter the URL we get details such as the type of protection , frame options and other HTML details .

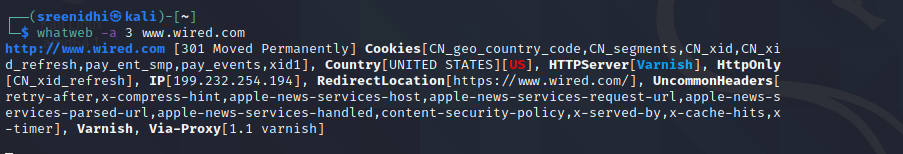


3) 

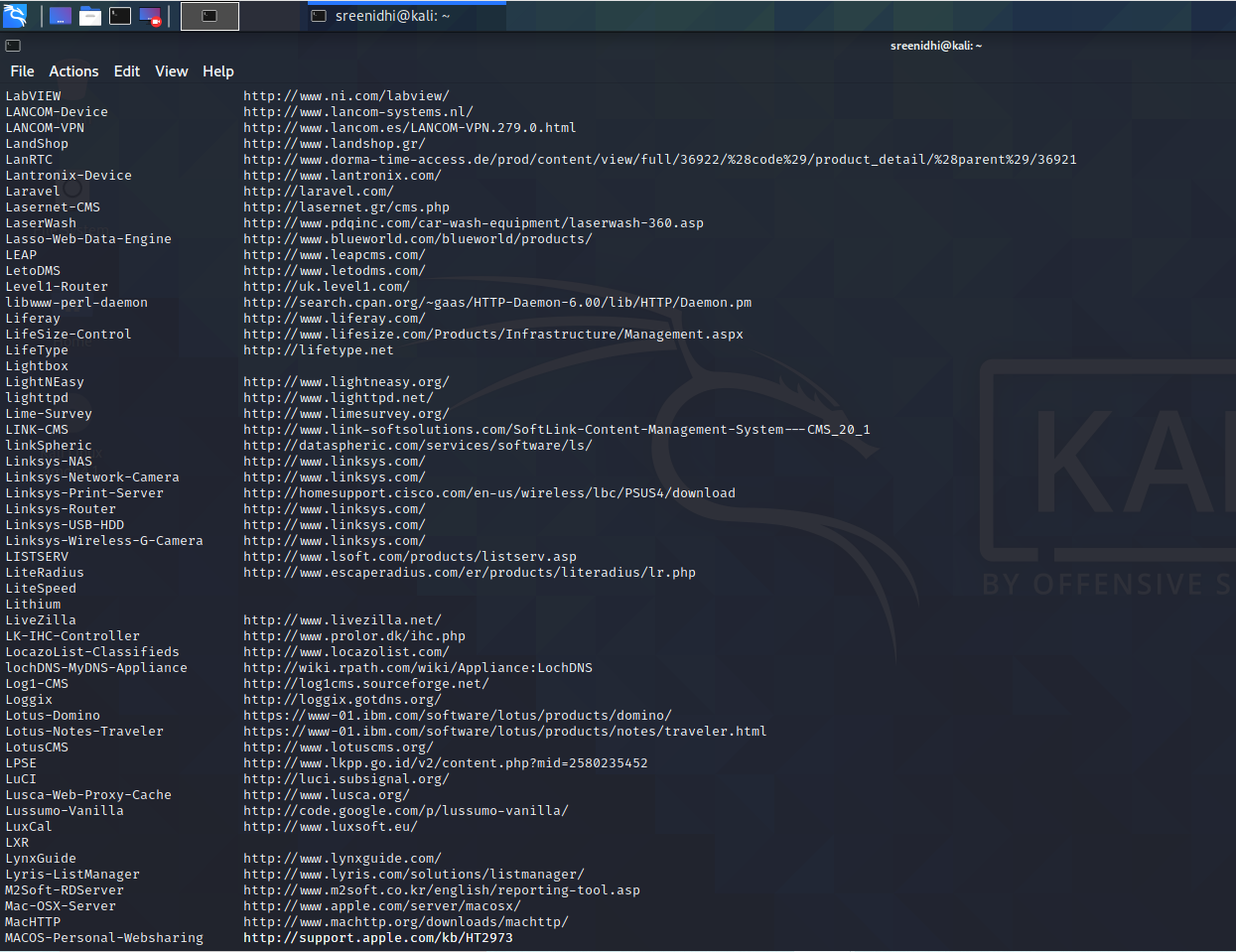
4)



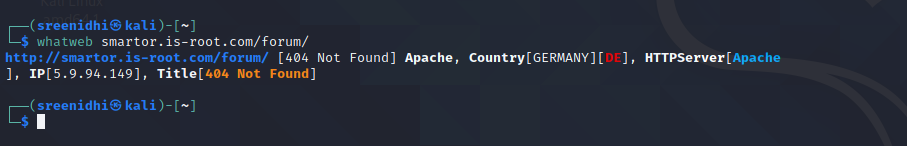
5)



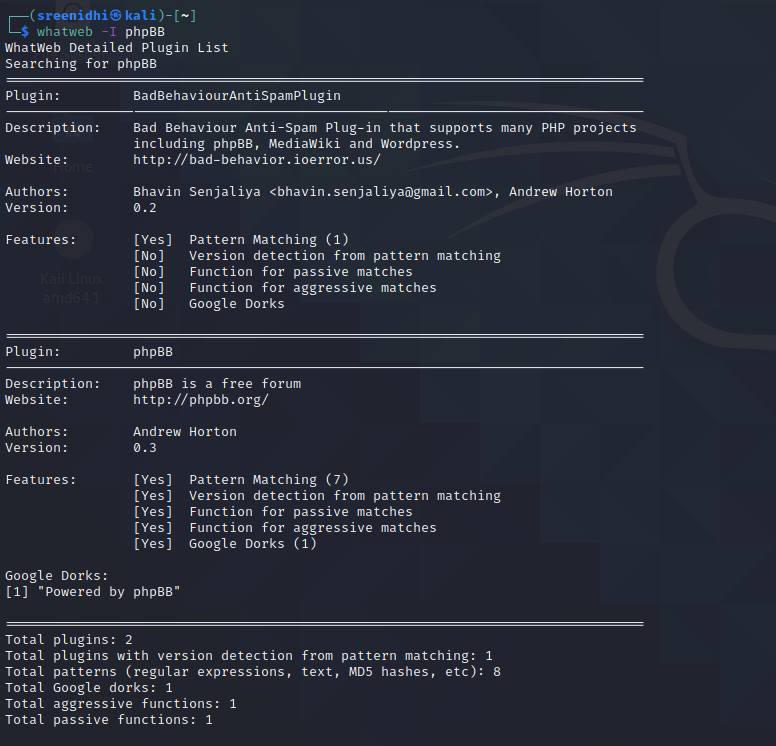
6) whatweb -l



7)

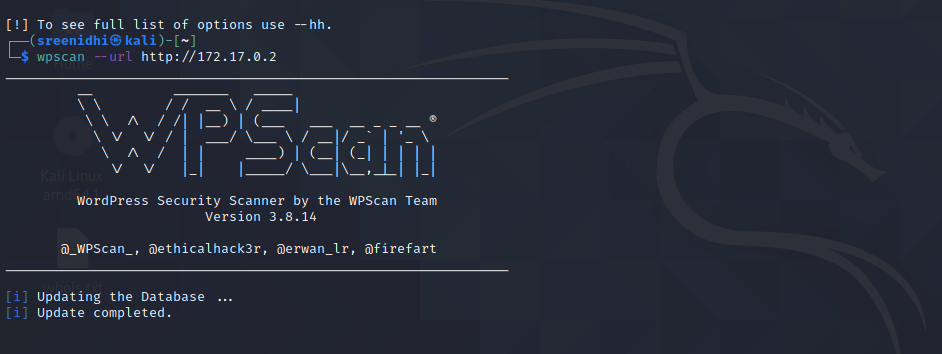


8) A detailed list of plugins

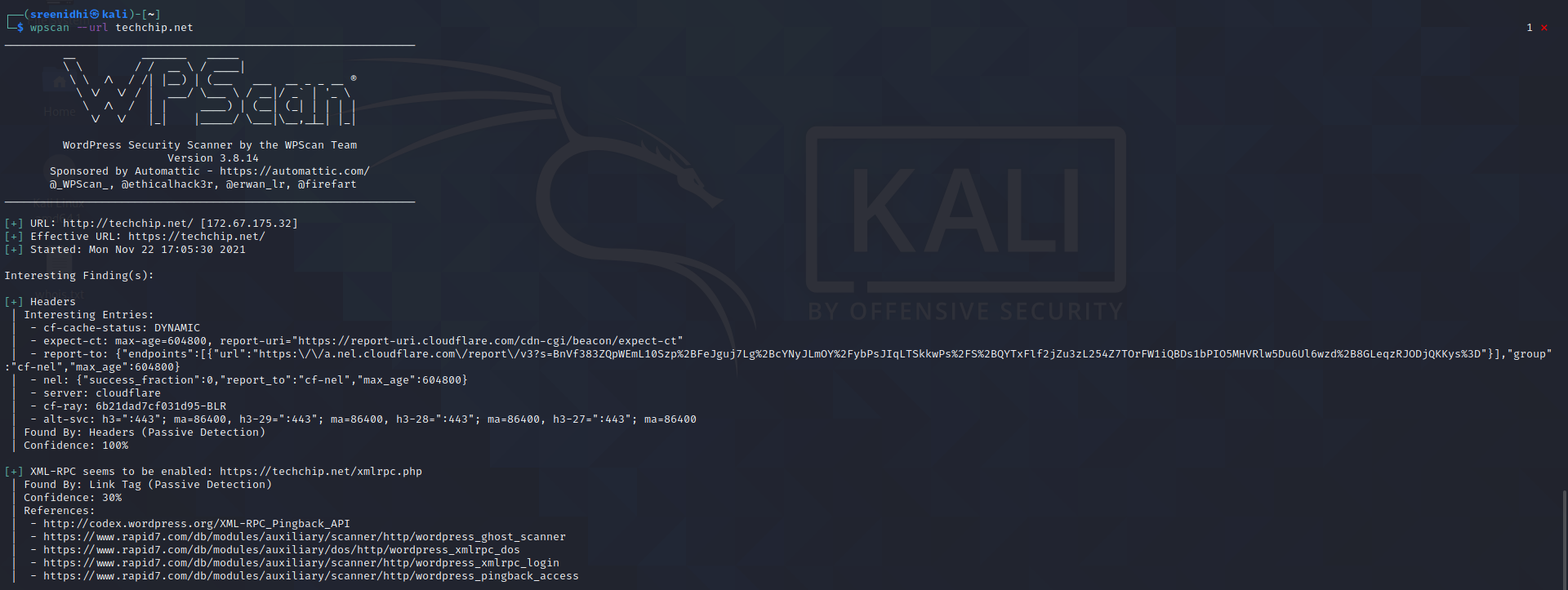


**1.3 WPScan**

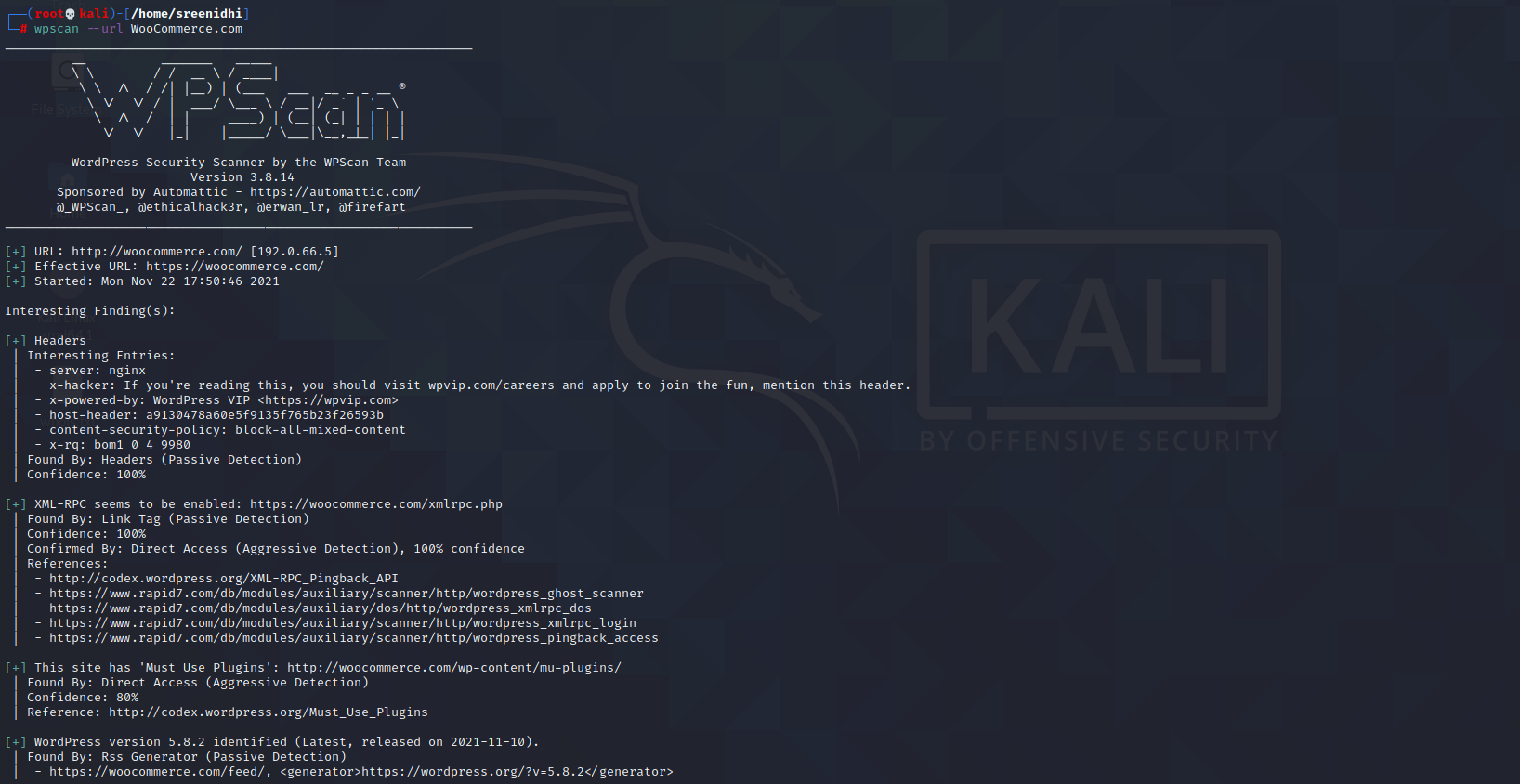
Vulnerabilities in WordPress can be uncovered by the WPScan utility, which comes installed by default in Kali Linux. There are numerous WordPress vulnerability scanners in the market like WordPress Security Scan, SUCURI, Detectify but WPScan is the scanner to scan your WordPress websites for vulnerable themes, plugins and security misconfigurations. . . WPScan is an all in one tool for scanning vulnerabilities in websites built using WordPress framework. It can be used to enumerate WordPress plugins and themes, brute-force logins and identify security misconfigurations. Currently. it is available only for Linux (Debian, Fedora, Arch, CentOS) and MacOSX, not for Windows. You can use Windows Subsystem for Linux (WSL) to install WPScan in Windows. In this tutorial, we’ll look at how to install and use WPScan to find security loopholes in your website.

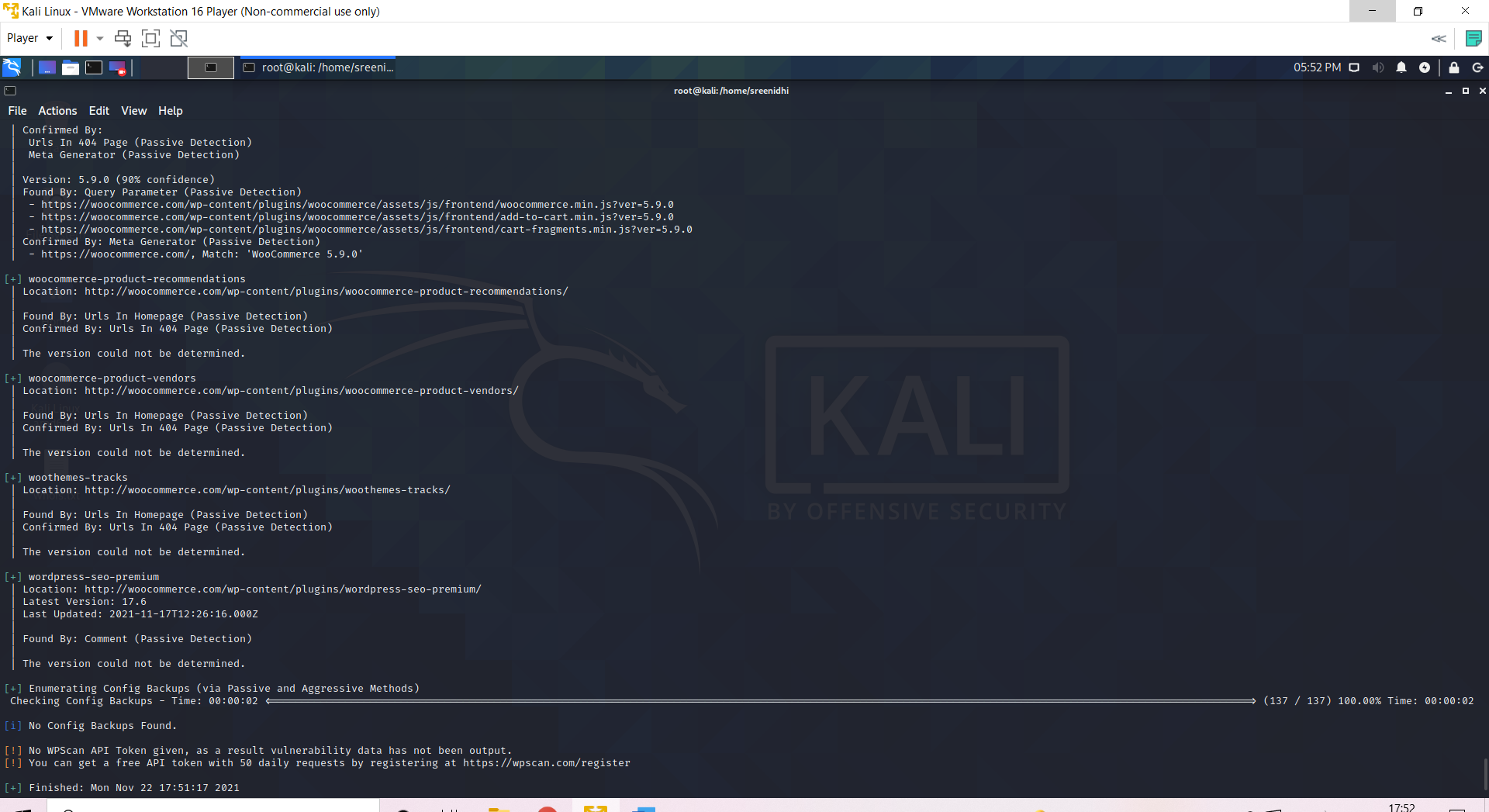


1) Using the wpscan to check for vulnerabilities in websites that use wordpress .

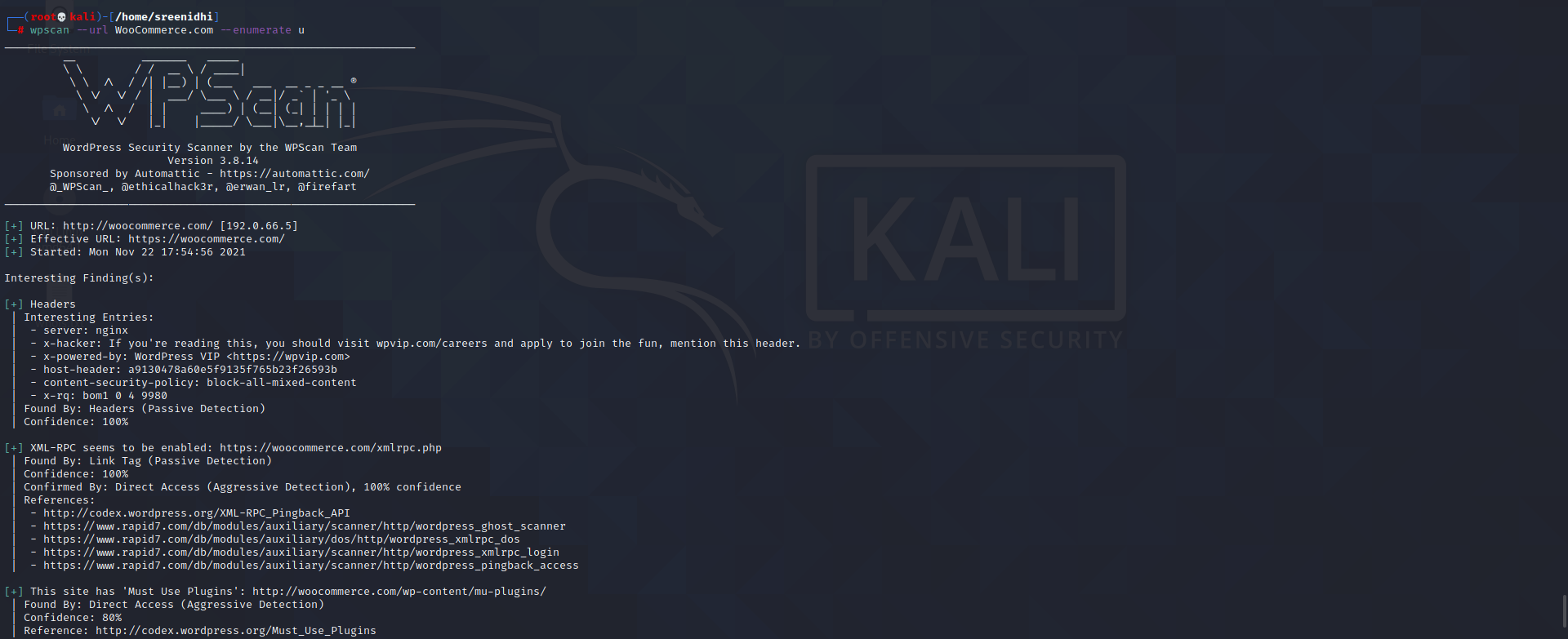








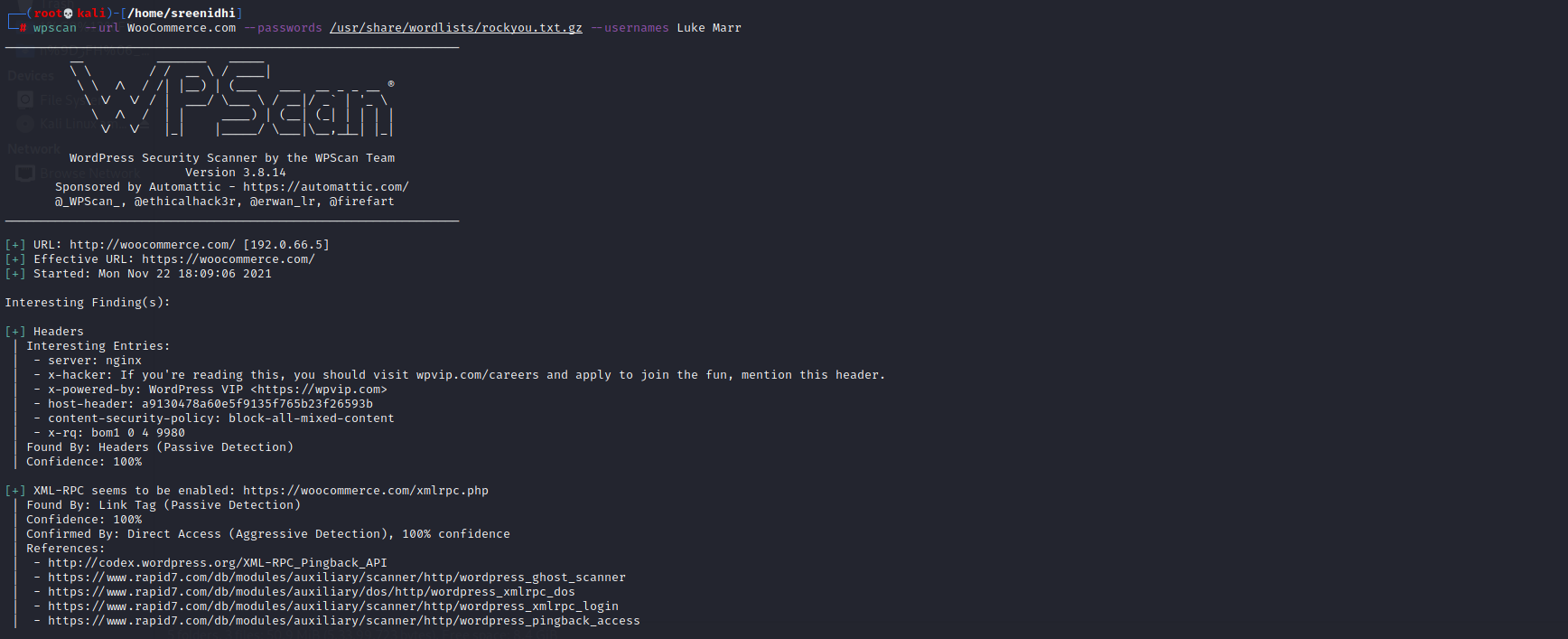
2) Here , we use the wpscan to detect the users of that particular website .

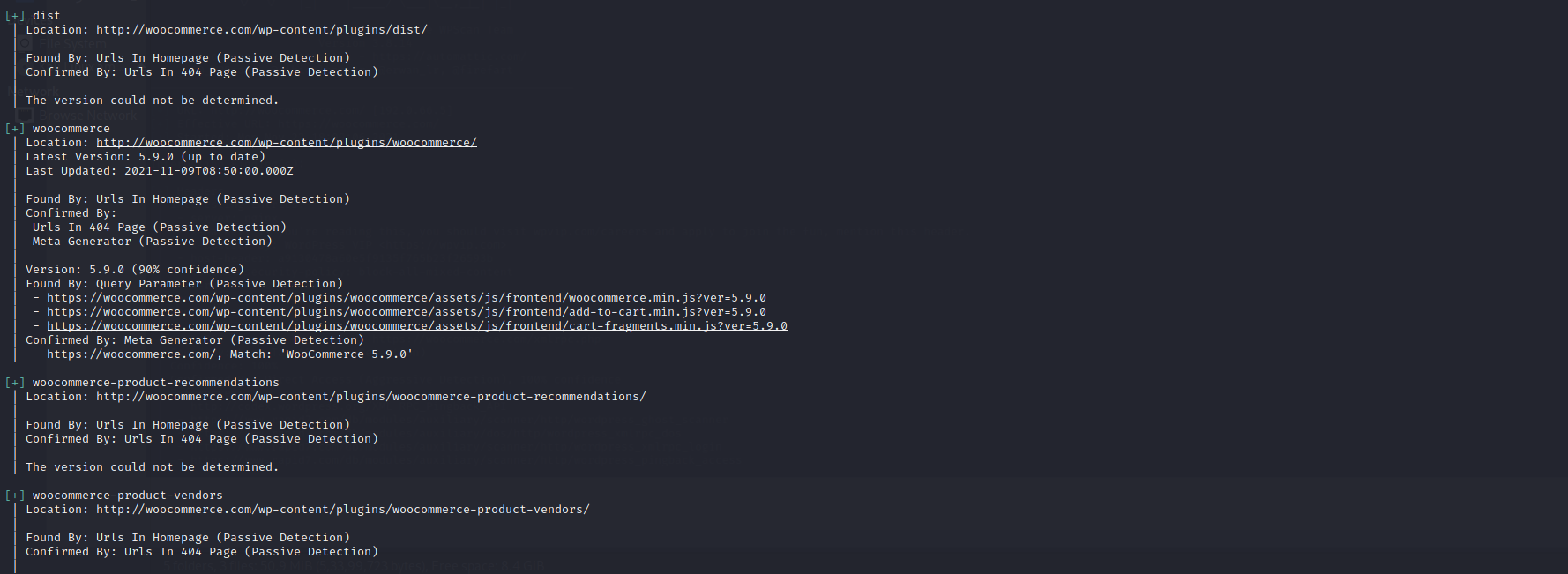


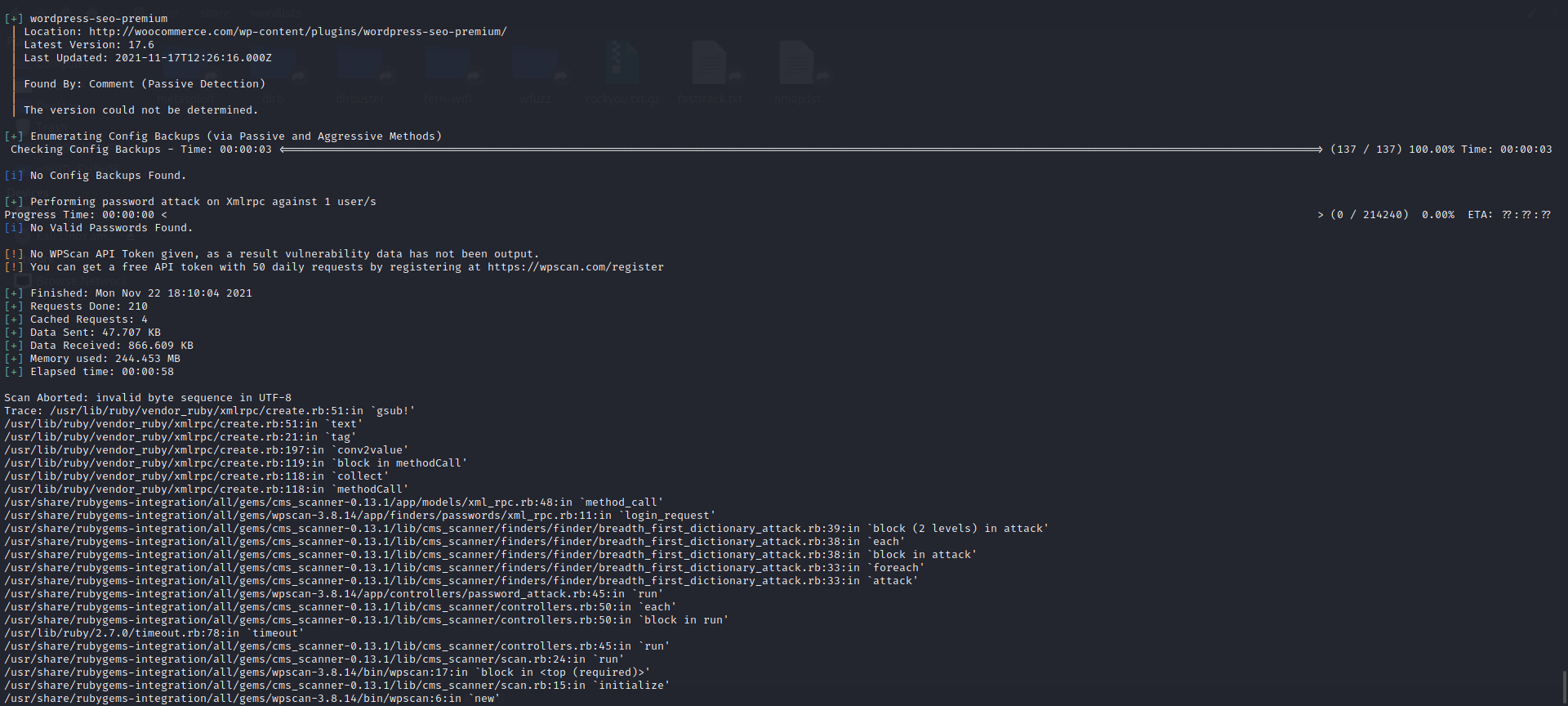
3) This website displays the users and the type of detection used .



4) We try to match the username with the wordlist in the system and crack the password of that particular user .







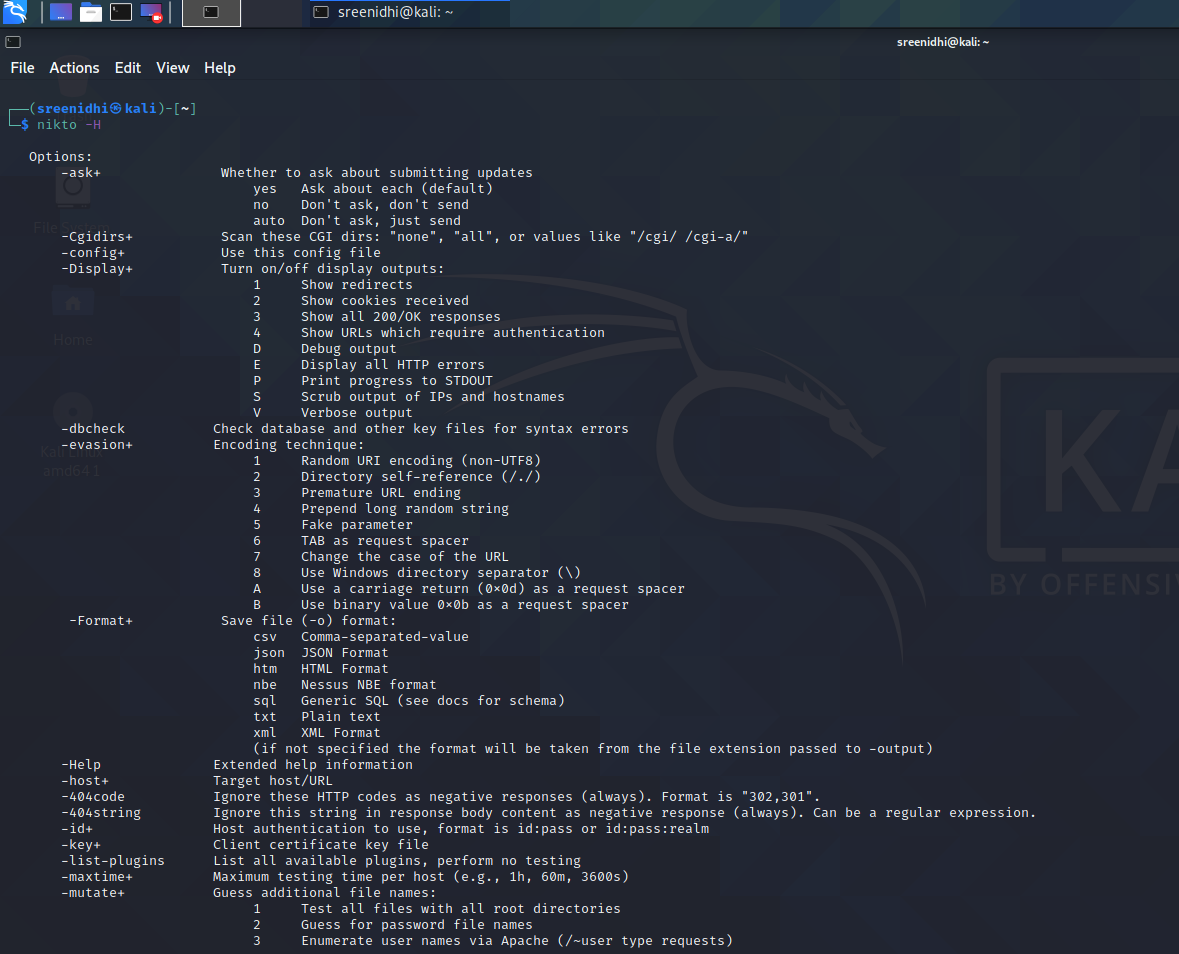
**1.4 Nikto**

Nikto is a pluggable web server and CGI scanner written in Perl, using rfp’s LibWhisker to perform fast security or informational checks. It is an open-source, free, and easy-to-use tool that helps you to find potential problems and vulnerabilities very quickly. Nikto checks for the presence of multiple index files, HTTP server options, and identifies installed web servers and software. Using Nikto allows you to test a web server very quickly.

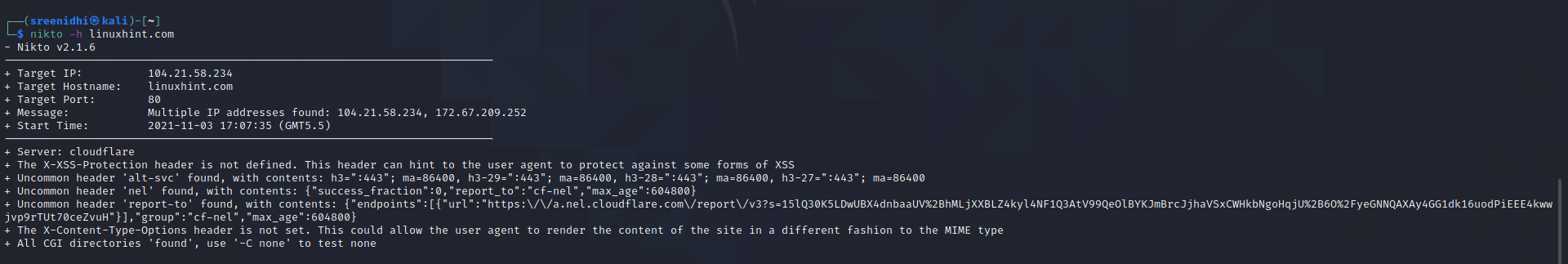
**Features:**

* Easily updatable CSV-format checks database
* Output reports in plain text or HTML
* Available HTTP versions automatic switching
* Generic as well as specific server software checks
* SSL support (through libnet-ssleay-perl)
* Proxy support (with authentication)
* Cookies support

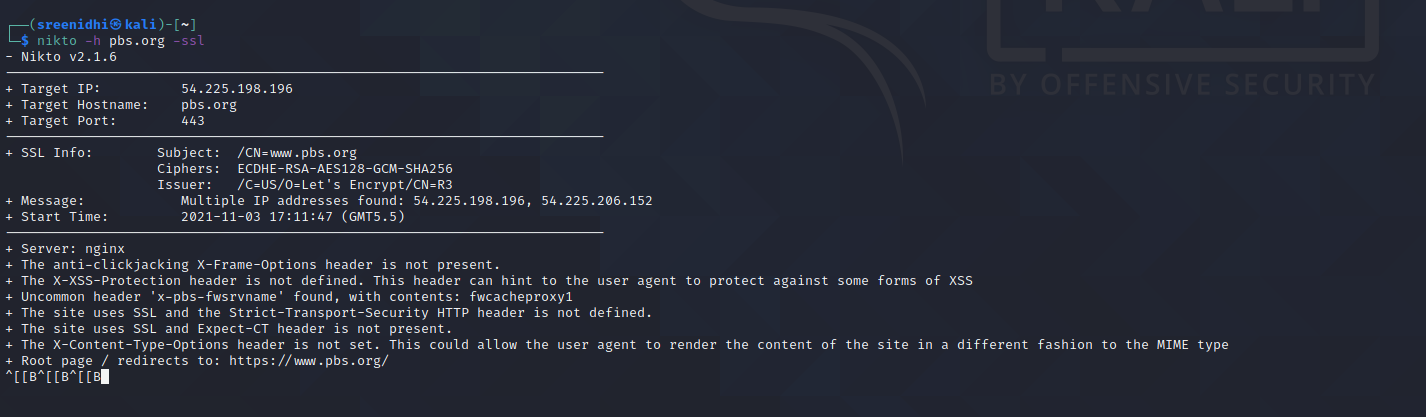
1) The nikto -h command provides us with the various commands available in this tool .



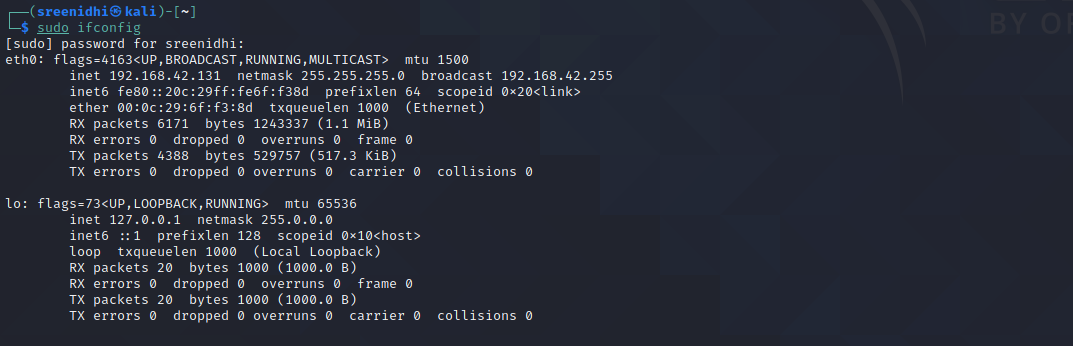
2) Finding the vulnerabilities in HTML pages of the website . After execution we come to know the website header is vulnerable to XSS attack and the user/ developer must protect it against XSS attack .



3) Here below we are looking at the secure server (ssl – secure socket layer ) details of a particular website .



4)



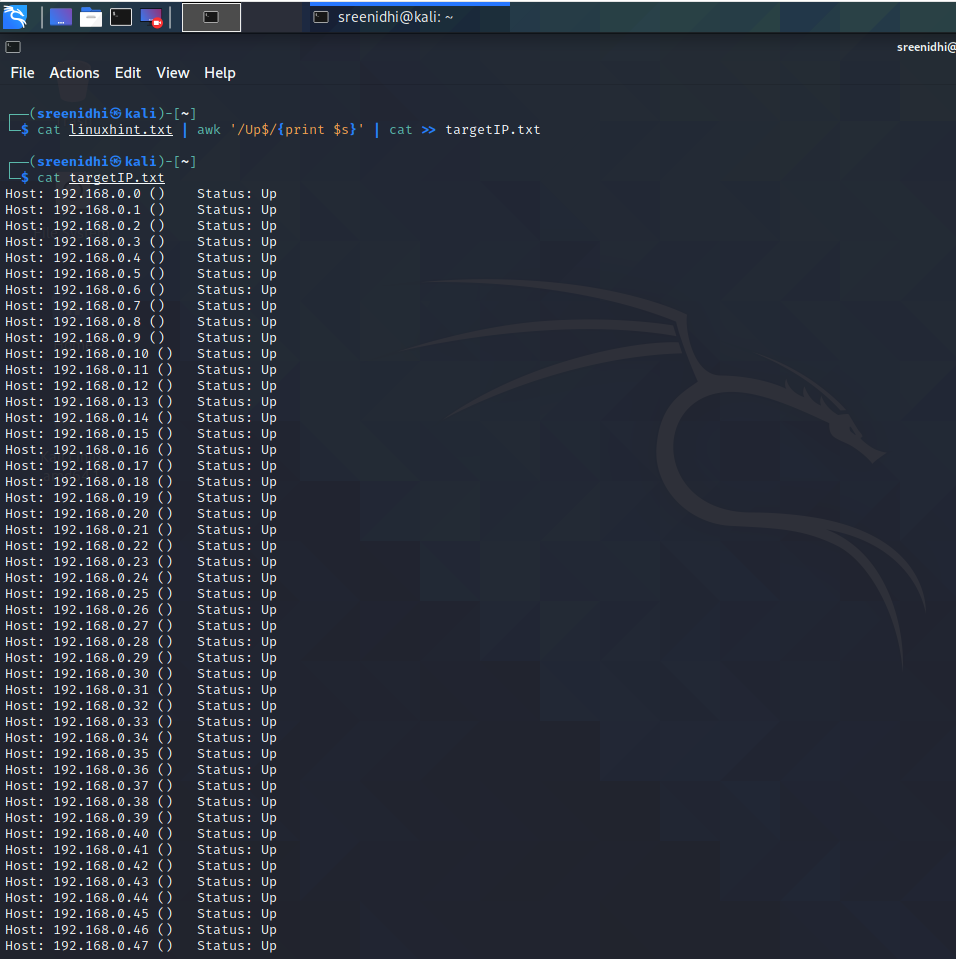
5)





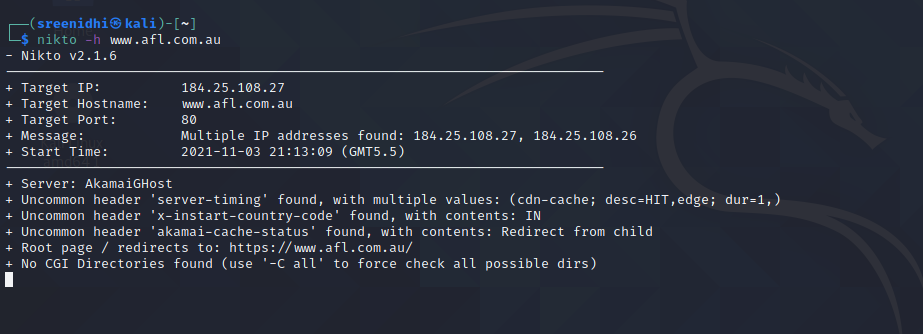
6)







7) Finding the vulnerabilities in HTML pages of the website



**CONCLUSION**

In this project we see various web application analysis tools that have been quite useful to understand the vulnerabilities that any web application is prone to . We try to detect these vulnerabilities and thus make the websites secure . These tools have been quite useful in the recent past as they let us know the various points the websites can be attacked .

After finding out the vulnerabilities , we can use other tools in kali linux to make our websites / systems more secure .